25 CO scenes, event 750002

Use of the application program

Product family:	Controllers
Product type:	Controller
Manufacturer:	Siemens
Name:	N 305 scenes-/event controller
Order no.:	5WG1 305-1AB01

Functional description

The N 305 scenes-/event controller is a device for DIN-rail mounting with N-system dimensions 1 SU wide. It needs the application program 25 CO scenes, event 750002. This application includes the scene control and event program functions. Either function can be chosen when parameterisation the controller.
To the load of the application program the Engineering Tool Software (ETS) is needed as of version ETS 3.0e.

Scene control:
A scene is a set of defined switching states or values which will be sent as a result of a scene opening to a number of actuators. This is used, for example, to set the lighting and shading, as well as the screen and projector in a lecture room, for a presentation by a single press of a button.
The scene control feature can program and recall up to eight scenes, in each of which up to eight defined switching states (e.g. lighting on/off, Venetian blinds up/down), or values (e.g. lighting 70%, comfort temperature 22°C) can be controlled. The number of scenes and the groups used in these are set by means of parameters, by which the communication objects used are faded in.

The following types of data for controlling the different functions are available:

- 1-bit switching
- 1-bit Venetian blind
- 1-bit scene
- 2-bit forced control
- 8-bit scene
- 1-byte % value
- 1-byte count value
- 1-byte ASCII character
- 2-byte floating point
- 2-byte count value
- 4-byte floating point
- 4-byte count value
- 6-byte fault information
- 14-byte character string

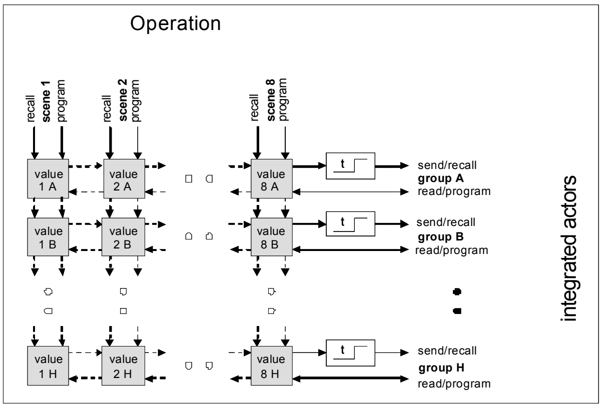
The desired switching states or values which will be sent upon calling up scenes can be initialised using parameters as well as changed, programmed and recalled in real time by the user.
To program a scene during operation, the actuators integrated in this scene are first brought to the desired state (e.g. lighting 70%, Venetian blinds down, etc.) with the keys or control elements provided for this purpose. The controller is then prompted by a corresponding bus telegram to interrogate the actuator states as currently set and to program them under the appropriate scene number. The programmed values are maintained even after a restart of the scenes-/event controller (e.g. after a bus power failure).
If individual scenes or groups are not to be editable during operation, then you can use parameters to set this. The relevant communication objects for interrogating the actuators while programming scenes will then not be created.
The sending of individual group telegrams on calling up a scene can be delayed by a variable time from 0 to 30 minutes.
For groups within a scene, there is an option to use proprietary communication objects or those from another scene. If objects from another scene are used, then the setting overheads are obviously reduced, because when connecting with group addresses via all participating scenes, only one object per group for calling up and one for programming with a group address must be connected in each case. If proprietary sending objects for groups of different scenes are used, combinations are possible in which individual actuators can be dispensed with or added in the different scenes (e.g. the passage lighting is not integrated in the Central-OFF scene, but an alarm message is armed accordingly). It is also possible to bar a scene number completely (in the 1-bit scene opening, 2 scenes are always combined and therefore always present in pairs).
Scenes are called up and programmed either with 1-bit switching commands, in which scenes 1, 3, 5 and 7 are called up and programmed with "1" telegrams and scenes 2, 4, 6 und 8 are called up and programmed with "0" telegrams, or with 8-bit telegrams, in which bits 0 - 5 determine the scene numbers (1 - 64) and the highest value bit 7 defines whether the scene is to be called up (bit = 0) or programmed (bit = 1).
For 8-bit scene control, there is a parameter used to set whether this is to be used and the corresponding communication object displayed.
When using 8-bit scene control, additional parameters are visible, with which you set the scenes with which an 8-bit scene opener is to be used and with which scene numbers (1 - 64) the relevant scene is to be called up or programmed.
Individual groups in scenes can also be defined as 1-bit scene call-ups or 8-bit scene telegrams. This facilitates conversion from 1-bit to 8-bit scene openings and vice versa.
You can use additional scenes-/event controllers to increase the number of scenes or groups within a scene.
Furthermore, the number of groups which will be addressed in a scene will be increased by parallel calling up of two or more scenes in a single scenes-/event controller.

Maximum number of group addresses:	280
Maximum number of assignments:	300

© Siemens AG 2008
Subject to change without further notice

25 CO scenes, event 750002

Scenes control functional diagram



Event programs:
You use the event programs feature to define and process event tasks. An event opening can thus trigger the despatch of several telegrams on the bus, so as to open various functions and make various settings concurrently.

The application program facilitates the creation of eight event programs, in each of which up to ten event telegrams can be opened with different data types.

The following data types are available:
- 1-bit switching
- 1-bit Venetian blind
- 1-bit scene
- 2-bit forced control
- 4-bit relative dimming
- 1 byte % value
- 1-byte count value
- Call up 8-bit scenes
- Program 8-bit scenes
- 1-byte ASCII character
- 2-byte floating point
- 2-byte count value
- 4-byte floating point
- 4-byte count value
- 6-byte fault information
- 14-byte character string

The defaults for the desired values in the individual scenes are set when installing the application parameters with ETS Release (Engineering Tool Software).
The sending of the individual event telegrams on calling up an event program can be delayed by a time variable from 0 to 120 minutes. Processing of the event programs can be triggered downstream.
This means that an event program running in real time, in which not all event tasks have been processed as yet owing to a set transmission delay, is stopped for a new call-up, as yet unprocessed event tasks are discarded and restarted.
You can use bus telegrams with various data types such as 1-bit, 2-bit, 4-bit and 8-bit values, 2 and 4 byte floating point and count values, time, 6-byte fault information and 8-byte system time as event openers.

