SIEMENS

Library description • 10/2016

Print mark acquisition with TO measuring input for S7-1500 / S7-1500T

LPrintMark for SIMATIC

https://support.industry.siemens.com/cs/ww/en/view/109475573

Warranty and liability

Note

The Application Examples are not binding and do not claim to be complete regarding the circuits shown, equipping and any eventuality. The Application Examples do not represent customer-specific solutions. They are only intended to provide support for typical applications. You are responsible for ensuring that the described products are used correctly. These Application Examples do not relieve you of the responsibility to use safe practices in application, installation, operation and maintenance. When using these Application Examples, you recognize that we cannot be made liable for any damage/claims beyond the liability clause described. We reserve the right to make changes to these Application Examples at any time without prior notice.
If there are any deviations between the recommendations provided in these Application Examples and other Siemens publications − e.g. Catalogs − the contents of the other documents have priority.

We do not accept any liability for the information contained in this document.
Any claims against us – based on whatever legal reason – resulting from the use of the examples, information, programs, engineering and performance data etc., described in this Application Example shall be excluded. Such an exclusion shall not apply in the case of mandatory liability, e.g. under the German Product Liability Act ("Produkthaftungsgesetz"), in case of intent, gross negligence, or injury of life, body or health, guarantee for the quality of a product, fraudulent concealment of a deficiency or breach of a condition which goes to the root of the contract ("wesentliche Vertragspflichten"). The damages for a breach of a substantial contractual obligation are, however, limited to the foreseeable damage, typical for the type of contract, except in the event of intent or gross negligence or injury to life, body or health. The above provisions do not imply a change of the burden of proof to your detriment.
Any form of duplication or distribution of these Application Examples or excerpts hereof is prohibited without the expressed consent of the Siemens AG.

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 only form one element of such a concept.
Customer is responsible to prevent unauthorized access to its plants, systems, machines and networks. Systems, machines and components should only be connected to the enterprise network or the internet if and to the extent necessary and with appropriate security measures (e.g. use of firewalls and network segmentation) in place.
Additionally, Siemens' guidance on appropriate security measures should be taken into account. For more information about industrial security, please visit http://www.siemens.com/industrialsecurity.

Siemens' products and solutions undergo continuous development to make them more secure. Siemens strongly recommends to apply product updates as soon as available and to always use the latest product versions. Use of product versions that are no longer supported, and failure to apply latest updates may increase customer's exposure to cyber threats.
To stay informed about product updates, subscribe to the Siemens Industrial Security RSS Feed under http://www.siemens.com/industrialsecurity.

Table of contents

1 Library Overview

What you get

This document describes the LPrintMark block library. The block library provides you with the 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 V14
- S7-1500 CPU as of firmware 2.0
- S7-1500T CPU as of firmware 2.0

1.1 Different user scenarios

Possible application for the LPrintMark library

Often in synchronous operations, the phase between the master axis and the slave axis is not the crucial factor. More the phase between the master axis and the product or rather the material that is transported via the slave axis is relevant.

