

SIEMENS

Ingenuity for life

Industry Online Support

Home

Controlling a SINAMICS Active Line Module using telegram 370

SINUMERIK 840D sl, SINUMERIK ONE

<https://support.industry.siemens.com/cs/ww/en/view/109474566>

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 content

Legal information	Fehler! Textmarke nicht definiert.
1 Task	5
1.1 Controlling an ALM using telegram 370	5
1.2 History behind ALM control	5
1.3 Usability of the control via telegram 370 for SINUMERIK 840Dsl	5
1.4 Usability of the control via telegram 370 for SINUMERIK ONE	5
2 Solution	6
2.1 Technical overview of the ALM control using telegram 370 for SINUMERIK 840Dsl	6
2.2 Technical overview of the ALM control using telegram 370 for SINUMERIK ONE	6
3 Commissioning	7
3.1 Activating telegram 370 in the ALM-DO SINUMERIK 840Dsl / SINUMERIK ONE	7
3.2 Including telegram 370 in cyclic data transfer SINUMERIK 840Dsl / SINUMERIK ONE	8
3.3 Activating PLC access to the telegram in NC SINUMERIK 840Dsl Step7 V5.X / Step7 V1X	9
3.4 Activating PLC access to the telegram in NC SINUMERIK ONE Step7 V15 -V1x	10
3.5 Creating an ALM interface DB in the PLC SINUMERIK 840Dsl	10
3.6 Creating an ALM interface DB in the PLC SINUMERIK ONE	12
3.7 Calling FC1031 for data transfer ALM interfaces DB SINUMERIK 840Dsl (telegram)	13
3.8 Creating a user-based PLC block to control the ALM	14
3.8.1 Controlling OFF1 of the ALM	14
3.8.2 Controlling OFF1 of the ALM SINUMERIK 840Dsl	14
3.8.3 Controlling OFF1 of the ALM SINUMERIK ONE	15
3.8.4 Permanently assigning additional signals SINUMERIK 840Dsl	16
3.8.5 Permanently assigning additional signals SINUMERIK ONE	17
3.9 PLC block sources	18
3.9.1 PLC block sources SINUMERIK 840Dsl	18
3.9.2 PLC block sources SINUMERIK ONE	18
3.9.3 Link to Drive_Lib_S7_1200_1500 library	18
4 Controlling ALM using block SINA_INFEED	19
4.1 Controlling ALM using block SINA_INFEED	19
4.1.1 SINA_INFEED block in project block folder	19
4.1.2 Description of the SINA_INFEED block	19
4.1.3 Function description	20
4.1.4 Input interface of SINA_INFEED	20
4.1.5 Default setting of input ConfigAxis	20
4.1.6 Output interface of SINA_INFEED	21
4.1.7 Troubleshooting function block SINA_INFEED	21
4.1.8 Input signals HWIDSTW and HWIDZTW	22
5 Additional information	23
5.1 Structure of telegram 370	23
5.2 Signals in telegram 370	23
5.2.1 E_STW1	23
5.2.2 E_ZSW1	23
5.3 Additional documentation about telegram 370	24

Table of content

6	Contact person	25
7	History.....	26

1 Task

1.1 Controlling an ALM using telegram 370

The document describes the configuration to control an Active Line Module (ALM) using telegram 370. Using telegram 370 to control the ALM, means that the ALM-relevant BICO interconnections to terminal strip X122/X132 on the NCU can be eliminated. Users can then use the terminals that become available. All ALM signal data is exchanged via an interface DB in the PLC.

1.2 History behind ALM control

The previous approach when using an Active Line Module (ALM) involved telegram "999" being preselected. In addition, while commissioning, various BICOs were preassigned in the ALM so that the ALM could be controlled via the X122/X132 interfaces on the NCU.

Telegram 370 is available in the S7 telegram container (HW Config); however, it was not involved in the cyclic data transfer of the NCU/NX.

1.3 Usability of the control via telegram 370 for SINUMERIK 840Dsl

An ALM can be controlled via telegram 370 from software release 4.7 and higher. In addition, various preliminary software releases exist, which already have the functionality required. However, these are not officially available to all users. To be able to use the subsequently described functionality, the following general machine data (MD) must exist:

- 10520 \$MN_PLCINTERN_LOGIC_ADRESS_IN[x] and
- 10525 \$MN_PLCINTERN_LOGIC_ADRESS_OUT[x]

If both MD arrays exist in the general MD, then telegram 370 can be used to control the ALM; if these are not available, then there is technically no way of controlling the ALM using telegram 370.

1.4 Usability of the control via telegram 370 for SINUMERIK ONE

Controlling the ALM using telegram 370 as described, is possible with any SINUMERIK ONE software version. Contrary to SIUMERIK 840Dsl, the two machine data mentioned in Section 1.3 do not exist in SINUMERIK ONE.

2 Solution

The configuration to control the Active Line Module (ALM) using telegram 370 is described in the following. The configuration can also be used to control Smart Line Modules controlled via Drive-CLiQ (SLM - e.g. SLM36kW).