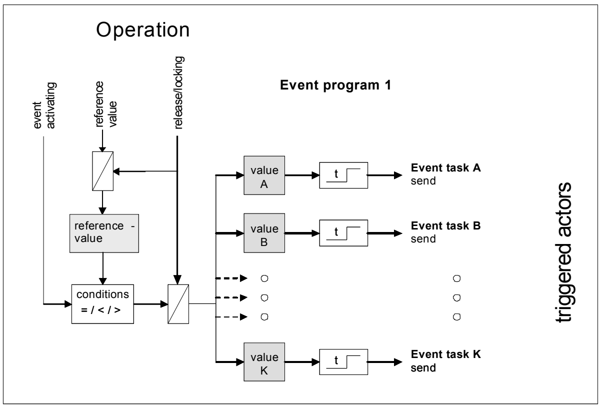
Event programs are opened by means of reference values and opening conditions. The comparator values can be determined when setting the controller, as well as changed during operation and programmed in the controller.
The following comparisons between a received event opener and the corresponding comparator value are available as conditions for opening an event program:
- if a value is received (no condition)
- if value = comparator value
- if value > comparator value
- if value < comparator value
- if comparator value is exceeded
- if comparator value is undershot

For every event program, there is a parameter with which you can set whether the program is always enabled or whether it can be enabled and disabled with a switching command. In this way, you can call up two or more event programs alternately.
A further parameter determines whether enable/disable is also used for exceeding the comparator value of a temporarily disabled event program.
The behaviour after a restart (e.g. after a bus power outage) for the enable states is variable.
For event tasks within an event program, there is an option to use proprietary communication objects or those from another scene. If the same groups or actuators are addressed by different event programs, then the setting overheads are obviously reduced by using objects from a single event program, because when connecting with group addresses in a number of event programs, only one communication object must be connected with a group address for each event task in each case.

Maximum number of group addresses:	280
Maximum number of assignments:	300

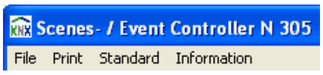
25 CO scenes, event 750002

Functional diagram of an event program



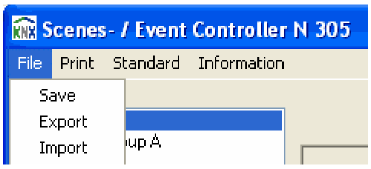
N 305 scenes-/event controller plug-in

The next picture shows the parameter window which pops up after you choose the ETS function "Edit parameters...".

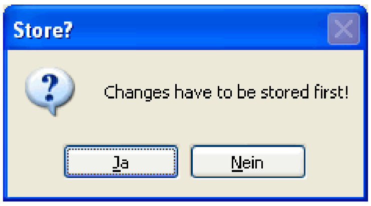


File

If you select the "File" tab in the header, then you can choose one of the following actions:

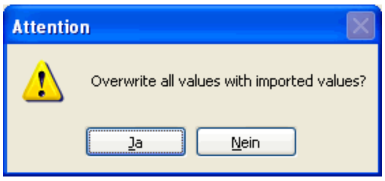


Program: Choosing this action programs the currently set parameters into the project database.
Export: The export function is used to export the device's current settings/group addresses in XML format from the ETS project database and save them on any drive in a file to be defined by the operator. The following window pops up after you select this action:



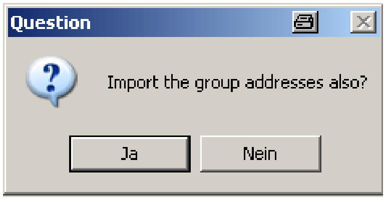
Changes have to be stored first!

At this point, you should remember that all final parameter settings entered and group addresses allocated are to be saved initially in the project database before exporting the data. Only when this prompt has been answered with "Yes" does a new window open for selecting the drive and file to which the file export is to be made.
Import: The import function is used to load the programmed settings/group addresses for an N 305 scenes-/event controller into another device via the export function. Settings/group addresses can be "copied" into additional devices in this way. The following window pops up after you select this action:



Overwrite all values with imported values?

Only when this prompt has been answered with "Yes" does a new window open for selecting the drive and the *.xml file to be imported. After the file to be imported has been selected, the following window prompts you to import the group addresses allocated as well.

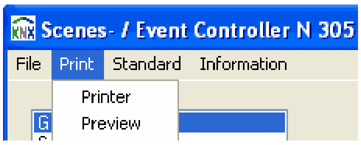


Import the group addresses also?

If you answer "Yes" to this prompt, the group addresses are imported.

Printing

If you select the "Print" tab in the header, then you can choose one of the following actions:

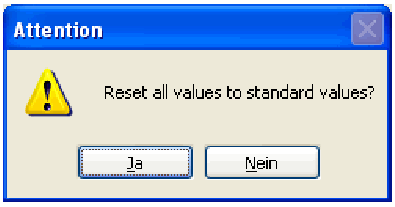


Printer: After you select "Printer", a window opens for you to choose the printer on which the device settings are to be printed out for documentation purposes.
Preview: After selecting "Preview", a window opens with a print view of the device parameters.

25 CO scenes, event 750002

Standard

After selecting this button, the following window opens:



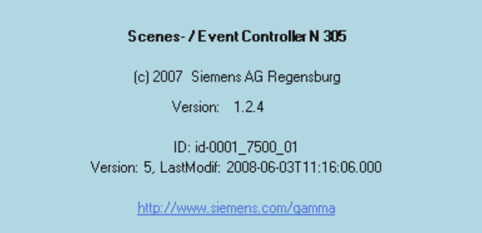
Attention

Reset all values to standard values?

If you press the "Yes" button, then all parameters are reset to their standard factory setting. This causes the loss of all settings that have not been archived via the export function.

Info

After this button is selected, a window opens with information about the current release of the application program and an Internet link to Siemens Building Systems Engineering.
Clicking on the Open Info window closes it.



Scenes- / Event Controller N 305

(c) 2007 Siemens AG Regensburg

Version: 1.2.4

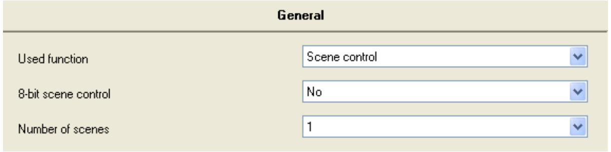
ID: id-0001_7500_01
Version: 5, LastModif: 2008-06-03T11:16:06.000

http://www.siemens.com/gamma

"Used Function" parameter

You use the "General" parameters page to select the desired N 305 scenes-/event controller function.

