Creation of cam disks at runtime for S7-1500T

Library LCamHdl – Additional functions

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 additional blocks 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 Functionality

The additional functions of the library LCamHdl provide blocks to calculate the minimum and maximum following value of a cam disk, and their derivatives - no matter if this cam disk was created

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

Furthermore, the maximum possible master velocity can be calculated with respect to the given maximum dynamics (velocity, acceleration and jerk) of the following axis. Prerequisite for this is that the cam disk is used at a constant master velocity.

In addition, there is also a function available to calculate the maximum occurring following axis dynamics (velocity, acceleration and jerk) for usage with a constant master velocity.

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 additional functions in the main and load memory.

Overall size

The overall size of the blocks of the LCamHdl library for additional functions (TO_Cam or TO_Cam_10K) in the code work-memory is 5 Kbytes, in the data work-memory 2 Kbytes and 70 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 31102	LCamHdl_GetCamStatusWord	1		8
FB 31112	LCamHdl_GetCam10kStatusWord	1		8
FB 31155	LCamHdl_GetCamFollowingMinMax	3		44
FB 31165	LCamHdl_GetCam10kFollowingMinMax	3		44
FC 31150	LCamHdl_GetCamMaxVeloMaster	1		12
FC 31151	LCamHdl_GetCamMaxSlave Dynamics	1		8
DB 31102	InstLCamHdl_GetCamStatusWord		1	2
DB 31112	InstLCamHdl_GetCam10kStatusWord		1	2
DB 31155	InstLCamHdl_GetCamFollowing MinMax		1	4
DB 31165	InstLCamHdl_GetCam10kFollowing MinMax		1	4

[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 additional blocks of the LCamHdl library. 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 additional blocks of the LCamHdl library.

Table 2-1: List of blocks

Block	Symbol	Classification
FB 31155	LCamHdl_GetCamFollowingMinMax	In-house development
FB 31165	LCamHdl_GetCam10kFollowingMinMax	In-house development
FC 31150	LCamHdl_GetCamMaxVeloMaster	In-house development
FC 31151	LCamHdl_GetCamMaxSlaveDynamics	In-house development
FB 31102	LCamHdl_GetCamStatusWord	In-house development
FB 31112	LCamHdl_GetCam10kStatusWord	In-house development

