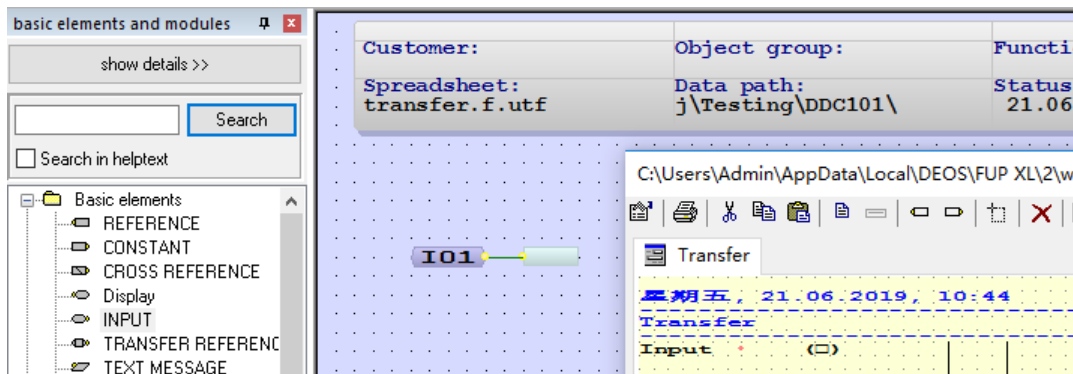
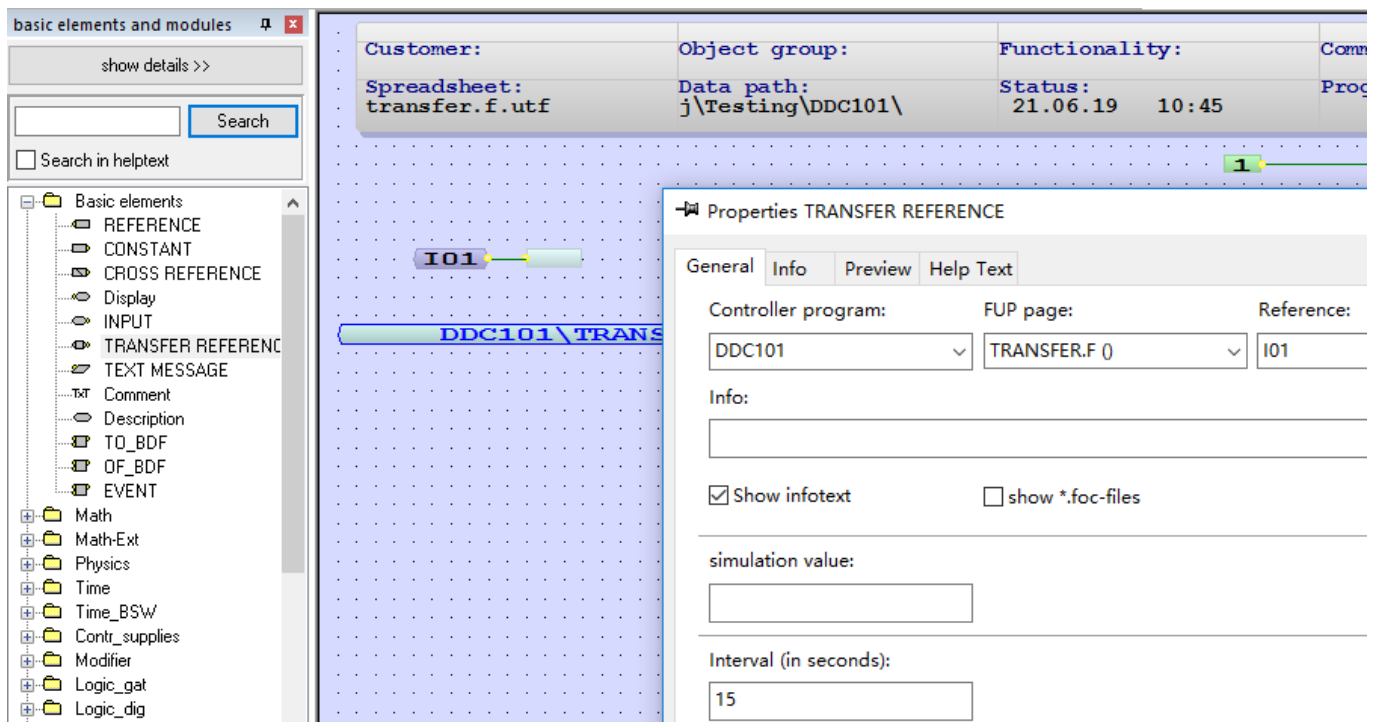


