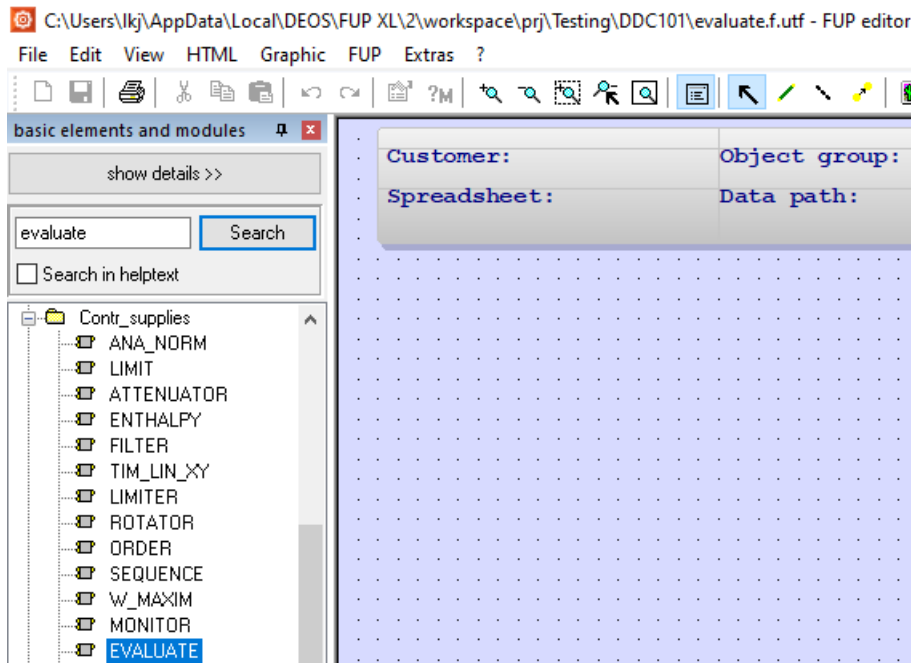# TT221203 – FUP - Evaluate Module

1. In this document, we will show you how to use the "EVALUATE" module. You can find the module under "Contr_supplies" in "Basic Elements and Modules". We will use it to do a simple sequencing function for 8 pumps, based on the runtime of the pumps.

2. Drag and drop the module to your FUP page. Double click on it and click the "Help Text" tab, and you will find a description of the module usage.
