Siemens
Industry
Online
Support

APPLICATION EXAMPLE

Simulating Telegrams 39x in SIMIT

SIMIT V10.3 HF2 / PLCSIM Adv. V4.0 SP1 HF1 / TIA V17

SIEMENS

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 under https://www.siemens.com/cert.

Table of contents

1. Introduction 5
1.1. Application description 5
1.2. Overview 5
1.3. Components used 5

2. Components 6
2.1. Telegram 390 6
2.1.1. PLC connection 6
2.1.2. Input, Output and Parameter overview 9
2.2. Telegram 391 10
2.2.1. PLC connection 11
2.2.2. Parameterization 13
2.2.3. Working principle 15
2.2.4. Input, Output and Parameter overview 16
2.3. Telegram 392 18
2.3.1. PLC connection 19
2.3.2. Parameterization 21
2.3.3. Working principle 23
2.3.4. Input, Output and Parameter overview 24
2.4. Telegram 393 27
2.4.1. PLC connection 28
2.4.2. Parameterization 30
2.4.3. Working principle 32
2.4.4. Input, Output and Parameter overview 33

3. Additional Components 36
3.1. Sensor High Precision 36
3.1.1. Basic settings 36
3.1.2. Working principle 36
3.1.3. Input, Output and Parameter overview 37

4. Example project 39
4.1. Telegram 390 40
4.2. Telegram 391 41
4.3. Telegram 392 43
4.4. Telegram 393 45

5. Appendix 47
5.1. Service and support 47
5.2. Industry Mall 48

5.3. Application support....48
5.4. Links and literature....48
5.5. Change documentation....48

1. Introduction

1.1. Application description

This package includes SIMIT simulation components to simulate the behavior of drive telegrams 390, 391, 392 and 393. Measuring inputs can be used with telegrams 391, 392 or 393.

1.2. Overview

Following components (see Table 1-1) can be used for the simulation:

Table 1-1 Provided simulation components

Component	Function
Telegrams 39x	
Telegram 390	Behavior simulation of drive *telegram 390*
Telegram 391	Behavior simulation of drive *telegram 391*
Telegram 392	Behavior simulation of drive *telegram 392*
Telegram 393	Behavior simulation of drive *telegram 393*
Additional Components	
Sensor High Precision	Generate sensor signals and time stamps for telegrams 39x

1.3. Components used

This application has been created with the following hardware and software components (see Table 1-2):

Table 1-2 Application example components

Component	Article number
SIMIT M V10.3 HF2	6DL8913-0BK30-0AB5
PLCSIM Advanced 4.0 SP1 HF1	6ES7823-1FA01-0YA5
STEP 7 Professional V17	6ES7822-1..07-..

This application example consists of the following components (see Table 1-3):

Table 1-3 Supplied components

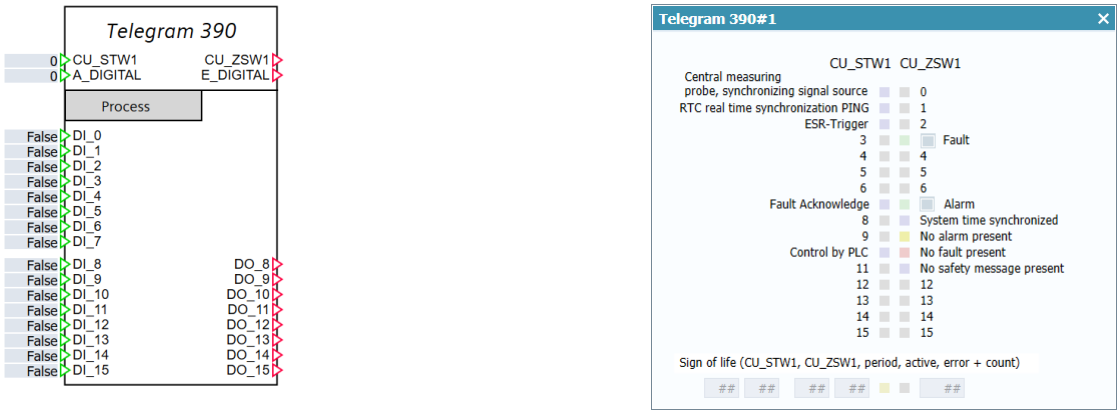
Component	File name
Documentation	Tel39x_SIMIT_Manual_V1.0.0.pdf
SIMIT Tel39x Components	Tel39x_SIMIT_V1.0.0.zip
SIMIT Tel39x Demo Project	Tel39x_V1.0.0.zip

2. Components

2.1. Telegram 390

This component represents the telegram 390 (see Figure 2-1).

Figure 2-1 *Telegram 390* - SIMIT component and operating window



Simulation requirements

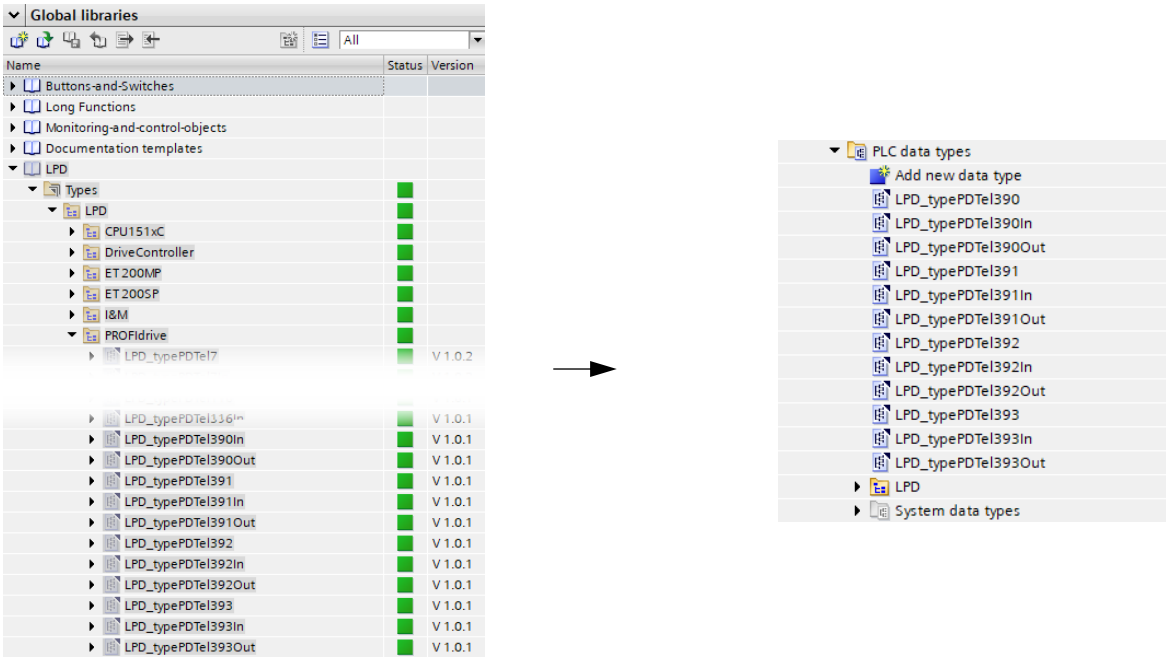
- Simulation must be set to bus- or event-synchronous mode
-

2.1.1. PLC connection

PLC data types

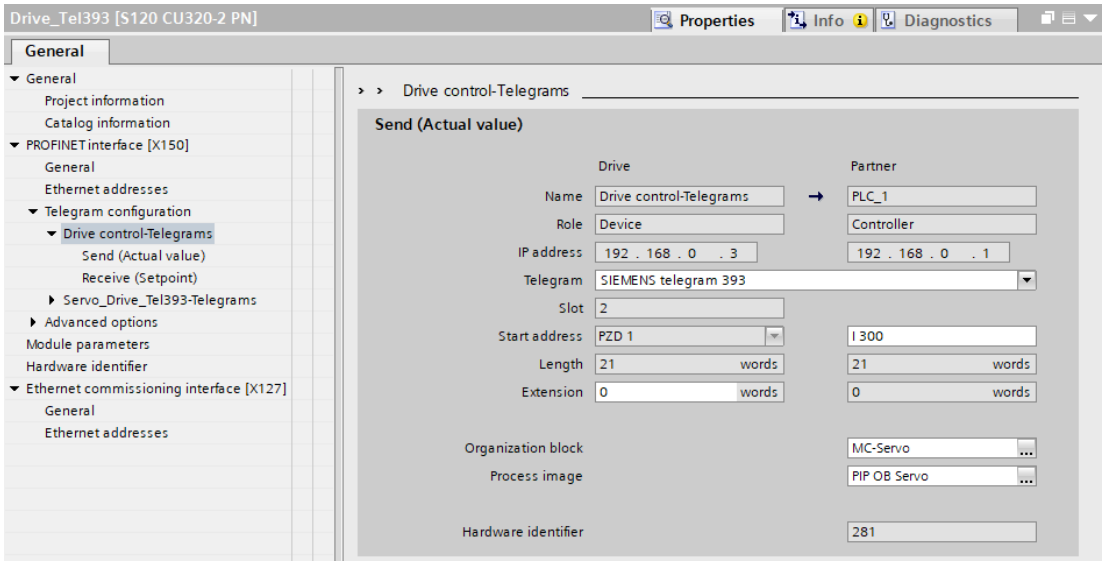
Connecting the simulation block in SIMIT with the relevant PLC in- and outputs requires adding the LPD library in the TIA Portal project \3\. After adding the library (see Figure 2-2), new data types are available which must be dragged and dropped to the *PLC data types* of the project.

Figure 2-2 LPD Library included in the TIA Portal project – Telegram 39x data types



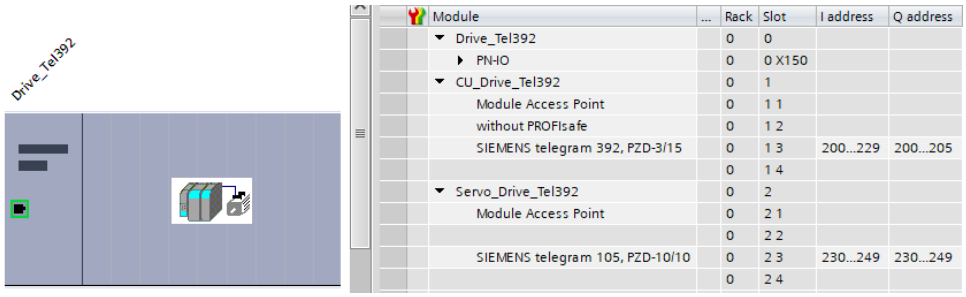
Note down the I and Q start-addresses of the telegrams 39x which can be found in the hardware configuration. The addresses of the telegrams in a drive control, configured via SINAMCIS Startdrive, can be found in the properties of the specific drive (see Figure 2-3).

Figure 2-3 I and Q addresses of *Telegram 393 – drive control (Startdrive)*



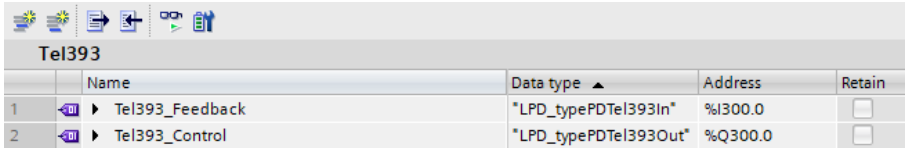
The addresses of the telegrams 39x, configured as GSDML, can directly be found in the hardware configuration of the drive (see Figure 2-4).

Figure 2-4 I and Q addresses of *Telegram 392 – CU (GSDML)*



In the PLC tags now the addresses of the telegrams must be added manually. If this step is skipped, the signals won't appear in the SIMIT coupling. Add a new tag with the starting address of the specific telegram and select the relevant PLC data type (see Figure 2-5). The structure will be created automatically.

Figure 2-5 PLC tags with LPD library data types for *Telegram 393*

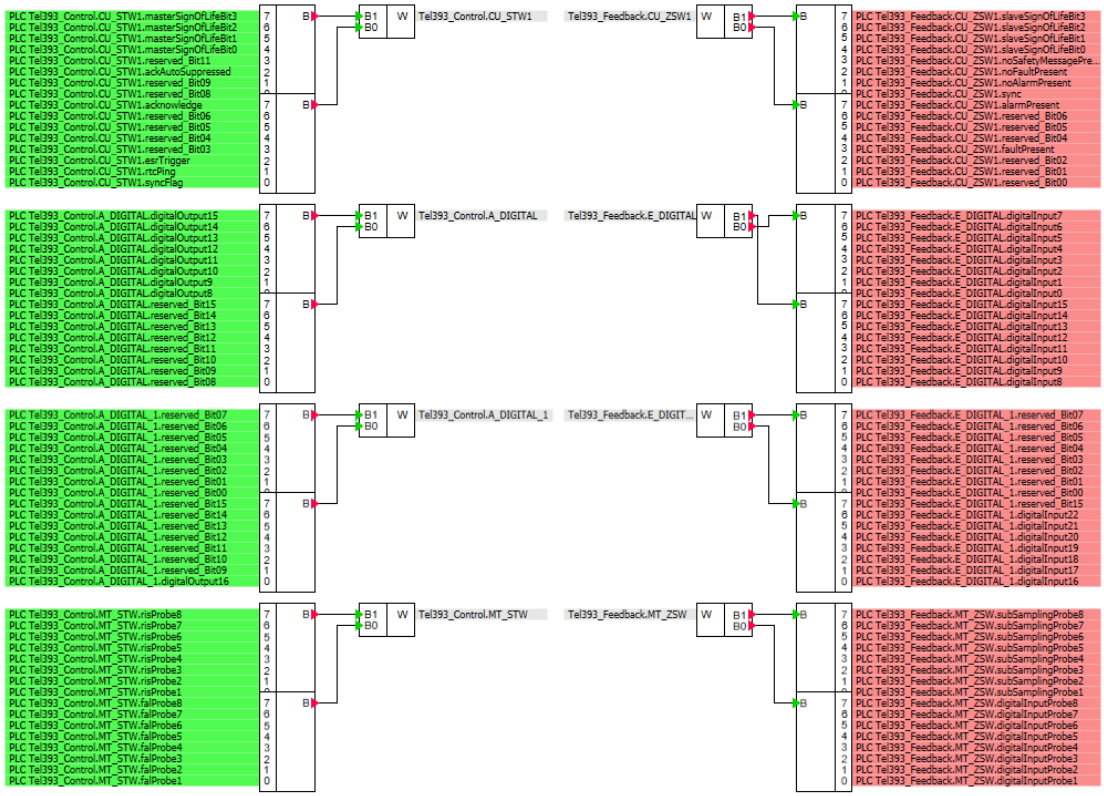


Connecting signals in SIMIT

After importing the coupling, all relevant signals for the telegrams can be found in SIMIT. Connect all signals with the simulation block in SIMIT. It is helpful to use word and byte splitters and mergers to connect the signals (see Figure 2-6).

