Creation of cam disks at runtime for S7-1500T

Library LCamHdl – Basic cam creation

https://support.industry.siemens.com/cs/ww/en/view/105644659

Siemens Industry Online Support

Legal information

Use of application examples

Application examples illustrate the solution of automation tasks through an interaction of several components in the form of text, graphics and/or software modules. The application examples are a free service by Siemens AG and/or a subsidiary of Siemens AG ("Siemens"). They are non-binding and make no claim to completeness or functionality regarding configuration and equipment. The application examples merely offer help with typical tasks; they do not constitute customer-specific solutions. You yourself are responsible for the proper and safe operation of the products in accordance with applicable regulations and must also check the function of the respective application example and customize it for your system.
Siemens grants you the non-exclusive, non-sublicensable and non-transferable right to have the application examples used by technically trained personnel. Any change to the application examples is your responsibility. Sharing the application examples with third parties or copying the application examples or excerpts thereof is permitted only in combination with your own products. The application examples are not required to undergo the customary tests and quality inspections of a chargeable product; they may have functional and performance defects as well as errors. It is your responsibility to use them in such a manner that any malfunctions that may occur do not result in property damage or injury to persons.

Disclaimer of liability

Siemens shall not assume any liability, for any legal reason whatsoever, including, without limitation, liability for the usability, availability, completeness and freedom from defects of the application examples as well as for related information, configuration and performance data and any damage caused thereby. This shall not apply in cases of mandatory liability, for example under the German Product Liability Act, or in cases of intent, gross negligence, or culpable loss of life, bodily injury or damage to health, non-compliance with a guarantee, fraudulent non-disclosure of a defect, or culpable breach of material contractual obligations. Claims for damages arising from a breach of material contractual obligations shall however be limited to the foreseeable damage typical of the type of agreement, unless liability arises from intent or gross negligence or is based on loss of life, bodily injury or damage to health. The foregoing provisions do not imply any change in the burden of proof to your detriment. You shall indemnify Siemens against existing or future claims of third parties in this connection except where Siemens is mandatorily liable.
By using the application examples you acknowledge that Siemens cannot be held liable for any damage beyond the liability provisions described.

Other information

Siemens reserves the right to make changes to the application examples at any time without notice. In case of discrepancies between the suggestions in the application examples and other Siemens publications such as catalogs, the content of the other documentation shall have precedence.
The Siemens terms of use (https://support.industry.siemens.com) shall also apply.

Security information

Siemens provides products and solutions with industrial security functions that support the secure operation of plants, systems, machines and networks.
In order to protect plants, systems, machines and networks against cyber threats, it is necessary to implement – and continuously maintain – a holistic, state-of-the-art industrial security concept. Siemens' products and solutions constitute one element of such a concept.
Customers are responsible for preventing unauthorized access to their plants, systems, machines and networks. Such systems, machines and components should only be connected to an enterprise network or the Internet if and to the extent such a connection is necessary and only when appropriate security measures (e.g. firewalls and/or network segmentation) are in place.
For additional information on industrial security measures that may be implemented, please visit https://www.siemens.com/industrialsecurity.
Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends that product updates are applied as soon as they are available and that the latest product versions are used. Use of product versions that are no longer supported, and failure to apply the latest updates may increase customer's exposure to cyber threats.
To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed at: https://www.siemens.com/industrialsecurity.

Table of contents

1 Library Overview

What you get

This document describes the block LCamHdl_CreateCamBasic of the LCamHdl block library. The block library provides you with tested code with clearly defined interfaces. They can be used as a basis for your task to be implemented.

A key concern of the document is to describe

- all blocks of the block library
- the functionality implemented through these blocks.

Furthermore, this documentation shows possible fields of application and helps you integrate the library into your STEP 7 project using step-by-step instructions.

Scope of application

- STEP 7 Professional V17
- Motion Control V6.0
- S7-1500T CPU as of firmware V2.9

1.1 Different user scenarios

Possible application(s) for the LCamHdl library

The present application is to support the user with the configuration of cam disks by applying the laws of motion.

In general, cam disks are electronic gears at a non-constant transition, where, for example, a constant drive motion is converted into a non-constant drive motion by applying the laws of motion.

