Display of a Cam Disk of the Technology CPU with WinCC flexible in HMI

Technology CPU

FAQ • Januar 2009

Service & Support

Answers for industry.

SIEMENS

Clicking the link below directly displays the download page of this document.

http://support.automation.siemens.com/WW/view/de/26680228

Question

How can I display a cam disk of the technology CPU on the HMI using WinCC flexible?

Answer

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

Table of content

1 Introduction

This FAQ should show you how to read out a cam disk from the integrated technology of the technology CPU and displaying it using WinCC flexible on the HMI.

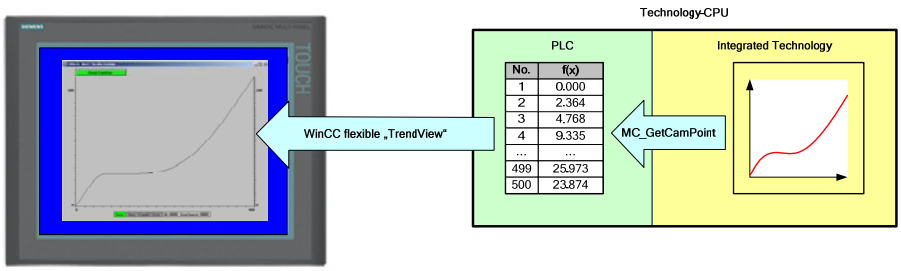
1.1 Procedure

Reading out and displaying a cam disk of the integrated technology of the technology CPU can be realized as follows:

- Reading out cam points of the cam disk via the technology function block **FB 438 "MC_GetCamPoint"** from the integrated technology.
- Saving the cam points in a data block, or a **data array** in the PLC section of the technology CPU.
- Display of the cam points via the **"Trend view"** component of WinCC flexible in HMI.

This procedure has been summarized in function block **FB 539 "GetCamDisc"** and can be easily integrated into your own application.

Figure 1-1 Display of a cam disk on the HMI



Note

The point-by-point reading of the points of a cam disk via the **FB 438 "MC_GetCamPoint"** causes a high load of data exchange between integrated technology and PLC of the technology CPU. The cam disk should therefore only be displayed if no other technology functions are active in the technology CPU or sufficient resources are available.

1.2 Required hardware and software

The representation and description of this FAQ builds on the following hardware and software components:

Hardware components

Table 1-1 Hardware components

Component	Quantity	MLFB / order number	Note
CPU 315T-2 DP	1	6ES7315-6TG10-0AB0 oder 6ES7315-6TH13-0AB0 Firmware: V2.6	The CPU executes the application program and the technological functions.
Micro Memory Card 4MB	1	6ES7953-8LM11-0AA0	The S7 program is stored on the MMC.

or alternatively:

Table 1 -2 Hardware components – Alternative 1

Component	Quantity	MLFB / order number	Note
CPU 317T-2 DP	1	6ES7317-6TJ10-0AB0 oder 6ES7317-6TK13-0AB0 Firmware: V2.6	As an alternative to CPU 315T-2 DP
Micro Memory Card 4MB	1	6ES7953-8LM11-0AA0	The S7 program is stored on the MMC.

Standard software components

Table 1-3 Standard software components

Component	Quantity	MLFB / order number	Note
STEP 7	1	6ES7810-4CC08-0YA7 Version: V5.4	STEP 7 is the basic package for programming the SIMATIC S7.
STEP 7 – SCL	1	6ES7811-1CC05-0YA5 Version: V5.3	Program package for generating and compiling high-level language programs in S7-SCL.
S7 Technology	1	6ES7864-1CC41-0YX0 Version: V4.1	Tool for parameterization and programming of technology objects of the technology CPU
WinCC flexible 2008	1	6AV6613-0AA51-3CA5 Version: 2008 Advanced	Tool for the configuration of HMI user interfaces, including WinCC flexible Runtime for PC.

Note The software component **STEP 7-SCL** is only necessary if you wish to make changes to function block **FB 539 "GetCamDisc"** and need to recompile the block afterwards.

The block itself can also be used unchanged even without the software component **STEP 7-SCL** in the example application or in some applications.

Example files and projects

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

The following list contains all files and archives used in this FAQ.

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

Component	Hinweis
26680228_CPU315T_HMI-CamDisc_CODE_v10.zip	The STEP 7 archive contains an example program for the respective technology CPU including WinCC flexible 2008 PC Runtime.
26680228_CPU317T_HMI-CamDisc_CODE_v10.zip	
26680228_HMI-CamDisc_DOKU_v10_e.pdf	This document

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

The list below includes 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 "S7 Technology" software component listed above.

