KNX Cookbook

2

Development Tools

4

Manufacturer Tool

1

Summary

This document is a development help for KNX newcomers.

This document describes the features and use of the KNX Manufacturer Tool.

This document is part of the KNX Specifications v2.1.

Document updates

Version	Date	Modifications
1.0.0	2011.05.13	Preparation of v1.0.0.
01.00.01	2013.10.14	Editorial updates for the publication of KNX Specifications 2.1.

References

[01]	Chapter 3/1/2	“Glossary”
[02]	Volume 6	“Profiles”

Filename:	02_04_01 Manufacturer Tool v01.00.01.docx
Version:	01.00.01
Status:	Final version
Savedate:	2013.10.14
Number of pages:	19

Contents

1 General

1.1 MT4

- stands for “Manufacturer Tool, Version 4”;
- is set up as MS Visual Studio 2008 package; and
- can be installed as standalone tool or integrated into an existing MS Visual Studio 2008 installation.

1.2 Repository

The repository of MT4 is based on XML, there is no database whatsoever involved.

1.3 XML

MT4 does not only store its data as XML, also its entire KNX knowledge is stored as a XML. This collection of XML files is called the “KNX XML Schema”. The advantage of this approach is that in a lot of cases the tool can be updated without breaking its code.

1.4 Input format

Also the input format of MT4 is based on XML.

1.5 Output format

The output format of MT4 depends on the target version of the project:

- XML if the target = ETS4;
- VD3, VD4 or VD5 if the target was set to resp. ETS3.0d, ETS3.0e or ETS3.0f.

2 ETS product

2.1 Set up

ETS Products are set up as follows: Solution -> Project(s) -> catalog structure, hardware & product elements, application program(s).

NOTE 1 Product = a commercial realization of hardware

2.2 Parameters

2.2.1 Overview

Device parameters in MT4 are implemented in a very specific way:

Structure: “ParameterRefRef” -> “ParameterRef” -> “Parameter”

- “Parameter” is the actual parameter as it stored in EEPROM
- “ParameterRef” refers directly to “Parameter”, it offers the possibility to override values and properties

  EXAMPLE 1 A parameter linked to a particular rocker might need different default values when set up for switching or dimming.
- “ParameterRefRef” refers directly to “ParameterRef” and is the resource that will be used to visualize “Parameter” in the ETS parameter dialog.
- “ParameterRefRef” is automatically created when its associated “ParameterRef” is added to the dynamic structure of the AP (see further).
- “Parameter” can have >1 associated “ParameterRef”, regardless of the target version.
- If the target version < ETS4.0 then “ParameterRef” can only have 1 associated “ParameterRefRef”
- If the target version = ETS4.0 then “ParameterRef” can have >1 associated “ParameterRefRef”.

2.2.2 Memory unions

- Are a way to save device memory.
- Put parameters that are not active at the same time, together on the same memory location.

2.3 Group Objects

- In the tool named Communication Objects (COs).
- COs in MT are implemented in the same way as parameters.

  Structure: “ComObjectRefRef” -> “ComObjectRef” -> “ComObject”

  - “ComObject” is the actual CO as it “resides” inside the device.
  - “ComObjectRef” refers directly to “ComObject”, it offers the possibility to override properties.

    EXAMPLE 2 A particular CO could need, depending on the “situation” different DPTs.
  - “ComObjectRefRef” refers directly to “ComObjectRef” and is the resource that will be used to visualize the “ComObject” in the ETS device detail dialog.
  - “ComObjectRefRef” is automatically created when its associated “ComObjectRef” is added to the dynamic structure of the AP (see further).
  - “ComObject” can have >1 associated “ComObjectRef”, regardless of the target version.

- If the target version < ETS4.0 then “ComObjectRef” can only have 1 associated “ComObjectRefRef”.
- If the target version = ETS4.0 then “ComObjectRef” can have >1 associated “ComObjectRefRef”.

2.4 AP

2.4.1 Model

An AP consists out of two parts; a Static - and a Dynamic part.

2.4.2 Static part

The Static part contains the following.

- Code Segments
- Parameter Types
- Parameters/Memory Unions
- ParameterRefs
- ComObjects
- ComObjectRefs
- Address Table
- Association Table
- Options

2.4.3 Dynamic part

The Dynamic part represents the logic behind the ETS Parameter dialog:

- parameter structure (tree),
- parameter dependencies,
- CO dependencies.

The entire Dynamic part of an AP is set up as a collection of Channels.

3 Channel concept

3.1 Introduction

In the context of MT4 a “Channel” is a resource to collect an individual and independent structure of:

- “Choose/When” elements: comparable to if/then statements
- ETS parameter dialog pages, called ParameterBlocks
- Parameters
- COs

NOTE 2 A Channel doesn’t necessarily correspond with a physical resource of the device, like e.g. a rocker or a switch contact. In practice however, it most probably will often be used like that.

