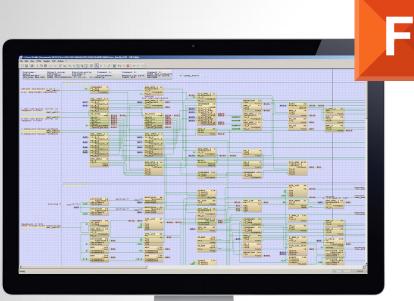


FUP XL (Programming Software) DEOS Controls Training Class

_∕ → Agenda







Agenda Day #2

- 1. Introduction to FXL
- 2. Function Block Groups
- 3. Function Blocks functionality
- 4. Create Project
- 5. Create Program
- 6. Create Graphics
- 7. Program Simulation
- 8. Graphics Simulation



 C:\Users\Erhard Dobler\AppData\L File Edit View HTML Graphic 	.ocal\ FUP		S\FUP ctras	? XL\2	\wo	rkspa	ace\p
			?м		-0	HQ	<u>_</u>
basic elements and modules 			- 14	~	~	1N	~
		Cu	ıst	om	er		
show details >>		Sr	ore	be	sh		+ •
Search	:	hc	me	.f	. u	tf	
Search in helptext				• •	: :	•••	• •
⊕ 🗁 Basic elements	1::				1.1	11	1.1
🚋 💼 Math			• • •	1.1	: :	11	
i Math-Ext							
ia - Canal Physics ia - Canal Time	· ·		• •		• •	• •	1.1
i⊞⊶ Con Time i⊞⊸ Con Time BSW						1.1	1.1
		• •	• •		• •	• •	
Modifier					1.1	1.1	1.1
			• • •		· ·		(\cdot, \cdot)
🗄 💼 Logic_dig				1.1	1	11	1.1
🗄 🗠 Logic_mux							
🗄 🛅 Logic_ana		• •	• • •	1.1	• •	• •	1.1
in Cogic_txt							1.1
i⊞⊶ ⊂⊐ Loop		• •	• • •		• •		
E-C C500-bus							1.1
· · · · · Other	· ·	• •	• •	• •	• •	· ·	· ·
				1.1	1	1.1	1.1
🗈 🗂 CAN-bus							
🗄 💼 SRC_modules		• •	• • •	1.1	• •	• •	1.1
Field_devices							1.1
ie ⊂ Sun		• •	• •		• •	• •	
				1.1	1.1	1.1	1.1
BACnet_MSTP Gracteristic_lines			• •	(\cdot, \cdot)	· ·	• •	(x_{i},y_{i})
Characteristic_lines Mbus_modules							1.1
	1						
					1.1		
🗄 💼 Change		• •	• •		• •		
						1.1	1.1
			• • •		· ·		
	· · ·						1.1

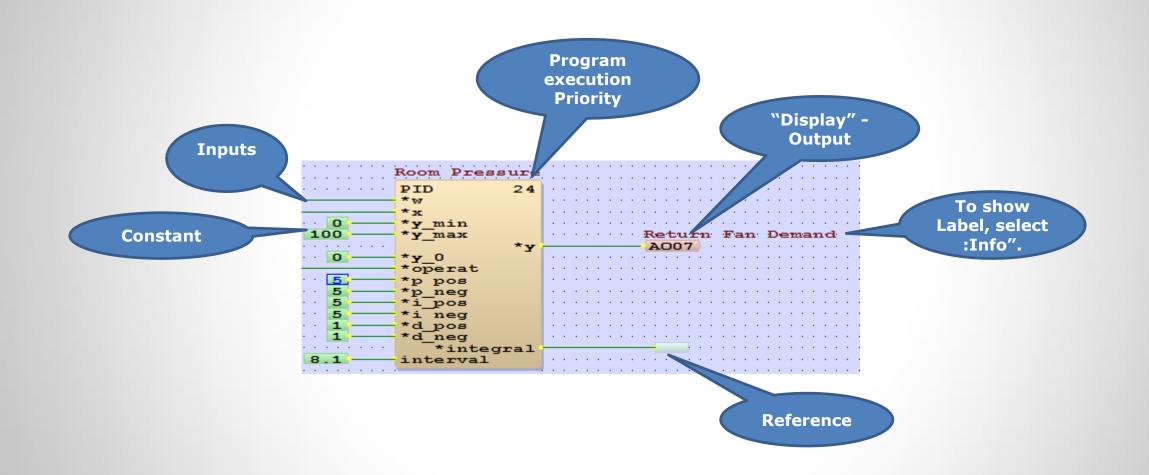
Open "Basic Elements and Modules" at the "View" in the main menu.

Tip: you can use the hot key "F2"

Each Element Group folder is expandable by pressing (+) and collapsible by (-)

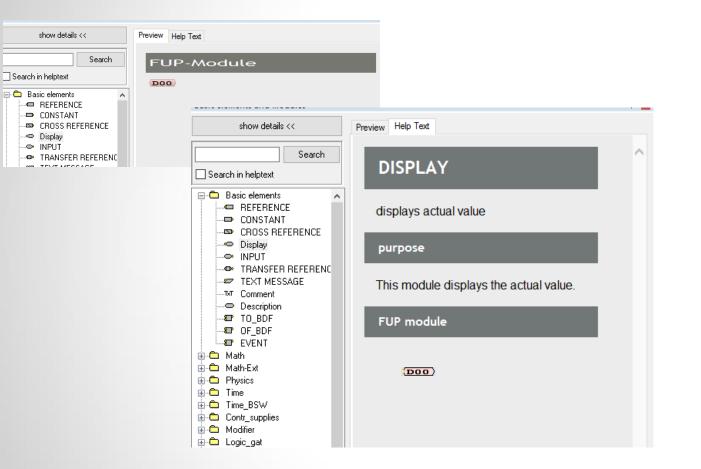
Function Basic Elements Groups





Function Block Groups



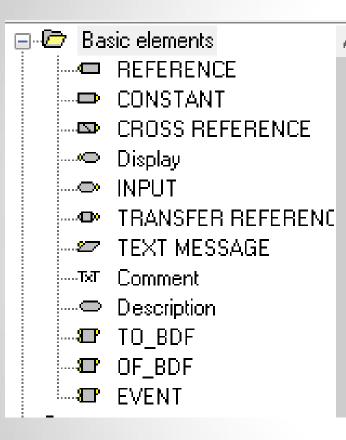


Every Function block has a detailed information, that can be displayed by the "show detail" button.

It opens the "Preview" and "Help Text" tabs, that show labels, inputs, outputs and explain functionality of the function block or element.

The help file is the part of the programming tool FUP XL and it can assist during development.

--- Basic Elements Group

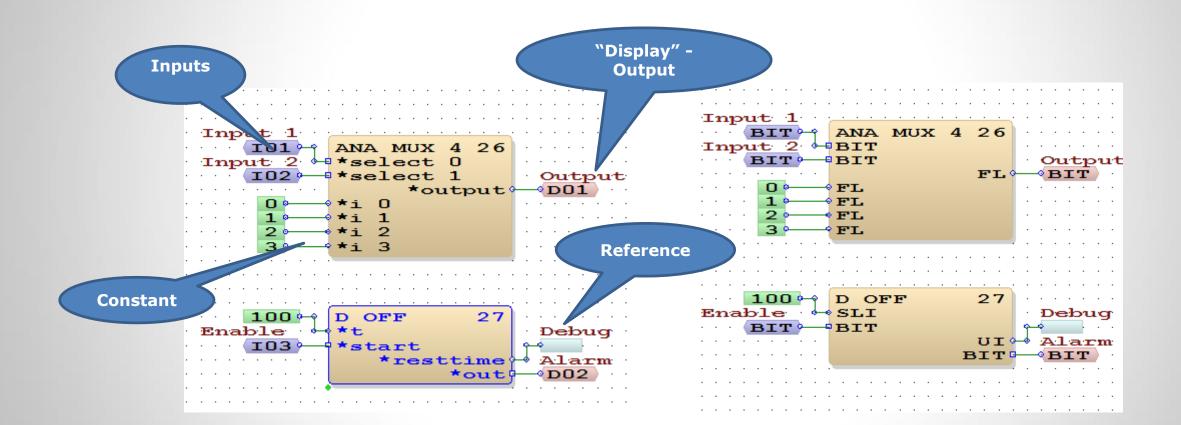


- Reference this value can be shared over multiple program pages and controllers. Tip: you can use "Reference" on the output side of the function block to avoid error during compilation.
- > Constant this value represents value, that does not change.
- Cross Reference This element returns value of a Reference inside the same controller.
- Display display or output. It is variable, that value can be changed by user or program. It represents the Output value.
- Input Represents Input values. It is variable, that value can be changed by user or program
- Text message displays text messages, in the "Events" page in the controller. Can work as alarm and text can be sent via e-Mail. Works as binary value (0 - off, 1 - on)
- Comments –text information which you want to place in the program page
- > **Description** to label points or write comments in the program page