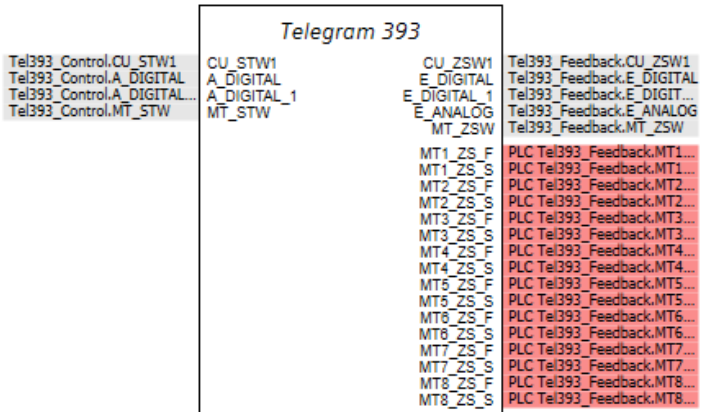
NOTE Listing of signal names in the PLC begin with number 0 as suffix. SIMIT arrays always start with number 1. Keep that mismatch in mind to do a correct signal mapping (see Figure 2-6).

Figure 2-6 Merging and splitting PLC signals for the telegrams 393



Now the signals can be connected to the simulation block (see Figure 2-7).

Figure 2-7 Connecting the merged signals with the telegram 393



2.1.2. Input, Output and Parameter overview

Inputs

Table 2-1 *Telegram 390* - Inputs

Inputs	Data Type	Description
PLC signals		
CU_STW1	integer	Control Word of the control unit
A_DIGITAL	integer	Digital outputs
Process signals		
DI_[16]	binary	Digital input is triggered

Outputs

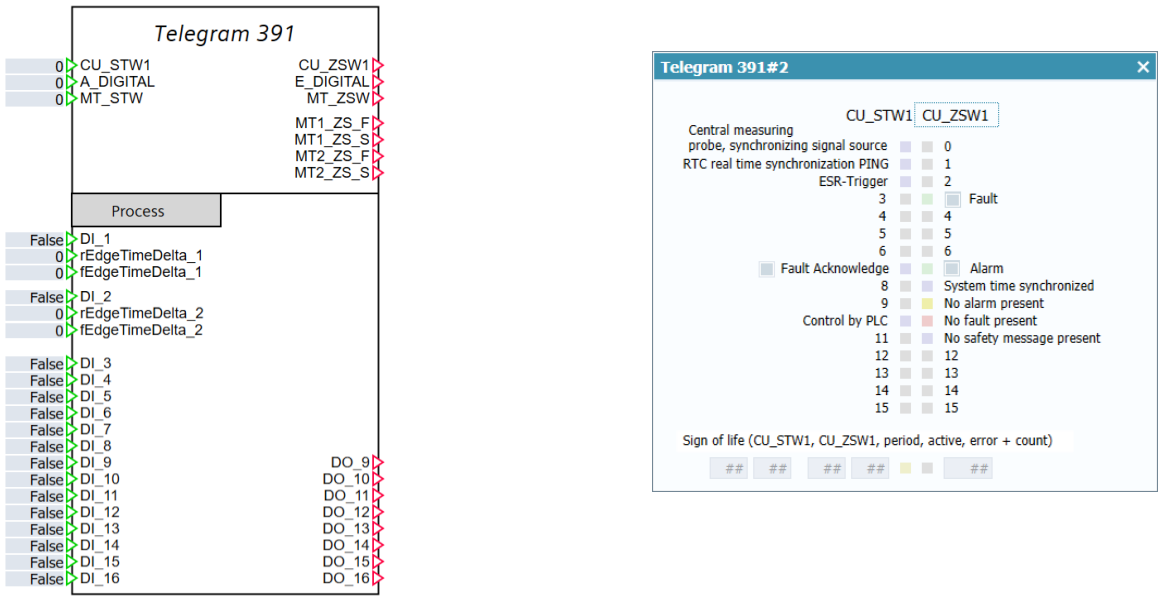
Table 2-2 *Telegram 390* - Outputs

Outputs	Data Type	Description
PLC signals		
CU_STW1	integer	Status Word of the control unit
E_DIGITAL	integer	Digital inputs
Process signals		
DO_[8]	binary	Digital output is set

2.2. Telegram 391

This component represents the telegram 391 (see Figure 2-8). It is used for the Measuring *Input* technology object with the *MC_MeasuringInput* function block in the PLC.

Figure 2-8 *Telegram 391* - SIMIT component and operating window



Simulation requirements

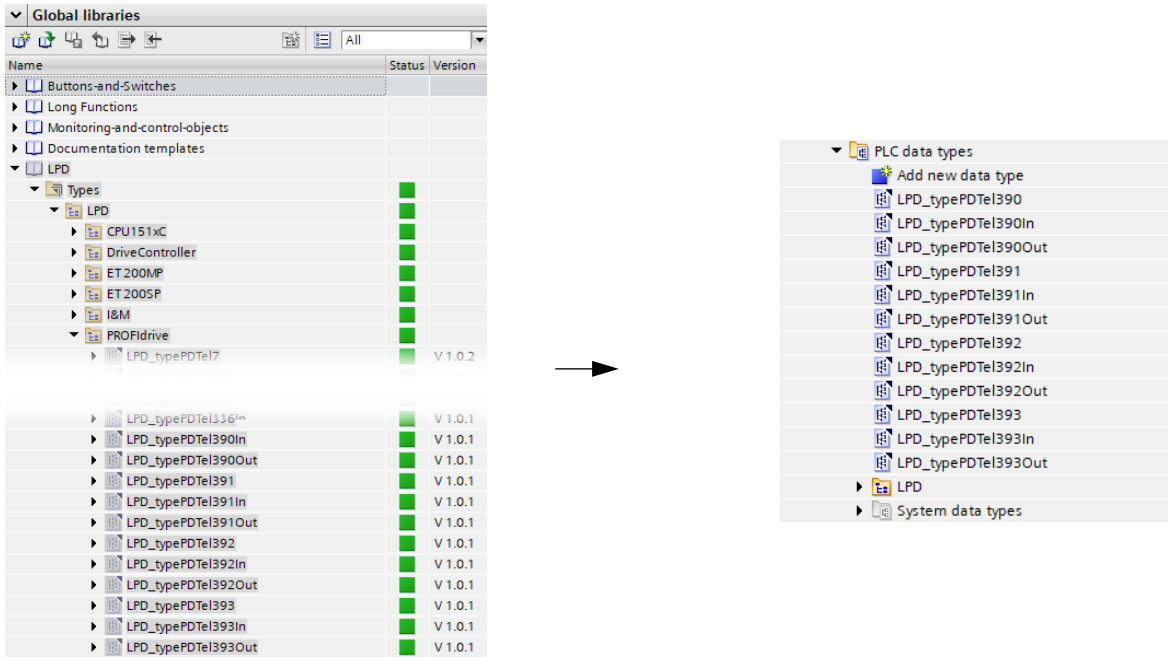
- Simulation must be set to bus- or event-synchronous mode

2.2.1. PLC connection

PLC data types

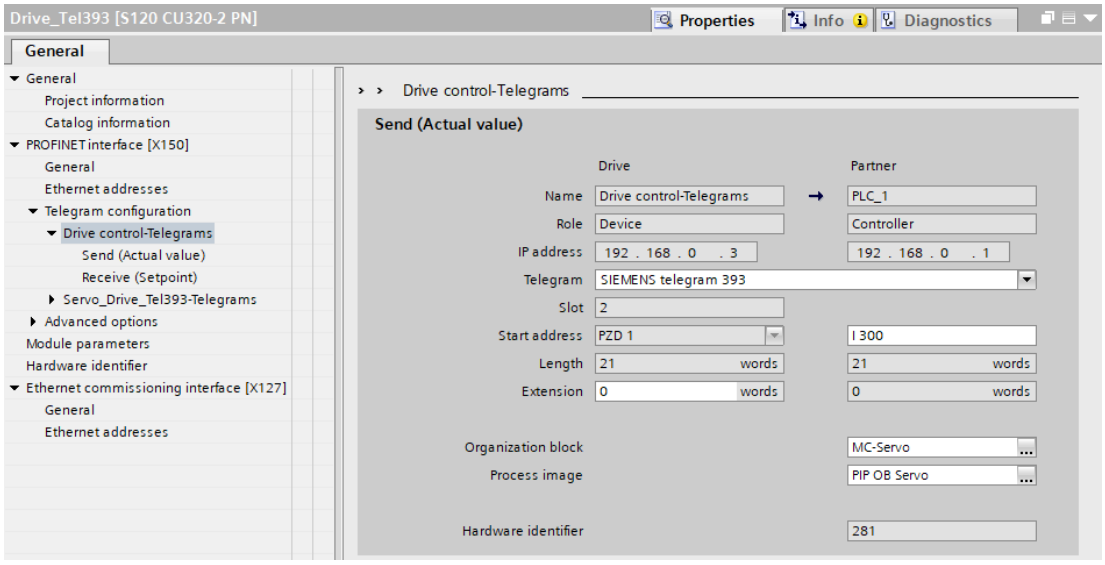
Connecting the simulation block in SIMIT with the relevant PLC in- and outputs requires adding the LPD library in the TIA Portal project \3\. After adding the library (see Figure 2-9), new data types are available which must be dragged and dropped to the *PLC data types* of the project.

Figure 2-9 LPD Library included in the TIA Portal project – Telegram 39x data types



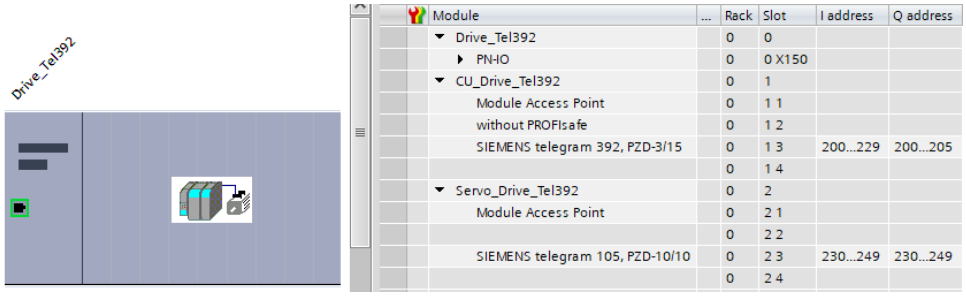
Note down the I and Q start-addresses of the telegrams 39x which can be found in the hardware configuration. The addresses of the telegrams in a drive control, configured via SINAMCIS Startdrive, can be found in the properties of the specific drive (see Figure 2-10).

Figure 2-10 I and Q addresses of *Telegram 393 – drive control (Startdrive)*



The addresses of the telegrams 39x, configured as GSDML, can directly be found in the hardware configuration of the drive (see Figure 2-11).

Figure 2-11 I and Q addresses of *Telegram 392 – CU* (*GSDML*)



In the PLC tags now the addresses of the telegrams must be added manually. If this step is skipped, the signals won't appear in the SIMIT coupling. Add a new tag with the starting address of the specific telegram and select the relevant PLC data type (see Figure 2-12). The structure will be created automatically.

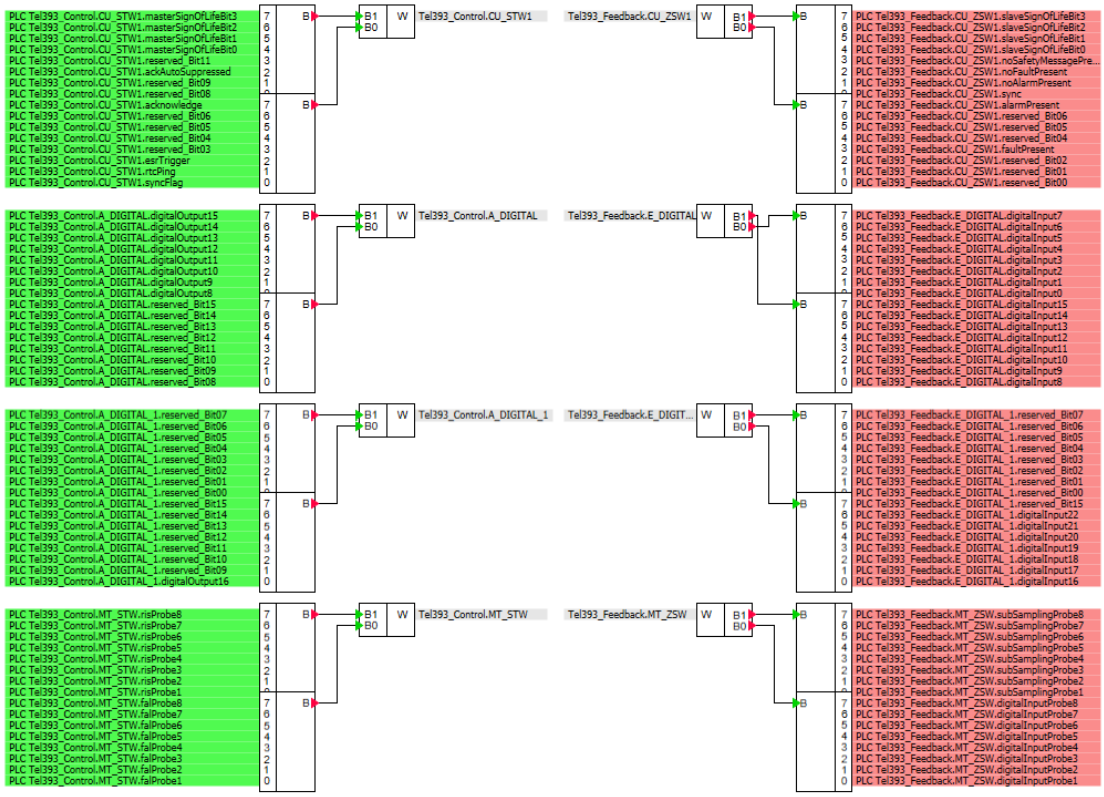
Figure 2-12 PLC tags with LPD library data types for *Telegram 393*

Connecting signals in SIMIT

After importing the coupling, all relevant signals for the telegrams can be found in SIMIT. Connect all signals with the simulation block in SIMIT. It is helpful to use word and byte splitters and mergers to connect the signals (see Figure 2-13).

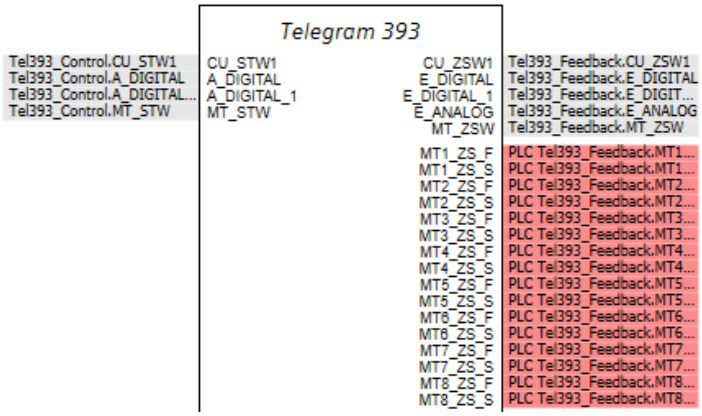
NOTE Listing of signal names in the PLC begin with number 0 as suffix. SIMIT arrays always start with number 1. Keep that mismatch in mind to do a correct signal mapping (see Figure 2-13).

