Cams with smooth transitions, based on line segments

Technology CPU

FAQ • September 2009



Service & Support

Answers for industry.

SIEMENS

This entry is from the web sites of Siemens AG, Sector Industry, Industry Automation and Drive Technologies, Service & Support. The terms of use listed on this web site apply (http://www.siemens.com/terms_of_use).

The link below takes you directly to the download page of this document.

http://support.automation.siemens.com/WW/view/en/35690076

Question

How can you create a cam during runtime that is based on line segments (interpolation point table) and that connects these segments by continuous transitions?

Answer

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

Table of Contents

Question **2**

How can you create a cam during runtime that is based on line segments (interpolation point table) and that connects these segments by continuous transitions? 2

Answer **2**

The instructions and notes listed in this document provide a detailed answer to this question. 2

1 Introduction 4
1.1 Procedure 5
1.2 Required hardware and software 6

2 FB 517 “CalcPoly” Function Block 8
2.1 Functionality realized in the block 8
2.2 Integration into a Step7 project 8
2.2.1 Block call 8
2.2.2 “CalcPoly” FB block interface 8
2.2.3 “CalcPoly_Data” DB interface 10

3 Sample Project 11

4 Error Messages of FB 517 “CalcPoly” 13
4.1 Indication of error events 13
4.2 Error codes at the “ErrorID” output 13
4.3 Error codes at the “ErrorSource” output 14

5 References 15
5.1 References 15
5.2 Internet links 15

6 History 16

Copyright © Siemens AG 2008 All rights reserved
CPU_31xT_Sanfte_Kurvenscheiben_e.doc

1 Introduction

Applications often require that cam functions be calculated based on production data. Frequently, it is to be possible to specify the significant cam sectors (e.g., dwell, velocity) as simple line segments in the form of interpolation points. The transitions between the line segments are then mostly to be connected by smooth transitions.

Due to the easy handling of interpolation point tables, it is frequently tried to use this form of cam definition and to subsequently achieve the objective of continuous transitions between the line segments by means of the selectable “Spline” interpolation mode. Since the interpolation affects the entire cam, the desired result can normally not be achieved.

To achieve an easy specification via interpolation points and a curve shape that still has continuous transitions, this FAQ provides the **FB517“CalcPoly”** block that realizes this. At this, the interpolation points are pretend in data blocks.