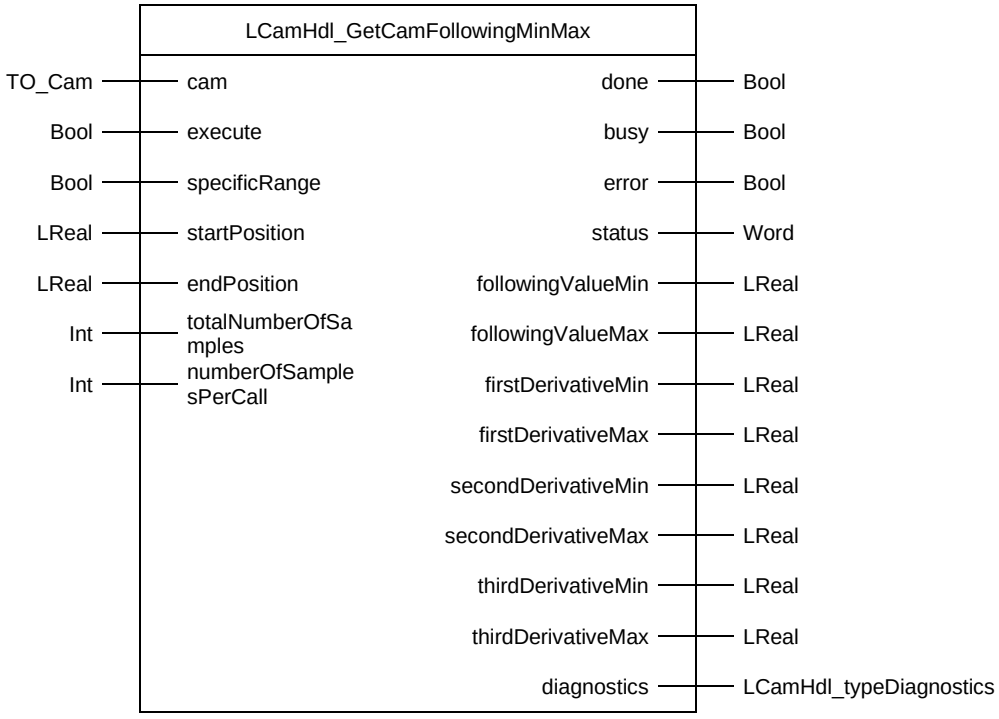
2.2 Explanation of the blocks

The following table explains all additional blocks of the LCamHdl library.

2.2.1 FB LCamHdl_GetCamFollowingMinMax (FB 31155)

Figure

Figure 2-1: Block diagram of *LCamHdl_GetCamFollowingMinMax*



Principle of operation

The function block LCamHdl_GetCamFollowingMinMax determines the minimum and maximum following values of the cam and their first, second and third derivatives. The minimum and maximum values are determined by scanning the cam with a defined number of samples (see input *totalNumberOfSamples*).

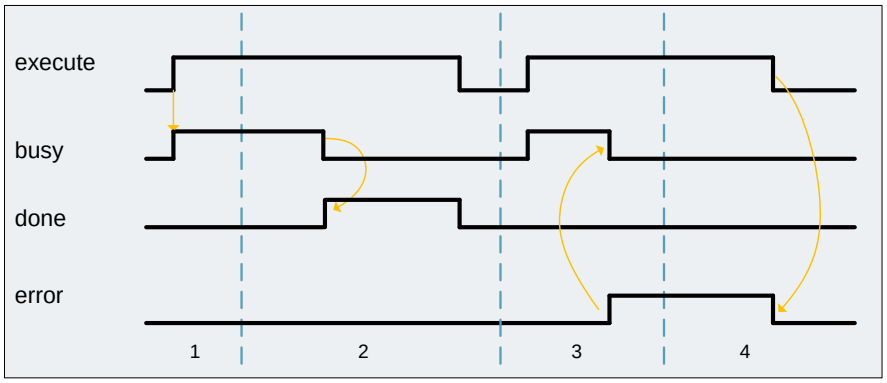
NOTE To determine the first and second derivatives of the cam's following values, the system function MC_GetCamFollowingValue is used. The third derivative is calculated by numerical differentiation of the second derivative.

NOTE The value of input *totalNumberOfSamples* defines the number of samples per complete leading value range (definition range) of the cam disk. If the minimum and maximum values are only determined in a specific subrange of the cam (input *specificRange* = TRUE), the resulting number of samples is therefore reduced accordingly, i.e. the total runtime of the block is also reduced.

During the determination the output busy indicates the value TRUE. The error free completion is shown with *done* = TRUE. Status and error are output at the outputs *status* and *error* as well as *diagnostics*.

Function characteristics

Figure 2-2: Timing diagram of the *LCamHdl_GetCamFollowingMinMax* 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.
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_GetCamFollowingMinMax*

Parameter	Data type	Comment
cam	TO_Cam	Reference to the cam disk
execute	Bool	Rising edge starts action once (default: FALSE)
specificRange	Bool	TRUE: Determine minima and maxima in a specified subrange; FALSE: Determine minima and maxima in the complete leading value range (definition range) of the cam disk (default: FALSE)
startPosition	LReal	Start position of the specific range for determining the minima and maxima (only relevant if specificRange = TRUE) (default: 0.0)
endPosition	LReal	End position of the specific range for determining the minima and maxima (only relevant if specificRange = TRUE) (default: 0.0)
totalNumberOfSamples	Int	Total number of samples per complete leading value range (definition range) of the cam disk for MC_GetCamFollowingValue functionality. Use of 721 samples is recommended for leading value range 360°, i.e. every 0.5° one sample (default: 1000)
numberOfSamplesPerCall	Int	Number of samples ("MC_GetCamFollowingValue calls") per block call. The higher the value, the higher the OB runtime, but less OB calls necessary (default: 30)