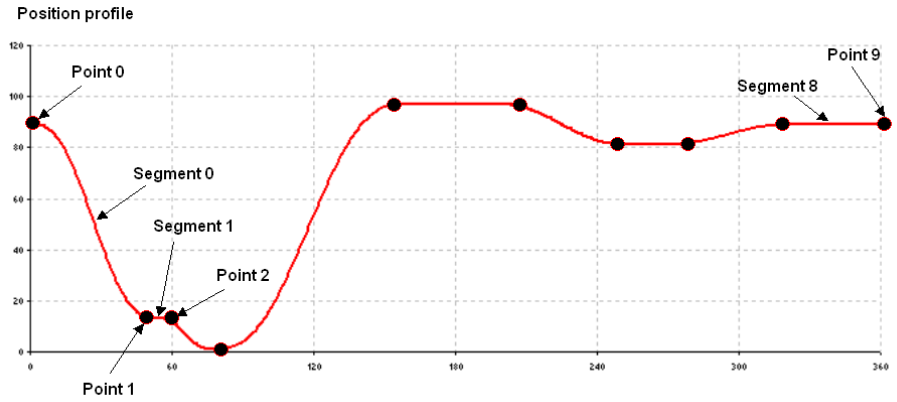
Within the scope of the SIMATIC S7-1500T, there are two ways to configure cam disks:

- at engineering in the TIA Portal with the help of the cam editor
- at runtime by definition of a cam profile

The present application is dedicated to the configuration of cam disks at runtime.

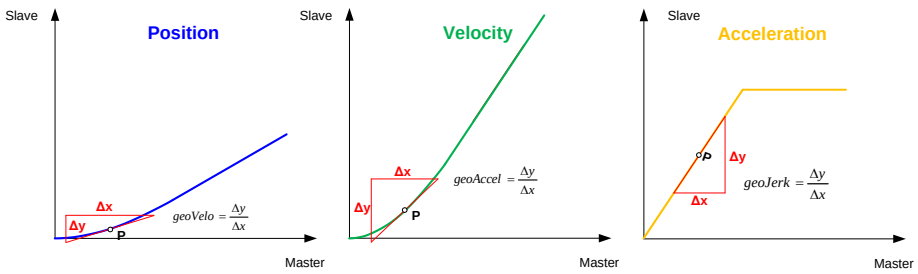
Usually cams can be defined by polynomial (5th degree), linear and standstill segments.

Figure 1-1: Cam disk consisting of several elements



To define the cam segments all points and their dynamics (1st and 2nd geometric derivation) have to be specified.

Figure 1-2: Derivation in the boundary points



The following FB calculates the parameters for the segments of the technology object.

The following section shows scenarios for a possible application of the LCamHdl library:

1.1.1 Scenario 1

A fully defined cam disk shall be created at runtime. Points in the cam disk and the according dynamics are known. Transitions can be made via straight lines and 5th degree polynomials, taken into account velocity and acceleration.

You should choose the **LCamHdl_CreateCamBasic** function block to create the cam disk. It eases the cam disk creation for cam disks with interpolation algorithms up to 5th degree polynomials.

