How to edit and administrate the interpolation point tables of the FB 488 "MC_MovePath" via HMI Interface?

Technology CPU

FAQ • May 2012



Service & Support

Answers for industry.



This entry is taken from the Service&Support Portal of Siemens AG, Sector Industry, Industry Automation and Drive Technologies. The Terms of Use stated there are applicable (<u>www.siemens.com/nutzungsbedingungen</u>).

The following link takes you directly to the download page of this document. http://support.automation.siemens.com/WW/view/en/59421716

Question

How can the interpolation point tables of the technology function FB 488 "MC_MovePath" be edited and administrated in the technology CPU via HMI Interface?

Answer

The instructions and notes listed in this document provide a detailed answer to this question.

Table of Contents

1	Introduc	ction	4
	1.1 1.2	Automation Task	
2	Principl	e of the Program	6
	2.1 2.2 2.2.1 2.2.2 2.3 2.3.1 2.3.2 2.3.3 2.4 2.4.1	Basic information Editing an interpolation point table Function block FB 551 "HMI_List_MP" HMI "MC_MovePath – PointTable Editor" Administration of the interpolation point tables Function block FB 553 "HMI_DB_Explorer_MP" Function block FB 554 "Delete_DB_MP" HMI "MC_MovePath – DB Explorer" Data block of an interpolation point table Data block structure	6 8 10 10 10 11 12
3	Installat	ion	. 15
	3.1 3.2 3.2.1 3.2.2 3.3 3.3.1 3.3.2 3.3.3 3.3.4 3.3.5	Basic proceeding Working within the SIMATIC Manager Unzipping process and content of the STEP 7 source project Transfer of the function blocks to the target project Working within WinCC flexible Preparation of the project texts Copying HMI pages into the target project Adjusting the connections in the HMI project Adjusting the connection of the variables in the HMI project Transfer of the HMI pages to the operatering structure	. 15 . 15 . 16 . 17 . 17 . 20 . 20 . 21
4	Sample	Program	. 24
	4.1 4.2 4.3	Basis Restrictions within the interpolation point tables Selection of the interpolation point tables	. 24

1 Introduction

1.1 Automation Task

The technology function FB 488 "MC_MovePath" can be used to easily realize a palletizing process in the technology CPU. Motion control via the technology function FB 488 "MC_MovePath" is determined by means of interpolation point tables which are stored in data blocks.

If you wish to keep the motion control flexible, it is necessary to edit the interpolation point tables during runtime of the program. Preferably, an HMI Interface should be used, which allows the user to view and edit the interpolation point tables.

Figure 1-1 Simple palletizing process with FB 488 "MC_MovePath"



1.2 Automation solution

This FAQ provides you with an HMI Interface for administrating and editing the interpolation point table of the FB 488 "MC_MovePath", which can be quickly and easily integrated into your own projects.

Figure 1-2 Automation solution



For this process, the HMI Interface provides the following functions:

- Creating new data blocks for integration an interpolation of a point table in the technology CPU.
- Comfortably editing the interpolation point tables that already exist in the technology CPU, including the possibility to insert, delete and copy lines within the interpolation point table.

- DB-Explorer to display the data blocks existing in the technology CPU, which contain an interpolation point table. The decision, whether an existing data block is an interpolation point table for the technology function FB 488 "MC_MovePath" is made according to the size of the data block (5280 Bytes).
- Deleting interpolation point tables that already exist in the technology CPU; for security reasons, this function is also limited to a data block size of 5280 bytes.
- **Note** The functionality of the HMI Interface presented in this document is limited to administration and editing the interpolation point table for the technology function FB 488 "MC_MovePath", which is stored in a data block.

The functionality for preparing the interpolation table via FB 489 "MC_PathSelect", which is required for the process, as well as the functionality for traversing the interpolation points via"MC_MovePath" are not part of the interface, but form part of the sample program presented in chapter 4.

2 Principle of the Program

2.1 Basic information

The HMI Interface presented in this document is closely linked to three data blocks, which provide the actual function for administration and editing the interpolation point tables. Additionally, the operator screens of the HMI Interface are also directly connected to the function blocks.



Figure 2-1 HMI Interface link to the function blocks

Note Array accesses and administration processes within the function blocks are quite complex. Therefore, the function blocks stated here are realized in the SCL creation language.

2.2 Editing an interpolation point table

Interpolation point tables are edited via the function block FB 551 "HMI_List_MP" or via "MC_MovePath – PointTable Editor" of the HMI Interface.

2.2.1 Function block FB 551 "HMI_List_MP"

The function block FB 551 "HMI_List_MP" provides the following functions for editing and manipulation of an interpolation point table stored in a data block:

• Creating a new interpolation point table

By using the system function block SFC 22 "CREAT_DB", a new interpolation point table can be created as data block in the technology CPU. Moreover, the interpolation point table will be opened for further editing right after creating the data block.

- **Opening an already existing interpolation point table for editing** An interpolation point table that already exists in the technology CPU is opened for further editing or manipulation. Within the open interpolation point table, the following administrative functions are available:
 - Inserting blank lines within the interpolation point table
 Within the interpolation point table, a definable number of lines can be inserted. For this purpose, the subsequent interpolation points are shifted downwards within the table.
 - **Deleting lines from the interpolation point table** Within the interpolation point table, a definable number of lines can be deleted. For this purpose, the subsequent interpolation points are shifted upwards accordingly within the table.
 - Copying lines within the interpolation point table
 Within the interpolation point table, a definable number of interpolation points can be copied. Interpolation points, which possibly exist in the target range of the copying process, will be overwritten by this function.
- Saving the performed changes in the interpolation point table Two different functions are available for saving an open interpolation point table:
 - Save

If this function is used, the data of the interpolation point table are written back into the same data block in the technology CPU.

- Save as...

If a new data block number is stated before using the function, the data are written back into the data block defined by the modified number. If this data block should not yet exist in the technology CPU, a new data block with the defined data block number will be created in the technology CPU and the data are written into this newly created data block.

• Reject performed changes

If the data changes shall not be written back into the data block of the interpolation point table, the performed changes can be rejected by using this function.

To edit or manipulate the data of the interpolation point table, there are always two data ranges available in the FB 551 "HMI_List_MP".

• DB_Data:

When opening the interpolation point table (Open), all data records of the data block are stored for editing in this data range. After editing, the manipulated data records – as far as they shall not be rejected (Abort) – are copied back from this data range into the data block of the interpolation point table (Save).

ContourList:

This data range contains a section of the block "DB_Data" and represents the interpolation points, which are currently displayed in the HMI Interface for editing. By performing a page jump (Next/Prev. Page) the data range can be filled with the respective new data from the block"DB_Data".