Table 1-5 Required PLC-Open blocks

PLC-Open blocks	Function
FB 406 "MC_ReadSysParameter"	Reading system files from the integrated technology of the technology CPU.
FB 438 "MC_GetCamPoint"	Reading individual cam points from the integrated technology of the technology CPU.

2 Function block FB 539 “GetCamDisc”

The functionality of the trend view in the HMI, represented in this FAQ, was realized in **FB 539 “GetCamDisc”**. The function and application of this block shall be explained in detail in the following chapter.

2.1 Functionality realized in the block

Block **FB 539 “GetCamDisc”** takes on entirely the reading out of the cam disk from the integrated technology of the technology CPU and the display of the cam points in HMI.

The block reads out **500 cam points** from a cam disk and enters it into the **Data[1..500]** array in the instance block of the **FB 539 “GetCamDisc”**.

2.2 Integration into a Step7 project

2.2.1 Block call

The block call can be integrated into a STEP 7 project by means of a simple call in OB1 or in a function block.

If several cam disks shall be represented at once, separate instance data blocks must be used for the respective block calls, since the cam points will be stored there.

2.2.2 Block interface

For reading and display of the cam disk data the **FB 539 “GetCamDisc”** has the following block interface:

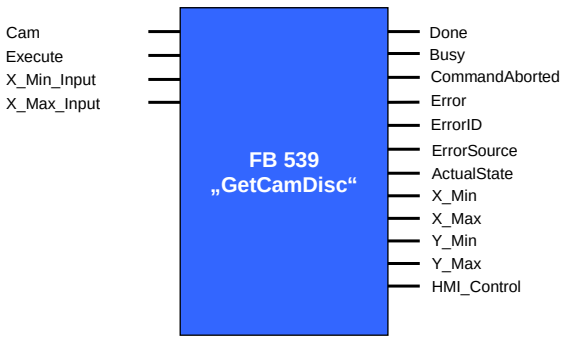


Table 2-1 Block interface

Parameter	Data type	Initial value	Description
Input parameters			
Cam	INT	0	Number of the cam disk which shall be displayed.
Execute	BOOL	False	Start of the block function with rising edge.
X_Min_Input	REAL	0.0	Minimum of the displayed area on the master axis.

Parameter	Data type	Initial value	Description
			If the same value is entered for **X_Min_Input** and for **X_Max_Input**, the block determines the definition area of the cam disk on the master axis.
X_Max_Input	REAL	0.0	Minimum of the displayed area on the master axis. If the same value is entered for **X_Min_Input** and for **X_Max_Input**, the block determines the definition area of the cam disk on the master axis.
Output parameters			
Done	BOOL	False	The block was processed successfully. The data of the cam disk are available for display by the HMI.
Busy	BOOL	False	The block is busy processing.
CommandAborted	BOOL	False	Processing of the block was terminated by an other technology function block.
Error	BOOL	False	An error has occurred during processing of the block. Further information on the localization of the cause of error is made available via the ErrorID and ErrorSource outputs.
ErrorID	WORD	W#16#0	Error code of the block or of an internally called technology function. In addition, it is possible to locate the error within the block via the ErrorSource output.
ErrorSource	WORD	W#16#0	Specification of an additional error code for the localization of the cause of error within the block.
ActualState	INT	0	Currently active state of the block.
X_Min	REAL	0.0	Minimum of the position value of the master axis (X-axis) for the display in HMI.
X_Max	REAL	0.0	Maximum of the position values of the master axis (X-axis) for the display in HMI.
Y_Min	REAL	0.0	Minimum of the position value

Parameter	Data type	Initial value	Description
			of the slave axis (Y-axis) for the display in HMI.
Y_Max	REAL	0.0	Maximum of the position value of the slave axis (Y-axis) for the display in HMI.
HMI_Control	WORD	W#16#0	Controlling the output of the cam disk in HMI by means of the block.

Additional interface in the “Statistical Variables” area

For the complete integration of the HMI into the **FB 539 “GetCamDisc”** the following parameter is available in the statistical variables of the block:

Table 2-2 Block interface (statistical variables)

Parameter	Data type	Initial value	Description
HMI_Response	WORD	W#16#0	Feedback of the HMI that the image with the output of the cam disk was displayed on the HMI.

This variable can be connected with the **“Trend request”** parameter of WinCC flexible. If the image for the cam disk representation is displayed in WinCC flexible Runtime, the respective bit is set to “1”.