General



General

Used function: Scene control

8-bit scene control: No

Number of scenes: 1

Parameter	Settings
Used Function	**Scenes control** Event program
This parameter determines whether the scenes control or event program is to be used. Note: Subsequent changes to this parameter leads to losing the group addresses in the relevant objects.	

Communication objects scenes control

The following communication objects will be displayed in the N 305 scene-/event controller if scene control has been selected in the "General" parameters window and all functions have been enabled.

Obj	Object name	Function	Type	Flag
1	Scene 1 group A send/recall	Object	1 bit	Send CRTU
1	Scene 1 group A send/recall	Object	2 bits	Send CRTU
1	Scene 1 group A send/recall	Object	1 byte	Send CRTU
1	Scene 1 group A send/recall	Object	2 bytes	Send CRTU
1	Scene 1 group A send/recall	Object	4 bytes	Send CRTU
1	Scene 1 group A send/recall	Object	6 bytes	Send CRTU
1	Scene 1 group A send/recall	Object	14 bytes	Send CRTU
On calling up scene 1, this object sends the value programmed for this purpose to group A. You use the "data type" parameter for this scene and group to select the desired object type. If the setting is "not used" this object is not displayed. According to the data type chosen, this object is to be linked via the appropriate group address with the corresponding actuator object (e.g. switching, Venetian blind up/down, forced control, brightness value in %, floating point value, count value, character string) for the device to be controlled with this group. The currently programmed Scene 1, Group A value can be read out for controlling via this object. Note: The "14 bytes character string" is always present only in the relevant group A of scenes 1 to 8. The "6 bytes fault information string" is always present only in the relevant group B of scenes 1 to 8.				
2	Scene 1 group A read/program	Object	1 bit	Read CWTU
2	Scene 1 group A read/program	Object	2 bits	Read CWTU
2	Scene 1 group A read/program	Object	1 byte	Read CWTU
2	Scene 1 group A read/program	Object	2 bytes	Read CWTU
2	Scene 1 group A read/program	Object	4 bytes	Read CWTU
2	Scene 1 group A read/program	Object	6 bytes	Read CWTU
2	Scene 1 group A read/program	Object	14 bytes	Read CWTU

25 CO scenes, event 750002

Obj	Object name	Function	Type	Flag
About this object the current value of the group of A is questioned when programming scene 1. The data type is identical to the setting made for the "Scene 1 Group A" object. If this is set to "No" when choosing "object read/program use", this object is not displayed. According to the data type chosen, this object is to be linked via the appropriate group address with the corresponding status object (e.g. switching, Venetian blind up/down, forced control, brightness value in %, floating point value, count value, character string) for the device to be read out. To read out a current value or status and program it into the scenes-/event controller, the group address is to be entered as "sending address" in the object of the device to be interrogated and the read flag is to be set. Note: The "14 bytes character string" is always present only in the relevant group A of scenes 1 to 8. The "6 bytes fault information string" is always present only in the relevant group B of scenes 1 to 8.				

The communication objects 3 - 16 have the relevant meaning for Scene 1, Groups B to H.

The communication objects 17 - 128 have the relevant meaning for Scenes 2 to 8, Groups A to H. These objects are only present if proprietary objects are to be used conditionally for calling up and programming through the parameter setting for the corresponding scenes.

Obj	Object name	Function	Type	Flag
129	Scenes 1/2 recover	recall	1 bit	Received CWTU
This object calls up Scenes 1 and 2. If a "0" telegram is received, the scenes-/event controller sends the programmed scene 1 values via the "send/recall group" objects to the addressed actuators. Scene 2 is called up correspondingly on receiving a "1" telegram.				

The communication objects 130 - 132 have the corresponding meaning for scenes 3/4, 5/6 and 7/8.

Obj	Object name	Function	Type	Flag
133	Scenes 1/2 program	program	1 bit	Received CWTU
This object programs Scenes 1 and 2. If a "0" telegram is received, the scenes-/event controller reads out the currently set statuses or values of the corresponding actuators via the "read/program group" objects and programs these in scene 1. Scene 2 is programmed correspondingly on receiving a "1" telegram.				

The communication objects 134 - 136 have the corresponding meaning for scenes 3/4, 5/6 and 7/8.

Obj	Object name	Function	Type	Flag
137	8-bit scene	read/ program	1 byte	Received CWU
This object calls up or programs the scene for which a telegram with the value (1-64) is received, which is set for this scene in the parameter "Scene number at 8-bit scene control".				

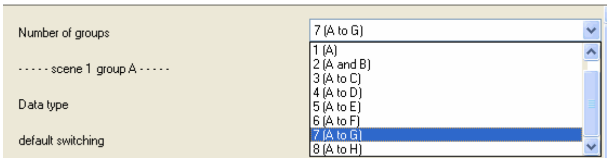
Scene control parameters

General

Parameter	Settings
8-bit scene control	**No** Yes
You use this parameter to set whether the scenes are also to be called up and programmed via 8-bit scene openers. If so, the communication object "8-bit scene" will be created, as well as a parameter for selecting the scene number, with which this is addressed for each scene.	
Number of scenes	**1** 2 3 4 5 6 7 8
This parameter determines the number of scenes which are to be used in the scenes-/event controller. This selection will fade in the parameter pages for setting and the communication objects for calling up and programming approved scenes.	

25 CO scenes, event 750002

Scene 1 Group A...



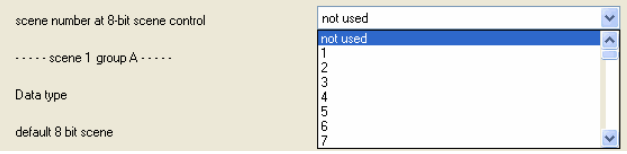
Number of groups: 7 (A to G)

----- scene 1 group A -----

Data type

default switching

Parameter	Settings
Number of groups	none **1 (A)** 2 (A and B) 3 (A to C) 4 (A to D) 5 (A to E) 6 (A to F) 7 (A to G) 8 (A to H)
This parameter determines the number of groups used in scene 1. This selection creates additional parameters for initialising default settings and communication objects for the groups used. If you set "none", scene 1 is not used.	
Scene number at 8-bit scene control	**not used** 1 2 64
You use this parameter to set whether 8-bit scene control is used for scene 1 and with which scene number (1-64) it is called up and programmed. Note: All scenes to be called up must be called up or programmed under different scene numbers. If several scenes are set with the same scene number for calling up or programming scenes, only the first program will be called up or programmed.	