Figure 2-13 Merging and splitting PLC signals for the telegrams 393



Now the signals can be connected to the simulation block (see Figure 2-14).

Figure 2-14 Connecting the merged signals with the telegram 393

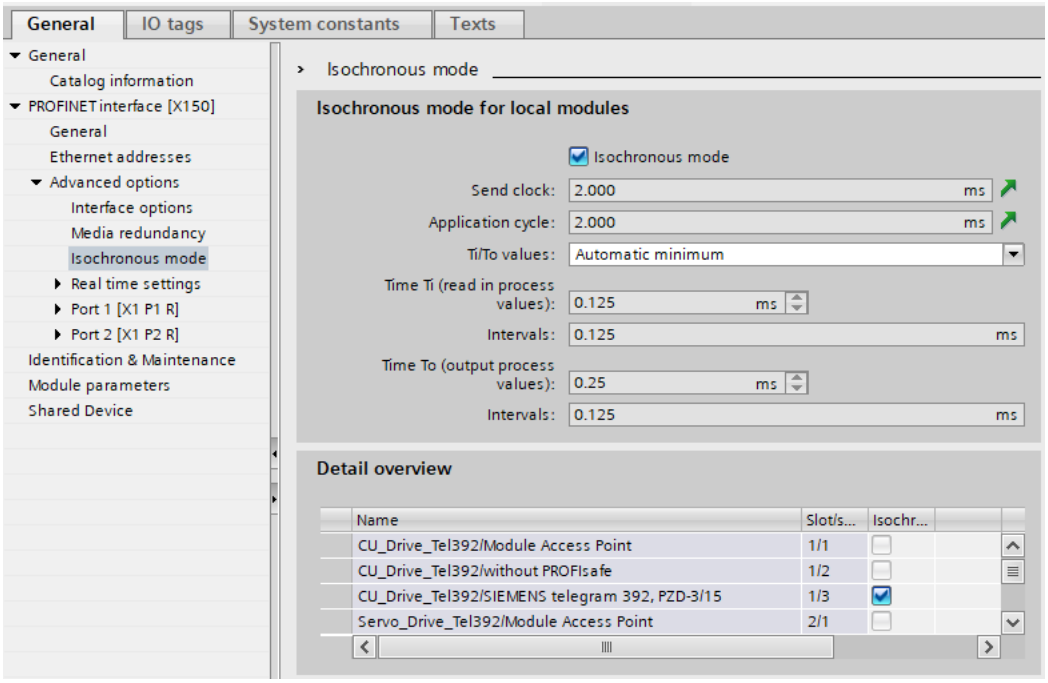


2.2.2. Parameterization

The parameters *Time_Ti*, *Time_To and Time_InternalControlLoop* of the telegram 39x components must be set equally to the PLC setup (in µs). The times of *Ti* and *To* can be found in the properties of the control unit in TIA Portal. Relevant

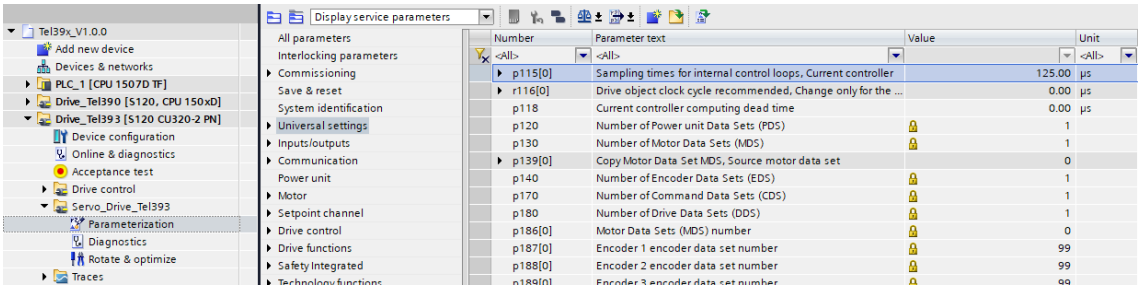
information can be found in the options of the Isochronous Mode (see Figure 2-15). Use the values of the *Intervals* below the *Time Ti* and *Time To* values.

Figure 2-15 Isochronous mode settings for Time_Ti and Time_To



If SINAMICS Startdrive is used, the *Time_InternalControlLoop* time can be found in the parameters of the servo axis. Minimally the extended parameters must be displayed to view parameter *p115* (see Figure 2-16).

Figure 2-16 Sampling times for internal control loops in p115 (Startdrive)



For drives which were configured in SINAMICS *STARTER* it can be useful to try the *Ti* or *To* time for the correct value of *Time_InternalControlLoop*.

NOTE

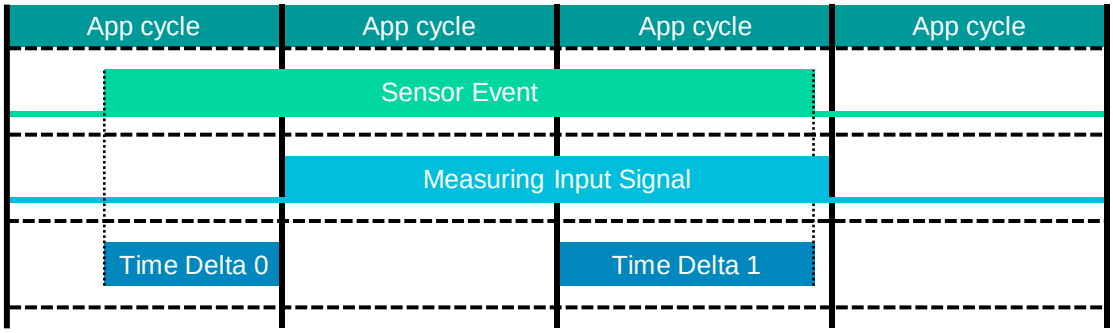
If resulting measured position values are incorrect use the secondary parameter of the telegram 39x component *Time_Offset_Simulation* to correct and fine tune the detected position. Sometimes it might be necessary to add the cycle time in µs once.

2.2.3. Working principle

Rising and falling edges of a *Sensor Event* normally occur during the application cycles. As a result, the *Measuring Input Signal* is sent to the PLC in the application cycle with the last state of the *Sensor Event*.

The time offset (*Time Delta* in µs) to the application cycle is processed by telegram 39x and it is sent as timestamp to the PLC. With the combination of the *Measuring Input Signal* and the *Time Delta* value the PLC is able to reproduce the exact positions of the occurred sensor edges relative to an axis or encoder position (see Figure 2-17).

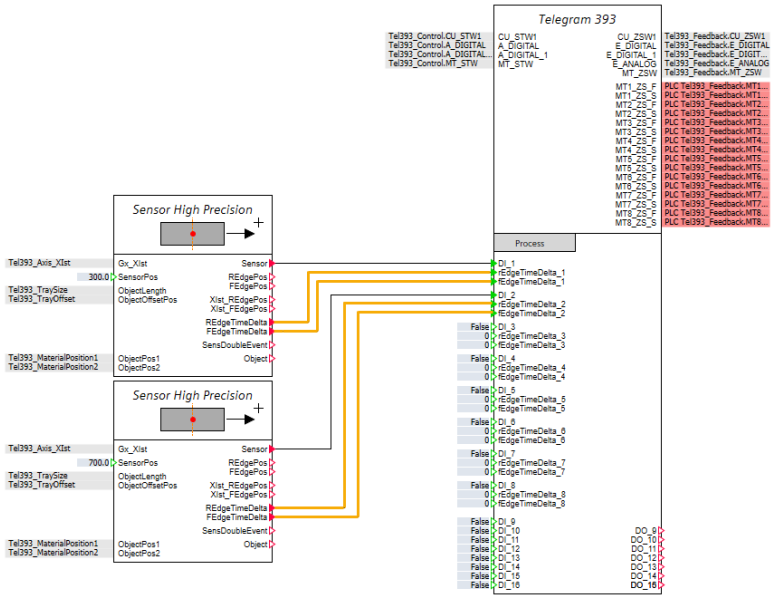
Figure 2-17 Telegram 39x measuring input cycle mechanism



Simulating exact time stamps

To simulate the exact time stamps for the *Measuring Input*, it is necessary to use and connect a Sensor High Precision simulation block to the telegram 39x components (see Figure 2-18).

Figure 2-18 *Sensor High Precision* connected to the *Telegram 393* component



NOTE

It is also possible to only use the binary *Sensor_Event* input (no connections of the yellow wires in Figure 2-18), however the measurement accuracy is reduced to the accuracy of the application cycle in the process.

2.2.4. Input, Output and Parameter overview

Inputs

Table 2-3 *Telegram 391* - Inputs

Inputs	Data Type	Description
PLC signals		
CU_STW1	integer	Control Word of the control unit
A_DIGITAL	integer	Digital outputs
MT_STW	integer	Control word of the measuring inputs
Process signals		
DI_[16]	binary	Digital input is triggered
rEdgeTimeDelta_[2]	integer	Rising edge time delta in µs to the application cycle of the sensor signal
fEdgeTimeDelta_[2]	integer	Falling edge time delta in µs to the application cycle of the sensor signal

Outputs

Table 2-4 *Telegram 391* - Outputs

Outputs	Data Type	Description
PLC signals		
CU_STW1	integer	Status Word of the control unit
E_DIGITAL	integer	Digital inputs
MT_ZSW	integer	Status Word of the measuring inputs
MT1_ZS_F	integer	Measuring input 1 falling edge time offset value
MT1_ZS_S	integer	Measuring input 1 rising edge time offset value
MT2_ZS_F	integer	Measuring input 2 falling edge time offset value
MT2_ZS_S	integer	Measuring input 2 rising edge time offset value
Process signals		
DO_[8]	binary	Digital output is set

Parameters

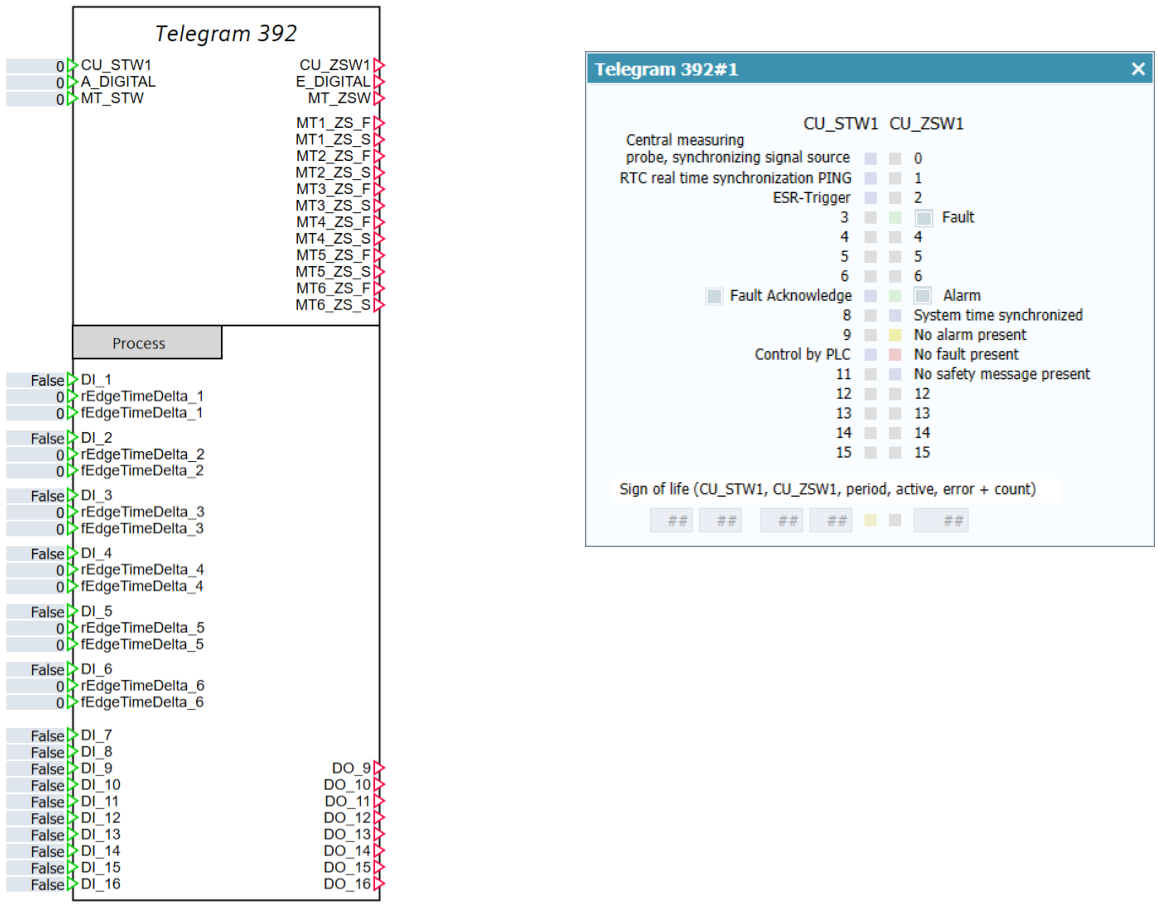
Table 2-5 *Telegram 391* - Parameters

Parameters	Data Type	Description
Parameters		
Time_Ti	integer	Read in process values time in µs
Time_To	integer	Output process values time in µs
Time_InternalControlLoop	integer	Cycle time of the internal control loop in µs
Additional parameters		
Time_Offset_Simulation	integer	Additional time delta in µs to correct the time offsets in the simulation

2.3. Telegram 392

This component represents the telegram 392 (Figure 2-19). It is used for the Measuring *Input* technology object with the *MC_MeasuringInput* function block in the PLC.