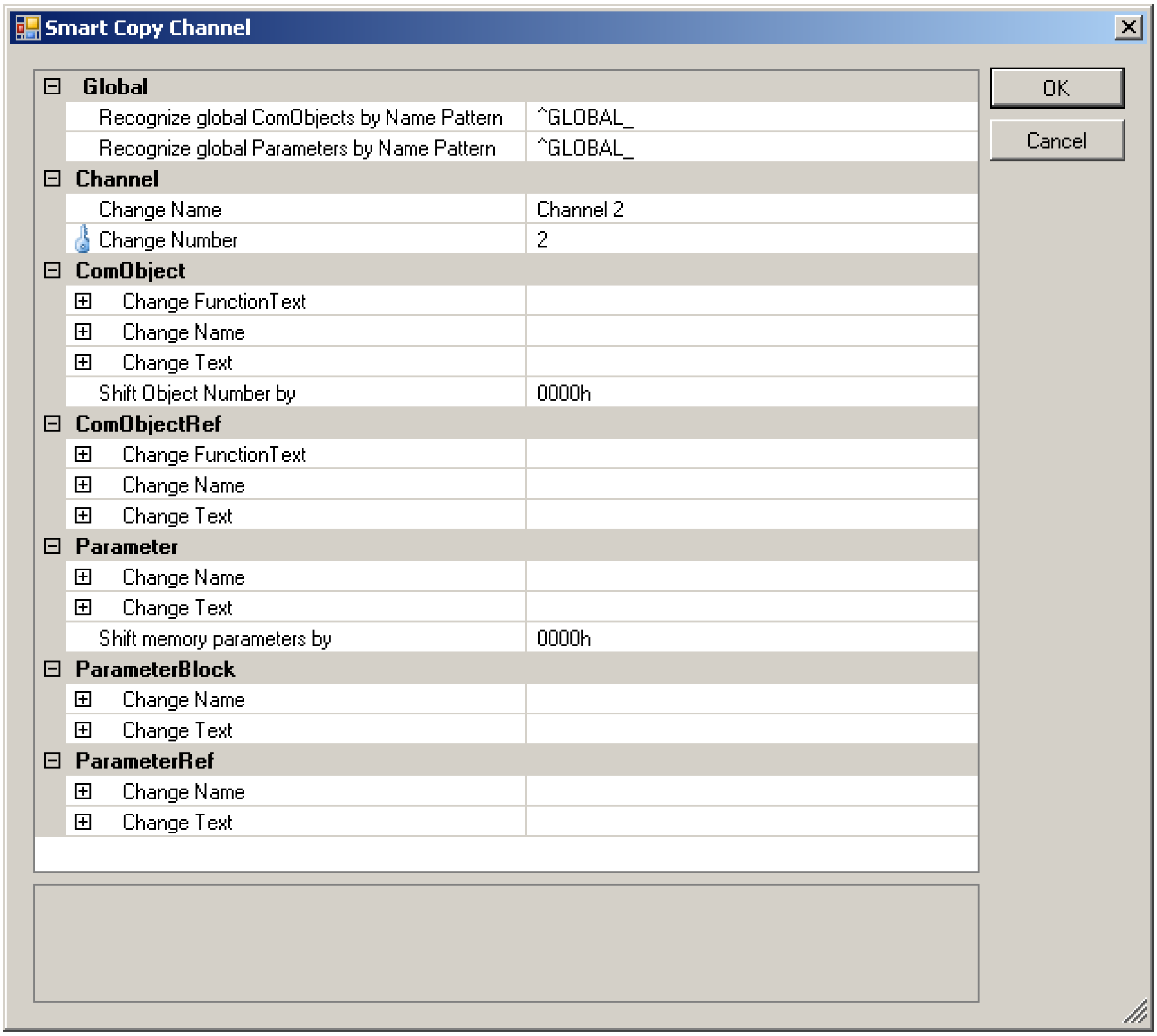
The biggest advantage of the MT4 Channel concept is that it allows “Smart Copy” & “Smart Update”. Imagine the situation where a rather complex choose/when structure with a lot of parameters has been set up and where several - nearly - the same copies of that structure are required.

With classic copy & paste operations this would mean a lot of repetitive work, like:

- changing parameter names,
- shifting CO number and
- shifting parameter memory locations.

3.2 Smart copy

- Please refer to the screenshot in Figure 1.
- Copy + Paste Special
  - allows to automatically change/replace data while a copy of the entire channel is being made
  - takes into account global parameters and/or COs: are taken over in the copy of the channel without any modification
  - can shift Object numbers
  - can shift parameter memory locations



⊟ **Global**	
Recognize global ComObjects by Name Pattern	^GLOBAL_
Recognize global Parameters by Name Pattern	^GLOBAL_
⊟ **Channel**	
Change Name	Channel 2
Change Number	2
⊟ **ComObject**	
⊞ Change FunctionText	
⊞ Change Name	
⊞ Change Text	
Shift Object Number by	0000h
⊟ **ComObjectRef**	
⊞ Change FunctionText	
⊞ Change Name	
⊞ Change Text	
⊟ **Parameter**	
⊞ Change Name	
⊞ Change Text	
Shift memory parameters by	0000h
⊟ **ParameterBlock**	
⊞ Change Name	
⊞ Change Text	
⊟ **ParameterRef**	
⊞ Change Name	
⊞ Change Text	

Figure 1 – Smart copy dialog

3.3 Smart update

- Only changes/replaces data, no copy of the resource is made.
- Can only be used for individual resources, not for an entire channel.

ETS3: channels are also supported for projects with target version = ETS3, the channel information will however not be “carried over” in the vd* file.

4 KNX Converter and MT4 Import

4.1 Situation

As already mentioned above MT4 can only import (open) XML based data. There are two possibilities.

- MT4 Project
- MT4 Product = output from a MT4 Project with ETS target version set to ETS4

Products created with MT3 can also be imported, but not directly: vd3, vd4 or vd5 files need to be converted with KNX Converter into a MT4 Product first; the result is a knxprod file. Then this knxprod file needs to be converted within MT4 into a MT4 Project. The ETS Target version of such MT4 Project is always set to ETS4. It is however possible to change its target version into ETS3, as described in clause 4.2 “Workflow (tasks)”.

KNX Converter cannot convert products created with MT2.

KNX Converter comes installed with MT4, typically under C:\Program Files\KNX Manufacturer Tool 4.0\Converter.

