

# Bedienungsanleitung KNXprodEditor V0.41

## Inhalt

1	Grundsätzliche Einstellungen für die Nutzung.....	2
2	Laden einer knxprod Datei .....	2
3	Dynamic Section: Parameter und KO .....	4
4	Parameter Typen.....	5
5	Parameter.....	7
6	Kommunikationsobjekte .....	10
7	Code.....	11
8	Eine Auswahlmöglichkeit erstellen .....	11
9	HowTo: Eine komplett neue knxprod erstellen.....	12
9.1	Parametertypen anlegen.....	15
9.2	Parameter anlegen .....	17
9.3	Erstellen von Kommunikationsobjekten .....	18
9.4	Anlegen des dynamischen Baumes .....	19
9.5	Fertig stellen und in die ETS laden .....	22
9.6	Das ARM Programm zur selbst erstellten knxprod .....	24

## 1 Grundsätzliche Einstellungen für die Nutzung

Für die Signierung von veränderten knxprod Dateien muss die ConverterEngine.dll aus dem ETS Ordner ausgewählt werden.

Es wurden folgende Dateien erfolgreich geprüft:

C:\Program Files (x86)\ETS5\CV\4.0.1477.15773\Knx.Ets.Converter.ConverterEngine.dll

C:\Program Files (x86)\ETS5\CV\4.0.1997.50261\Knx.Ets.Converter.ConverterEngine.dll

C:\Program Files (x86)\ETS5\CV\5.0.204.12971\Knx.Ets.Converter.ConverterEngine.dll

C:\Program Files (x86)\ETS5\CV\5.1.84.17602\Knx.Ets.Converter.ConverterEngine.dll

C:\Program Files (x86)\ETS5\CV\5.6.241.33672\Knx.Ets.Converter.ConverterEngine.dll

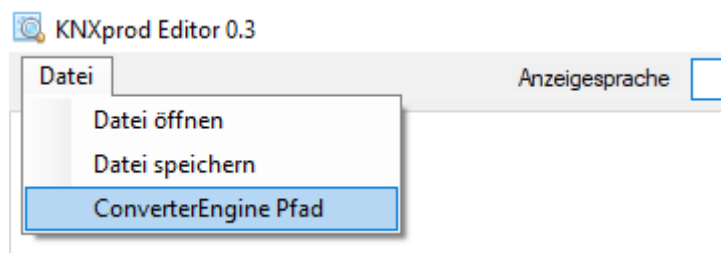


Abbildung 1: Pfad der ConverterEngine.dll eingeben

Eine dieser Dateien muss unter dem in Abbildung 1 gezeigten Button ausgewählt werden. Dieses muss für einen Nutzer an einem Rechner nur einmalig gemacht werden, da der Pfad zur DLL im Userkonto abgelegt wird.

## 2 Laden einer knxprod Datei

Für die Betrachtung und eventuelle Veränderung einer knxprod Datei muss diese über den Button „Datei öffnen“ geöffnet werden.

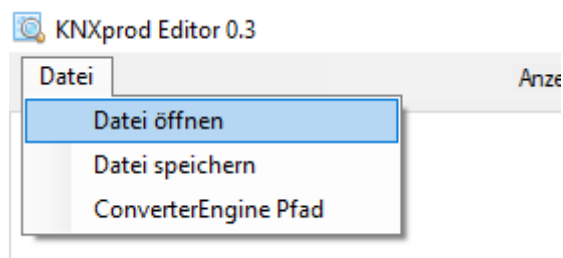


Abbildung 2: Eine knxprod Datei öffnen

Im Hintergrund wird die knxprod Datei entpackt und in einen Ordner „unzippedKnxprod“ zwischengespeichert. Dieser Ordner wird an dem Ort erstellt, an dem die KNXprodEditor.exe Datei geöffnet wurde.

Die knxprod Datei enthält immer eine knx\_master.xml, in der alle Datentypen, Hersteller und sonstige übergeordnete Dinge stehen.

In einem Ordner, der als Namen die ID des Herstellers trägt, sind unter anderem die Dateien Catalog.xml und Hardware.xml.

Der Einstiegspunkt ist die Catalog.xml, die im KNXprodEditor als erstes geöffnet wird.

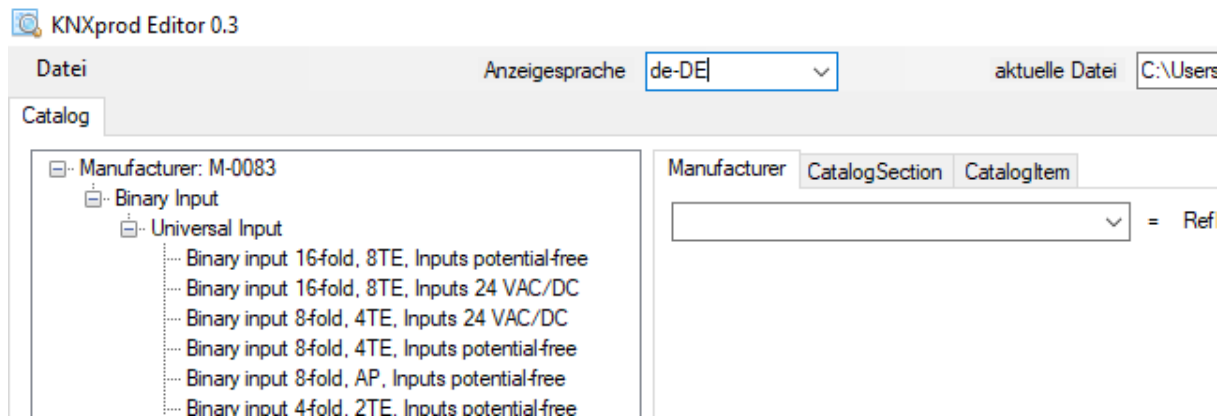


Abbildung 3: Catalog.xml geöffnet

Hier sollte nach Möglichkeit zuerst die Sprache gewählt werden, in der alle übersetzbaren Elemente angezeigt werden sollen.

Anschließend kann in der Baumansicht links durch die einzelnen Elemente navigiert werden und es können verschiedenste Parameter eingesehen und verändert werden.

Grundsätzlich ist nach jeder Änderung auf einer Seite der speichern Button zu betätigen, ansonsten sind die Änderungen verloren.

Soll der Hersteller verändert werden, muss man dieses direkt nach dem Laden der knxprod Datei machen, da diese Änderung sich über alle Dateien und Ordner erstreckt und die komplette knxprod Datei umgeschrieben und anschließend neu geöffnet wird. Daher also alle Änderungen vorher als knxprod abspeichern, diese erneut öffnen und den Hersteller ändern.

Sobald man ein Gerät auswählt (in Abbildung 3 z.B. Binary Input 16-fold, 8TE, Inputs potential-free), wird die dazugehörige Hardware in einem neuen Tab geöffnet.



Abbildung 4: Hardware geöffnet

Im Hardware Tab können wieder alle Elemente des Baumes an der linken Seite ausgewählt werden und gegebenenfalls geändert werden.

Sobald das ApplikationProgram (in der Abbildung 4 M-0083\_A-0030-20-FCCB) ausgewählt wurde, wird die Applikation in einem neuen Tab geöffnet.