Output parameters

Table 2-3: Output parameters of *LCamHdl_GetCamFollowingMinMax*

Parameter	Data type	Comment
done	Bool	TRUE: Commanded functionality 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: 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)
followingValueMin	LReal	Cam following value minimum (valid when 'done' = TRUE) (default: 0.0)

Parameter	Data type	Comment
followingValueMax	LReal	Cam following value maximum (valid when 'done' = TRUE) (default: 0.0)
firstDerivativeMin	LReal	Cam following value first derivative minimum (valid when 'done' = TRUE) (default: 0.0)
firstDerivativeMax	LReal	Cam following value first derivative maximum (valid when 'done' = TRUE) (default: 0.0)
secondDerivativeMin	LReal	Cam following value second derivative minimum (valid when 'done' = TRUE) (default: 0.0)
secondDerivativeMax	LReal	Cam following value second derivative maximum (valid when 'done' = TRUE) (default: 0.0)
thirdDerivativeMin	LReal	Cam following value third derivative minimum - calculated by numerical differentiation (valid when 'done' = TRUE) (default: 0.0)
thirdDerivativeMax	LReal	Cam following value third derivative maximum - calculated by numerical differentiation (valid when 'done' = TRUE) (default: 0.0)
diagnostics	LCamHdl_typeDiagnostics	Diagnostics information of FB

Status and error displays

Table 2-4 *LCamHdl_GetCamFollowingMinMax* diagnostics messages

Status	Meaning	Remedy / notes
16#0000	STATUS_EXECUTION_FINISHED	Execution finished without errors
16#7000	STATUS_NO_CALL	No job being currently processed
16#7001	STATUS_FIRST_CALL	First call after incoming new job (rising edge 'execute')
16#7002	STATUS_SUBSEQUENT_CALL	Subsequent call during active processing without further details
16#8210	ERR_TOTAL_NUMBER_OF_SAMPLES	Invalid totalNumberOfSamples, 2<=totalNumberOfSamples
16#8211	ERR_NUMBER_OF_SAMPLES_PER_CALL	Invalid numberOfSamplesPerCall, 1<=numberOfSamplesPerCall
16#8601	ERR_INVALID_STATE	Invalid state of the state machine
16#8603	ERR_GETCAMFOLLOWINGVALUE	Error at MC_GetCamFollowingValue – see return value of system function (diagnostics.subfunctionStatus)

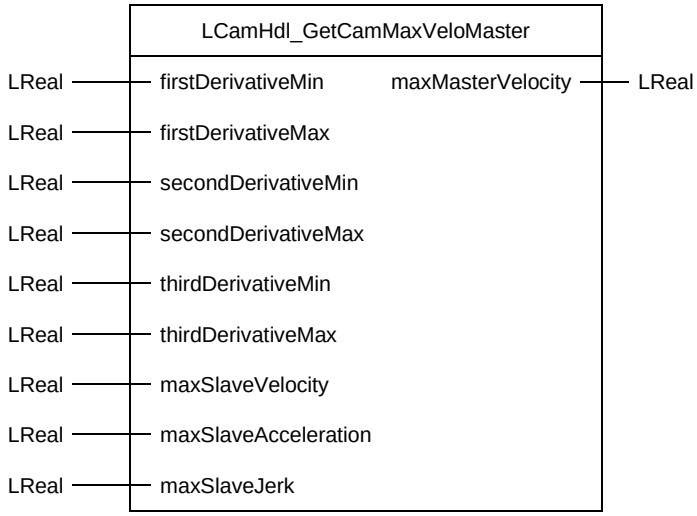
2.2.2 FB LCamHdl_GetCam10kFollowingMinMax (FB 31165)

The function block LCamHdl_GetCam10kFollowingMinMax is a copy of the function block LCamHdl_GetCamFollowingMinMax. The "..10k.." version enables using a cam technology object of type TO_Cam_10k instead of TO_Cam.