Figure 2-2 Data flow during manipulation of an interpolation point table

2.2.2 HMI "MC_MovePath – PointTable Editor"

All functions of the FB 551 "HMI_List_MP" can be operated via the page "MC_MovePath – PointTable Editor" of the HMI Interface:

🖬 SIMATIC W	inCC flexible Runtime					
MC_Mov	/ePath - PointTab	le Editor				Technology CPU
PointTable D	Oata Block Editor					
			Create	e Open s	Save Abort D	B-Number 1001
Line	Inter	polationPoint (X/Y/Z)		CornerDistance	Override	M-Fkt.
1	10.000	20.000	0.000	0.00	0 100.000	0
2	10.000	10.000	0.000	-1.00	0 100.000	1
3	10.000	40.000	0.000	10.00	0 100.000	17
4	60.000	40.000	0.000	10.00	0 100.000	1
5	60.000	10.000	0.000	-1.00	0 100.000	0
6	60.000	40.000	0.000	10.00	0 100.000	0
7	10.000	40.000	0.000	10.00	0 100.000	0
8	10.000	10.000	0.000	-1.00	0 100.000	1
9	10.000	40.000	0.000	10.00	0 100.000	17
10	70.000	40.000	0.000	10.00	0 100.000	1
Done	Busy Activ	e Error		Page	1 of 24	< >
Administrati	ion					
>>>	Insert a number of	lines	StartLine		0	_
>>>	Delete a number of	lines	Number of lines		0	_
>>>	Copy a number of line	es to	Copy to line		0	
Main 9	Screen Axis	Control	FB : PathObject	FB : GroupFunctions	PointTable Editor	DB Explorer

Figure 2-3 HMI "MC_MovePath – PointTable Editor"

Selection of the desired function in the "PointTable Editor" is effected via the buttons above the display area of the interpolation point table.



Figure 2-4 Function selection via the "PointTable Editor"

If an interpolation point table is opened, the individual lines of the table can be comfortably edited. During this process, the HMI Interface always displays 10 lines of the table. The buttons in the lower right corner are used for switching between the "individual pages" of the display of the interpolation point table. This way, all 240 lines of the interpolation point table can be displayed on the HMI Interface.

SIMATIC WinCC flexibl	e Runtime					
MC_MovePath -	PointTable Editor					Technology CPU
PointTable Data Block Edi	itor					
			Create	Open Save	e Abort De	3-Number 1001
Line	InterpolationPoint (>	/Y/Z)	C	ornerDistance	Override	M-Fkt.
1	10.000 2	0.000	0.000	0.000	100.000	0
2	10.000 1	0.000	0.000	-1.000	100.000	1
3	10.000 4	0.000	0.000	10.000	100.000	17
4).000	0.000	10.000	100.000	1
5	1).000	0.000	-1.000	100.000	0
6).000	0.000	10.000	100.000	0
7		0.000	0.000	10.000	100.000	0
8).000	0.000	-1.000	100.000	1
10			0.000	10.000	100.000	1
Done	Active Erro			Page 1	of 24	< >
Administrat						
>>> /	t a number of lines	StartLine		$\sim \sim$	0	
	•	page r	iys the pro respective lus data b	ely the	respecti	uent page

Figure 2-5 Editing the interpolation point table within a data block

If the interpolation point table is opened, there also exists the possibility to perform administrative functions within the table, such as inserting and deleting lines or copying lines within the data block of the interpolation point table.



Figure 2-6 Administrative functions within the interpolation point table

2.3 Administration of the interpolation point tables

Interpolation point tables that already exist in the technology CPU are administrated by using the function blocks FB 553 "HMI_DB_Explorer_MP" and FB 554 "Delete_DB_MP" or via the page "MC_MovePath – DB Explorer" of the HMI Interface.

2.3.1 Function block FB 553 "HMI_DB_Explorer_MP"

The function block FB 553 "HMI_DB_Explorer_MP" allows browsing the main memory of the technology CPU for data blocks, in which interpolation point tables for the FB 488 "MC_MovePath" are stored.

To identify a data block as interpolation point table, the size of 5280 bytes of the data block is decisive. The DB Explorer does not recognize other data blocks.

The FB 553 "HMI_DB_Explorer_MP" uses the SFC 24 "TEST_DB" to recognize interpolation point tables or to determine data sizes by applying this system function to all data block numbers from DB 1 to DB 2047.

- **Note** A maximum of 100 data blocks can be captured via the function block FB 553 "HMI_DB_Explorer_MP" in the RAM of the CPU.
- **Note** The searching process can result in an increased cycle load within the CPU. For this reason, it might be better to avoid using the DB Explorer during time-critical processes.

2.3.2 Function block FB 554 "Delete_DB_MP"

Data blocks can be deleted from the main memory of the technology CPU by using the function block FB 554 "Delete_DB_MP". The function block uses the system function SFC 23 "DEL_DB" to delete the data blocks.

Due to safety reasons, it is, however, only possible to delete data blocks with a size of 5280 Byte from the main memory of the technology CPU. The system function SFC 24 "TEST_DB" is used for determining the data block size.

Thus, no longer required interpolation point tables can be removed from the main memory of the technology CPU in order to provide space for new interpolation point tables which are to be created.

2.3.3 HMI "MC_MovePath – DB Explorer"

All functions of the function blocks 553 "HMI_DB_Explorer_MP" and FB 554 "Delete_DB_MP" can be operated via the page "MC_MovePath – DB Explorer" of the HMI Interface:

SIMATIC WinCC flexible Runtime			
MC_MovePath - DB Exp	orer		Technology CPU
Data Block Explorer - Point	able Data Blocks		
		Start Exploring	Stop Exploring
1001 1002			
Done Busy	Error		Actual DB 0
Done Busy	Error		
Administration - Delete DB	Created in CPU)		
DB-Number 0	Done Busy Error Delet	te DB Continue Deleting	Abort Deleting
Main Screen Ax	is Control FB : PathObject F	FB : GroupFunctions PointTable Ed	itor DB Explorer

Figure 2-7 HMI "MC MovePath - DB Explorer"

The DB-Explorer checks the presence of the respective data block in the CPU main memory for the data block numbers DB 1 to DB 2047 and displays the number on the HMI Interface if the building block features a size of 5280 byte and is therefore identified as interpolation point table.

The search can be started and stopped via the buttons "Start Exploring" and "Stop Exploring". The data blocks found until the search process is stopped are listed in the display range of the operator screen. The process is automatically stopped when the whole range of the data block numbers has been searched through.

- **Note** A maximum of 100 data blocks can be captured via the function block FB 553 "HMI_DB_Explorer_MP" in the RAM of the CPU.
- **Note** The searching process can result in an increased cycle load within the CPU. For this reason, it might be better to avoid using the DB Explorer during time-critical processes.

Within in the administrative area of the HMI Interface, data blocks can be deleted from the main memory of the technology CPU. The deleting process can be started by indicating the data block number and confirmed by pressing the button "Delete DB". If you wish to finally delete the data block, the deletion process needs to be confirmed via the button "Continue Deleting". You can stop the deletion process by pressing the button "Abort Deleting". Consequently, the selected data block will not be deleted from the main memory of the technology CPU.





