Standards Compliance according to IEC 61131-3

1. Introduction:

The **IEC 61131** standard is applicable for the programmable logic controllers (PLC).
In accordance with the rules of the European Union, this international standard has been accepted in Germany as DIN EN 6 1131, in France as NF EN 6 1131, and in England as BS EN 6 1131.
The most important parts of the standard are quoted below. Quotes are in italics.

Part 3 of this standard defines the application area in Section 1.1:
" This Part of IEC 1131 applies to the printed and displayed representation, using characters of the ISO 646 character set, of the programming languages to be used for Programmable Controllers as defined in Part 1 of IEC 1131. Graphic and semigraphic representation of the language elements which are defined in this Part is allowed, but is not defined in this Part. The functions of program entry, testing, monitoring, operating system, etc., are specified in Part 1 of IEC 1131."

Section 1.4 explains the overview and general requirements.
" This Part of IEC 1131 specifies the syntax and semantics of a unified suite of programming languages for programmable controllers (PCs). These consist of two textual languages, IL (Instruction List) and ST (Structured Text), and two graphical languages, LD (Ladder Diagram) and FBD (Function Block Diagram).
Sequential Function Chart (SFC) elements are defined for structuring the internal organization of programmable controller programs and function blocks. Also, configuration elements are defined which support the installation of programmable controller programs into programmable controller systems.
The programming language elements defined in this part may be used in an interactive programming environment. The specification of such environments is beyond the scope of this Part; however, such an environment shall be capable of producing textual or graphic program documentation in the formats specified in this part."

Section 1.5 of the standards fulfillment specifies:
" A programmable controller system, as defined in IEC 1131-1, which claims to comply, wholly or partially, with the requirements of this Part of IEC 1131 shall do so only as described below. A compliance statement shall be included in the documentation accompanying the system, or shall be produced by the system itself. The form of the compliance statement shall be:
"This system complies with the requirements of IEC 1131-3, for the following language features:", followed by a set of compliance tables in the following format:" (see tables).

2. Standards Compliance in STEP 7

The <u>SIMATIC STEP 7</u> system complies with the requirements of IEC 1131-3 for the following programming languages

- **Statement List   AWL/STL**   **(corresponds to the IEC 1131-3 language "STL/IL")**
- **Ladder Logic   KOP/LAD**   **(corresponds to the IEC 1131-3 language "LAD/LD")**
- **Function Block Diagram FUP/FBD**   **(corresponds to the IEC 1131-3 language "FUP/FBD")**
- **Structured Control Language SCL**   **(corresponds to the IEC 1131-3 language "ST")**
- **S7-GRAPH**   **(corresponds to the IEC 1131-3 language "AS/SFC")**

in the characteristics described in the following Chapter 4 and in the appendices.

<u>**SIMATIC STEP 7 Lite**</u> **only offers the programming languages STL/LAD/FBD. In accordance with this, the following statements concerning the IEC languages ST / SFC do not apply to STEP 7 Lite.**

3. Substitutes and Additional Language Elements

In addition, the standard stipulates that

a) A standardized PLC system may not include any substitute or additional language element to attain a standardized characteristic.
b) All implementation-dependent parameters must be specified according to *Annex D*.
c) User errors from *Annex E* must be reported; (for a partial program check, reference must be made to incompleteness).
d) User errors during converting and/or during start-up must be reported, and appropriate measures must be specified or introduced.
e) All characteristics not permissible or not present in the standard must be described as "expansions."
f) These expansions must be treated in the same way as is specified for errors (as a test that can be used as an option).
g) All implementation-independent characteristics from *Appendix D* must be handled as is specified for errors (as a test that can be used as an option*).*
h) No standardized names with meanings that vary can be used for manufacturer-defined characteristics.
i) The formal syntax of the text languages is described according to *Annex A*.

The STEP 7 programming software fulfills the requirements of the standard in points b), c), d), e), h), i). In respect to a), there exist language elements for compatibility reasons with STEP 5, which might be taken for additional elements. The f) and g) requirements are not used for STEP 7.

4.   Elements Realized According to the Standard

The standard defines all standardized language elements in tables, the rows of which reference the realized feature by number. The language elements which are realized in STEP 7 according to the standard are specified below.

(A good knowledge of the norm mentioned is a prerequisite for understanding the following tables. The German version of **DIN IEC 61131-3 : 1994-08** can be obtained from
    Beuth Verlag GmbH, 10772 Berlin, Germany, Fax (..30) 2601-1231.
The International Standard **IEC 1131-3 : First edition 1993-03** (English/French) can be obtained from
    Central Office of IEC, 3 rue de Varembe, Geneve, Switzerland.)