The advantages when controlling using telegram 370 include:

- PLC→NCU (X122.1) wiring is no longer required to control the ALM
- Wiring to the line contactor control can be implemented using standard SIMATIC I/O [wiring to the NCU (X132) is not required]

2.1 Technical overview of the ALM control using telegram 370 for SINUMERIK 840Dsl

In principle, the following points are necessary to implement ALM control using telegram 370:

- Telegram 370 is activated in the ALM-DO of the SINAMICS
- The telegram is included in cyclic data transfer
- Access to the telegram in the NC is activated
- Creating an ALM interface DB in the PLC
- FC1031 is called for data transfer ALM interface DB ↔ telegram
- Creating a user-based PLC block to control the ALM

2.2 Technical overview of the ALM control using telegram 370 for SINUMERIK ONE

In principle, the following points are necessary to implement ALM control using telegram 370:

- Telegram 370 is activated in the ALM-DO of the SINAMICS
- The telegram is included in cyclic data transfer
- Access to the telegram in the NC is activated
- Creating an ALM interface DB in the PLC
- Creating a user-based PLC block to control the ALM or PLC block from the TIA Drive Library

3 Commissioning

3.1 Activating telegram 370 in the ALM-DO SINUMERIK 840Dsl / SINUMERIK ONE

The parameter in the ALM-DO should be set to p2079 = 370 to activate telegram 370 in the ALM-DO. When setting p2079 to 370, all of the necessary BICOs in the ALM-DO are automatically set corresponding to the specification of telegram 370. The following comparison shows the different parameter assignments of the BICO-relevant parameters in the ALM between settings p2079 = 999 (left-hand data column) and p2079 = 370 (right-hand data column).

Fig. 3-1

# p840[0] (ON / OFF (OFF1))	1:722.0	63:2090.0
# p844[0] (OFF2 S_src 1)	0:1.0	63:2090.1
# p852[0] (Operation enable)	0:1.0	63:2090.3
# p854[0] (Master ctrl by PLC)	0:1.0	63:2090.10
# p2051[0] (IF1 PZD send word)	0:0.0	63:2089.0
# p2079 (IF1 PD PZD tel ext)	999	370
# p2080[0] (Bin/con ZSW1)	0:0.0	63:899.0
# p2080[1] (Bin/con ZSW1)	0:0.0	63:899.1
# p2080[2] (Bin/con ZSW1)	0:0.0	63:899.2
# p2080[3] (Bin/con ZSW1)	0:0.0	63:2139.3
# p2080[4] (Bin/con ZSW1)	0:0.0	63:899.4
# p2080[6] (Bin/con ZSW1)	0:0.0	63:899.6
# p2080[7] (Bin/con ZSW1)	0:0.0	63:2139.7
# p2080[9] (Bin/con ZSW1)	0:0.0	63:899.9
# p2080[11] (Bin/con ZSW1)	0:0.0	63:899.11
# p2080[12] (Bin/con ZSW1)	0:0.0	63:899.12
# p2103[0] (1. Acknowledge)	0:0.0	63:2090.7
# p3532 (INF mot mode inhib)	0:0.0	63:2090.5
# p3533 (INF gen mode inhib)	0:0.0	63:2090.6


3.2 Including telegram 370 in cyclic data transfer SINUMERIK 840Dsl / SINUMERIK ONE

To include telegram 370 in cyclic data transfer, it is necessary to adapt the arrangement of the DO numbers in parameter array p0978 of the CU/NX. In p0978, the DOs are assigned corresponding to the telegram arrangement in the PLC HW Config (Fig. 3-2).

In principle, to include telegram 370 in cyclic data transfer, the existing DO numbers of the ALM before the end of the table (first "0" in the data set) must be located at the corresponding parameters for arrangement in the telegram container (standard case, p0978[7]):

Fig. 3-2

... = p978 [0] (Liste der DO)	3	3
= p978 [1] (Liste der DO)	4	4
= p978 [2] (Liste der DO)	5	5
= p978 [3] (Liste der DO)	6	6
= p978 [4] (Liste der DO)	255	255
= p978 [5] (Liste der DO)	255	255
= p978 [6] (Liste der DO)	1	1
≠ p978 [7] (Liste der DO)	255	2
= p978 [8] (Liste der DO)	0	0
≠ p978 [9] (Liste der DO)	2	60
≠ p978 [10] (Liste der DO)	60	0