Figure 1-1, Example of an easy specification of line segments

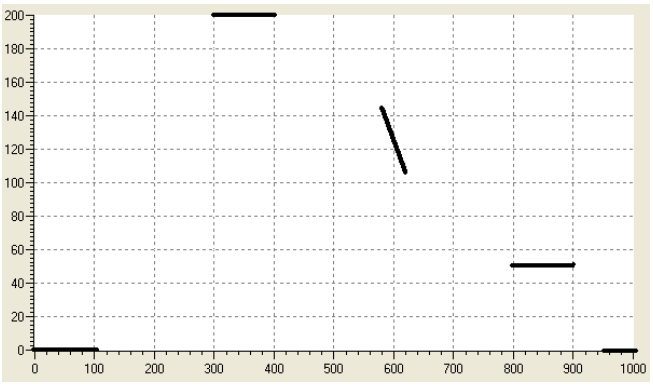


Figure 1-2, Desired interpolated cam

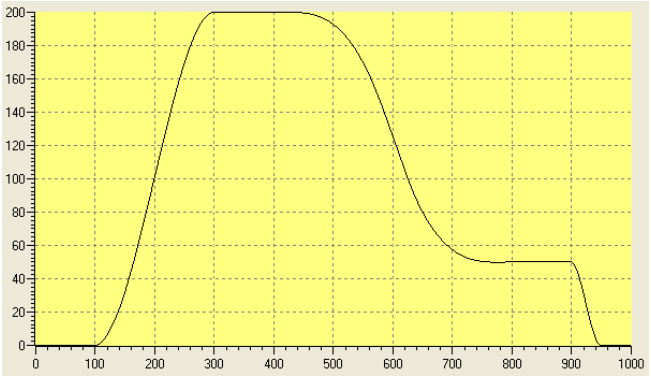
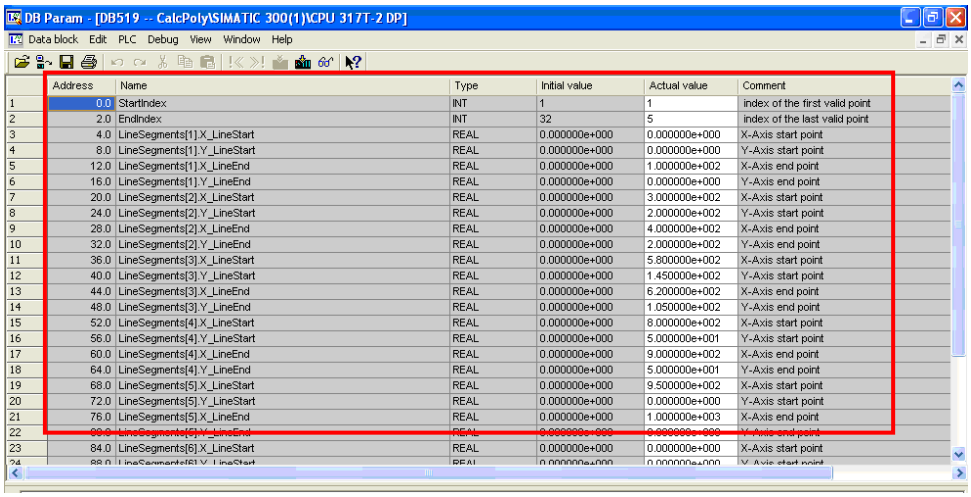


Figure 1-3, Easy specification of the line segments via interpolation points



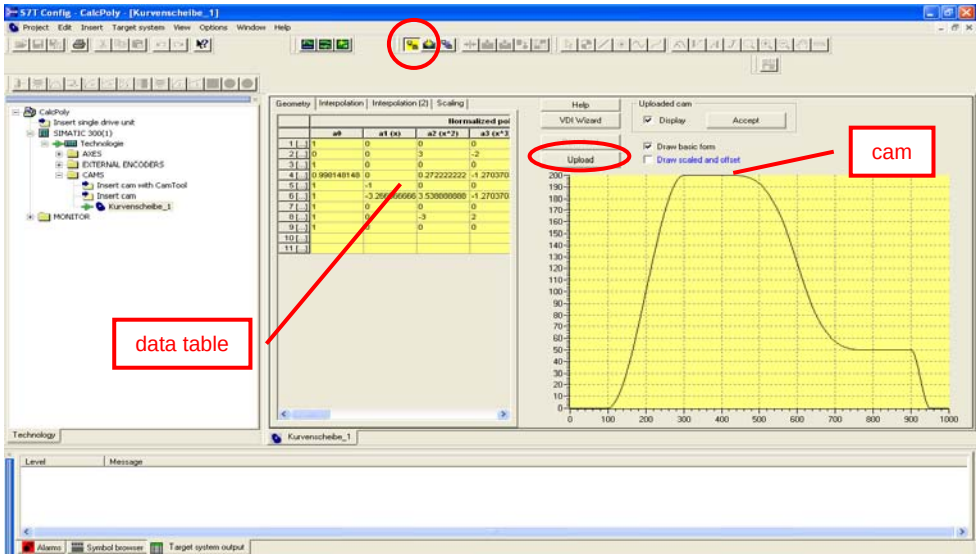
	Address	Name	Type	Initial value	Actual value	Comment
1	0.0	StartIndex	INT	1	1	index of the first valid point
2	2.0	EndIndex	INT	32	5	index of the last valid point
3	4.0	LineSegments[1].X_LineStart	REAL	0.000000e+000	0.000000e+000	X-Axis start point
4	8.0	LineSegments[1].Y_LineStart	REAL	0.000000e+000	0.000000e+000	Y-Axis start point
5	12.0	LineSegments[1].X_LineEnd	REAL	0.000000e+000	1.000000e+002	X-Axis end point
6	16.0	LineSegments[1].Y_LineEnd	REAL	0.000000e+000	0.000000e+000	Y-Axis end point
7	20.0	LineSegments[2].X_LineStart	REAL	0.000000e+000	3.000000e+002	X-Axis start point
8	24.0	LineSegments[2].Y_LineStart	REAL	0.000000e+000	2.000000e+002	Y-Axis start point
9	28.0	LineSegments[2].X_LineEnd	REAL	0.000000e+000	4.000000e+002	X-Axis end point
10	32.0	LineSegments[2].Y_LineEnd	REAL	0.000000e+000	2.000000e+002	Y-Axis end point
11	36.0	LineSegments[3].X_LineStart	REAL	0.000000e+000	5.800000e+002	X-Axis start point
12	40.0	LineSegments[3].Y_LineStart	REAL	0.000000e+000	1.450000e+002	Y-Axis start point
13	44.0	LineSegments[3].X_LineEnd	REAL	0.000000e+000	6.200000e+002	X-Axis end point
14	48.0	LineSegments[3].Y_LineEnd	REAL	0.000000e+000	1.050000e+002	Y-Axis end point
15	52.0	LineSegments[4].X_LineStart	REAL	0.000000e+000	8.000000e+002	X-Axis start point
16	56.0	LineSegments[4].Y_LineStart	REAL	0.000000e+000	5.000000e+001	Y-Axis start point
17	60.0	LineSegments[4].X_LineEnd	REAL	0.000000e+000	9.000000e+002	X-Axis end point
18	64.0	LineSegments[4].Y_LineEnd	REAL	0.000000e+000	5.000000e+001	Y-Axis end point
19	68.0	LineSegments[5].X_LineStart	REAL	0.000000e+000	9.500000e+002	X-Axis start point
20	72.0	LineSegments[5].Y_LineStart	REAL	0.000000e+000	0.000000e+000	Y-Axis start point
21	76.0	LineSegments[5].X_LineEnd	REAL	0.000000e+000	1.000000e+003	X-Axis end point
22						
23	84.0	LineSegments[6].X_LineStart	REAL	0.000000e+000	0.000000e+000	X-Axis start point
24						