4.1 Common Elements

Table   No.   Language Elements

_____

1 Character set features
1   Required character set
2   Lower case characters
3a Number sign
3b Pound sign
4a Dollar sign
4b Currency sign
5a Vertical bar
5b Exclamation mark
6a Left and right brackets
6b Left and right parenthesis

Note: Windows character set, which includes these characters.
Identifiers are case-sensitive.

2 Identifier features

1   Upper case and numbers
2   Upper and lower case, numbers, embedded underlines
3   Upper and lower case, numbers

3 Comment features

1   Comment

Note: STL only line comments starting with // and ending with new line.

4 Numeric literals

1   Integer literals
2   Real literals
3   Real literals with exponents
4   Base 2 literals
5   Base 8 literals   (SCL only)
6   Base 16 literals *)
7   Boolean digits 0/1   (SCL only)
8   Boolean FALSE and TRUE

Note *): bit length required: W#16#ADAC, DW#16#ADAC_4711

5   1 Character string literals

Note: No print hex characters.

6 Two-character combinations in character strings
2   $$
3   $'
4   $L or $
5   $N
6   $P or $p
7   $R or $r
8   $T or $t

7 Duration literal features
1a without underlines: short prefix
1b long prefix
2a with underlines: short prefix
2b long prefix

Note: keywords are not case-sensitive.

8   Date and time of day literals
1   Date literals (long prefix)
2   Date literals (short prefix)
3   Time of day literals (long prefix)
4   Time of day literals (short prefix)
5   Date and time literals (long prefix)
6   Date and time literals (short prefix)

Note: keywords are not case-sensitive.

10   Keywords
1   BOOL
3   INT
4   DINT
10   REAL
12   TIME
13   DATE
14   TIME_OF_DAY or TOD
15   DATE_AND_TIME or DT
16   STRING *)
17   BYTE
18   WORD
19   DWORD

*): STRING n with length n; otherwise 254 bytes.

12   Data type declaration
4   Array data types
5   Structured data types   *)

*): data types have to be declared individually.

14   Data type initial value declaration features
4   Initialization of array data type elements
5   Initialization of structured data type elements

15   Memory Location and size prefix features for directly represented variables
1   I or E   depending in the language setting
2   Q or A   depending in the language setting
3   M
4   X *)
5   none
6   B
7   W
8   D

*): for DB and DI only.

16   Keywords for variable declarations
VAR, VAR_INPUT, VAR_OUTPUT, VAR_IN_OUT according to IEC

17   Assignment of types to variables
5   Automatic memory allocation of symbolic variables *) see note in table 10
6   Array declaration
8   Declaration of structured variables

18 **Assignment of initial values for variables**
    5   Initialization of symbolic variables *) see note in table 10
    6   Array initialization

20 **Use of EN input and ENO output**
    1   Use of "EN" and "ENO " with FBD
    2 Use "EN" and "ENO" with LAD

21 **Typed and overloaded functions**
    1   Overloaded functions   (SCL only)
    2   Typed functions

Note: Only 2 parameters with FDB/LAD; EN + ENO additional.

22 **Type conversion function features**
    1   *_TO_**
    2   TRUNC
    3   BCD_TO_**   (not for SCL)
    4   *_TO_BCD   (not for SCL)

23 **Standard functions of one numeric variable**
    1   ABS
    2   SQRT
    3   LN *)
    4   LOG   (SCL only)
    5   EXP
    6   SIN *)
    7   COS *)
    8   TAN *)
    9   ASIN *)
    10   ACOS *)
    11   ATAN *)

*): The implementation of these functions is CPU specific.
Note: The I/O type of the functions is REAL.

24 **Standard arithmetic functions**
    12   ADD +
    13   MUL *
    14   SUB -
    15   DIV /
    16   MOD
    18   MOVE :=

Note: All functions with FBD/LAD are typed (e.g. integer).

25 **Standard bit shift functions**
    1   SHL
    2   SHR
    3   ROR
    4   ROL

Note: All functions are typed (e.g. word).

26 **Standard bitwise Boolean functions**
  5   AND
  6   OR
  7   XOR *)
  8   NOT

  *): not for LAD

27 **Standard selection functions**
  1   SEL
  2a   MAX
  2b   MIN
  3   LIMIT

28 **Standard compare functions**
  5   GT >
  6   GE >=
  7   EQ =
  8   LE <=
  9   LT <
  10   NE <>

  Note: All functions are typed with FBD/LAD (e.g. Word).

29 **STRING data type functions**
  1   LEN
  2   LEFT
  3   RIGHT
  4   MID
  5   CONCAT
  6   INSERT
  7   DELETE
  8   REPLACE
  9   FIND

  Note: the implementation of these functions is CPU-specific.