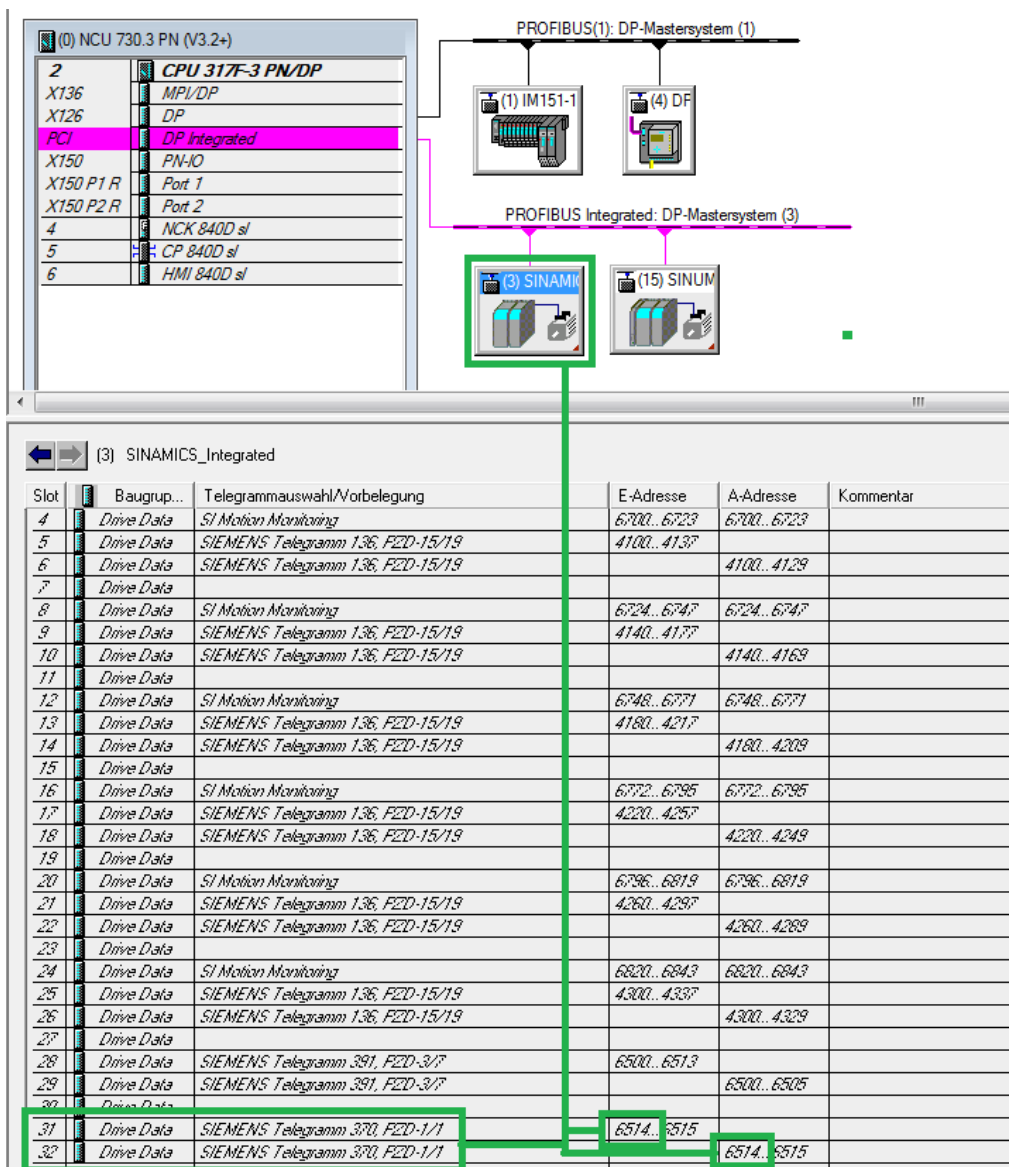


3.3 Activating PLC access to the telegram in NC SINUMERIK 840Dsl Step7 V5.X / Step7 V1X

To activate PLC access to the telegram in the NC, the basis address of telegram 370 from HW Config must be written to the following MD (standard case: 6514) (Fig. 3-3).

- 10520 \$MN_PLCCINTERN_LOGIC_ADDRESS_IN[x]
- 10525 \$MN_PLCCINTERN_LOGIC_ADDRESS_OUT[x]

Fig. 3-3 Excerpt from HW Config, with reference to ALM telegram 370



3.4 Activating PLC access to the telegram in NC SINUMERIK ONE Step7 V15 -V1x

No special measures are required to activate PLC access to the telegram in the NC.

3.5 Creating an ALM interface DB in the PLC SINUMERIK 840Dsl

An ALM interface DB is required in the PLC to establish communication. The user must actively create this ALM interface DB in the PLC. The ALM interface data block is not automatically generated by the basic PLC program.

The user should preferably generate the ALM interface data block as DB1031 in the PLC program. The necessary structure of the DB can be defined using the UDT1031 block provided. When required, the user can adapt the comments to the interface signals in the UDT1031. In the ALM data block, the following signal arrangement is defined (Table 3-1) for each of the six possible ALMs. The data sets for all ALMs are saved in the DB without any intermediate spaces.

Table 3-1

DB1031	Signals from/to the infeed via telegram 370							
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DBB ALM +0	Reserved	Reserved	Reserved	Reserved	Reserved	Control by PLC	Reserved	Reserved
DBB ALM +1	Fault Acknowledge	Inhibit generating mode	Operation motoring mode inhibit	Reserved	Enable operation	Reserved	No OFF2	ON/OFF1
DBB ALM +2	Reserved	Reserved	Reserved	Feedback signal, line contactor closed	Precharging completed	Reserved	Control requested by PLC	Reserved
DBB ALM +3	Alarm active	Switching on inhibited	Reserved	No OFF2 active	Fault active	Operation enabled	Ready	Ready for switching on
DBW ALM +4	LADDR_in address input range ALM							
DBW ALM +6	RET_VAL_IN // fault code SFC14							

DBW ALM +8	LADDR_output address output range ALM
DBW ALM+1 0	RET_VAL_out // fault code SFC15

The assignment of DB No. 1031 is not absolutely necessary, and it can be changed by the PLC programming (the parameterization of FC1031 must be observed!).

The ALM interface DB emulates all STWs and ZSWs of the telegram communication to the maximum of six possible ALM/SLM.

The telegram includes:

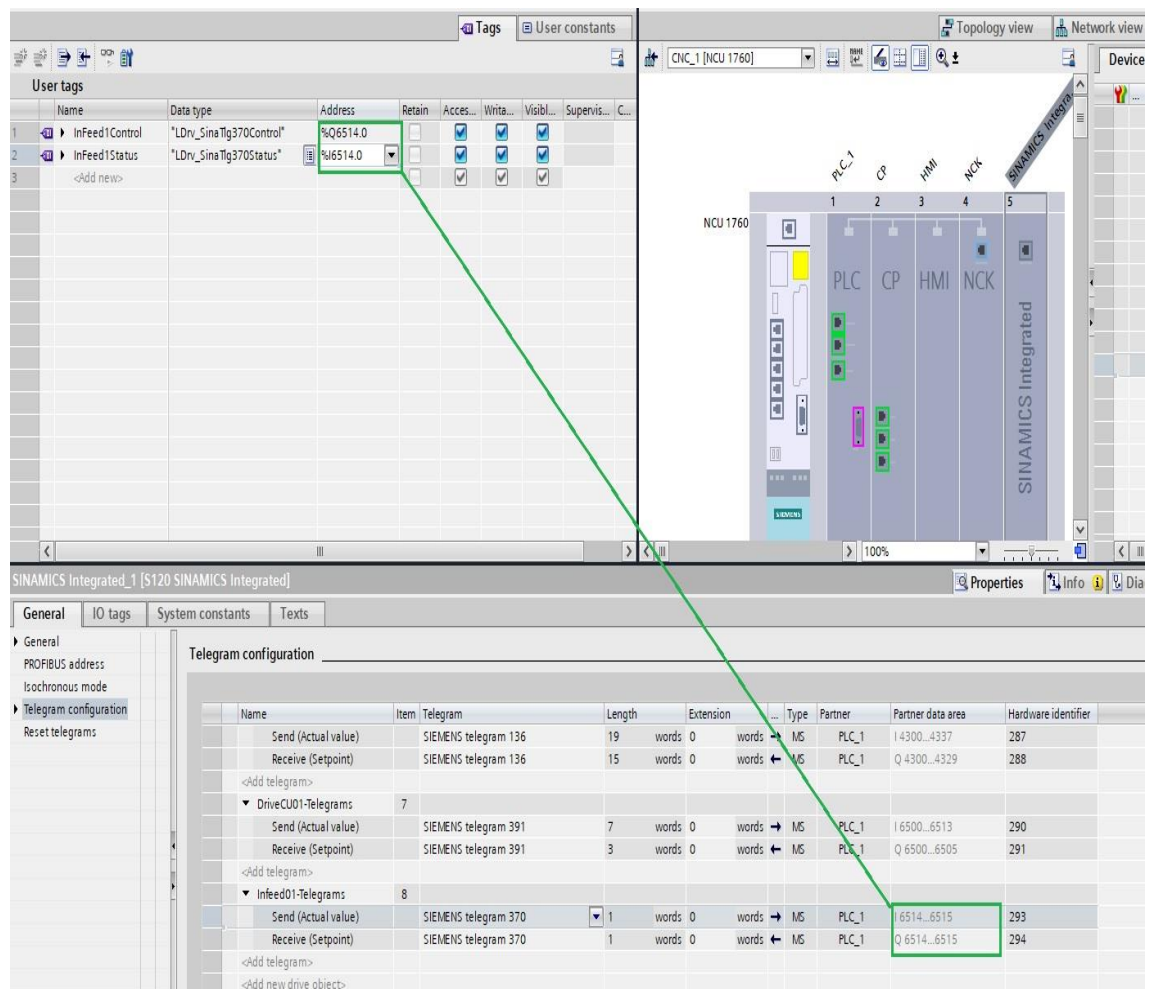
- One STW (E_STW) and
- One ZSW (E_ZSW)

3.6 Creating an ALM interface DB in the PLC SINUMERIK ONE

If the block SINA_INFEED is taken from library Drive_Lib_S7_1200_1500, then this step is not necessary.

If you wish to write your own block for controlling the ALM, then you must import and compile data types LDrv_SinaTlg370Control and LDrv_SinaTlg370Status from the attached source, and assign to the input/output addresses of the ALM. The assignment is made in the PLC variables, here InFeed1Control and InFeed1Status as example

Fig. 3-4



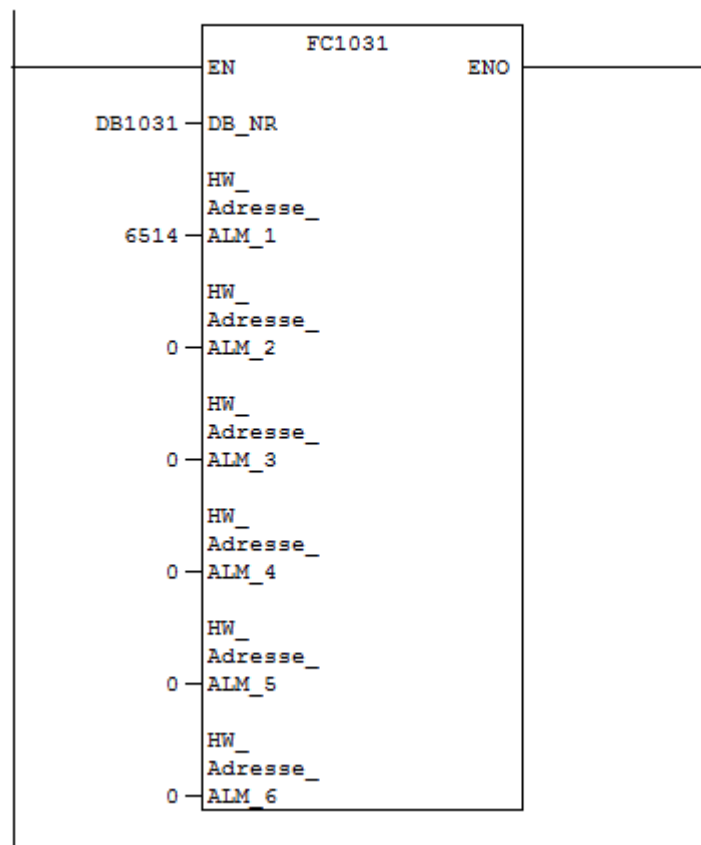
3.7 Calling FC1031 for data transfer ALM interfaces DB SINUMERIK 840Dsl (telegram)

Data transfer to/from the ALM interface DB to/from the telegram is established by cyclically calling the FC1031 provided in the PLC program (OB1). When called in OB1, FC10.31 should be parameterized using seven input parameters. A parameter assignment example is subsequently provided (ALM interface DB: DB1031, one ALM in the system, basis address: 6514):

For SINUMERIK ONE, this step is not necessary.