Figure 2-19 *Telegram 392* SIMIT component and operating window



Simulation requirements

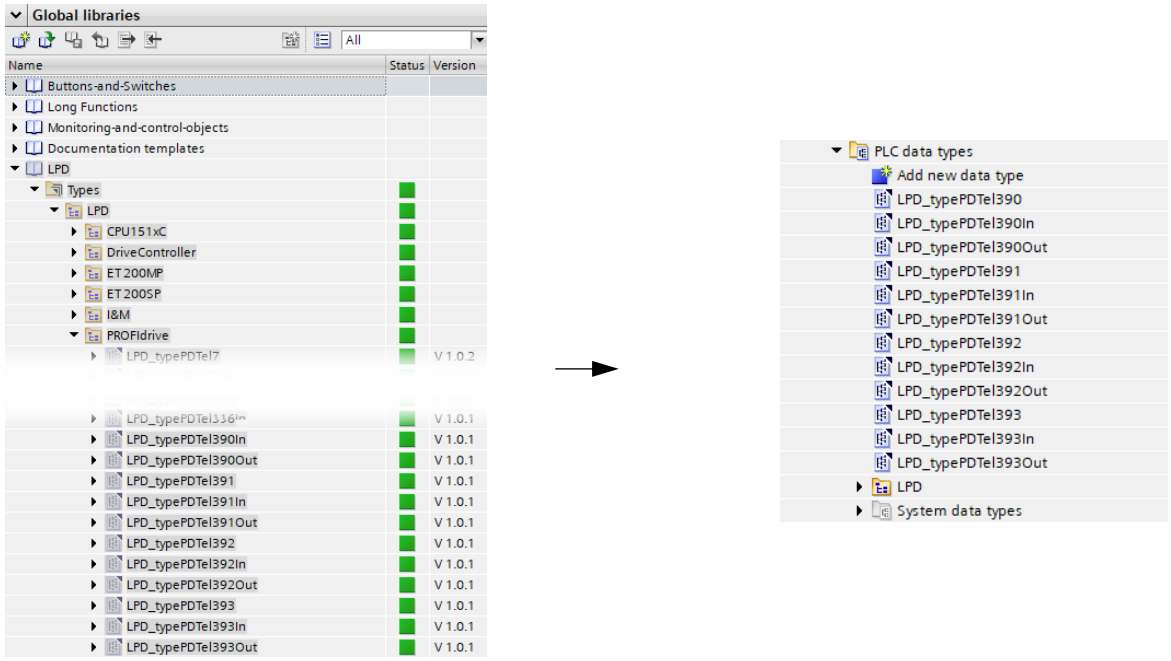
- Simulation must be set to bus- or event-synchronous mode

2.3.1. PLC connection

PLC data types

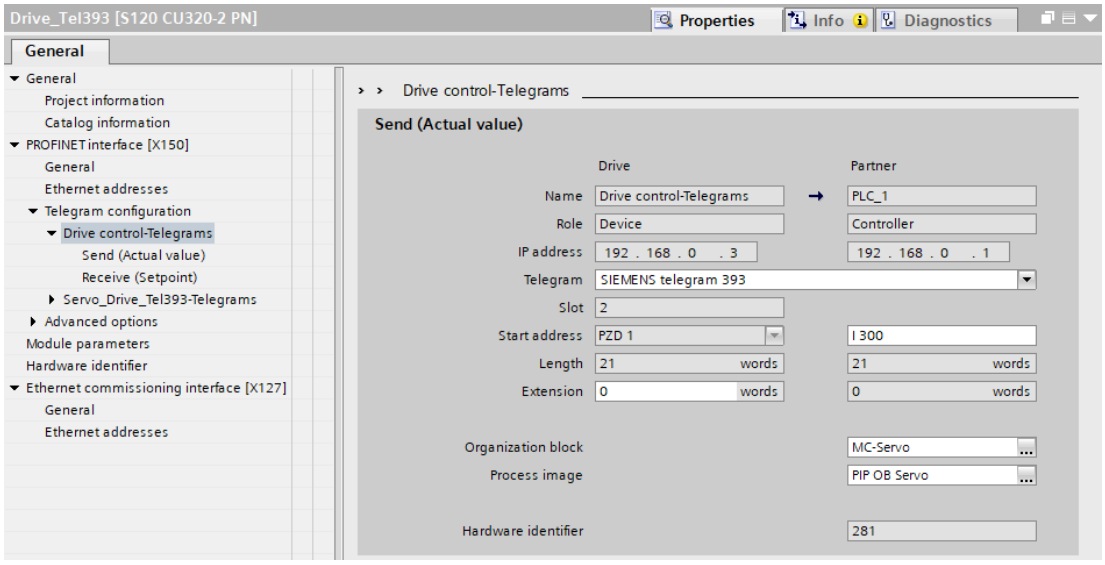
Connecting the simulation block in SIMIT with the relevant PLC in- and outputs requires adding the LPD library in the TIA Portal project \3\. After adding the library (see Figure 2-20), new data types are available which must be dragged and dropped to the *PLC data types* of the project.

Figure 2-20 LPD Library included in the TIA Portal project – Telegram 39x data types



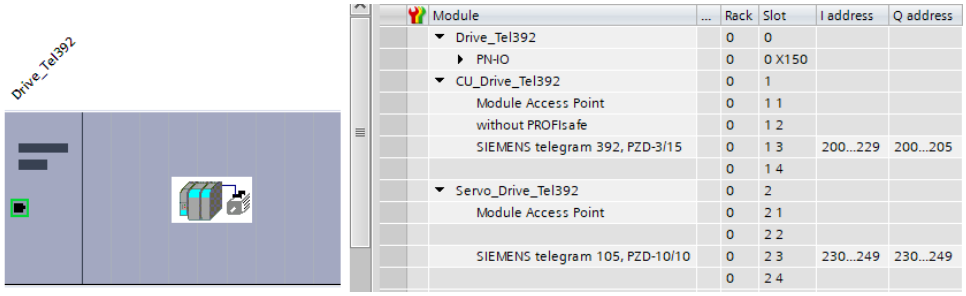
Note down the I and Q start-addresses of the telegrams 39x which can be found in the hardware configuration. The addresses of the telegrams in a drive control, configured via SINAMCIS Startdrive, can be found in the properties of the specific drive (see Figure 2-21).

Figure 2-21 I and Q addresses of *Telegram 393 – drive control (Startdrive)*



The addresses of the telegrams 39x, configured as GSDML, can directly be found in the hardware configuration of the drive (see Figure 2-22).

Figure 2-22 I and Q addresses of *Telegram 392 – CU (GSDML)*



In the PLC tags now the addresses of the telegrams must be added manually. If this step is skipped, the signals won't appear in the SIMIT coupling. Add a new tag with the starting address of the specific telegram and select the relevant PLC data type (see Figure 2-23). The structure will be created automatically.

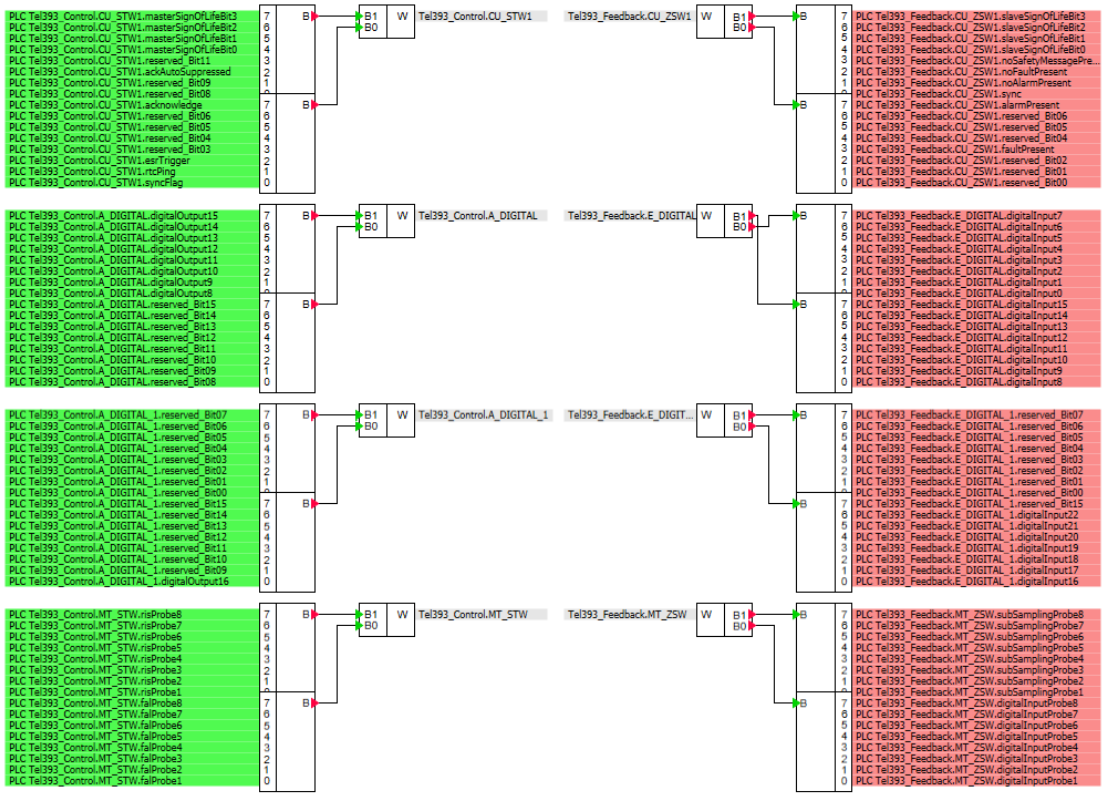
Figure 2-23 PLC tags with LPD library data types for *Telegram 393*

Connecting signals in SIMIT

After importing the coupling, all relevant signals for the telegrams can be found in SIMIT. Connect all signals with the simulation block in SIMIT. It is helpful to use word and byte splitters and mergers to connect the signals (see Figure 2-24).

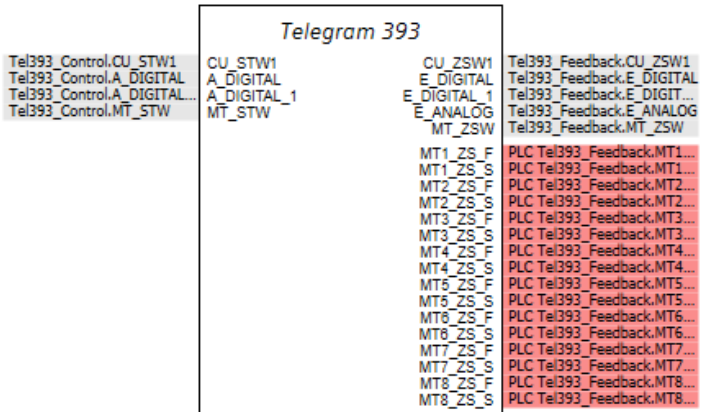
NOTE Listing of signal names in the PLC begin with number 0 as suffix. SIMIT arrays always start with number 1. Keep that mismatch in mind to do a correct signal mapping (see Figure 2-24).

Figure 2-24 Merging and splitting PLC signals for the telegrams 393



Now the signals can be connected to the simulation block (see Figure 2-25).

Figure 2-25 Connecting the merged signals with the telegram 393

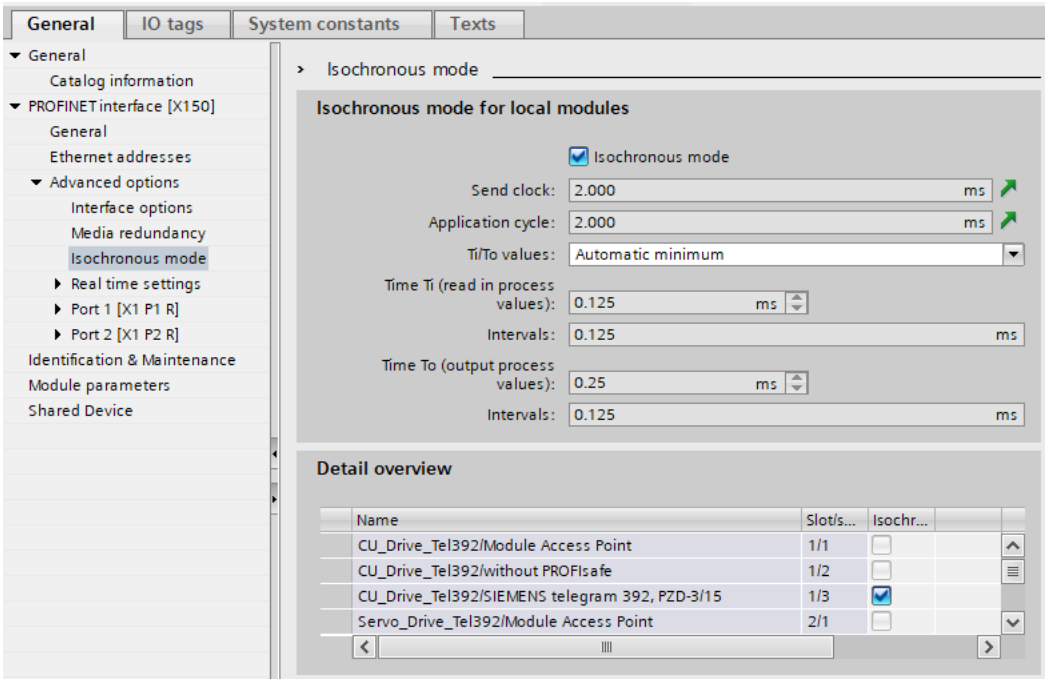


2.3.2. Parameterization

The parameters *Time_Ti*, *Time_To and Time_InternalControlLoop* of the telegram 39x components must be set equally to the PLC setup (in µs). The times of *Ti* and *To* can be found in the properties of the control unit in TIA Portal. Relevant

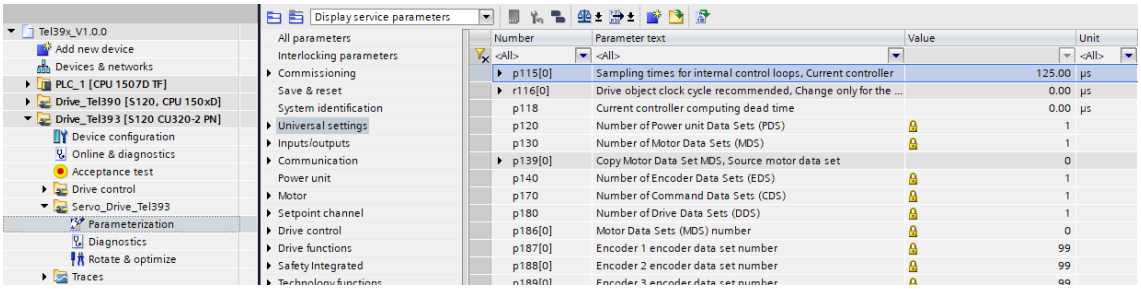
information can be found in the options of the Isochronous Mode (see Figure 2-26). Use the values of the *Intervals* below the *Time Ti* and *Time To* values.

Figure 2-26 Isochronous mode settings for Time_Ti and Time_To



If SINAMICS Startdrive is used, the *Time_InternalControlLoop* time can be found in the parameters of the servo axis. Minimally the extended parameters must be displayed to view parameter *p115* (see Figure 2-27).

Figure 2-27 Sampling times for internal control loops in p115 (Startdrive)



For drives which were configured in SINAMICS *STARTER* it can be useful to try the *Ti* or *To* time for the correct value of *Time_InternalControlLoop*.

NOTE

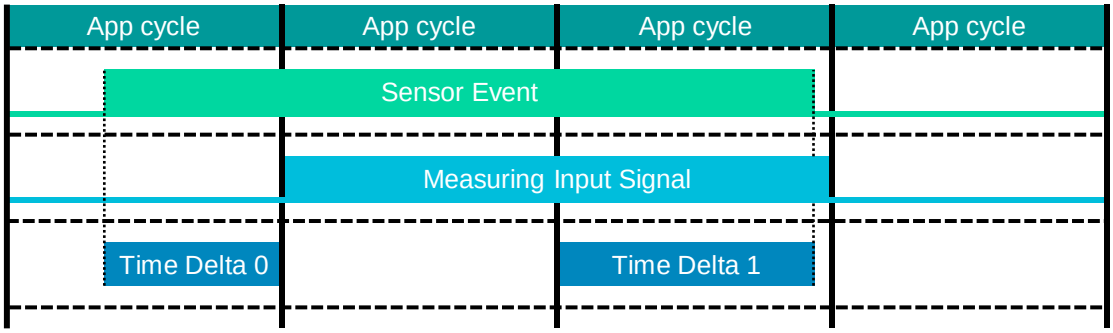
If resulting measured position values are incorrect use the secondary parameter of the telegram 39x component *Time_Offset_Simulation* to correct and fine tune the detected position. Sometimes it might be necessary to add the cycle time in µs once.