## TT190704 – FUP - Transfer Points between Controllers

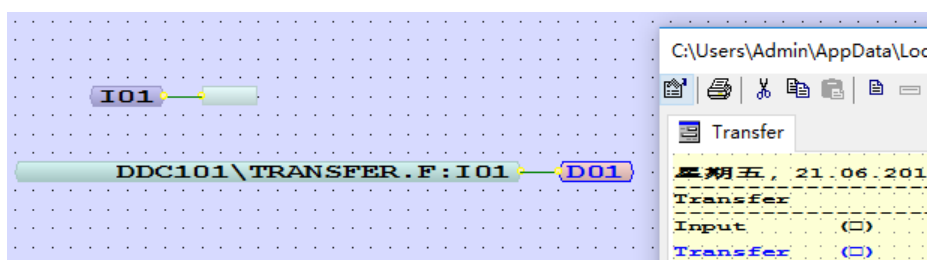
1. To transfer points between controllers, we can use the FUP module “Transfer Reference”. Now, we create a new FUP page called “transfer”. Add an “Input” and empty reference. Save and close the FUP page



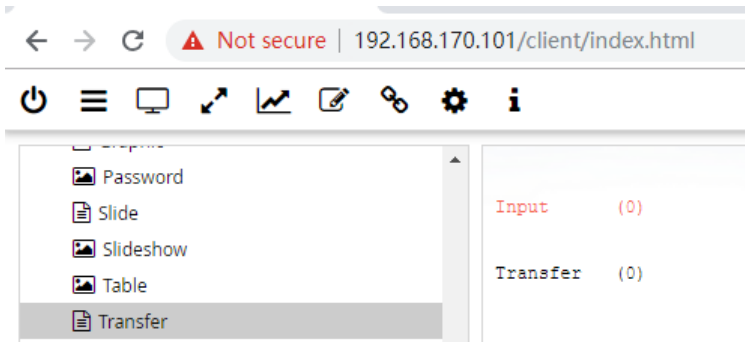
2. To read this “input” from another controller, we use “Transfer Reference” in the other controller FUP page. For testing purpose, here we use the same controller and same FUP page
3. So, we open the “transfer.f” FUP page again, and add the “Transfer Reference” module. Select the controller, FUP page, and the “Input” or “Display” you want to read



4. Please note that there is an “Interval (in seconds)” property at the bottom. This is how long the value is read from the controller. Leave it at default (15 seconds) for typical application (e.g. sending the outside air temperature to other controllers). Connect it to an “Display”. Compile and upload to the controller for testing (note: cannot test transfer reference in simulation)



- Now, when you change the “Input”, the “Transfer Reference” will change within 15 seconds. Please note that this “Transfer Reference” is read only.



- Important Note:** There is a limit of “An OPEN EMS controller can communicate with a maximum of 18 controllers as a data source via Transfer Reference”. For example, if you have an outside air temperature sensor in controller A, and you need to share this temperature to 20 controllers in the building, then you have to first send it to another controller B, and then use transfer reference in the other controllers to read the value from controller A or B.
- If you need to read/write the points from another controller, then you may consider using BACnet objects to transfer points between controllers. First, create the BACnet object for the “Input” in “BACnet Server” configuration (refer to TT190603 for details)

C:\Users\Admin\AppData\Local\DEOS\FUP XL\2\workspace\prj\Testing\DDC101 - External protocol - server - BACNET

File Extras Help

Main tree

- AHU
- Events
- Graphic
- Password
- Slide
- Transfer
- COSMOS IO modules

Input (0)

Transfer (0)

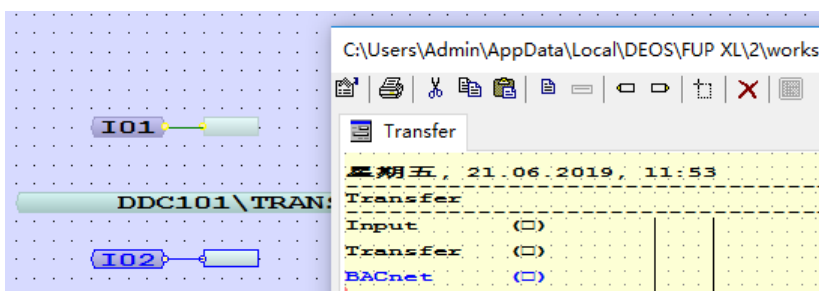
Number of BACnet objects: 11

COSMOS 600 OPEN

- COSMOS 600 OPEN
  - I
  - P
  - Setpoint
  - AHU Cooling Valve
  - AHU RA Temp
  - AHU Fan Control
  - AHU Supply Fan Trip
  - AHU Supply Fan Status
  - Input

