

Siemens S7 SCL cheat sheet

Conversion	Class A
BOOL_TO_BYTE	
BOOL_TO_DWORD	
BOOL_TO_WORD	
BYTE_TO_DWORD	
BYTE_TO_WORD	
CHAR_TO_STRING	
DINT_TO_REAL	
INT_TO_DINT	
INT_TO_REAL	
WORD_TO_DWORD	

Conversion	Class B
BOOL_TO_INT	
BOOL_TO_DINT	
BYTE_TO_BOOL	
BYTE_TO_CHAR	
BYTE_TO_INT	
BYTE_TO_DINT	
CHAR_TO_BYTE	
CHAR_TO_INT	
DATE_TO_DINT	
DINT_TO_DATE	
DINT_TO_DWORD	
DINT_TO_INT	
DINT_TO_TIME	
DINT_TO_TOD	
DINT_TO_BOOL	
DINT_TO_BYTE	
DINT_TO_STRING	
DINT_TO_WORD	
DWORD_TO_BOOL	
DWORD_TO_BYTE	
DWORD_TO_DINT	
DWORD_TO_REAL	
DWORD_TO_WORD	
DWORD_TO_INT	
INT_TO_CHAR	
INT_TO_WORD	
INT_TO_BOOL	
INT_TO_BYTE	
INT_TO_DWORD	
INT_TO_STRING	
REAL_TO_DINT	
REAL_TO_DWORD	
REAL_TO_INT	
REAL_TO_STRING	
STRING_TO_CHAR	
STRING_TO_INT	
STRING_TO_DINT	
STRING_TO_REAL	
TIME_TO_DINT	
TOD_TO_DINT	
WORD_TO_BOOL	
WORD_TO_BYTE	
WORD_TO_INT	
WORD_TO_DINT	
WORD_TO_BLOCK_DB	
BLOCK_DB_TO_WORD	
BCD_TO_INT(x)	
WORD_BCD_TO_INT(x)	
INT_TO_BCD(x)	
INT_TO_BCD_WORD(x)	
BCD_TO_DINT(x)	
DWORD_BCD_TO_DINT(x)	
DINT_TO_BCD(x)	
DINT_TO_BCD_DWORD(x)	

Numeric	
ABS	Number
SQR	Square
SQRT	Square Root
EXP	e to the power IN
EXPD	10 to the power IN
LN	Natural logarithm
LOG	Common logarithm
ACOS	Arc cosine
ASIN	Arc sine
ATAN	Arc tangent
COS	Cosine
SIN	Sine
TAN	Tangent

Bit String Functions	
ROL	Rotate left by N
ROR	Rotate right by N
SHL	Shift left by N
SHR	Shift right by N

Processing string	
LEN	
CONCAT	
LEFT or RIGHT	
MID	
INSERT	
DELETE	
REPLACE	
FIND	
EQ_STRNG and NE_STRNG	
GE_STRNG and LE_STRNG	
GT_STRNG and LT_STRNG	
INT_TO_STRING and STRING_TO_INT	
DINT_TO_STRING and STRING_TO_DINT	
REAL_TO_STRING and STRING_TO_REAL	

Selecting Values	
SEL	Select
MAX	Maximum
MIN	Minimum
LIMIT	Limit
MUX	Select from N

Counters	
S_CU	Count up
S_CD	Count down
S_CUD	Count up/down

Timers	
S_PULSE	pulse timer
S_PEXT	extended pulse timer
S_ODT	on-delay timer
S_ODTS	retentive on-delay timer
S_OFFDT	off-delay timer

Conversion	
ROUND	3.56 -> 4 (even)
TRUNC	3.56 -> 3

Constants	
BOOL	FALSE TRUE BOOL#0 BOOL#1
BYTE	B#16#00 B#16#FF BYTE#0 B#2#101 Byte#'a' b#16#f
CHAR	'A' CHAR#49
STRING	'Address'
WORD	W#16#0000 W#16#FFFF word#16#f WORD#8#177777 8#177777 W#2#1001_0100 WORD#32768
DWORD	DW#16#0000_0000 DW#16#FFFF_FFFF Dword#8#3777777777 8#3777777777 DW#2#1111_0000_1111_0000 dword#32768
INT	-32768 +32767 INT#16#3f_ff int#-32768 Int#2#1111_0000 inT#8#77777
DINT	-2147483648 +2147483647 DINT#16#3fff_ffff dint#-65000 Dint#2#1111_0000 dinT#8#1777777777
REAL	Decimal format 123.4567 REAL#1 real#1.5 Exponential format real#2e4 +1.234567E+02
S5TIME	T#0ms TIME#2h46m30s T#0.0s TIME#24.855134d
TIME	-T#24d20h31m23s647ms TIME#24d20h31m23s647ms T#0.0s TIME#24.855134d
Date	D#1990-01-01 DATE#2168-12-31
TIME_OF_DAY	TOD#00:00:00 TIME_OF_DAY#23:59:59.999
DATE_AND_TIME	DT#95-01-01-12:12:12.2