2.3.3. Working principle

Rising and falling edges of a *Sensor Event* normally occur during the application cycles. As a result, the *Measuring Input Signal* is sent to the PLC in the application cycle with the last state of the *Sensor Event*.

The time offset (*Time Delta* in µs) to the application cycle is processed by telegram 39x and it is sent as timestamp to the PLC. With the combination of the *Measuring Input Signal* and the *Time Delta* value the PLC is able to reproduce the exact positions of the occurred sensor edges relative to an axis or encoder position (see Figure 2-28).

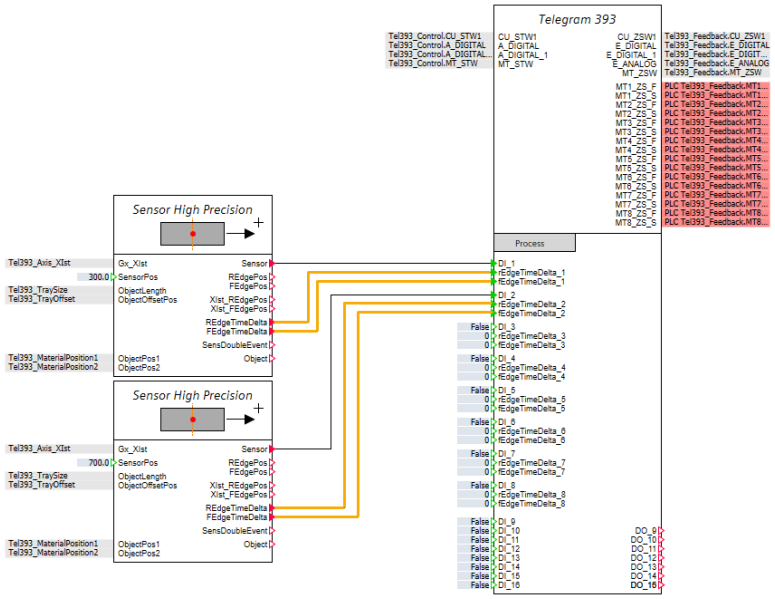
Figure 2-28 Telegram 39x measuring input cycle mechanism



Simulating exact time stamps

To simulate the exact time stamps for the *Measuring Input*, it is necessary to use and connect a *Sensor High Precision* simulation block to the telegram 39x components (see Figure 2-29).

Figure 2-29 *Sensor High Precision* connected to the *Telegram 393* component



NOTE

It is also possible to only use the binary *Sensor_Event* input (no connections of the yellow wires in Figure 2-29), however the measurement accuracy is reduced to the accuracy of the application cycle in the process.

2.3.4. Input, Output and Parameter overview

Inputs

Table 2-6 *Telegram 392* - Inputs

Inputs	Data Type	Description
PLC signals		
CU_STW1	integer	Control Word of the control unit
A_DIGITAL	integer	Digital outputs
MT_STW	integer	Control word of the measuring inputs
Process signals		
DI_[16]	binary	Digital input is triggered
rEdgeTimeDelta_[6]	integer	Rising edge time delta in µs to the application cycle of the sensor signal
fEdgeTimeDelta_[6]	integer	Falling edge time delta in µs to the application cycle of the sensor signal

Outputs

Table 2-7 *Telegram 392* - Outputs

Outputs	Data Type	Description
PLC signals		
CU_STW1	integer	Status Word of the control unit
E_DIGITAL	integer	Digital inputs
MT_ZSW	integer	Status Word of the measuring inputs
MT1_ZS_F	integer	Measuring input 1 falling edge time offset value
MT1_ZS_S	integer	Measuring input 1 rising edge time offset value
MT2_ZS_F	integer	Measuring input 2 falling edge time offset value
MT2_ZS_S	integer	Measuring input 2 rising edge time offset value
MT3_ZS_F	integer	Measuring input 3 falling edge time offset value
MT3_ZS_S	integer	Measuring input 3 rising edge time offset value
MT4_ZS_F	integer	Measuring input 4 falling edge time offset value
MT4_ZS_S	integer	Measuring input 4 rising edge time offset value
MT5_ZS_F	integer	Measuring input 5 falling edge time offset value
MT5_ZS_S	integer	Measuring input 5 rising edge time offset value
MT6_ZS_F	integer	Measuring input 6 falling edge time offset value
MT6_ZS_S	integer	Measuring input 6 rising edge time offset value
Process signals		
DO_[8]	binary	Digital output is set

Parameters

Table 2-8 *Telegram 392* - Parameters

Parameters	Data Type	Description
Parameters		
Time_Ti	integer	Read in process values time in µs
Time_To	integer	Output process values time in µs
Time_InternalControlLoop	integer	Cycle time of the internal control loop in µs
Additional parameters		
Time_Offset_Simulation	integer	Additional time delta in µs to correct the time offsets in the simulation

2.4. Telegram 393

This component represents the telegram 393 (see Figure 2-30). It is used for the Measuring *Input* technology object with the *MC_MeasuringInput* function block in the PLC.

Figure 2-30 *Telegram 393* SIMIT component and operating window

Simulation requirements

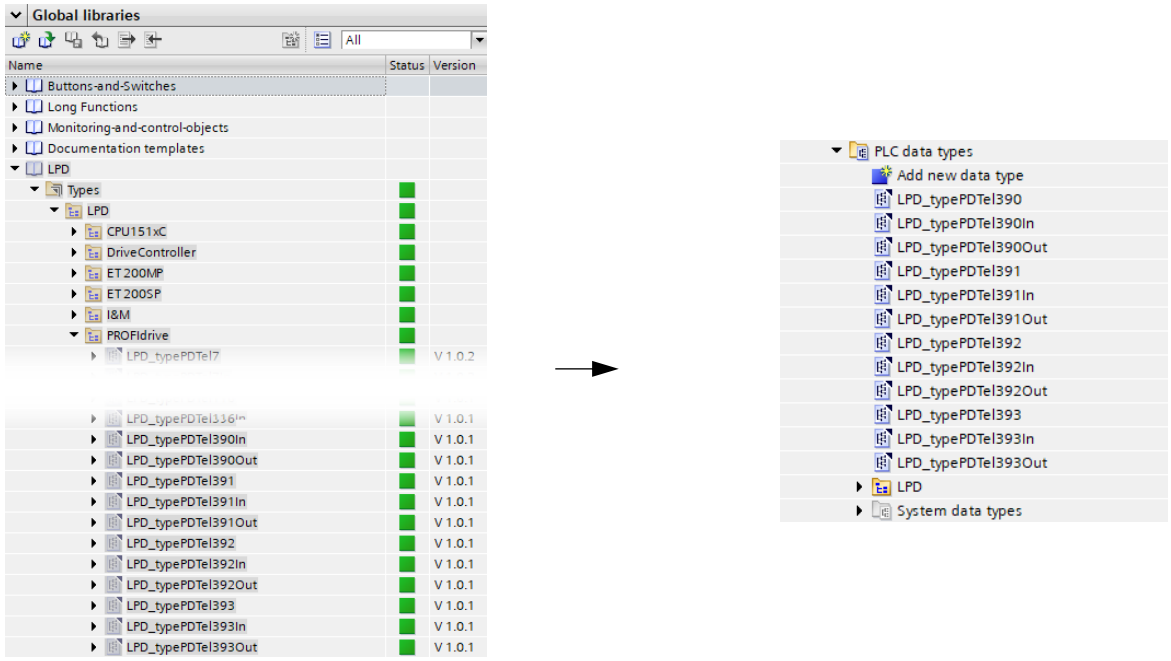
- Simulation must be set to bus- or event-synchronous mode

2.4.1. PLC connection

PLC data types

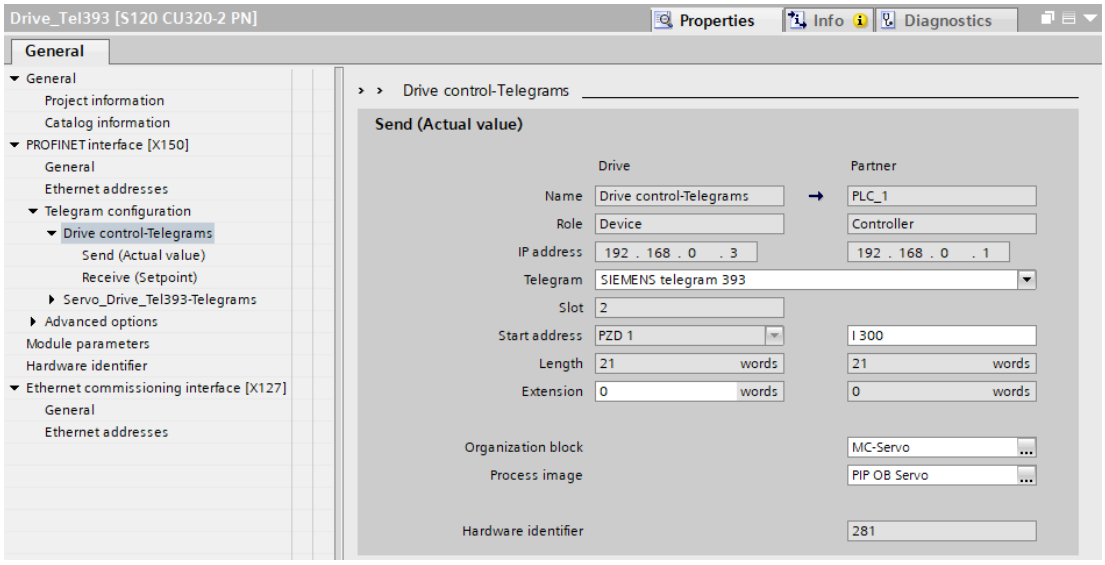
Connecting the simulation block in SIMIT with the relevant PLC in- and outputs requires adding the LPD library in the TIA Portal project \3\. After adding the library (see Figure 2-31), new data types are available which must be dragged and dropped to the *PLC data types* of the project.

Figure 2-31 LPD Library included in the TIA Portal project – Telegram 39x data types



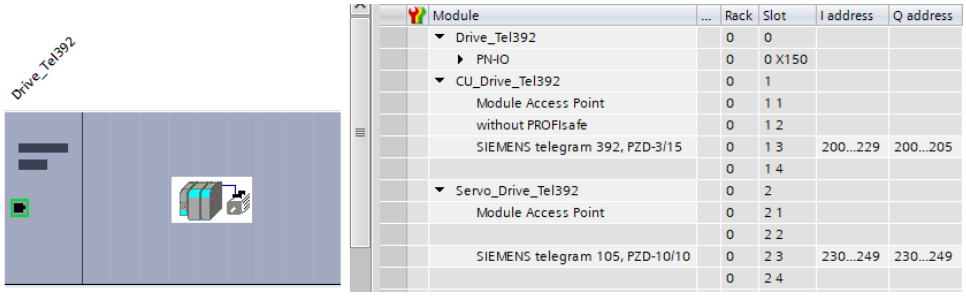
Note down the I and Q start-addresses of the telegrams 39x which can be found in the hardware configuration. The addresses of the telegrams in a drive control, configured via SINAMCIS Startdrive, can be found in the properties of the specific drive (see Figure 2-32).

Figure 2-32 I and Q addresses of *Telegram 393 – drive control (Startdrive)*



The addresses of the telegrams 39x, configured as GSDML, can directly be found in the hardware configuration of the drive (see Figure 2-33).

Figure 2-33 I and Q addresses of *Telegram 392 – CU* (*GSDML*)



In the PLC tags now the addresses of the telegrams must be added manually. If this step is skipped, the signals won't appear in the SIMIT coupling. Add a new tag with the starting address of the specific telegram and select the relevant PLC data type (see Figure 2-34). The structure will be created automatically.

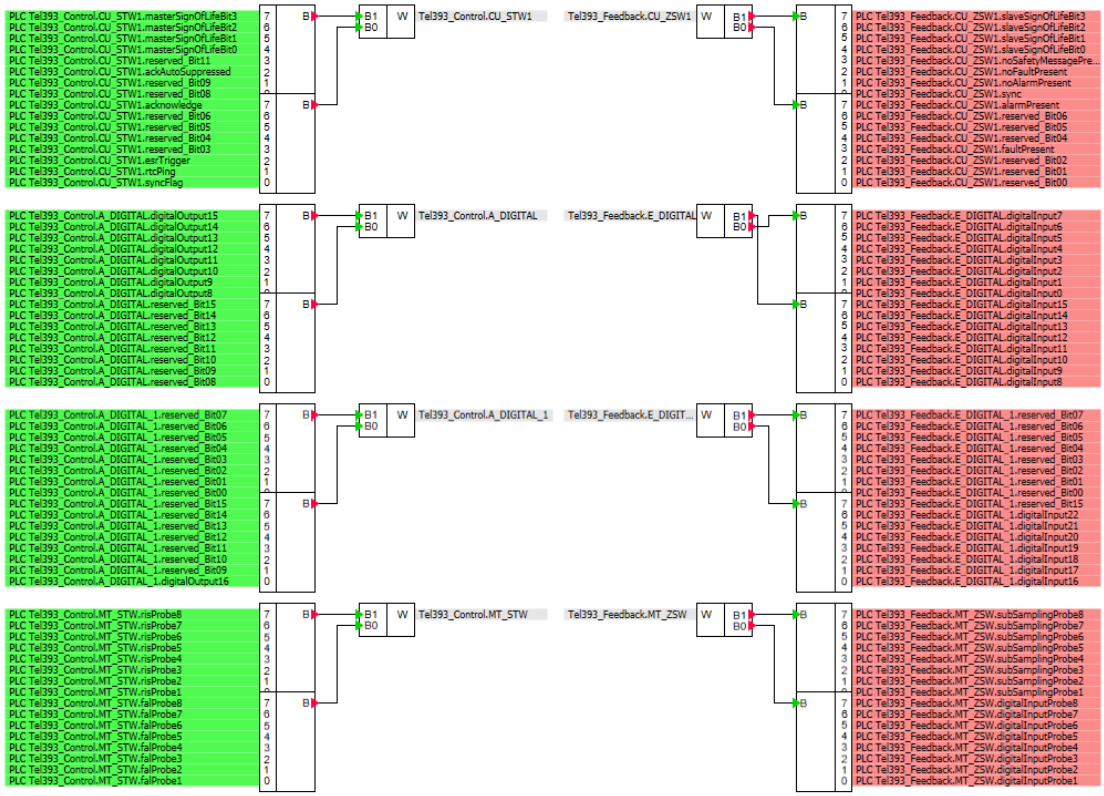
Figure 2-34 PLC tags with LPD library data types for *Telegram 393*

Connecting signals in SIMIT

After importing the coupling, all relevant signals for the telegrams can be found in SIMIT. Connect all signals with the simulation block in SIMIT. It is helpful to use word and byte splitters and mergers to connect the signals (see Figure 2-35).