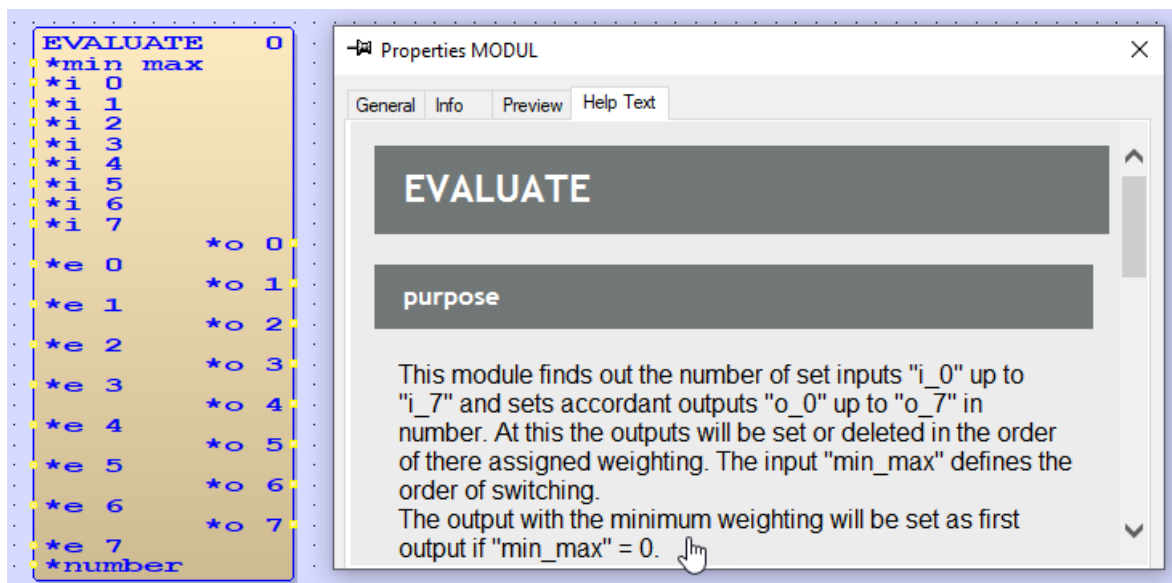
3. Click the "Preview" tab and you can see the input and output point types.
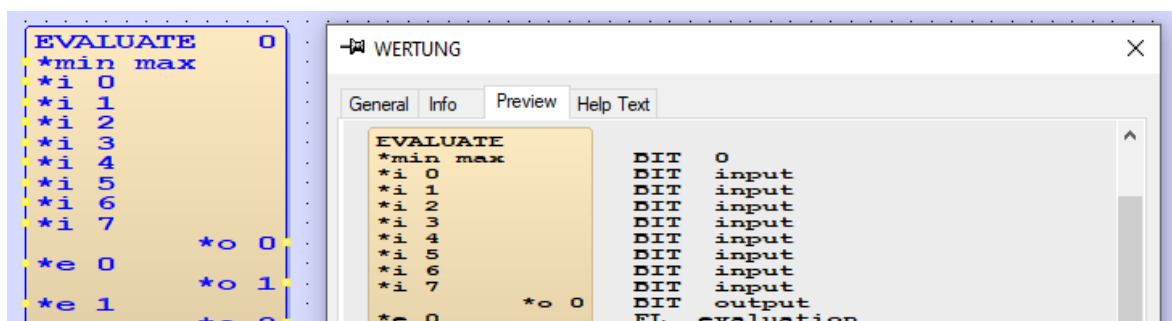
4. Connect the input and display and set the type accordingly.
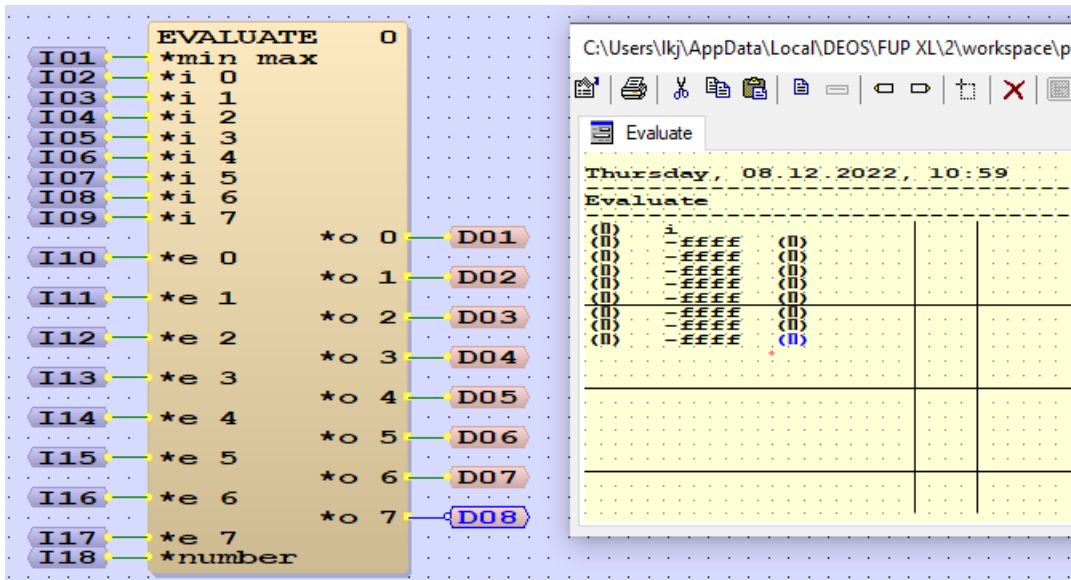
5. Start simulation test by clicking the "SIM" button. Enter the numbers like below and we're now ready for testing.
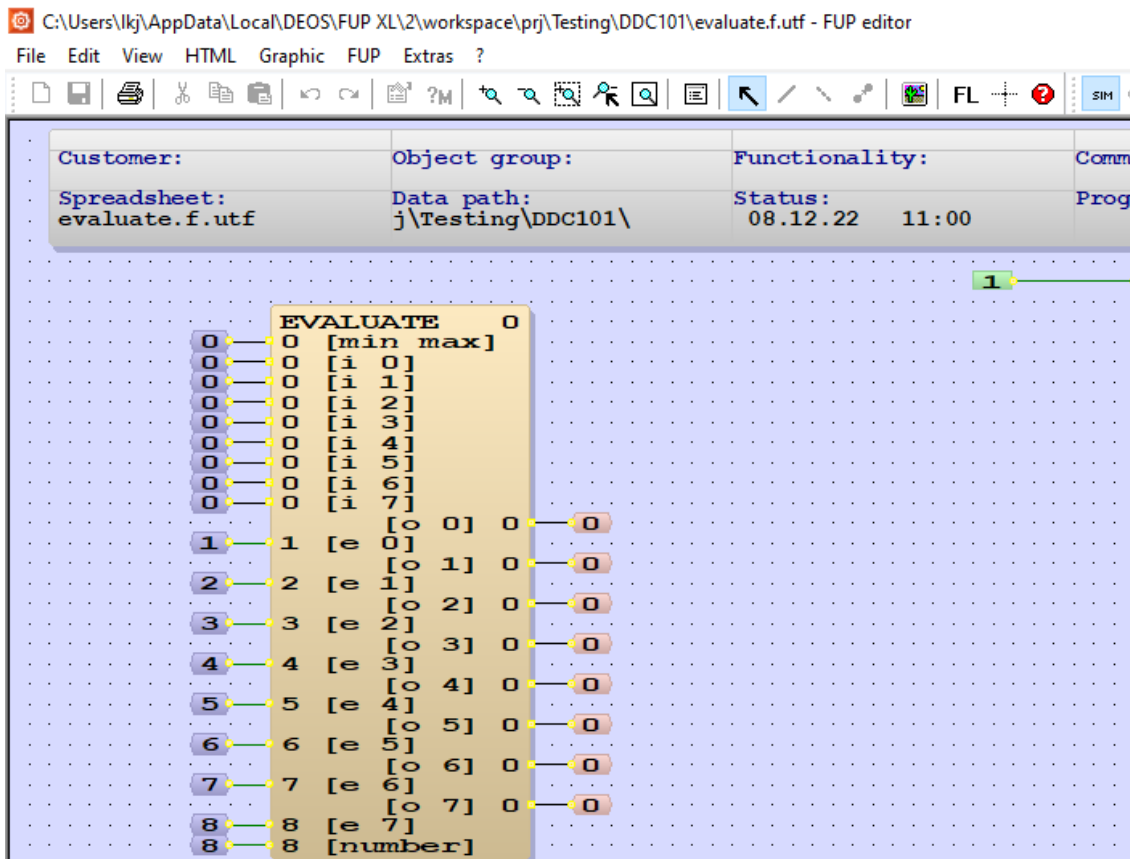
6. Now this module will select the lowest number in "e_0" to "e_7" and control the on/off for "o_0" to "o_7". Enter "1" in "i_0" and you can see "o_1" is ON.