2.2.3 FC LCamHdl_GetCamMaxVeloMaster (FC 31150)

Figure

Figure 2-3: Block diagram of *LCamHdl_GetCamMaxVeloMaster*



Principle of operation

The function LCamHdl_GetCamMaxVeloMaster calculates the maximum possible master velocity with respect to the given maximum dynamics(velocity, acceleration and jerk) of the following axis and the cam following values (min. and max first, second and third derivatives).

Prerequisite is, that the cam disk is used at a constant master velocity - accelerations or decelerations of the master axis are not taken into account. The time base of the master and slave axis must be equal (e.g. seconds).

The minimum and maximum following value derivatives can be determined with the function block LCamHdl_GetCamFollowingMinMax.

The function calculates the maximum master velocity using the following equations:

$$v_{MasterMax\_v} = \frac{s'}{v_{LimitSlave}}$$

$$v_{MasterMax\_a} = \sqrt{\frac{s''}{a_{LimitSlave}}}$$

$$v_{MasterMax\_j} = \left(\frac{s'''}{j_{LimitSlave}}\right)^{1/3}$$

s' - following axis first derivative (min / max)

s'' - following axis second derivative (min / max)

s''' - following axis third derivative (min / max)

Input parameters

Table 2-5: Input parameters of *LCamHdl_GetCamMaxVeloMaster*

Parameter	Data type	Comment
firstDerivativeMin	LReal	Cam following value first derivative minimum
firstDerivativeMax	LReal	Cam following value first derivative maximum
secondDerivativeMin	LReal	Cam following value second derivative minimum
secondDerivativeMax	LReal	Cam following value second derivative maximum
thirdDerivativeMin	LReal	Cam following value third derivative minimum
thirdDerivativeMax	LReal	Cam following value third derivative maximum
maxSlaveVelocity	LReal	Maximum permissible velocity of the slave axis
maxSlaveAcceleration	LReal	Maximum permissible acceleration of the slave axis
maxSlaveJerk	LReal	Maximum permissible jerk of the slave axis, ignored if <= 0.0

Output parameters

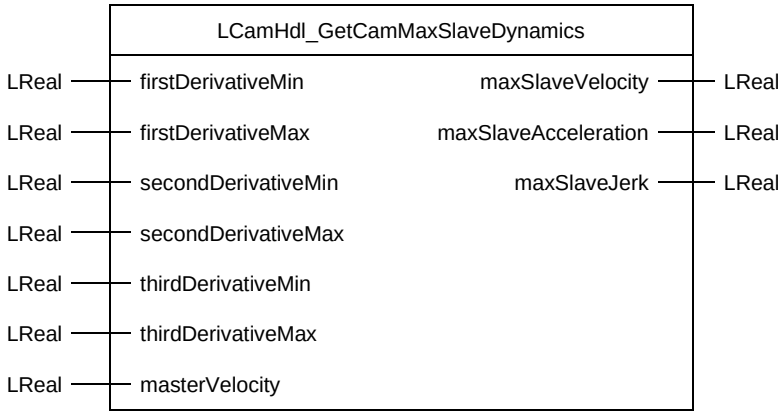
Table 2-6: Input parameters of *LCamHdl_GetCamMaxVeloMaster*

Parameter	Data type	Comment
maxMasterVelocity	LReal	Maximum permissible velocity of the master axis to not exceed the maximum slave axis dynamics

2.2.4 FC LCamHdl_GetCamMaxSlaveDynamics (FC 31151)

Figure

Figure 2-4: Block diagram of *LCamHdl_GetCamMaxSlaveDynamics*



Principle of operation

The function LCamHdl_GetCamMaxSlaveDynamics calculates the resulting maximum following (slave) axis velocity, acceleration and jerk with respect to the given master velocity and the cam following values (min. and max of first, second and third derivatives).