controls

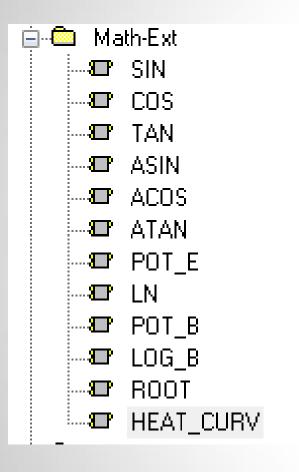
Function Basic Elements Groups







Math Operation (Float Numbers) Group



- Math Extension are Mathematical calculation which are implemented in the function block. Typical Math Operations, mostly with float numbers.
- Heat_Curv This function is used to schedule the water temp. for Radiation and Under floor heating systems. It calculates the Supply Water set point based on OAT temperature.

Physics Group

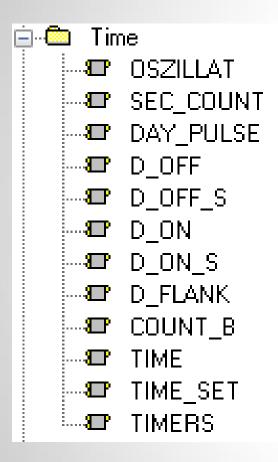


Physics ----**E** P T_H20 ----- TPH20 H_X_DIA ----**P** DELTA_H -**P** HNN PM WHR MIXAIR**T** TEMPHUM

- P_T_NH3 Converts the saturation pressure in the boiling temperature for NH3
- P_T_H20 Calculates the dew point temperature for ice and water at given vapor pressure.
- T_P_H20 calculates the partial pressure for ice and water by given temperature and humidity.
- H_X_DIA calculates the air situation of temperature "theta", humidity "rh" and barometric pressure "p".
- Delta H calculates the heating and cooling energy which is needed to turn air from one condition to another.
- HNN_PM calculates the average barometric pressure "p_medial" by given location height "h".
- WHR Control of the waste heat recovery with outside and circulating air throttles.
- > MIXAIR Calculation of temperature and humidity of mixed air
- TEMPHUM Calculates the humidity, if the temperature of the air is changed.

Timers and Time Delay Group

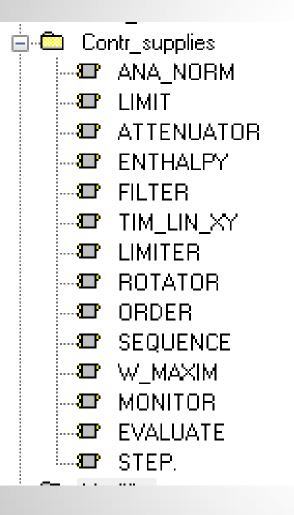




- OSZILLAT- Clocks the output "out" when the input "start" is assigned to "1". The pulse and pause times will be set by the variables "pulse" and "pause". While the input "start" is assigned to "0", the output "out" is set to "0"
- **SEC_Count** Executes the counting of seconds.
- DAY_PULSE Writes out a "1" to the variable "out" for the time which is given by the variable "duration".
- **DAY_OFF** Executes a falling delay. Reset time in 1/10 Sec.
- >DAY_OFF_S Executes a falling delay. Reset time in Seconds.
- > DAY_ON Executes a Turn on delay. Reset time in 1/10 Sec.
- >DAY_ON_S Executes a Turn on delay. Reset time in Seconds.
- ➤ D_FLANK Delay module.
- **Count_B** Operating hours counter (Runtime).
- **TIME** Used to present the current time.
- >TIME_SET Set UST-clock
- >TIMERS Measures the time of the input impulses at the inputs "more" and "less".

Control Group





- Limit Limits the OUTPUT by two set values.
- **Filter** Harmonizes inputs signals
- Tim_Lin_XY Liniear output value which can be timed to avoid cycling of OUTPUT signals
- Limiter Limits output signals to pre-set values. For example 0-100% limit, 20% to 80%.
- Rotator Rotates outputs to equal run times on equipment. For example rotating pumps ext. up to 16 devices per function block

Variable Modifier Group



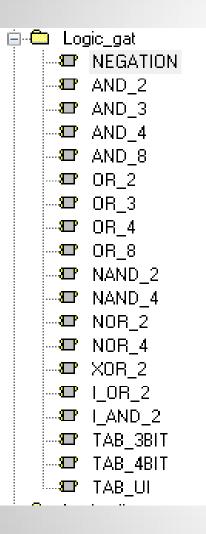
💼 👛 Modifier E FL_BIT - 💵 FL_UI E FL_SI 🖅 SLI_BIT 🖅 SLIFL BIT2 - 🖅 🛛 BIT3 - 🖅 BIT4 - 🖅 BIT2 FIELD - 💵 BIT4 FIELD - 🖅 UI_BIT BIT_UI BIT_MAX3 - 🖅 🛛 BIT_MAX7 BIT_MX15 INT3ULI INT3SLI ULISINT SLI3INT - 🖅 STARTOP - 🖅 HM_DEC ANZBLES ANZALES -🖅 SLI_ULI 💵 ULI_SLI - 💵 ULI_TXT - 🕶 MWU E MTR

- FL_BIT Converts a Float value (analog) to a Bit (digital) signal
- **FL_UI** Changes a float number to an unsigned integer.
- **FL_SI** Changes a float number to an signed integer.
- SLI_BIT analyzes the value of an integer and operates the output to 1 while the integer value is not equal to 0.
- >SLI_FL converts an integer to a float number.
- > BIT2 analyzes the states of bit-fields of the BDF.
- BIT3 analyzes the states of bit-fields of the BDF. A field of 3 bit will be created if there are given 8 ASCII-characters
- BIT4 analyzes the states of bit-fields of the BDF. A field of 4 bit will be created if there are given 16 ASCII-characters.
- BIT2_FIELD and BIT4_FIEL- This module creates characters at a bit display field of the BDF as a result of the bit pattern in the UST.
- >**UI_Bit** Converts an integer to a bit pattern.
- **BIT_UI** Converts a bit pattern into a number.
- > BIT_MAX3/7/15 sets the number of the highest set bit to the output "out".
- START_OP converts the marker states of the day- \ night- \ clock-choose for the graphic.

NOTE: Compiler does not need BIT (binary) to FL (float) conversion.

Logic Operations Group





Typical Binary or Boolean Operations.

> **NEGATION –** Reversing a Digital Signal

≻AND_2, 3,4,8 – AND-GATE

≻OR_2,3,4,8 – OR-GATE

➤ NAND_2,4 - NAND - GATE

≻NOR_2,4 – NOR – GATE

XOR- Exclusive OR Gate

I-AND_2 – Exclusive AND Gate

Logic Operations Group



j. • 🙆	🛛 Log	jic_dig
	8	BIT_OUT
	8	BIT_IN
	•••••	DECOD_2
	8	DECOD_3
	8	FF_DYNM
	•••••	FF_DYN
	C	FF_RSM
	8	FF_RS
	8	FLANK
	D	LATCHBIT
	8	LATCHINT
	8	LATCHSLI
	•••••	LATCHFLT
	8	RESET
	8	RESETN
	8	GATE8_BIT
	:	COUNTER
	8	COUNTERL

> **BIT_OUT –** Output selector, one inputs and 5 outputs.