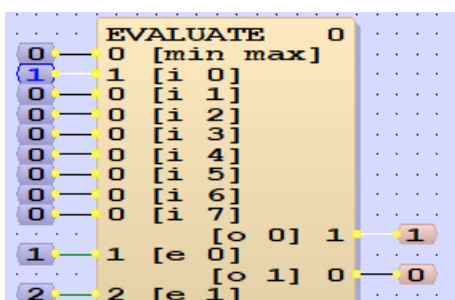
7. Now enter "9" in "e_0" and you can see "o_0" is OFF and "o_1" is ON. This is because "e_1" is now the lowest number.

```
EVALUATE     0
0   0 [min max]
1   1 [i 0]
0   0 [i 1]
0   0 [i 2]
0   0 [i 3]
0   0 [i 4]
0   0 [i 5]
0   0 [i 6]
0   0 [i 7]
            [o 0] 0    0
9   9 [e 0]
            [o 1] 1    1
2   2 [e 1]
            [o 2] 0    0
3   3 [e 2]
            [o 3] 0    0
4   4 [e 3]
            [o 4] 0    0
5   5 [e 4]
            [o 5] 0    0
6   6 [e 5]
            [o 6] 0    0
7   7 [e 6]
            [o 7] 0    0
8   8 [e 7]
8   8 [number]
```
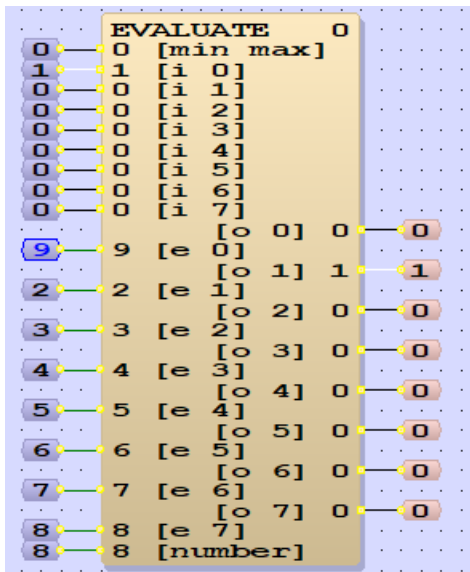
8. You can try entering different number to "e_0" – "e_7" and you can see the corresponding output will turn on which has the smallest value. So, you can use this module to select which equipment (e.g. pump) to start based on the runtime of each equipment. By connecting the runtime of each pump to "e_0" – "e_7", this module will always turn on the pump with lowest runtime.

```
EVALUATE     0
0   0 [min max]
1   1 [i 0]
0   0 [i 1]
0   0 [i 2]
0   0 [i 3]
0   0 [i 4]
0   0 [i 5]
0   0 [i 6]
0   0 [i 7]
            [o 0] 0    0
9   9 [e 0]
            [o 1] 0    0
10  10 [e 1]
            [o 2] 1    1
3   3 [e 2]
            [o 3] 0    0
4   4 [e 3]
            [o 4] 0    0
5   5 [e 4]
            [o 5] 0    0
6   6 [e 5]
            [o 6] 0    0
7   7 [e 6]
            [o 7] 0    0
8   8 [e 7]
8   8 [number]
```
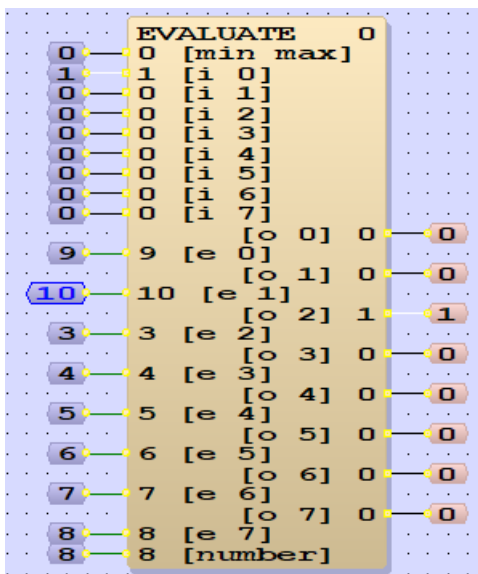
9. Input "i_0" to "i_7" control how many pumps to start. So, to start 2 pumps, enter "1" to "i_1". The module will then select the next lowest runtime pump to start automatically.

