

Data Manipulation and Math Instructions



Learning Outcome

Apply common instruction set such as

- Math
- Logic Word
- Move
- Compare



Common Math Instructions

Data manipulation and arithmetic are frequently used in automation

- Coverage in LAD, however other programming languages is possible with the same concept
- Data types and range would be important for us to note:
 - Range capability
 - Sign or unsigned
 - Would a decimal point precision be required (REAL)
 - Memory allocation (how many bytes to assign to the variable as not to overlap address)
- Data type might need to be converted from Double Integer to REAL before we would do the arithmetic

Format	Data type	Number of bits	Value range	Initial value
Integer (w/sign) ^a	<i>SINT</i>	8	-128 to + 127	0
	<i>INT</i>	16	-32 768 to + 32 767	0
	<i>DINT</i>	32	-2 ³¹ to + 2 ³¹ -1	0
	<i>LINT</i>	64	-2 ⁶³ to + 2 ⁶³ -1	0
Positive integer (unsigned) ^b	<i>USINT</i>	8	0 to 255	0
	<i>UINT</i>	16	0 to 65 535	0
	<i>UDINT</i>	32	0 to 2 ³² -1	0
	<i>ULINT</i>	64	0 to 2 ⁶⁴ -1	0
Floating-point numbers ^c	<i>REAL</i>	32	± 10 ^{±38}	0.0
	<i>LREAL</i>	64	± 10 ^{±308}	0.0

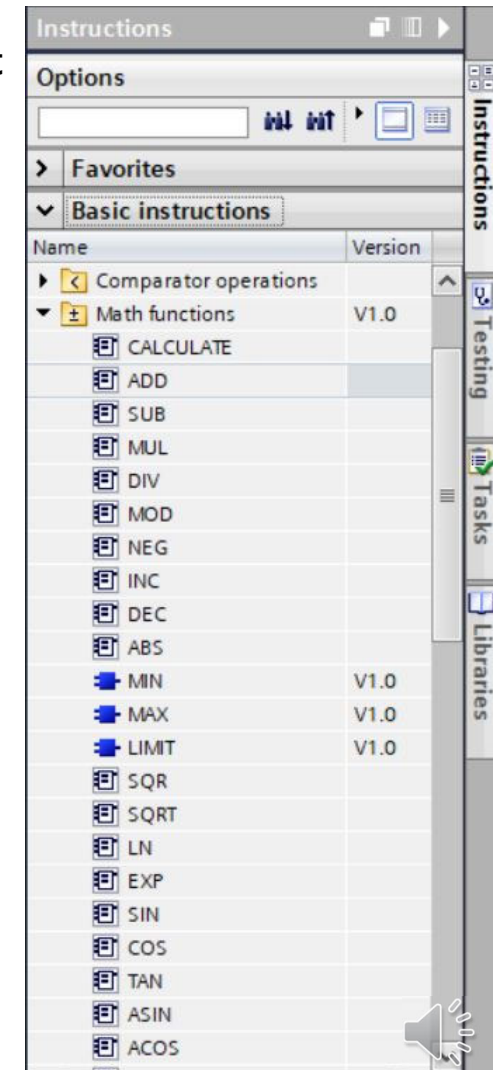
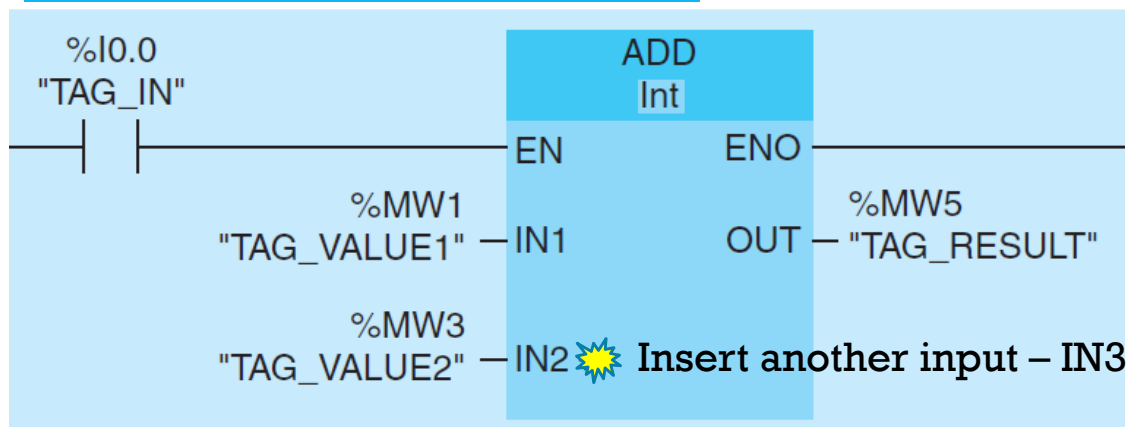
D, double; INT, integer; L, long; S, short; U, unsigned.