- > **BIT_IN**, Totalizes four Digital inputs and summarizes on the Output
- > **DECOD_2** analysis of 2 digital inputs.
- > **DECOD_3** analysis of 3 digital inputs
- **FF_DYNM** works like an impulse relay.
- **FF_DYN -** works like an impulse relay.
- FF_RS RS flip flop. While the input "i_0" is set to 1 a 0 is written on the variable "output". If "i_0" is set to 0 and "i_1" to 1 then the output "output" is overwritten with a 1 until "i_0" is set to 1 again.
- LATCHBIT transfers the status of "input" to "output" as long as the input "enabling" = 1. If "enabling" = 0, the status of "input" will not be transmitted to "output" anymore. Then "output" will save the latest transmitted value.
- **RESET** This module creates a reset impulse of adjustable time.
- > "RESET" sets independently the input to null after run of the given time
- **RESETTN** This module creates a reset impulse of adjustable time.
- ➤ "RESETN" sets independently the input to null after run of the given time.



Selectors or Switches and If statements

👛 Logic mux 🖅 🖅 ANA MUX 2 🖅 ANA MUX 8 🖅 DIG MUX 2 🖅 DIG_MUX_4 DIG MUX 8 ----**C** BITDMUX16 BIT_MUX16 E FLTDMUX16 🖅 🖅 ANADMUX 2 INTDMUX16 INT_MUX16 ----**-**----🖅 🕶 SLI MUX 2 📖 🖽 BUS MUX 8

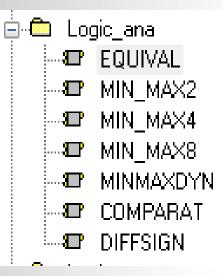
Mostly Selectors, Switches and If statements.

>ANA_MUX2 – Analog Multiplexer – 2 dig Inputs >ANA MUX4 – Analog Multiplexer – 4 dig Inputs > ANA_MUX8 – Analog Multiplexer – 8 dig Inputs **DIG_MUX2 –** Digital Multiplexer – 2 dig Inputs > **DIG_MUX4 –** Digital Multiplexer – 4 dig Inputs > DIG_MUX8 – Digital Multiplexer – 8 dig Inputs **BITDMU-16** – De-multiplexer for marker **BIT_MUX16** - Multiplexer for marker **FLTDMUX** – De-multiplexer for float numbers **FLT_MUX16** - multiplexer for float numbers >ANADBUX_2- De-multiplexer for float values >INTDMUX 16 – De-multiplexer for long data words. One Input and 16 Outputs >INT_MUX16 - Multiplexer for long data words. 16 Inputs and 1 Output. >SLI_MUX_2 - SLI-multiplexer, If input "selector" = 0, then "Output" = "i 0". \triangleright If input "selector" = 1, then "output" = "i 1". >BUS_MUX_8 - Switching of an 8-bit wide bus. If the input "select" is taken by a number higher than "0" the states of the inputs "in1_x" will be set to the outputs

"out_x", otherwise the states of the inputs "in0_x" will be set to the outputs "out_x".

Comparisons and Minimum Maximum Group

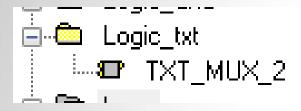




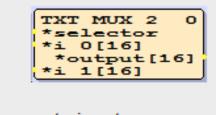
- EQUIVAL- Analog Multiplexer 2 dig Inputs. For example IF IP1=1 and IP =1 THEN OP1=1, IF IP1=1 and IP2=0 Then OP1 =0.
- MIN_MAX2 This module compares the value of the variable "input" with the value of "minimum" and "maximum" and writes out the accordant limited value to "output" and visible for the user to "input".
- MIN_MAX4 This module compares the values of the variables "i_1" up to "i_3" and writes out the minimum value to the output "min" and the maximum value to the output "max".
- "MIN_MAX_4" sets up the AND- and OR-relation respectively of 4 elements for FUZZYcontrollers.
- MIN_MAX8 This module compares the values of the variables "i_1" up to "i_7" and writes out the minimum value to the output "min" and the maximum value to the output "max".
- "MIN_MAX_8" sets up the AND- and OR-relation respectively of 4 elements for FUZZYcontrollers.
- MINMAXDYN This module compares the values of the variables "i_1" up to "i_7" and writes out the minimum value to the output "min" and the maximum value to the output "max".
- "MIN_MAX_8" sets up the AND- and OR-relation respectively of 4 elements for FUZZYcontrollers.
- COMPARAT This module compares the values of the variables "i_min" and "i_max". While "i_min" < "i_max", a "1" will be set to the variable "output".</p>
- > **DIFFSIGN** This module describes a comparator and compares the variables "i_1" and "i_2".

Text Switch Group





>TXT_MUX_2



elector	gate input
_0[16]	text input if gate input = 0
utput[16]	text output
1[16]	text input if gate input = 1

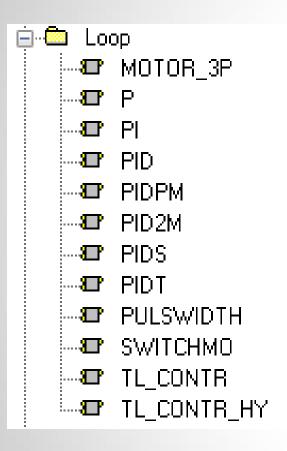
s

O

i

Loops Group





> **MOTOR_3P** – Floating point control with pre-set run-time of motor.

- ➤ P Proportional Control
- PI Proportional Integral Control
- PID Proportional–Integral–Derivative Control
- > **PIDPM** Proportional–Integral–Derivative Floating Point Control
- PID2M Proportional–Integral–Derivative Floating Point Control and analog Control output.

PIDS - If "tn" <= 0, the PI-loop will not change the value of "integral" but always keeps the latest given or calculated value.

If "y_min" > "y_max", the effective direction of the controller will be reversed.

PIDT - If ("i_pos" = 0) and ("i_neg" = 0) then does the PID-loop never change the value of "integral", but always keeps the latest entered value.

SWITCHMO - switching module and switches four digital outputs depending to the value of the variables "w".

- TL_CONTR switching module and switches a digital output in dependance of the value of the variable "x".
- TL_CONTRL_HY two-level controller with adjustable switching hysteresis

CAN-bus Group – Part 1



CAN-bus INIT_CAN SRV_CAN CAN_DI CAN_DI CAN_DI_LED CAN_DO CAN_DO_HR CAN_AO CAN_AO CAN_AO_HR

- CAN PUSHBIN
- CAN_COUNT
- CAN_PULSTIME
- CAN_RELEASE
- 🖙 🗊 CAN_STATUS

- INIT_CAN This module is for the initialization of a CAN-bus. Used if I use the Can bus port on the controller.
- SRV_CAN service-module for CAN-bus. Used if I use the Can bus port on the controller.
- CAN_DI Can bus Digitial Input (For DI16) It is not used with the OPRN UI8x8
- CAN_DI_LED This module handles the LED-colors of digital inputs. CAN_DO – Can Bus Module with Digital output
- CAN_DO_HR This module connects the COSMOS IO modules DA8T(H), DA8R(H)
- **CAN_AI** This module connects the COSMOS IO modules AI8AO4(H).
- \succ Reads the current state of an analog input.
- **CAN_AO** This module connects the COSMOS IO modules AE8AA4(H).
- Sets the status of an analog output.

CAN-bus Group – Part 2



CAN-bus

- SRV CAN
- CAN_DI
- CAN_DI_LED
- E CAN_DO
- CAN_DO_HR
- CAN_AL
- CAN_AO
- CAN_AO_HR
- CAN_COUNT
- CAN_PULSTIME
- CAN_RELEASE

📖 🖅 CAN STATUS

CAN_AO_HR - This module connects the COSMOS IO modules AE8AA4(H).

- > Sets the status of an analog output.
- **CAN_PUSHBTN** Connects the COSMOS IO modules DE16

CAN_COUNT - This module connects the COSMOS IO modules DE16. Output of the counter reading of a digital input.

"CAN_PUSHBTN" only has to be used once per hardware module and terminal in the project.

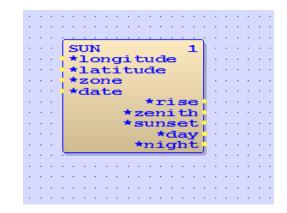
- CAN_PULSE_TIME With this module the time spans between two impulses of an digital input can be measured. It will be measured the time span of the "1" as well as the time span of two impulses.
- CAN_RELEASE Dropping of an output DI8DO8T
- **CAN_STATUS** Provides runtime information about the CAN data bus.