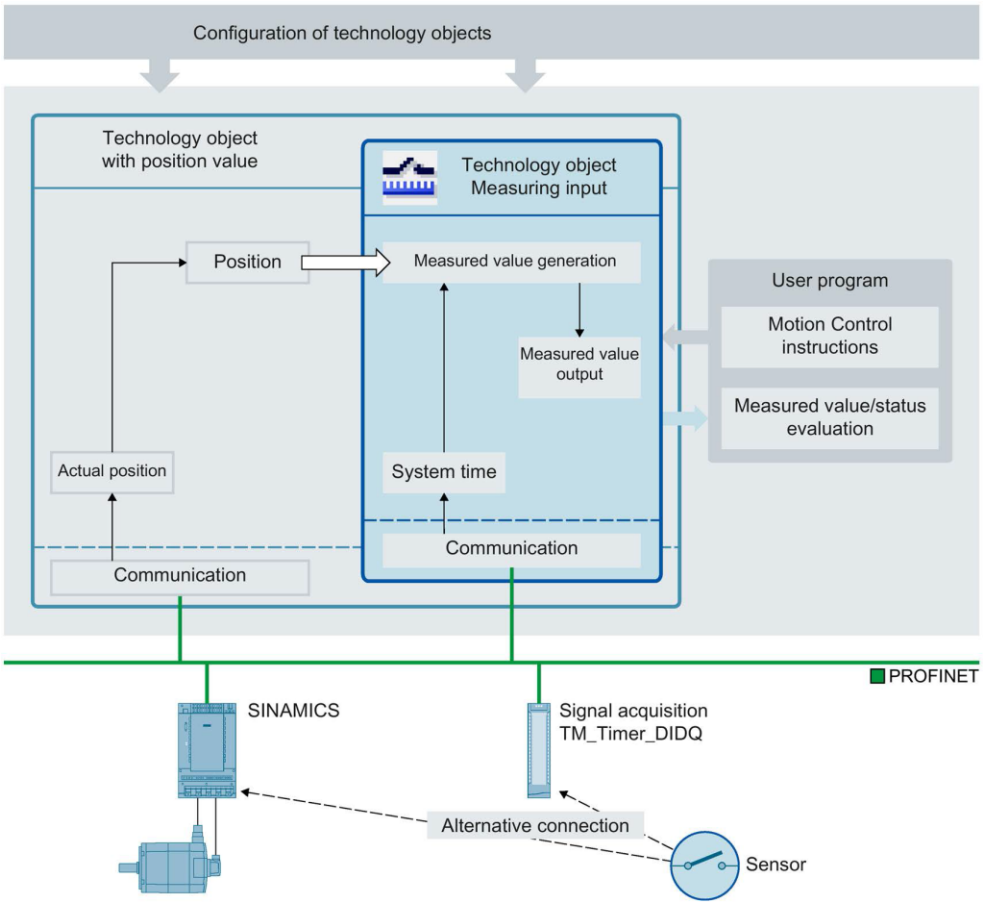
Possible reasons for arising deviations between product and feeding axis are:

- Slippage between product and axis
- Wearing of rollers
- Material faults or rather tolerances of the product
- Intervention of machine operators

The blocks of the LPrintMark library are dedicated to support the user in acquiring these deviations or rather displacements by means of the technology object (TO) measuring input. The deviation is the calculated difference between a specified (expected) setpoint position and the measured value. The handling of the measuring input technology object (i.e. Motion Control instructions MC_MeasuringInput and MC_MeasuringInputCyclic) and the subsequent calculation of the present difference are the main features of the library blocks.

The figure below shows the basic principle of operation of the measuring input technology object:

Figure 1-1



NOTE The maybe necessary correction routines (movements) are not part of the library. The blocks of the LPrintMark library are only dedicated to acquire the arising deviations.

The correct configuration of the measuring input technology object is not part of this manual.