4.2 Workflow (tasks)

4.2.1 OEM

- The prerequisite is a completed product (but not necessarily registered).
- Create a new project using the “OEM product” template and follow further instructions.
- Define the new manufacturer.
- The resulting product is an unregistered copy.
- Submit the OEM product for registration

4.2.2 Registration

- prerequisite = completed product
- select the AP -> Edit/Registration (via the MT4 menu)
- open the template: \KNX Certification\KNX_certification_forms_ETS4.xltx -> MS Excel
- open within this template the knxei file from the “out” folder of the MT4 project
- edit the resulting xlsx file (if required) + save
- send the knxei + xlsx file to KNX
- KNX Association signs the product data & sends back a knxrcd file
- import the knxrcd file -> this will mark your product as registered

4.2.3 Translation

- prerequisite = completed product
- select the AP -> Edit/Translation (via the MT4 menu)
- select the target language(s)
- select the required format: ETA or XLIFF
- translate with a 3rd party tool
- import the translated texts
- use the MT4 preview function to check the translation
- (small) corrections can be made within MT4

4.2.4 Create product databases

- prerequisite = registered product
- which however has to be error free -> Build/Check (via the MT4 menu)
- build the project
- result is to be found in the “out” folder of the project, depending on the target version this is a: vd3, vd4, vd5 (ETS3) or knxprod file (ETS4)

4.2.5 Edit an existing product

- product created with MT2: not supported
- product created with MT3
  - use KNX Converter to convert the vd3/4/5 file into a knxprod file
  - create a new project: File/New/Project
    - use type = “Convert knxprod file to KNX MT Project”
    - target is automatically set to **ETS4**
  - create a second project (within the same solution)
    - but use type = “KNX Manufacturer Tool Project “
    - and set target = **ETS3**
  - copy the desired files from the “ETS4” project” to the “ETS3” project via the menu Project/Add existing item
  - open the AP and select "Prepare for targeting ETS3" from the context menu
  - then change at least the application number, in order to avoid consistency conflicts between the AP from the “ETS4” project (=original) and the one from the “ETS3” project (=new)
  - edit the data as usual
- ETS3 product created with MT4
  - use the original project (MTXML) files
  - remember: importing vd3/4/5 files is not possible
- ETS4 product

  Two possibilities:
  - use the original project (MTXML) files
  - alternative: (re)import the knxprod file

5 Exercises

5.1 Goal

At the end of this document a special clause with exercises is foreseen in order to get use to the first steps in MT4.

These exercises are pure MT4, there is no external development involved, like e.g. creating an s19 file.

If you do not have any MT4 experience yet, then it is recommended having a look at these exercises before you go to the next section (Solutions).

5.2 Exercise 1: Base

5.2.1 Description

This exercise builds up the following AP in 4 parts:

- 4 fold push button
- dimming
- Parameters / channel (rocker) :
    - User level : base / advanced
    - Min dim value
    - Timeframe till max -> only if user level = advanced
- Communication Objects / channel (rocker)
    - 1-bit on/off object
    - 4-bit dim object

5.2.2 Part1: Add project

- KNX Manufacturer Tool Project
- Name = KNXproduct100203-01
- Target ETS Version = ETS3.0f

Add: Hardware

- Serial Number = SN03.02.2010-01
- Name = PCB SN03.02.2010-01
- Version = 1

Define Product (= commercial realization of HW)

- Order Number = 4f-PB 03.02.2010-01
- Product Name = 4 fold Push Button V1.0
- Language