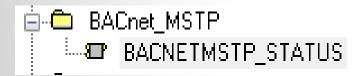


___ Misc. Groups



SUN- SUN allows to calculate the sunrise and the sunset



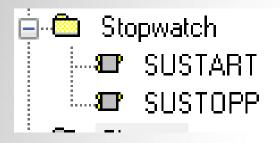


BACNETMSTP - Providing runtime information about the MS/TP data bus.

2	<u> </u>	
	BACNETMSTP 1	·
	*quali	
	*devic	•
	*count	•
	*sent	•
	*recei	•
	*error	•
	*poll	•
	*unwan	•
	*lost	•
	*flowr	•
- 2		<u>ا</u>

<u>Stop</u> Watch





SUSTART – By the stopwatch function it is possible to create multiple stopwatches on a controller which can detect with a high resolution intervals in a runnable controller program.

These stopwatches can be used for example to optimize controller programs according to reaction timing on changed input values. With this module a stopwatch can be started.

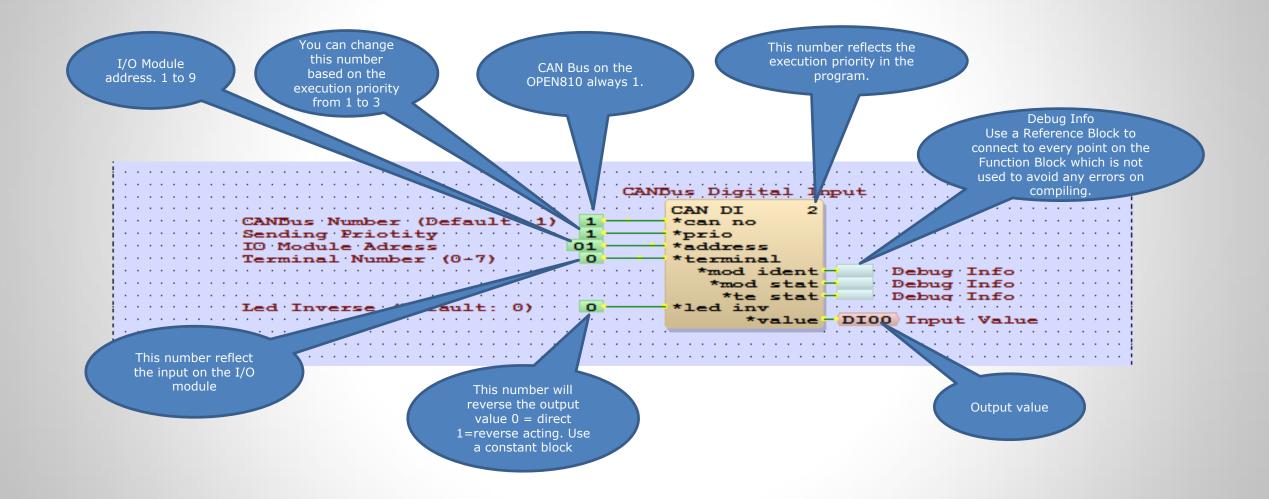
SUSTOP - By the stopwatch function it is possible to create multiple stopwatches on a controller which can detect with a high resolution intervals in a runnable controller program.

These stopwatches can be used for example to optimize controller programs according to reaction timing on changed input values. The interval of a running stopwatch is measured by this module.

You can use as many stop times depending on one stopwatch as you whish.

CAN Bus Function Block Setup





CAN Bus Function Block Setup

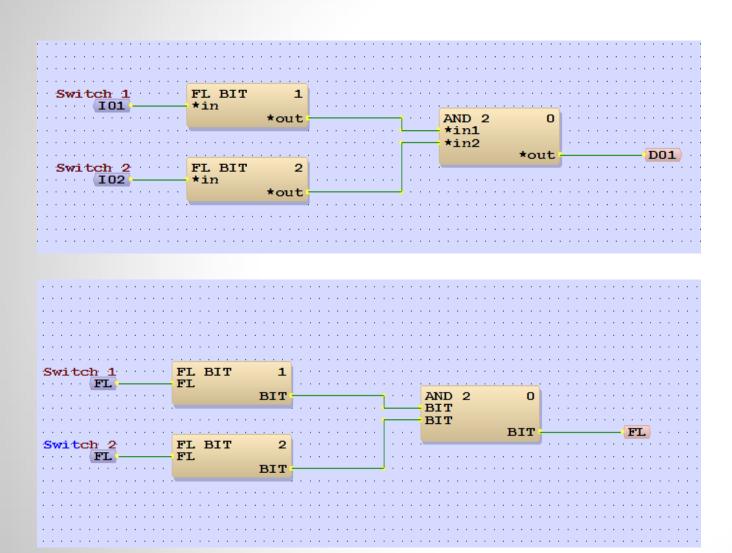


· · · · · · · · · ·	· · · · · · · · · · · · · · · · · ·		· · · · · · · · · · · · · ·
	ADD 1 *summarrd1 *summarrd2	Setpoint IN04 Min output Max Output I01 *y min I02 *y max	
•	ADD 2 *sym- *symand1 *symand2	Actual Val IN03 Controller Enabled On/Off PID Loop Proportional Band IO5 *y 0 IO4 *operat IO6 *xp	OUT3
· · · · · · · · ·	· · · · · · · · · · · · · · · ·	Regulate Time; sec I07 - *tn Derivative Tim; sec I08 - *td Accuracy I09 - *i tot Number of loops in sec I11 - *interval	
· · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · ·

	ADD 1	Cotroint			PII) E D		
		Setpoint						
	[sum] 20	20				[w]		
· · · · [O]	0 [summand1]	· · · · · · · · · · · · · · · · · · ·			15	[x]		
	20 [summand2]		Min output			[y min]		
	,		Max Output		100-100	[y max]	Output	
	ADD 2	Actual Val			I I I I I I I I I I I I I I I I I I I	52.613331	- 52.61333	1
	[sum] 15	15	Controller E	nabled	n h í i	v 01		
		10	On/Off PID I		1 1			
	0 [summand1]				15 10	[operat]		
	15 [summand2]		Proportional		10-10	[qx]		
			Regulate Tim	e; seconono de	120 - 120) [tn]		
			Derivative T	'im, sec	··· O - O	[td]		
	· · · · · · · · · · · · · · ·		Accuracy	· · · · · · · · · · · · · · ·	0.1 - 0.1	[i tot]		
					all		-2.613332	5
			Number of lo	ops in sec				
				ops in 366		[III COL VAL]		



Simple Block Programming



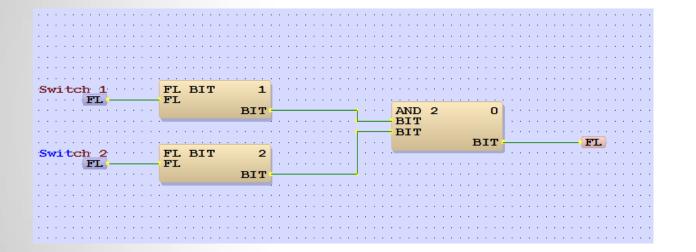
Tip:

To show the Input type, select **VIEW** and select **Show Type** and the I01 will show the Type of the input module you have selected.

This is a great tool to avoid a mixing type of constant and variables during the programming

Simple Block Programming





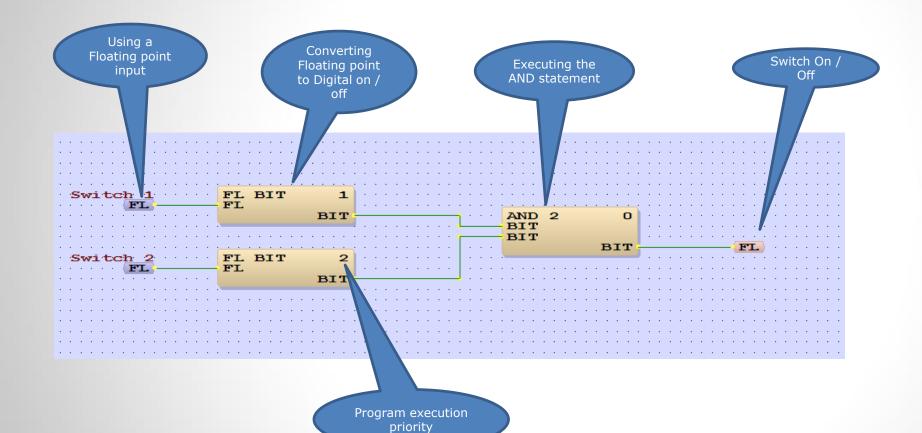
Tip:

To show the Input type, select VIEW and Select show and the I01 will show the Type of the input module you have selected.