Note For safety reasons, you can only delete data blocks with a size of 5280 bytes, i.e. if it can be assumed that these data blocks are interpolation point tables.

2.4 Data block of an interpolation point table

2.4.1 Data block structure

The data block structure of an interpolation point table is determined via one of the following user-defined data types:

- UDT 121 "PathData"
- UDT 120 "PathPoint"
- **Note** When using the UDT 121 "PathData" to define the interpolation point table, the UDT 120 "PathPoint" must be available in the program as well!

UDT 121 "PathData"

This data type defines length and structure of the interpolation point table. By means of this UDT, the interpolation point table can be defined for a path motion in a data block.

The UDT, and thus the data block, feature the following structure:

Figure 2-9 UDT 121 "PathData"

Address	Name	Туре	Initial value	Comment
0.0		STRUCT		
+0.0	Point	ARRAY[1240]		Collection of points defining a segment
*22.0		"PathPoint"		
=5280.0		END_STRUCT		

UDT 120 "PathPoint"

This data type defines the individual interpolation points of the interpolation point table.

The UDT features the following structure:

Figure 2-10 UDT 120 "PathPoint"

Address	Name	Туре	Initial value	Comment
0.0		STRUCT		
+0.0	х	REAL	0.000000e+000	X-Coordinate
+4.0	Y	REAL	0.000000e+000	Y-Coordinate
+8.0	Z	REAL	0.000000e+000	Z-Coordinate
+12.0	CornerDistance	REAL	0.000000e+000	Corner distance for transition start.
+16.0	Override	REAL	1.000000e+002	Velocity override for path velocity.
+20.0	M_Function	INT	0	Spezial function code for path point.
=22.0		END_STRUCT		

The individual elements of the interpolation points have the following meaning or function, which can also be selected via the HMI Interface when editing the interpolation point table:

Table 2-1 Description of parameters

Parameter	Description
X/Y/Z	Description of the coordinates of the respective interpolation point. For two-dimensional kinematics, the coordinate not in use (coordinate outside the processing level) does not have to be defined.
CornerDistance	Corner distance (ball-shaped around the defined interpolation point) within which the transition between the two straight lines of the motion path takes place.
	Stop point
	If a <u>negative</u> corner distance is entered, the path motion is stopped at this interpolation point. The motion can then be continued via the technology function FB 483 "MC_GroupContinue".
	If the job at FB 483 "MC_GroupContinue" is started directly before reaching the stop point, the interruption of the path motion can be skipped until the next stop point, i.e. there is no interruption of the motion.

Parameter	Description
Override	Through the override parameter, the travel speed of the respective path segment following the interpolation point can be influenced.
	If the table is traversed from the first to the last interpolation point, each override will have an effect on the path segment following the interpolation point. If the table is followed in reverse direction (last till first interpolation point), the corresponding override setting of the previous interpolation point will have an effect on the path segment.
M_Function	User-definable interpolation point detection. As soon as the interpolation point was passed by the motion, the value defined at the output code of FB 488 "MC_MovePath" is output until reaching the next interpolation point.
	Via this parameter, additional functions (e.g. open/close grippers) can be selected, which are to be carried out at certain path positions.

The following figure serves as an example for the definition of a path motion via individual interpolation points.





3 Installation

3.1 Basic proceeding

To integrate the HMI Interface for administration and processing of the interpolation point table of the FB 488 "MC_MovePath" into your own application, you need to proceed as follows:

Step	Process / Action	Required Software / Tools
1.	Copy the required function blocks into the target application. Dont' forget to copy also the corresponding SCL sources together with the function blocks into the target application, otherwise you might be facing problems during compiling of the whole project.	SIMATIC Manager STEP 7
2.	Enter the function block calls in OB 1 of the target program.	
3.	Load the modified user program or the modified STEP 7 blocks into the SIMATIC CPU.	
4.	Prepare the project texts in WinCC flexible for the transfer from the source project to the target project.	WinCC flexible
5.	Copy the required pages of the HMI Interface into the HMI Interface of the target application or target project.	
6.	Adjust the picture names if necessary, so that they match the target project.	
7.	Delete the additional connections resulting from copying the pages of the HMI Interface.	
8.	Allocate the variables of the HMI Interface pages to the still existing connection (entries in variable table highlighted in orange).	
9.	Reconnect the variables of the HMI Interface pages by means of the address and data type without changing the names of the variables.	
10.	Integrate the page call into the operating structure of the HMI Interface of the target project.	
11.	Save the changes in the HMI project and recreate the HMI Interface.	

Table 3-1 Basic proceeding for the installation of the functionality

3.2 Working within the SIMATIC Manager

3.2.1 Unzipping process and content of the STEP 7 source project

Unzip the STEP 7 archive of the source project, in which the STEP 7 blocks required for the functionality and the HMI Interface are contained.

D 🛩 🏭 🐖 X 🖻 🛍 🏜	© º_ º_ :- :::	and the Charles				
Denote the second second		Ko Filter >		- 20	않 🛞 🔣 🖷 🖃	□ \ \?
- 🎒 MovePath_Editor_EXP	Object name	Symbolic name	Created	Size	Туре	Version (Header)
🗄 🔠 317TF-2 DP	Systemdaten	•••	•••	•••	SDB	•••
🖻 📲 CPU 317TF-2 DP	🕀 0B1	MAIN_CYCLE	STL	140	Organization Block	0.1
⊡ 🔄 S7:Programm(1)	🕵 FB551	HMI_List_MP	SCL	13016	Function Block	1.0
D Quellen	🚰 FB553	HMI_DB_Explorer_MP	SCL	468	Function Block	1.0
Bausteine	🚰 FB554	Delete_DB_MP	SCL	342	Function Block	1.0
E and technologie	DB551	idb_HMI_List_MP	DB	5590	Instance data block	0.0
SIMATIC HMI-Station	🕀 DB553	idb_HMI_DB_Explorer_MP	DB	248	Instance data block	0.0
	🕀 DB554	idb_Delete_DB_MP	DB	46	Instance data block	0.0
	UDT120	PathPoint	STL		Data Type	4.2
D Quellen	SFC22	CREAT_DB	STL		System function	1.0
Bausteine	SFC23	DEL_DB	STL		System function	1.0
_	🚰 SFC24	TEST_DB	STL		System function	1.0

Figure 3-1 Content of the STEP 7 source project

The individual folders of the STEP 7 source project have the following meaning:



Figure 3-2 Project tree of the STEP 7 source project

3.2.2 Transfer of the function blocks to the target project

To transfer the function blocks from the STEP 7 source project to the target project, copy all function blocks (FB), instance data blocks (DB), user-defined data tyes (UDT) and system functions (SFC) into the target project.

