

# Siemens S7 Statement List (STL)

*sorted alphabetically*

Mnemonic	Description
)	Nesting Closed
+	Add Integer Constant (16, 32-Bit)
+AR1	Add ACC1 to Address Register 1
+AR2	Add ACC1 to Address Register 2
+I +D +R	Add ACC1 and ACC2
-I -D -R	Subtract ACC1 from ACC2
*I *D *R	Multiply ACC1 and ACC2
/I /D /R	Divide ACC2 by ACC1
=	Assign
==I ==D	ACC2 is equal to ACC1
==R	ACC2 is less then equal to ACC1
<=I <=D	ACC2 is less then equal to ACC1
<=R	ACC2 is not equal to ACC1
<>I <>D	ACC2 is not equal to ACC1
<>R	ACC2 is less then to ACC1
<I <D <R	ACC2 is less then to ACC1
>=I >=D	ACC2 is greater then equal to ACC1
>=R	ACC2 is greater then to ACC1
>I >D >R	ACC2 is greater then to ACC1
A	And
A(	And with Nesting Open
ABS	Absolute Value
ACOS	Arc Cosine
AD	AND Double Word
AN	And Not
AN(	And Not with Nesting Open
ASIN	Arc Sine
ATAN	Arc Tangent
AW	AND Word
BE	Block End
BEC	Block End Conditional
BEU	Block End Unconditional
BLD	Program Display Instruction (Null)
BTD	BCD to Integer

Mnemonic	Description
BTI	BCD to Integer
CAD	Change Byte Sequence in ACC1 Double
CALL	Call FC, FB, SFC, SFB
CAR	Exchange Address Register 1 with Address Register 2
CAW	Change Byte Sequence in ACC1 Word
CC	Conditional Call
CD	Counter Down
CDB	Exchange Shared DB and Instance DB
CLR	Clear RLO (=0)
COS	Cosine of Angles
CU	Counter Up
DEC	Decrement ACC
DTB	Double Integer to BCD
DTR	Double Integer to Floating-Point
ENT	Enter ACC Stack
EXP	Exponential Value
FN	Edge Negative
FP	Edge Positive
FR	Enable Timer/Counter (Free)
INC	Increment ACC
INVD	Ones Complement Double Integer
INVI	Ones Complement Integer
ITB	Integer to BCD
ITD	Integer to Double Integer
JBI	Jump if BR = 1
JC	Jump if RLO = 1
JCB	Jump if RLO = 1 with BR
JCN	Jump if RLO = 0
JL	Jump to Labels
JM	Jump if Minus
JMZ	Jump if Minus or Zero
JN	Jump if Not Zero
JNB	Jump if RLO = 0 with BR
JNBI	Jump if BR = 0

Mnemonic	Description
JO	Jump if OV = 1
JOS	Jump if OS = 1
JP	Jump if Plus
JPZ	Jump if Plus or Zero
JU	Jump Unconditional
JUO	Jump if Unordered
JZ	Jump if Zero
L	Load
L	Load Current Timer/Counter Value into ACC1 as Integer (i.e. L T 32)
L DBLG	Load Length of Shared DB in ACC1
L DBNO	Load Number of Shared DB in ACC1
L DILG	Load Length of Instance DB in ACC1
L DINO	Load Number of Instance DB in ACC1
L STW	Load Status Word into ACC1
LAR1	Load Address Register 1 from ACC1
LAR1 <D>	Load Address Register 1 with Double Integer (32-Bit Pointer)
LAR1 AR2	Load Address Register 1 from Address Register 2
LAR2	Load Address Register 2 from ACC1
LAR2 <D>	Load Address Register 2 with Double Integer (32-Bit Pointer)
LC	Load Current Timer/Counter Value into ACC1 as BCD (i.e. LC T 32)
LEAVE	Leave ACC Stack
LN	Natural Logarithm
LOOP	Loop