This is a great tool to avoid a using the wrong type of constant of variables during the programmin

Simple Block Programming

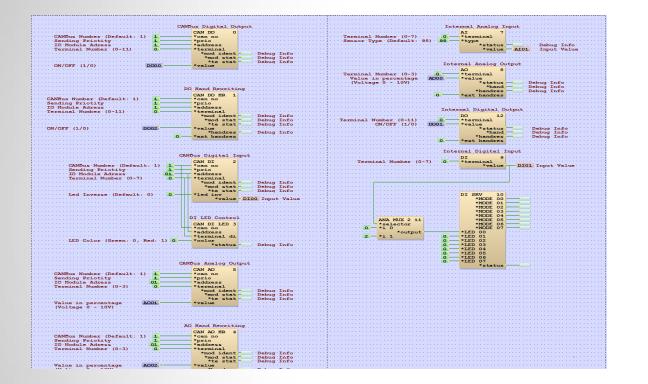




Tip: Don't mix Floating function blocks with Digital function blocks. You always have to match Digital to Digital and convert with function block Floating to Bit.

___ Exercise #2





Create Pre-set I/O & PID Blocks

Create your own Library of pre-set Can-bus – I/O blocks Pre-set PID loops UI 8x8 DI 16 DO8

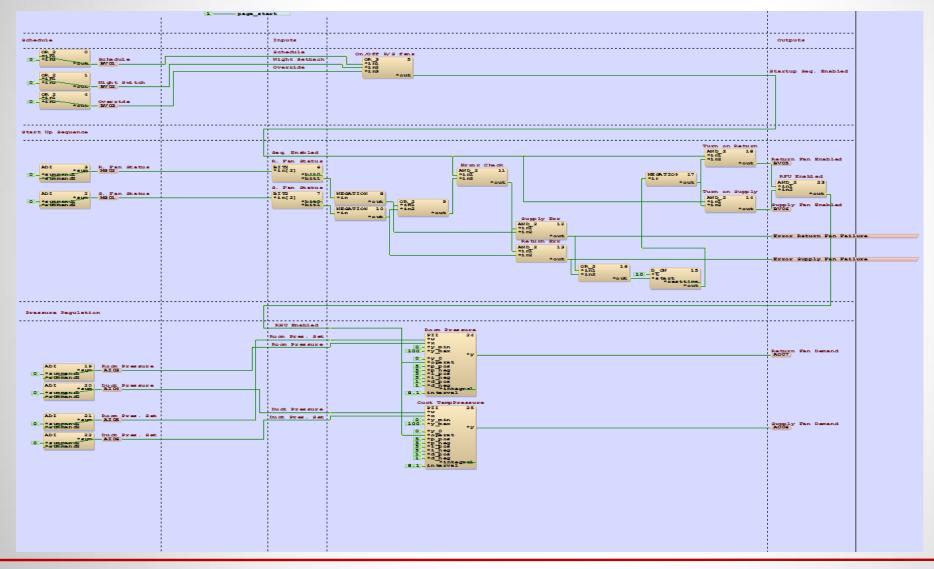
The exercise is to build up your library with module which we have testing in class.

Create a AHU Program

Inputs		Outputs	
1	Night Set Back	1	RF-enable
2	Override	2	SF-Enable
3	RF-Status	3	RF-Speed
4	SF-Status	4	SF-Speed
5	Room Press.	5	
6	Room Press. SP	6	
7	Supply Duct Press.	7	
8	Supply Duct Press. SP	8	



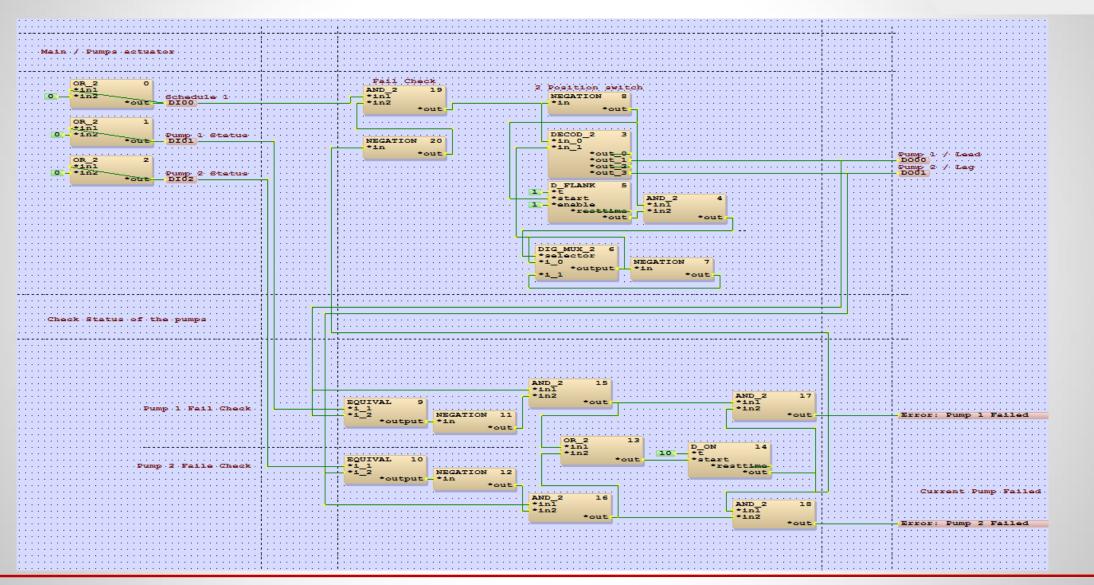






Create a Lead Lag Program

Inputs		Outputs		
1	Pump 1 Status	1	Pump 1 Enable	
2	Pumpt 2 Status	2	Pump 2 enable	
3	Pump 1 Fail	3		
4	Pump 2 Fail	4		
5	Supply Water Temp		Heating Valve Control	
6	Return Water Temp.	6		
7	Supply Water Setpint	7		
8		8		







___ Exercise #5

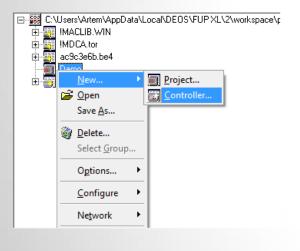
Create a functional Library with applications for your own library

Simple Cont	rols Loops	
Lead Lag Pro	ogram	
Start up Seq	unce	
Damper Con		
Heating & Co	ooling valve controls	
Boiler Contro	I (with 5 open flame boilers)	
Chiller Start u	up and Control sequence	

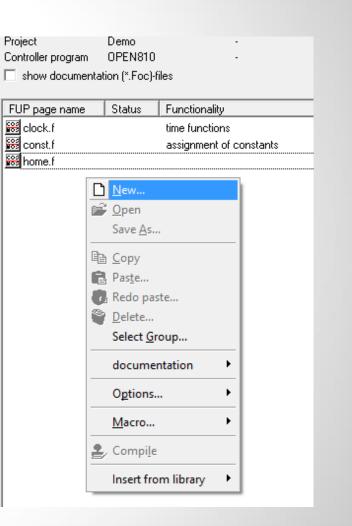
FXL Graphics

Create New Project and Controller Add new programming Page.

