SELECTION OPERATORS

MAX

The IEC operator is used for the maximum function. It yields the greatest value of all inputs.

Syntax

```
OUT := MAX(IN0, IN1, <further inputs>)
```

Permitted data types: All

```
PROGRAM Selection_Operators
VAR
   rRoomTemp : REAL := 22.5; // Room temperature
   rExtTemp : REAL := 18.3;      // External temperature
   rMaxTemp : REAL;  // Maximum temperature
   iMaxAge : INT;  // Maximum age of the plant workers
END_VAR
```

```
// Get the highest temperature between the two
rMaxTemp 22.5  := MAX(rRoomTemp 22.5 , rExtTemp 18.3 );

// Result is  the  maximum age of the plant workers
iMaxAge 39  := MAX(32, 25, 36, 24, 18, 39,  30);
```

LIMIT

The IEC selection operator is used for limitation.

Syntax

```
OUT :=LIMIT(Min, IN, Max)
```

Permitted data types: All.

Input Value (IN)	Output (OUT)	Explanation
IN $<$ Min	OUT = Min	The value is below the lower limit.
Min $\leq$ IN $\leq$ Max	OUT = IN	The value is within the specified range.
IN $>$ Max	OUT = Max	The value exceeds the upper limit.

The LIMIT function ensures that the output remains within a defined range. The following table demonstrates how it behaves with different input values.

```
PROGRAM Selection_Operators
    iInPressure  :  INT;      // Raw pressure sensor value (input)
    iPressure  :  INT;              // Adjusted pressure value (output)
END_VAR
VAR CONSTANT
    cMinPressure : INT := 100;     // Minimum allowed pressure
    cMaxPressure : INT := 150;     // Maximum allowed pressure
END_VAR


iPressure100   := LIMIT(cMinPressure100 , iInPressure80 , cMaxPressure150 );
```

Input (iInPressure)	Min Limit (cMinPressure)	Max Limit (cMaxPressure)	Output (iPressure)
80	100	150	100
120	100	150	120
170	100	150	150

The LIMIT operator, conceptually is: OUT :=MIN(MAX(IN, Min), Max) .

MIN

The IEC operator is used for the minimum function. It yields the least value of all inputs.

Syntax

```
OUT := MIN(IN0, IN1, <further inputs>)
```

Permitted data types: All.

```
PROGRAM Selection_Operators
VAR
rRoomTemp : REAL := 22.5; // Room temperature
rExtTemp : REAL := 18.3;      // External temperature
rMinTemp : REAL;   // Minimum temperature
iMinAge : INT;  // Minimum age of the plant workers
END_VAR

// Get the lowest temperature between the two
rMinTemp 18.3  := MIN(rRoomTemp 22.5 , rExtTemp 18.3 );
```

```
// Result is  the  minimum age of the plant workers
iMinAge 18  := MIN(32, 25, 36, 24,  18,  39,  30);
// Result is  18,  the  minimum age of the plant workers
```

MUX

The IEC operator is used as a multiplexer.

Syntax

```
OUT := MUX(K, IN0, ..., INn)
```

This means OUT takes the value of IN_K, where K is the specified index.

Permitted data types:

- K: e BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, LINT, ULINT, UDINT.
- IN0, ..., INn and OUT: Any identical data type.

```
PROGRAM Selection_Operators
VAR
   nSelectedValue : INT;    // The selected value
   iK : INT := 2;           // The index (K), initialized with 2
END_VAR

// iK=2, so the selected value will be 30
nSelectedValue30 := MUX(iK2 , 10, 20, 30, 40, 50);
```

When the index `iK` is negative (e.g., `iK = −1`), the MUX operator selects the last value (IN4).

When `iK` exceeds the number of available inputs (e.g., `iK = 5` for a 5-input MUX), the operator will also select the last value (IN4).

(Index)	Values	(OUT)
iK	IN0, IN1, IN2, IN3, IN4	nSelectedValue
−iK	10, 20, 30, 40, 50	50
0	10, 20, 30, 40, 50	10
1	10, 20, 30, 40, 50	20
2	10, 20, 30, 40, 50	30
3	10, 20, 30, 40, 50	40
4	10, 20, 30, 40, 50	50
iK > 4	10, 20, 30, 40, 50	50

SEL

The IEC operator is used for bitwise selection.

Syntax

```
OUT := SEL(G, IN0, IN1)
```

Permitted data types:

- IN0, ..., INn and OUT: any identical data type.
- G: BOOL

G	OUT
FALSE	IN0
TRUE	IN1

```
VAR
   iVarSel  :  INT; // Result of  SEL
END_VAR


iVarSel  := SEL(FALSE, 3, 4);  (*  Result:  3  *)
iVarSel  := SEL(TRUE, 3, 4); (*  Result:  4  *)
```