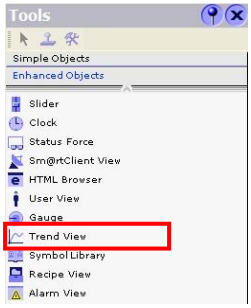
Within the **FB 539 “GetCamDisc”** the variable has no function, however, it can be used for an evaluation if the **Execute** input of the block is to be automatically set to “True”, as soon as the image for the cam disk display is called up in the HMI.

Note More detailed information on the **“Trend request”** parameter of WinCC flexible is contained in the following chapter.

3 Display of the Cam Disk in HMI

For the display of the cam disk in HMI the cam points stored in block **FB 539 “GetCamDisc”** are read in at the trend view of WinCC flexible and displayed there as a curve.

3.1.1 Integrating the WinCC flexible component



From the **Tools** component catalog of WinCC flexible, from the **Enhanced Objects** section, you draw the **Trend view** component onto the desired image of your HMI project.

When defining the size of the **Trend View** component check the size of the data array in the PLC. Do not read out more cam points from the cam disk than available to you as pixels for the representation of the cam disk in the HMI, in order to prevent unnecessary system load.

3.1.2 Configuration of the WinCC flexible component

Now set the properties of the **Trend View** component in order to be able to display the cam disk on the HMI.

Properties X-axis

The beginning and end of the X-axis as well as the resolution of the cam disk display are configured in this dialog.

Figure 3-1 Properties X-axis

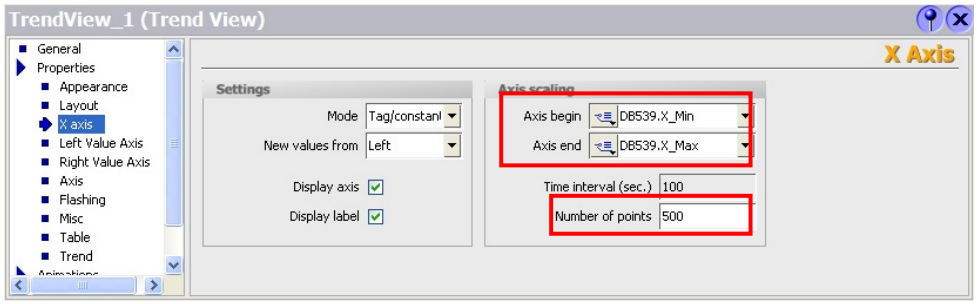


Table 3-1 Parameter settings

Parameter	Comment
Axis begin	Minimum value of the definition rage for the cam disk regarding the master axis (X_Min).
Axis end	Maximum value of the definition rage for the cam disk regarding the master axis (X_Max).
Number of points	Resolution of the display of the cam disk. This value must correspond with the length of the data array in the PLC.

Properties value axis left and right

In these dialogs the Y-axes of the left and right display are formatted for the representation of the slave axis.

Figure 3-2 Properties left value axis

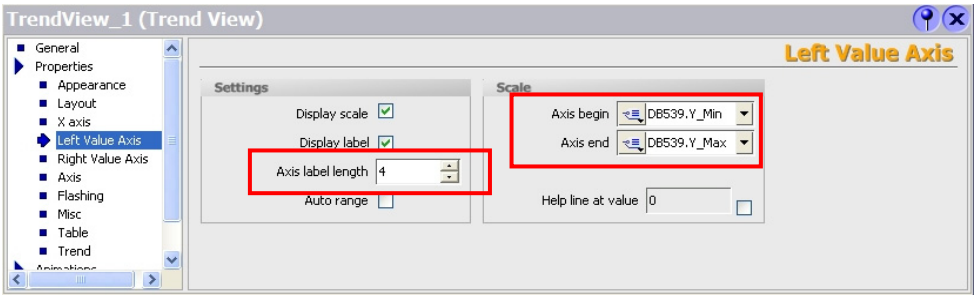


Figure 3-3 Properties right value axis

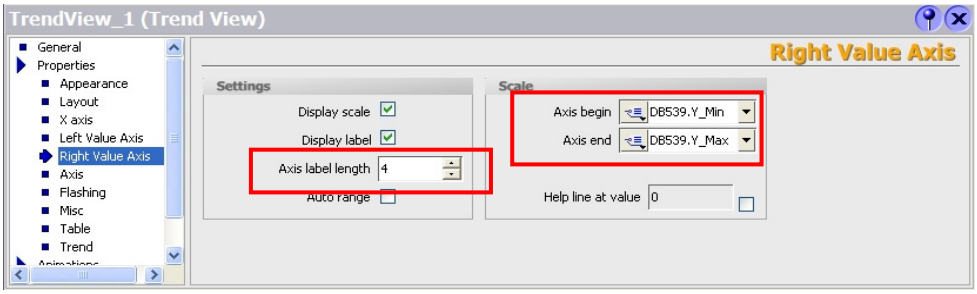


Table 3-2 Parameter settings

Parameter	Comment
Axis begin	Minimum value of the possible slave axis position (Y_Min). Please ensure that the minimum value of the slave axis position can also occur within the cam disk!
Axis end	Maximum value of the possible slave axis position (Y_Max). Please ensure that the maximum value of the slave axis position can also occur within the cam disk!
Axis label length	Permitted number of digits for the value label on the Y-axis.