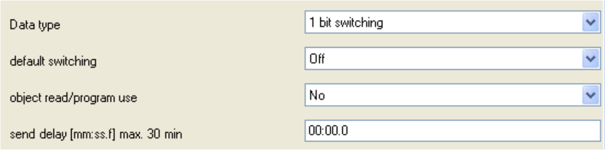
scene number at 8-bit scene control: not used

----- scene 1 group A -----

Data type

default 8 bit scene

Scene 1 Group A...



Data type: 1 bit switching

default switching: Off

object read/program use: No

send delay [mm:ss.f] max. 30 min: 00:00.0

Parameter	Settings
Data type	not used **1 bit switching** 1 bit Venetian blind 1 bit scene 2 bit forced control 8-bit scene 1 byte % value 1 byte count value 1 byte ASCII character 2 byte floating point 2 byte count value 4 byte floating point 4 byte count value 6 byte fault information 14 byte character string
This parameter determines the data type for the "send/recall up Group A" and "read/program Group A" communication objects. If the setting is "not used", this group is not used in scene 1 and the corresponding parameters and communication objects are faded out. This is how you can delete or add individual groups in different scenes. Note: The "14 bytes character string" is always present only in the relevant group A of scenes 1 to 8. The "Fault information" data type is always present only in the relevant group B of scenes 1 to 8. On bus power recovery "send/recall up Group A" communication objects are set to zero.	

Different value ranges are allowed for the following parameters, depending on the group data type set.

25 CO scenes, event 750002

Scene 1 Group A...

Parameter	Settings
default switching	**Off** On
default venetian blind	**Up** Down
default 1-bit scene	**Scene A** Scene B
default forced control	**Forced control Off (0)** Force controlled to Off (2) Force controlled to On (3)
default 8-bit scene	**1** 2 64
default value in %	**0**
default 1-byte count value	**0**
default ASCII characters	**0**
default 2-byte floating point	**0**
default 2-byte count value	**0**
default 4-byte floating point	**0**
default 4-byte count value	**0**
default 6-byte fault information	**0x00:0:00:00:00:00**
default 14-byte character string	**0**

Parameter	Settings

This parameter sets the initial Group A default value for scene 1. This is how you plan scene settings and use these directly after starting up the scene-/event controller. The value is held in the controller until it is overwritten when programming scenes by reading out the current status of the linked actuator.
If the communication object "recall/program Group A"
is not used, or is not populated with a group address, which can be interrogated via the bus, the preset value here is maintained when programming scenes.
The value is maintained even after a bus power outage.

Note:
Entries are limited to counts in the maximum possible value range in each case:

Value in %	0 to 100
1-byte count value	0 to 255
1-byte ASCII character	Entry (a character)
2-byte floating point	-671,088.64 to 671,088.96
2-byte count value	0 to 65,535
4-byte floating point	-3.40 E38 to 3.40 E38
4-byte count value	0 to 4,294,967,295
6-byte fault information	to KNX specification, DPT_Alarminfo
14-byte character string	Text input, max. 14 characters

Scene A (corresponds to scenes 1, 3, 5 and 7 and is called up via the value "0".
Scene B (corresponds to scenes 2, 4, 6 and 8 and is called up via the value "1".

Note:
If a value is sent to the object "read/program Group A" (e.g. by using the ETS, an external display-/control unit or a visualisation) and scene 1 is then programmed, without the temporary value of the object having been overwritten (because the object "read/program Group A" is not linked with the status object of the actuator), this value will be programmed as the new scene value and sent in future when the scene is called up.
This is how you can program a new scene value without moving the actuator to this status in advance and reading it out to program the scene.

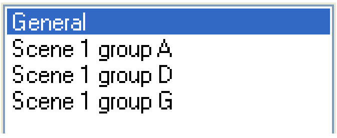
Scene 1 Group A...

Parameter	Settings
object read/program use	**No** Yes
You use this parameter to set whether the object "read/program Group A" is to be created. If so, the communication object is created. You use this communication object to read and program the current state of the status object, the actuator, in each case. Note: When using the "8-bits scene" data type, the parameter setting "Use read/program object" does not pop up. In an 8-bit scene, you call up and program a scene with the "read/program" communication object. The highest value bit 7 states whether the scene is called up (bit = 0) or programmed (bit = 1). On bus power recovery, "read/program object" is set to zero.	

Scene 1 Group A...

Parameter	Settings
send delay [mm:ss.f] max. 30 min	**00:00.0**
This parameter determines now much time should elapse between the scene opening and sending of the scene 1 group A value. This is how you can comply with a given sequence when sending group telegrams on calling up a scene. This is how you can produce dynamic functions, such as staircase lighting control with preliminary warning, by linking an actuator with various groups in a scene. If a scene is called up again, then processing is suspended before all values have been sent owing to the time delay. All time delays will be reset and restarted. Note: The time input is limited to a value range between 0.1 seconds and 30 minutes.	

Depending on the number of groups to be used, the settings for these groups are displayed in a maximum of 3 windows, Scene 1 Group A..., Scene 1 Group D... and Scene 1 Group G...



General
Scene 1 group A
Scene 1 group D
Scene 1 group G

Scene 1 Group A...

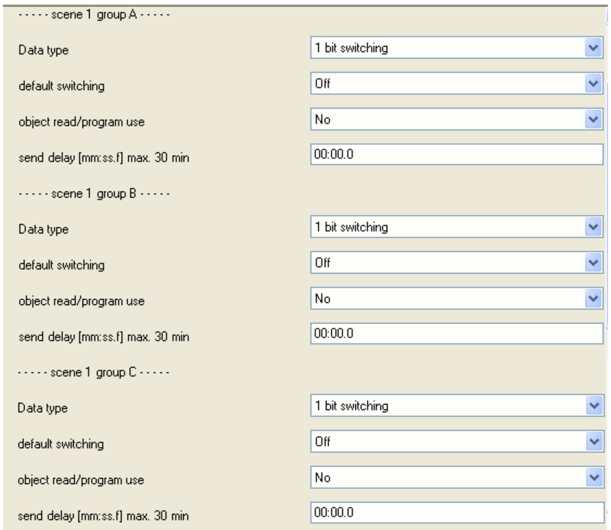
You use Scene 1 Group A... to select settings up to Group C.

Scene 1 Group D...

You use Scene 1 Group D... to select settings up to Group F.

Scene 1 Group G...

You use Scene 1 Group G... to select settings up to Group H.