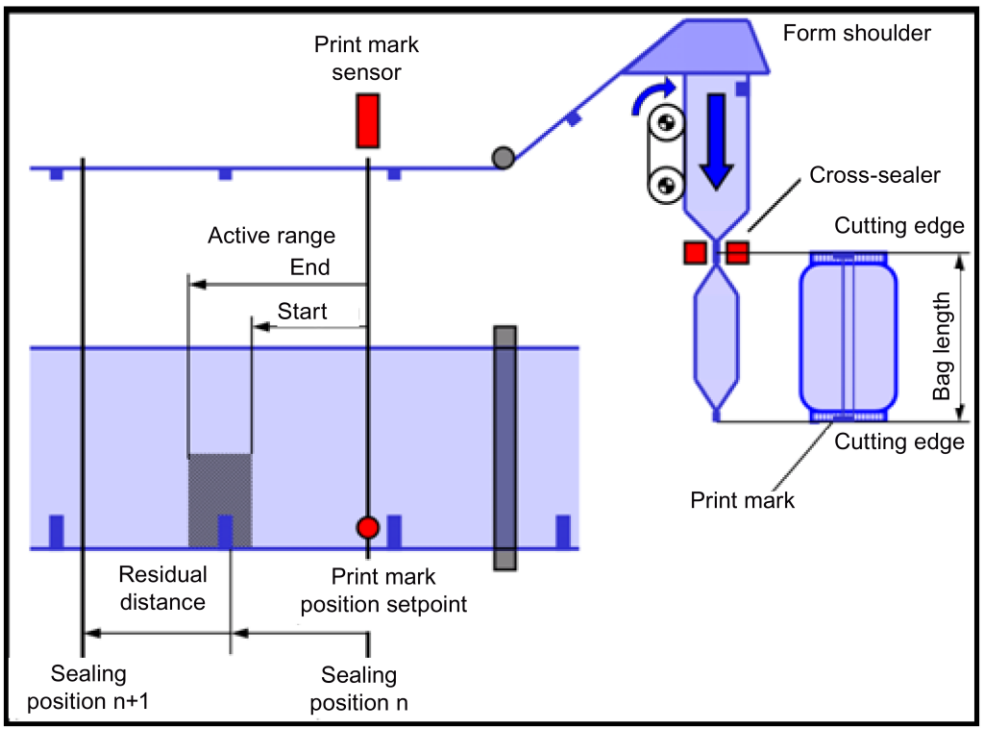
The following section shows a scenario for a possible application of the LPrintMark library:

Scenario

In flow wrapping machines, print marks are used to adjust the seal or rather the cut position of the foil. Because of e.g. slippage between the feeding unit and the packaging material, the feeded material length differs from the specified feeding length.

The position of a print mark is registered with a measuring input (measuring probe). The measured position is compared with a specified setpoint position. The resulting difference is compensated e.g. with a superimposed motion.

Figure 1-2



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	Quantity	Alternative
1.	CPU 1513-1 PN	6ES7513-1AL00-0AB0	1	Other S7-1500 CPU with FW V2.0
2.	or CPU 1515T-2 PN	6ES7515-2TM01-0AB0	1	Other S7-1500T CPU with FW V2.0

Software

Table 1-2

No.	Component	Article number	Quantity
3.	Step7 Professional V14	6ES7822-1..04-..	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 LPrintMark library in the main and load memory.

Overall size

The overall size of all blocks of the LPrintMark library in the main memory is 4 Kbytes in the code work memory, 1 Kbyte in the data work memory and 60 Kbytes in the load memory. The technology objects need additional memory.

Size of the individual blocks

Table 1-3

Block	Symbol	Size in main memory [Kbytes]	Size in load memory [Kbytes]
FB 31200	LPrintMark_HandleMeasuringInput	3.5	51
FC 31200	LPrintMark_CalcMeasuredDifference	0.2	5
DB 31200	Inst LPrintMark_HandleMeasuringInput [1]	0.6	4.3

[1] Instance data blocks (prefix InstLPrintMark_) 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 LPrintMark 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 blocks of the LPrintMark library.

