

1 Funzioni definite da LogicLab

LogicLab supporta in modo nativo una serie di funzioni che possono essere utilizzata dai vari programmi scritti nei linguaggi supportati. Per utilizzare queste funzioni occorre dalla libreria nella sezione **Operator and standard blocks** attingere la funzione desiderata. Nei linguaggi testuali IL ed ST è anche possibile scrivere il nome della funzione esattamente così come è definita.

Library

ABS

ACOS

+

?

&

ASIN

ATAN

ATAN2

CEIL

CONCAT

COS

COSH

DELETE

DIV

=

e^x

FIND

FLOOR

GE

GT

INSERT

JMP

LE

LEFT

LEN

LIMIT

ln

log

<

MAX

MID

MIN

MOD

MOVE

MUL

MUX

NE

NOT

OR

POW

R

REPLACE

RET

RIGHT

ROL

ROR

S

SEL

SHL

SHR

SIN

SINH

sizeof

SQRT

SUB

TAN

TANH

TO_BOOL

TO_DINT

TO_INT

TO_REAL

TO_SINT

TO_UDINT

TO_UINT

TO_USINT

XOR

Operator and standard blocks

Target variables

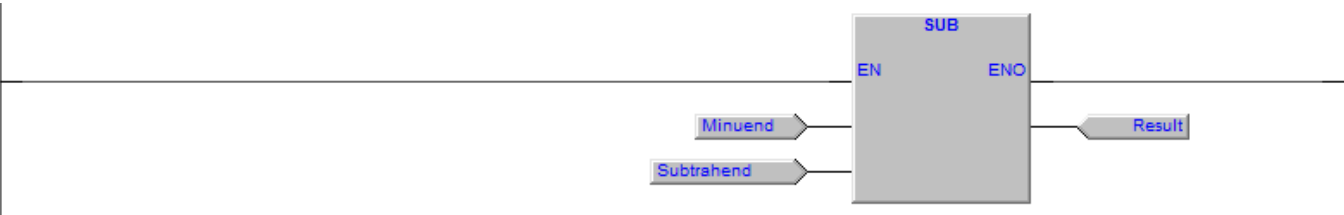
Target blocks

Ecco alcuni esempi di utilizzo nei vari linguaggi.

Definizione variabili

	Name	Type	Address	Array	Init value	Attribute	Description
1	Minuend	UINT	Auto	No	0	..	Minuend
2	Subtrahend	UINT	Auto	No	0	..	Subtrahend
3	Result	UINT	Auto	No	0	..	Result

Esempio LD



Esempio IL

```
LD Minuend
SUB Subtrahend
ST Result
```

Esempio ST

```
Result:=SUB(Minuend, Subtrahend); (* Subtraction using SUBB *)
Result:Minuend-Subtrahend; (* Subtraction as before *)
```

1.1 Funzioni matematiche e trigonometriche

LogicLab supporta tutte le funzioni matematiche richieste dalla norma CEI EN 61131-3 nella Parte 3: Linguaggi di programmazione. Riporto di seguito un elenco delle funzioni supportate ed un esempio di utilizzo in linguaggio ST.

ABS	ANY_NUM ABS(ANY_NUM) Calcola il valore assoluto di un numero. Result:=ABS(-10.5); (* Result is 10.5 *)
SQRT	ANY_REAL SQRT(ANY_REAL) Calcola la radice quadrata di un numero. Result:=SQRT(9.0); (* Result is 3.0 *)
LN	ANY_REAL LN(ANY_REAL) Calcola logaritmo naturale (Base "e" 2,71828) di un numero. Result:=LN(10.0); (* Result is 2.30259 *)
LOG	ANY_REAL LOG(ANY_REAL) Calcola logaritmo (Base "10") di un numero. Result:=LOG(10.0); (* Result is 1.0 *)
EXP	ANY_REAL EXP(ANY_REAL) Calcola numero elevato "e" 2,71828. Result:=EXP(1.0); (* Result is 2.71828 *)
SIN	ANY_REAL SIN(ANY_REAL) Calcola il seno di un angolo in radianti. Result:=SIN(1.57); (* Angle is 90°, Result is 1.0 *)
COS	ANY_REAL COS(ANY_REAL) Calcola il coseno di un angolo in radianti. Result:=COS(3.1416); (* Angle is 180°, Result is -1.0 *)
TAN	ANY_REAL TAN(ANY_REAL) Calcola la tangente di un angolo in radianti. Result:=TAN(0.7854); (* Angle is 45°, Result is 1.0 *)
ASIN	ANY_REAL ASIN(ANY_REAL) Calcola l'arcoseno di un angolo in radianti. Result:=ASIN(1.0); (* Result is: 1.5708 *)
ACOS	ANY_REAL ACOS(ANY_REAL) Calcola l'arcocoseno di un angolo in radianti. Result:=ACOS(-1.0); (* Result is 3.14159 *)
ATAN	ANY_REAL ATAN(ANY_REAL) Calcola l'arcotangente di un angolo in radianti. Result:=ATAN(1.0); (* Result is 3.14159 *)
ADD	ANY_NUM ADD(ANY_NUM, ANY_NUM) Esegue somma tra due numeri. Result:=ADD(1.0, 2.0); (* Result is 3.0 *)
MUL	ANY_NUM MUL(ANY_NUM, ANY_NUM) Esegue moltiplicazione tra due numeri. Result:=MUL(1.0, 2.0); (* Result is 2.0 *)
SUB	ANY_NUM SUB(ANY_NUM, ANY_NUM) Esegue sottrazione tra due numeri. Result:=SUB(2.0, 1.0); (* Result is 1.0 *)