### 3 Dynamic Section: Parameter und KO

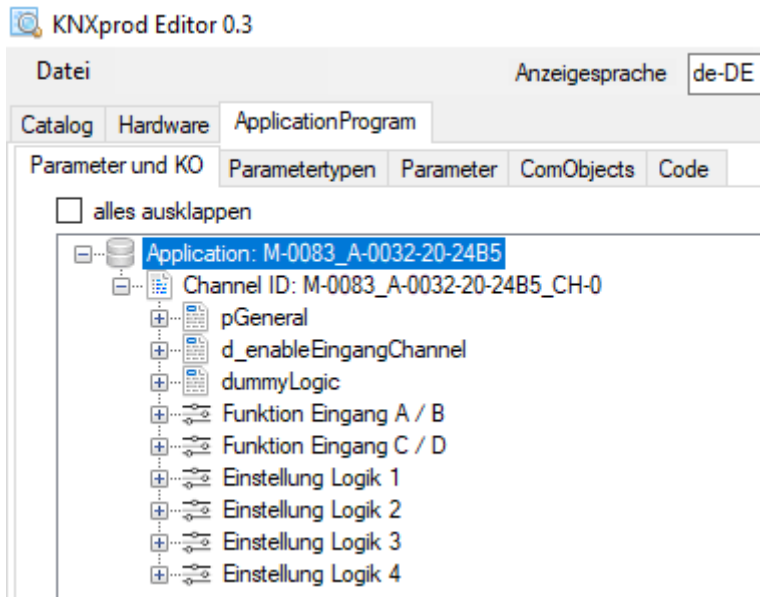


Abbildung 5: ApplicationProgram geöffnet

In diesem Tab sind die verschiedenen Elemente des ApplicationPrograms aufgeführt.

Der zuerst geöffnete Tab „Parameter und KO“ beinhaltet sämtliche Inhalte der „Dynamic“ Section der Applikations XML. Hier ist es je nach Anwendung recht komplex, die Zusammenhänge zu verstehen. Jedes Element in der Baumansicht kann ausgewählt und gegebenenfalls verändert werden.

Die Tabs „Parametertypen“, „Parameter“, „ComObjects“ und „Code“ stellen die Sektionen der Static Sektion im ApplikationProgram dar.

## 4 Parameter Typen

The screenshot shows the 'Parameter Typen' window with the 'en\_ValueLongShort' parameter type selected in the left-hand tree. The right-hand pane displays the configuration for this type, which is a 'Restriction'.

**Item:** Restriction

**Id:** M-0083\_A-0032-20-24B5\_PT-en.5FValueLongShort

**Name:** en\_ValueLongShort

**InternalDescription:**

**Plugin:**

**ValidationErrorRef:**

**TypeRestriction:**

**Base:** Value

**SizeInBit:** 8

**uiHint:** Text

**Enumeration:**

- Aus
- An
- Umschalten
- Wert senden
- Nichts

Abbildung 6: ein Beispiel: Parameter Typ Restriction, also Aufzählung

In Abbildung 6 ist beispielhaft ein möglicher Parameter Typ gezeigt, die Aufzählung, in der knxprod Welt als „TypeRestriction“ benannt.

Dieser Typ ist der aufwändigste, da er aus weiteren Elementen besteht, die in dem Feld „Enumeration“ aufgelistet werden. Wird dort ein Element ausgewählt, werden die Details zu dem Element angezeigt.

The screenshot shows the 'Enumeration Item' configuration window for the 'Aus' element. The left-hand pane shows the 'Enumeration' list with 'Aus' selected. The right-hand pane displays the configuration for this element.

**Base:** Value

**SizeInBit:** 8

**uiHint:** Text

**Enumeration:**

- Aus
- An
- Umschalten
- Wert senden
- Nichts

**Enumeration Item:**

**Text:** Off

**Icon:**

**PictureAlignment:** Left

**Value:** 0

**Id:** M-0083\_A-0032-20-24B5\_PT-en.5FValueLongShort\_EN-0

**DisplayOrder:** 0

**BinaryValue:**

**Translations:**

Sprache	Text
de-DE	Aus
en-US	Off
es-ES	Off

**Enum Item speichern**

Abbildung 7: Detailansicht eines Enumeration Elementes

Im Beispiel aus Abbildung 7 zu sehen ist ein Enumeration Element für den Wert 0. Bedeutet also, wenn man diesen Parameter Typ für eine Auswahl verwenden möchte, ist dieses Element unter der Nummer (Value) 0 zu finden.

Entsprechend zu dem Element gibt es in den verschiedenen Sprachen die Übersetzung, die auch verändert werden kann.

Soll ein neues Enumeration Element angelegt werden, muss in die Enumeration Auflistung per Rechtsklick der Eintrag „hinzufügen“ ausgewählt werden.

Es werden alle Felder der Detailansicht geleert und diese können dann befüllt werden. Sobald auf „Enum Item speichern“ geklickt wurde, wird das Element der Liste angefügt.

Zum Löschen eines Enumeration Elementes wird per Rechtsklick auf das entsprechende Element der Eintrag „löschen“ geklickt.

Um einen neuen Parameter Typ anzulegen wird in die Auflistung der Parameter Typen ein Rechtsklick ausgeführt und der Eintrag „hinzufügen“ ausgewählt. Für das Löschen eines Parameter Typs wird entsprechend gehandelt.

Um den Typ eines Parameter Typs zu verändern bzw einen neuen Parameter Typ anzulegen, wird in der DropDown Liste unter „Item“ eines der folgenden Elemente ausgewählt:

- Restriction: wie vorher beschrieben, eine Auflistung
- Number: ein ganzzahliger, numerische Parameter Typ (uint oder int)
- Float: ein Gleitkomma Parameter Typ
- Text: Ein Text Parameter Typ

Je nach Auswahl des Typs verändert sich die Maske entsprechend und verlangt unterschiedliche Eingaben.

## 5 Parameter

Ein Parameter kann einen Speicherplatz im KNX Gerät haben oder auch nicht. Die Parameter ohne Speicherplatz sind nur für die Steuerung der dynamischen Elemente in der ETS nötig.

Die Parameter mit Speicherplatz werden an die entsprechende Speicherstelle an das KNX Gerät bei dem Schreiben des Applikationsprogramms geschrieben.

The screenshot shows the ETS3 software interface for configuring a parameter. The left pane lists various parameters, with 'OM\_EingangUsage\_timerConfigOff\_0' selected. The main pane shows the configuration for this parameter, including its ID, name, type, and memory address. The 'Memory' tab is active, showing the code segment, offset, and base address.

Parameter	Value								
Id	M-0083_A-0032-20-24B5_P-5511								
Name	OM_EingangUsage_timerConfigOff_0								
ParameterTypeParams									
Text									
SuffixText									
Access	None								
Value	1								
InitialValue									
CustomerAdjustable	<input type="checkbox"/>								
InternalDescription									
LegacyPatchAlways	<input type="checkbox"/>								
Translations	<table border="1"><thead><tr><th>Sprache</th><th>Text</th></tr></thead><tbody><tr><td>de-DE</td><td></td></tr><tr><td>en-US</td><td></td></tr><tr><td>es-ES</td><td></td></tr></tbody></table>	Sprache	Text	de-DE		en-US		es-ES	
Sprache	Text								
de-DE									
en-US									
es-ES									

☒ Parameter mit Memory oder Property

**Memory**

CodeSegment	M-0083_A-0032-20-24B5_AS-4400
Offset	0x A8
BitOffset	7
Memory BaseAddress	17408 (dez) = 0x4400
Parameter Memory Address	17583 (dez) = 0x44AF

speichern

Abbildung 8: Parameter mit Memory Eintrag

Es gibt auch wohl noch Parameter, die anstatt eines Memory Eintrages ein Property Einträge haben, das Programm sollte dafür ausgelegt sein, nur leider habe ich noch keine knxprod gesehen, worin das verwendet wurde.