Axis properties

In this dialog the axis labels and the increments of the X and Y-axes are adjusted.

Figure 3-4 Axis properties

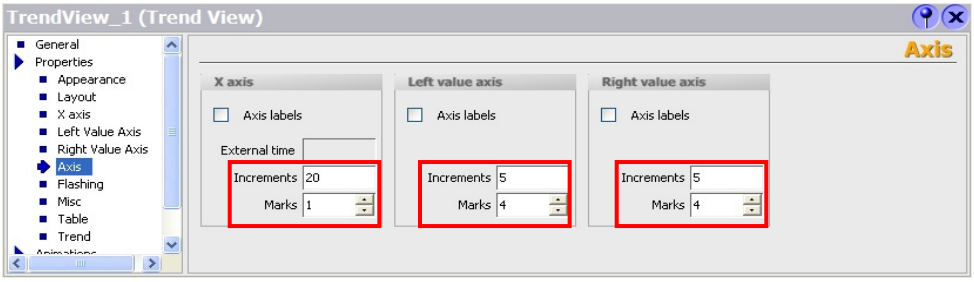


Table 3-3 Parameter settings

Parameter	Comment
Increments	Increments between the marks on the respective axis.
Marks	Number of small marks with a distance of the increment between two large marks.

Trend properties

In this dialog the actual connection is made between the display and the cam disk data.

Figure 3-5 Curve properties

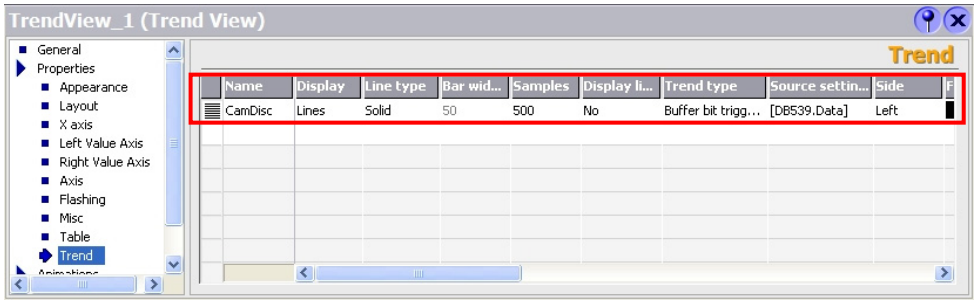


Table 3-4 Parameter settings

Parameter	Comment
Name	Name of the curve to be represented.
Display	**Points:** The cam points are represented individually as point. **Lines:** The individual cam points are connected with each other and so represented as an interconnected curve.
Samples	Defining the cam points available in the data array which are transferred to the HMI from the data array.
Trend type	**Buffer bit triggered:** The cam points are filed in a data storage of the PLC and can be called up from there on demand by changing a bit and be displayed.
Source settings	Here the connection of the display with the curve data is defined in the PLC. The respective settings are explained in greater detail in the subsequent chapter.

Note

For the trend properties please ensure setting the value in **Samples**. The value entered here must correspond with the Number of points in the properties of the X-axis and the dimension of the array for the data storage.

3.1.3 Connecting the cam disk data

Via the input field of the **Source settings** in **Properties Trend** the connection between the display and the cam disk data is realized.