Figure 1-3: Cam disk with 8 points created by LCamHdl_CreateCamBasic

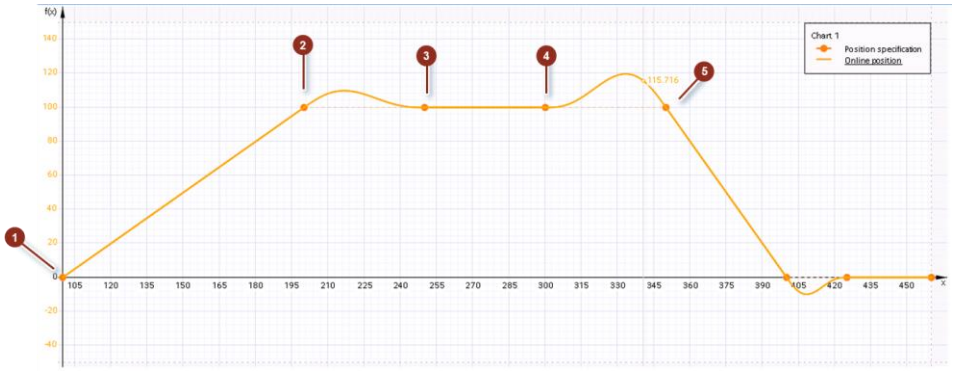


Figure 1-4: Configuration of cam with 8 points created by LCamHdl_CreateCamBasic

	Cam4_Basic	Array[1..8] of "LCa...	
1	Cam4_Basic[1]	"LCamHdl_typeBas...	
	leadingValue	LReal	100.0
	followingValue	LReal	0.0
	velocityRatio	LReal	1.0
	accelerationRa...	LReal	0.0
2	Cam4_Basic[2]	"LCamHdl_typeBas...	
	leadingValue	LReal	200.0
	followingValue	LReal	100.0
	velocityRatio	LReal	1.0
	accelerationRa...	LReal	0.0
3	Cam4_Basic[3]	"LCamHdl_typeBas...	
	leadingValue	LReal	250.0
	followingValue	LReal	100.0
	velocityRatio	LReal	0.0
	accelerationRa...	LReal	0.0
4	Cam4_Basic[4]	"LCamHdl_typeBas...	
	leadingValue	LReal	300.0
	followingValue	LReal	100.0
	velocityRatio	LReal	0.0
	accelerationRa...	LReal	0.0
5	Cam4_Basic[5]	"LCamHdl_typeBas...	
	leadingValue	LReal	350.0
	followingValue	LReal	100.0
	velocityRatio	LReal	-2.0
	accelerationRa...	LReal	0.0
	Cam4_Basic[6]	"LCamHdl_typeBas...	
	Cam4_Basic[7]	"LCamHdl_typeBas...	
	Cam4_Basic[8]	"LCamHdl_typeBas...	

NOTE Two cam points lead into one cam segment. One cam point is end of the segment before and start of next segment (except first and last point).

A maximum of 51 points is possible in a cam profile.

1.1.2 Scenario 2

The function block (FB) **LCamHdl_CreateCamAdvanced** can be used to merge working ranges and motion transitions into one cam disk at runtime. Unlike directly assigning the cam's data block, the FB can be used without having to calculate the polynomial coefficients before.

The FB is based on the motion rules for cam mechanisms according to VDI 2143.

The cam profile configuration of the position as well as the geometric derivations is made in the real section (e.g. velocity, acceleration, jerk).

There are different mathematic functions available for the motion transitions (elements), subsequently called profile types. Besides polynomials

- 3rd degree polynomial,
- 5th degree polynomial,
- 7th degree polynomial

further profiles exist

- straight line,
- quadratic parabola,
- basic sine,
- inclined sine,
- modified acceleration trapezoid,
- modified sine,
- sine-straight line-combination – velocity trapezoid,
- harmonic combination

In addition to that, it is also possible to transfer single points, which makes it possible to generate cam disks with combined ranges consisting of transition functions and of single points.

In difference to the **LCamHdl_CreateCamBasic** block the function block **LCamHdl_CreateCamAdvanced** works segment based. This allows gaps between segments and also the usage of the points array in the cam technology object.

NOTE Further information can be found in \2\.

Figure 1-5: Cam disk with 4 segments created with LCamHdl_CreateCamAdvanced

1.1.3 Scenario 3

A cam disk based on interpolation points is to be created at runtime. Only the X and Y coordinates of the interpolation points are known (X - master, Y - slave).

You should choose the **LCamHdl_CreateCamBasedOnXYPoints** function block. It eases the cam disk creation for cam disks consisting of just interpolation points.

The interpolation mode (linear / C splines / B splines) can be defined via the TO-Cam DB - *TO-Cam.InterpolationSettings.InterpolationMode*.

Figure 1-6: Cam disk example (interpolation mode C splines) created by LCamHdl_CreateCamBasedOnXYPoints

NOTE Further information can be found in \2\.