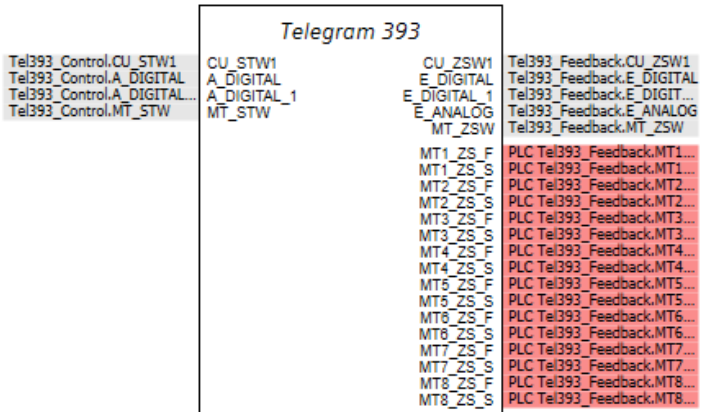
NOTE Listing of signal names in the PLC begin with number 0 as suffix. SIMIT arrays always start with number 1. Keep that mismatch in mind to do a correct signal mapping (see Figure 2-35).

Figure 2-35 Merging and splitting PLC signals for the telegrams 393



Now the signals can be connected to the simulation block (see Figure 2-36).

Figure 2-36 Connecting the merged signals with the telegram 393

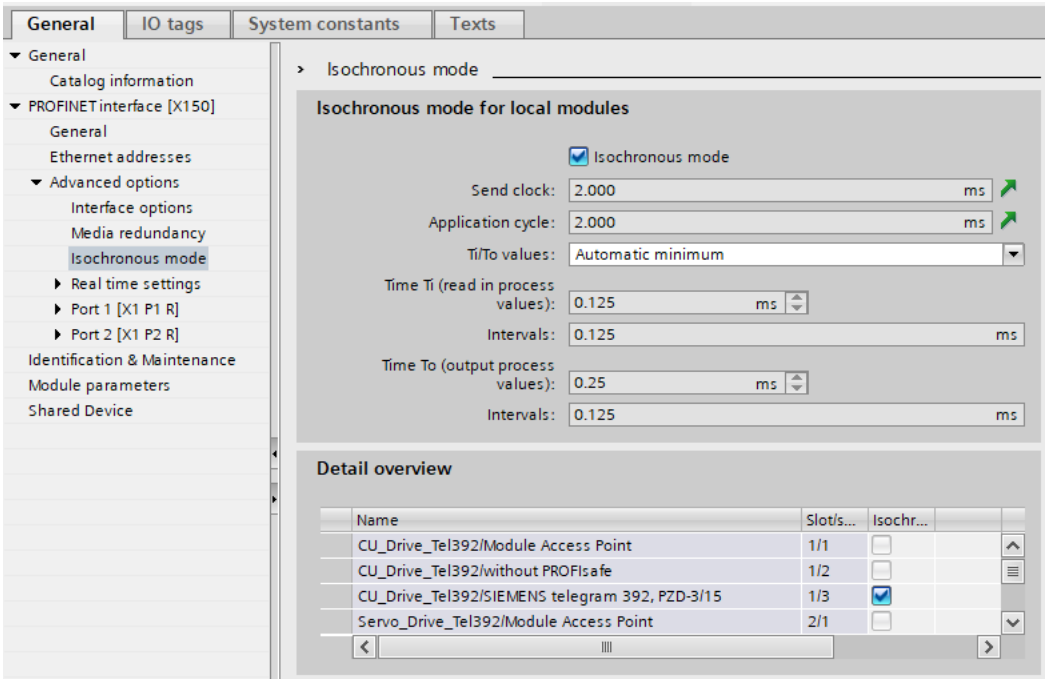


2.4.2. Parameterization

The parameters *Time_Ti*, *Time_To and Time_InternalControlLoop* of the telegram 39x components must be set equally to the PLC setup (in µs). The times of *Ti* and *To* can be found in the properties of the control unit in TIA Portal. Relevant

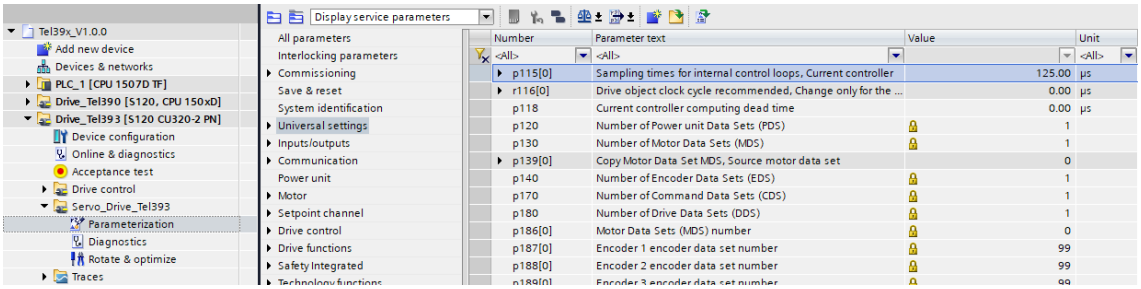
information can be found in the options of the Isochronous Mode (see Figure 2-37). Use the values of the *Intervals* below the *Time Ti* and *Time To* values.

Figure 2-37 Isochronous mode settings for Time_Ti and Time_To



If SINAMICS Startdrive is used, the *Time_InternalControlLoop* time can be found in the parameters of the servo axis. Minimally the extended parameters must be displayed to view parameter *p115* (see Figure 2-38).

Figure 2-38 Sampling times for internal control loops in p115 (Startdrive)



For drives which were configured in SINAMICS *STARTER* it can be useful to try the *Ti* or *To* time for the correct value of *Time_InternalControlLoop*.

NOTE

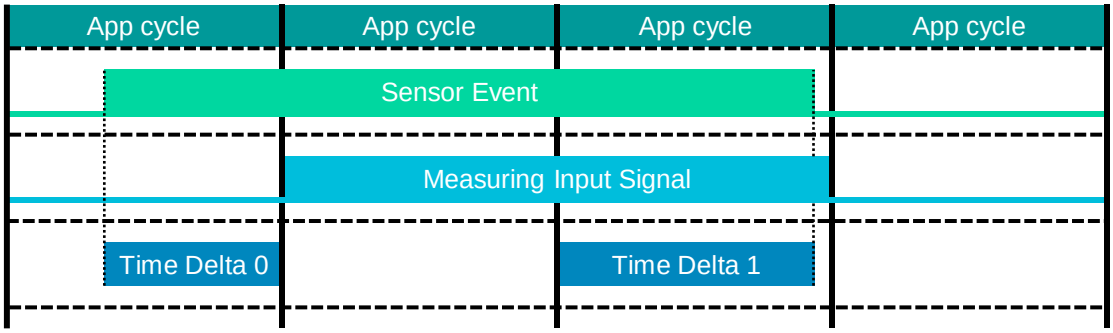
If resulting measured position values are incorrect use the secondary parameter of the telegram 39x component *Time_Offset_Simulation* to correct and fine tune the detected position. Sometimes it might be necessary to add the cycle time in µs once.

2.4.3. Working principle

Rising and falling edges of a *Sensor Event* normally occur during the application cycles. As a result, the *Measuring Input Signal* is sent to the PLC in the application cycle with the last state of the *Sensor Event*.

The time offset (*Time Delta* in µs) to the application cycle is processed by telegram 39x and it is sent as timestamp to the PLC. With the combination of the *Measuring Input Signal* and the *Time Delta* value the PLC is able to reproduce the exact positions of the occurred sensor edges relative to an axis or encoder position (see Figure 2-39).

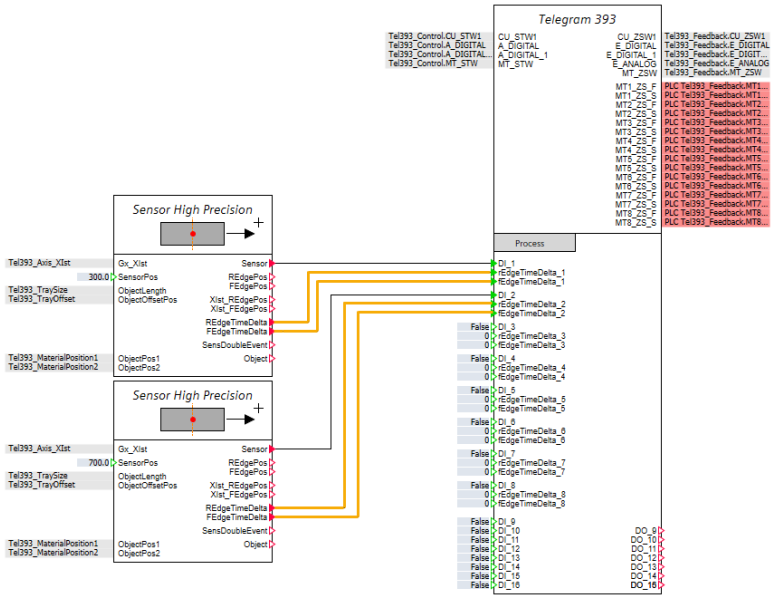
Figure 2-39 Telegram 39x measuring input cycle mechanism



Simulating exact time stamps

To simulate the exact time stamps for the *Measuring Input*, it is necessary to use and connect a *Sensor High Precision* simulation block to the telegram 39x components (see Figure 2-40).

Figure 2-40 *Sensor High Precision* connected to the *Telegram 393* component



NOTE

It is also possible to only use the binary *Sensor_Event* input (no connections of the yellow wires in Figure 2-40), however the measurement accuracy is reduced to the accuracy of the application cycle in the process.

2.4.4. Input, Output and Parameter overview

Inputs

Table 2-9 *Telegram 393* - Inputs

Inputs	Data Type	Description
PLC signals		
CU_STW1	integer	Control Word of the control unit
A_DIGITAL	integer	Digital outputs
A_DIGITAL_1	integer	Additional digital outputs
MT_STW	integer	Control word of the measuring inputs
Process signals		
DI_[16]	binary	Digital input is triggered
rEdgeTimeDelta_[8]	integer	Rising edge time delta in µs to the application cycle of the sensor signal
fEdgeTimeDelta_[8]	integer	Falling edge time delta in µs to the application cycle of the sensor signal

Outputs

Table 2-10 *Telegram 393* - Outputs

Outputs	Data Type	Description
PLC signals		
CU_STW1	integer	Status Word of the control unit
E_DIGITAL	integer	Digital inputs
E_DIGITAL_1	integer	Additional digital inputs
MT_ZSW	integer	Status Word of the measuring inputs
MT1_ZS_F	integer	Measuring input 1 falling edge time offset value
MT1_ZS_S	integer	Measuring input 1 rising edge time offset value
MT2_ZS_F	integer	Measuring input 2 falling edge time offset value
MT2_ZS_S	integer	Measuring input 2 rising edge time offset value
MT3_ZS_F	integer	Measuring input 3 falling edge time offset value
MT3_ZS_S	integer	Measuring input 3 rising edge time offset value
MT4_ZS_F	integer	Measuring input 4 falling edge time offset value
MT4_ZS_S	integer	Measuring input 4 rising edge time offset value
MT5_ZS_F	integer	Measuring input 5 falling edge time offset value
MT5_ZS_S	integer	Measuring input 5 rising edge time offset value
MT6_ZS_F	integer	Measuring input 6 falling edge time offset value
MT6_ZS_S	integer	Measuring input 6 rising edge time offset value
MT7_ZS_F	integer	Measuring input 7 falling edge time offset value
MT7_ZS_S	integer	Measuring input 7 rising edge time offset value
MT8_ZS_F	integer	Measuring input 8 falling edge time offset value
MT8_ZS_S	integer	Measuring input 8 rising edge time offset value
Process signals		
DO_[8]	binary	Digital output is set

Parameters

Table 2-11 *Telegram 393* - Parameters

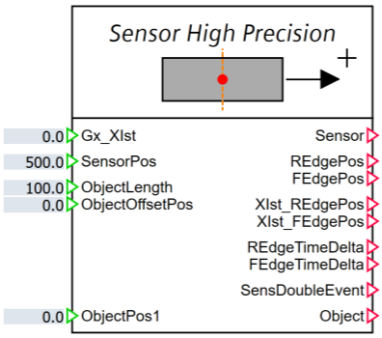
Parameters	Data Type	Description
Parameters		
Time_Ti	integer	Read in process values time in µs
Time_To	integer	Output process values time in µs
Time_InternalControlLoop	integer	Cycle time of the internal control loop in µs
Additional parameters		
Time_Offset_Simulation	integer	Additional time delta in µs to correct the time offsets in the simulation

3. Additional Components

3.1. Sensor High Precision

The *Sensor High Precision* block (see Figure 3-1) is used to generate a binary sensor signal with exact position and time stamp values in SIMIT.

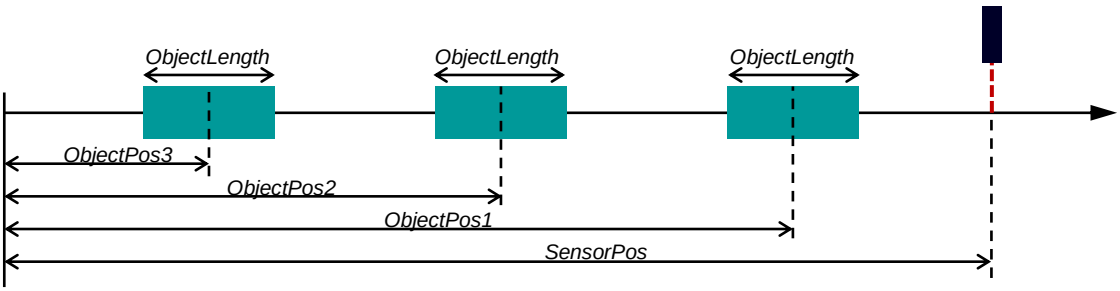
Figure 3-1 *Sensor High Precision* - SIMIT component



3.1.1. Basic settings

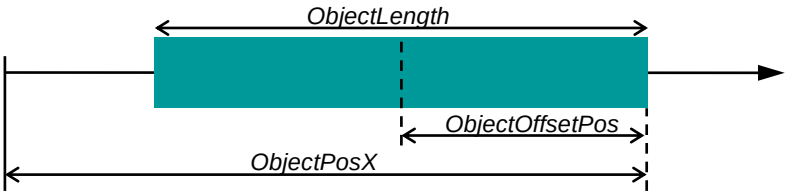
Basically, the component creates the sensor signal based on object position values (*ObjectPos*) which is compared to the sensor position (*SensorPos*) value (see Figure 3-2). Additionally, the object length (*ObjectLength*) is respected.

Figure 3-2 *Sensor High Precision* - Working principle



It is also possible to use an offset (*ObjectOffsetPos*) for the object relative to the *ObjectPos*.