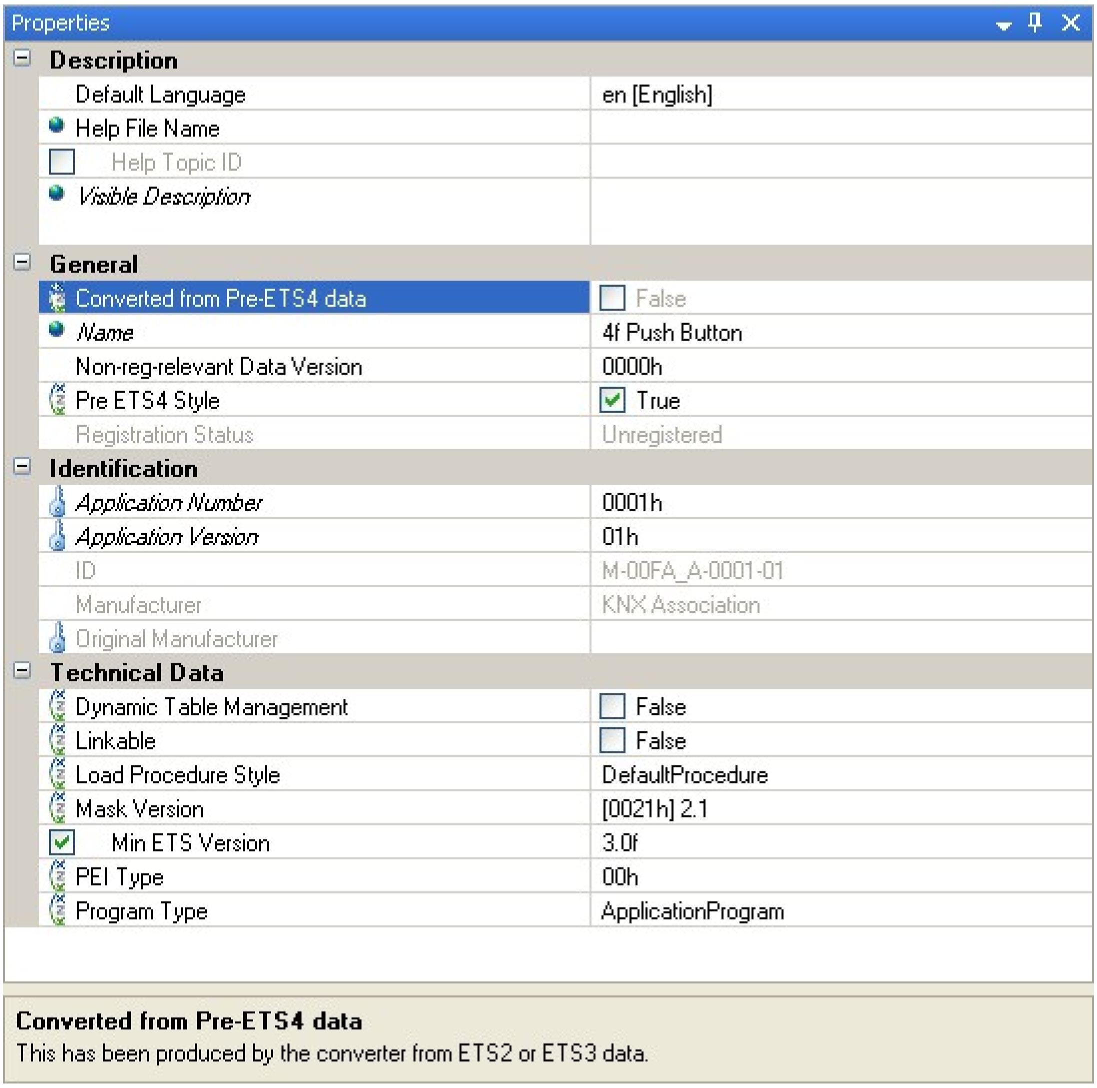
Add: Application Program

- Number = 1 & Version = 1
- Name = 4f Push Button
- Mask Version = 2.1

Open Hardware

- select Application Programs
- add new Hardware2Program
- Select 4f Push Button

Result



Description	
Default Language	en [English]
Help File Name	
☐ Help Topic ID	
Visible Description	
General	
Converted from Pre-ETS4 data	☐ False
Name	4f Push Button
Non-reg-relevant Data Version	0000h
Pre ETS4 Style	☑ True
Registration Status	Unregistered
Identification	
Application Number	0001h
Application Version	01h
ID	M-00FA_A-0001-01
Manufacturer	KNX Association
Original Manufacturer	
Technical Data	
Dynamic Table Management	☐ False
Linkable	☐ False
Load Procedure Style	DefaultProcedure
Mask Version	[0021h] 2.1
☑ Min ETS Version	3.0f
PEI Type	00h
Program Type	ApplicationProgram

Converted from Pre-ETS4 data
This has been produced by the converter from ETS2 or ETS3 data.

5.2.3 Part2: Add Parameter Types

- Restriction : name = TypeEnumUserLevel , range = base , advanced
- Number : name = TypeNumberMinDimValue , 8 bit, range = 15..100 %
- Number : name = TypeNumberDimTimeFrame , 8 bit, range = 1..10 seconds

Add Parameters

- Virtual : ParUserLevel , default = base
- Memory : ParMinDimValue01 , default = 75
- Memory : ParDimTimeFrame01 , default = 3

Add Group Objects

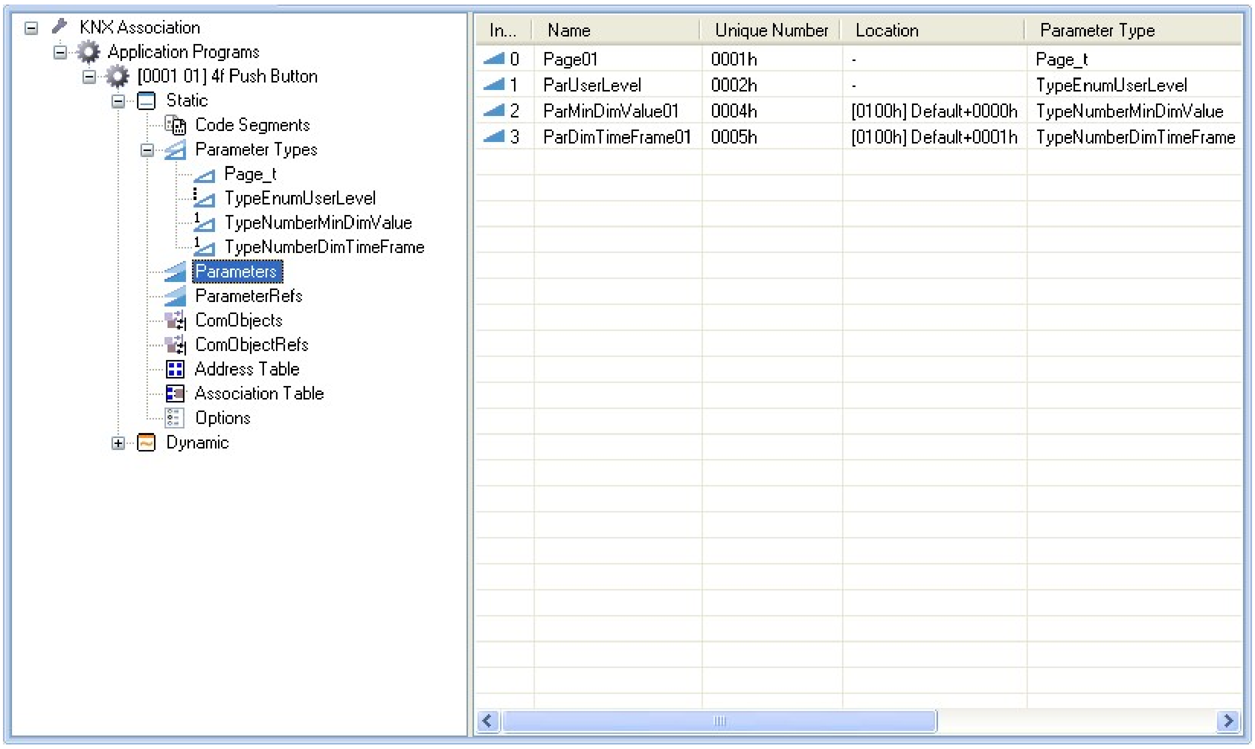
- Object #0 : SW01 , switch object , 1 bit, DPT 1.1
- Object #1 : DM01 , dim object , 4 bit, DPT 3.7