1.1 Procedure

The calculation of a cam of the integrated technology of the technology CPU can be realized as follows:

- The cam to be described must be created in the integrated technology of the interpolation point table or polynomial type.
- The sample project with all included blocks has to be downloaded to the CPU.
- The **“CalcPoly_Datax”** data block must be supplied with plausible values.
- The **FB 517 “CalcPoly”** block has to be called in the Step7 program.
- After successful execution (CalcPoly Done = True), the calculated cam can be read out online to check it with SIMATIC Technology.

Figure 1-1, online display of a cam with S7TConfig



Copyright © Siemens AG 2008 All rights reserved
35690076_CPU_31xT_Smooth_cams_DOKU_e.doc

Notes:

- After selecting the „upload“ button, the interpolated cam will readout and display out of the int. technology.
- The data table shows the details of the cam with polynomial coefficients and the domain of definition (X min/max, Y min/max)
- If a several line of the data table is marked, it will be marked in the cam, too.

1.2 Required hardware and software

The representations and descriptions of this FAQ are based on the following hardware and software components:

Hardware components

Table 1-1, Hardware components

Component	No.	order number	Note
CPU 315T-2 DP	1	6ES7315-6TG10-0AB0 or 6ES7315-6TH13-0AB0 Firmware: V2.6	The CPU executes the user program and the technological functions.
Micro Memory Card min. 4MB	1	6ES7953-8LM11-0AA0	The S7 program is stored on the MMC.

Or alternatively:

Table 1-2, Alternative hardware components

Component	No.	order number	Note
CPU 317T-2 DP	1	6ES7317-6TJ10-0AB0 or 6ES7317-6TK13-0AB0 Firmware: V2.6	As an alternative to the CPU 315T-2 DP.
Micro Memory Card min. 4MB	1	6ES7953-8LM11-0AA0	The S7 program is stored on the MMC.

Standard software components

Table 1-3, Standard software components

Component	No.	order number	Note
STEP 7	1	6ES7810-4CC08-0YA7 Version: V5.4 SP2	STEP 7 is the basic package for programming the SIMATIC S7.
S7-Technology	1	6ES7864-1CC41-0YX0 Version: V4.1	Tool for parameterizing and programming the technology objects of the technology CPU.

Sample files and projects

The functionality described in this FAQ was realized in a function block that can be downloaded from the Service & Support Portal together with a sample application.

The following list includes all files and archives used in this FAQ:

Table 1-4, Files and STEP 7 archives of the FAQ

Component	Note
35690076_CPU31xT_smooth_cam_CODE.zip	The STEP 7 archive contains a sample program for the respective technology CPU.
35690076_CPU31xT_smooth_cam_DOKU.pdf	This document.