30 **TIME data type functions**
  3   ADD_DT_T
  8   SUB_DT_T
  9   SUB_DT_DT
  12   CONTACT_D_TOD

  Note: the implementation of these functions is CPU-specific.

33 **Function block declaration features**
  4a   Input/output declaration (textual)
  4b   Input/output declaration (graphical)

34 **Standard bistable function blocks**
  1   SR
  2   RS

  Note: SR_FF is reset dominant; RS_FF is set dominant.

4.2 S7-GRAPH Elements (Sequential Function Chart, SFC)

Table   No.   Language Elements
40   **Step**
   1   Graphical form: step, initial step
   2   Textual form: *   step, initial step
   3a   Step flag - general form
   3b   Step flag - direct connection
   4   Step elapsed time
Note *): textual form: *.GR7

41   **Transitions and transition conditions**
   2   Condition in FBD
   5   Condition in textual form
   7d   Name

42   **Declaration of actions**
   1   Any Boolean variable

43   **Step/action association**
   1   Action block
   2   Concatenated actions
   3   Textual step body

44   **Action block features**
   1   "a": qualifier
   2   "b": action name

45   **Action qualifiers**
   2   N
   3   R
   4   S
   5   L
   6   D
Note: for detailed information see S7-Graph. S7-Graph has additional qualifiers for actions with conditions.

46   **Sequence evolution**
   1   Single sequence
   2a   Divergence of sequence selection
   3   Convergence of sequence selection
   4   Simultaneous sequences - divergence/convergence
   5a   Sequence skip
   6a   Sequence loop
   7   Directional arrows

4.3 Configuration Elements

50   1-5   **Task features**

In STEP 7, tasks are provided by organization blocks (OBs).

4.4 Instruction List Language (IL)

Remark for no. 52:

STL in STEP 7 differs essentially from the standard regarding the command syntax (see no. 52). This difference is necessary to meet the requirements concerning compatibility with STL in STEP 5.

In addition, STL in both STEP 5 and STEP 7 include far more commands than defined in the basic command set of the standard.

Table	No.	Language Elements

52   **Operators**
     1-21   Note: different syntax, same functionality.

53   **Function block invocation features for IL language**
     1-3   CAL with input list, with load/store of inputs, use of input operators
     3   Use of input operators *)

Note: different syntax, same functionality.

4.5 Language (Structured Text(, ST)

Table	No.	Language Elements

55   **Operators of the ST language**

No.	Operator	Symbol
1	Parenthesis	( )
2	Function evaluation	
3	Exponent**	
5	Complement	NOT
6	Multiply	*
7	Divide	/
8	Modulo	MOD
9	Add	+
10	Subtract	-
11	Comparison	<, >, <=, >=
12	Equality	=
13	Inequality	<>
14	Boolean AND	& AND
16	Boolean Exclusive OR	XOR
17	Boolean OR	OR

56   **ST language statements**

No.	Statement
1	Assignment
2	Function block invocation and FB output usage
3	RETURN
4	IF
5	CASE
6	FOR
7	WHILE
8	REPEAT
9	EXIT
10	Empty statement

4.6 Common Graphical Elements

Table	No.	Language Elements
57		Representation of lines and blocks
		Horizontal lines:
	1	ISO/IEC 646 "minus" character
	2	Graphic or semigraphic
		Vertical lines:
	3	ISO/IEC 646 "vertical line" character
	4	Graphic or semigraphic
		Horizontal/vertical connection:
	5	ISO/IEC 646 "plus" character
	6	Graphic or semigraphic
		Blocks with connecting lines:
	11	ISO/IEC 646 characters
	12	Graphic or semigraphic

58 Graphic execution control elements

	2	Unconditional jump FBD
	3	Conditional jump LAD
	4	Conditional jump FBD
	5	Conditional return FBD
	6	Conditional jump LAD
	7	Unconditional return

8
Note: in FBD represented as coils.

4.7 Ladder Diagram Language (LD)

Table	No.	Language Elements

59 Power rail symbols
| | 1 | Left power rail |
| | 2 | Right power rail |

60 Link element symbols
| | 1 | Horizontal link |
| | 2 | Vertical link |

61 Contacts
	1	Normally open contact
	3	Normally closed contact
	5	Positive transition-sensing contact
	7	Negative transition-sensing contact

62 Coils
	1	Coil
	2	Negated coil
	3	SET (latch) coil
	4	RESET (unlatch) coil

ANNEX A - Syntax

See manual for STL and SCL.

ANNEX D - Implementation-dependent Parameters

Below, you will find the parameters for the language elements defined in the standard and the limits realized in STEP 7.