Subsequently, reconnect the function blocks (FB) to the instance data blocks (DB) by calling the function blocks in OB 1, or a cyclically called function block (FB), or a function (FC). Instead of individually calling the function blocks, you can also transfer Network 1, which is contained in OB 1 of the source project, to the target project.

Note Do not change the numbers of the function blocks and the instance data blocks when transferring the blocks from the source project to the target project. Otherwise you might be facing problems when connecting the HMI Interface.

```
OB1 : "Main Program Sweep (Cycle)"
Comment:
Network 1: Additional Tools
Comment:
      CALL "HMI_List_MP" , "idb_HMI_List_MP"
                                                             FB551 / DB551
      DB_Number
                     :=
      Generate_DB
                     : =
      Open DB
                     :=
      Close_DB
                     :=
      Abort DB
                     :=
      Prev_Block
                     :=
      Next_Block
                     :=
      InsertLines
                     :=
      DeleteLines
                     :=
      CopyLines
                     :=
       StartLine
                    :=
       Quantity
                     : =
      DestinationLine:=
      Done
                    :=
      Busv
                     :=
                    :=
      Active
      Error
                     :=
      ActPage
                    :=
      MaxPage
                     :=
      CALL "HMI_DB_Explorer_MP" , "idb_HMI_DB_Explorer_MP"
                                                           FB553 / DB553
      StartExplore:=
      StopExplore :=
      Done
                 : =
                 :=
      Busv
      Error
                 :=
      Act_DB
                 :=
      CALL "Delete_DB_MP" , "idb_Delete_DB_MP"
                                                            FB554 / DB554
                 :=
      DB_Number
      Delete_DB
      Del Continue:=
      Del_Abort :=
                :=
      Done
      Busv
       Error
                :=
```

Figure 3-3 Network 1 from OB 1 of the source project

3.3 Working within WinCC flexible

3.3.1 Preparation of the project texts

In order to be able to completely and successfully copy the HMI Interface pages, all project texts need to be prepared accordingly.

Basic requirements:

There must be at least one match in the project languages of both projects.

If there is no language in the project texts, which exists in both, the source project and the target project, we recommend the following proceeding:

1. Create a language in the project texts of the source project, which also exists in the target project.

		ROJECT LANGUAGE
eneral		
Editing language	Reference language	
German (Germany)	English (United States)	
elect project languages		
Afrikaans (South Africa)	French (France)	Serbian (Cyrillic) (Serbia)
Albanian (Albania)	French (Luxembourg)	Serbian (Latin) (Serbia)
Azeri (Cyrillic) (Azerbaijan)	French (Principality of Monaco)	📃 Slovak (Slovakia)
Azeri (Latin) (Azerbaijan)	French (Switzerland)	📃 Slovenian (Slovenia)
Basque (Spain)	FYRO Macedonian (Former Yugoslav Republic of Macedonia)	Spanish (Argentina)
Belarusian (Belarus)	Galician (Spain)	Spanish (Bolivia)
Bulgarian (Bulgaria)	German (Austria)	Spanish (Chile)
Catalan (Spain)	German (Germany)	Spanish (Colombia)
Croatian (Croatia)	German (Liechtenstein)	Spanish (Costa Rica)
Czech (Czech Republic)	German (Luxembourg)	Spanish (Dominican Republic)
Danish (Denmark)	German (Switzerland)	Spanish (Ecuador)
Dutch (Belgium)	Greek (Greece)	Spanish (El Salvador)
Dutch (Netherlands)	Hungarian (Hungary)	Spanish (Guatemala)
English (Australia)	Icelandic (Iceland)	Spanish (Honduras)
English (Belize)	🔲 Indonesian (Indonesia)	Spanish (International Sort)
English (Canada)	Italian (Italy)	Spanish (Mexico)
English (Caribbean)	Italian (Switzerland)	Spanish (Nicaragua)
English (Ireland)	Kazakh (Kazakhstan)	Spanish (Panama)
English (Jamaica)	Kyrgyz (Cyrillic) (Kyrgyzstan)	Spanish (Paraguay)
English (New Zealand)	Latvian (Latvia)	Spanish (Peru)
English (Republic of the Philippines)	Lithuanian (Lithuania)	Spanish (Puerto Rico)
English (South Africa)	Malay (Brunei Darussalam)	Spanish (Uruguay)
English (Trinidad and Tobago)	Malay (Malaysia)	Spanish (Venezuela)
English (United Kingdom)	Mongolian (Cyrillic) (Mongolia)	Swahili (Kenya)
English (United States)	Norwegian (Bokmal) (Norway)	Swedish (Finland)
English (Zimbabwe)	Norwegian (Nynorsk) (Norway)	Swedish (Sweden)
Estonian (Estonia)	Polish (Poland)	Tatar (Tatarstan)
Faeroese (Faeroe Islands)	Portuguese (Brazil)	Turkish (Turkey)
Finnish (Finland)	Portuguese (Portugal)	Ukrainian (Ukraine)
French (Belgium)	Romanian (Romania)	Uzbek (Cyrillic) (Uzbekistan)
French (Canada)	Russian (Russia)	Uzbek (Latin) (Uzbekistan)

Figure 3-4 Languages available in the WinCC flexible project

2. Copy the labels of the HMI pages, which are to be copied, into the newly created language from one of the languages of your source project, e.g. the creation language.