Required PLC-Open blocks from the “S7-Tech V4.1” library

Copyright © Siemens AG 2008 All rights reserved
35690076_CPU_31xT_Smooth_cams_DOKU_e.doc

The list below contains all PLC-Open blocks from the “S7-Tech V4.1” library used for technology function calls in this technology template. The “S7-Tech V4.1” library is included in the above-listed “S7-Technology” software component.

Table 1-5, Required PLC-Open blocks

PLC-Open blocks	Function
FB 434 “MC_CamClear”	Clearing cams
FB 435 “MC_CamSectorAdd”	Adding cam sectors
FB 436 “MC_CamInterpolate”	Interpolating cams

Table 1-6, Further required blocks of the example

Further blocks	Function
FB 521 “WriteCamData”	Writing cams, used internally by FB 517
DB 518 “CalcPoly_Data1”	Contains the line segments of cam contour 1 as interpolation points
DB 519 “CalcPoly_Data2”	Contains the line segments of cam contour 2 as interpolation points
UDT 517 „LineSegmentsType”	Master to create the data blocks

2 FB 517 “CalcPoly” Function Block

The functionality shown in this FAQ was implemented in **FB 517 “CalcPoly”**. The function and application of this block will be explained in detail in the next chapter.

2.1 Functionality realized in the block

The **FB 517 “CalcPoly”** block reads up to **32 line segments** out of **“DBDataSource”** parameterized during the call. The area to be read can be freely selected via the DB’s **StartIndx** and **EndIdx**.

The block then transfers these segments as polynomials and triggers the interpolation of the segments in the integrated technology. In the integrated technology, the “Close Gap” mechanism is used to achieve continuous transitions between the line segments.

2.2 Integration into a Step7 project

2.2.1 Block call

The block can be integrated into a STEP 7 project by a single call in OB 1 or in a function block.

If several cams are to be calculated simultaneously, separate instance data blocks have to be used for the relevant block calls since the curve points are stored there.

2.2.2 “CalcPoly” FB block interface

To calculate the cams, **FB 517 “CalcPoly”** features the following block interface:

Figure 2-1, Block interface of FB 517

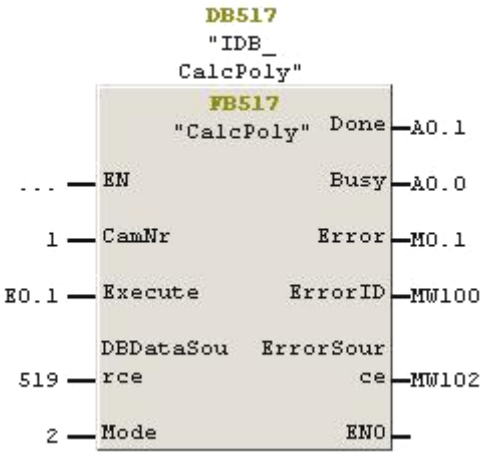


Table 2-1, Block interface

Parameter	Data type	Initial value	Description
Input parameter			
CamNr	INT	0	Number of the cam to be calculated.
Execute	BOOL	False	Start of the block function with rising edge.
DBDataSource	INT	0	Data block in which the segments are stored.
Mode	INT	2	Interpolation mode: Value = 0: Linear interpolation Value = 1: Cubic splines (characteristic through the interpolation points or curve segments) Value = 2: Bezier splines (characteristic along the interpolation points or curve segments)
Output parameter			
Done	BOOL	False	Processing of the block has been successfully completed.
Busy	BOOL	False	Job is being processed.
Error	BOOL	False	An error has occurred while processing the block. Further information on the localization of the error cause is provided via the ErrorID and ErrorSource outputs.
ErrorID	WORD	W#16#0	Error code of the block or of a technology function that has been called internally. In addition, the error location within the block can be read via the ErrorSource output.
ErrorSource	WORD	W#16#0	Specification of an additional error code for localizing the error cause within the block.