DIV **ANY_NUM DIV(ANY_NUM, ANY_NUM)**
Esegue divisione tra due numeri.

`Result:=DIV(2.0, 1.0); (* Result is 2.0 *)`

Considerazione a parte v  fatta per l'operatore modulo **MOD**. L'aritmetica modulare si applica ai soli numeri interi, nel quale i numeri "si avvolgono su se stessi" ogni volta che raggiungono i multipli di un determinato numero n, detto modulo.

L'operazione `Result:=x MOD 10`, dar  come risultato valori compresi tra 0 e 10 per qualsiasi valore assuma x.
L'operazione `Result:=x MOD 1000`, dar  come risultato valori compresi tra 0 e 1000 per qualsiasi valore assuma x.

1.2 Funzioni stringa

LogicLab supporta tutte le funzioni di gestione stringa richieste dalla norma CEI EN 61131-3 nella Parte 3: Linguaggi di programmazione. Riporto di seguito un elenco delle funzioni supportate ed un esempio di utilizzo in linguaggio ST.

CONCAT	STRING CONCAT(STRING S0, STRING S1) Concatena le stringhe S0 ed S1. <pre> AString:='Hello '; BString:='World !'; CString:=CONCAT(AString, BString); (* CString is 'Hello World !' *) CString:=CONCAT(AString, 'World !'); (* CString is 'Hello World !' *) </pre>
DELETE	STRING DELETE(STRING IN, ANY_INT L, ANY_INT P) Cancella dalla stringa IN, L caratteri partendo dalla posizione P a scendere. <pre> AString:='Hello World !'; BString:=DELETE(AString, 2, 3); (* BString is 'Heo World !' *) </pre>
FIND	UINT FIND(STRING S0, STRING S1) Cerca la posizione della prima occorrenza di S1 in S0. Se non trovata torna 0. <pre> AString:='Hello World !'; i:=FIND(AString, 'World'); (* Result is: 7 *) j:=FIND('Hello World World !', 'World'); (* Result is 7 *) k:=FIND('World', 'Hello'); (* Result is: 0 *) </pre>
INSERT	STRING INSERT(STRING S0, STRING S1, ANY_INT P) Inserisce nella stringa S0 la stringa S1 partendo dalla posizione P. <pre> AString:='Hello everybody'; BString:='World !'; CString:=INSERT(AString, BString, 6); (* CString is 'Hello World !' *) </pre>
LEFT	STRING LEFT(STRING IN, ANY_INT L) Ritorna gli L caratteri più a sinistra della stringa IN. <pre> AString:='Hello World !'; BString:=LEFT(AString, 7); (* BString is 'Hello W' </pre>
LEN	UINT LEN(STRING IN) Ritorna la lunghezza (Numero di caratteri) della stringa IN. <pre> AString:='Hello World !'; i:=LEN(AString); (* i is: 13 *) </pre>
MID	STRING MID(STRING IN, ANY_INT L, ANY_INT P) Ritorna gli L caratteri della stringa IN partendo dal carattere in posizione P. <pre> AString:='Hello World !'; BString:=MID(AString, 5, 7); (* BString is 'World' *) </pre>
REPLACE	STRING REPLACE(STRING S0, STRING S1, ANY_INT L, ANY_INT P) Rimpiazza L caratteri della stringa S0 con la stringa S1 a partire dalla posizione P. <pre> AString:='Hello World !'; BString:=REPLACE(AString, 'to you ', 5, 7); </pre>
RIGHT	STRING RIGHT(STRING IN, ANY_INT L) Ritorna gli L caratteri più a destra della stringa IN. <pre> AString:='Hello World !'; BString:=RIGHT(AString, 7); (* BString is 'World !' *) </pre>