	Ba. 2 B ↔ Ma. 4 . Ba. 8 ↔ Ma. Pope		
nary) 👱 .			
(Project Texts		
IC HMI-Station(WinCC Beable Runtime)			PROJECT T
(bary)		(11.551.511
Add Screen	German (Germany)	Referenced by	Ingin (United States)
Template 01 PointTableEditor MP	e gan Pactes	CONTINUES AND	- QGR
02 DB Explorer MP	Poel	 CLIPHA I JL THE COLICET (AddataviorSetting) (Schem Navigation) (Invisible) (Schemhavigation) CLIPHA I JL THE Station/SobalDeviceSetting(Screen Navigation) (Invisible) (Schemhavigation) 	tione
invitation	Unten	 Carear A, Her Scout Packader Revise Settings (Screen Kavigation (Instable (Screen Nevigation)) Carear Nevigation 	Down
Law	Late	USIMATIC HMI-Station/SideaDeviceSetting/SovenNavigation/Invisible/SovenNavigation	Left
Tag: Connections	E Chan	C TREATS WILL BUILD all soluble data and all soluble data and an internal solution of the soluble data and the solution of	0.
Dycles	1 <	ISINATIC HMI Station/HmisoreenFeider(01_PointTableEdex_HPEschaltTade_1	
ni Management	Abort	(SIMATECHMI-Station)HmiScreenFelder(0)_PointTableEdter_HF(SchaltRache_LD	Abort
spei	Teot	(SIMATICHMI Station/PrintScreen/Folder(DI_PointTableEdtor_PIP(SchabSadie_L3	Text
orical Data		(SP4AT3CHMI-Station)HniSchen/Filder(DI_PointTableEdtor_HP(SchabtNate.2	
pis sets	Open	LSINATSCHMI Statun/preiSonen/hider(01_pointTableEdex_pm(schabiliste_4 LSINATSCHMI-Statun/pmiSonen/Edder(01_pointTableEdex_pm(schabiliste_5	Open
orts Land Graphics Lists	Save Coate		Save Create
Text Lists	Create Insert a number of irres		Literation of li
Graphice Lists	Delete a number of lices		Delete a number of lines
time Uper Administration	Copy a number of lines to		Copy a number of lines to
ice Setting:	Ure	(SINATICHMI-Station)/initioneenFolder(01_PointTableEditor_PF(Textfold_1	Ure
pe Settings	a d	(SIPATIC HMI Station)/millioreen/Folder(01_PointTableEdear_HMI(textFold_32	at
ect Longuages	Athe	(SIMATICHMI-Station)HmiScreenFolder(0)_PointTableEdtar_HF(Textfeld_3)	Adhre
shics	Satile Satile	(SINATICHMI Statur/presscreenfelder(DI_PointTableEdtor_PIP)(TextFelE_PI	Satine
ect Texts	Number of lines	(SPATICHMI-Station/pini/ScreenFeider(0)_PointTableEdtor_PPUestfeld_p5	Number of lines
ionaries	Copy to line	STRATSCHMI Status/ymiScreenPolder(0)_PointTableEditar_PR(Textfiel)_35 ISBNTCHMI-Status/emiScreenFolder(0)_PointTableEditar_PR(Textfiel)_05	Copy to line Adventuation
n Management	PointTable Data Block Editor	GJPALIC PRESIDENT PROCEEPENDERUS PORTADECOS / PEUESTED_7 ISTRATIC PRE-Staten/IntelSovenPoder(C) PortTableEdtor / PEUestfeld 35	Administration Frint Table Data Rinck Editor
Management	Pontiable Data slook Editor		Punctable Lona Book Editor
	InterpolationPoint (QV)(Z)	ISINATIO PRI Station Principer Pader (or principale Calify Principale)	InterpolationPoint (17/52)
	Technology (PU		Technology OPU
	Convertigance	SIPATE HMI-Station/HmiGoverFulder(0) PointTableEdbar HPITectfrid 4	ConeOstaton
	MC_MovePath - PointTable Editor	ISINATIC HMI Station/ImmiGreen/Folder(0) PointTableEditor PIF(TextFeld 5	MC MovePath - PointTable Editor
	Overside	(STMATECHMI Station/HistiGovernFolder(01_PointTableEdetry_HP(Textfield 6	Override
	I H-Fit.	(SIMATIC HMI-Status/presScreen#sider(DI_PointTableEdtor_P#)(TextFeb[7	M-Rz.
	E Page	(SIMATICHMI-Station/FiniScreenFeider(D)_PointTableEdtor_PPUestFeb;9	Fage
	PointTable Editor	(SIMATIC HMI-Station/HmiScreen/Edder(CC_D6_Explorer_MF(SchaldSadw_J2	PointTable Editor
	D8 Explorer	USINATICHMI-Station/HmiSoreenFeider(U2_DB_Explorer_IMP(schaftfliche_13) USINATICHMI-Station/HmiSoreenFeider(U2_DB_Explorer_IMP(schaftfliche_5))	16 Explorer
	Stop Exploring		Stop Exploring
	Start Exploring Delete 08		Start Exploring Delete 06
	Continue Deleting		Continue Deleting
	Abort Deleting		Abort Deleting
	I + Dane	USINATISCHMI (Statun'envisionen/tuider(US, D6 Explorer MPUTextfeld 30	Done
	I Dury	(SPMATE HMI-Station/HmiSover-Folder(Q2_D0_Explore_JPF(TextFeld_1)	buty
	Data Block Explorer - PointTable Data Blocks	(S3NAT3CHMS-Statum/philScreenPalder(CC_D6_Explorer_)/P(Textfeld_35	Data Block Explorer - PointTable Data Blocks
	Administration - Delete DB (Created in CPU)	ISIMATICHMI-Station/HmiScreenFelder(U2_DR_Explorer_HPUextfeld_20	Administration - Delete D6 (Created in CPU)
	Evror	(SINATIC HMI-Station)/HMI-ScreenPolder(02_06_Explorer_HPI(Textfold_24	Eror
	MC_MovePath - DB Explorer	USIMATIC HMI-Station/HmiSoreerFelder(UC_DB_Explorer_MPUText#d15	MC_MovePath - 06 Explorer
	D8-Nunber	(SINATSCHM-Station/HistoreenFilder(CL_DE_Explorer_JMF(Excthel.) (SINATSCHM) Station/HistoreenFilder(CL_DE_Explorer_FMF(Excthel.))	16 Number
	Actual CO	CLIMATEC MML Statum Printsoreen Prider (UC_DE_Explorer_PP/DevtTeld) (SIMATEC MML Statum (NessagingFolder Path (SettingeFolder Path) (NessaginGasel Islam Path) Name	Actual DB
		 Careat 2, TPG - Station precessingly coder vitry participations in precession cash removance 	
		LIMATECHNI-Station/MessagingFolderPath/SettingsFolderPath/MessageClassFoldeFath/Events	0
	2	CLIMAL JL, HML Scatton (Hessagnor) oder Ham pettingsr oder Ham (Hessage Lassr-oder Ham) virts (20144732 HML Scatton (Hessagnor) oder Path (Settingsr oder Path (Hessage Classr oder Ham) virts	2
		 CLIPHA I JL, PHA COLICITI JHRESSAGINGF BORN VAIN (SettingsFolder Path) (Hessage ClassFolder Hattigetes CLIPHA I JL, PHA COLICITI JHRESSAGINGFolder Path) (SettingsFolder Path) (Hessage ClassFolder Path) (SettingsFolder Path) 	
		1314473C HMI-Station' MessagingPolder Path/SettingsPolder Path/Message ClassPolderPath/Setem	
	Vervaken	 Carrier of the control presenting of the end of the of the en	P Administration
	Bedenen	(31%AT3CHMI-Station/RTUserAdmin)Authoritations(1	Operate
	II Überwachen	- ISBNATICHMI-Station/PTUserAdmir/Authorizations/2	Manitor
	Gruppe (9)	(SINATECHMI-Station)/PTUserAdmin)/Siroups)/Administrations	Group (9)
	Group identical texts		

Figure 3-5 Copying the labels of the HMI pages

To do this, proceed as follows:

- Sort the project text list in ascending or descending order according to the references.
- Mark all entries belonging to the HMI pages, which are to be copied, in the already existing language and select the function "Copy" from the context menu.