Ein neuer Parameter kann über einen Rechtsklick in die Auflistung aller Parameter hinzugefügt werden, gleiches gilt für das Löschen eines Parameters.

Dabei ist noch zu beachten: Es gibt zusätzlich zu den einfachen Parametern, die an einer Speicherstelle stehen noch die Unions. Eine Union fasst mehrere Parameter zusammen, die an dieser einen Speicherstelle, die durch die Union definiert wird, stehen können.

Eine solche Union hat leider keinen wirklichen Namen, daher werden sie im Baum als „Union Adresse: 0xFFFF“ angezeigt. Durch einen Klick auf solch eine Union lassen sich die Parameter, die in dieser Union zusammengefasst werden, anzeigen und bearbeiten.

Parameter und KO

Parametertypen

Parameter

ComObjects

Code

OM\_EingangUsage\_timerConfigOff\_0

OM\_inputUsage\_objType\_0

Union Adresse: 0x44B6

Union Adresse: 0x44B8

Union Adresse: 0x44BA

Union Adresse: 0x44BE

Union Adresse: 0x44C6

OM\_EingangUsage\_valueType\_00\_1

OM\_EingangUsage\_valueType\_03\_1

OM\_inputUsage\_counterLimit\_1

OM\_inputUsage\_counterDiffSend\_1

Union Adresse: 0x44D4

OM\_EingangUsage\_timerOnActive\_1

OM\_EingangUsage\_timerOffActive\_1

OM\_EingangUsage\_timerConfigOn\_1

OM\_EingangUsage\_timerConfigOff\_1

OM\_inputUsage\_objType\_1

Union Adresse: 0x44E4

Union Adresse: 0x44E6

Union Adresse: 0x44E8

Union Adresse: 0x44EC

Union Adresse: 0x44F4

OM\_EingangUsage\_valueType\_00\_2

OM\_EingangUsage\_valueType\_03\_2

OM\_inputUsage\_counterLimit\_2

OM\_inputUsage\_counterDiffSend\_2

Union Adresse: 0x4502

OM\_EingangUsage\_timerOnActive\_2

OM\_EingangUsage\_timerOffActive\_2

OM\_EingangUsage\_timerConfigOn\_2

OM\_EingangUsage\_timerConfigOff\_2

OM\_inputUsage\_objType\_2

Union Adresse: 0x4512

Union Adresse: 0x4514

Union Adresse: 0x4516

Union Adresse: 0x451A

Union

SizeInBit (Union) 8

InternalDescription (Union)

☒ Union mit Memory oder Property

UnionMemory

CodeSegment M-0083\_A-0032-20-24B5\_AS-4400

Offset 0x E8

BitOffset 0

Memory BaseAddress 17408 (dez) = 0x4400

Parameter Memory Address 17640 (dez) = 0x44E8

speichern

Union Parameters

OM\_EingangUsage\_values\_00\_2

OM\_EingangUsage\_values\_00\_2

OM\_EingangUsage\_values\_00\_2

OM\_EingangUsage\_values\_00\_2

OM\_EingangUsage\_values\_00\_2

Id M-0083\_A-0032-20-24B5\_UP-50

Name OM\_EingangUsage\_values\_00\_2

ParameterTypeParams

Offset 0x 0

BitOffset 0

Text Value for falling edge

SuffixText

Access None

Value 0

InitialValue

CustomerAdjustable ☐

DefaultUnionParameter ☒

InternalDescription

Translations

Sprache	Text
de-DE	Wert für fallende Flanke
en-US	Value for falling edge
es-ES	Valor del flanco de bajada

Parametertype Name ui\_unsignedChar

SelectableParameters

ParametertypeSizeInBit 8

speichern

**Abbildung 9: Ein Union Parameter mit mehreren Parametern**

In der Abbildung 9 ist ein Beispiel für ein Union Parameter zu sehen. Die Adresse der Union wird angezeigt, sowie alle untergeordneten Union Parameter. Durch einen Klick auf ein Union Parameter lassen sich die Details dieses Elementes betrachten und verändern.

Wichtig hierbei: der speichern Button im Feld des Union Parameters ist unbedingt bei jeder Änderung eines Union Parameters zu betätigen. Der speichern Button im Union Bereich übernimmt diese Funktion nicht!



Der Übersichtlichkeit halber wurde hier in den Abbildungen die Memory Table entfallen lassen.

CodeSegment Adressbereich: 0x4400 - 0x458E

UserEEPROM Adress Table

	Bit 0 2 <sup>0</sup>	Bit 1 2 <sup>1</sup>	Bit 2 2 <sup>2</sup>	Bit 3 2 <sup>3</sup>	Bit 4 2 <sup>4</sup>	Bit 5 2 <sup>5</sup>	Bit 6 2 <sup>6</sup>	Bit 7 2 <sup>7</sup>
0x44CF	Par...							
0x44D0	Par...							
0x44D1	Par...							
0x44D2	Sen...							
0x44D3	Sen...							
0x44D4	Abs...							
0x44D5	Abs...							
0x44D6					Par...	Par...	Par...	Par...
0x44D7	Par...							
0x44D8								
0x44D9								
0x44DA								
0x44DB								
0x44DC								
0x44DD								
0x44DE								
0x44DF								
0x44E0								
0x44E1								
0x44E2								
0x44E3								
0x44E4	Par...							
0x44E5	Par...							
0x44E6	Par...							
0x44E7	Par...							
▶ 0x44E8	We...							

Abbildung 10: Memory Table

Oberhalb der Memory Table wird der aktuell gewählte Adressbereich angezeigt und kann auch verändert werden. Beim Anklicken eines Parameters mit Memory Eintrag oder einer Union wird die korrekte Speicherstelle angesprungen. Dieses Element dient nur zur visuellen Darstellung des Speichers.

## 6 Kommunikationsobjekte

Die Kommunikationsobjekte werden in einem weiteren Tab „ComObjects“ angezeigt.

Auch hier ist wieder eine Liste aller Kommunikationsobjekte auf der linken Seite zu sehen. Ein Klick auf eines der Elemente lässt die Details zu diesem Element erscheinen.

Über der Liste der Kommunikationsobjekte ist der Speicherbereich angegeben, in dem die Kommunikationsobjekte liegen.

Parameter und KOParameter typenParameterComObjectsCode

ComObjects Memory  
Start 0x4400Ende 0x458E

ComObject Number: 0 (Input A Send forced setting)  
ComObject Number: 1 (Input A / B Stop/Blinds Open/Close)  
ComObject Number: 2 (Input A push-button long)  
ComObject Number: 3 (Input A Counter)  
**ComObject Number: 4 (Input A / B Blocking Object)**  
ComObject Number: 5 (Input B Send forced setting)  
ComObject Number: 6 (Input B Blinds/Stop)  
ComObject Number: 7 (Input B push-button long)  
ComObject Number: 8 (Input B Counter)  
ComObject Number: 9 (Input B Blocking Object)  
ComObject Number: 10 (Input C Send forced setting)  
ComObject Number: 11 (Input C / D Stop/Blinds Open/Close)  
ComObject Number: 12 (Input C push-button long)  
ComObject Number: 13 (Input C Counter)  
ComObject Number: 14 (Input C / D Blocking Object)  
ComObject Number: 15 (Input D Send forced setting)  
ComObject Number: 16 (Input D Blinds/Stop)  
ComObject Number: 17 (Input D push-button long)  
ComObject Number: 18 (Input D Counter)  
ComObject Number: 19 (Input D Blocking Object)  
ComObject Number: 20 (Logic input 1 A Logic input 1 A)  
ComObject Number: 21 (Logic input 1 B Logic input 1 B)  
ComObject Number: 22 (Logic output 1 Logic output 1)  
ComObject Number: 23 (Logic input 2 A Logic input 2 A)  
ComObject Number: 24 (Logic input 2 B Logic input 2 B)  
ComObject Number: 25 (Logic output 2 Logic output 2)  
ComObject Number: 26 (Logic input 3 A Logic input 3 A)  
ComObject Number: 27 (Logic input 3 B Logic input 3 B)  
ComObject Number: 28 (Logic output 3 Logic output 3)  
ComObject Number: 29 (Logic input 4 A Logic input 4 A)  
ComObject Number: 30 (Logic input 4 B Logic input 4 B)  
ComObject Number: 31 (Logic output 4 Logic output 4)