Fig. 3-5

▣ **Netzwerk 7 : ALM-DB access via telegram 370**



- Parameter DB No. S7 identifier of the ALM-DB (interface DB)
- Parameter HW_address_ALM_x: basis addresses of the ARM telegram from HW Config.

Note "0" should be written to basis addresses that are not used/not required.

3.8 Creating a user-based PLC block to control the ALM

The logic to control the ALM should be programmed in a PLC block to be created by the user.

Depending on the particular application, the following must be programmed:

- Controlling OFF1 of the ALM
- Fixed assignment of additional signals to telegram 370

3.8.1 Controlling OFF1 of the ALM

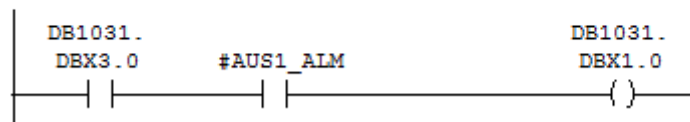
Signal OFF1 (E_STW1.Bit0) is used to switch on the ALM.

3.8.2 Controlling OFF1 of the ALM SINUMERIK 840DsI

Signal OFF1 (E_STW1.Bit0) is used to switch on the ALM. In order to ensure that the ALM starts without any faults, signal OFF1 may only be set after the ALM has signaled "Ready to switch on". The "Ready to switch on" signal is provided in telegram 370 using ZSW1.Bit0. Programming example for actively switching on the ALM by the PLC.

Fig. 3-6

Netzwerk 1: Zuschaltung ALM (frueher X122.1: AUS1)



- DB1031.DBX3.0 = "Ready to switch on", ZSW1.Bit0
- #AUS1_ALM = switch-on signal from the user PLC
- DB1031.DBX1.0 = "OFF1", STW1.Bit0

Programming example for automatically switching-on the ALM when it is ready:

Fig. 3-7

Netzwerk 1: Zuschaltung ALM (frueher X122.1: AUS1)

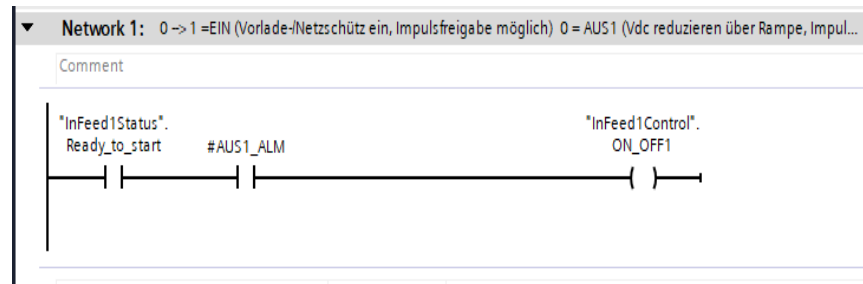


- DB1031.DBX3.0 = "Ready to switch on", ZSW1.Bit0
- DB1031.DBX1.0 = "OFF1", STW1.Bit0

3.8.3 Controlling OFF1 of the ALM SINUMERIK ONE

Signal OFF1 (E_STW1.Bit0) is used to switch on the ALM. In order to ensure that the ALM starts without any faults, signal OFF1 may only be set after the ALM has signaled "Ready to switch on". The "Ready to switch on" signal is provided in telegram 370 using ZSW1.Bit0. Programming example for actively switching on the ALM by the PLC.

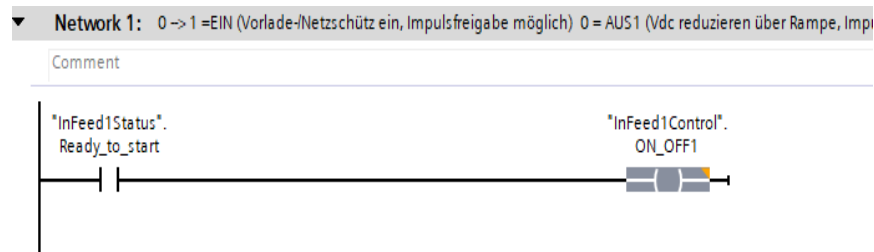
Fig. 3-8



- "InFeed1Status".Ready_to_start = "ready to switch on", ZSW1.Bit0
- #AUS1_ALM = switch-on signal from the user PLC
- "InFeed1Control".ON_OFF1 = "OFF1", STW1.Bit0

Programming example for automatically switching-on the ALM when it is ready:

Fig. 3-9



- "InFeed1Status".Ready_to_start = "ready to switch on", ZSW1.Bit0
- "InFeed1Control".ON_OFF1 = "OFF1", STW1.Bit0

3.8.4 Permanently assigning additional signals SINUMERIK 840DsI

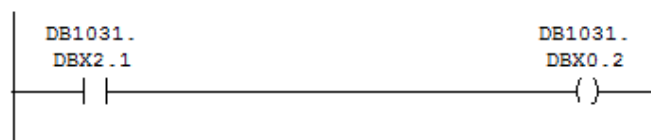
Up until now, the following signals were permanently interconnected via BICO; now, the user must supply them when telegram 370 is used for control:

- Control by PLC (E_STW1.Bit10)
- No OFF2(E_STW1.Bit1)
- Operation enabled (E_STW1.Bit3)
- Inhibit infeed motor operation (E_STW1.Bit5)
- Inhibit infeed regenerative operation (E_STW1.Bit6)

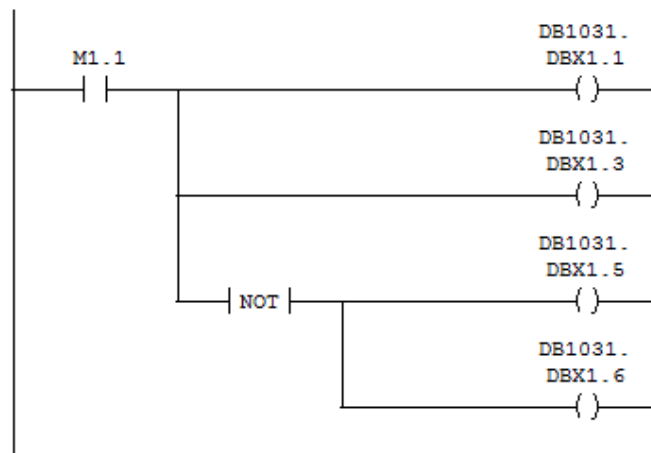
The example (Fig 3-10) shows the interconnection of the five signals to be supplied.