- Mark the same entries in the list of the newly added language and select the function "Paste" from the context menu.

Thus, the project texts belonging to the HMI pages are available in a language which also exists in the target project and can now be copied, together with the HMI pages, into the target project without any problems.



Figure 3-6 Project texts: Mark and copy source





Note

The list of project texts and the settings for the project languages are located in the WinCC flexible project tree under the following entry:

SIMATIC HMI-STATION (WinCC flexible Runtime) > Language Settings > ...

3.3.2 Copying HMI pages into the target project

Open the source project and target project of the HMI Interface each in a seperate instance of WinCC flexible.

Select the page to be copied of the HMI Interface in the project tree under the section "Screens" and copy the respective page into the target project by means of the context menu (right mouse click) using the function "Copy".

Note The pages of the HMI Interface have to be copied one by one from the source project into the target project. It is not possible to collectively copy all pages.

WinCC flexible Advanced - MovePath_Editor_E	CP - SIMATIC HMI-Station			
Project Edit Yew Insert Format Paceplates Op				
Geman (Gemany) 💌 🚬	■ B Z U B . ■ N N N . 4 A A A	(3) 國家開幕會理會國家國家,	🧑 🖉 Al Select 🔹 🛛 🔅 🖉 🔅	× .
Project 💡 🗙	01_PointTableEditor_MP			Tools 🦓 🕱
Iw Prinkt				A 1.2.8
SIMATIC HMI Station(WinCC Rooble Puntime)	MC_MovePath - PointTable Editor		Technology CPU	Simple Objects
Screen:				Line
- I emplate	PointTable Data Block Editor			7 Palyline
CI Port		Create Open Save	Abort D8-Number 00100	A Palegon
B Communical Rename			in the second se	C Elipse
S Connects C Undo Cort+3	, Line InterpolationPoint (X/Y/Z)	CornerDistance	Override M-Fkt.	Circle Rectangle
Cycles X Cut, Col+0 Alam Manac Corv Col+0	000 0000000,000 000000,000	00000000,000 0000000,000	0000000,000 00000000 000000000000000000	A Textfield
	000 0000000,000 000000,000	00000000,000	60505000,700 00000000	a) to neld
Historical Da Fist Copy Scripts Scripts Column 2 Column 2	000 000000,000 000100,000	01010000,010 000101,000	000000000000000000000000000000000000000	State-Time Field
🛪 🧺 Reports			······································	& Graphic 10 Field
Tend List: X Delete Delete	000 0000000,000 000000,000	01010000,010 000101,000	6000000,000 0000000	Graphics View
Baphics Brit Selection Onlyw Brit Selection Onlyw	000 0000000,000 000000,000	000,0000000 000,000	60000000,000 00000000	20 Button
🐵 🎦 Device Sette 🞦 Cross-References	000 0000000,000 000000,000	01010100.010	000000000000000000000000000000000000000	ET Switch
Export texts Project Lang	000 000000,000 000000,000	0000000.000 000000.000	000000000000000000000000000000000000000	Ber .
Graphics Import texts				
Project Tests Object properties	000 0000000,000 000000,000	0000000,000 0000000,000	6000000,000 0000000	
Stuctures Version Management	000 0000000,000 000000,000	000,0000000 000,000	6000000,000 00000000	
	000 0000000,000 010100,000	01010100.010 010101010	000000000000 00000000000000000000000000	
	Done Busy Active Error	Page 000	e > 000 le	
	Dutte Basy Active Error	Page 000		
	Administration			
	>>> Insert a number of lines	StartLine	000	
	>>> Delete a number of lines	Number of lines	000	
	222 Copy a number of lines to	Copy to line	000	
	PointTable Editor		DB Explorer	
				v
	01_PointTableEditor_MP (Screen)			() ×
	General			General
	Properties			General
	Animations Settlings Events Name 01_PointTable			Enhanced Objects
	Number 12			Ny Controls
	Use template 🗹			Graphics Ubrary
	Background color			
				Drop any object here to delete it.
🗂 Output 🙀				

Figure 3-8 Copying the HMI pages

Note To copy the pages from the HMI user interface, always select the function "Copy" from the context menu. Only by using this function, the page will be completely transferred into the target project.

If you use the function "Flat Copy" (simple copying process) the relations of the variables to the individual objects would get lost and have to be reconnected afterwards.

3.3.3 Adjusting the connections in the HMI project

When copying each HMI page separately, a new connection is established in the WinCC flexible project for this page. The communication in the target project should, however, be effected for all HMI pages of the user interface via one single connection.

For this reason, the newly created connections can be deleted from the WinCC flexible project again. To do so, mark the newly created connections and select the function "Delete" from the context menu.

TARR

Figure 3-9 Deleting additionally created connections

Name	Active	Communication driver	Station		Partner	Node	Online	Comment
Technology-CPU	On	SIMATIC S7 300/400	\Palletizer_MovePa	th\SIMATIC 300	CPU 317T-2 DP	MPI/DP	On	
Technology-CPU_0	On	SIMATIC 57 300/400					On	
Technology-CPU_1	On	SIMATIC S7 300/400	5	Add Connection			On	
			×	Cut	Ctrl+X			
				⊆ору	Ctrl+C			
			6	Flat Copy				
				Paste	Ctrl+V			
		<		Replace	Ctrl+E			
arameters Area p	ointer		×	Delete	Delete			
				Reconnect				
WinCC flexible Runtime		<u>(x)</u>	Cross-References				Station	
	Interface		PS-	Properties	Ctrl+Shift+X			#### 3 5 5 5 6

3.3.4 Adjusting the connection of the variables in the HMI project

When deleting the connections created during copying, the connections of the HMI variables to the technology CPU are lost. These have to be recreated now.

For this purpose, select an entry highlighted in orange from the variables table for the connection and subsequently mark the desired new connection in the dropdown list.