----- scene 1 group A -----

Data type: 1 bit switching
default switching: Off
object read/program use: No
send delay [mm:ss.f] max. 30 min: 00:00.0

----- scene 1 group B -----

Data type: 1 bit switching
default switching: Off
object read/program use: No
send delay [mm:ss.f] max. 30 min: 00:00.0

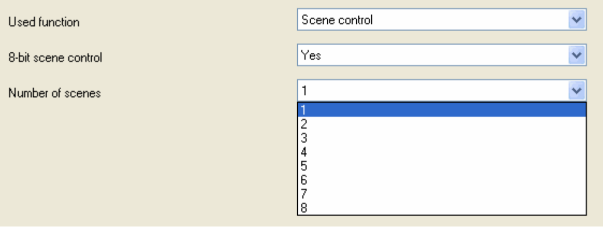
----- scene 1 group C -----

Data type: 1 bit switching
default switching: Off
object read/program use: No
send delay [mm:ss.f] max. 30 min: 00:00.0

The parameters for groups B to H are identical with those for Group A. They will only be displayed if they have been enabled via the "Number of groups" parameter.

Note:
The "14 bytes character string" is always present only in the relevant group A of scenes 1 to 8.
The "Fault information" data type is always present only in the relevant group B of scenes 1 to 8.

Scenes 1...8



Used function: Scene control

8-bit scene control: Yes

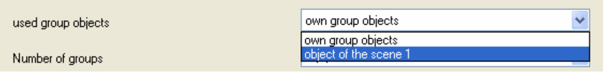
Number of scenes: 1

You use the parameter "Number of scenes" to set how many scenes are to be used. The communication objects and parameters will be visible according to the number of scenes set.
The parameters for scenes 1 and 5, for the relevant groups, are identical.

To reduce the setting overheads, there is an option, to be determined for each enabled scene, whether proprietary sending objects or sending objects for the first allocated scene range are to be used.
For this purpose, 8 scenes are divided into 2 scene ranges. Each scene range can be allocated a maximum of 4 scenes. You can allocate scenes 2, 3 and 4 to scene 1. Scene 1 is therefore the first scene range. You can allocate scenes 6, 7 and 8 to scene 5. Scene 5 is therefore the first scene range.

Note:
For all scenes which are allocated to a scene range, only one "send/read object" and one "read/program object" are available and these must be linked with a group address.
If proprietary sending objects are used for individual scenes, combinations are possible in which, in the various scenes, individual actuators can be dispensed with or added, or a scene number can be disabled completely. In 1-bit scene opening, 2 scenes are always combined and are therefore always present in pairs.

Accordingly, scenes 2, 3 and 4 have an additional parameter:

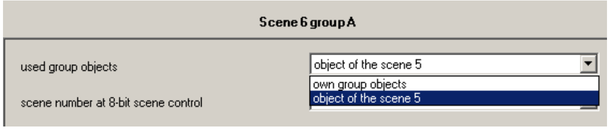


used group objects: own group objects

Number of groups

Parameter	Settings
used group objects	**own group objects** object of the scene 1
This parameter determines whether proprietary sending objects are to be produced for the selected groups in each case or whether scene 1 sending objects are to be used for scenes 2, 3 and 4.	

Accordingly, scenes 6, 7 and 8 have an additional parameter:



Scene 6 group A

used group objects: object of the scene 5

scene number at 8-bit scene control

Parameter	Settings
used group objects	**own group objects** object of the scene 5
This parameter determines whether proprietary sending objects are to be produced for the selected groups in each case or whether scene 5 sending objects are to be used for scenes 6, 7 and 8.	

General planning note:
In theory, a large number of random channels can be addressed via a single scene by using scenes with the same opener in different controllers. You should therefore note that, at the same time, the bus loading through sending individual scene values does not become too great, because otherwise the bus would be blocked temporarily or other information.

Event program communication objects

The following communication objects will be displayed in the N 305 scene-/event controller if event program has been selected in the "General" parameters window and all functions have been enabled.

Obj	Object name	Function	Type	Flag
1	EP 1: event opening	Object	1 bit	Received CWTU
1	EP 1: event opening	Object	2 bits	Received CWTU
1	EP 1: event opening	Object	4 bits	Received CWTU
1	EP 1: event opening	Object	8 bits	Received CWTU
1	EP 1: event opening	Object	1 byte	Received CWTU
1	EP 1: event opening	Object	2 bytes	Received CWTU
1	EP 1: event opening	Object	3 bytes	Received CWTU
1	EP 1: event opening	Object	4 bytes	Received CWTU
1	EP 1: event opening	Object	6 bytes	Received CWTU
1	EP 1: event opening	Object	8 bytes	Received CWTU

Use this object to call up event program 1. On receiving a telegram with a value for event opening or on receiving a telegram which fulfils the condition when compared with the currently valid comparator value and the event program is approved, the controller sends the set event task values for event program 1 to the addressed actuators via the objects "Event task A" to "Event task K".
You use the "Event opening / reference value data type" parameter to select the desired object type.

Note:
The difference between calling up an 8-bit scene and programming an 8-bit scene is that when calling up an 8-bit scene the highest value bit is set to 0, whereas when programming an 8-bit scene the highest value bit is set to 1.
This fact should be borne in mind when comparing the event opening with the comparator value.

Obj	Object name	Function	Type	Flag
2	EP 1: reference value	Object	1 bit	Received CRWTU
2	EP 1: reference value	Object	2 bits	Received CRWTU
2	EP 1: reference value	Object	4 bits	Received CRWTU
2	EP 1: reference value	Object	8 bits	Received CRWTU
2	EP 1: reference value	Object	1 byte	Received CRWTU
2	EP 1: reference value	Object	2 bytes	Received CRWTU
2	EP 1: reference value	Object	3 bytes	Received CRWTU
2	EP 1: reference value	Object	4 bytes	Received CRWTU
2	EP 1: reference value	Object	6 bytes	Received CRWTU
2	EP 1: reference value	Object	8 bytes	Received CRWTU

The reference value is received via this object and it is then compared with the event opening value to decide whether the event program is to be called up. If no value has been received, the set reference value is used.
You use the "Event opening / reference value data type" parameter to select the desired object type.
A received reference value is programmed if the voltage fails and after the power recovery, the voltage is preset for a fresh comparison.
The currently set comparator value can be read out for checking. The read flag must be set for this purpose.
Note:
If the object is interrogated actively after the ETS download, then the interrogated value will not equal the currently applicable value.
This is only valid after the object is updated.