Check

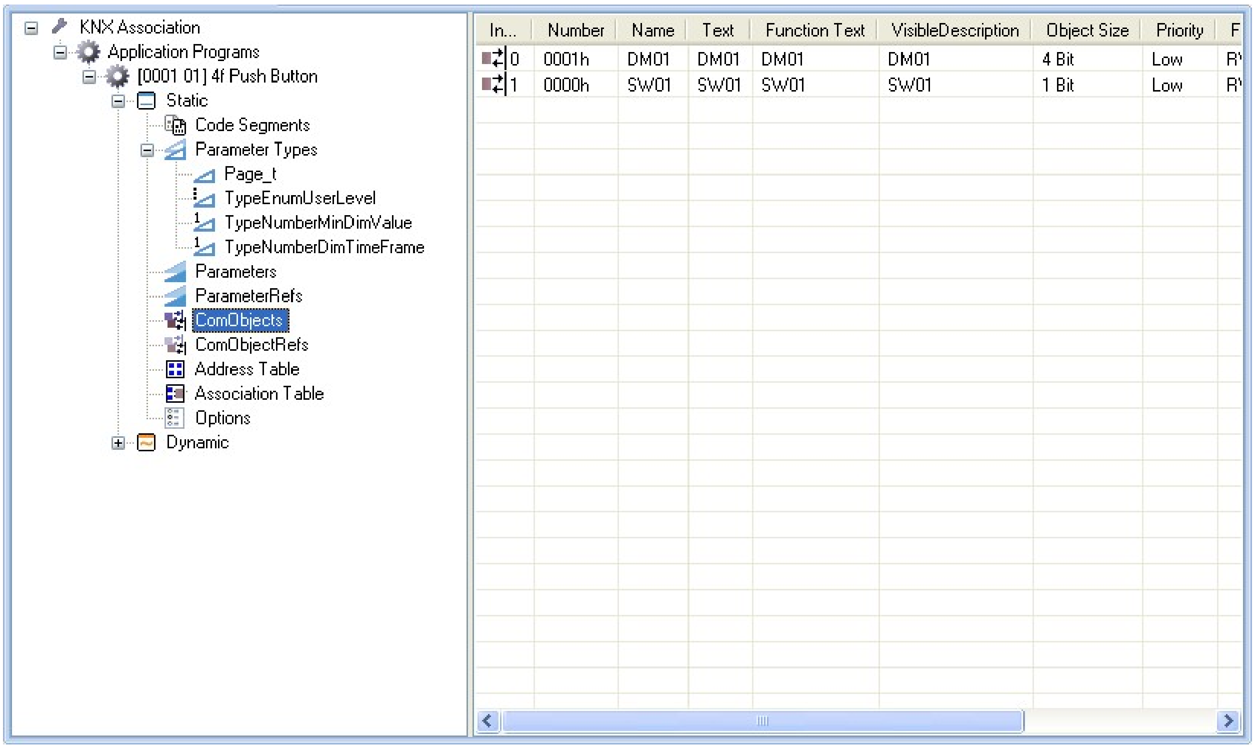
- shift memory location for ParDimTimeFrame01 (to avoid overlap)

Preview

Result



In...	Name	Unique Number	Location	Parameter Type
0	Page01	0001h	-	Page_t
1	ParUserLevel	0002h	-	TypeEnumUserLevel
2	ParMinDimValue01	0004h	[0100h] Default+0000h	TypeNumberMinDimValue
3	ParDimTimeFrame01	0005h	[0100h] Default+0001h	TypeNumberDimTimeFrame



In...	Number	Name	Text	Function Text	VisibleDescription	Object Size	Priority	F
0	0001h	DM01	DM01	DM01	DM01	4 Bit	Low	R'
1	0000h	SW01	SW01	SW01	SW01	1 Bit	Low	R'

5.2.4 Part3: Channel 1

- Add Choose / When -> ParUserLevel to Page01
- Add “base”
- Add “advanced”
- Delete “Page1”

Channel 1 / “base”

- Add ParameterBlock : Name = Page01 , Text = page 01

Channel 1 / “base” / Page01

- Add ParMinDimValue01
- Add SW01
- Add DM01

Channel 1 / “base” / Page01

- Copy
- Paste into Channel 1 / “advanced”