```
EVALUATE     0
0   0 [min max]
1   1 [i 0]
1   1 [i 1]
0   0 [i 2]
0   0 [i 3]
0   0 [i 4]
0   0 [i 5]
0   0 [i 6]
0   0 [i 7]
            [o 0] 0    0
9   9 [e 0]
            [o 1] 0    0
10  10 [e 1]
            [o 2] 1    1
3   3 [e 2]
            [o 3] 0    0
12  12 [e 3]
            [o 4] 1    1
5   5 [e 4]
            [o 5] 0    0
```
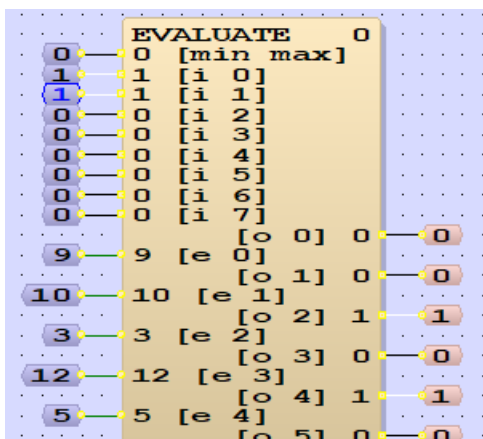
10. So, set the inputs and the module will automatically start/stop the pumps according to their runtime.

```
        EVALUATE    0
0   ── 0 [min max]
1   ── 1 [i 0]
1   ── 1 [i 1]
1   ── 1 [i 2]
1   ── 1 [i 3]
0   ── 0 [i 4]
0   ── 0 [i 5]
0   ── 0 [i 6]
0   ── 0 [i 7]
            [o 0] 0 ── 0
9   ── 9 [e 0]
            [o 1] 0 ── 0
10  ── 10 [e 1]
            [o 2] 1 ── 1
3   ── 3 [e 2]
            [o 3] 0 ── 0
12  ── 12 [e 3]
            [o 4] 1 ── 1
5   ── 5 [e 4]
            [o 5] 1 ── 1
6   ── 6 [e 5]
            [o 6] 1 ── 1
7   ── 7 [e 6]
            [o 7] 0 ── 0
8   ── 8 [e 7]
8   ── 8 [number]
```
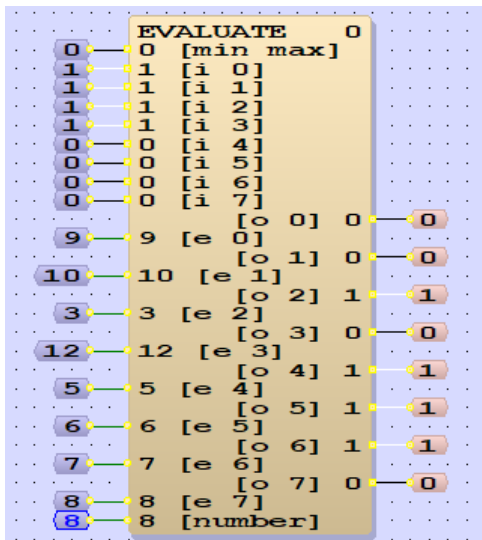
11. The module can control up to 8 pumps (or any other equipment). If you only have 3 pumps, then you can set the input "number" to 3, and then only the first 3 outputs ("o_0" to "o_2") will be controlled.

```
        EVALUATE    0
0   ── 0 [min max]
1   ── 1 [i 0]
1   ── 1 [i 1]
1   ── 1 [i 2]
1   ── 1 [i 3]
0   ── 0 [i 4]
0   ── 0 [i 5]
0   ── 0 [i 6]
0   ── 0 [i 7]
            [o 0] 1 ── 1
9   ── 9 [e 0]
            [o 1] 1 ── 1
10  ── 10 [e 1]
            [o 2] 1 ── 1
3   ── 3 [e 2]
            [o 3] 0 ── 0
12  ── 12 [e 3]
            [o 4] 0 ── 0
5   ── 5 [e 4]
            [o 5] 0 ── 0
6   ── 6 [e 5]
            [o 6] 0 ── 0
7   ── 7 [e 6]
            [o 7] 0 ── 0
8   ── 8 [e 7]
3   ── 3 [number]