The communication objects:

17 ≙ EP 2: Event opening

33 ≙ EP 3: Event opening

49 ≙ EP 4: Event opening

65 ≙ EP 5: Event opening

81 ≙ EP 6: Event opening

97 ≙ EP 7: Event opening

113 ≙ EP 8: Event opening
and have the same meaning as in EP 1: Event opening.

The communication objects:

18 ≙ EP 2: Reference value

34 ≙ EP 3: Reference value

50 ≙ EP 4: Reference value

66 ≙ EP 5: Reference value

82 ≙ EP 6: Reference value

98 ≙ EP 7: Reference value

114 ≙ EP 8: Reference value
and have the same meaning as in EP 1: Reference value.

25 CO scenes, event 750002

Obj	Object name	Function	Type	Flag
3	EP 1: release/locking	Object	1 bit	Received CWT
Use this object to release or locking the event program. If the event program is locking, the event telegrams A to K are not sent receiving an event opening telegram, even if the comparison is positive. The release or locking EP status is programmed if the voltage fails and is preset after the power recovery. Note: If the object is interrogated actively after the ETS download, then the interrogated value will not equal the currently applicable value. This is only valid after the object is updated. The read flag must be set for this purpose.				

The communication objects:

19 ≙ EP 2: Release/Locking

35 ≙ EP 3: Release/Locking

51 ≙ EP 4: Release/Locking

67 ≙ EP 5: Release/Locking

83 ≙ EP 6: Release/Locking

99 ≙ EP 7: Release/Locking

115 ≙ EP 8: Release/Locking
and have the same meaning as in EP 1: Release/Locking.

Obj	Object name	Function	Type	Flag
4	EP 1: event task A	Object	1 bit	Send CTU
4	EP 1: event task A	Object	2 bits	Send CTU
4	EP 1: event task A	Object	4 bits	Send CTU
4	EP 1: event task A	Object	1 byte	Send CTU
4	EP 1: event task A	Object	8 bits	Send CTU
4	EP 1: event task A	Object	2 bytes	Send CTU
4	EP 1: event task A	Object	4 bytes	Send CTU
4	EP 1: event task A	Object	6 bytes	Send CTU
4	EP 1: event task A	Object	14 bytes	Send CTU
Use this object to send the event task A value when the event program opens. You use the "Data type" parameter for this event task to select the desired object type. Depending on the data type selected, this object is to be linked via group address with the corresponding actuator object of the device to be controlled with this event task. Note: The "14 bytes character string" is always present only in the relevant event task A of event programs 1 to 8. The "6 bytes character string" is always present only in the relevant event task B of event programs 1 to 8.				

The communication objects:

5-13 ≙ EP 1: Event tasks B-K

20-29 ≙ EP 2: Event tasks A-K

36-45 ≙ EP 3: Event tasks A-K

52-61 ≙ EP 4: Event tasks A-K

68-77 ≙ EP 5: Event tasks A-K

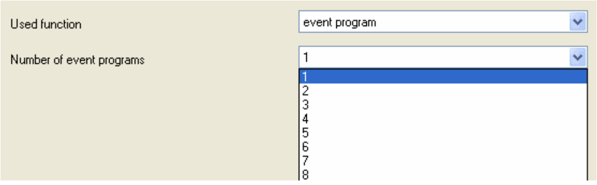
84-93 ≙ EP 6: Event tasks A-K

100-109 ≙ EP 7: Event tasks A-K

116-125 ≙ EP 8: Event tasks A-K
and have the same meaning as in EP 1: Event task A. These objects are only present if proprietary objects are used to send the corresponding event tasks if conditioned by the parameter settings.

Event program parameters

General

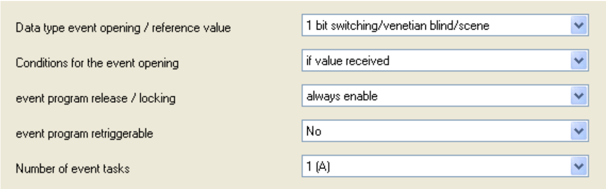


Used function: event program

Number of event programs: 1

Parameter	Settings
Number of event programs	**1** 2 3 4 5 6 7 8
This parameter determines the number of event programs which are to be used in the scene-/event controller. Its selection fades in the parameter pages for setting and the communication objects for the corresponding event programs.	

Event program 1



Data type event opening / reference value: 1 bit switching/venetian blind/scene

Conditions for the event opening: if value received

event program release / locking: always enable

event program retriggerable: No

Number of event tasks: 1 (A)

25 CO scenes, event 750002

Parameter	Settings
Data type event opening / reference value	**1 bit switching/venetian blind/scene** 2 bit forced control 4 bit dimming 8 bit scenes recall 8 bit scenes program 1 byte value % 1-byte count value 2 byte floating point 2 byte count value 3 byte time 4 byte floating point 4 byte count value 6 byte fault information 8 byte system time
This parameter determines the data types for the communication objects "EP 1: Event opening" and "EP 1: Reference value". There is accordingly the possibility of using any bus telegrams with different values to open event programs.	

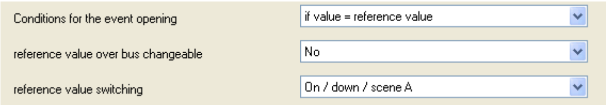
Event program 1

Parameter	Settings

Condition for the event opening	**if value received**
	if value = reference value
	if value > reference value
	if value < reference value
	if reference value is exceeded
	if reference value is undershot

This parameter defines the condition with which the event program is to be called up. The value of a telegram received on the object "EP 1: Event opening" will be compared with the current reference value according to the parameter setting. If the condition is fulfilled and the EP is enabled, this will be run. Note: If the 6-byte fault information data type is used, the settings "if reference value is exceeded" and "if reference value is undershot" are not available.	

Event program 1



Conditions for the event opening: if value = reference value

reference value over bus changeable: No

reference value switching: On / down / scene A

Parameter	Settings
reference value over bus changeable	**No** Yes
This parameter defines whether the set reference value can be changed during operation via the object "EP 1: Reference value". If the setting is "No", this object is not available. A received reference value is programmed if the voltage fails and after the power recovery, the voltage is preset for a fresh comparison. Note: There is accordingly the possibility to change the reference value during operation via an external display-/control unit or viewing via the bus.	