Fig. 3-10

□ **Netzwerk 2 : Führung durch PLC: lesen ZSW und anfordern in STW**



□ **Netzwerk 3 : notwendige Bits zum ALM-Betrie, früher über BiCo fest vorgegeben**



- DB1031.DBX2.1 = "Control requested", ZSW1.Bit9
- DB1031.DBX0.2 = "Control by PLC", STW1.Bit10
- DB1031.DBX0.1 = "No OFF2", STW1.Bit1
- DB1031.DBX0.3 = "Operation enabled", STW1.Bit3
- DB1031.DBX0.5 = "Inhibit motor operation", STW1.Bit5
- DB1031.DBX0.6 = "Inhibit regenerative operation", STW1.Bit6

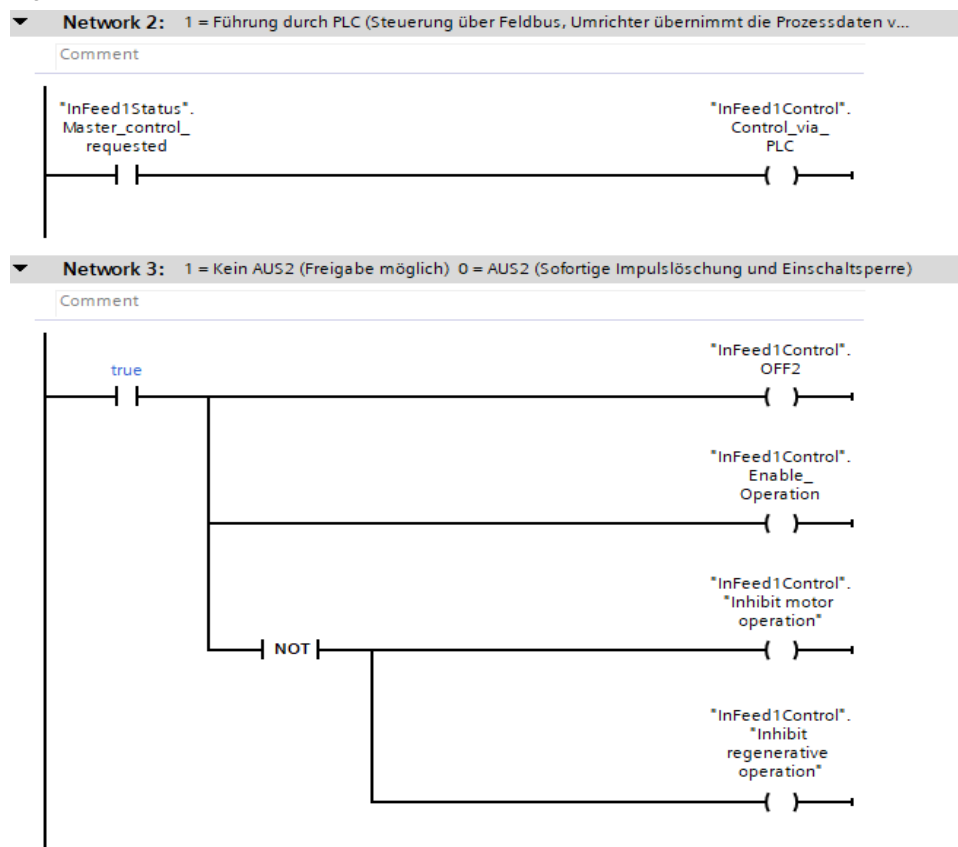
3.8.5 Permanently assigning additional signals SINUMERIK ONE

Up until now, the following signals were permanently interconnected via BICO; now, the user must supply them when telegram 370 is used for control:

- Control by PLC (E_STW1.Bit10)
- No OFF2(E_STW1.Bit1)
- Operation enabled (E_STW1.Bit3)
- Inhibit infeed motor operation (E_STW1.Bit5)
- Inhibit infeed regenerative operation (E_STW1.Bit6)

The example (Fig 3-11) shows the interconnection of the five signals to be supplied.

Fig. 3-11



- "InFeed1Status".Master_control_requested = "Control requested", ZSW1.Bit9
- "InFeed1Control".Control_via_PLC = "PLC control", STW1.Bit10
- "InFeed1Control".OFF2= "No OFF2", STW1.Bit1
- "InFeed1Control".Enable_Operation= "Enable operation", STW1.Bit3
- "InFeed1Control"."Inhibit motor operation", STW1.Bit5
- "InFeed1Control"."Inhibit regenerative operation", STW1.Bit6

3.9 PLC block sources

3.9.1 PLC block sources SINUMERIK 840Dsl

The three sources supplied include the S7 code for this application:

- UDT1031_ALMB.AWL – source for UDT1031
- FC1031_ALMdata.AWL – source for FC1031
- FC70_ALMcontrol.AWL - example, source for block to control ALM from a PLC

3.9.2 PLC block sources SINUMERIK ONE

- LDrv_Sina370.udt – source for data types LDrv_SinaTlg370....