Prerequisite is, that the cam disk is used at a constant master velocity - accelerations or decelerations of the master axis are not taken into account. The time base of the master and slave axis must be equal (e.g. seconds).

The minimum and maximum following value derivatives can be determined with the function block LCamHdl_GetCamFollowingMinMax.

The function calculates the maximum output values using the following equations:

$$v_{Slave} = s' * v_{Master}$$
$$a_{Slave} = s'' * v_{Master}^2$$
$$j_{Slave} = s''' * v_{Master}^3$$

s' - following axis first derivative (min / max)

s'' - following axis second derivative (min / max)

s''' - following axis third derivative (min / max)

Input parameters

Table 2-7: Input parameter of *LCamHdl_GetCamMaxSlaveDynamics*

Parameter	Data type	Comment
firstDerivativeMin	LReal	Cam following value first derivative minimum
firstDerivativeMax	LReal	Cam following value first derivative maximum
secondDerivativeMin	LReal	Cam following value second derivative minimum
secondDerivativeMax	LReal	Cam following value second derivative maximum
thirdDerivativeMin	LReal	Cam following value third derivative minimum
thirdDerivativeMax	LReal	Cam following value third derivative maximum
masterVelocity	LReal	Master velocity

Output parameters

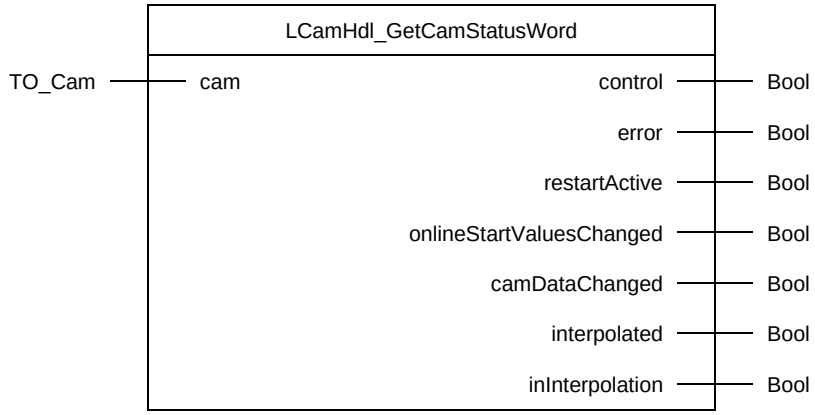
Table 2-8: Output parameter of *LCamHdl_GetCamMaxSlaveDynamics*

Parameter	Data type	Comment
maxSlaveVelocity	LReal	Maximum slave velocity
maxSlaveAcceleration	LReal	Maximum slave acceleration
maxSlaveJerk	LReal	Maximum slave jerk

2.2.5 FB LCamHdl_GetCamStatusWord (FB 31102)

Figure

Figure 2-5: Block diagram of *LCamHdl_GetCamStatusWord*



Principle of operation

The LCamHdl_GetCamStatusWord FB is splitting the status word of a TO_Cam into bits.

Input parameters

Table 2-9 Input parameters of *LCamHdl_GetCamStatusWord*

Parameter	Data type	Comment
cam	TO_Cam	Reference to the cam

Output parameters

Table 2-10 Output parameters of *LCamHdl_GetCamStatusWord*

Parameter	Data type	Comment
control	Bool	TRUE: Cam in use FALSE: Cam not in use (default: FALSE)
error	Bool	TRUE: Error present FALSE: No error present (default: FALSE)
restartActive	Bool	TRUE: "Restart" active. The technology object is being reinitialized FALSE: No "Restart" active (default: FALSE)
onlineStartValuesChanged	Bool	TRUE: Change to "Restart" tags. For the changes to be applied, the technology object must be reinitialized FALSE: "Restart" tags unchanged (default: FALSE)
camDataChanged	Bool	TRUE: The definition range of the cam has