Name	Display name	Connection 🔍 Data type	Symbol	Address
idb_HMI_DB_Explorer_MP.DB_List[52]		Technology-C Technology-C	DB_List[52]	 DB 553 DBW 108
idb_HMI_DB_Explorer_MP.DB_List[34]		Technolo Icon Name	Info	DB 553 DBW 72
idb_HMI_DB_Explorer_MP.DB_List[43]		Technolo		DB 553 DBW 90
idb_HMI_DB_Explorer_MP.DB_List[97]		Technolo S Technology-CPU	CPU 317T-2 DP	DB 553 DBW 198
idb_HMI_DB_Explorer_MP.Error		Technolo		DB 553 DBX 2.2
idb_HMI_DB_Explorer_MP.DB_List[98]		Technolo		DB 553 DBW 200
idb_HMI_DB_Explorer_MP.DB_List[88]		Technolo		DB 553 DBW 180
idb_HMI_DB_Explorer_MP.DB_List[87]		Technolo		DB 553 DBW 178
idb_HMI_DB_Explorer_MP.DB_List[40]		Technolo		DB 553 DBW 84
idb_HMI_DB_Explorer_MP.DB_List[32]		Technolo		DB 553 DBW 68
idb_HMI_DB_Explorer_MP.DB_List[73]		Technolo <<< New	✓ ×	DB 553 DBW 150
idb_HMI_DB_Explorer_MP.DB_List[83]		Technology-CPU_1 Inc	Do Listicol	DB 553 DBW 170
idb_HMI_DB_Explorer_MP.DB_List[57]		Technology-CPU_1 Int	DB_List[57]	DB 553 DBW 118
idb_HMI_DB_Explorer_MP.DB_List[1]		Technology-CPU_1 Int	DB_List[1]	DB 553 DBW 6
idb_HMI_DB_Explorer_MP.DB_List[50]		Technology-CPU_1 Int	DB_List[50]	DB 553 DBW 104
idb_HMI_DB_Explorer_MP.DB_List[10]		Technology-CPU_1 Int	DB_List[10]	DB 553 DBW 24
idb_HMI_DB_Explorer_MP.DB_List[49]		Technology-CPU_1 Int	DB_List[49]	DB 553 DBW 102
idb_HMI_DB_Explorer_MP.DB_List[26]		Technology-CPU_1 Int	DB_List[26]	DB 553 DBW 56
idb_HMI_DB_Explorer_MP.DB_List[93]		Technology-CPU_1 Int	DB_List[93]	DB 553 DBW 190
idb_HMI_DB_Explorer_MP.DB_List[93] idb_HMI_DB_Explorer_MP.DB_List[4]		Technology-CPU_1 Int	DB_List[4]	DB 553 DBW 12
idb_Delete_DB_MP.Error		Technology-CPU 1 Bool	Error	DB 554 DBX 4.2
idb_HMI_DB_Explorer_MP.DB_List[33]		Technology-CPU_1 Int	DB_List[33]	DB 553 DBW 70
idb HMI DB Explorer MP.DB List[18]		Technology-CPU 1 Int	DB_List[18]	DB 553 DBW 40
idb_HMI_DB_Explorer_MP.DB_List[12]		Technology-CPU_1 Int	DB_List[12]	DB 553 DBW 28
idb_HMI_DB_Explorer_MP.DB_List[95]		Technology-CPU 1 Int	DB_List[95]	DB 553 DBW 194
idb_Delete_DB_MP.Done		Technology-CPU 1 Bool	Done	DB 554 DBX 4.0
idb HMI DB Explorer MP.StartExplore		Technology-CPU 1 Bool	StartExplore	DB 553 DBX 0.0
idb HMI DB Explorer MP.DB List[65]		Technology-CPU 1 Int	DB List[65]	DB 553 DBW 134
idb_HMI_DB_Explorer_MP.DB_List[94]		Technology-CPU 1 Int	DB_List[94]	DB 553 DBW 192
idb_HMI_DB_Explorer_MP.DB_List[35]		Technology-CPU 1 Int	DB_List[35]	DB 553 DBW 74
idb_HMI_DB_Explorer_MP.DB_List[35] idb_Delete_DB_MP.Del_Continue		Technology-CPU_1 Bool	Del_Continue	DB 554 DBX 2.1
idb_Delete_DB_MP.Del_Continue	<			>

Figure 3-10 Selecting a new connection

Once you have marked the newly selected connection, you can transfer the setting made to all other entries highlighted in orange by dragging the small square at the lower right corner of the marking.

Then you need to reconnect the variable provided with a new connection to the data source in the technology CPU. For this purpose, select the entry "Connections" from the context menu and the function "Reconnect" from the menu tree.

ect Edit View Inse	ert <u>F</u> ormat	Faceplates Options Window Help				
New 🔸 🍃 🧮 🖒	• Ci • X	🗴 🖬 🔒 🎺 🐻 🗛 🧏 👌 🖬 🤅 前 🖉 😚	🖟 🖏 Technology-CPU 🛛 🖌 🍕	?: ?		
glish (United States)	× .					
	ex 1	01_MainScreen				90
oject Project		01_MainScreen S ^a Connections Tags				~ ~
SIMATIC HMI Station	WinCC fles					TAG
😑 🔚 Screens		Name	Display name	Connection V Dat	a type Symbol	Address
		idb_HMI_List_MP.ContourList[4].CornerDistance		Technology-CPU Real	<undefined></undefined>	DB 551 DBD 94
- Template 01 MainScre		idb_HMI_List_MP.DestinationLine		Technology-CPU Int	<undefined></undefined>	DB 551 DBW 8
01_Manscree		idb_HMI_List_MP.ContourList[9].V		Technology-CPU Real	<undefined></undefined>	DB 551 DBD 196
02_DB_Explo	xer_MP	idb_HMI_List_MP.ContourList[5].Override		Technology-CPU Real	<undefined></undefined>	D6 551 D6D 120
02_FunctionE		idb_HMI_List_MP.ContourList[3].Z		Technology-CPU Real	<undefined></undefined>	DB 551 DBD 68
03_FunctionE		kb_HMI_List_MP.ContourList[2].CornerDistance		Technology-CPU Real	<undefined></undefined>	DB 551 DBD 50
Screen 1	u 🛛	idb HMI List MP.ContourList[4].Override		Technology-CPU Real	<undefined></undefined>	D8 551 D8D 98
E Tommunication				Technology-CPU Int	<undefined></undefined>	DB 551 DBW 58
Tags		idb_HMI_List_MP.ContourList[6].Override		Technology-CPU Real	<undefined></undefined>	DB 551 DBD 142
S Connections	Open editor			Technology-CPU Real	<undefined></undefined>	DB 551 DBD 64
B 🔄 Alarm Manac 🤇	Add Connecti			Technology-CPU Int	<undefined></undefined>	DB 551 DBW 190
😽 Analog A				Technology-CPU Real	<undefined></undefined>	DB 551 DBD 134
Discrete	Generate			Technology-CPU Bool	<undefined></undefined>	DB 551 DBX 3.0
🗄 🐙 Settings 🖌	Undo-Pasting	of Screen 02_DB_Explorer_MP Ctrl+Z >		Technology-CPU Real	<undefined></undefined>	DB 551 DBD 126
🗄 🚾 Historical Da 📈	Cut	Otrl+X		Technology-CPU Real	<undefined></undefined>	D6 551 D6D 38
🗈 🊾 Scripts 🛛 📭	Copy	Ctrl+C		Technology-CPU Int	<undefined></undefined>	DB 551 DBW 124
🗄 🧏 Reports 🛛 🕞	Flat Copy			Technology-CPU Real	<undefined></undefined>	DB 551 DBD 138
😨 🚧 Text and Gra 🛄 🗑 🎽 Runtime Use 🛍	Paste	Ch1+V		Technology-CPU Int	<undefined></undefined>	DB 551 DBW 14
æ 🚝 Horitille Ose	Replace	Orl+E		Technology-CPU Real	<undefined></undefined>	DB 551 DBD 54
😋 Language Settin	Delete	Delete		Technology-CPU Int	<undefined></undefined>	DB 551 DBW 36
Project Lang				Technology-CPU Bool	<undefined></undefined>	DB 551 DBX 10.1
Graphics	Print Selection			Technology-CPU Real	<undefined></undefined>	DB 551 DBD 72
Dictionaries	Cross-Referen	nces		Technology-CPU Bool	<undefined></undefined>	DB 551 DBX 2.1
Structures	Reconnect			Technology-CPU Int	<undefined></undefined>	DB 551 DBW 240
🗧 Version Manager	Object proper	rties		Technology-CPU Real	<undefined></undefined>	DB 551 DBD 104
				Technology-CPU Bool	<undefined></undefined>	DB 551 DBX 10.2
		idb_HMI_List_MP.ContourList[9].X		Technology-CPU Real	<undefined></undefined>	DB 551 DBD 192
		HMI_Interface.MovePath.Execute		Technology-CPU Bool	Execute	DB 500 DBX 30.0
		HMI_Interface.MovePath.Direction		Technology-CPU Int	Direction	DB 500 DBW 34
		HMI_Interface.Axis_SyncFollow.Reset.Commandaborted		Technology-CPU Bool	Commandaborted	DB 500 DBX 290.4
		HMI Interface.GroupStop.Active		Technology-CPU Bool	Active	DB 500 DBX 80.2