3.9.3 Link to Drive_Lib_S7_1200_1500 library

<https://support.industry.siemens.com/cs/document/109475044/sinamics-communication-blocks-drivelib-for-reading-and-writing-drive-data-within-tia-portal-context?dti=0&lc=en-VWV>

4 Controlling ALM using block SINA_INFEED

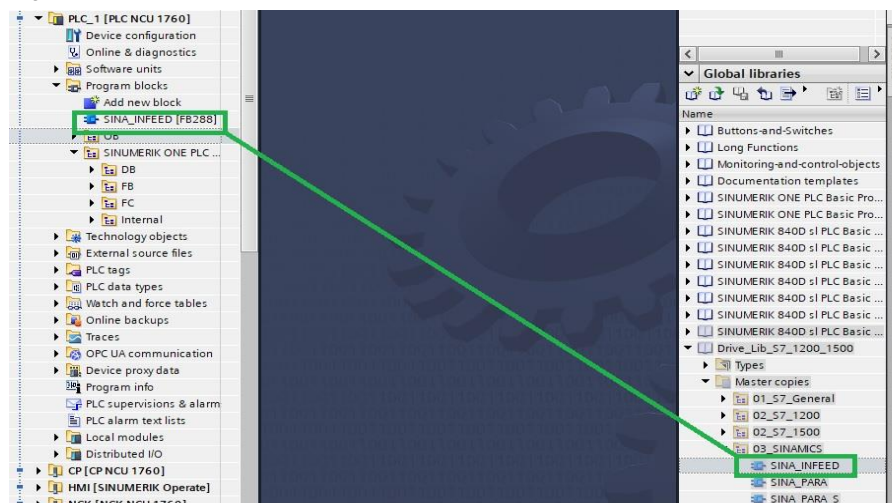
4.1 Controlling ALM using block SINA_INFEED

If block SINA_INFEED is to be used to control ALM, then Drive_Lib_S7_1200_1500 library must first be opened in the TIA Portal. The link to the library is provided in the appendix, Open global library in the TIA Portal see Help for the TIA Portal.

4.1.1 SINA_INFEED block in project block folder

Drag the block from the library and drop in project block folder

Fig. 4-1



4.1.2 Description of the SINA_INFEED block

The block is employed to use and operate a SINAMICS S120 infeed unit. The block only uses control word STW1, and evaluates status word ZSW1 of the infeed unit (standard telegram 370).

The appropriate instance DB is automatically created with the integration of SINA_INFEED (FB288).

Calling OBs

The block can be alternatively inserted in the following OBs:

- Cyclic task: OB1
- Cyclic interrupt OB: e.g. OB32

Called blocks/instructions

DPRD_DAT read consistent data of a standard DP slave

DPWD_DAT write consistent data of a standard DP slave

4.1.3 Function description

The hardware ID of the setpoint slot is specified using input "HWIDSTW" – and the actual value slot is specified using input "HWIDZSW".

By setting input "EnablePrecharging" (STW1.0), the infeed unit can be precharged and using input "EnableInfeed" (STW1.3) it can be switched on (by setting the corresponding control bit in STW1).

The functions are only executed if the infeed unit is in the necessary state (evaluation of the actual ZSW1).

The individual feedback signals (relevant status bits) of the infeed unit and complete status word 1 are output via the block outputs.

In addition to inputs "EnablePrecharging", "EnableInfeed" and "AckError", the user can make additional entries in control word 1 using parameter "ConfigAxis" (standard: 3h). For immediate operation, certain bits in the telegram are preassigned using this input.

Bit "Control requested" (STW1.10) is cyclically set within the block.

4.1.4 Input interface of SINA_INFEED

Table 4-1

Input signal	Type	Default	Meaning
EN	BOOL	1	
EnablePrecharging	BOOL	0	Precharge infeed unit
EnableInfeed	BOOL	0	Switch on infeed unit
AckError	BOOL	0	Acknowledge infeed unit fault
ConfigAxis	WORD	16#0003	For additional information, see Chapter 4.1.5
HWIDSTW	HW_IO	0	Symbolic name or HW ID on the SINUMERIK of the setpoint slot (SetPoint) → see Fig. 3.11
HWIDZSW	HW_IO	0	Symbolic name or HW ID/ on the SINUMERIK of the actual value slot (Actual Value) → see Fig. 3.11

4.1.5 Default setting of input ConfigAxis

Table 4-2

ConfigAxis	Meaning	PZD	Interconnection	Default
Bit0	OFF2	1	r2090.1 = p 844[0]	1
Bit1	Inverter enable	1	r2090.3 = p 852[0]	1
Bit2	1 = inhibit motoring operation	1	r2090.5 = p 3532	0
Bit3	1 = inhibit generator operation	1	r2090.6 = p 3533	0
Bit4- 12	Reserve – can be used as required	1	R2090.2 -15	0

4.1.6 Output interface of SINA_INFEED

Table 4-3

Output signal	Type	Default	Meaning
ENO	BOOL	1	
Ready	BOOL	1	Ready for switch on (ZSW1.0)
Operation	BOOL	0	Ready to operate (ZSW1.1)
Run	BOOL	0	In operation (ZSW1.2)
Fault	BOOL	0	Infeed unit fault (ZSW1.3)
Lockout	BOOL	0	Infeed inhibited (ZSW1.6)
Alarm	BOOL	0	Infeed unit alarm (ZSW1.7)
ZSW1	WORD	16#0	Status word 1
Error	BOOL	0	Error
DiagID	WORD	0	Extended communication error RET_VAL from system functions DPRD_DAT or DPWR_DAT (also see parameter "Status")
Status	WORD	16#0	16#7002: No error active 16#7200: Infeed unit alarm 16#8400: Precharging fault 16#8401: Infeed unit fault 16#8600: Error: DPRD_DAT 16#8601: Error: DPWR_DAT

4.1.7 Troubleshooting function block SINA_INFEED

The "Error" output signals a general error, which can be specified in more detail using the "Status" output.