Declaration Section	
Constants	CONST...END_CONST
Labels	LABEL...END_LABEL
TemporaryVariables	VAR_TEMP...END_VAR
Static variables	VAR...END_VAR
Input parameters	VAR_INPUT...END_VAR
Output parameters	VAR_OUTPUT...END_VAR
In/out parameters	VAR_IN_OUT...END_VAR

Address Identifiers		
Mnemonic (internat.)	addresses	Data Type
Qx.y	Ouptut (via the process image)	Bit
QBx	Ouptut (via the process image)	Byte
QDx	Ouptut (via the process image)	Double word
QWx	Ouptut (via the process image)	Word
QXx.y	Ouptut (via the process image)	Bit
Dx.y	Data block	Bit
DBx	Data block	Byte
DDx	Data block	Double word
DWx	Data block	Word
DXx.y	Data block	Bit
Ix.y	Input (via the process image)	Bit
IBx	Input (via the process image)	Byte
IDx	Input (via the process image)	Double word
IWx	Input (via the process image)	Word
IXx.y	Input (via the process image)	Bit
Mx.y	Memory bit	Bit
MBx.y	Bit memory	Byte
MDx	Bit memory	Double word
MWx	Bit memory	Word
MXx	Bit memory	Bit
PQBx	Output (Direct to peripherals)	Byte
PQDx	Output (Direct to peripherals)	Double word
PQWx	Output (Direct to peripherals)	Word
PIBx	Input (Direct from peripherals)	Byte
PIDx	Input (Direct from peripherals)	Double word
PIWx	Input (Direct from peripherals)	Word

x = number between 0 and 65535 (absolute address)
y = number between 0 and 7 (bit number)

Operations	Symbol	Precedence
Assignment Operation:		
:	=	Assignment
Arithmetic Operations:		
**		Power
+		Unary plus
-		Unary minus
*		Multiplication
/		Division
MOD		Modulo function
DIV		Integer division
+		Addition
-		Subtraction
Comparison Operations:		
<		Less than
>		Greater than
<=		Less than or equal to
>=		Greater than or equal to
=		Equal to
<>		Not equal to
Logical Operations:		
NOT		Negation
AND or &		And
XOR		Exclusive or
OR		Or
Parentheses :		
()		Parentheses

Control Statements	
IF...THEN...ELSIF...THEN...ELSE...END_IF	provides one or more options and selects one (or none) of its statement components for execution
CASE...OF.....ELSE...END_CASE	select one of several alternative program sections
FOR...TO...BY...DO...END_FOR	repeat a sequence of statements as long as a control variable is within the specified range of values
WHILE...DO...END_WHILE	repeat execution of a sequence of statements controlled by an execution condition
REPEAT...UNTIL...END_REPEAT	repeated execution of a sequence of statements between REPEAT and UNTIL until a terminate condition occurs
CONTINUE	terminate the execution of the current iteration of a loop statement (FOR, WHILE or REPEAT).
EXIT	exit a loop (FOR, WHILE or REPEAT)

Elementary Data Types		
Bit Data Types	BOOL BYTE WORD DWORD	Date elements of this type occupy either 1 bit, 8 bits, 16 bits or 32 bits
Character Types	CHAR	Data elements of this type occupy exactly 1 character in the ASCII character set
Numeric Types	INT DINT REAL	Data elements of this type are available for processing numeric values.
Time Types	TIME DATE TIME_OF_DAY S5TIME	Data elements of this type represent the various time and date values in STEP 7.
Complex Data Types		
	DATE_AND_TIME DT	Defines an area of 64 bits (8 bytes). This data type stores date and time (as a binary coded decimal) and is a predefined data type in S7-SCL.
	STRING	Defines an area for a character string of up to 254 characters (data type CHAR).
	ARRAY	Defines an array consisting of elements of one data type (either elementary or complex).
	STRUCT	Defines a group of data types in any combination of types. It can be an array of structures or a structure consisting of structures and arrays.
Parameter Types		
	TIMER	declare timer functions as parameters.
	COUNTER	declare counter functions as parameters.
	BLOCK_xx	declare FCs, FBs, DBs and SDBs as parameters.
	ANY	allow an address of any data type as a parameter.
	POINTER	allow a memory area as a parameter.