```
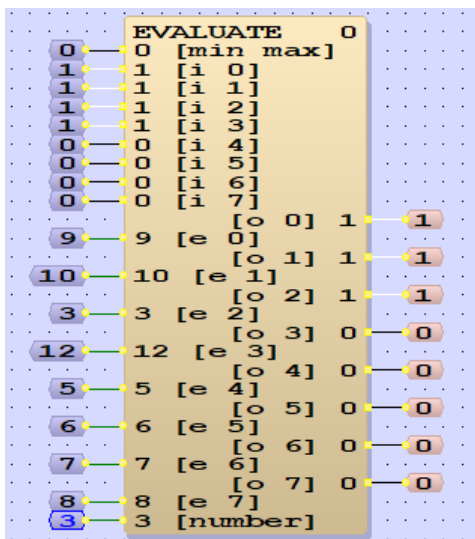
12. Using the "SWITCHMO" modules, you can set the "i_0" to ""i_7" automatically.

```
            SWITCHMO    0
0.9  ── *in 0
0.8  ── *out 0
            *d o 0
1.9  ── *in 1
1.8  ── *out 1
            *d o 1
2.9  ── *in 2
2.8  ── *out 2
            *d o 2
3.9  ── *in 3
3.8  ── *out 3
            *d o 3
I19  ── *x

            SWITCHMO    1
4.9  ── *in 0
4.8  ── *out 0
            *d o 0
5.9  ── *in 1
5.8  ── *out 1
            *d o 1
6.9  ── *in 2
6.8  ── *out 2
            *d o 2
7.9  ── *in 3
7.8  ── *out 3
            *d o 3
     ── *x

            EVALUATE    2
0    ── *min max
     ── *i 0
     ── *i 1
     ── *i 2
     ── *i 3
     ── *i 4
     ── *i 5
     ── *i 6
     ── *i 7
                *o 0 ── D01
I10  ── *e 0
                *o 1 ── D02
I11  ── *e 1
                *o 2 ── D03
I12  ── *e 2
                *o 3 ── D04
I13  ── *e 3
                *o 4 ── D05
I14  ── *e 4
                *o 5 ── D06
I15  ── *e 5
                *o 6 ── D07
I16  ── *e 6
                *o 7 ── D08
I17  ── *e 7
I18  ── *number
```
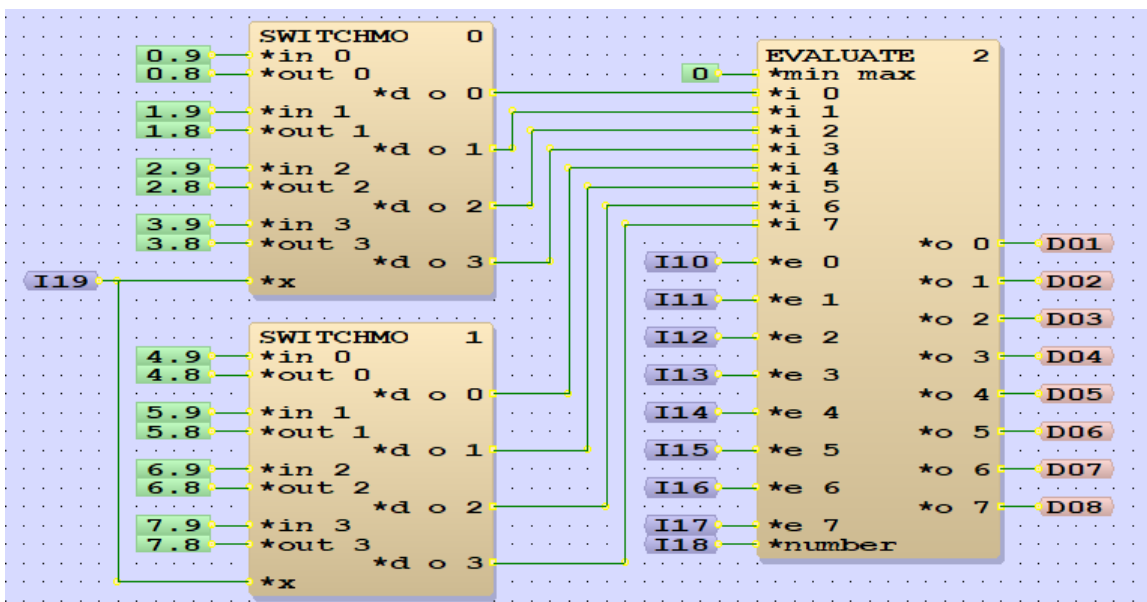
13. For example, to start 6 pumps, enter "6" in the input and the modules will automatically select the 6 pumps with low runtime to start.
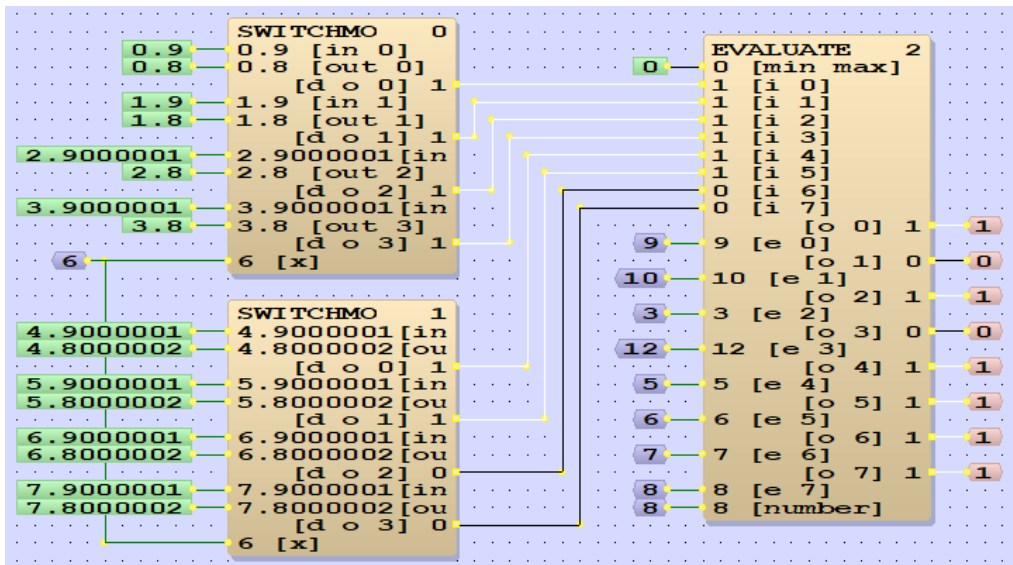
14. Now if you change the runtime of any pump, you'll see the running pumps change immediately. This is normally not what we want, as this means the pumps will start/stop frequently.