Mnemonic	Description
<b>MCR (</b>	Save RLO in MCR Stack, Begin MCR
<b>)MCR</b>	End MCR
<b>MCRA</b>	Activate MCR
<b>MCRD</b>	Deactivate MCR
<b>MOD</b>	Division Remainder Double Integer
<b>NEGD</b>	Twos Complement Double Integer
<b>NEGI</b>	Twos Complement Integer
<b>NEGR</b>	Negate Floating-Point Number
<b>NOP 0</b>	Null Instruction
<b>NOP 1</b>	Null Instruction
<b>NOT</b>	Negate RLO
<b>O</b>	Or
<b>O (</b>	Or with Nesting Open
<b>OD</b>	OR Double Word
<b>ON</b>	Or Not
<b>ON (</b>	Or Not with Nesting Open
<b>OPN</b>	Open a Data Block
<b>OW</b>	OR Word
<b>POP</b>	Pop accumulators
<b>PUSH</b>	Push accumulators
<b>R</b>	Reset
<b>R</b>	Reset Timer/Counter Value (i.e. R T 32)
<b>RLD</b>	Rotate Left Double Word
<b>RLDA</b>	Rotate ACC1 Left via CC 1
<b>RND</b>	Round
<b>RND-</b>	Round to Lower Double Integer
<b>RND+</b>	Round to Upper Double Integer
<b>RRD</b>	Rotate Right Double Word
<b>RRDA</b>	Rotate ACC1 Right via CC 1
<b>S</b>	Set
<b>S</b>	Set Counter Preset Value (i.e. S C 15)
<b>SAVE</b>	Save RLO in BR Register
<b>SD</b>	On-Delay Timer
<b>SE</b>	Extended Pulse Timer

Mnemonic	Description
<b>SET</b>	Set RLO (=1)
<b>SF</b>	Off-Delay Timer
<b>SIN</b>	Sine of Angles
<b>SLD</b>	Shift Left Double Word
<b>SLW</b>	Shift Left Word
<b>SP</b>	Pulse Timer
<b>SQR</b>	Square
<b>SQRT</b>	Square Root
<b>SRD</b>	Shift Right Double Word
<b>SRW</b>	Shift Right Word
<b>SS</b>	Retentive On-Delay Timer
<b>SSD</b>	Shift Sign Double Integer
<b>SSI</b>	Shift Sign Integer
<b>T</b>	Transfer
<b>T STW</b>	Transfer ACC1 into Status Word
<b>TAK</b>	Toggle ACC1 with ACC2
<b>TAN</b>	Tangent of Angles
<b>TAR1</b>	Transfer Address Register 1 to ACC1
<b>TAR1 &lt;D&gt;</b>	Transfer Address Register 1 to Destination (32-Bit Pointer)
<b>TAR1 AR2</b>	Transfer Address Register 1 to Address Register 2
<b>TAR2</b>	Transfer Address Register 2 to ACC1
<b>TAR2 &lt;D&gt;</b>	Transfer Address Register 2 to Destination (32-Bit Pointer)
<b>TRUNC</b>	Truncate
<b>UC</b>	Unconditional Call
<b>X</b>	Exclusive Or
<b>X (</b>	Exclusive Or with Nesting Open
<b>XN</b>	Exclusive Or Not
<b>XN (</b>	Exclusive Or Not with Nesting Open
<b>XOD</b>	Exclusive Or Double Word
<b>XOW</b>	Exclusive Or Word

Formats	
<b>B#</b>	Byte (8 bit)
<b>W#</b>	Word (16 bit)
<b>L#</b>	Long (32 bit)
<b>S5Time#</b>	S5 Time (2H46M30S0MS)
<b>T#</b>	IEC Time (24D20H31M23S648MS)
<b>D#</b>	IEC Date (2007-10-28)
<b>TOD#</b>	Time of Day (23:59:59.999)
<b>C#</b>	BCD
<b>P#</b>	Pointer Address
<b>2#</b>	Binary
<b>16#</b>	Hexadecimal
<b>#Symbol</b>	Local stack variable
<b>//</b>	Comment

OBs	
<b>1</b>	Main Program Scan
<b>10-17</b>	Time of Day
<b>20-23</b>	Time Delay
<b>30-38</b>	Cyclic (Periodic)
<b>40-47</b>	Hardware
<b>80</b>	Time Error
<b>81</b>	Power Supply Error
<b>82</b>	Diagnostic Interrupt
<b>83</b>	Insert/Remove Module Interrupt
<b>84</b>	CPU Hardware Fault
<b>85</b>	Program Cycle Error
<b>86</b>	Rack Failure - Missing Profibus device
<b>87</b>	Communication Error
<b>100</b>	Warm restart
<b>101</b>	Hot restart
<b>102</b>	Cold restart
<b>121</b>	Programming Error
<b>122</b>	I/O Access Error



[www.plcdev.com](http://www.plcdev.com)