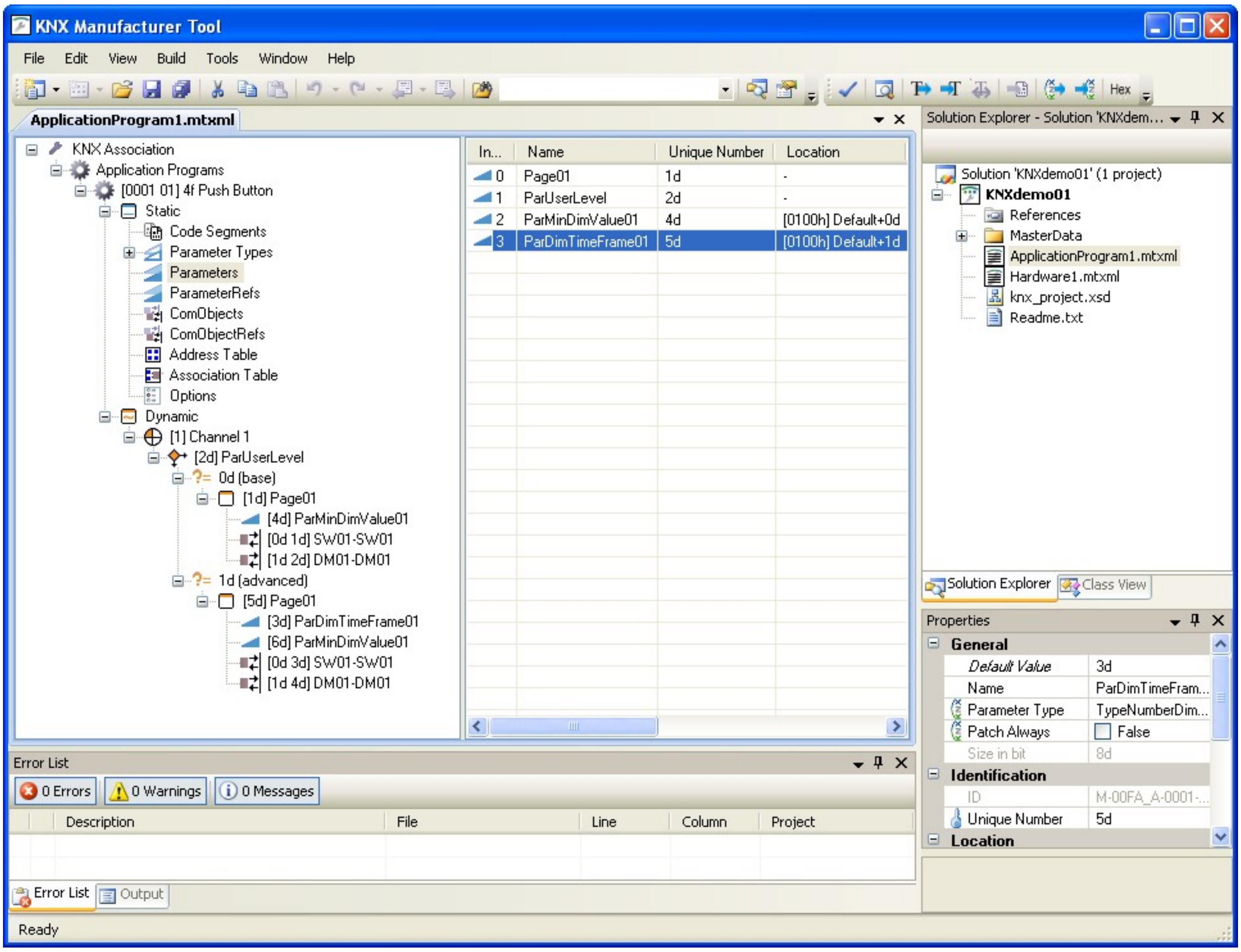
Channel 1 / “advanced” / Page01

- Add ParMinTimeFrame01

Check

Preview

Result



In...	Name	Unique Number	Location
0	Page01	1d	-
1	ParUserLevel	2d	-
2	ParMinDimValue01	4d	[0100h] Default+0d
3	ParDimTimeFrame01	5d	[0100h] Default+1d

Properties

General	
Default Value	3d
Name	ParDimTimeFram...
Parameter Type	TypeNumberDim...
Patch Always	□ False
Size in bit	8d
Identification	
ID	M-00FA_A-0001-...
Unique Number	5d
Location	

5.2.5 Part4: copy Channel 1

Paste Special

- Global
  - ^ParUserLevel
- Channel
  - Accept default (“Channel 2”, 2)
- ComObject
  - Change Function Text, Name, Text : 01-> 02..04
  - Shift Object Number by 2
- Parameter
  - Change Name, Text : 01 -> 02..04
  - Shift memory by 2 (bytes)

- ParameterBlock
    - Change Name, Text : 01 -> 02..04

Check

Add (main) Channel

- Number = e.g. 10 (preferable not in the range of the "normal" channels)
- Name = Main
- Add (main) ParameterBlock : Name = Main , Text = main page
- Add ParUserLevel to Main Channel/Main Page

Preview

5.3 Exercise 2: Extend exercise 1 with Blind / Shutter functionality

5.3.1 Part 1: Delete channels 2, 3, 4

- Including the Parameters and ParametersRefs from these channels
- Including the ComObjects and ComObjectRefs from these channels

Add Parameter Types

- Restriction : name = TypeEnumFunction , 8 bit, range = dimming , blinds
- Restriction : name = TypeEnumStepSize , 8 bit, range = 1/64, 1/32, 1/16, 1/8
- Number : name = TypeNumberLongPushTimeFrame , 8 bit, range = 3..100 x 10ms

Add Parameters

- virtual : name = ParFunction01 , default = dimming
- memory : name = ParStepSize01 , default = 1/32
- memory : name = ParLongPushTimeFrame01 , default = 50

Add Group Objects

- Object #2 : name = MV01 , move object, 1 bit, DPT 1.008
- Object #3 : name = ST01 , step/stop object, 1 bit, DPT 1.007

Check

Probably you need to shift the offset from the new parameters

5.3.2 Part 2: Channel 1

Delete existing structure for "advanced"

- make however sure not to remove any Parameters or ComObjects
- ParameterRefs and ComObjectRefs can be deleted

Build up "base"

- Add ParFunction01 to Page01
- Add Choose based on ParFunction01

- Add When for “dimming”
    - Add (move) ParMinDimValue01
    - Add (move) SW01
    - Add (move) DM01
- Add When for “blinds”
    - Add ParStepSize01
    - Add MV01
    - Add ST01

Copy Channel 1 / “base” / Page01

Paste into Channel 1 / “advanced”

Edit Channel 1 / “advanced” / Page01

- Add ParDimTimeFrame01 for “dimming”
- Add ParLongPushTimeFrame01 for “blinds”