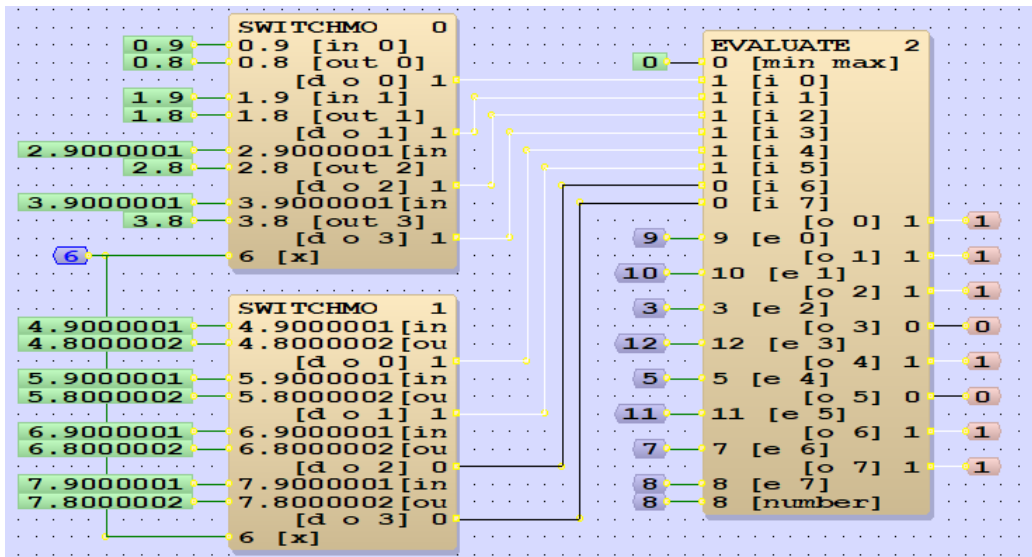
15. To change it, we add a "LATCHFLT" module. This module keeps the runtime unchanged, except when a pulse is given to input "enabl" (e.g. every day at midnight).
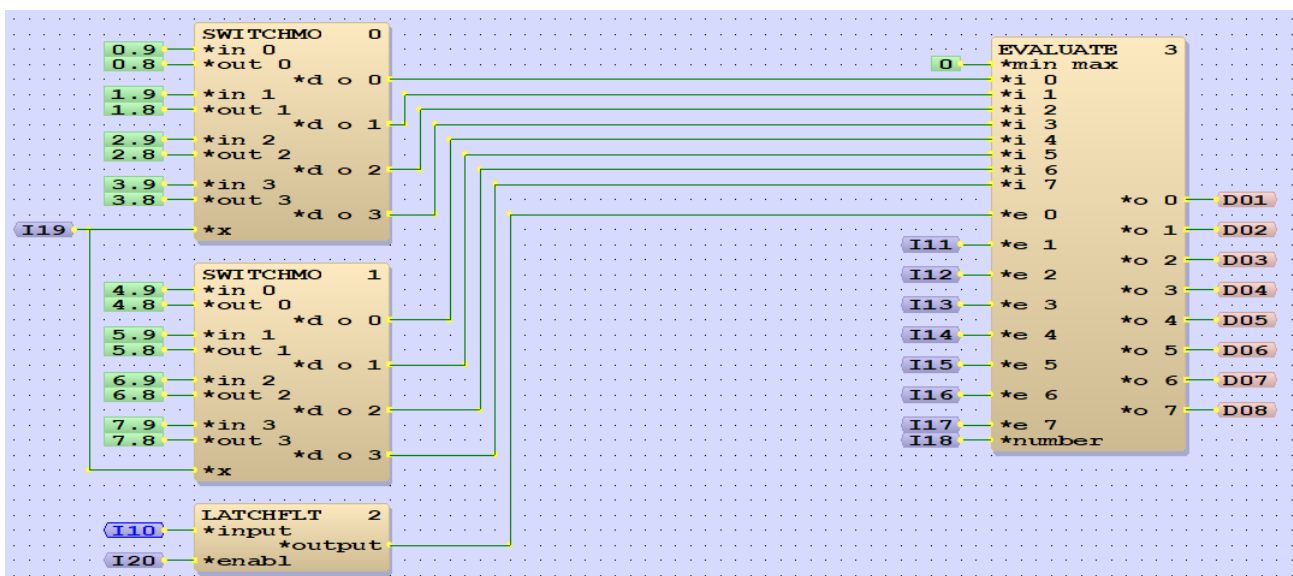
16. You can see from the example below that the runtime for pump 1 increase from 10 to 16, but "e_0" still kept at 10, so the pumps running order will not change during daytime.

```
SWITCHMO    0                                          EVALUATE    3
0.9 ── 0.9 [in 0]                              0 ── 0 [min max]
0.8 ── 0.8 [out 0]                                    1 [i 0]
           [d o 0] 1                                  1 [i 1]
1.9 ── 1.9 [in 1]                                     1 [i 2]
1.8 ── 1.8 [out 1]                                    1 [i 3]
           [d o 1] 1                                  1 [i 4]
2.9000001 ── 2.9000001[in                             1 [i 5]
2.8 ── 2.8 [out 2]                                    0 [i 6]
           [d o 2] 1                                  0 [i 7]
3.9000001 ── 3.9000001[in                                  [o 0] 1 ── 1
3.8 ── 3.8 [out 3]                             8 ── 8 [e 0]
           [d o 3] 1                                       [o 1] 1 ── 1
6 ── 6 [x]                                     10 ── 10 [e 1]
                                                           [o 2] 1 ── 1
SWITCHMO    1                                  3 ── 3 [e 2]
4.9000001 ── 4.9000001[in                                  [o 3] 0 ── 0
4.8000002 ── 4.8000002[ou                      12 ── 12 [e 3]
           [d o 0] 1                                       [o 4] 1 ── 1
5.9000001 ── 5.9000001[in                      5 ── 5 [e 4]
5.8000002 ── 5.8000002[ou                                  [o 5] 0 ── 0
           [d o 1] 1                            11 ── 11 [e 5]
6.9000001 ── 6.9000001[in                                  [o 6] 1 ── 1
6.8000002 ── 6.8000002[ou                      7 ── 7 [e 6]
           [d o 2] 0                                       [o 7] 1 ── 1
7.9000001 ── 7.9000001[in                      8 ── 8 [e 7]
7.8000002 ── 7.8000002[ou                      8 ── 8 [number]
           [d o 3] 0
6 ── 6 [x]

LATCHFLT    2
16 ── 16 [input]
         [output] 8
0 ── 0 [enabl]
```

17. At midnight, you can send a pulse to input "enabl". The new runtime will then be sent to "e_0" and so pump will stop, and another pump will start automatically.