IdM-0083\_A-0032-20-24B5\_O-4

NameInput A / B

TextInput A / B

Number4

FunctionTextBlocking Object

PriorityLow

ObjectSize1 Bit

ReadFlag☐

WriteFlag☒

CommunicationFlag☒

TransmitFlag☐

UpdateFlag☒

ReadOnInitFlag☐

Datapoint TypeDPT 1.003 Enable  
DPT 1.004 Ramp  
DPT 1.005 Alarm

InternalDescription

SecurityRequiredNone

MayRead☐

ReadFlagLocked☐

WriteFlagLocked☐

TransmitFlagLocked☐

UpdateFlagLocked☐

ReadOnInitFlagLocked☐

Translations Text

Sprache	Text
de-DE	Eingang A / B
en-US	Input A / B
es-ES	Entrada A / B

Translations FunctionText

Sprache	Text
de-DE	Sperrobjekt
en-US	Blocking Object
es-ES	Objeto de bloqueo

speichern

Abbildung 11: Die Kommunikationsobjekte

### 6.1.1.1

## 7 Code

Im Tab „Code“ werden alle Codeselemente aufgeführt, die in der Applikation verwendet werden. Bisher noch nicht weiter genutzt.

## 8 Eine Auswahlmöglichkeit erstellen

In der ETS sind häufig Auswahlmöglichkeiten gegeben: Zum Beispiel, wenn ein Aktor eine Schalt- sowie eine Rollladen-Funktion haben soll, werden je nach Auswahl der Funktion die entsprechenden Kommunikationsobjekte angezeigt oder ausgeblendet.

Dazu sind im „Parameter und KO“ Tab sogenannte „choose“ Elemente verantwortlich. Wenn man sich durch eine bestehende knxprod durcharbeitet wird man in der Detailansicht die Tabs „ChannelChoose“ und auch „ComObjectParameterChoose“ finden. Diese Tabs werden nicht vorrangig angezeigt, sondern immer der dazugehörige Parameter, aber man kann sie anwählen und auch verändern.

Da es 2 verschiedene „choose“ Elemente gibt, hängt mit der „Höhe“ des Objektes zusammen. Weiter unten im Baum sind die ComObjectParameterChoose Elemente zu finden, weiter oben die ChannelChoose Elemente. Wenn durch einen Rechtsklick auf ein Element ein choose hinzugefügt werden soll, wird automatisch das korrekte Element durch den KNXprodEditor ausgewählt.

Sehr wichtig bei der Erstellung eines choose Elementes ist die Variable „DisplayOrder“. Diese ist im Reiter „ParameterRef“ zu finden. Der Haken vor „DisplayOrder“ muss gesetzt werden (sonst wird der Wert nicht angenommen) und ein logischer numerischer Wert muss eingetragen werden. Dieser Wert bestimmt, an welcher Position die Auswahlmöglichkeit in der ETS angezeigt werden wird.

Daher ist es immer ratsam, etwas Platz zwischen den Werten zu lassen, um eventuelle Änderungen noch einfügen zu können. Diese Werte haben keinen Anspruch auf Durchgängigkeit, es dürfen also Lücken gelassen werden.

Wichtig ist nur: In der ETS wird die kleinste DisplayOrder oben angezeigt und nach unten immer größere Werte.

## 9 HowTo: Eine komplett neue knxprod erstellen

Um eine komplett neue knxprod Datei zu erstellen, wird eine „Spenderdatei“ benötigt. Also eine knxprod, die komplett entleert ist.

Solch eine Datei wird es für eine BCU1, BCU2 und BIM112 geben, worin der Speicherbereich für die sblib angepasst wurde.

In dieser Datei sollte zuerst der gewünschte Hersteller unter dem Reiter „Catalog“ in der Baumansicht unter „Manufacturer“ eingestellt werden.

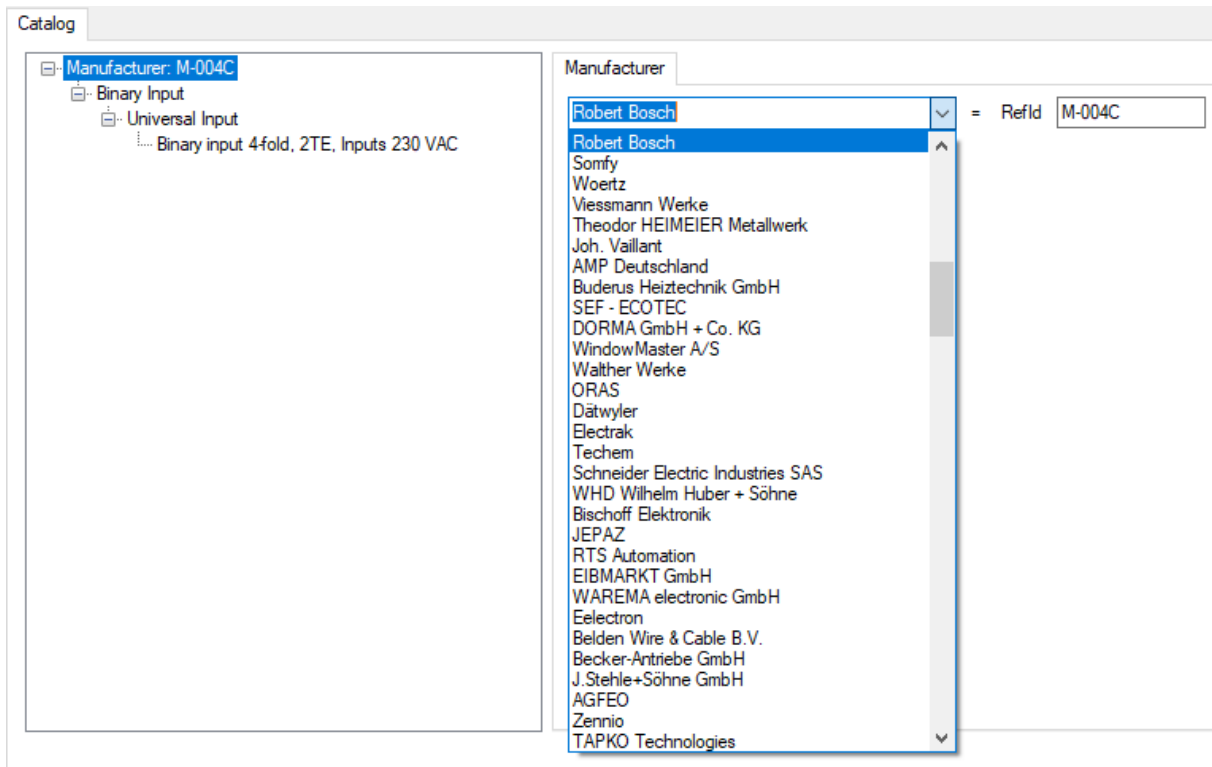


Abbildung 12: Einstellen des Manufacturers

Anschließend bietet es sich an, den Baum über die „Catalog Sections“ Elemente hinunter zu gehen und in jeder Detailansicht die gewünschten Beschreibungen einzugeben. Umlaute sollten nur in den „Translations“ verwendet werden, nicht in Feldern wie „Name“ oder ähnlichem.