Mnemonics	Description	Data Type	Address Range
I/O Signals			
I	Input bit	BOOL	0.0–65535.7
IB	Input byte	BYTE, CHAR	0–65535
IW	Input word	WORD, INT, S5TIME, DATE	0–65534
ID	Input double word	DWORD, DINT, REAL, TOD, TIME	0–65532
Q	Output bit	BOOL	0.0–65535.7
QB	Output byte	BYTE, CHAR	0–65535
QW	Output word	WORD, INT, S5TIME, DATE	0–65534
QD	Output double word	DWORD, DINT, REAL, TOD, TIME	0–65532
Marker Memory			
M	Memory bit	BOOL	0.0–65535.7
MB	Memory byte	BYTE, CHAR	0–65535
MW	Memory word	WORD, INT, S5TIME, DATE	0–65534
MD	Memory double word	DWORD, DINT, REAL, TOD, TIME	0–65532



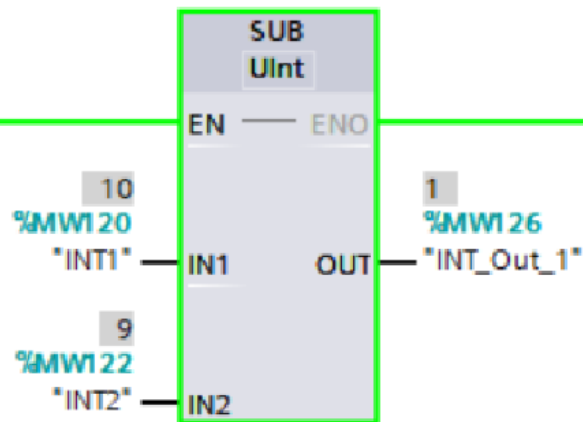
ADD

- The Math functions are functions (FC) that you need not declare instance
- EN – Enable
- ENO – Enable Output
- IF TAG_IN = 1
TAG_VALUE1 is added to TAG_VALUE_2,
Result is stored in TAG_RESULT
- Number of inputs could be extended 🌟

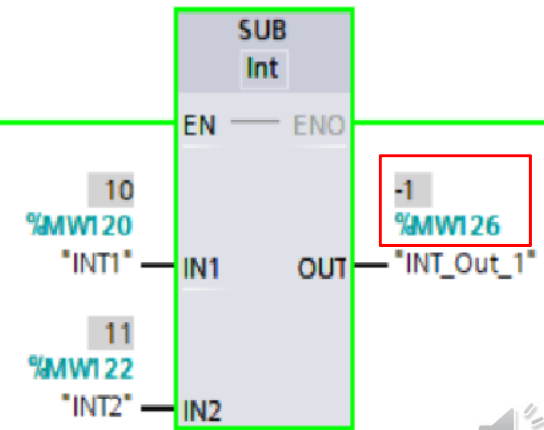
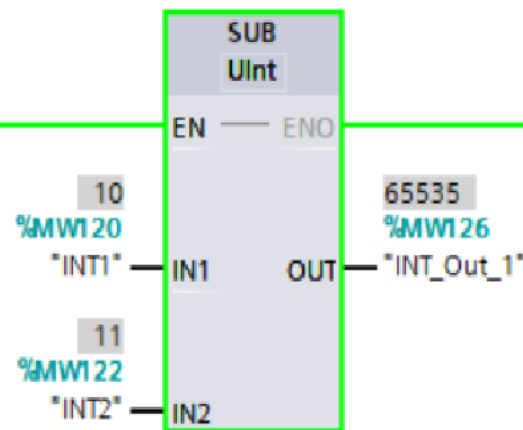