```
SWITCHMO    0                                          EVALUATE    3
0.9 ── 0.9 [in 0]                              0 ── 0 [min max]
0.8 ── 0.8 [out 0]                                    1 [i 0]
           [d o 0] 1                                  1 [i 1]
1.9 ── 1.9 [in 1]                                     1 [i 2]
1.8 ── 1.8 [out 1]                                    1 [i 3]
           [d o 1] 1                                  1 [i 4]
2.9000001 ── 2.9000001[in                             1 [i 5]
2.8 ── 2.8 [out 2]                                    0 [i 6]
           [d o 2] 1                                  0 [i 7]
3.9000001 ── 3.9000001[in                                  [o 0] 0 ── 0
3.8 ── 3.8 [out 3]                             16 ── 16 [e 0]
           [d o 3] 1                                       [o 1] 1 ── 1
6 ── 6 [x]                                     10 ── 10 [e 1]
                                                           [o 2] 1 ── 1
SWITCHMO    1                                  3 ── 3 [e 2]
4.9000001 ── 4.9000001[in                                  [o 3] 0 ── 0
4.8000002 ── 4.8000002[ou                      12 ── 12 [e 3]
           [d o 0] 1                                       [o 4] 1 ── 1
5.9000001 ── 5.9000001[in                      5 ── 5 [e 4]
5.8000002 ── 5.8000002[ou                                  [o 5] 1 ── 1
           [d o 1] 1                            11 ── 11 [e 5]
6.9000001 ── 6.9000001[in                                  [o 6] 1 ── 1
6.8000002 ── 6.8000002[ou                      7 ── 7 [e 6]
           [d o 2] 0                                       [o 7] 1 ── 1
7.9000001 ── 7.9000001[in                      8 ── 8 [e 7]
7.8000002 ── 7.8000002[ou                      8 ── 8 [number]
           [d o 3] 0
6 ── 6 [x]

LATCHFLT    2
16 ── 16 [input]
         [output] 16
1 ── 1 [enabl]
```

18. To stop the pump when it's in alarm, and start another pump automatically, you can use the "ANA_MUX_2" module. Set "i_1" to "999999" so it will always be the highest when in alarm.

```
                *d o 2                                 *i 6
3.9 ── *in 3                                   *i 7
3.8 ── *out 3                                          *o 0 ── DO1
                *d o 3
I19 ── *x                                      *e 0
                                               I11 ── *e 1       *o 1 ── DO2
SWITCHMO    1                                  I12 ── *e 2       *o 2 ── DO3
4.9 ── *in 0                                   I13 ── *e 3
4.8 ── *out 0                                                    *o 3 ── DO4
                *d o 0                                           *o 4 ── DO5
5.9 ── *in 1                                   I14 ── *e 4
5.8 ── *out 1                                  I15 ── *e 5       *o 5 ── DO6
                *d o 1                          I16 ── *e 6      *o 6 ── DO7
6.9 ── *in 2
6.8 ── *out 2                                   I17 ── *e 7      *o 7 ── DO8
                *d o 2                          I18 ── *number
7.9 ── *in 3
7.8 ── *out 3
                *d o 3
*x

LATCHFLT    2          ANA MUX 2    4
I10 ── *input          I21 ── *selector
         *output             *i 0
I20 ── *enabl          999999 ── *i 1    *output
```
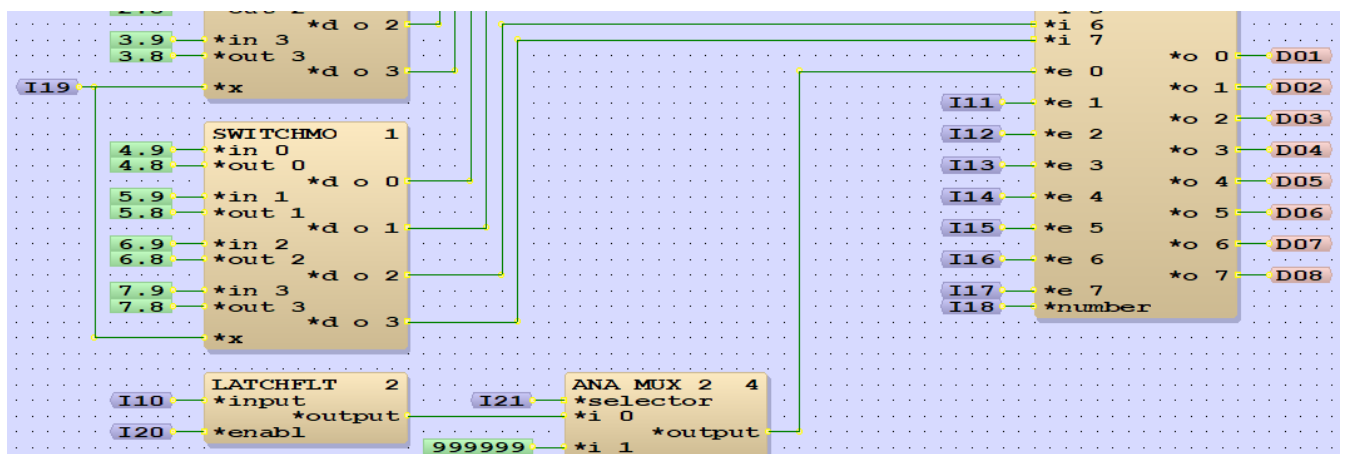
19. So, when it's in alarm (input "selector" is 1), "e_0" becomes 999999, and it will be stop and another pump start. It will start again when alarm return to normal.