Tip – You can import any other pictures or Visio (Auto Cad) floor plan Graphics.



ame:			Default se	ettings of the macro	library:
)PEN			•		•
ontroller type					
Controller program	Info	description program	version	IP-address	Subnet mas
WINLIB_SRU_01.UST	SRU	0012c			
WINLIBOPEN_3100.UST	OPEN 3100 EMS	1.054m		192.168.170.100	255.255.0.0
WINLIBOPEN_4100.UST	OPEN 4100 EMS	1.054m		192.168.170.100	255.255.0.0
WINLIBOPEN_500.UST	OPEN 500 EMS	1.054m		192.168.170.100	255.255.0.0
WINLIBOPEN_600.UST	OPEN 600 EMS	1.054m		192.168.170.100	255.255.0.0
WINLIBOPEN_710.UST	OPEN 710 EMS	1.054m		192.168.170.100	255.255.0.0
WINLIBOPEN_810.UST	OPEN 810 EMS	1.054m		192.168.170.100	255.255.0.0
<					3





Creating a new Graphic Page



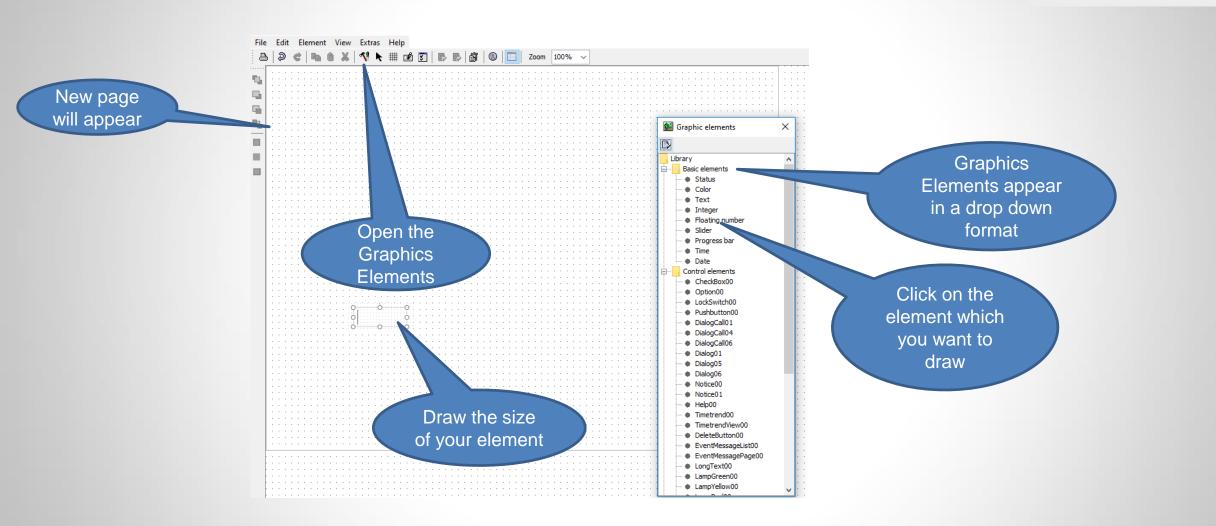
<u>@</u> (:\User:	s\Erhar	d Dobler\/	AppData\L	ocal\D	EOS\FU	P XL\2\w	/orkspace\p	rj\Steve2\OI	PEN81			
File	Edit	View	HTML	Graphic	FUP	Extras	?						
E D		A	X 🗈 I	New	Page	L	+0 -0	x 🔯 🌾		К			1. Open p
_		9	00					N LEN N					2. Select
•				🏙 <u>1</u> 🤤	graphic								
	Cus	stor	ner:			ບມີເ	ect	group	>:	Fu			Menu
	Spi	read	dshee	et:		Data	a pa	th:	N810	Sta			3. New g
:	hon	ne.1	f.uti	E		j/s	teve	2\OPE	N810	11			0
													created
• •	• •				• • •								
	•••	· · · ·	e Edit View I	HTML Graphic	FUP Extras	••••							
							~ R Q E	🔨 🗸 🔨 🥜	📓 FL 🕂 🚱	SIM ONL	TXT		
• •	• •	• • •	Custome	ar:	Obi	ect gr	0110 :	Function	ality:	Conmer	ot 1:	Comment 2:	Comment 3:
			Spreads	sheet:	Dat	a path	:	Status:	_	Progra	ammer:	Module lib path: LIB01.FUP	Executing cy
			nome.1		,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	itevez (OPENSIO	0\ 10.05.1				LIBUL.FOP	SECOND
									· · · · · 1	pa	ge start		
		-			· · · · · ·			· · · · · · · · · · · ·				· · · · · · · · · · · · · · · · · · ·	
			· · · · · · · ·		•••••		· · · · · · · ·	· · · · · · · · · ·	· · · · · · · · · ·	· · · · · · ·		· · · · · · · · · · · · · · · · · · ·	
								· · · · · · · · · ·	· · · · · · · · · ·	· · · · · · ·	· · · · · · · · · · · ·		
			Swi	tch 1		L BIT	1						
		:		I01 —		in	*out	· · · · · · · · · · · · · · · · · · ·	AND 2		o		
		:		· · · · · · · · · · ·				· · · · · · · · · · · · · · · · · · ·	*in1 *in2	creat	e graphic page	• • • • • • • • • • • • • • • • • • •	×
		-	Swi	tch 2		L BIT	2				h page number should the ne	w graphic page get?	
							*out		· · · · · · · · · · ·		ar page number should the ne	u grobino hago gor:	ОК
			· · · · · · · · ·	· · · · · · · · · ·	· · · · · · · ·	· · · · · · · ·	· · · · · · · · · · · ·	· · · · · · · · · · · ·	· · · · · · · · · · · · ·				Cancel
		•											