IEC Reference	Parameters	STEP 7
1 General	Error handling procedures	see Annex E
2 Common elements	National characters used # or "pounds Sterling" sign $ or "currency" sign | or !	yes
	Maximum length of identifiers	24
	Maximum comment length	Network comment max. 64 KB Line comment max. 255 characters Address comment max. 40 characters
	Range of values of duration	- 24D_20H_31m_23.648s to 24D_20H_31m_23.647s
2.3 Data types	Range of values for variables of TIME type Precision of representation of seconds in TIME_OF_DAY and DATE_AND_TIME types	- 24D_20H_31m_23.648s 24D_20H_31m_23.647s milliseconds
	Maximum number of array subscripts Maximum array size Maximum number of structure elements Maximum structure size Maximum number of variables per declaration	6 max. 65536 Elements dependent on memory limitations, used data type. max. 64 KB of data possible approx. 4000 (estimated)
	Maximum number of enumerated values	not available

	Default maximum length of STRING variables	254
	Maximum allowed length of STRING variables	254
2.4 Variables	Maximum number of hierarchical levels	2
	Logical or physical mapping	physical mapping
	Maximum number of subscripts	6
	Maximum range of subscript values	65 536 entries
	Maximum number of levels of structures	8
	Initialization of system inputs	system : 0 user: definable initial values
	Maximum number of variables per declaration	approx. 4000 (estimated)
2.5 Program organization units	Information to determine execution times of program organization units	not available
	Method of function representation (names or symbols)	symbols
	Maximum number of function specifications	dependant on PLC 128 to 65536
	Maximum number of inputs of extensible functions	not available
	Effects of type conversions on accuracy	not available
	Accuracy of functions of one variable	IEEE Floating point
	Implementation of arithmetic functions	
	Maximum number of function	dependent on

	block specifications and instantiations	PLC 128 to 65536
	PVmin, PVmax of counters	-32568 to 32567
	Number/length limitations on SEND inputs and RCV outputs	
	Program size limitations	dependant on the PLC memory, from approx. 1000 instructions per block
2.6 Sequential function chart	Timing and portability effects of execution control elements	
	Precision of step elapsed time Maximum number of steps per SFC	1 ms 250
	Maximum number of transitions per SFC and per step	250
	Action control mechanism	all qualifiers except for "stored delayed"
	Maximum number of actions per step	100
	Graphic indication of step state Transition clearing time Maximum width of diverge/ converge constructs	colors selectable <10 µs 8 parallel / alternative branches in 8 chains (64)
2.7 Configuration elements	Contents of RESOURCE libraries	integrated functions / FB and FC
	Maximum number of tasks Task interval resolution Pre-emptive or non-pre-emptive scheduling	not available
3.3 Structured Text	Maximum length of expressions Partial evaluation of Boolean expressions	no restriction no

	Maximum length of statements	no restriction
	Maximum number of CASE selections	no restriction
	Value of control variable upon termination of FOR loop	end value +1
4 Graphic Languages	Graphic/semigraphic representation of restrictions on network topology	graphic
	Evaluation order of feedback loops	not available

ANNEX E - Error Conditions

Below, you will find the error conditions named in the standard and their occurrence in STEP 7.

Error conditions	IL (AWL) and LAD (KOP)	ST (SCL) only additions
Value of a variable exceeds the specified subrange	at run time time	at run time (compilation option)
Length of initialization list does not match number of array entries	at compilation	at compilation
Improper use of directly represented or external variables in functions	at compilation time	at compilation time
Type conversion errors	at compilation time	at compilation time
Numerical result exceeds range for data type Division by zero	at run time request a system flag	request ok flag
Mixed input data types to a selection function Selector (K) out of range for MUX function	at compilation time	at compilation time
Invalid character position specified Result exceeds maximum string length	at run time	at run time

Result exceeds range for data type	request of a system flag	request ok flag
Zero or more than one initial steps in SFC network User program attempts to modify step state or time	zero > Compiler message, up to 8 init steps possible, no error message when modifying step state or time	
Simultaneously true, non-prioritized transitions in a selection divergence	no non-prioritized transitions possible	
Side effects in evaluation of transition condition	no, compiler message	
Action control contention error	missing SD, therefore no message	
"Unsafe" or "unreachable" SFC	no message	
Data type conflict in VAR_ACCESS	not available	
Tasks require too many processor resources Execution deadline not met Other task scheduling conflicts	not available	
Numerical result exceeds range for data type	request of a system flag	request ok flag
Division by zero Invalid data type for operation	request of a system flag	request ok flag
Return from function without value assigned	at compilation time	at compilation time
Iteration fails to terminate	at run time	
Same identifier used as connector label and element name	not possible, compiler error message	

```
- - -
Un-initialized feedback variable                        I  not available       I
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -    |- - - - - - - - - - -   |- - - - - - - - - - - - - - - - - -
- - -
```