**CatalogSection**

Id: M-004C\_CS-B.2001

Name: Gateway

Number: B 01

VisibleDescription: Binary input

DefaultLanguage: en-US

NonRegRelevantDataVersion: 0

InternalDescription:

Sprache	TextCol
en-US	
de-DE	
es-ES	

speichern

Abbildung 13: CatalogSection anpassen

Auch das „CatalogItem“ benötigt eine passende Beschreibung.

**CatalogItem**

Id: M-004C\_H-31-1\_HP-0032-20-24B5\_CI-BE.2D04230.2E01-1

Name: OneWireGateway, 2TE, xx Channels

Number: 1

VisibleDescription: Input voltage 230 VAC

ProductRefId: M-004C\_H-31-1\_P-BE.2D04230.2E01

Hardware2ProgramRefId: M-004C\_H-31-1\_HP-0032-20-24B5

DefaultLanguage: en-US

NonRegRelevantDataVersion: 0

InternalDescription:

Sprache	Text
en-US	OneWireGateway, 2TE, xx Channels
de-DE	OneWireGateway, 2TE, xx Kanäle

speichern

Abbildung 14: CatalogItem anpassen

Jetzt wird in den Hauptreiter „Hardware“ gewechselt.

Hier lässt sich der „Manufacturer“ nicht mehr verändern, jedoch aber alle weiteren Elemente. Am besten hier auch wieder von oben nach unten vorgehen und jedes Element entsprechend beschriften.

Bei einem komplett neuen Gerät ist es ratsam, die Id im Tab „Product“ zu ändern. Über diese ID erkennt die ETS ein Produkt, aber diese ID kann nicht in der ETS eingesehen werden.

The screenshot shows the 'Hardware' tab in the ETS software. On the left, a tree view shows the hierarchy: Manufacturer: M-004C, OneWire Gateway, OneWireGateway, 2TE, xx Kanäle, Hardware2Program: M-004C\_H-31-1\_HP-0032-20-6CD5, ApplikationProgram: M-004C\_A-0032-20-6CD5. The 'Product' tab is active, showing the following fields:

Id	M-004C_H-31-1_P-	BE.2D04230.2E01				
Text	OneWireGateway, 2TE, xx Channels					
OrderNumber	BE-04230.01					
IsRailMounted	<input checked="" type="checkbox"/>					
WidthInMillimeter	<input checked="" type="checkbox"/>	36,30				
VisibleDescription						
DefaultLanguage	en-US					
NonRegRelevantDataVersion	0					
Hash	172 166 149					
InternalDescription						
Translations	<table border="1"><thead><tr><th>Sprache</th><th>Text</th></tr></thead><tbody><tr><td>de-DE</td><td>OneWireGateway, 2TE, xx Kanäle</td></tr></tbody></table>		Sprache	Text	de-DE	OneWireGateway, 2TE, xx Kanäle
Sprache	Text					
de-DE	OneWireGateway, 2TE, xx Kanäle					

At the bottom, there is a 'speichern' button.

Abbildung 15: Hardware Product anpassen

Das „Hardware2Program“ benötigt keine Beschreibung, somit normalerweise auch keine Veränderung.

Durch die Auswahl des „Application Programs“ lässt es sich entsprechend laden. Das erste Bauelement, die „Application“ hat noch einen wichtigen Beschreibungsparameter, den „Name“. Dieser wird später als Beschreibung des Applikationsprogramms in der ETS angezeigt.

The screenshot shows the 'ApplicationProgram' tab in the ETS software. On the left, a tree view shows the hierarchy: Application: M-004C\_A-0032-20-24B5. The 'Application' tab is active, showing the following fields:

Id	M-004C_A-0032-20-24B5
ApplicationNumber	50
ApplicationVersion	32
Program Type	ApplicationProgram
Mask Version	MV-0701
VisibleDescription	
Name	OneWire Gateway
LoadProcedureStyle	ProductProcedure
PeiType	1
Help Topic	<input type="checkbox"/> 0
HelpFile	

Abbildung 16: ApplicationProgram anpassen

Ab hier haben wir nun alle formalen Dinge erledigt und können die Erstellung des eigentlichen Applikationsprogramms beginnen.

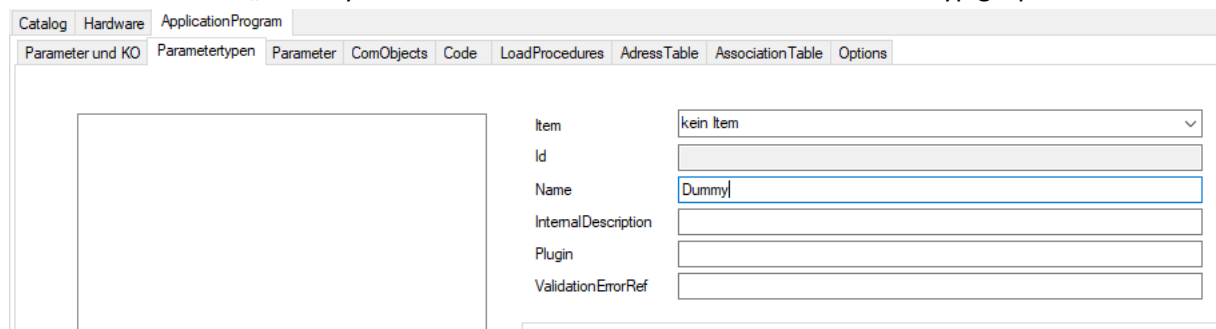
## 9.1 Parametertypen anlegen

Als aller erstes werden Parametertypen benötigt. In vielen knxprod Dateien sind „Dummy“ Parametertypen angelegt, um viele verschiedene Dinge damit zu belegen, die später beispielsweise nur in der ETS anzuzeigen sind (Texte, Trennstriche oder ähnliches). Daher soll auch dieses hier der erste Parametertyp sein.

Dazu wird in den Tab „Parametertypen“ gewechselt und per Rechtsklick in das leere Feld auf der linken Seite ein Element hinzugefügt.

Die Felder der Detailansicht stehen nun für die Eintragung eines neuen Parametertyps bereit. Dieses Dummy Element benötigt kein Item, daher wird in dem DropDown Menü von „Item“ „kein Item“ ausgewählt.

Die Felder im unteren Bereich verschwinden und es muss ein Name eingetragen werden. Dieser sollte in diesem Fall „Dummy“ lauten. Anschließend wird der neue Parametertyp gespeichert.



Catalog Hardware ApplicationProgram		
Parameter und KO Parametertypen Parameter ComObjects Code LoadProcedures AdressTable AssociationTable Options		
<div></div>	Item	kein Item
	Id	
	Name	Dummy
	InternalDescription	
	Plugin	
	ValidationErrorRef	

Abbildung 17: ParameterType "Dummy" anlegen

Für dieses HowTo entscheiden wir uns noch für einen zweiten Parametertyp. Es soll der aufwändigste Parametertyp sein, nämlich eine Auflistung. Dazu wird wieder per Rechtsklick in das linke Feld ein neuer Parametertyp hinzugefügt. Dieses Mal wird als „Item“ die „Restriction“ ausgewählt. Es muss ein Name eingegeben werden, wir entscheiden uns für „AnAus“.