Table 2-1

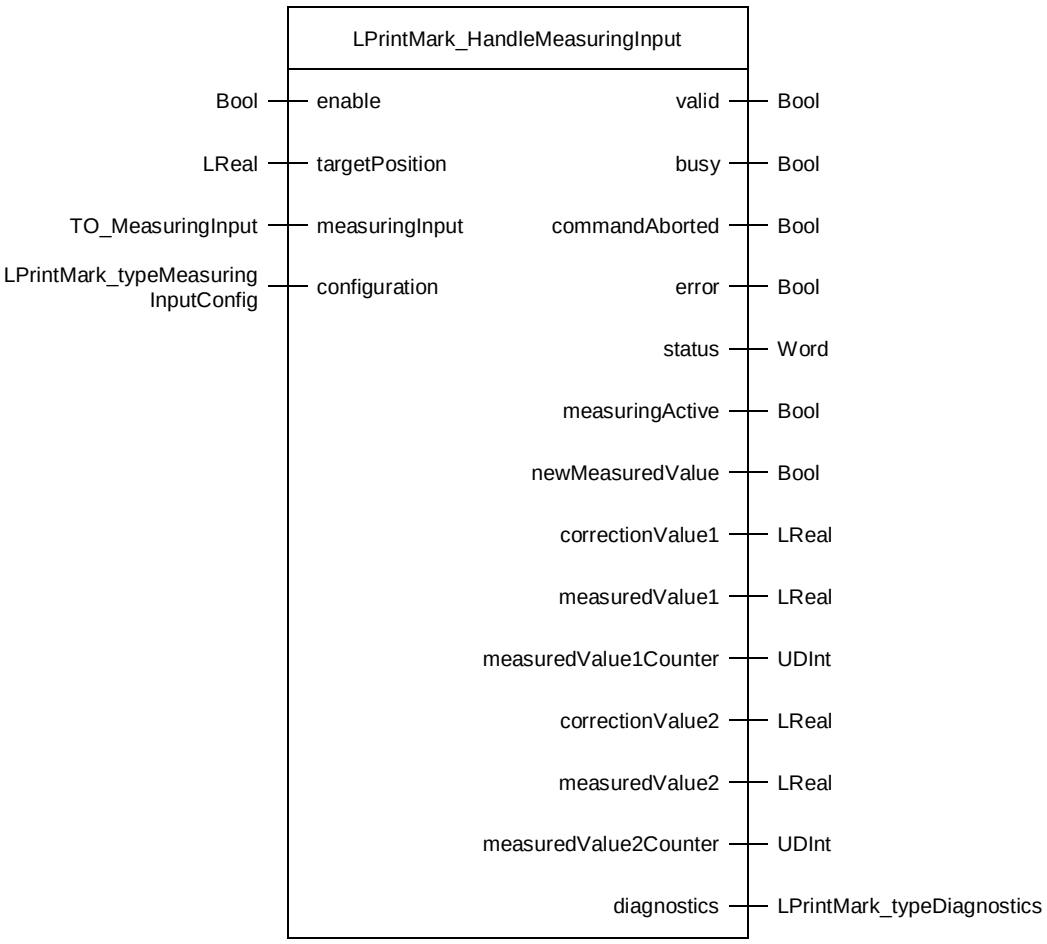
Block	Symbol	Classification
FB 31200	LPrintMark_HandleMeasuringInput	In-house development
FC 31200	LPrintMark_CalcMeasuredDifference	In-house development

2.2 Explanation of the blocks

The following table explains all blocks of the LPrintMark library.

2.2.1 FB LPrintMark_HandleMeasuringInput (FB 31200)

Figure



Principle of operation

Main functionalities of function block LPrintMark_HandleMeasuringInput:

- Start measuring jobs at technology object measuring input by means of the corresponding Motion Control instruction (MC_MeasuringInput, MC_MeasuringInputCyclic)
- Automatic restart of measuring jobs when the MC_MeasuringInput function block is internally active to get a cyclic measuring via application
- Calculation of the position deviation (specified target position - measured value). A maybe existent modulo range can be specified and is taken into account.

The configuration of the block is done via input parameter *configuration*. This data structure is made up of the following components:

- mode: Type of measurement (see *Mode* input of the used Motion Control instruction MC_MeasuringInput or MC_MeasuringInputCyclic)
- measuringRange: Acquisition of measured values (see *MeasuringRange* input of the used MC instruction MC_MeasuringInput or MC_MeasuringInputCyclic)
- startPosition: Start position of the measuring range (see *StartPosition* input of the used MC instruction MC_MeasuringInput or MC_MeasuringInputCyclic). Only relevant if *measuringRange* = *TRUE*.
- endPosition: End position of the measuring range (see *EndPosition* input of the used MC instruction MC_MeasuringInput or MC_MeasuringInputCyclic). Only relevant if *measuringRange* = *TRUE*.
- moduloLength: Modulo length of the assigned axis or external encoder (to be taken into account for the calculation of the position deviation *correctionValue*). Only relevant if the modulo function is active at the axis or external encoder.
- commandType: The internally used MC instruction (MC_MeasuringInput or MC_MeasuringInputCyclic) is determined with this parameter. The following options are available:
  - LPRINTMARK_CMD_TYPE_AUTO_DETECT (0)