Subtraction

- If you do not need a condition to be true to execute the instruction, EN could be directly connected to the network
- INT2 would be subtracted from INT1 and the result stored in INT_Out_1
- Notice the result when INT2 is larger than INT1
- Result should be Negative but yield 65535 as it is an unsigned Integer. (Note that all variables are UINT data type)
- **OUT** need to be an **INT** type to reflect -1



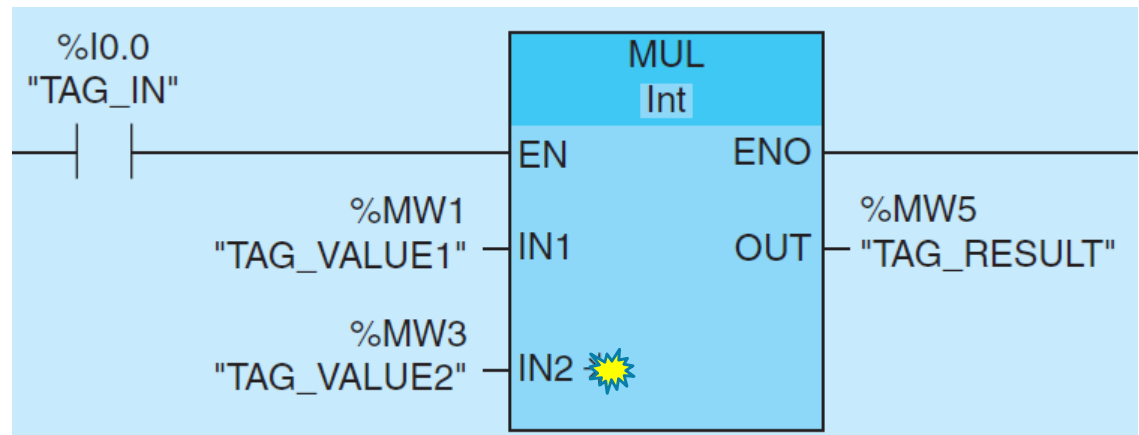
All variables are UINT data type



Changed to INT data type

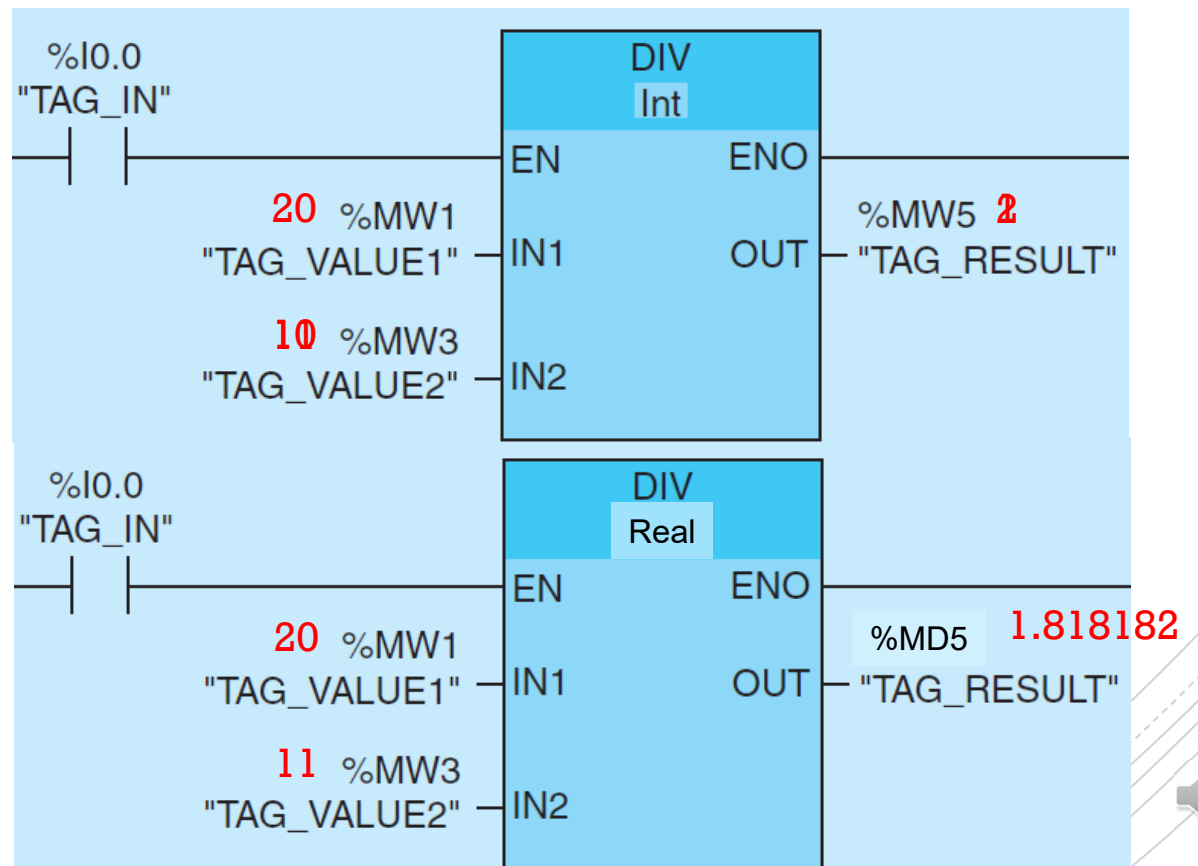
Multiply

- IF TAG_IN = 1
- TAG_VALUE1 is multiplied by TAG_VALUE_2, Result is stored in TAG_RESULT
- The number of inputs could be expanded 🌟



DIVIDE

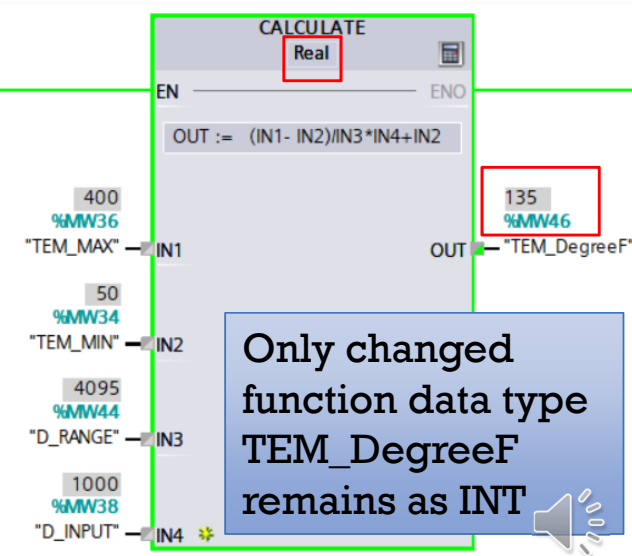
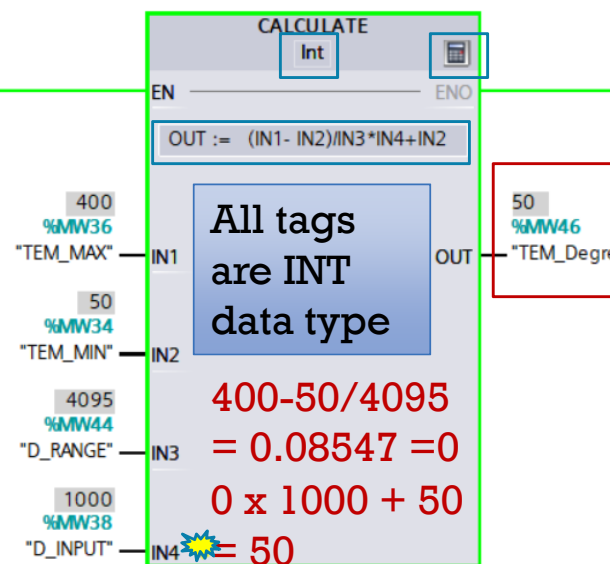
- IF TAG_IN = 1
- TAG_VALUE1 is divided by TAG_VALUE_2, Result is stored in TAG_RESULT



Calculation Function

Define and execute an expression for the calculation of mathematical operation or complex logic operation

- Select data type from "???" drop down list to suitable data type
- Enter expression via dialog containing input parameters and syntax of the instructions
 - **IN1, IN2** and any additional inputs ☀
 - Note: Operand names / addresses cannot be specified
- The result of the instruction is transferred to **OUT**



Quiz

Click the **Quiz** button to edit this object

Refer to the attached image. Why is "TAG_RESULT" 65535?
Select the correct answer option:

- ☐ TAG_RESULT is of UINT data type which is unsigned, it should be change to INT data type
- ☐ TAG_RESULT is of UINT data type which is unsigned, it should be change to REAL data type
- ☐ The SUB function shall be in REAL to produce correct result