1.2 Hardware and software requirements

Requirements for this library

To be able to use the functionality of the library described in this document, the following hardware and software requirements must be met:

Hardware

Table 1-1

No.	Component	Article number	Alternative
1.	CPU 1515T-2 PN	6ES7 515-2TM01-0AB0	Other S7-1500T CPU with FW 2.9

Software

Table 1-2

No.	Component	Article number	Quantity
2.	STEP 7 Professional V17	6ES7822-1A.07-..	1

1.3 Library resources

What will you find in this section?

The following section gives you an overview of the size of the blocks of the LCamHdl library usable for basic cam creation in the main and load memory.

Overall size

The overall size of the blocks of the LCamHdl library usable for basic cam creation (TO_Cam or TO_Cam_10K) in the code work-memory is 3 Kbytes, in the data work-memory 5 Kbytes and 58 Kbytes in the load memory.

Size of the individual blocks

Table 1-3 Size of the blocks[1]

Block	Symbol	Size in code work-memory [Kbytes]	Size in data work-memory [Kbytes]	Size in load memory [Kbytes]
FB 31101	LCamHdl_CreateCamBasic	3		49
FB 31111	LCamHdl_CreateCam10kBasic	3		49
DB 31101	InstLCamHdl_CreateCamBasic		5	9
DB 31111	InstLCamHdl_CreateCam10kBasic		5	9

[1] Instance data blocks (prefix InstLCamHdl_) are not delivered with the library. They will be generated automatically with the call of a function block.

2 Blocks of the Library

What will you find in this section?

This chapter lists and explains all blocks of the LCamHdl library usable for basic cam creation. Before that, however, you are informed of the blocks that are essentially involved in the implementation of the functionality.

2.1 List of the blocks

The following table lists all blocks of the LCamHdl library usable for basic cam creation.

Table 2-1: List of blocks

Block	Symbol	Classification
FB 31101	LCamHdl_CreateCamBasic	In-house development
FB 31111	LCamHdl_CreateCam10kBasic	In-house development

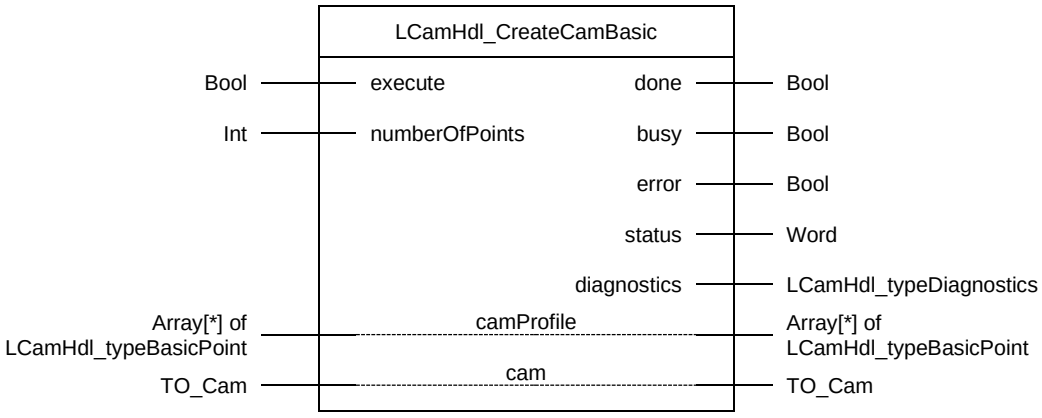
2.2 Explanation of the blocks

The following table explains all blocks of the LCamHdl library usable for basic cam creation.