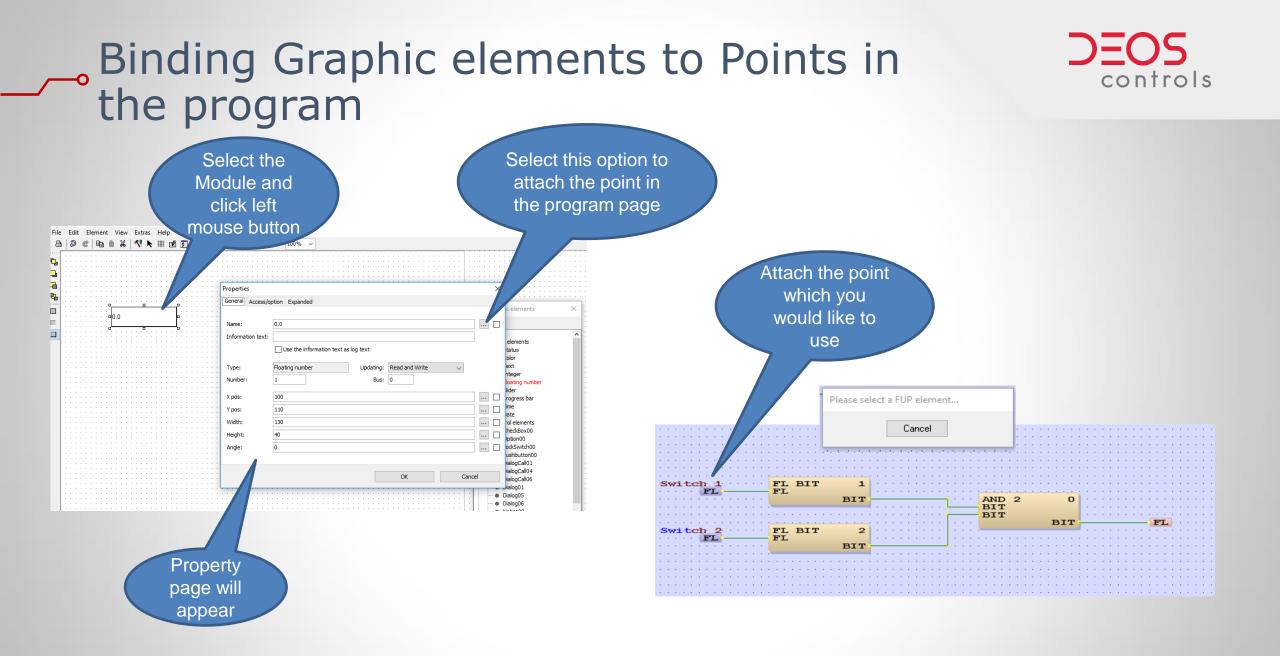
pen program page

- elect Graphics in the Main enu
- ew graphical page will be eated

DEOS Controls – Products

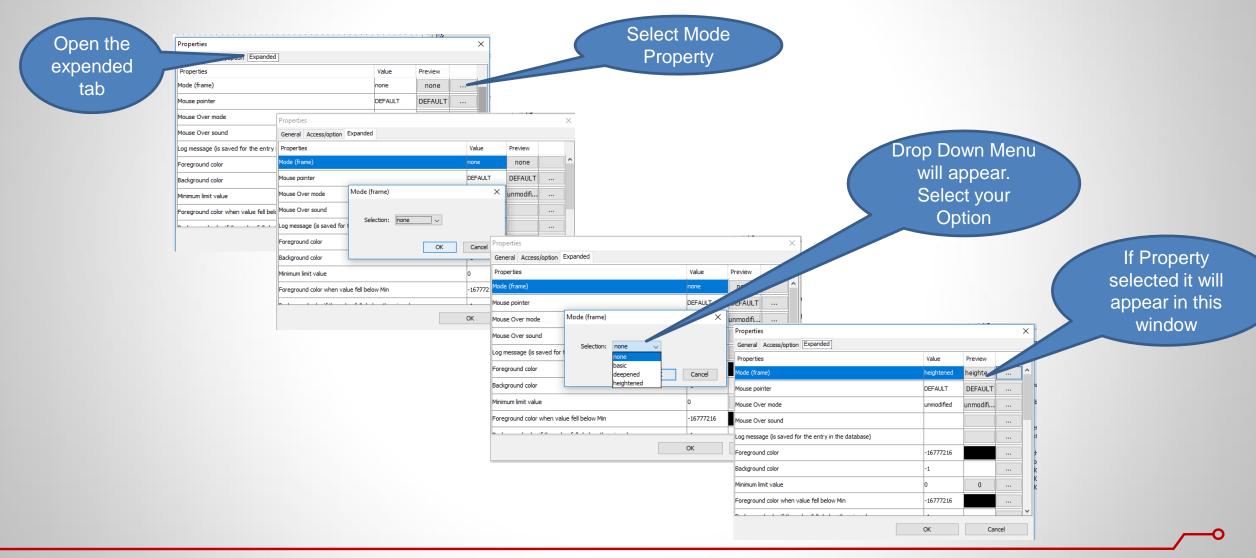






Configure Graphic elements visual effect





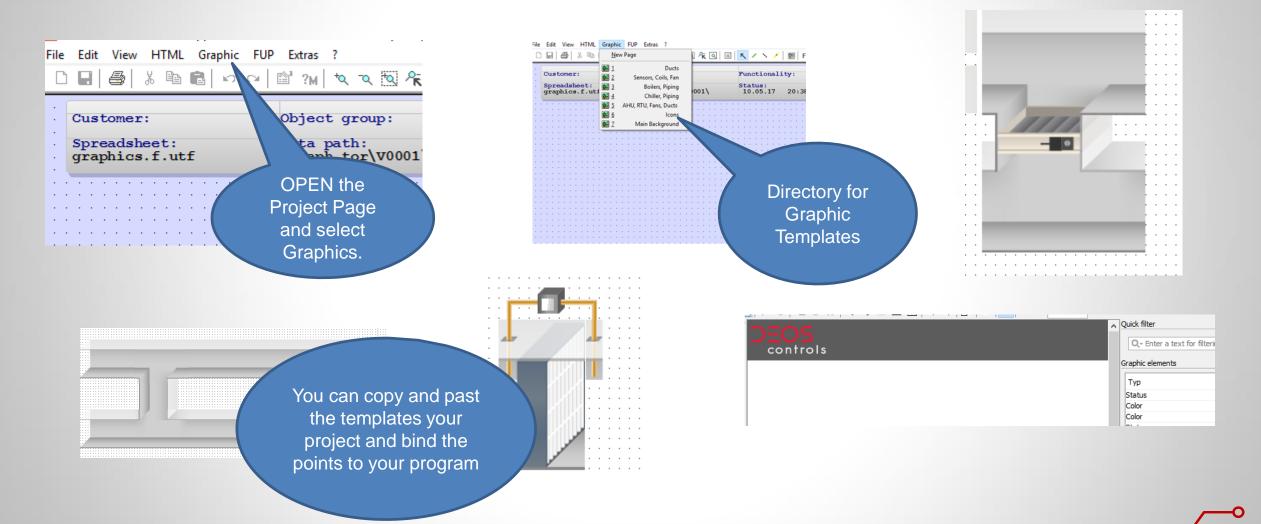
Working with Graphics Library





Working with Graphics Library





5/12/17

DEOS Controls - Technology for Intelligent Buildings



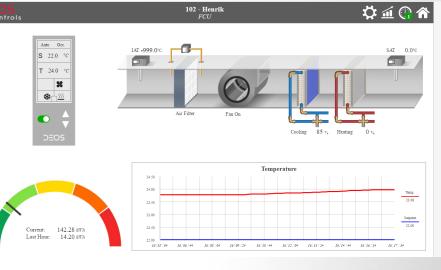
Create Graphics for our Trainings Project

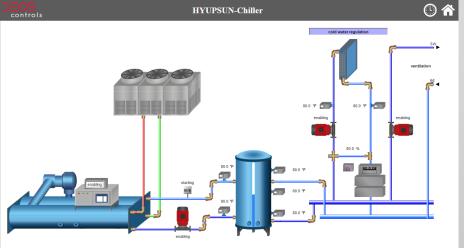
"Start Page" with Building Picture and Labels for the Floor Levels Single "Floor Level" with Room's and Labels for Temperature Sensors and VAV links to VAV Graphic Create "Navigation" function so you can navigate though your project



Project Example







DEOS Controls – Technology for Intelligent Buildings



➤ Right Click on your project in the tree and choose "Compile"

ppData\Local\DEOS\FUPXL Froject Controller pro v show do	Training gram OPEN810 umentation (*.Foc)-							
FUP page r	ame Status	Functionality	Object group	Date modified	Programmer	Customer	Extension	Execution cycle
🗈 area.f00	MACRO	Main Area	Additional	24.02.17 12:57	Artem Pavlov	DCA	f00	Second
clock.f		time functions	General	04.06.12 09:15	DCS		F	Second
🚟 const.f		assignment of constants	General	29.11.11 17:05	DCS		F	Minute
🕨 err_dca.	MACRO	Alarms Style Sheet	Essential	13.03.17 12:30	Artem Pavlov	DCA	f	Second:0.25
🗈 firpin.f	MACRO	Floor Plan	Essential	27.01.17 08:39	Artem Pavlov	DCA	f	Second:0.25
🚺 🗈 gr_dus.f	MACRO	Common Graphics, STAT, Icons	Essential	09.03.17 14:38	Artem Pavlov	DCA	f	Second
🗈 home.f0		homepage Area		04.01.17 11:48	Artem Pavlov	DCA	f00	Second
🗈 main e M		Trainings Building - Floor 01 - RM-01-Frank S		13.03.17 11:02	Artem Pavlov	DCA	f00	Second
	FupToC		×	13.03.17 11:02	Artem Pavlov	DCA	f01	Second
🗈 ma				13.03.17 11:02	Artem Pavlov	DCA	f02	Second
🕑 ma		<u> </u>		13.03.17 11:02	Artem Pavlov	DCA	f03	Second
🗈 ma		÷ <u></u>	Essential	11.07.16 13:52	Ho	DEOS	f	Execution cycle
🗈 ms			Essential	09.03.17 12:34	Artem Pavlov	DCA	f	Second:0.25
🗈 rec			Essential	28.02.17 12:04	Artem Pavlov	DCA	- F	Second
D rec			Essential	13.04.17 13:51	Artem Pavlov	DCA	- F	Second
		controller and creating C-Code	Essential	01.03.17 09:28	Artem Pavlov	DCA	f	Second
	f 30 (0 Error)		Essential	27.01.17 11:29	Artem Pavlov	DCA	E	Second
	Isers\Erhard Doble	\AppData\Local\DEOS\FUP	Essential	13.04.17 14:03	Artem Pavlov	DCA	E	Second
🗈 rec			Essential	29.03.17 10:31	Artem Pavlov	DCA	F	Second
🗈 rtu			Trainings Building	09.03.17 13:42	Artem Pavlov	DCA	f00	Second:0.25
🗈 sci			Essential	05.01.17 11:28	Artem Pavlov	DCA	f	Second
🗈 sei			Essential	09.03.17 14:19	Artem Pavlov	DCA	f	Execution cycle
🗈 sti			Essential	28.02.17 14:21	Artem Pavlov	DCA	f	Second
🗈 tre			Trainings Building	05.01.17 15:42	Artem Pavlov	DCA	f00	Second:0.25
🗈 tre			Trainings Building	05.01.17 15:42	Artem Pavlov	DCA	f01	Second:0.25
	ating C-Code	Cano	cel Trainings Building	05.01.17 15:42	Artem Pavlov	DCA	f02	Second:0.25
🗈 tre			Trainings Building	05.01.17 15:42	Artem Pavlov	DCA	f03	Second:0.25
▶ vvt.f01	MACRO	WT With Stat	Trainings Building	13.03.17 13:40	Artem Pavlov	DCA	f01	Second:0.25
>	MACDO.	ANZTA CALCARA	Tanisis as Daibling	10.0017 10.40	Astern Decileri	DC/	(0.0	C 4.0 OE