    First of all the block tries to start measuring via the Motion Control instruction MC_MeasuringInputCyclic. If this instruction is not successful because the input type of the corresponding measuring input does not support cyclic measuring, the block automatically starts measuring with the Motion Control instruction MC_MeasuringInput (one-time measuring). The MC_MeasuringInput instruction is also automatically restarted when completed, i.e. cyclic measuring via application is implemented. Measuring events will not be acquired until the new restarted one-time measuring job is active.
  - LPRINTMARK_CMD_TYPE_USE_ONE_TIME (1) default

    The Motion Control instruction MC_MeasuringInput (one-time measuring) is used for measuring. The MC_MeasuringInput instruction is also automatically restarted when completed, i.e. cyclic measuring via application is implemented. Measuring events will not be acquired until the new restarted one-time measuring job is active.
  - LPRINTMARK_CMD_TYPE_USE_CYCLIC (2)

    The Motion Control instruction MC_MeasuringInputCyclic (cyclic measuring) is used for measuring. The input type of the corresponding measuring input must support cyclic measuring. Otherwise an error will be thrown.

Input parameters

Table 2-2

Parameter	Data type	Description
enable	Bool	TRUE: Enable functionality of FB (default: FALSE)
targetPosition	LReal	Target position (value must be within modulo range) (default: 0.0)
measuringInput	TO_Measuring Input	Technology object measuring input
configuration	LPrintMark_ typeMeasuring InputConfig	Structure for parameters

Output parameters

Table 2-3

Parameter	Data type	Description
valid	Bool	TRUE: Valid set of output values available at the FB (default: FALSE)
busy	Bool	TRUE: FB is working and new output values can be expected (default: FALSE)
commandAborted	Bool	TRUE: Internal called MC instruction has been aborted by external reason (default: FALSE)
error	Bool	TRUE: Rising edge informs that an error occurred during the execution of the FB (default: FALSE)
status	Word	Current status of FB (default: 16#7000, STATUS_NO_CALL)
measuringActive	Bool	TRUE: Technology object measuring input is waiting for trigger (default: FALSE)
newMeasuredValue	Bool	Rising edge: New measuring value available (default: FALSE)
correctionValue1	LReal	Correction value for first measured value: target position - measured value (default: 0.0)
measuredValue1	LReal	First measured value (default: 0.0)
measuredValue1 Counter	UDInt	Count value for the first measured value (default: 0)
correctionValue2	LReal	Correction value for second measured value target position - measured value (default: 0.0)
measuredValue2	LReal	Second measured value (default: 0.0)
measuredValue2 Counter	UDInt	Count value for the second measured value (default: 0)

Parameter	Data type	Description
diagnostics	LPrintMark_ typeDiagnostics	Diagnostics information of FB