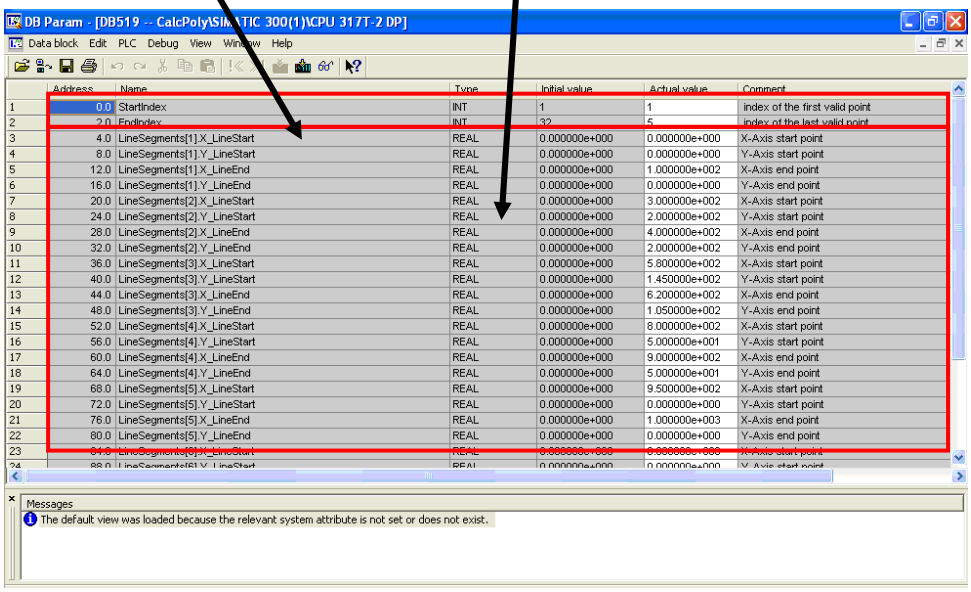
Copyright © Siemens AG 2008 All rights reserved
35690076_CPU_31xT_Smooth_cams_DOKU_e.doc

2.2.3 “CalcPoly_Data” DB interface

Definition of the data area to be transferred
StartIndex (min=1, max=32)
EndIndex (min=1, max=32)

Definition of the line segments

Figure 2-2



If the defined segments are overlapping in the X-axis, the latest definition is used.

3 Sample Project

The sample project is functional and can be directly downloaded to hardware provided for this purpose (chapter 1.2) for testing.

The following sections show the program's principle of operation using the example of the line segments of cam contour 1 (DB 518).

Figure 3-1, Step7 project

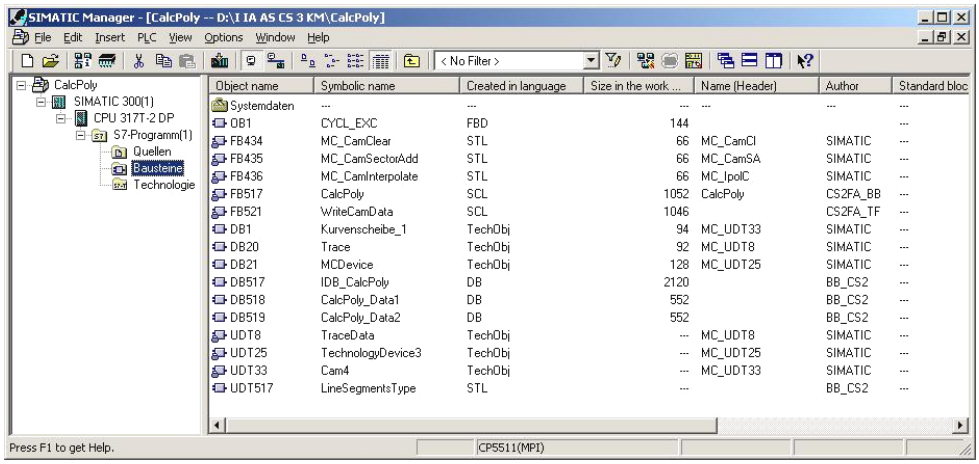
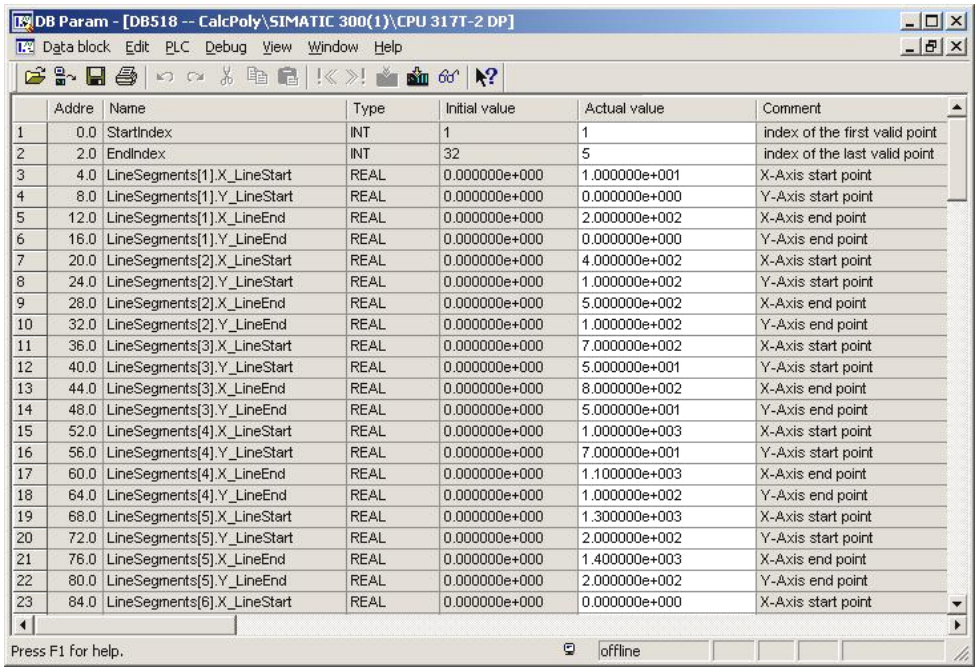


Figure 3-2, Step7 project, DB518 cam definition



	Addre	Name	Type	Initial value	Actual value	Comment
1	0.0	StartIndex	INT	1	1	index of the first valid point
2	2.0	EndIndex	INT	32	5	index of the last valid point
3	4.0	LineSegments[1].X_LineStart	REAL	0.000000e+000	1.000000e+001	X-Axis start point
4	8.0	LineSegments[1].Y_LineStart	REAL	0.000000e+000	0.000000e+000	Y-Axis start point
5	12.0	LineSegments[1].X_LineEnd	REAL	0.000000e+000	2.000000e+002	X-Axis end point
6	16.0	LineSegments[1].Y_LineEnd	REAL	0.000000e+000	0.000000e+000	Y-Axis end point
7	20.0	LineSegments[2].X_LineStart	REAL	0.000000e+000	4.000000e+002	X-Axis start point
8	24.0	LineSegments[2].Y_LineStart	REAL	0.000000e+000	1.000000e+002	Y-Axis start point
9	28.0	LineSegments[2].X_LineEnd	REAL	0.000000e+000	5.000000e+002	X-Axis end point
10	32.0	LineSegments[2].Y_LineEnd	REAL	0.000000e+000	1.000000e+002	Y-Axis end point
11	36.0	LineSegments[3].X_LineStart	REAL	0.000000e+000	7.000000e+002	X-Axis start point
12	40.0	LineSegments[3].Y_LineStart	REAL	0.000000e+000	5.000000e+001	Y-Axis start point
13	44.0	LineSegments[3].X_LineEnd	REAL	0.000000e+000	8.000000e+002	X-Axis end point
14	48.0	LineSegments[3].Y_LineEnd	REAL	0.000000e+000	5.000000e+001	Y-Axis end point
15	52.0	LineSegments[4].X_LineStart	REAL	0.000000e+000	1.000000e+003	X-Axis start point
16	56.0	LineSegments[4].Y_LineStart	REAL	0.000000e+000	7.000000e+001	Y-Axis start point
17	60.0	LineSegments[4].X_LineEnd	REAL	0.000000e+000	1.100000e+003	X-Axis end point
18	64.0	LineSegments[4].Y_LineEnd	REAL	0.000000e+000	1.000000e+002	Y-Axis end point
19	68.0	LineSegments[5].X_LineStart	REAL	0.000000e+000	1.300000e+003	X-Axis start point
20	72.0	LineSegments[5].Y_LineStart	REAL	0.000000e+000	2.000000e+002	Y-Axis start point
21	76.0	LineSegments[5].X_LineEnd	REAL	0.000000e+000	1.400000e+003	X-Axis end point
22	80.0	LineSegments[5].Y_LineEnd	REAL	0.000000e+000	2.000000e+002	Y-Axis end point
23	84.0	LineSegments[6].X_LineStart	REAL	0.000000e+000	0.000000e+000	X-Axis start point

Copyright © Siemens AG 2008 All rights reserved
35690076_CPU_31xT_Smooth_cams_DOKU_e.doc

Figure 3-3, Step7 project, OB1 call of FB517 CalcPoly

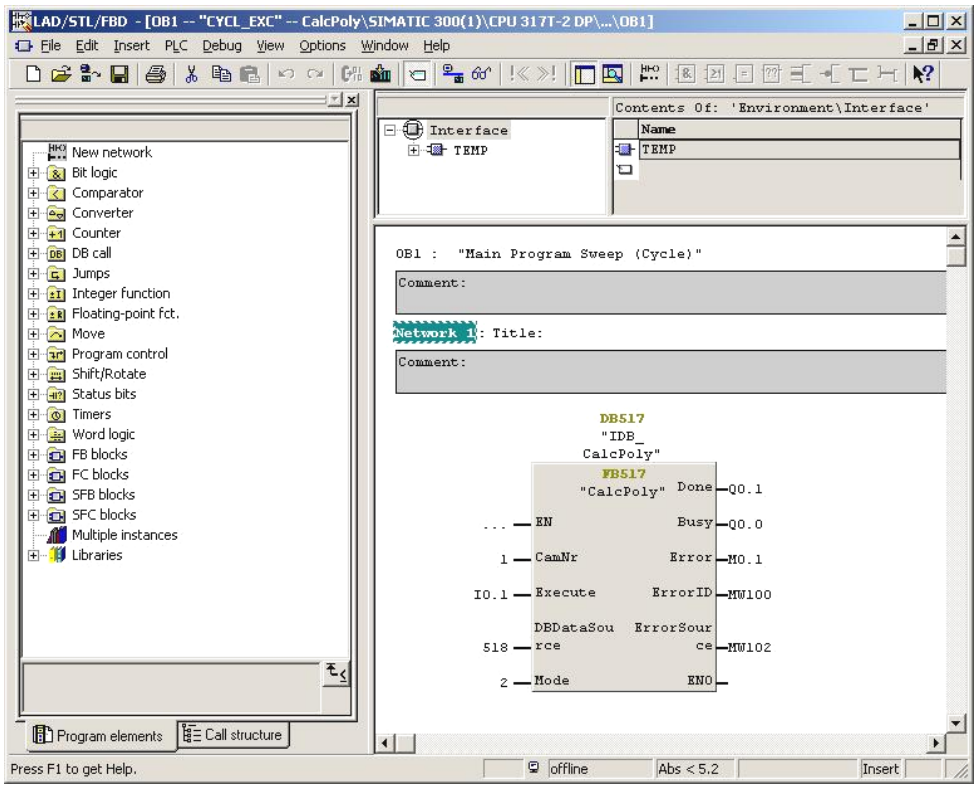
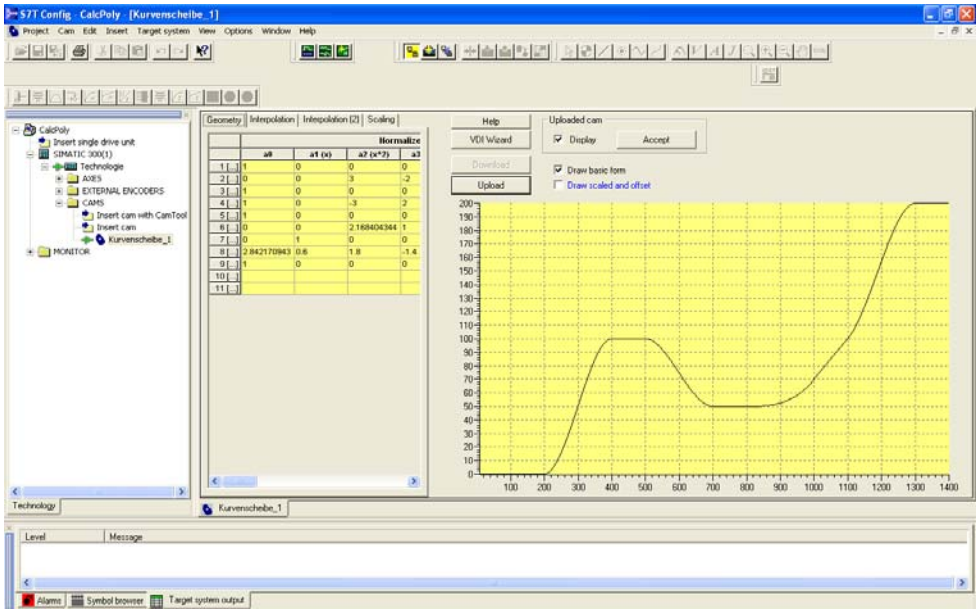