Figure 3-6 Source settings

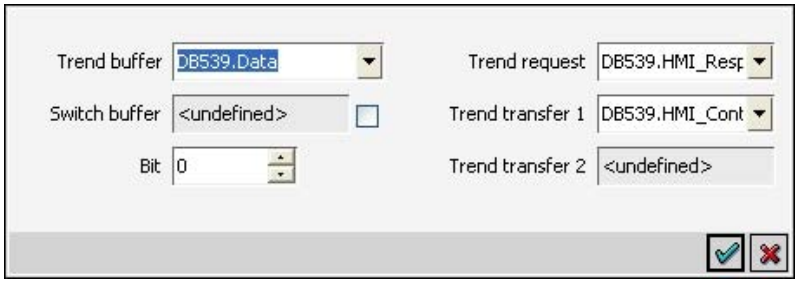


Table 3-5 Parameter settings

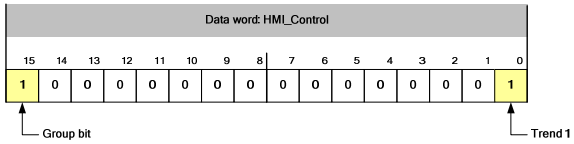
Parameter	Comment
Trend buffer	Connecting the data array in the PLC from which the cam points are read our for the display
Trend request	Via this data area (WORD) the HMI reports to the PLC the trend to be displayed. As soon as the image of the trend view has been called up in the HMI, the respective bit is set in this data area. The PLC can now provide the required data (e.g. for setting the **Execute** input at **FB 539 "GetCamDisc"**)
Trend transfer 1	Via this data area (WORD) the PLC (bit triggering) orders the HMI to transfer the cam points and to display the curve in the trend view.
Bit	Defining the bit which is used for this curve for bit triggering and feedback of the display to the PLC in the parameters **Trend request** and **Trend transfer 1**. When using **FB 539 "GetCamDisc"** value 0 must always be entered here.

3.1.4 Performing the cam disk display

In order to be able to display the cam points in the HMI, the **bit triggering** for this curve must be performed.

The PLC must set the trend bit (e.g. bit 0) and the group bit (bit 15) for **trend transfer 1**.

Figure 3-7 Data word for the curve transfer 1



This process is realized in **FB 539 "GetCamDisc"** by setting the bits displayed in the image in the **HMI_Control** output of the block.

Note For further information on bit-triggered **trend transfer** an **FAQ** is available in the i**nternet link display**

3.2 Connecting the FB 539 "GetCamDisc" to the HMI

To display a cam disk using the function block **FB 539 "GetCamDisc"** the parameters of the HMI can be directly connected with the output parameters of the function block.

The block then takes on all required control signals for the display of the read out curve data in the HMI. The block uses the control bit 0 for the trend transfer to the HMI in the **HMI_Control** parameter.

Connect the following block outputs or block parameters with the respective parameters of the trend view in WinCC flexible:

Table 3-6 Connecting the FB 539 "GetCamDisc" to the HMI

Parameter	Comment
X_Min	Properties X-axis • Axis begin
X_Max	Properties X-axis • Axis end
Y_Min	Properties left value axis • Axis begin Properties right value axis • Axis begin

Parameter	Comment
Y_Max	Properties left value axis • Axis end Properties right value axis • Axis end
HMI_Control	Trend properties – Source settings • Trend transfer 1
HMI_Response (Static variable in the instance block)	Trend properties – Source settings • Trend request

Note

For the connection to the HMI also check the following settings which must correspond to the length of the data array:

- Properties X-axis – Number of points
- Trend properties – Samples

3.3 Additional information on the FB 539 “GetCamDisc”

3.3.1 Reducing the cam points

If a different number than **500 cam points** are required for the display, this number can be adjusted by changing the block. Perform the following changes:

Figure 3-8 Adjusting the number of cam points

```
// ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
// + FunctionBlock Parameters                                                    +
// ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

CONST

   Steps                                := 500;
   ShowGraph1                           := W#16#8001;
   ShowNoGraph                          := W#16#0000;

END_CONST
```

In the SCL source of the block you adjust the **Steps** constant to the desired array size (ArrayLength). When compiling the block the data array is adjusted to the set size:

Data : ARRAY[1..Steps] OF REAL

Note

Do not forget to adjust the required parameters in WinCC flexible to the changed array size (ArrayLength).

3.3.2 Realizing a zoom function

The block can also be used to realize a zoom function within a cam disk.

For the zoom function you simply adjust the entries **X_Min_Input** and **X_Max_Input** of the block to the desired master axis section which is to be zoomed. This curve section is then zoomed over the entire width of the “Trend View” in the HMI.

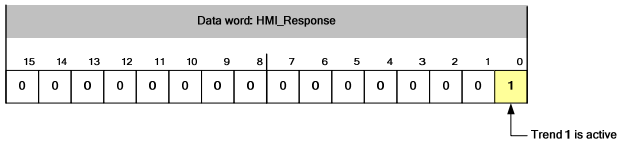
When using this function please note that the values of the **X_Min_Input** and **X_Max_Input** parameters lie within the definition area of the cam disk, otherwise an error message is output by the block.

3.3.3 Automatic triggering of the trend view

If the cam disk display is to be triggered as soon as the image of the trend view is called in the HMI, the static variable **HMI_Response** of the **FB 539 “GetCamDisc”** can be used. This variable is connected with the **Trend request** parameter of the HMI.