5.3.3 Part 3: Copy Channel 1

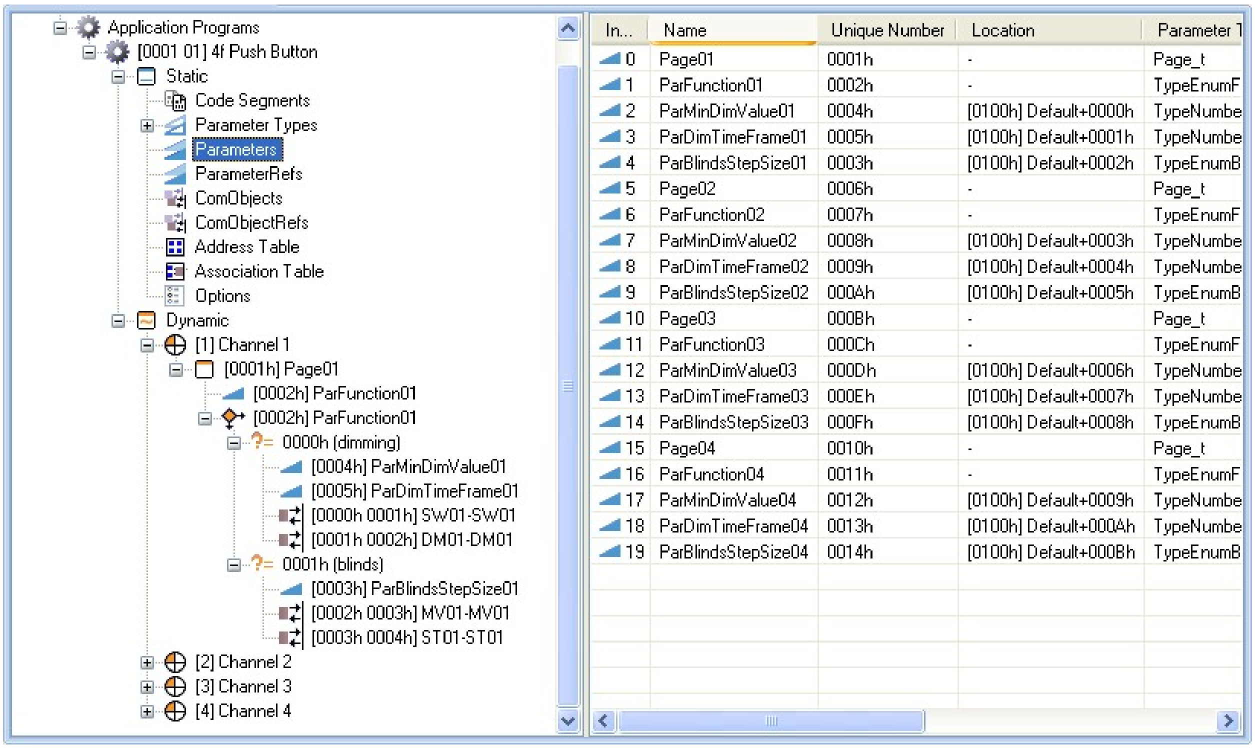
Paste Special

- Global
    - ^ParUserLevel
- Channel
    - Accept default (“Channel 2”, 2)
- ComObject
    - Change Function Text, Name, Text : 01-> 02..04
    - Shift Object Number by 4/8/12 (!!)
- Parameter
    - Change Name, Text : 01 -> 02..04
    - Shift memory by 4/8/12 (bytes)
- ParameterBlock
    - Change Name, Text : 01 -> 02..04

Check

Preview

Result:



```
Application Programs
  [0001 01] 4f Push Button
    Static
      Code Segments
      Parameter Types
      Parameters
      ParameterRefs
      ComObjects
      ComObjectRefs
      Address Table
      Association Table
      Options
    Dynamic
      [1] Channel 1
        [0001h] Page01
          [0002h] ParFunction01
          [0002h] ParFunction01
            0000h (dimming)
              [0004h] ParMinDimValue01
              [0005h] ParDimTimeFrame01
              [0000h 0001h] SW01-SW01
              [0001h 0002h] DM01-DM01
            0001h (blinds)
              [0003h] ParBlindsStepSize01
              [0002h 0003h] MV01-MV01
              [0003h 0004h] ST01-ST01
      [2] Channel 2
      [3] Channel 3
      [4] Channel 4
```

In...	Name	Unique Number	Location	Parameter T
0	Page01	0001h	-	Page_t
1	ParFunction01	0002h	-	TypeEnumF
2	ParMinDimValue01	0004h	[0100h] Default+0000h	TypeNumbe
3	ParDimTimeFrame01	0005h	[0100h] Default+0001h	TypeNumbe
4	ParBlindsStepSize01	0003h	[0100h] Default+0002h	TypeEnumB
5	Page02	0006h	-	Page_t
6	ParFunction02	0007h	-	TypeEnumF
7	ParMinDimValue02	0008h	[0100h] Default+0003h	TypeNumbe
8	ParDimTimeFrame02	0009h	[0100h] Default+0004h	TypeNumbe
9	ParBlindsStepSize02	000Ah	[0100h] Default+0005h	TypeEnumB
10	Page03	000Bh	-	Page_t
11	ParFunction03	000Ch	-	TypeEnumF
12	ParMinDimValue03	000Dh	[0100h] Default+0006h	TypeNumbe
13	ParDimTimeFrame03	000Eh	[0100h] Default+0007h	TypeNumbe
14	ParBlindsStepSize03	000Fh	[0100h] Default+0008h	TypeEnumB
15	Page04	0010h	-	Page_t
16	ParFunction04	0011h	-	TypeEnumF
17	ParMinDimValue04	0012h	[0100h] Default+0009h	TypeNumbe
18	ParDimTimeFrame04	0013h	[0100h] Default+000Ah	TypeNumbe
19	ParBlindsStepSize04	0014h	[0100h] Default+000Bh	TypeEnumB

5.4 Exercise 3: Optimize memory usage - Memory Unions

5.4.1 Part 1: Delete channel 2, 3, 4 from exercise 2

Add an 8-bit Memory Union (Parameters) for address [0100] + 0

- Add ParMinDimValue01 -- type TypeNumberMinDimValue
- Add ParStepSize01 -- type TypeEnumStepSize