If inputs "EnablePrecharging" and "EnableInfeed" are set, and the drive signals a fault/error, then the control bits for precharging and switching-on are reset.

If input "EnableInfeed" is set and "EnablePrecharging" is not set, then output "Error" = 1 – and status is set = 16#8400. If input "EnablePrecharging" is again set to 1, then output "Error" is immediately set to 0 again (acknowledgment is not required).

Communication between the SIMATIC CPU and the infeed unit is realized via system blocks "DPRD_DAT" and "DPWR_DAT".

If an error occurs while the system blocks are being executed, then output "Error" is set to 1, and the error message of the system function is output via output "DiagID".

Depending on which system function signals the error, then the "Status" output is set to 16#8600 (DPRD_DAT) or to 16#8601 (DPWR_DAT).

If, for the two system functions, an error is active, then the error message of block DPRD_DAT is first output, and when this is no longer active, then that of DPWR_DAT, assuming that this is still active.

An infeed unit fault is displayed using output "Fault" = 1 and "Status" = 16#8401 – and it can be acknowledged using input "AckError".

An infeed unit alarm is displayed using output "Alarm" = 1 and "Status" = 16#7200. If the block is operating without any errors, then at the output "Status" = 16#7002 is displayed.

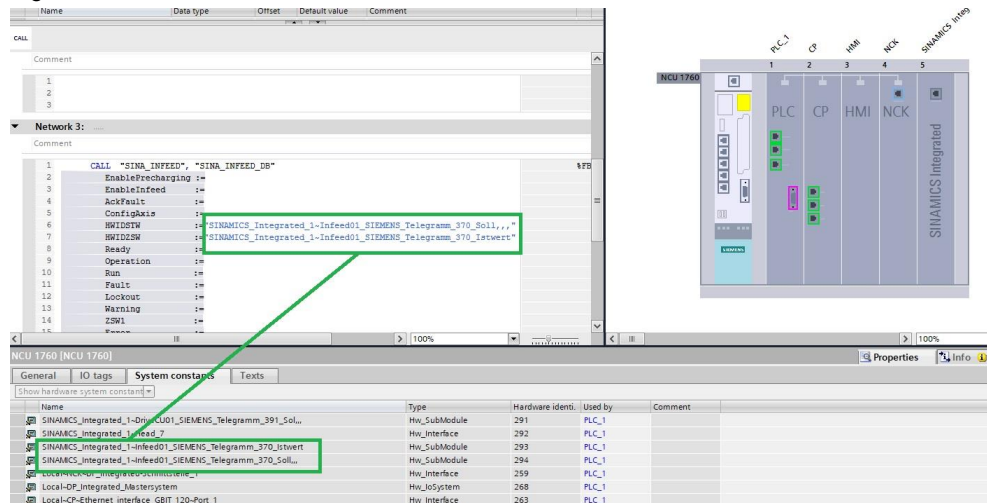
Note

The user must reset input "AckError" again, as the fault acknowledgment expects an edge change (01).

4.1.8 Input signals HWIDSTW and HWIDZTW

Input signals HWISTW and HWIDZTW can be interconnected at the block by dragging and dropping. To do this, system constants of the NCU17XX must be displayed. In Device View, select the NCU, and in the inspector window, the system constants are then displayed under Properties.

Fig. 4-2



5 Additional information

5.1 Structure of telegram 370

Telegram 370 includes:

- A control word (E_STW1)
- A status word (E_ZSW1)

5.2 Signals in telegram 370

5.2.1 E_STW1

The following signals are available in the control word to the ALM.

Table 5-1

Signal	Comment
STW1.0	On
STW1.1	No OFF2
STW1.2	
STW1.3	Enable operation
STW1.4	
STW1.5	Infeed, inhibit motor operation
STW1.6	Infeed, inhibit regenerative operation
STW1.7	Acknowledge fault
STW1.8	
STW1.9	
STW1.10	Control by PLC
STW1.11	
STW1.12	
STW1.13	
STW1.14	
STW1.15	

5.2.2 E_ZSW1

The following signals are available in the control word to the ALM.

Table 5-2

Signal	Comment
ZSW1.0	Ready for switching on
ZSW1.1	Ready
ZSW1.2	Operation enabled
ZSW1.3	Fault active
ZSW1.4	No OFF2 active
ZSW1.5	
ZSW1.6	Switching on inhibited
ZSW1.7	Alarm active
ZSW1.8	
ZSW1.9	Control requested
ZSW1.10	
ZSW1.11	Precharging completed
ZSW1.12	Line contactor closed
ZSW1.13	
ZSW1.14	
ZSW1.15	

5.3 Additional documentation about telegram 370

Detailed information relating to the telegram structure is provided in the Sinamics S120 documentation, List Manual in Chapter "Function diagrams".

- Function diagram 2421 (telegram structure)
- Function diagram 2447 (E_STW1)
- Function diagram 2457 (E_ZSW1)

6 Contact person

Siemens AG
Industry Sector
DF MC MTS APC
Frauenauracher Strasse 80
D-91056 Erlangen, Germany
mailto: MC-MTS-APC-Tech-Team.i-dt@siemens.com

7 History

Table 7-1

Version	Date	Change
V1.0	04/2015	First edition
V2.0	12/2020	Supplement SINUMERIK ONE