2.2.1 FB LCamHdl_CreateCamBasic (FB 31101)

Figure

Figure 2-1: Block diagram of *LCamHdl_CreateCamBasic*



Principle of operation

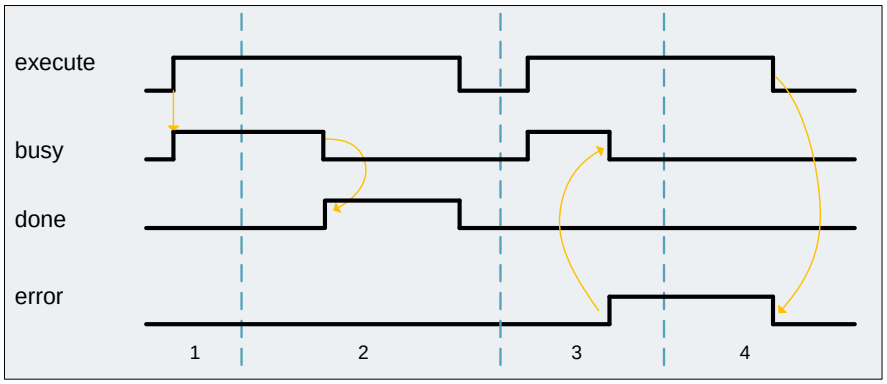
A cam disk can be created at runtime with a SIMATIC S7-1500T CPU. To interpolate the movement between two points of the cam profile a 5th degree polynomial is used. The leading value of the first point defines the start of the cam disk and the leading value of the last point defines the end of the cam disk. As the indexes in the cam profile increase also the according leading values have to increase.

A maximum number of 51 points can be used in a cam profile to define a cam.

The FB LCamHdl_CreateCamBasic first fills the necessary segments in the cam technology object and then interpolates the cam.

Function characteristics

Figure 2-2: Timing diagram of the *LCamHdl_CreateCamBasic* function block



1. The function block is activated by setting the *execute* input. The processing of the FB is indicated with *busy* = TRUE and *done* = FALSE.
2. If *busy* is FALSE and *done* is TRUE the function block finished processing. The cam disk is interpolated – a later addition of points / segments requires a new interpolation of the technology object cam.
3. If an error occurs during processing (e.g. error when executing the function block, internal errors of system functions, parameter supply error etc.), the *error* output is set and the corresponding error ID is written to the output *status*. Additional error information is provided at the output *diagnostics*.
4. All errors will be acknowledged and *error* output is reset if the *execute* input is reset. Signals stay active for at least one cycle. Reset of the diagnostic information is done with the next rising edge of *execute*.

Input parameters

Table 2-2 Input parameters of *LCamHdl_CreateCamBasic*

Parameter	Data type	Comment
execute	Bool	Rising edge starts action once (default: FALSE)
numberOfPoints	Int	Number of used points of *camProfile* (default: -1 for whole array; maximum 51)

Output parameters

Table 2-3 Output parameters of *LCamHdl_CreateCamBasic*

Parameter	Data type	Comment
done	Bool	TRUE: Commanded action has been completed successfully (default: FALSE)
busy	Bool	TRUE: FB is not finished and new output values can be expected (default: FALSE)
error	Bool	TRUE: Rising edge informs that an error occurred during the execution of the FB (default: FALSE)
status	Word	16#0000 - 16#7FFF: Status of the FB, 16#8000 - 16#FFFF: Error identification (default: 16#7000, STATUS_NO_CALL)
diagnostics	LCamHdl_typeDiagnostics	Diagnostics information of FB

InOut parameters

Table 2-4 InOut parameters of *LCamHdl_CreateCamBasic*

Parameter	Data type	Comment
camProfile	Array[*] of LCamHdl_typeBasicPoint	Definition of the cam disk to be created
cam	TO_Cam	Technology object cam disk

Status and error displays

Table 2-5 *LCamHdl_CreateCamBasic* diagnostics messages

Status	Meaning	Remedy / notes
16#0000	STATUS_EXECUTION_FINISHED	Execution finished without errors
16#7000	STATUS_NO_CALL	No call of FB
16#7001	STATUS_FIRST_CALL	First call of FB after enabling
16#7002	STATUS_SUBSEQUENT_CALL	Subsequent call of FB
16#8200	ERR_NO_OF_POINTS_OUT_OF_ BOUNDS	numberOfPoints is greater than the points in the camProfile or there is only one point in the camProfile
16#8201	ERR_CAM_SEGMENTS_OUT_OF_ BOUNDS	Too many segments needed to define cam (Maximum 50)
16#8202	ERR_INVALID_LEADING_VALUE	Leading value is not valid (has to increase from one point to the next)
16#8400	ERR_CAM_DISK_IN_USE	Cam disk is in use and can't be interpolated
16#8600	ERR_INTERPOLATE_CAM	Error occurred while interpolating cam – see return value of system function (diagnostics.subfunctionStatus)
16#8601	ERR_INVALID_STATE	Invalid state of the state machine
16#8602	ERR_RESET_CAM	Error at reset cam – see return value of system function (diagnostics.subfunctionStatus)

Status	Meaning	Remedy / notes
16#8604	ERR_COPY_CAM_DATA	Error at copy cam data – see return value of system function (diagnostics.subfunctionStatus)

2.2.2 FB LCamHdl_CreateCam10kBasic (FB 31111)

The function block LCamHdl_ CreateCam10kBasic is a copy of the function block LCamHdl_ CreateCamBasic. The "..10k.." version enables using a cam technology object of type TO_Cam_10k instead of TO_Cam.

2.2.3 PLC data types

LCamHdl_typeBasicPoint

Table 2-6: Parameter of *LCamHdl_typeBasicPoint*

Name	Data type	Value	Comment
leadingValue	LReal	0.0	Leading value at current point
followingValue	LReal	0.0	Following value at current point
velocityRatio	LReal	0.0	Gradient of the cam disk at current point
accelerationRatio	LReal	0.0	Curvature of the cam disk at current point

LCamHdl_typeDiagnostics

Table 2-7: Parameter of *LCamHdl_typeDiagnostics*

Name	Data type	Value	Comment
status	Word	16#0000	Status of FB
subfunctionStatus	Word	16#0000	Status or return value of called FBs, FCs and system blocks
state	DInt	0	State of the state machine
errorElementNo	DInt	-1	Index of the camProfile with the first error (-1: no parameter with error)

3 Working with the Library

What will you find in this section?

This chapter consists of instructions for integrating the LCamHdl library into your STEP 7 project and instructions for using the library blocks.

3.1 Integrating the library into STEP 7

The table below lists the steps for integrating the LCamHdl library into your STEP 7 project. Subsequently, you can use the blocks of the LCamHdl library.

Note The following section assumes that a STEP 7 project exists.

Table 3-1: Integrating the library into STEP 7

No.	Action
1.	Extract the library LCamHdl_V1_x_x.zip to a local folder.
2.	In TIA Portal select “Options” -> “Global libraries” -> “Open library…”.
3.	Browse to the file LCamHdl.al17. It can be found in the subfolder LCamHdl of the extracted zip file.
4.	Open the global library in read-only mode.
5.	The LCamHdl library is now available in the task card “Global libraries”.

3.2 Integrating the library blocks into STEP 7

The table below lists the steps for integrating the blocks of the LCamHdl library into your STEP 7 program.

Figure 3-1: Integrating the library blocks into STEP 7

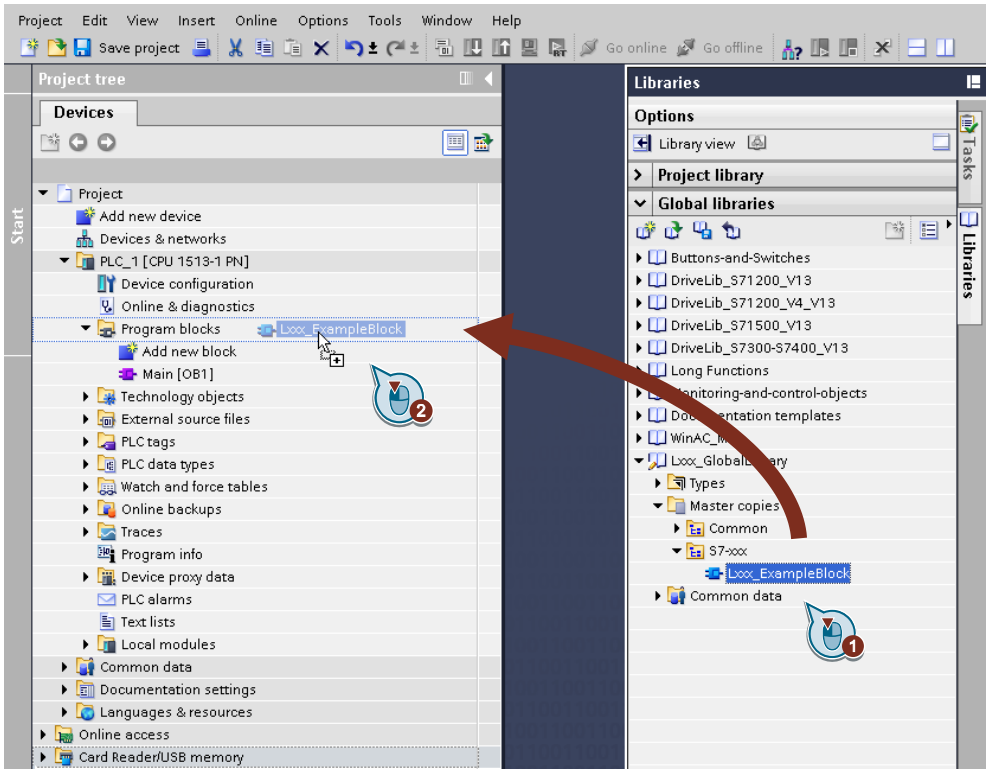


Table 3-2: Integrating the library blocks into STEP 7

No.	Action	Note
1.	Copy the folder *LCamHdl_Types* with Drag & Drop into the “PLC data types” in the PLC.	Copy PLC data types
2.	Copy the *LCamHdl_CreateCamXXX* FB with Drag & Drop into the “Program blocks” in the PLC.	Copy program blocks
3.	Now the blocks can be configured and called in the user program.	

4 Notes and Support

What will you find in this section?

This chapter provides further support in handling the described LCamHdl library.

NOTE Parameter comments in the programming editor are only available in language 'English (United States)'

4.1 Creating a straight line between two points

To create a straight line between two points the correct velocity and acceleration ratio have to be chosen.

In this example a straight line should be created between point 1 and point 2. A straight line means that the slave axis follows the master axis in gearing.

Figure 4-1: Cam disk with 8 points created by *LCamHdl_CreateCamBasic*

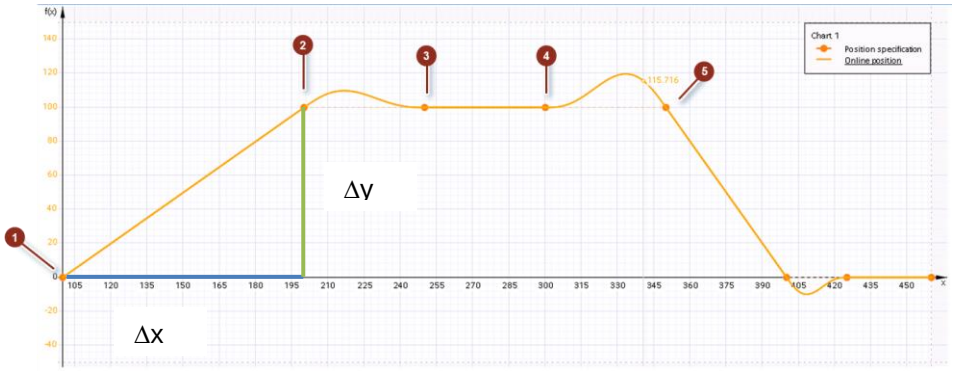
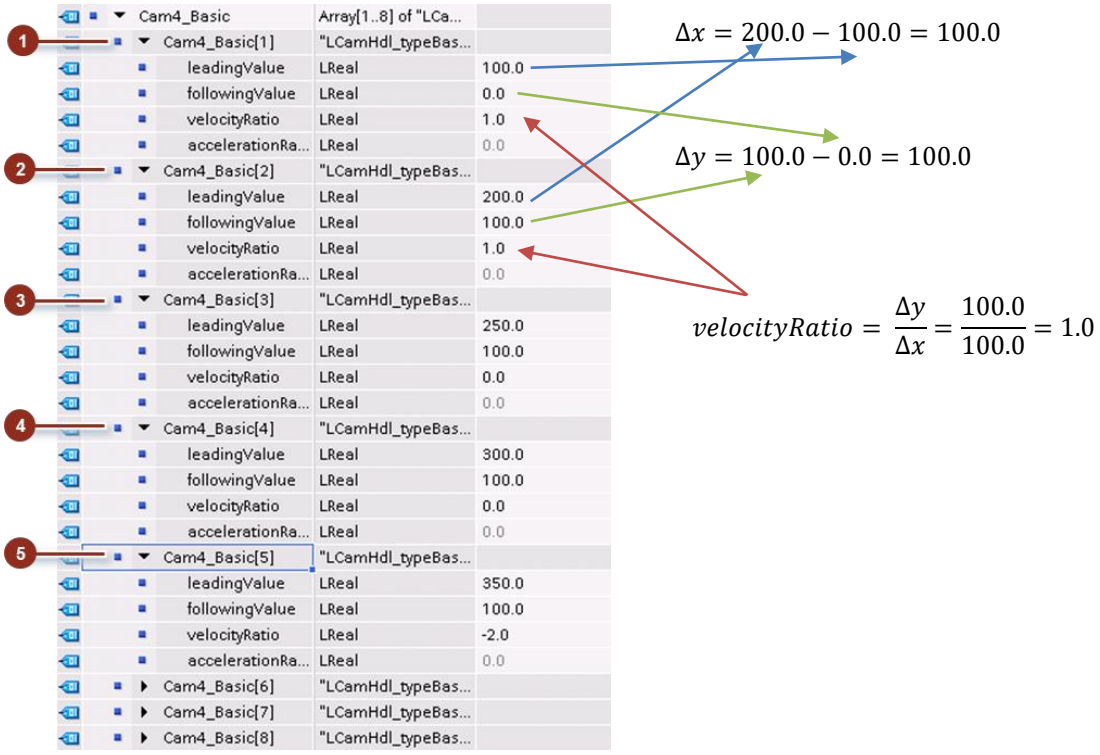