Figure 3-11 Reconnecting variables

Select the settings for the connection of the variables as shown in the following figure. However, do not forget to uncheck the setting "Replace tag name with symbol name", otherwise the names of the variables will also be readjusted.

Figure 3-12 Settings for the function "Reconnect Variables"

We Options for Reconnecting Symbols	\mathbf{X}
Connect STEP7/SIMOTION symbols with WinCC flexible tags if:	
symbol name, address and data type match	
⊙ one or more of the following conditions are true:	
symbol name matches (address and data type are not relevant)	
✓ address and data type match (symbol name is not relevant)	
Replace tag name with symbol name	
ок	Cancel

Once the connections of the variables have been recreated successfully, the symbol names of the variables from the data blocks will reappear in the variables list.

Figure 3-13 Reconnected variables

						TAG	5
	Name	Display name	Connection 🔍	Data type	Symbol	Address	
	idb_HMI_DB_Explorer_MP.DB_List[52]		Technology-CPU	Int 🔻	DB_List[52]	DB 553 DBW 108	^
	idb_HMI_DB_Explorer_MP.DB_List[34]		Technology-CPU	Int	DB_List[34]	DB 553 DBW 72	-
=	idb_HMI_DB_Explorer_MP.DB_List[43]		Technology-CPU	Int	DB_List[43]	DB 553 DBW 90	
=	idb_HMI_DB_Explorer_MP.DB_List[97]		Technology-CPU	Int	DB_List[97]	DB 553 DBW 198	
-	idh HMI DB Evolorer MD Error		Technology-CPU	Bool	Error	DB 553 DBV 2-2	

3.3.5 Transfer of the HMI pages to the operatering structure

Now the operator screens have to be transferred to the operator structure of the HMI user interface of the target project.

The operator screens can, for example, be activated via the menu buttons at the bottom of the HMI Interface. For this purpose, insert the appropriate buttons into the HMI user interface, add the property "ActivateScreen" to these buttons and select the respective HMI page as picture name.

Figure 3-14 Calling the HMI pages via buttons

eneral roperties	X		Function
nimations	1	ActivateScreen	
vents Click		Screen name	01_PointTableEditor_MP
Press		Object number	0
Release	2	<no function=""></no>	
Activate			
Deactivate Change			
Change			

Figure 3-15 Menu buttons at the bottom of the HMI Interface

SIMATIC	WinCC flexible Runtim	e				
MC_M	ovePath - Point	Table Editor				Technology CPU
PointTab	le Data Block Editor					
			Create	e Open Sa	ve Abort De	-Number 1001
Line		InterpolationPoint (X/Y/Z)		CornerDistance	Override	M-Fkt.
1	10.000	20.000	0.000	0.000	100.000	0
2	10.000	10.000	0.000	-1.000	100.000	1
3	10.000	40.000	0.000	10.000	100.000	17
4	60.000	40.000	0.000	10.000	100.000	1
5	60.000	10.000	0.000	-1.000	100.000	0
6	60.000	40.000	0.000	10.000	100.000	0
7	10.000	40.000	0.000	10.000	100.000	0
8	10.000	10.000	0.000	-1.000	100.000	1
9	10.000	40.000	0.000	10.000	100.000	17
10	70.000	40.000	0.000	10.000	100.000	1
Done	e Busy	Active Error		Page 1	of 24	< >
Administ	ration					
>>>	Insert a numb	er of lines	StartLine		0	
>>>	>>> Delete a number of lines Number of lines				0	
>>>	>>> Copy a number of lines to Copy to line				0	_
_						
Mai	in Screen	Axis Control	FB : PathObject	FB : GroupFunctions	PointTable Editor	DB Explorer

4 Sample Program

4.1 Basis

The STEP 7 Program offered as download in the following FAQ serves as basis for the sample program:

FAQ How can a simple palletizer be realized with a technology CPU and the FB 488 "MC_MovePath"?

Link: http://support.automation.siemens.com/WW/view/en/48206063

Please refer to the documentation of the above mentioned FAQ for a detailled description of the sample program operation.

4.2 **Restrictions within the interpolation point tables**

The following restrictions apply for the creation of interpolation point tables via the PointTable Editor in the sample program:

- On the HMI user interface, motions can only be shown two-dimensionally (2D); for this reason, the interpolation points in the interpolation point table should only be entered with X and Y coordinates. Motions in Z direction are not displayed.
- To represent the interpolation point positions on the HMI Interface, the following value ranges are to be kept:
 - X coordinates: $0 \le X \le 100$
 - Y coordinates: $0 \le X \le 70$

Figure 4-1 Sample program



4.3 Selection of the interpolation point tables

Within the sample program, the interpolation point table is selected via the operating range "FB: PathObject". If a new interpolation point table has been created in the technology CPU via the PointTable Editor, it has to be entered here for processing.

Figure 4-2 Selecting the interpolation point table via the operating range "FB: PathObject"



The FC 530 "Path_IdxToAny" is used for creating the ANY pointer at the input of the technology function FB 489 "MC_PathSelect", via which the data block number and the StartIndex or EndIndex can be entered directly.

Figure 4-3 Block FC 530 "Path_IdxToAny"



Note By defining StartIndex and EndIndex of the interpolation point table, or start line and end line respectively, it is also possible to store more than one interpolation point table within one data block.