Figure 3-4, S7T Config calculated online cam



4 Error Messages of FB 517 “CalcPoly”

4.1 Indication of error events

If errors occur at the **FB 517 “CalcPoly”** block, they are displayed at the block interface as follows:

- **“Error”** output:
  This output is set in the event of an error. The error cause can be read at the **“ErrorID”** and **“ErrorSource”** outputs.
- **“ErrorID”** output:
  Output of the error code associated with the error event or of a warning code.
- **“ErrorSource”** output:
  Location of the error code output at the **“ErrorID”** output.
-
-

4.2 Error codes at the “ErrorID” output

Copyright © Siemens AG 2008 All rights reserved
35690076_CPU_31xT_Smooth_cams_DOKU_e.doc

Figure 4-1, Error codes at the “ErrorID” output

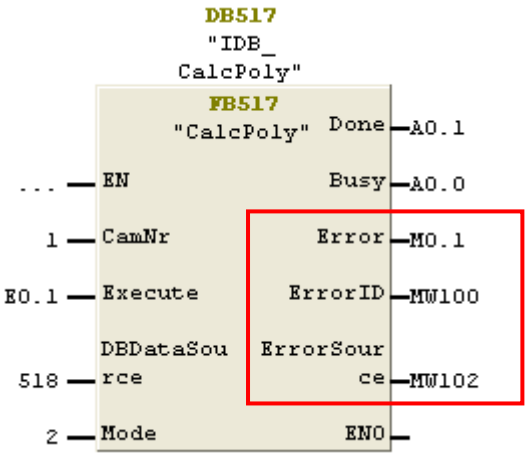


Table 4-1, Error codes at the “ErrorID” output

ErrorID [HEX]	Meaning	Note
0000	No error.	
8xxx	Error of a technology function block called in FB 517 “CalcPoly”. The error cause can be determined via the error description for the respective technology function block.	The affected technology function block can be determined via the error code at the **ErrorSource** output.
9000	StardIndx or EndIndx of DBDatasource invalid.	Is shown too, when a wrong DB(type) is named at DBDatasource

4.3 Error codes at the “ErrorSource” output

•

Table 4-2, Error codes at the “ErrorSource” output

Error Source [HEX]	Meaning	Note
000	No error	
001	Error of transfer the line segments to the internal technology	See WriteCamData error description
002	Error of delete or set back the cam	See MC_CamClear error description
003	Error of adding of a cam segment	See MC_CamSectorAdd error description
004	Error of the interpolation of the cam	See MC_CamInterpolate error description

5 References

5.1 References

This list is by no means complete and only provides a selection of appropriate sources.

Table 5-1, References

	Topic	Title
/1/	STEP 7	Automating with STEP7 in STL and SCL Hans Berger Publicis MCD Verlag – 4th edition, 2004 ISBN 3-89578-242-4
/2/	STEP 7	SIMATIC – S7-SCL V5.3 for S7-300/400 Siemens manual Edition 02/2004 MLFB: A5E00290608-01
/3/	Technology CPU	SIMATIC – S7 Technology Siemens manual Edition 03/2008 MLFB: A5E00251797-06

Copyright © Siemens AG 2008 All rights reserved
35690076_CPU_31xT_Smooth_cams_DOKU_e.doc

5.2 Internet links

This list is by no means complete and only provides a selection of appropriate sources.

Table 5-2, Internet links

	Topic	Title
\1\	Reference to the entry	http://support.automation.siemens.com/WW/view/en/26680228
\2\	Industrial Automation and Drive Technology “Service & Support”	Service & Support Portal: www.automation.siemens.de/support
\3\	Technology CPU manual	S7-Technology: http://support.automation.siemens.com/WW/view/en/30119663
\4\	FAQ	How do you configure bit-triggered trends (curves) in WinCC flexible? http://support.automation.siemens.com/WW/view/en/21913875

6 History

Table 6-1, History

Version	Date	Modification
V1.0	09/2009	First edition