Figure 4-2: Configuration of cam with 8 points created by *LCamHdl_CreateCamBasic*



5 Appendix

5.1 Service and support

Industry Online Support

Do you have any questions or need assistance?

Siemens Industry Online Support offers round the clock access to our entire service and support know-how and portfolio.

The Industry Online Support is the central address for information about our products, solutions and services.

Product information, manuals, downloads, FAQs, application examples and videos – all information is accessible with just a few mouse clicks:
support.industry.siemens.com

Technical Support

The Technical Support of Siemens Industry provides you fast and competent support regarding all technical queries with numerous tailor-made offers – ranging from basic support to individual support contracts. Please send queries to Technical Support via Web form:
www.siemens.com/industry/supportrequest

SITRAIN – Training for Industry

We support you with our globally available training courses for industry with practical experience, innovative learning methods and a concept that's tailored to the customer's specific needs.

For more information on our offered trainings and courses, as well as their locations and dates, refer to our web page:
www.siemens.com/sitrain

Service offer

Our range of services includes the following:

- Plant data services
- Spare parts services
- Repair services
- On-site and maintenance services
- Retrofitting and modernization services
- Service programs and contracts

You can find detailed information on our range of services in the service catalog web page:
support.industry.siemens.com/cs/sc

Industry Online Support app

You will receive optimum support wherever you are with the "Siemens Industry Online Support" app. The app is available for iOS and Android:
support.industry.siemens.com/cs/ww/en/sc/2067

5.2 Application support

Siemens AG
Digital Industries
Factory Automation
Production Machines
DI FA PMA APC
Frauenauracher Str. 80
91056 Erlangen, Germany

mailto: tech.team.motioncontrol@siemens.com

5.3 Links and literature

Table 5-1

No.	Topic
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Link to this entry page of this application example https://support.industry.siemens.com/cs/ww/en/view/105644659

5.4 Change documentation

Table 5-2

Version	Date	Modifications
V1.0.0	10/2016	First version
V1.0.1	04/2017	Additional information in chapter ‘Notes and Support‘
V1.1	11/2017	Change of interface: order of busy and done reversed
V1.2	03/2020	Updated, e.g. chapter 1 - scenario 3
V1.3	05/2021	Scope of application is now STEP 7 Professional V17 (Motion Control V6.0, firmware V2.9) New block for handling the new cam technology object of type TO_Cam_10k