Im unteren Bereich befindet sich nun ein Feld mit der Überschrift „TypeRestriction“. In diesem Feld wird die „Base“ auf „Value“ stehen gelassen. Im Feld „SizeInBit“ muss die gewünschte Speichergröße angegeben werden. Dieser Wert wird aus den Werten bestimmt, die die Aufzählungselemente später repräsentieren sollen. In unserem Beispiel möchten wir nur 2 Elemente erstellen, daher reicht uns eine Größe von 1 Bit aus (0=aus, 1= ein).

Nach diesen grundsätzlichen Einstellungen für den Parametertypen muss dieser nun gespeichert werden.

ApplicationProgram

Parameter und KO | Parametertypen | Parameter | ComObjects | Code | LoadProcedures | AdressTable | AssociationTable | Options

.... Dummy

Item: Restriction

Id:

Name: AnAus

InternalDescription:

Plugin:

ValidationErrorRef:

TypeRestriction

Base: Value

SizeInBit: 1

ulHint: Text

Enumeration:

Abbildung 18: ParameterType "AnAus" erstellen

Mit einem Rechtsklick in das leere Feld „Enumeration“ kann ein neues Auflistungselement hinzugefügt werden.

In der sich nun öffnenden Detailansicht muss im Feld „Text“ und in den „Translations“ etwas entsprechendes eingetragen werden. In diesem Fall an beiden Stellen entsprechend „Aus“ bzw. „Off“ usw. Ein wichtiges Feld ist noch das „Value“ Feld, mit dem der zu repräsentierende Wert eingegeben wird. Für „Nein“ soll hier 0 stehen, für „Ein“ die 1. Wird ein Wert doppelt vergeben, wird eine Fehlermeldung ausgegeben.

Abschließend muss jedes Element über den Button „Enum Item speichern“ gespeichert werden.

Item: Restriction

Id: M-004C\_A-0032-20-24B5\_PT-AnAus

Name: AnAus

InternalDescription:

Plugin:

ValidationErrorRef:

TypeRestriction

Base: Value

SizeInBit: 1

ulHint: Text

Enumeration: .... Aus

Enumeration Item

Text: An

Icon:

PictureAlignment: Left

Value: 1

Id:

DisplayOrder: 1

BinaryValue:

Translations

Sprache	Text
de-DE	An
en-US	On
es-ES	

Enum Item speichern

Abbildung 19: ParameterType Restriction Item "An" anlegen



## 9.2 Parameter anlegen

Für den Aufbau des dynamischen Baumes benötigen wir Parameter. Parameter sind im einfachsten Fall nur für die ETS wichtig, in anderen Fällen werden sie auch in das Gerät übertragen und benötigen einen Speicherplatz. Zusätzlich gibt es auch Unions, mit denen sich mehrere Parameter auf einen Speicherplatz legen lassen.

Für die Anlage eines neuen Parameters wird per Rechtsklick in das leere, linke Feld ein Parameter hinzugefügt. Hier ist die Eingabe im Feld „Name“ notwendig. Als erstes wollen wir ein Element zur Navigation erzeugen. Als Beispiel nehmen wir hier „Allgemeine Einstellungen“. Dieses wird auch unter den „Translations“ entsprechend eingetragen.

Dieser Parameter soll später lesbar in der ETS erscheinen, daher muss das Feld „Access“ auf „Read“ verändert werden.

Hier wird der Parametertyp „Dummy“ benötigt und per DropDown“ Menü aus dem Feld „Parametertype Name“ ausgewählt.

Dieser nun erstellte Parameter wird gespeichert und über den bekannten Weg ein neuer Parameter angelegt.

Sprache	Text
de-DE	Allgemeine Einstellungen
en-US	General Settings
es-ES	

Abbildung 20: Parameter "Allgemeine Einstellungen" erstellen

Dieser Parameter soll später die Möglichkeit eröffnen, sich für „Ein“ oder „Aus“ zu entscheiden. Für dieses Beispiel möchten wir, dass die Möglichkeit im KNX Gerät besteht eine Leuchte leuchten zu lassen oder nicht. Daher wird dieser Parameter mit „Leuchte“ im Feld „Name“ beschrieben.

Entsprechendes muss auch in das „Translations“ Feld eingetragen werden.

Da dieser Parameter in der ETS nicht nur angezeigt, sondern auch verändert werden soll, muss das Feld „Access“ auf ReadWrite“ verändert werden.

Nun muss der Parametertyp eingestellt werden. Dazu wird im Feld „Parametertype Name“ der vorher erstellte Parametertyp „AnAus“ ausgewählt werden.

Da dieser Parametertyp eine Aufzählung ist, muss hierfür zwingend ein Default Wert in das „Value“ Feld eingetragen werden. Dieser Wert repräsentiert die Standardeinstellung der Auswahl aus der Aufzählung in der ETS.

Da dieser Parameter im Gerät eine Veränderung auslösen soll, muss dieser auch in das Gerät übertragen werden. Dazu wird der Haken in der Checkbox „Parameter mit Memory oder Property“

gesetzt.

Es öffnet sich ein weiteres Feld, indem der Reiter „Memory“ ausgewählt bleiben muss.

Im CodeSegment wird das einzige Element mit der Beschriftung „M-004C\_A-0032-20-24B5\_AS-4400“ ausgewählt. Hier werden die Parameter in diesem Fall gespeichert.

Der Offset wird großzügig gewählt, für dieses Beispiel nehmen wir 0x100, damit ist dieser Parameter 256Byte weiter angelegt, als der Anfang des Speicherbereichs.

Der BitOffset soll hier 0 sein.

Hierbei ist zu beachten, dass ein Bit Offset von 0 den Parameter in den obersten Bits der Speicheradresse liegen lässt.

In diesem Beispiel von das Byte an Adresse mit dem Offset von 0x100 bei der Auswahl des Auflistungselementes mit der Value 1 (Enum Item „Ein“) folgendermaßen aussehen: 1000 0000

Mit dem MSB links und LSB rechts wird dann ein Hexwert von 0x80 ausgelesen werden können.

Auch dieser Parameter wird dann gespeichert und ist bereits für seinen Einsatz.

The screenshot shows the 'Parameter' configuration window. The 'Parameter' tab is selected, and the 'Union' sub-tab is active. The 'Name' field is 'Leuchte'. The 'Access' is 'ReadWrite'. The 'Value' is '0'. The 'InitialValue' is empty. The 'CustomerAdjustable' checkbox is unchecked. The 'InternalDescription' is empty. The 'LegacyPatchAlways' checkbox is unchecked. The 'Translations' table has three rows: 'de-DE' with 'Leuchte 1', 'en-US' with 'Lamp 1', and 'es-ES' with an empty text field. The 'Parameter mit Memory oder Property' checkbox is checked. Under this, the 'Memory' sub-tab is active, showing 'CodeSegment' as 'M-004C\_A-0032-20-24B5\_AS-4400', 'Offset' as '0x 100', and 'BitOffset' as '0'. The 'Memory BaseAddress' and 'Parameter Memory Address' fields are empty. A 'speichern' button is at the bottom left.

Abbildung 21: Parameter "Lampe" erstellen

## 9.3 Erstellen von Kommunikationsobjekten

Zum Erstellen von Kommunikationsobjekten wird in den Tab "ComObjects" gewechselt.

Hier wird über einen Rechtsklick in das leere Feld auf der linken Seite ein Element hinzugefügt. Die Felder „Name“, „Text“ und „FunctionText“ sollten entsprechend gefüllt werden. Wichtig ist, das Feld „number“, mit dem die Nummer des Kommunikationsobjektes eingestellt wird. Hier darf jede Nummer immer nur einmal vorkommen, ansonsten wird ein Fehler angezeigt.