As soon as the trend view is called, the HMI requests the display of the desired trend by setting the appropriate trend bit (e.g. bit 0) in the data area **Trend request**.

Figure 3-9 Trend request by the HMI



The automatic triggering of the trend view can be performed by assigning the appropriate bit of the **HMI_Response** variable from the instance data block to the **Execute** input of the block. As soon as the image of the trend view is called up in the HMI, the **Execute** input is set to “True” and the representation of the cam disk in HMI is started.

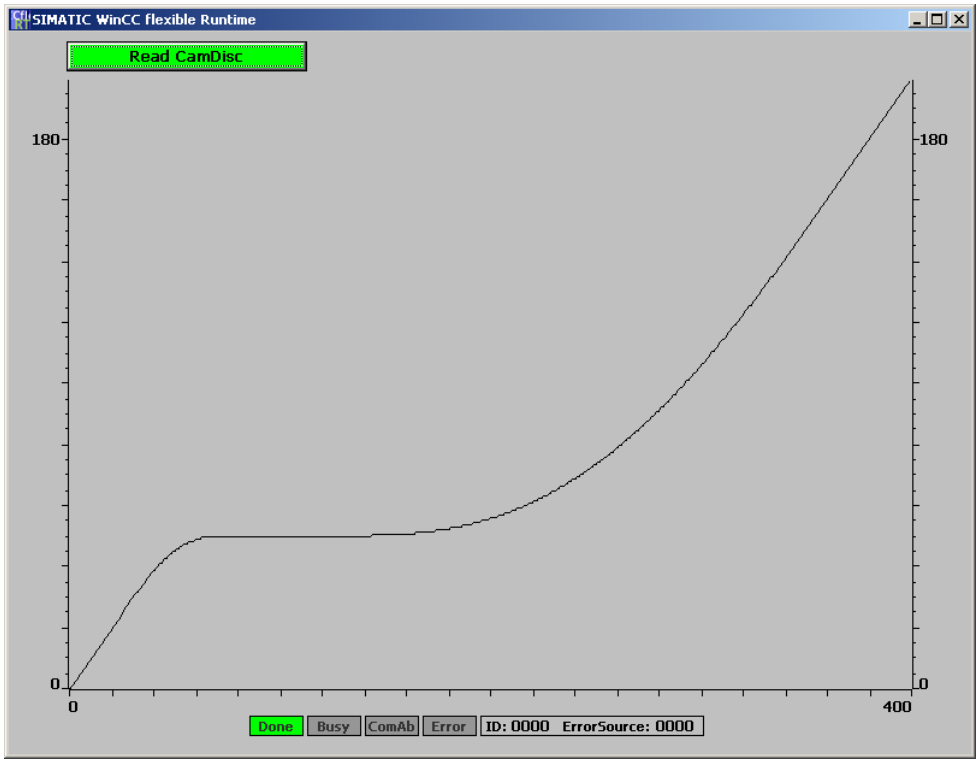
Figure 3-10 Automatic triggering of the trend view

```
OB1 :   "Main Program Sweep (Cycle)"
Comment:

Network 1: Read CamDisc
Comment:

      U     DB539.DBX   39.0
      =     "idb_GetCamDisc".Execute

      CALL  "GetCamDisc" , "idb_GetCamDisc"
       Cam            :=1
       Execute        :=
       X Min Input    :=
```

3.4 Display in WinCC flexible Runtime

As an example for the display of a cam disk, a WinCC flexible project is included in the Step7 archive as PC Runtime.

Figure 3-11 PC Runtime of the WinCC flexible project for the FB 539 “GetCamDisc”



Via the **“Read CamDisc“** button the reading and displaying of a cam disk (DB 1), which is saved permanently in the integrated technology, can be triggered.

The status display at the bottom edge of the trend view shows selected output signals of the FB 539 “GetCamDisc”.

4 Functionality used in the FB 539 “GetCamDisc”

In this chapter all functions used in **FB 539 “GetCamDisc”** shall be explained in greater detail in order to familiarize you with the internal function of the block.

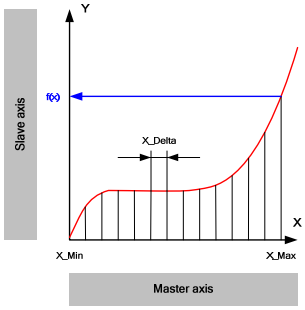
4.1 Read out the cam points

Since the cam disk data of the integrated technology cannot be displayed directly from the HMI, the cam disk must be read out point-by-point.

4.1.1 Preparations

In order to read out the cam points the definition area of the cam disk is divided into equal parts on the master axis. For each increment the respective slave axis value is determined from the cam disk.