object name	object identifier	property name	property value	var-type	writable by BACnet
COSMOS 600 OPEN	Device, 4194303	object_name	COSMOS 600 OPEN		
I	Analog Value, 0	present_value	AHU. F: IO3	FL	Yes
P	Analog Value, 1	present_value	AHU. F: IO2	FL	Yes
Setpoint	Analog Value, 2	present_value	AHU. F: IO1	FL	Yes
AHU Cooling Valve	Analog Value, 3	present_value	AHU. F: DO5	FL	No
AHU RA Temp	Analog Value, 4	present_value	AHU. F: DO3	FL	No
AHU Fan Control	Binary Value, 0	present_value	AHU. F: DO4	BIT	No
AHU Supply Fan Trip	Binary Value, 1	present_value	AHU. F: DO2	BIT	No
AHU Supply Fan Status	Binary Value, 2	present_value	AHU. F: DO1	BIT	No
Input	Binary Value, 3	present_value	TRANSFER. F: IO1	BIT	Yes

- Compile and upload to the controller. In the other controller, use the “BACnet Client” configuration to setup the connection (refer to TT180813 for details)
- Again, here we use the same controller for testing. Open “transfer.f”, and add an “Input” called “BACnet”



10. In “BACnet Client” configuration, first import the EDE file of the controller, and then drag and drop this point to the BACnet object called “Input”.

C:\Users\Admin\AppData\Local\DEOS\FUP XL\2\workspace\prj\Testing\DDC101 - External protocol - client - BACNET

File Extras Help

Main tree

- AHU
- Events
- Graphic
- Password
- Slide
- Transfer

Input (0)

Transfer (0)

BACnet (0)

Number of BACnet objects: 10

COSMOS 600 OPEN\_170101

object name	object identifier	property name	property value	var-type	read/write	read cycle (s)
COSMOS 600 OPEN_170101	Device, 170101	object_name	COSMOS 600 OPEN_170101			
I	Analog Value, 0	present_value	0			
P	Analog Value, 1	present_value	0			
Setpoint	Analog Value, 2	present_value	0			
AHU Cooling Valve	Analog Value, 3	present_value	0			
AHU RA Temp	Analog Value, 4	present_value	0			
AHU Fan Control	Binary Value, 0	present_value	0			
AHU Supply Fan Trip	Binary Value, 1	present_value	0			
AHU Supply Fan Status	Binary Value, 2	present_value	0			
Input	Binary Value, 3	present_value	TRANSFER.F:IO2	BIT	RPWP	15

11. Compile and upload to the controller. Now you can read/write the point in another controller via BACnet.

← → ↺ ⚠ Not secure | 192.168.170.101/client/index.html

⏻ ☰ 🖥️ ↗️ ↘️ 📝 🔗 ⚙️ ⓘ

Password

Slide

Slideshow

Table

Transfer

COSMOS IO modules

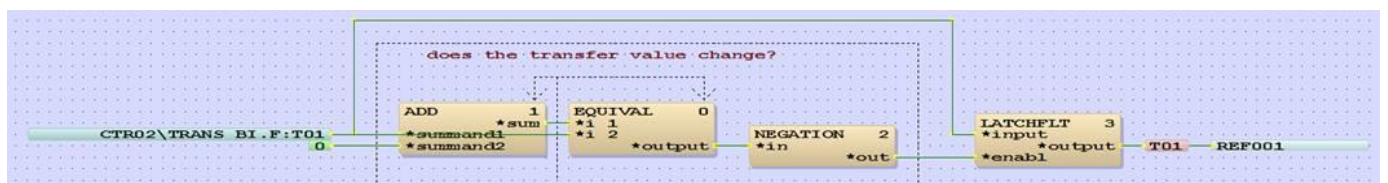
Input (1)

Transfer (1)

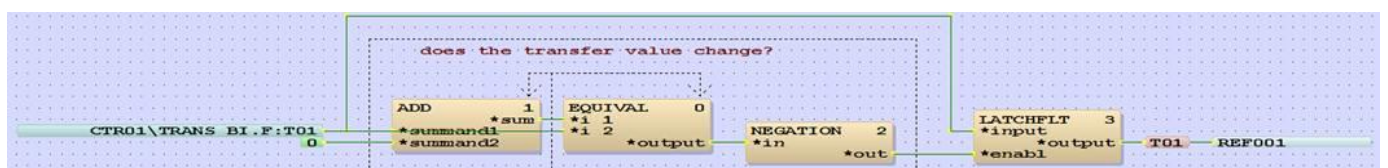
BACnet (1)

12. There is a way to read/write points from another controller using “Transfer Reference”, but it’s a bit complicate so it’s for your reference only.

In controller “CTR01”, create a FUP page “trans\_bi.f”, like this



In controller “CTR02”, create a FUP page “trans\_bi.f”, like this



Note: The Display “T01” is set to “Read and Write”

The “Number” of the FUP module “ADD” and “EQUIVAL” must be reversed