Figure 3-3 *Sensor High Precision* - *ObjectOffsetPos*

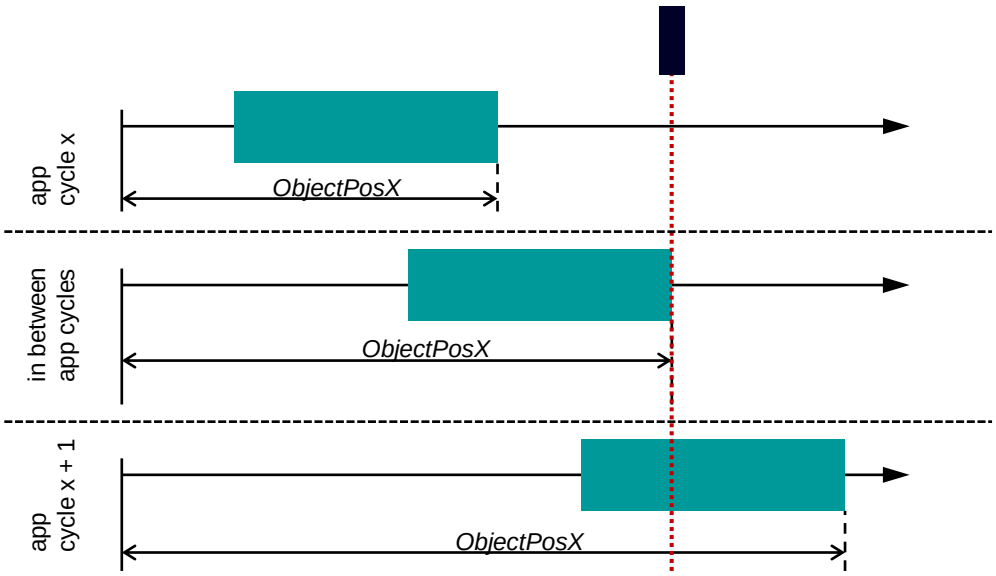


NOTE Currently it is not possible to detect objects with different lengths and object offsets. Still it is possible to change the inputs online.

3.1.2. Working principle

The *Sensor High Precision* block is calculating the exact position of the rising and falling edge of the sensor signal. Therefore, it uses the position value of an object in an application cycle before the sensor event occurs and the position value of the object in the following application cycle where the sensor signal is already triggered (see Figure 3-4). With this information and the sensor position, the exact object position and time deltas of the sensor edges are calculated.

Figure 3-4 *Sensor High Precision* - Calculating the object position in between application cycles



3.1.3. Input, Output and Parameter overview

Inputs

Table 3-1 *Sensor High Precision* Inputs

Inputs	Data Type	Description
Gx_XIst	° / mm	Optional: Axis position to create output values for *XIst_REdgePos* and *XIst_FEdgePos* – (used for sensor probes at the drive telegram)
SensorPos	mm	Position of the sensor
ObjectLength	mm	Length of the object which will be detected
ObjectOffsetPos	mm	Offset of the object to the *ObjectPosX*
ObjectPos1	mm	Object positions
...	mm	Resize the component to add *ObjectPos* inputs

Outputs

Table 3-2 *Sensor High Precision* Outputs

Outputs	Data Type	Description
Sensor	binary	Sensor is triggered
R Edge Pos	mm	Optional: Exact detected position of the rising edge
F Edge Pos	mm	Optional: Exact detected position of the falling edge
XIst_REdgePos	° / mm	Optional: *Gx_XIst* value of the rising edge – (used for sensor probes at the drive telegram)
XIst_FEdgePos	° / mm	Optional: *Gx_XIst* value of the falling edge – (used for sensor probes at the drive telegram)
REdgeTimeDelta	µs	Optional: Time offset of the rising edge to the cycle time – (used for Timer functionality)
FEdgeTimeDelta	µs	Optional: Time offset of the falling edge to the cycle time – (used for Timer functionality)
Sensor Double Event	binary	Optional: *True* if there were two sensor events (rising and falling edge) in one simulation cycle – (used for Timer functionality)

4. Example project

The example project includes the simulation block of all Tel39x blocks of this summary. The top-level charts

Telegram 390

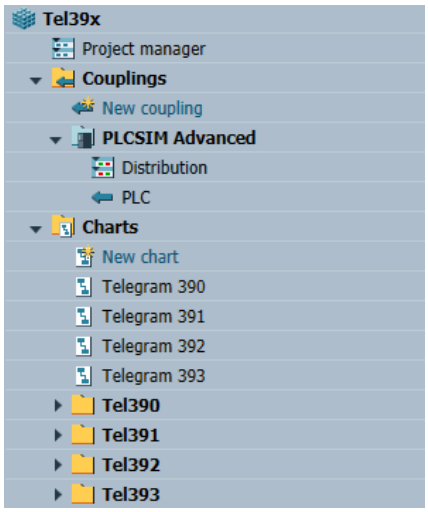
Telegram 391

Telegram 392

Telegram 393

include control panels and axis models to interact with the simulation blocks. There also are folders with relevant additional charts (see Figure 4-1).

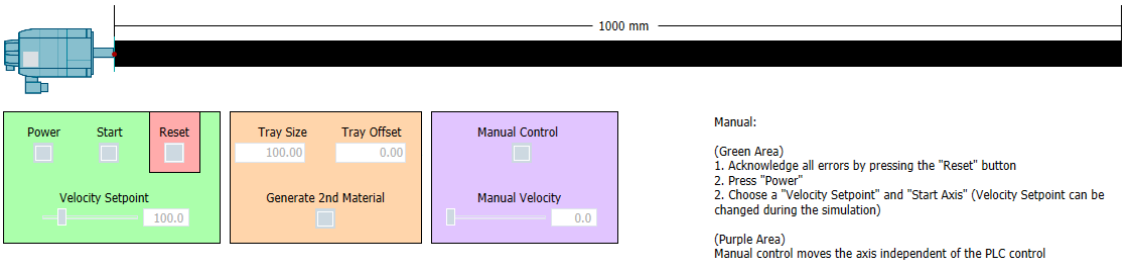
Figure 4-1 Example project structure



4.1. Telegram 390

The simulation example for telegram 390 component includes one axis with a control unit including telegram 390. No measuring inputs come with telegram 390 so there is only one axis in this example (see Figure 4-2).

Figure 4-2 SIMIT chart - *Telegram 390*



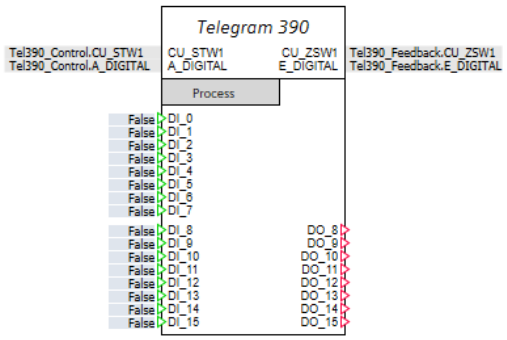
Operating the model

Acknowledge all errors via the red *Reset* button. *Power* up and *Start* (enable) the axis. Now the axis can be moved with a velocity setpoint. This will not have any influence on telegram 390. There will be no error message displayed.

Simulation setup

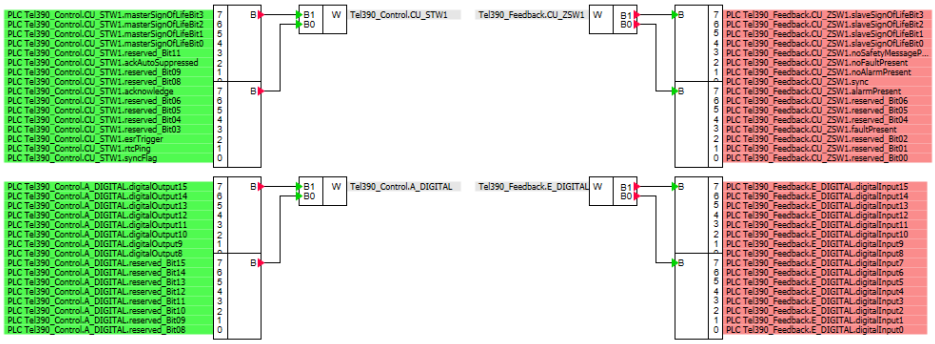
Besides the axis model there are two charts which are relevant for telegram 390. The first chart called *Tel390* includes the *Telegram 390* simulation block (see Figure 4-3). The DIs of the control unit can be set here.

Figure 4-3 SIMIT chart - *Tel390*



The second chart called *Tel390_Signals* includes *word* and *byte* splitters and mergers to connect the PLC IOs with the simulation block (see Figure 4-4).

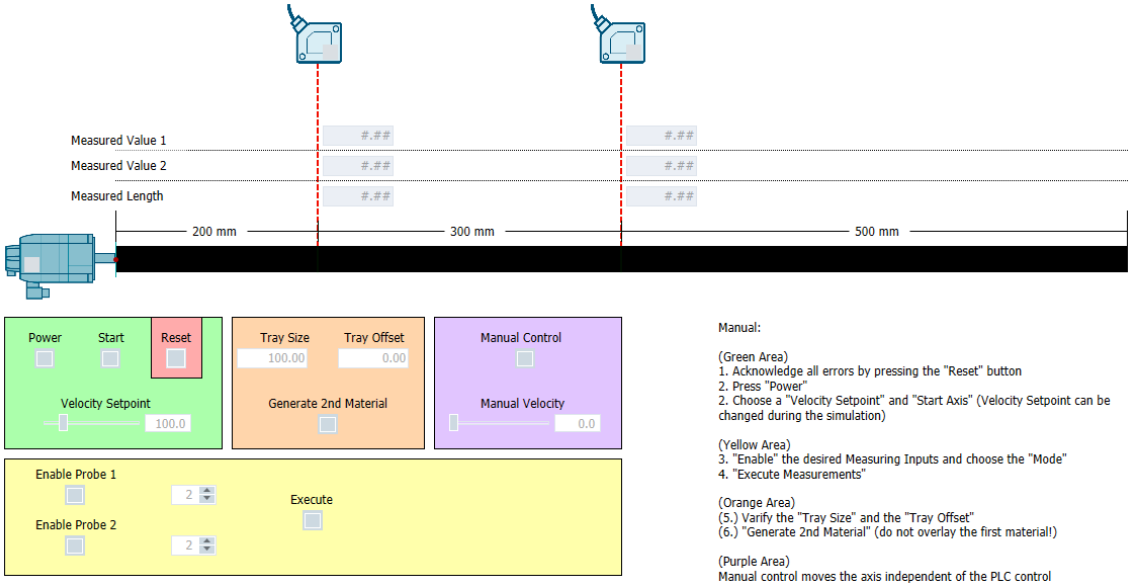
Figure 4-4 SIMIT chart - *Tel390_Signals*



4.2. Telegram 391

The simulation example for telegram 391 component includes one axis with a control unit including telegram 391. Two measuring inputs come with telegram 391 which are both used in this example (see Figure 4-5).

Figure 4-5 SIMIT chart - *Telegram 391*



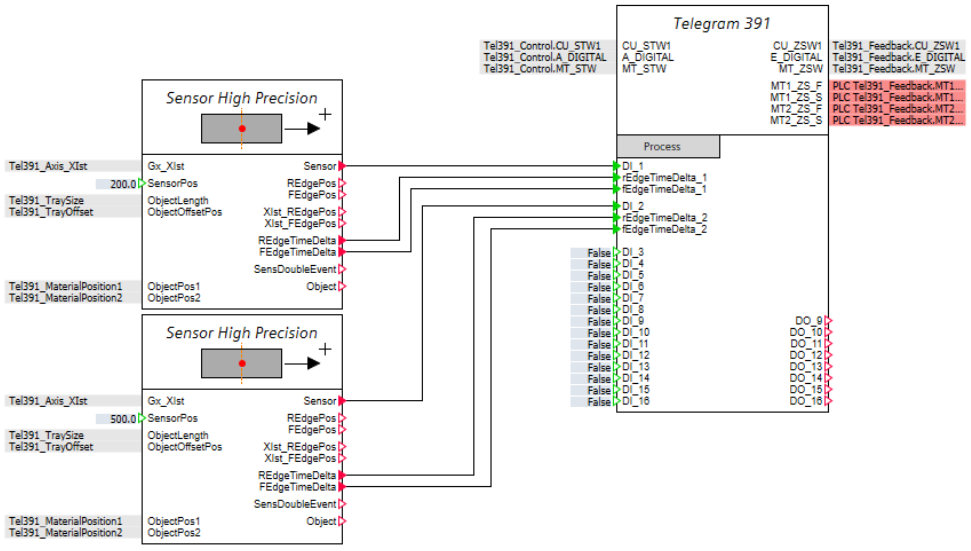
Operating the model

Acknowledge all errors via the red *Reset* button. *Power* up and *Start* (enable) the axis. Now the axis can be moved with a velocity setpoint. When the axis is running, enable the desired probes, select a mode, and execute the measuring inputs. The resulting measurement values will be shown in the displays next to the sensor beams.

Simulation setup

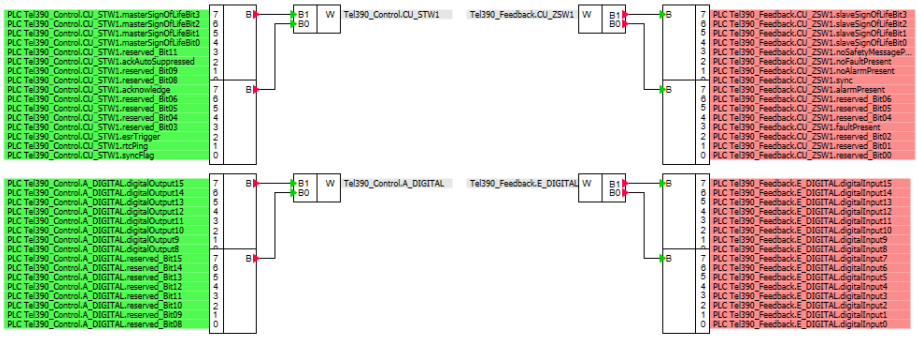
Besides the axis model there are two charts which are relevant for telegram 391. The first chart called *Tel391* includes the *Telegram 391* simulation block (see Figure 4-6). The DIs of the control unit can be set here. Also, *Sensor High Precision* simulation blocks are used to create sensor signals with time delta values for *Telegram 391*.

Figure 4-6 SIMIT chart - *Tel391*



The second chart called *Tel391_Signals* includes word and byte splitters and mergers to connect the PLC IOs with the simulation block (see Figure 4-7).

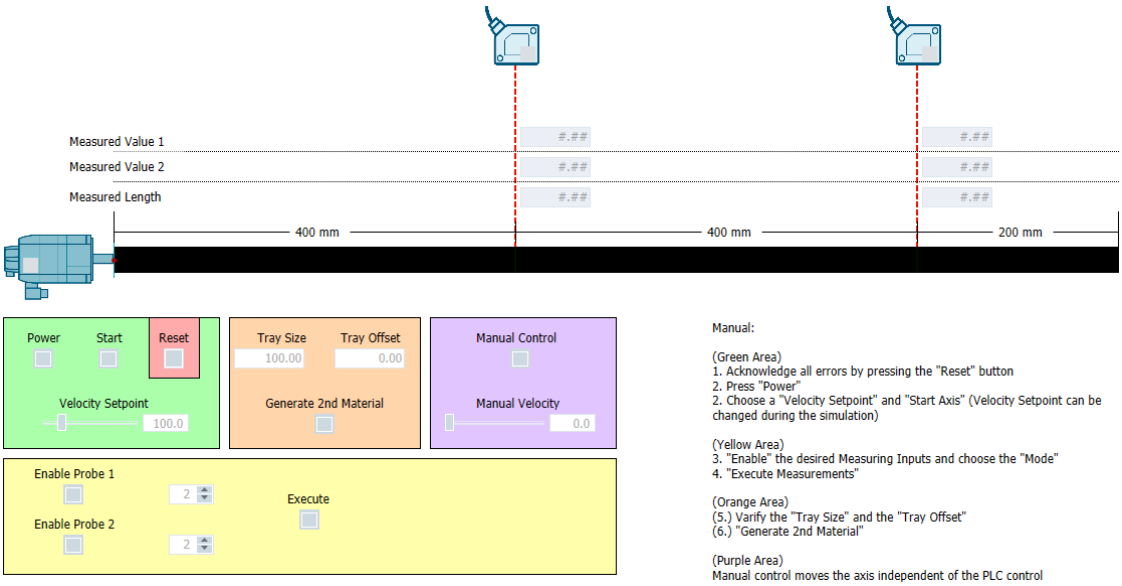
Figure 4-7 SIMIT chart - *Tel391_Signals*



4.3. Telegram 392

The simulation example for telegram 392 component includes one axis with a control unit including telegram 392. Six measuring inputs come with telegram 392. Two of them are used in this example (see Figure 4-8).

Figure 4-8 SIMIT chart - *Telegram 392*



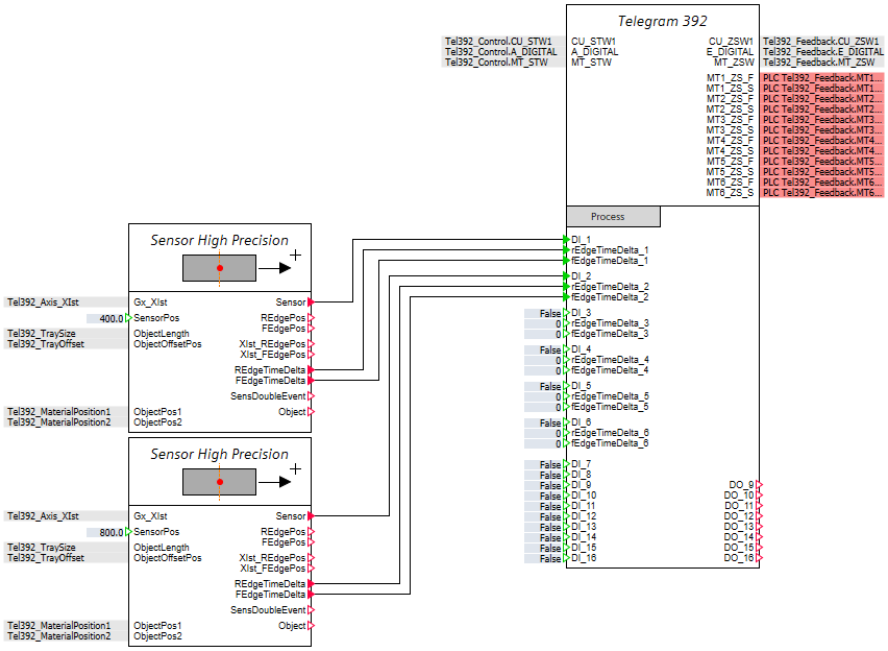
Operating the model

Acknowledge all errors via the red *Reset* button. *Power* up and *Start* (enable) the axis. Now the axis can be moved with a velocity setpoint. When the axis is running, enable the desired probes, select a mode, and execute the measuring inputs. The resulting measurement values will be shown in the displays next to the sensor beams.

Simulation setup

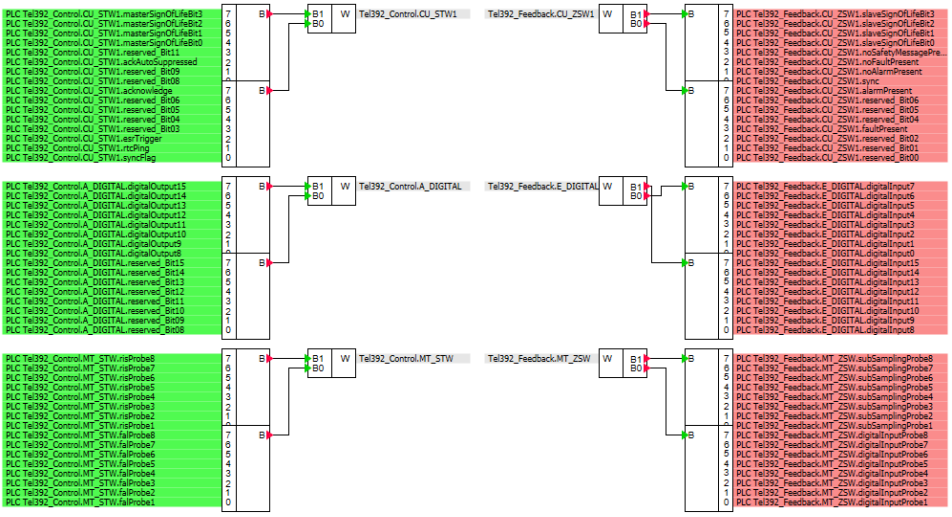
Besides the axis model there are two charts which are relevant for telegram 392. The first chart called *Tel392* includes the *Telegram 392* simulation block (see Figure 4-9). The DIs of the control unit can be set here. Also, *Sensor High Precision* simulation blocks are used to create sensor signals with time delta values for *Telegram 392*.

Figure 4-9 SIMIT chart - *Tel392*



The second chart called *Tel392_Signals* includes word and byte splitters and mergers to connect the PLC IOs with the simulation block (see Figure 4-10).

Figure 4-10 SIMIT chart - *Tel392_Signals*

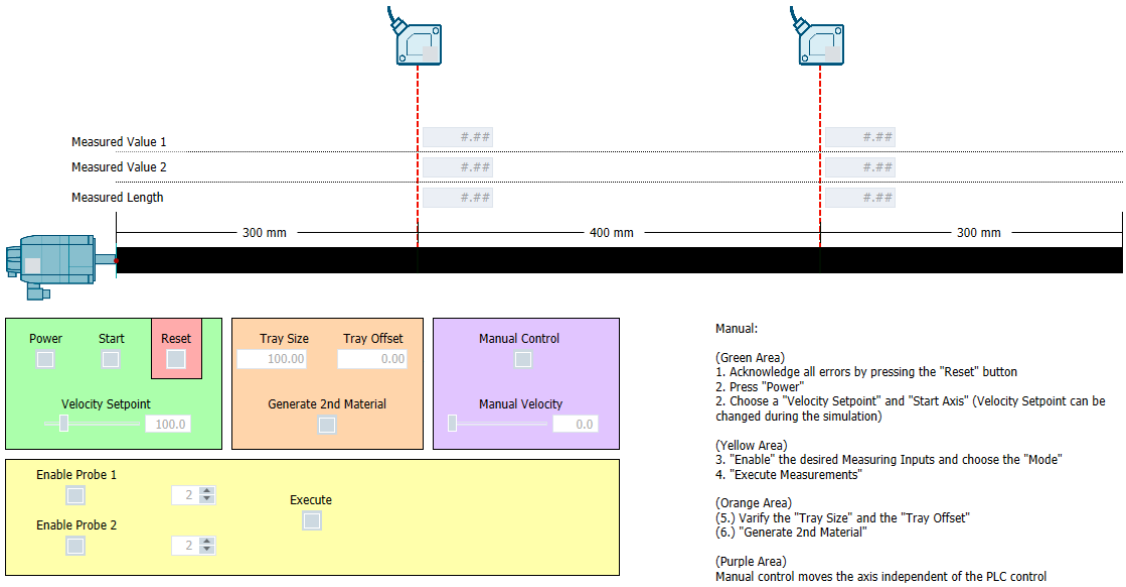


-
-

4.4. Telegram 393

The simulation example for telegram 393 component includes one axis with a control unit including telegram 393. Six measuring inputs come with telegram 393. Two of them are used in this example (see Figure 4-11).

Figure 4-11 SIMIT chart - *Telegram 393*



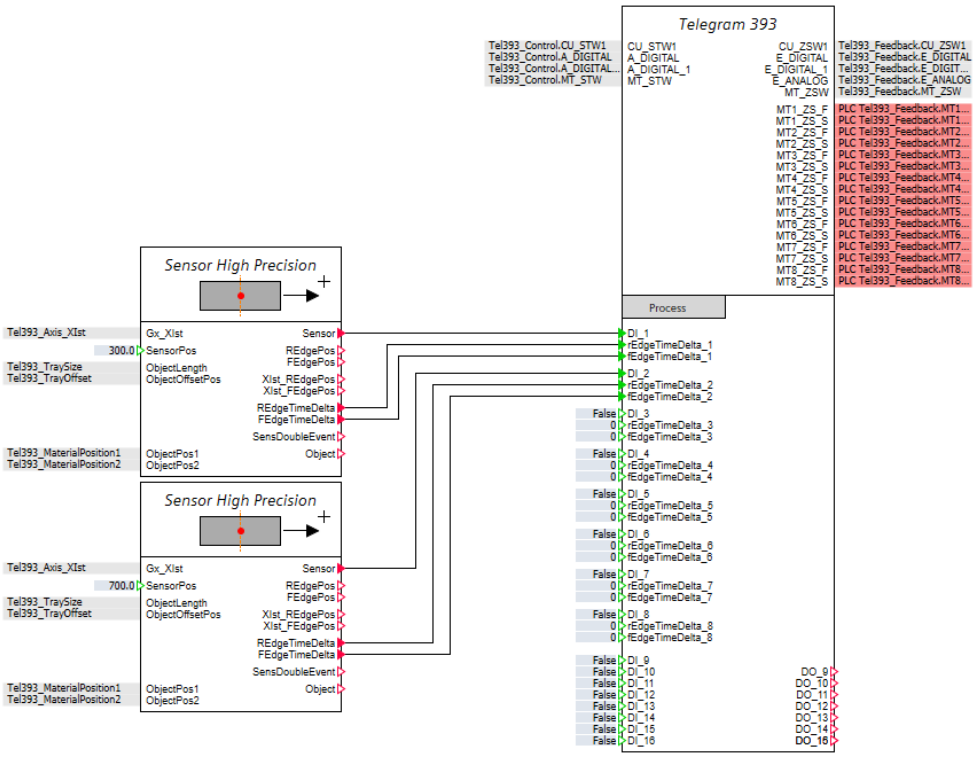
Operating the model

Acknowledge all errors via the red *Reset* button. *Power* up and *Start* (enable) the axis. Now the axis can be moved with a velocity setpoint. When the axis is running, enable the desired probes, select a mode, and execute the measuring inputs. The resulting measurement values will be shown in the displays next to the sensor beams.

Simulation setup

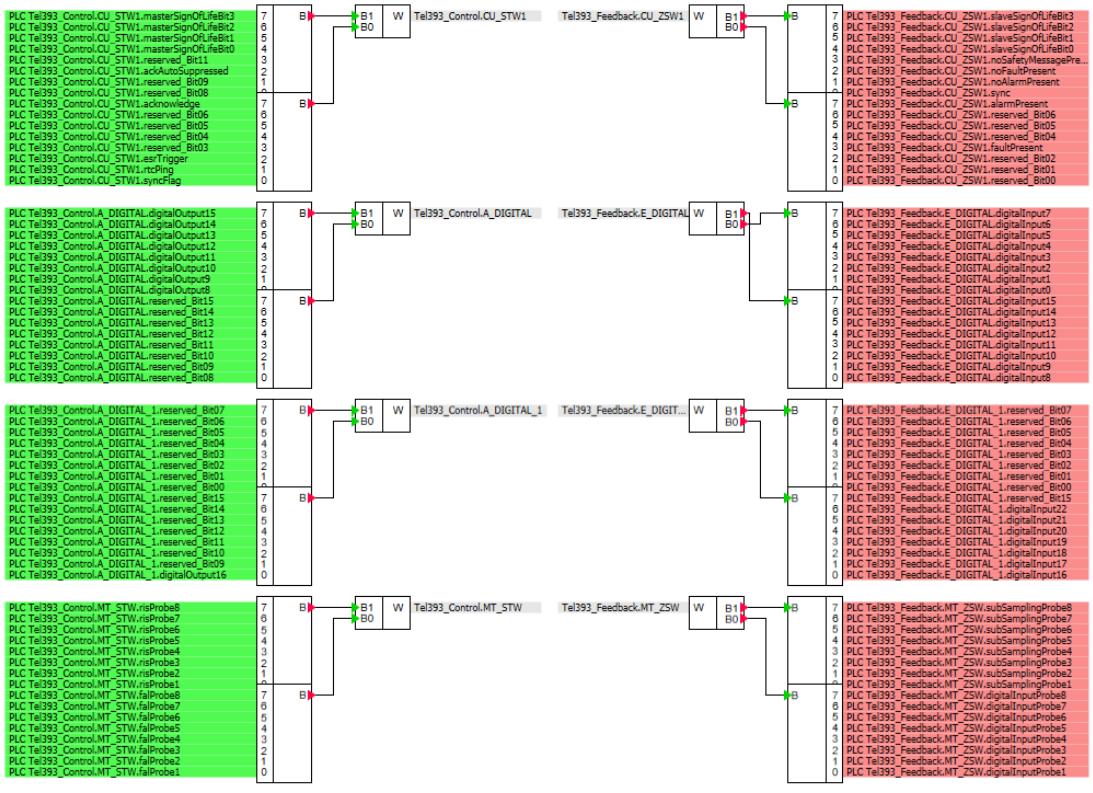
Besides the axis model there are two charts which are relevant for telegram 393. The first chart called *Tel393* includes the *Telegram 393* simulation block (see Figure 4-12). The DIs of the control unit can be set here. Also, *Sensor High Precision* simulation blocks are used to create sensor signals with time delta values for *Telegram 393*.

Figure 4-12 SIMIT chart - *Tel393*



The second chart called *Tel393_Signals* includes word and byte splitters and mergers to connect the PLC IOs with the simulation block (see Figure 4-13).

Figure 4-13 SIMIT chart - *Tel393_Signals*



•

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:

siemens.com/SupportRequest

SITRAIN – Digital Industry Academy

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:

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. Industry Mall

The Siemens Industry Mall is the platform on which the entire siemens Industry product portfolio is accessible. From the selection of products to the order and the delivery tracking, the Industry Mall enables the complete purchasing processing – directly and independently of time and location:
mall.industry.siemens.com

5.3. Application support

Siemens AG
Digital Industries
Factory Automation
Kinematics & Simulation
DI FA S SUP K&SIM
Gleiwitzer Str. 555
90475 Nuremberg, Germany

5.4. Links and literature

Table 5-1

Nr.	Thema
\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/123456789
\3\	Libraries of PLC data types (LPD) for STEP 7 (TIA Portal) and SIMATIC S7-1200 / S7-1500 https://support.industry.siemens.com/cs/us/en/view/109482396
\4\	SIMIT libraries with behavior models of devices and systems https://support.industry.siemens.com/cs/ww/en/view/109793203

5.5. Change documentation

Table 5-2

Version	Date	Modifications
V1.0.0	07/2023	First version