Parameter	Data type	Comment
		changed in the technology data block FALSE: No change (default: FALSE)
interpolated	Bool	TRUE: Cam is interpolated FALSE: Cam is not interpolated (default: FALSE)
inInterpolation	Bool	TRUE: Cam undergoing interpolation FALSE: Cam not undergoing interpolation (default: FALSE)

2.2.6 FB LCamHdl_GetCam10kStatusWord (FB 31112)

The function block LCamHdl_GetCam10kStatusWord is a copy of the function block LCamHdl_GetCamStatusWord. The "..10k.." version enables using a cam technology object of type TO_Cam_10k instead of TO_Cam.

2.2.7 PLC data types

LCamHdl_typeDiagnostics

Table 2-11: 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

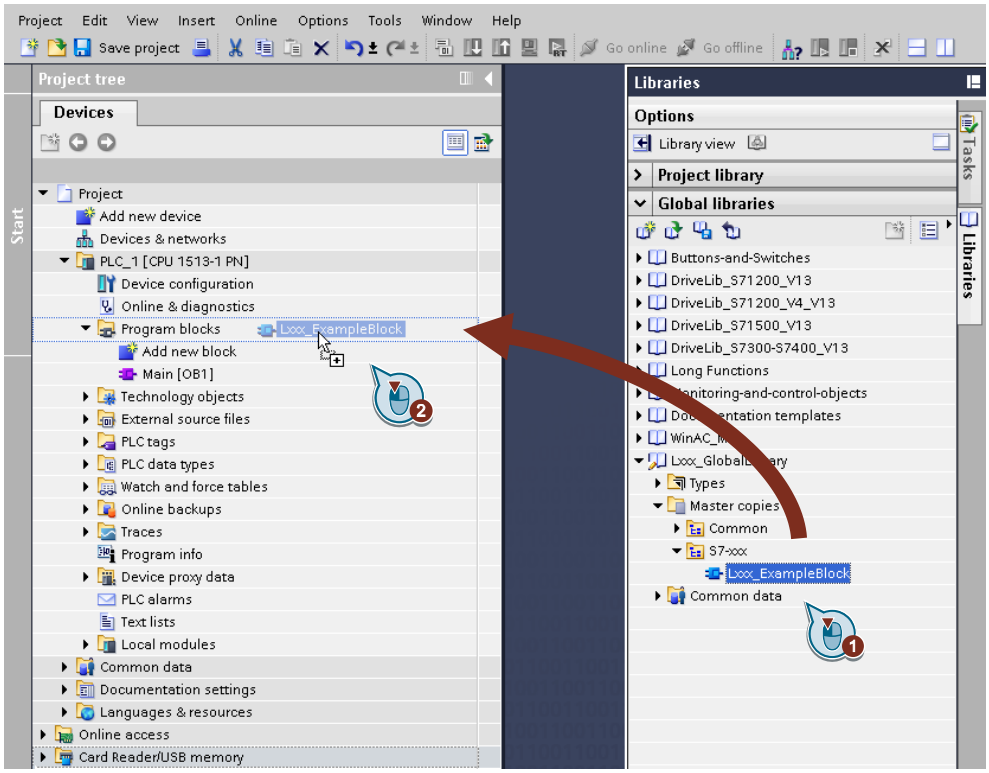


Table 3-2: Integrating the library blocks into STEP 7

No.	Action	Note
1.	Copy the folder *LCamHdl_Types* with Drag & Drop into the “PLC data types” in the PLC. The type *LCamHdl_typeDiagnostics* is used.	Copy PLC data types
2.	Copy the folder *LCamHdl_AdditionalFunctions* 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)'

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
V1.2	03/2020	First version
V1.3	05/2021	Scope of application is now STEP 7 Professional V17 (Motion Control V6.0, firmware V2.9) New blocks for handling the new cam technology object of type TO_Cam_10k