Als Beispiel tragen wir hier folgende Daten ein:

Name: Ausgang 1

Text: Ausgang 1

Number: 0  
 FunctionText: Schalten  
 ObjectSize: 1Bit  
 WriteFlag: true  
 CommunicationFlag: true  
 Translations Text: de-DE: Ausgang 1  
 Translations Function Text: de-DE: Schalten

ComObjects Memory

Start 0x 4400 Ende 0x 47E8

Id

Name: Ausgang 1

Text: Ausgang 1

Number: 0

FunctionText: Schalten

Priority: Low

ObjectSize: 1 Bit

ReadFlag: ☐

WriteFlag: ☒

CommunicationFlag: ☒

TransmitFlag: ☐

UpdateFlag: ☐

ReadOnInitFlag: ☐

Datapoint Type: DPT 1.001 Switch, DPT 1.002 Bool, DPT 1.003 Enable

InternalDescription:

SecurityRequired: None

MayRead: ☐

ReadFlagLocked: ☐

WriteFlagLocked: ☐

TransmitFlagLocked: ☐

UpdateFlagLocked: ☐

ReadOnInitFlagLocked: ☐

Translations Text

Sprache	Text
de-DE	Ausgang 1
en-US	Output 1
es-ES	

Translations FunctionText

Sprache	Text
de-DE	Schalten
en-US	switch
es-ES	

Abbildung 22: Kommunikationsobjekt "Ausgang 1 Schalten" erstellen

## 9.4 Anlegen des dynamischen Baumes

In einfachsten Fall möchte man keine dynamischen Entscheidungen haben sondern nur eine statische Auswahl von Parametern und ComObjekten.

Um das Ganze dann etwas zu strukturieren wird noch eine Ebene für übergeordnete Strukturelemente angelegt. Man kennt sie aus der ETS als die Blöcke auf der linken Seite im Parameter Tab.

Zunächst wird unter der „Application“ ein „ChannelIndependentBlock“ hinzugefügt. Dieser Bedarf keinerlei Einstellungen. An diesem „ChannelIndependentBlock“ fügen wir einen „ParameterBlock“ hinzu. Dieser „ParameterBlock“ bedarf der Auswahl eines Parameters. Wir wählen dazu den vorher erstellten Parameter „Allgemeine Einstellungen“ und speichern.

Parameter und KO | Parameter | ComObjects | Code | LoadProcedures | AdressTable | AssociationTable | Options

☐ alles ausklappen

Application: M-004C\_A-0032-20-24B5

Channel Independent Block

Parameter

Parameter:

Id:

Name:

ParameterType:

ParameterTypeParams:

Text:

SuffixText:

Access:

Value:

InitialValue:

CustomerAdjustable:

InternalDescription:

LegacyPatchAlways:

Translations

Sprache	Text
de-DE	
en-US	
es-ES	

speichern

Parameter Eigenschaft sind nur im Parameter Tab zu verändern!

Parametertyp Name:

SelectableParameters:

ParametertypSizeInBit:

Abbildung 23: ParameterBlock "Allgemeine Einstellungen" erstellen

Um den ParameterBlock später in der ETS sichtbar zu haben müssen wir in den Tab „ParameterRef“ wechseln. Dort ist der Haken bei „DisplayOrder“ zu setzen und eine Zahl einzutragen. Diese Zahl bestimmt die Position des angezeigten Parameters in der ETS. Um noch Platz für spätere Änderungen vorzuhalten, tragen wir hier „100“ ein und speichern.

Catalog | Hardware | ApplicationProgram

Parameter und KO | Parameter | ComObjects | Code | LoadProcedures | AdressTable | AssociationTable | Options

☐ alles ausklappen

Application: M-004C\_A-0032-20-24B5

Channel Independent Block

Allgemeine Einstellungen

ComObjectParameterBlock | ParameterRef | Parameter

Id:

RefId:

Name:

Text:

SuffixText:

Tag:

DisplayOrder: ☒

Access:

Value:

InitialValue:

CustomerAdjustable: ☐

TextParameterRefId:

InternalDescription:

forbidGrantingUseByCustomer: ☐

Semantics:

Translations Text

Sprache	Text
de-DE	
en-US	
es-ES	

Abbildung 24: ParameterRef einstellen

An diesem ParameterBlock wird nun ein „Parameter“ hinzugefügt. Als Parameter wird hier der vorher erstellte Parameter „Leuchte“ gewählt und gespeichert. Auch hier muss für die Sichtbarkeit in den „ParameterRef“ Tab gewechselt werden und der Haken bei „DisplayOrder“ gesetzt werden. Da dieser Parameter unterhalb der „Allgemeinen Einstellungen“ angezeigt werden soll, tragen wir hier die Nummer „200“ ein und speichern.

Parameter und KO

ParameterRef

Id: M-004C\_A-0032-20-24B5\_P-2\_R-2

RefId: M-004C\_A-0032-20-24B5\_P-2

Name:

Text:

Suffix Text:

Tag: 2

DisplayOrder: ☒ 200

Access: ☐ None

Value:

Initial Value:

CustomerAdjustable: ☐

TextParameterRefId:

InternalDescription:

forbidGrantingUseByCustomer: ☐

Semantics:

Sprache	Text
de-DE	
en-US	
es-ES	

Abbildung 25: Parameter ParameterRef einstellen

An den ParameterBlock hängen wir noch ein ComObject an. Hier wählen wird das erstellte ComObject „Ausgang 1 Schalten“ aus und speichern.

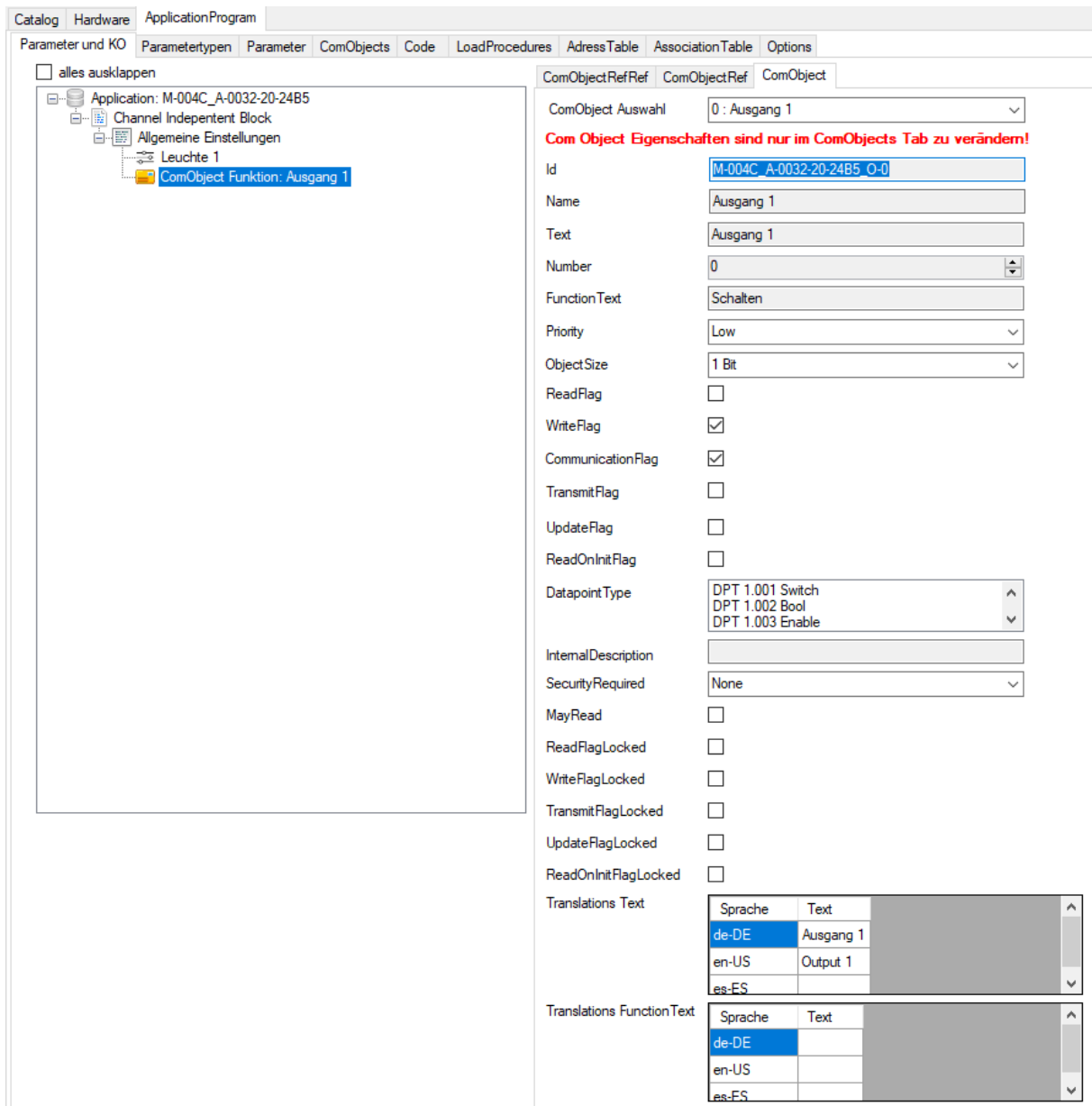


Abbildung 26: ComObject an ParameterBlock anfügen

## 9.5 Fertig stellen und in die ETS laden

Anschließend wird sichergestellt, dass der Pfad zur ConverterEngine.dll korrekt eingestellt ist und die knxprod Datei über Date->Datei speichern an einem beliebigen Ort gespeichert.

Die knxprod Datei kann nun in die ETS geladen werden.

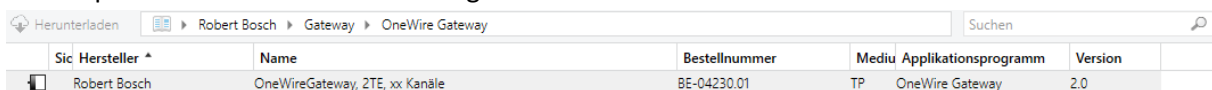


Abbildung 27: Katalogeintrag der knxprod in der ETS

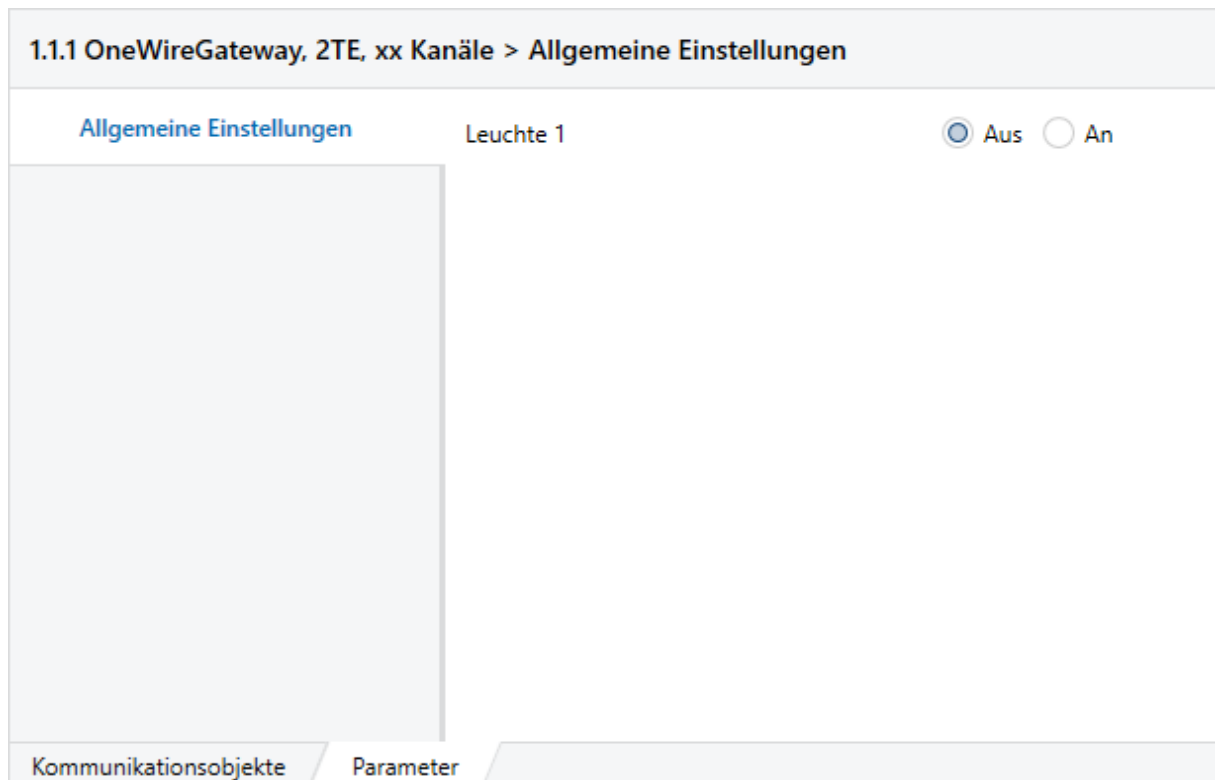


Abbildung 28: Parameteransicht in der ETS

Nummer	Name	Objektfunktion	Beschreibung	Gruppenadresse	Länge	K	L	S	Ü	A	Datentyp	Priorität
2 0	Ausgang 1	Schalten			1 bit	K	-	S	-	-		Niedrig

Abbildung 29: Kommunikationsobjekt Ansicht in der ETS

## 9.6 Das ARM Programm zur selbst erstellten knxprod

Das ARM Programm mit sblib wird folgendermaßen erstellt:

Im KNXprodEditor ist die Möglichkeit vorgesehen, eine vorgefertigte Header Datei für die Erstellung eines KNX Gerätes mit der sblib zu extrahieren. Dazu wird der Button „sblib header erstellen“ im „Parameter und KO“ Tab gewählt. Die header Datei sollte in MCUXpresso kopiert werden und in die app\_main eingebunden werden.

```
* Geräteinformationen:
* Device: OneWireGateway, 2TE, xx Channels
* ApplicationProgram: M-004C_A-0032-20-24B5
*
*/

#define MANUFACTURER 76 //!< Manufacturer ID
#define DEVICETYPE 50 //!< Device Type
#define APPVERSION 32 //!< Application Version

#define EE_PARAMETER_4500_LEUCHTE_1 0x4500 //!< Addr: 0x4500, Size: 0x0001, Leuchte 1

#define COMOBJ_0_AUSGANG_1_SCHALTEN 0 //!< Com-Objekt Nummer: 0, Beschreibung: Ausgang 1, Funktion: Schalten
```

Abbildung 30: maschinell erstellte Header Datei aus der knxprod

Anschließend ist nur noch der Start der bcu notwendig und die Applikation kann aus der ETS in das Gerät geladen werden.

```
/*
 * Initialize the application.
 */
void setup()
{
    bcu.begin(MANUFACTURER, DEVICETYPE, APPVERSION);
}
```

Abbildung 31: Initialisierung der BCU