Event program 1

Parameter	Settings
reference value switching	**On/down/scene A** Off/up/scene B
reference value forced control	**Forced control Off (0)** Force controlled to Off (2) Force controlled to On (3)
reference value dimming relative	**Stop (0)** 100% darker (1) ½ darker (2) 1/4 darker (4) 1/8 darker (4) 1/16 darker (5) 1/32 darker (6) 1/64 darker (7) 100% brighter (9) ½ brighter (10) 1/4 brighter (11) 1/8 brighter (12) 1/16 brighter (13) 1/32 brighter (14) 1/64 brighter (15)
reference value scene number	**1,** 2, ..., 64
reference value in %	**0**
reference value count value	**0**
reference value floating point	**0**
reference value time	**00:00:00**
reference value fault information **AlarmPriority: ApplicationArea: ErrorClass**	**0x0:00:00**
Reference value system time hh:mm:ss	**00:00:00**

© Siemens AG 2008
Subject to change without further notice

25 CO scenes, event 750002

Parameter	Settings

Use this parameter to initialise the default reference value with which a value received via the object "EP 1: Event opening" will be compared.
The reference value data type is the same as the event opening data type.
If the condition is fulfilled and the EP is enabled, this will be run.
The set reference value can be changed via the object "EP 1: reference value".
Note:
Entries are limited to counts in the maximum possible value range in each case:

Value in %	0 to 100
1 byte count value	0 to 255
2 byte floating point	-671,088.64 to 671,088.96
2 byte count value	0 to 65,535
3 byte time	00:00:00 to 23:59:59
4 byte floating point	-3.40 E38 to 3.40 E38
4 byte count value	0 to 4,294,967,295
6-byte fault information	to KNX specification, DPT_Alarminfo
8-byte system time	to KNX specification

Alarminfo:

AlarmPriority:	0x03 ≥ will not be analysed
ApplicationArea	0xFF = will not be analysed
ErrorClass	0xFF = will not be analysed

ApplicationArea and ErrorClass will be analysed for the event opening conditions only “via Value = Reference value".

Event program 1

Parameter	Settings
event program release / locking	**always enable** enabled at object value = 0 enabled at object value = 1

This parameter determines whether the event program for running is always enabled or whether it can be enabled or disabled via the object "EP 1: release / locking".

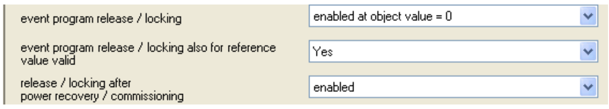
With the setting "enabled at object value = 0" the EP is enabled on receiving a "0" and disabled on receiving a "1".

With the setting "enabled at object value = 1" the EP is enabled on receiving a "1" and disabled on receiving a "0".

With the selection "always enabled", the EP is permanently enabled and the object "EP 1: Release/Locking" and the parameters "Event program Release/Locking also valid for reference value" and "Release/Locking after power return/start-up" are not available.

Note:
There is accordingly the possibility to enable or disable Release/Locking during operation via an external display-/control unit or viewing via the bus.

Event program 1



event program release / locking: enabled at object value = 0
event program release / locking also for reference value valid: Yes
release / locking after power recovery / commissioning: enabled

Parameter	Settings
Event program release / locking also for reference value valid	**Yes** No

This parameter determines whether the parameter "Event program Release/Locking" is also valid for the comparator value.
If the setting is "No", the reference value can always be updated, if the setting is "Yes", the reference value is ignored if disable is set.

Event program 1

Parameter	Settings
release / locking after power recovery / commissioning	**enabled** disabled how before power recovery / release how before power recovery / locked

This parameter determines whether the event program for running after a power recovery and after start-up is enabled or disabled.
With the setting "how before power recovery/release", the status is set after a power recovery and its return to that extant before the power failure. The event program is enabled after start-up or after downloading.
With the setting "how before power recovery/locked", the status is set after a power recovery and its return to that extant before the power failure. The event program is disabled after start-up or after downloading.

Event program 1

Parameter	Settings
event program retriggerable	**No** Yes

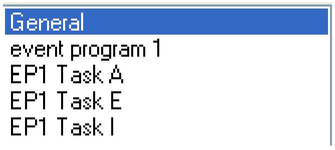
This parameter determines whether a running event program is stopped and restarted on receiving a fresh event opening and fulfilled condition during processing, or whether the fresh event opening is ignored and all event tasks are processed first.
This works if the event tasks in an EP are sent with different delay times.
If the openable retriggerable setting is "Yes", the running EP, in which not all event tasks are processed yet owing to a set sending delay, is stopped and restarted in the event of a fresh call-up.

© Siemens AG 2008
Subject to change without further notice

Event program 1

Parameter	Settings
Number of event tasks	**1 (A)** 2 (A and B) 3 (A to C) 4 (A to D) 5 (A to E) 6 (A to F) 7 (A to G) 8 (A to H) 9 (A to I) 10 (A to K)
This parameter determines the number of event tasks which are used in event program 1. If this is selected, additional parameters and communication objects are created for the relevant event tasks.	

Event program 1



Depending on the number of event tasks to be used, these task settings will be displayed in a maximum of 3 windows,
EP1: Task A..., EP1: Task E..., EP1: Task I...

EP1: Task A...

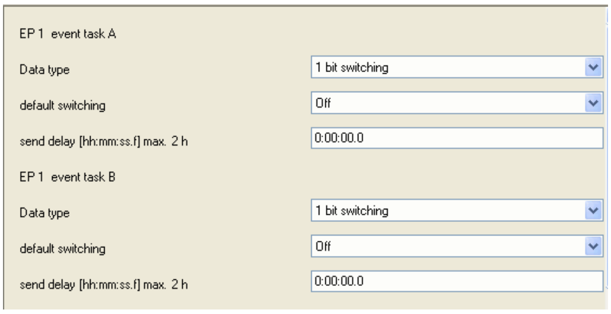
Select EP1: Task A... to select the settings up to task D.

EP1: Task E...

Select EP1: Task E... to select the settings up to task H.

EP1: Task I...

Select EP1: Task I... to select the settings up to task K.