```
    3.8 — 3.8 [out 3]                                        [o 0] 0 — 0
            [d o 3] 1                              999999 [e 0]
      6 — 6 [x]                              10 — 10 [e 1]        [o 1] 1 — 1
                                                                 [o 2] 1 — 1
            SWITCHMO    1                      3 — 3 [e 2]
 4.9000001 — 4.9000001[in                                        [o 3] 0 — 0
 4.8000002 — 4.8000002[ou                    12 — 12 [e 3]
            [d o 0] 1                                            [o 4] 1 — 1
 5.9000001 — 5.9000001[in                     5 — 5 [e 4]
 5.8000002 — 5.8000002[ou                                        [o 5] 1 — 1
            [d o 1] 1                         11 — 11 [e 5]
 6.9000001 — 6.9000001[in                                        [o 6] 1 — 1
 6.8000002 — 6.8000002[ou                     7 — 7 [e 6]
            [d o 2] 0                                            [o 7] 1 — 1
 7.9000001 — 7.9000001[in                     8 — 8 [e 7]
 7.8000002 — 7.8000002[ou                     8 — 8 [number]
            [d o 3] 0
      6 — 6 [x]

            LATCHFLT    2              ANA MUX 2    4
      8 — 8 [input]              1 — 1 [selector]
            [output] 0                 0 [i 0]
      0 — 0 [enabl]                    utput] 999999
                             999999 — 999999 [i 1]
```

20. The final program looks like this. It will automatically select the lowest runtime pump(s) to start, and switch to another pump when the pump is in alarm.

```
            SWITCHMO    0                                  EVALUATE   18
    0.9 — *in 0                                       0 — *min max
    0.8 — *out 0                                          *i 0
            *d o 0                                         *i 1
    1.9 — *in 1                                            *i 2
    1.8 — *out 1                                           *i 3
            *d o 1                                         *i 4
    2.9 — *in 2                                            *i 5
    2.8 — *out 2                                           *i 6
            *d o 2                                         *i 7
    3.9 — *in 3
    3.8 — *out 3                                           *e 0        *o 0 — D01
            *d o 3                                         *e 1        *o 1 — D02
   I19 — *x                                                            *o 2 — D03
                                                          *e 2         *o 3 — D04
            SWITCHMO    1                                 *e 3         *o 4 — D05
    4.9 — *in 0                                           *e 4         *o 5 — D06
    4.8 — *out 0                                          *e 5         *o 6 — D07
            *d o 0                                         *e 6        *o 7 — D08
    5.9 — *in 1
    5.8 — *out 1                                          *e 7
            *d o 1                           I18 — *number
    6.9 — *in 2
    6.8 — *out 2
            *d o 2
    7.9 — *in 3
    7.8 — *out 3
            *d o 3
         — *x

            LATCHFLT    2              ANA MUX 2    3
   I10 — *input                 I21 — *selector
            *output                    *i 0
   I20 — *enabl                            *output
                             999999 — *i 1

            LATCHFLT    4              ANA MUX 2    5
   I11 — *input                 I22 — *selector
            *output                    *i 0
         — *enabl                          *output
                             999999 — *i 1

            LATCHFLT    6              ANA MUX 2    7
   I12 — *input                 I23 — *selector
            *output                    *i 0
         — *enabl                          *output
                             999999 — *i 1

            LATCHFLT    8              ANA MUX 2    9
   I13 — *input                 I24 — *selector
            *output                    *i 0
         — *enabl                          *output
                             999999 — *i 1

            LATCHFLT   10              ANA MUX 2 11
   I14 — *input                 I25 — *selector
            *output                    *i 0
         — *enabl                          *output
                             999999 — *i 1

            LATCHFLT   12              ANA MUX 2 13
   I15 — *input                 I26 — *selector
            *output                    *i 0
         — *enabl                          *output
                             999999 — *i 1

            LATCHFLT   14              ANA MUX 2 15
   I16 — *input                 I27 — *selector
            *output                    *i 0
         — *enabl                          *output
                             999999 — *i 1

            LATCHFLT   16              ANA MUX 2 17
   I17 — *input                 I28 — *selector
            *output                    *i 0
         — *enabl                          *output
                             999999 — *i 1
```

21. Please note that it's only an example program that shows you how to use the "EVALUATE" module. Additional logics are required for full sequencing functionality.