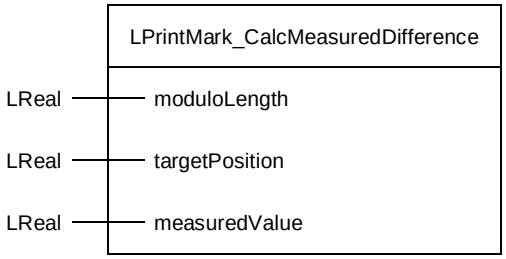
Status and error displays

Table 2-4

Status	Meaning	Remedy / notes
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_INVALID_COMMAND_TYPE	Value of parameter *configuration.commandType* is invalid/unknown
16#8600	ERR_MC_RESET	An error occurred while processing a MC_Reset instruction. The cause of the error (ErrorID of the MC instruction) can be found in *diagnostics.subfunctionStatus*.
16#8601	ERR_MC_MEASURINGINPUTCYCLIC	An error occurred while processing a MC_ MeasuringInputCyclic instruction. The cause of the error (ErrorID of the MC instruction) can be found in *diagnostics.subfunctionStatus*.
16#8602	ERR_MC_MEASURINGINPUT	An error occurred while processing a MC_ MeasuringInput instruction. The cause of the error (ErrorID of the MC instruction) can be found in *diagnostics.subfunctionStatus*.
16#8603	ERR_MC_ABORTMEASURINGINPUT	An error occurred while processing a MC_ AbortMeasuringInput instruction. The cause of the error (ErrorID of the MC instruction) can be found in *diagnostics.subfunctionStatus*.
16#8604	ERR_UNDEFINED_STATE	Error due to an undefined FB state

2.2.2 FC LPrintMark_CalcMeasuredDifference (FC 31200)

Figure



Principle of operation

The function LPrintMark_CalcMeasuredDifference calculates the position deviation between a target position and a measured value. A modulo range is also taken into account if input *moduloLength* is unequal to 0.0.

Input parameters

Table 2-5

Parameter	Data type	Description
moduloLength	LReal	Modulo length, in case of no modulo (axis is not a modulo axis) the input value must be 0.0
targetPosition	LReal	Target position, used for calculation of difference between *targetPosition* and *measuredValue*
measuredValue	LReal	Measured value

Return value

Table 2-6

Data type	Description
LReal	Difference between *targetPosition* and *measuredValue*

2.2.3 PLC data types

LPrintMark_typeMeasuringInputConfig

Table 2-7: Parameter of *LPrintMark_typeMeasuringInputConfig*

Parameter	Data type	Description
mode	DInt	Measuring mode for MC_MeasuringInput/MC_MeasuringInputCyclic (depending on *commandType*) 0: Measurement of positive edges 1: Measurement of negative edges (see system manual/help for further values) (default: 0)
measuringRange	Bool	Specific measuring range FALSE: Always acquire measured values TRUE: Acquire measured values only within the measuring range (default: FALSE)
startPosition	LReal	Start position of the measuring range (only relevant if *measuringRange = TRUE*) (default: 0.0)
endPosition	LReal	End position of the measuring range (only relevant if *measuringRange = TRUE*) (default: 0.0)
moduloLength	LReal	Modulo length, in case of no modulo (axis is not a modulo axis) the parameter must be 0.0 (default: 0.0)
commandType	DInt	The internally used Motion Control instruction is determined with this parameter 0: Automatic detection 1: Use one-time measuring (MC_MeasuringInput) 2: Use cyclic measuring (MC_MeasuringInputCyclic) (default: 1, LPRINTMARK_CMD_TYPE_USE_ONE_TIME)

LPrintMark_typeDiagnostics

Table 2-8: Parameter of *LPrintMark_typeDiagnostics*

Parameter	Data type	Description
status	Word	Status of FB when error occurred
subfunctionStatus	Word	ErrorID of called subfunction
state	DInt	Current state when error occurred

2.2.4 PLC tags

LPrintMark_Common

Table 2-9: User constants in *LPrintMark_Common*

Name	Data type	Value	Comment
LPRINTMARK_CMD_TYPE_AUTO_DETECT	DInt	0	Detect automatically which Motion Control instruction fits to the measuring input
LPRINTMARK_CMD_TYPE_USE_ONE_TIME	DInt	1	Use Motion Control instruction MC_MeasuringInput
LPRINTMARK_CMD_TYPE_USE_CYCLIC	DInt	2	Use Motion Control instruction MC_MeasuringInputCyclic

3 Working with the Library

What will you find in this section?

This chapter consists of instructions for integrating the LPrintMark 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 LPrintMark library into your STEP 7 project. Subsequently, you can use the blocks of the LPrintMark 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 LPrintMark_V1_x_x.zip to a local folder.
2.	In TIA Portal select “Options” -> “Global libraries” -> “Open library…”.
3.	Browse to the file LPrintMark.al14.
4.	It can be found in the subfolder LPrintMark of the extracted zip file.
5.	Open the global library in read-only mode.
6.	The LPrintMark 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 LPrintMark library into your STEP 7 program.