Add an 8-bit Memory Union (Parameters) for address [0100] + 1

- Add ParDimTimeFrame01 -- type TypeNumberDimTimeFrame
- Add ParLongPushTimeFrame01 -- type TypeNumberLongPushTimeFrame

Delete from the “regular” parameter list

- ParMinDimValue01
- ParStepSize01
- ParDimTimeFrame01
- ParLongPushTimeFrame01

Channel 1 / “base” / Page01 / “dimming”

- Add ParMinDimValue01

Channel 1 / “base” / Page01 / “blinds”

- Add ParStepSize01

Delete Channel 1 / “advanced”

Copy Channel 1 / “base” / Page01

Paste into Channel 1 / “advanced”

Channel 1 / “advanced” / Page01 / “dimming”

- Add ParStepSize01

Channel 1 / “advanced” / Page01 / “blinds”

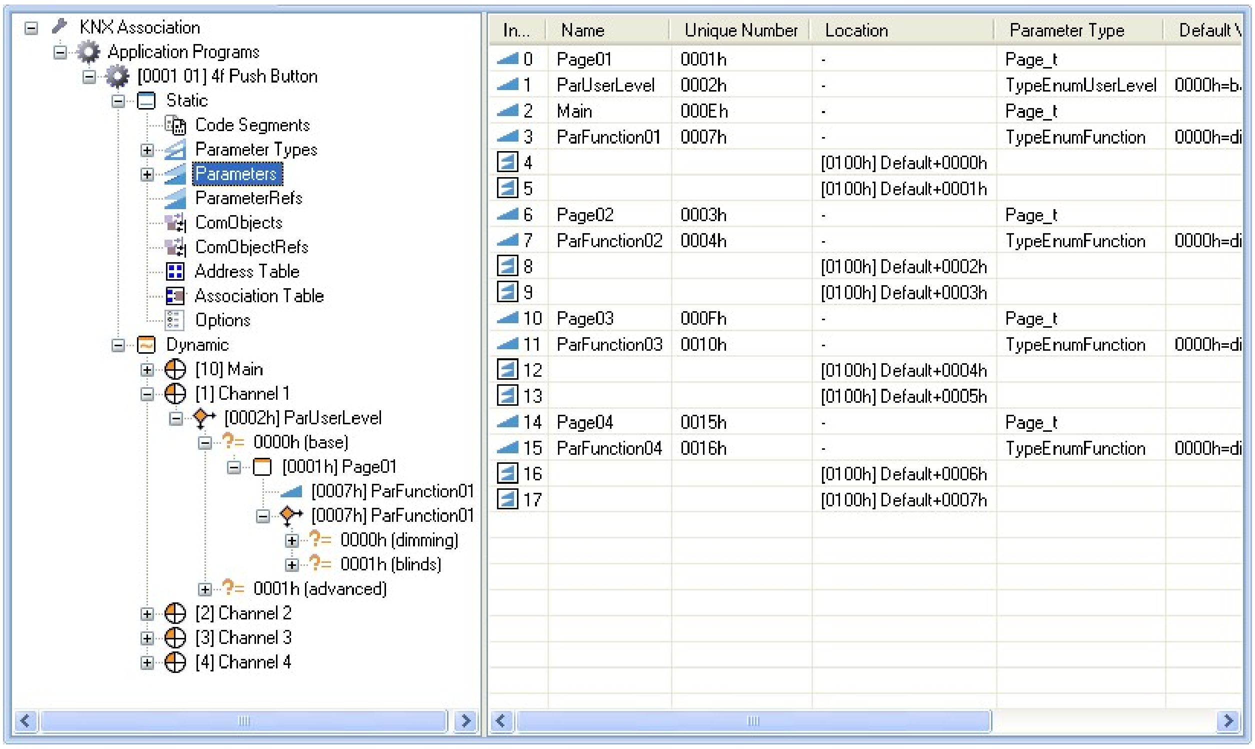
- Add ParLongPushTimeFrame01

5.4.2 Part 2: Copy Channel 1

Paste Special

- As above
- But only shift memory of parameters by 2/4/6 bytes
- # objects still by 4/8/12

Result



- KNX Association
  - Application Programs
    - [0001 01] 4f Push Button
      - Static
        - Code Segments
        - Parameter Types
        - Parameters
        - ParameterRefs
        - ComObjects
        - ComObjectRefs
        - Address Table
        - Association Table
        - Options
      - Dynamic
        - [10] Main
        - [1] Channel 1
          - [0002h] ParUserLevel
            - 0000h (base)
              - [0001h] Page01
                - [0007h] ParFunction01
                - [0007h] ParFunction01
                  - 0000h (dimming)
                  - 0001h (blinds)
            - 0001h (advanced)
        - [2] Channel 2
        - [3] Channel 3
        - [4] Channel 4

In...	Name	Unique Number	Location	Parameter Type	Default \
0	Page01	0001h	-	Page_t	
1	ParUserLevel	0002h	-	TypeEnumUserLevel	0000h=b.
2	Main	000Eh	-	Page_t	
3	ParFunction01	0007h	-	TypeEnumFunction	0000h=di
4			[0100h] Default+0000h		
5			[0100h] Default+0001h		
6	Page02	0003h	-	Page_t	
7	ParFunction02	0004h	-	TypeEnumFunction	0000h=di
8			[0100h] Default+0002h		
9			[0100h] Default+0003h		
10	Page03	000Fh	-	Page_t	
11	ParFunction03	0010h	-	TypeEnumFunction	0000h=di
12			[0100h] Default+0004h		
13			[0100h] Default+0005h		
14	Page04	0015h	-	Page_t	
15	ParFunction04	0016h	-	TypeEnumFunction	0000h=di
16			[0100h] Default+0006h		
17			[0100h] Default+0007h		