Figure 4-1 Reading out the points of a cam disk



The following data of the cam disk must be known for reading out the cam points:

Table 4-1 Information required for the cam disk

Parameter	Comment
X_Min	Minimum value of the definition rage of the cam disk regarding the master axis. This value is available in the technology parameter **4203 “Leadingrange.Start”** using the technology function FB 406 “ReadSysParameter”.
X_Max	Minimum value of the definition rage of the cam disk regarding the master axis. This value is available in the technology parameter **4204 “Leadingrange.End”** using the technology function FB 406 “ReadSysParameter”.

The distance **X_Delta** of the cam points on the master axis for which the slave axis position shall be determined depends directly on the number of cam points to be saved, hence on the size of the data array (**ArrayLength**) which is used for saving the cam points.

$$X\_Delta = \frac{X\_Max - X\_Min}{ArrayLength - 1}$$

Note In order to reduce the technology function calls and the required memory in the PLC adjust the number of cam points to be determined to the resolution of the WinCC flexible component.

4.1.2 Reading out the cam points via FB 438 “MC_GetCamPoint”

The cam points are read out by repeatedly calling the **FB 438 “MC_GetCamPoint”**, which is used to determine the appropriate slave axis position via a given master axis position.

Configuring the FB 438 “MC_GetCamPoint”

For reading out the cam points the FB 438 “MC_GetCamPoint” is configured as follows, whereby the parameters not listed here are not relevant for this application case.

Figure 4-2 Technology function block FB 438 “MC_GetCamPoint”

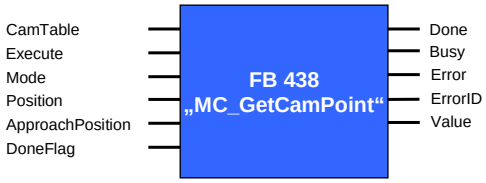


Table 4-2

Parameter	Note
Input variables	
CamTable	Here you enter the number of the data block of the desired cam disk from which the value is to be read out.
Mode	Via **Mode=0** you define that the value at the **Position** parameter is interpreted as master axis position to which the position of the slave axis is to be determined.
Position	Specifying the master axis position for which the appropriate slave axis position is to be determined using the cam disk.
Output variables	
Value	Output of the desired slave axis position which can be determined via the cam disk at the given master axis position.

Specifying the master axis positions for determining the cam points

In order to read out the cam disk completely, the cam point determination is started with the minimum value of the definition area of the cam disk with regards to the master axis.

$$x_1 = X\_Min$$

For all further points the master axis position to be specified is respectively increased by the determined distance **X_Delta** to the previous value. In order to keep the error reproduction for calculating the given master axis position as low as possible, the master axis position is calculated for the current step according to the following formula:

$$x_n = X\_Min + ((n-1) \cdot X\_Delta) \quad \text{with} \quad 2 \le n \le ArrayLength$$

Example:

A cam disk is given with the following parameters:

$X\_Min = 0.000$

$X\_Max = 400.000$

The cam points shall be filed in an array according to the following size:

$ArrayLength = 500$

The resulting distance between the given master axis positions is:

$$X\_Delta = \frac{X\_Max - X\_Min}{ArrayLength - 1} = \frac{400.000 - 0.000}{500 - 1} = 0.8016032$$

The resulting values for selected master axis positions to be defined is:

$$x_1 = X\_Min = 0.000$$

$$x_{200} = X\_Min + ((n-1) \cdot X\_Delta)$$
$$= 0.000 + ((200-1) \cdot 0.8016032 = 159.5190368$$

$$x_{500} = X\_Min + ((n-1) \cdot X\_Delta)$$
$$= 0.000 + ((500-1) \cdot 0.8016032 = 399.9999968$$

For n=500 the last readable value of the cam disk within the definition range has been reached.

4.2 Saving the cam points in a data array:

In order to be able to display the cam points in the HMI, they must be saved in a data area in the PLC. A data array lends itself to it.

4.2.1 Creating the required memory area

The represented resolution of the cam disk in the HMI depends on the size of the created memory area, hence the array size (ArrayLength).

A single-column data array of the required size is created at which the **array index runs from 1 to ArrayLength** and in which the slave axis positions are stored to the respective value on the master axis.

Note In order to prevent unnecessary system loads, less cam points should be read out from the cam disks than pixel available for the representation of the cam disk in the HMI.

4.2.2 Saving the cam points

The display of the cam disk in the HMI occurs from the minimum of the definition area of the cam disk, ascending to the maximum of the definition area of the cam disk. For this reason the cam points must also be read out from the cam disk in this order and be saved in the data array.