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

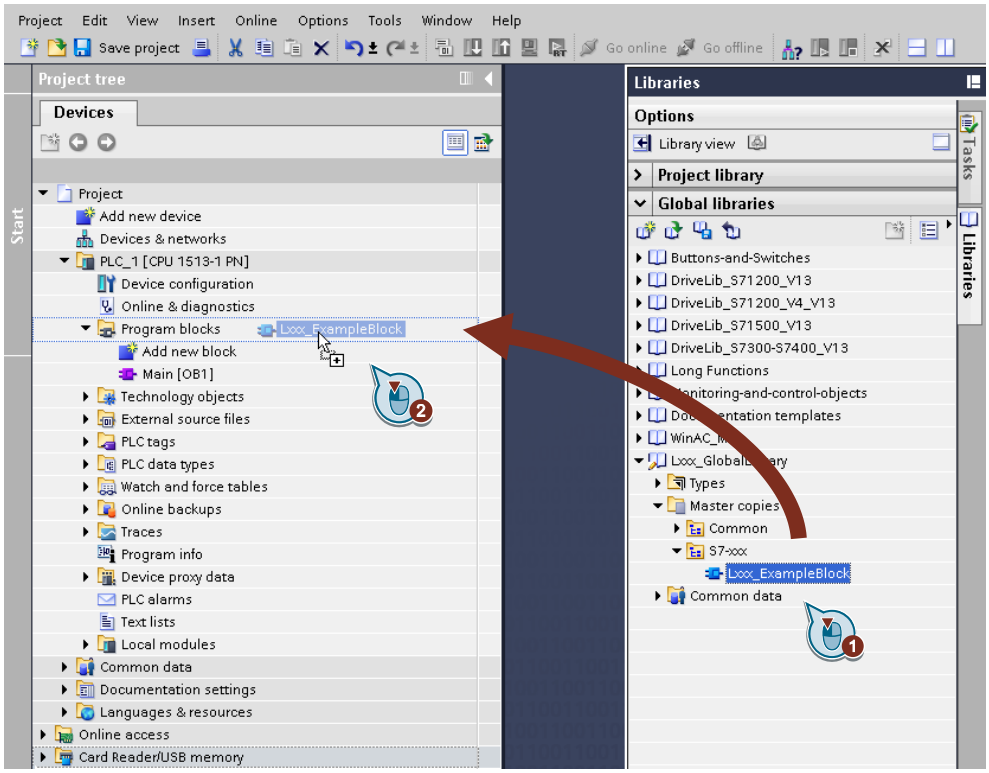


Table 3-2: Integrating the library blocks into STEP 7

No.	Action
1.	Copy the folder *LPrintMark_Tags* with Drag & Drop into the “PLC tags” in the PLC.
2.	Copy the folder *LPrintMark_Types* with Drag & Drop into the “PLC data types” in the PLC.
3.	Copy the folder *LPrintMark_Blocks* with Drag & Drop into the “Program blocks” in the PLC.
4.	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 LPrintMark library.

5 Related literature

Table 5-1

	Topic
\1\	Siemens Industry Online Support https://support.industry.siemens.com
\2\	Download page of this entry https://support.industry.siemens.com/cs/ww/en/view/109475573
\3\	SIMATIC S7-1500 Motion Control V3.0 in the TIA Portal V14 https://support.industry.siemens.com/cs/ww/en/view/109739589
\4\	SIMATIC S7-1500T Motion Control V3.0 in the TIA Portal V14 https://support.industry.siemens.com/cs/ww/en/view/109481326

6 Contact

Siemens AG

Digital Factory Division
Factory Automation
Production Machines
DF FA PMA APC
Frauenauracher Str. 80
91056 Erlangen, Germany

mailto: tech.team.motioncontrol@siemens.com

7 History

Table 7-1

Version	Date	Modifications
V1.0	10/2016	First version