---- Checking hard disk capacity (Minimum required is 25 MByte....) -------- Checking FUP pages in controller C:\Users\Erhard Dobler\AppData\Local\DEOS\FUP XL\2\workspace\pri\Training\OPEN810\ and creating C-Code -----



G:\Users\Erhard Dobler\AppData\Local\DEOS\FUP XL\2\workspa Jim JMDCA.tor	ac Controller program	OPEN810 ation (*.Foc)-fil
ing-∰ !MGraph.tor Er-∰ Axyan	FUP page name	Status
E Avvan	area, f00	MACRO
	Clock.f	MACHO
🕀 🚎 Confi	const.f	
🖶 🊟 Steve	err_dca.f	MACRO
⊞ <mark>≣</mark> Steve2	flrpln.f	MACRO
🖻 📆 Training	🖸 📴 gr_dus.f	MACRO
⊡	home.f00	MACRO
🖞 🚘 Open	main_r.f00	MACRO
Gave As	🗈 main_r.f01	MACRO
	🗈 main_r.f02	MACRO
👸 Delete	🗈 🗈 main_r.f03	MACRO
Select <u>G</u> roup	manager.f	MACRO
	mstpdiag.f	MACRO
<u>B</u> ACnet	reccoil.f	MACRO
System integration	recdamp.f	MACRO
Import •	recduct.f	MACRO
Ortigen	recfan.f	MACRO
Options •	recmain.f	MACRO
😩 Compile F7	rtu_vvt.f00	MACRO MACRO
🛗 Recompile all	schedule.f	MACRO
Stop Compiling	service.f	MACRO
Upload F5	stl_dus.f	MACRO
	trenddyn.f00	MACRO
Configuration	trenddyn.f01	MACRO
	trenddyn.f02	MACRO
OPEN <u>w</u> eb	trenddyn.f03	MACRO
Properties	🕑 vvt.f01	MACRO
	🕑 vvt.f02	MACRO
Info <u>f</u> ile •	Vvt.f03	MACRO
Insert from library		
c	, I	

✓ show documentation (".Foc)-files				
FUP page name	Status	Functionality	ОЬ	
🗈 area.f00	MACRO	Main Area	Adı	
🚟 clock.f		time functions	Ge	
😂 const.f		assignment of constants	Ge	
▶ err_dca.f	MACRO	Alarms Style Sheet	Es:	
▶ firpin.f	MACRO	Floor Plan	Es:	
▶ gr_dus.f	MACRO	Common Graphics, STAT, Icons	Es:	
▶ home.f00	MACRO	homepage Area		
┣ main_r.f00	MACRO	Trainings Building - Floor 01 - RM-01-Frank S.		
▶ main_r.f01	MACRO	Trainings Building - Floor 01 - RM-02-John M.		
┣ main_r.f02	MACRO	Trainings Building - Floor 01 - RM-03 - Albert F.		
▶ main_r.f03	MACRO	Trainings Building - Floor 01 - Mechanical RM-01		
▶ manager.f	MACRO	Manager to NETVAR	Es:	
▶ mstpdiag.f	MACRO	MSTP Status	Es:	
▶ reccoil.f	MACRO	Resources: Coils	Es:	
▶ recdamp.f	MACRO	Resources: Damper	Es:	
🖻 recduct.f	MACRO	Resources: Ducts	Es:	
▶ recfan.f	MACRO	Resources: FAN	Es:	
▶ recmain.f	MACRO	Resources: Main	Es:	
🗈 recsens.f	MACRO	Resources: Sensors	Es:	
▶ rtu_vvt.f00	MACRO	RTU WT	Tra	
▶ schedule.f	MACRO	Schedule	Es:	
🖹 service.f	MACRO	Service Menu	Es:	
🖻 stl_dus.f	MACRO	Style Sheet	Es:	
▶ trenddyn.f00	MACRO	10240 Samles	Tra	
🖹 trenddyn.f01	MACRO	10240 Samles	Tra	
▶ trenddyn.f02	MACRO	10240 Samles	Tra	
▶ trenddyn.f03	MACRO	10240 Samles	Tra	
🖻 vvt.f01	MACRO	VVT With Stat	Tra	
🕑 vvt.f02	MACRO	VVT With Stat	Tra	
▶ vvt.f03	MACRO	VVT With Stat	Tra	

- Right Click on your project in the tree and choose "Upload"
- Make sure, your computer is connected to the OPEN controller you with to load



Workflow x Image: CPENSID (Loader) x <t< th=""><th>ools 7.0.4</th><th></th><th>- 🗆 ×</th></t<>	ools 7.0.4		- 🗆 ×
Workflow x Image: DePB810 (Loader) x Image: DepB810 (Loader) x Image: DepB810 (Loader) x	View Window Help		
Workflow x Image: DePB810 (Loader) x Image: DepB810 (Loader) x Image: DepB810 (Loader) x	i 🗟 🗙 🕨 🗎 🦻 격		
Image: Solution of the set of the			
Connection settings IP address: 192.158.170.100 Comment: Test OPEN Project name: Training Controller program name: OPEN810 Cancel Load Problems Output - Training OPEN810 (Loader) ×	1	× ± • C	
Comment: Test OPEN Project name: Training Controller program name: OPEN810 Cancel Load Problems Output - Training OPEN810 (Loader) ×			
Project name: Training Controller program name: OPENB10 Cancel Load		IP address: 192.168.170.100	
Project name: Training Controller program name: OPENB10 Cancel Load		Commants Tast ODEN	
Controller program name: OPEN810 Cancel Load Problems Output - Training OPEN810 (Loader) ×			
Cancel Load		Project name: Training	
Problems Output - Training OPEN810 (Loader) ×		Controller program name: OPEN810	
Problems Output - Training OPEN810 (Loader) ×			
The connection to the OPEN EMS could not be established. (192.168.170.100)			
		ine connection to the OPEN EMS could not be established. (192.168.170.100)	

Enter the IP address of the connected OPEN controller

➢ Click "Load"



Enter the controller's IP address in the address bar of your browser and navigate through the graphical user interface

← → C (i) 192.168.1.105/client/index	.html		
👖 Apps 🔽 107 🔽 111 G Google 🛅 Lin	nkedIn 💿 QuickBooks 🔵 DCA 👍 Ana	ilytics 📕 Sign in to your accour 🛛 🔽 Configuration Tool	
⊍ ≡ i ⊑ 🗸 🗠 %	0		
 O1: PR.Open Events Homepage 	DEOS controls	Floor 01	🌣 🕒 🕀
 Homepage Password Circuit times Essential General Service controller Training Building system 	Floor 01	S 20.0 RM-01-Frank S. T 0.0 RM-02-John M. T 0.0 RM-03-Albert F. T 0.0 Mechanical RM- T 0.0 I/AIV-IVIT 153002 T 0.0 I/AIV-IVIT T 0.0 RTU-IVIT 153002 T 0.0 I/AIV-IVIT 153003 T 0.0 RTU-IVIT	



Questions

5/12/17

Ю

DEOS Controls – Technology for Intelligent Buildings

47