Example: The data array in which the cam points are saved is defined as follows:

Data : ARRAY[1..500] OF REAL

The cam points must be filed in the array as follows:

$$Data[1] = f(X\_Min)$$

...

$$Data[200] = f(x_{200})$$

...

$$Data[500] = f(x_{500}) \approx f(X\_Max)$$

Note: Due to the restricted precision f the data type REAL it may happen that $f(x_{500})$ represents only an approximation of the function value of f(X_Max), as x_{500} may also only correspond to an approximation of X_Max.

5 Error Messages of the FB 539 “GetCamDisc”

5.1 Indication of error conditions

If warnings or errors occur at the **FB 539 “GetCamDisc”** block, they are indicated at the block interface as follows:

- **Error** output:
  This output is set if an error occurs. The cause of error can be read at the **ErrorID** and **ErrorSource** outputs.
- **ErrorID** output:
  Output of the error/warning code allocated to the error condition.
- **ErrorSource** output:
  More detailed specification of the error code indicated at the **ErrorID** output for an easy localization of the cause of error.

5.2 Error codes at the ErrorID output

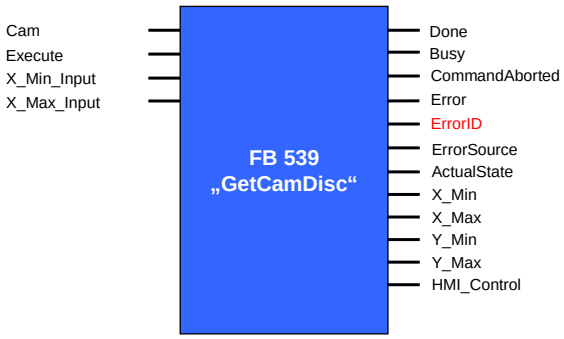


Table 5-1 Error codes at the “ErrorID” output

ErrorID [HEX]	Description	Note
0000	No error	
8xxx	Error of a technology function block called in FB 539 “GetCamDisc”. The error cause can be determined via the error description to the respective technology function block.	The affected technology function block can be determined via the error code at the **ErrorSource** output.
91FF	Internal error of FB 539 “GetCamDisc”	The exact technology function block can be determined via the error code at the **ErrorSource** output.

5.3 Error codes at the ErrorSource output

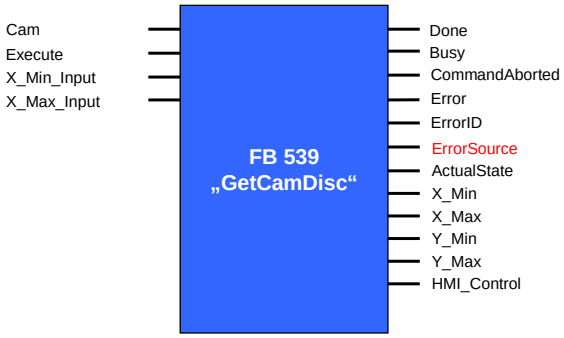


Table 5-2 Error codes at the “ErrorSource” output

Error Source [HEX]	Description	Note
0000	No error	
F001	DB Number of CamDisc not set The number of the technology DB of the cam disk was not defined at the **Cam** input of the block.	
F002	Getting Start of LeadingRange of CamDisc not possible The start value of the definition area of the cam disk on the master axis could not be determined with the block.	FB 406 „MC_ReadSysParameter“
F003	Getting End of LeadingRange of CamDisc not possible The end value of the definition area of the cam disk on the master axis could not be determined with the block.	FB 406 ”MC_ReadSysParameter“
F004	Value X_Min_Input is set too small The value defined at the input X_Min_Input is smaller than the start value of the definition area of the cam disk on the master axis.	
F005	Value X_Max_Input is set too big The value defined at the input X_Max_Input is larger than the end value of the definition area of the cam disk on the master axis.	
F006	Getting CamPoint not possible A cam point could not be determined.	FB 438 “MC_GetCamPoint”

6 Bibliography

6.1 Bibliographic References

This list is not complete and only represents a selection of relevant literature.

Table 6-1 Literature

	Topic	Title
/1/	STEP 7	Automatisieren mit STEP7 in AWL und SCL (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 Issued 03/2008 MLFB: A5E00251797-06

6.2 Internet Links

This list is not complete and only represents a selection of relevant literature.

Table 6-2 Internet links

	Topic	Title
\1\	Reference to this entry	http://support.automation.siemens.com/WW/view/en/26680228
\2\	Industrial Automation and Drive Technologies “Service & Support“	Service & Support Portal: http://www.automation.siemens.com/support
\3\	Handbuch Technologie-CPU	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