EP 1 event task A

Data type: 1 bit switching

default switching: Off

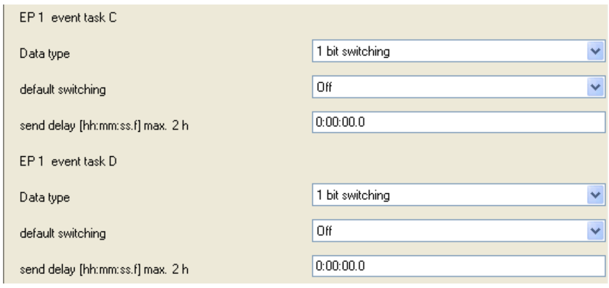
send delay [hh:mm:ss.f] max. 2 h: 0:00:00.0

EP 1 event task B

Data type: 1 bit switching

default switching: Off

send delay [hh:mm:ss.f] max. 2 h: 0:00:00.0



EP 1 event task C

Data type: 1 bit switching

default switching: Off

send delay [hh:mm:ss.f] max. 2 h: 0:00:00.0

EP 1 event task D

Data type: 1 bit switching

default switching: Off

send delay [hh:mm:ss.f] max. 2 h: 0:00:00.0

Parameter	Settings
Data type	**1 bit switching** 1 bit venetian blind 1 bit scene 2 bit forced control 4 bit dimming 1 byte value % 1 byte count value 8-bit scene recall 8-bit scene program 1 byte ASCII character 2 byte floating point 2 byte count value 4 byte floating point 4 byte count value 6 byte fault information 14 byte character string
This parameter determines the data type for the communication object "EP 1: Event task A". This provides the option to send a bus telegram with any value. Note: The "14 bytes character string" is always present only in the relevant task A of event programs 1 to 8. The "Fault information" data type is always present only in the relevant task B of event programs 1 to 8.	

Different value ranges are allowed for the following parameters, depending on the task data type set.

25 CO scenes, event 750002

EP1: Task A...

Parameter	Settings
default switching	**Off** On
default venetian blind	**Up** Down
default 1-bit scene	**Scene A** Scene B
default forced control	**Forced control Off (0)** Force controlled to Off (2) Force controlled to On (3)
default dimming relative	**Stop (0)** 100% darker (1) 1/2 darker (2) 1/4 darker (3) 1/8 darker (4) 1/16 darker (5) 1/32 darker (6) 1/64 darker (7) 100% brighter (9) 1/2 brighter (10) 1/4 brighter (11) 1/8 brighter (12) 1/16 brighter (13) 1/32 brighter (14) 1/64 brighter (15)
default value in %	**0**
default count value	**0**
default 8 bit scene	**1** **2** **...** **64**
default ASCII characters	**0**
default floating point	**0**
default count value	**0**
default 4-byte floating point	**0**
default 4-byte count value	**0**
default 6-byte fault information	**0x00:0:00:00:00:00**
default 14-byte character string	**0**

Parameter	Settings

Use this parameter to set the initial default value of event task A in event program 1.
The value is maintained even after a bus power failure and cannot be changed via the bus.

Note:
Entries are limited to counts in the maximum possible value range in each case:

Value in %	0 to 100
1-byte count value	0 to 255
1-byte ASCII character	Entry (a character)
2-byte floating point	-671,088.64 to 671,088.96
2-byte count value	0 to 65,535
4-byte floating point	-3.40 E38 to 3.40 E38
4-byte count value	0 to 4,294,967,295
6-byte fault information	to KNX specification, DPT_Alarminfo
14-byte character string	Text input, max. 14 characters

Note:
The "14 bytes character string" is always present only in the relevant task A of event programs 1 to 8.
The "Fault information" data type is always present only in the relevant task B of event programs 1 to 8.

EP1: Task A...

Parameter	Settings
send delay [hh:mm:ss.f] max. 2 h	**0:00:00.0**

This parameter determines now much time should elapse between the event opening and sending of event task A.
This is how you can determine a given sequence when sending event tasks while running an event program.

Note:
The time entry is restricted to a value range between 0.1 seconds and 2 hours.

The parameters for event tasks B to K are identical with those for event task A. They will only be displayed if they have been enabled via the "Number of event tasks" parameter.

Event programs 1...8

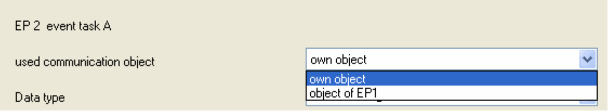
The parameters for event programs 1 and 8, for the relevant event tasks, are identical.
To reduce the setting overheads, there is an option, to be determined for each event task, whether proprietary sending objects or the relevant sending objects for the first event program in the allocated event range are to be used.
For this purpose, 8 event programs are divided into 2 event ranges. Each event range can be allocated a maximum of 4 event programs.
You can allocate event programs 2, 3 and 4 to event program 1. Event program 1 thus forms the first event range.
You can allocate event programs 6, 7 and 8 to event program 5. Event program 5 thus forms the second event range.

© Siemens AG 2008
Subject to change without further notice

25 CO scenes, event 750002

Event tasks from event programs can only be allocated to objects from the first or fifth event program for which event tasks have also been set in the first or second event range, e.g. if event tasks A to E were created for EP 1, then only objects for event tasks A to E can also be allocated from event programs 2 to 4. Event task A from EP 2 uses the event task A object from EP 1.
Event task B from EP 2 uses the event task B object from EP 1, etc.

Accordingly, the event tasks for event programs 2, 3 and 4 have an additional parameter:



EP 2 event task A

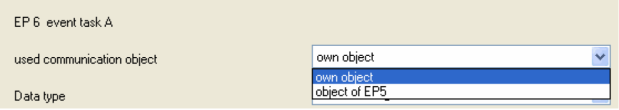
used communication object: own object

own object
object of EP1

Data type

Parameter	Settings
used communication object	**own object** object from EP1
This parameter determines whether a proprietary object "EP 2: Event task A" is to be generated for event task A, or whether the object "EP 1: Event task A" is to be used. You can send several telegrams sequentially in using a single communication object in this way.	

Accordingly, the event tasks for event programs 6, 7 and 8 have an additional parameter:



EP 6 event task A

used communication object: own object

own object
object of EP5

Data type

Parameter	Settings
used communication object	**own object** Object from EP5
This parameter determines whether a proprietary object "EP 6: Event task A" is to be generated for event task A, or whether the object "EP 5: Event task A" is to be used. You can send several telegrams sequentially in using a single communication object in this way.	

General notes

- Any faulty device is to be returned to the local Siemens office.
- If you have further questions concerning the product, please contact our Technical Support:

☎ +49 (0) 180 50 50-222
🖷 +49 (0) 180 50 50-223
✉ www.siemens.com/automation/support-request

Space for notices: