

The application unit is a DIN rail mounted device for insertion in the distribution board. The connection to the EIB is established via a bus connecting terminal at the front of the device.

The device can be downloaded with application programs with the aid of ETS2 from version 1.1 onwards and can execute functions which are required in building system technology. For example:

- Using the application program **Logic Time 200 IO/1**, gates and timing elements can be positioned in a very simple way using an integrated graphical interface in ETS2 and then linked in order to implement control systems.
- The application program **Logging/1**

enables textual information to be printed out via the ABB i-bus® EIB on one or more logging printers. The printers are connected via RS 232 interfaces which are downloaded with the corresponding application program. A typical application is the printing out of faults which have occurred in the relevant services.

- The application program **Times Groups/1** contains a year time switch program which allows day routines and week routines to be defined individually. The end user can modify the time switch program that has been created by the installer under Windows™ using the parameterisation software PZM 1.

#### Technical Data

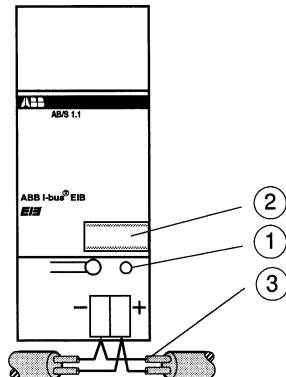
9

<b>Power supply</b>	– EIB	24 V DC, via the bus line
<b>Operating and display elements</b>	– LED (red) and push button	for assigning the physical address
<b>Type of protection</b>	– IP 20, EN 60 529	
<b>Protection class</b>	– II	
<b>Ambient temperature range</b>	– Operation	- 5 °C ... 45 °C
<b>Connection</b>	– EIB	Bus connecting terminal included with supply
<b>Mounting</b>	– on 35 mm mounting rail, DIN EN 50 022	
<b>Dimensions</b>	– 90 x 36 x 64 mm (H x W x D)	
<b>Weight</b>	– 0.1 kg	
<b>Certification</b>	– EIB-certified	
<b>CE norm</b>	– in accordance with the EMC guideline and the low voltage guideline	

9

Application programs	Number of communication objects	Max. number of group addresses	Max. number of associations
Logging /1	72	250	250
Logic Time 200 IO /1	200	250	250
Times Groups /1	250	250	250

## Circuit diagram

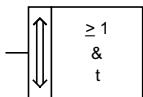


SK 0083 Z 98

1 Programming LED, push button  
 2 Nameplate  
 3 Bus connection

9

9

**Logging /1****Selection in ETS2**

- ABB
  - └ Controller
  - └ Controller

**Logging**

The application program enables the output of textual information on logging printers or computers that are connected to the EIB via serial interfaces.

Due to the high number of communication objects and parameters, the program memory is significantly larger than that of many other devices. The commissioning process therefore takes several minutes.

With this application program, the application unit can address up to ten serial interfaces, which are designated as "Channel 01" to "Channel 10". The interfaces then use the application program "Logging /2".

Two communication objects are used each time for linking the application unit to the serial interfaces. The application unit sends text to the serial interface via the 14 byte object "Channel ... Send text". Once the interface has received the text, it sends an acknowledgement to the application unit via the 1 bit object "Receive acknowledge".

So that the application unit can clearly detect whether the serial interface has received the logging text correctly, each channel may only address one serial interface.

**Date and time**

A separate real-time clock is integrated into the application unit. This real-time clock has the two 3 byte communication objects "Time" and "Date". These communication objects can either be used to synchronise the internal real-time clock e.g. using a DCF-77 receiver or to send the current time from the internal real-time clock to other devices.

If the internal real-time clock cannot be synchronised externally, the application unit can carry out an automatic con-

version between normal time and daylight saving time. The parameter "Automatic change to daylight saving time" is available for this purpose. If this parameter is set to "yes", it is possible to define when the time conversion should be implemented with the help of the subsequent parameters.

If the parameter "Send date and time" is set to "yes", the application unit is able to synchronise other EIB devices. The communication object "Date" then sends a telegram each time the day changes. It can be selected for the communication object "Time" at what intervals the telegrams should be sent.

In the default setting, the application unit reports the current date and time in every telegram. It is possible to specify in the parameter "Logging with date and time" that the date and time are not sent.

The Commissioning module of the ETS2 program can be used to set the integrated real-time clock if no other EIB device is able to send appropriate synchronisation telegrams.

The EIB Interworking Standard specifies 3 byte communication objects for the date and time with the following formats:

**Date:**

Byte 1	Day (1 ... 31)
Byte 2	Month (1 ... 12)
Byte 3	Year (0 ... 255)

**Time:**

Byte 1	Weekday (0 ... 7)* 32
	+ hour (0 ... 23)
Byte 2	Minute (0 ... 59)
Byte 3	Second (0 ... 59)

As only 1 byte has been reserved for the date in the third byte, years are usually coded in the following way:

Value	Year
90...99	=> 1990...1999
00...89	=> 2000...2089

The time may be displayed both with and without a weekday. The following codes are used: 0 = no weekday, 1 = Monday, 2 = Tuesday, ... , 6 = Saturday and 7 = Sunday.

The correct calculation of the first byte is somewhat time-consuming. Since the application program of the application unit does not evaluate the day of the week, the first byte can simply be carried out without indicating the day.

In the window "Groups: Write value", the values must be entered in hexadecimal format. The most frequently used values are listed in the following table.

By way of example, the information 29th December 1999, 7:35:00 should be transferred. The conversion of the individual elements in the hexadecimal notation produces the following result:

29 Ø 1D  
December Ø 0C  
1999 - 1900 = 99 Ø 63

7 Ø 07  
35 Ø 23  
00 Ø 00

To identify the hexadecimal notation, a dollar sign is placed in front of the value.

In the "Value" field, "\$1D0C63" must be entered for the date and "\$072300" for the time, each time without the quotation marks.

#### Events

The intervals at which the textual information is sent are specified by telegrams called "Events". The application unit makes up to 50 communication objects available for these events. If one of these communication objects receives a telegram, the application unit then sends the appropriate logging information.

The parameter "Send text" defines for each of the 50 events whether the application unit issues a message and which of the ten channels is used to send the message. An event can only be assigned to one of the channels.

If text should be issued to several logging printers, the communication objects of two or more events must be linked by the same group address but by different channels.

#### Logging text

Depending on the parameter setting, the size of the objects "Event ..." can be 1 bit, 2 bit, 4 bit, 1 byte, 2 byte or 4 byte. Depending on the object types, the logging information can consist of the following fields: date, time, value, event text and user text.

- "Event text" is assigned to a specific value of an event object. For example, there is different event text for a 1 bit object for the object values "0" and "1".
- "User text" is sent each time an event occurs, regardless of the object value.

Not all the fields are used for each object type. If the field for the event text is completely empty (no space), no logging text is issued for this event. Likewise, if only one field is available for text input and it is completely empty, no logging text is sent.

The transmitted logging text for the various types of event objects appears e.g. as follows:

**Logging text**

Examples of the various event object types

**1 Bit: Switching**

Date	Time	Event text
8 char.	8 char.	max. 60 characters
28.03.98	14:05:03	Lamps in room 524 switched on

**2 Bit: Priority**

Date	Time	Event text	User text
8 char.	8 char.	max. 20 char.	max. 40 char.
28.03.98	14:05:03	Forced ON	Fan in East Wing

**4 Bit: Dimming**

Date	Time	Value	Event text	User text
8 char.	8 char.	5 char.	max. 15 char.	max. 40 char.
28.03.98	14:05:03	1.25%	Dim brighter	Lecture room, North

9

9

**8 Bit: Value**

Date	Time	Value	User text
8 char.	8 char.	11 char.	max. 50 characters
28.03.98	14:05:03	10.54	A in circuit GF 23

**16 Bit: Value**

Date	Time	Value	Event text
8 char.	8 char.	11 char.	max. 50 characters
28.03.98	14:05:03	37.58	°C temperature in warehouse too high

**16 Bit: Counter**

Date	Time	Value	Event text
8 char.	8 char.	11 char.	max. 50 characters
28.03.98	14:05:03	23456	Visitors counted

**32 Bit: Counter**

Date	Time	Value	Event text
8 char.	8 char.	11 char.	max. 50 characters
28.03.98	14:05:03	47483647	Switching cycles? Replace relay 554

**32 Bit: Floating point value**

Date	Time	Value	User text
8 char.	8 char.	11 char.	max. 50 characters
28.03.98	14:05:03	11.111	m/s wind: Retract blinds

**Conversion**

Decimal and hexadecimal numbers

dec	hex	dec	hex	dec	hex	dec	hex
0	0	30	1E	60	3C	90	5A
1	1	31	1F	61	3D	91	5B
2	2	32	20	62	3E	92	5C
3	3	33	21	63	3F	93	5D
4	4	34	22	64	40	94	5E
5	5	35	23	65	41	95	5F
6	6	36	24	66	42	96	60
7	7	37	25	67	43	97	61
8	8	38	26	68	44	98	62
9	9	39	27	69	45	99	63
10	0A	40	28	70	46	100	64
11	0B	41	29	71	47	101	65
12	0C	42	2A	72	48	102	66
13	0D	43	2B	73	49	103	67
14	0E	44	2C	74	4A	104	68
15	0F	45	2D	75	4B	105	69
16	10	46	2E	76	4C	106	6A
17	11	47	2F	77	4D	107	6B
18	12	48	30	78	4E	108	6C
19	13	49	31	79	4F	109	6D
20	14	50	32	80	50	110	6E
21	15	51	33	81	51	111	6F
22	16	52	34	82	52	112	70
23	17	53	35	83	53	113	71
24	18	54	36	84	54	114	72
25	19	55	37	85	55	115	73
26	1A	56	38	86	56	116	74
27	1B	57	39	87	57	117	75
28	1C	58	3A	88	58	118	76
29	1D	59	3B	89	59	119	77

**Parameters**

The default setting for the values is **printed in bold type**.

## Calendar:

- Automatic change to daylight saving time

**no**  
yes

Only if "yes" is selected:

- Daylight saving time starts on (last Sunday)

no beginning  
January  
February  
**March**  
...  
June

- Daylight saving time ends on (last Sunday)

no end  
July  
...  
**October**  
November  
December

- Daylight saving time: change time by

**1.0 h**  
2.0 h  
3.0 h

- Send date and time

**no**  
yes

Only if "yes" is selected:

- Time is sent

**when the minute changes**

when the hour changes  
when the day changes

NOTE

- Date is sent only if day changes

- Logging with date and time

**no**  
yes

**9**

**9**

## For each event:

- Received value

**no function**  
1 Bit: Switching  
2 Bit: Priority  
4 Bit: Dimming  
8 Bit: Value  
16 Bit: Value  
16 Bit: Counter  
32 Bit: Counter  
32 Bit: Floating point value

- Send text

**no**  
via channel 01  
via channel 02  
...  
via channel 10

**For 1 Bit: Switching**

- Text for OFF  
(max. 60 characters) **Please enter text here**
- Text for ON  
(max. 60 characters) **Please enter text here**

**For 2 Bit: Priority**

- Fixed allocated text  
(max. 40 characters) **Please enter text here**
- Text for RELEASE OFF  
(max. 20 characters) **Text input**
- Text for RELEASE ON  
(max. 20 characters) **Text input**
- Text for FORCED OFF  
(max. 20 characters) **Text input**
- Text for FORCED ON  
(max. 20 characters) **Text input**

**For 4 Bit: Dimming**

- Fixed allocated text  
(max. 40 characters) **Please enter text here**
- Text for STOP DIMMING  
(max. 15 characters) **Text input**
- Text for DIM DARKER  
(max. 15 characters) **Text input**
- Text for DIM BRIGHTER  
(max. 15 characters) **Text input**

**9****9****For 8 Bit: Value**

- Object value 0 is displayed as  
(Range: -1000 ... 1000) **0**
- Object value 255 is displayed as  
(Range: -1000 ... 1000) **255**
- Number of displayed digits  
behind point **0**
- Fixed allocated text  
(max. 50 characters) **Please enter text here**

**For 16 Bit: Value**

- Text is sent **if value out of limits  
if value within limits**
- If value is within limits:
  - Lower limit = input / 100  
(67108864 ... 67076096) **-67108864**
  - Upper limit = input / 100  
(-67108864 ... 67076096) **67076096**
  - Fixed allocated text  
(max. 50 characters) **Please enter text here**
- If value is outside limits:
  - Lower limit = input / 100  
(-67108864 ... 67076096) **-67108864**
  - Text for value < lower limit  
(max. 50 characters) **Please enter text here**
  - Upper limit = input / 100  
(-67108864 ... 67076096) **67076096**
  - Text for value >= upper limit  
(max. 50 characters) **Please enter text here**

**For 16 Bit: Counter**

– Signed value	yes / no
– Text is sent	if value within limits if value out of limits
If value within limits:	
– Lower limit (-32768 ... 65535)	-32768
– Upper limit (-32768 ... 65535)	65535
– Fixed allocated text (max. 50 characters)	Please enter text here
If value outside limits:	
– Lower limit (-32768 ... 65535)	-32768
– Text for value < lower limit (max. 50 characters)	Please enter text here
– Upper limit (-32768 ... 65535)	65535
– Text for value >= upper limit (max. 50 characters)	Please enter text here

**For 32 Bit: Counter**

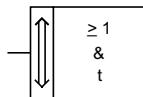
– Signed value	yes / no
– Text is sent	if value within limits if value out of limits
If value within limits:	
– Lower limit (-2147483647 ... 2147483647)	-2147483647
– Upper limit (-2147483647 ... 2147483647)	2147483647
– Fixed allocated text (max. 50 characters)	Please enter text here
If value out of limits:	
– Lower limit (-2147483647 ... 2147483647)	-2147483647
– Text for value < lower limit (max. 50 characters)	Please enter text here
– Upper limit (-2147483647 ... 2147483647)	2147483647
– Text for value >= upper limit (max. 50 characters)	Please enter text here

9

9

**For 32 Bit: Floating point value**

– Fixed allocated text (max. 50 characters)	Please enter text here

**Logic Time 200 IO /1**

The application program enables central functions to be integrated within an EIB installation which would be difficult to implement with individual logic units.

A comprehensive manual is available for this application.

**Selection in ETS2**

– ABB  
  └ Controller  
    └ Controller

**Graphical project design**

The logic functions are entered in a graphical function chart which is based on DIN 40900. The following are available

- input objects,
- output objects,
- input/output objects,
- logical elements,
- gates,
- timing elements,
- staircase lighting functions and
- worksheets.

The graphical display is retrieved via the “Edit Device” window with the “Parameter” button. All the data is stored together with the other project data in the ETS2 database.

A link is established by dragging the mouse, while holding down the left mouse button. The link is displayed as a line. Junctions are not drawn. If a line must cross another in order to represent a link, a marker is indicated. The name of the marker is automatically assigned and cannot be modified.

To simplify the connection process in larger projects e.g. if elements need to be linked that are not visible at the same time, the command “Link to Interface” can be used. The available inputs and outputs are displayed in the window and the required links can be selected.

Links can be deleted by selecting and pressing the Del button or by using the menu command.

Plausibility tests are carried out during the parameterisation. If an incorrect entry is made, possible sources of error are indicated.

The following tests are conducted:

1. The maximum number of group addresses is 250.

2. The maximum number of associations is 250.

3. Only one group address may be assigned to an output.

4. If an input is linked to more than one group address, the initialisation on bus voltage recovery may not take place via the bus after a bus failure with data loss.

5. The same group addresses may not be assigned to the input and output objects.

6. Various function-related plausibility tests e.g. permitted links etc.

7. Up to 200 communication objects can be created. They can be used as input or output objects. A dual assignment is not permitted i.e. a communication object is either an input or an output object. The objects that are used for staircase lighting functions are an exception.

**Worksheets**

The graphical project design can rapidly become confusing on a computer with a small screen or a low resolution.

9

9

## Program sequence

In the event of a bus voltage failure, the values of the input objects are stored for at least one hour. If the bus voltage failure lasts longer than an hour, the values of the input objects may be lost but yet still be retained. When the application program is restarted, the device checks whether a data loss has occurred before it processes the values of the input objects. Depending on the parameter settings, the input communication objects then use

- the values stored before the bus voltage failure,
- the parameterised values or
- the values read out via the bus.

Once this process has been completed, the device executes the programmed logic functions and determines the values of the output communication objects. Depending on the extent of the internal connections, this can take approx. 2 seconds. The telegrams of the output communication objects are then sent. Since approximately 50 telegrams per second can be transmitted in an EIB line, the time required depends on the number of outputs used and their programming.

## Inputs and outputs

200 input or output symbols can be used in the graphical project design tool. The positioned symbols correspond to the communication objects. They can have various object types. The object type can be freely selected, provided that the objects in the graphical display have not yet been linked with other symbols.

Each object automatically receives a consecutive number by which it can be identified in the other windows of the ETS2 program.

The behaviour on bus voltage recovery can be set, whereby it is possible to distinguish between bus voltage failures with and without data loss.

If an input should read its input value after bus voltage recovery, it should be ensured that only a communication object of another device responds and that no other devices are switched unintentionally by the receipt of the reply telegram.

The outputs normally only send a telegram if their value has been modified. If required, they can also send a telegram each time that an associated input receives a telegram.

Only one group address can be assigned to each output communication object, whereby this group address cannot be allocated to an input object at the same time.

## Logical elements

50 logical elements with AND, OR or XOR connections are available as logic functions.

Inputs and outputs can only be linked to 1 bit object types.

Once they have been inserted in the project, the logical elements have 2 inputs each. Further inputs can be created in the parameters for the logical element. Each element can have up to 8 inputs.

Unused inputs can also be deleted again in the parameter window. If an input is linked, the graphical association must first be deleted before the input of the logical element can be deleted.

Both the inputs and the output of a logical element can be negated.

## Gates

50 gates can be used in the graphical interface. Gates have one input, one output and a control input.

The control input is always 1 bit and can either be activated with a "0" or a "1".

The input and the output of a gate can be linked with all the object types, whereby the input and output type must be identical. If the object type is 1 bit, the output can be negated.

A filter function can be entered for the input of a gate, if the object type is 1 bit. In the setting "Pass through object value 1", the gate only routes telegrams with the value "1".

The following rules apply to the operation of a gate:

- If the control input is not linked, the gate is open. All changes at the input are routed directly to the output.
- If the control input is enabled, the gate is open. All changes at the input are routed directly to the output.
- If the control input is not enabled, the gate is closed. Changes at the input are stored but not routed to the output.
- If the control input is enabled and the input value has been stored during the disable operation, the gate is open. The stored value is routed to the output.
- If the control input is enabled and the input value has not been stored, the gate is open. No values are routed to the output.

## Timers

30 timers or staircase lighting functions can be used in the graphical display. Timers have a 1 bit input and a 1 bit output.

The output can be inverted. A filter function can be entered for the input. In the setting “Pass through object value 1”, the timer only evaluates telegrams with the value “1”.

Each timer has an ON and an OFF delay which can be set separately up to a maximum of 18 hours, 12 minutes and 15 seconds. The times can be triggered. If a new telegram is received during an active delay, the period restarts.

In the event of a bus voltage failure, the timers immediately switch to their output state i.e. the outputs switch to zero signal regardless of the signals that are present at the input and the time that has already elapsed.

## Staircase lighting functions

Staircase lighting functions can only be linked with a 1 bit input object that is used simultaneously as a combined input/output object. If a telegram with the value “1” is received, the period starts that was specified in the “Auto-reset” parameter.

In the event of a bus voltage failure, the staircase lighting function immediately switches to the output state i.e. the outputs switch to zero signal, regardless of the signals that are present at the input and the time that has already elapsed.

**Parameters**

The default setting for the values is **printed in bold type**.

The parameters of the device are set in the graphical project design program.

Each symbol has its own settings. The display of the parameters is dependent on the links entered.

**For input objects**

– Object Type	<b>1 Bit / 4 Bit / 1 Byte / 2 Byte</b>
– Initial Type (Hot boot)	<b>Use initial value</b>
	Ask bus for value
	Use value before bus voltage dropped
– Initial Value	<b>0</b>
– Initial Type (Cold boot)	<b>Use initial value</b>
	Ask bus for value
– Initial Value	<b>0</b>

**For output objects**

– Type	<b>1 Bit / 4 Bit / 1 Byte / 2 Byte</b>
– Send	On each calculation
	<b>Only on alteration</b>

9

**For input/output objects**

– On bus voltage recovery the ON switched output is	<b>not reset</b>
	reset after time period

9

**For logical elements**

– Name	<b>LE</b>
– Number	<b>1</b>
– Type	XOR
	OR
	<b>AND</b>
For each input:	
–	Negated / ---
–	Used / <b>Not Used</b>
– Negate output	Yes / <b>No</b>

**For gates**

– Name	<b>Gate</b>
– Number	<b>1</b>
– Negate Control Input	<b>No</b>
	Yes
– Negate Output	Yes / <b>No</b>
– Filter	Pass through object value 0
	Pass through object value 1
	<b>Nothing</b>
– Save input value	No
	<b>Yes</b>

**For timers**

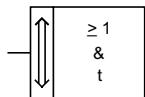
– Name	<b>Timer</b>
– Number	1
– Negate Output	<b>Yes / No</b>
– Filter	Pass through object value 0 Pass through object value 1
– Delay switch on (h:min:s)	<b>Nothing</b> <b>00:00:00</b>
– Delay switch off (h:min:s)	<b>00:00:00</b>

**For staircase lighting functions**

– Name	<b>Stairlight</b>
– Number	1
– Autoreset (h:min:s)	<b>00:00:00</b>

**For worksheets**

– Name	<b>Worksheet</b>
– Number	1

**Times Groups /1****Selection in ETS2**

- ABB
  - └ Controller
  - └ Controller

An extensive product manual is available for the application program "Times Groups /1".

The application program "Times Groups /1" for the application unit enables the installer of the EIB system to implement complex time switch programs and control systems for building automation via the EIB e.g. in schools, factories and office buildings as well as residential buildings.

With the help of the user-friendly, graphical interface, you can define and modify the parameters of the application program "Times Groups/1" both simply and clearly.

The parameterisation data is stored together with the project in the ETS2 database. The configured data can be printed out as a list for documentation purposes.

You can store the time switch program that has been created with the application program "Times Groups/1" as a file and transfer it to the system user on a data carrier.

The programming software PZM 1 is available for the user of the installation and enables him to adapt the time switch program created by the installer to his own requirements. The parameterisation software PZM 1 runs under Microsoft Windows from Version 95/98/ 2000/ ME/ NT/ XP.

The user of the installation has the following options with the parameterisation software PZM 1,

- read in the file that has been transferred by the installer of the installation,
- modify the time switch program according to his requirements and download it to the application unit via the EIB,
- store the modified time switch program as a file,
- print out the time switch program in sections or in full,
- read out and set the date and time of the application unit and read out specific information from the device.

The application program "Times Groups /1" is available on our product data CD-ROM and on the home page.

The reading in of the application program "Times Groups /1" is carried out in the usual way with ETS2 from version 1.1 onwards.

**Times**

With the help of the user-friendly, graphical interface, the parameters of the application program "Times Groups /1" can be defined and modified both clearly and simply. The graphical project design tool is started by calling up the parameters in the device view in ETS2. The ETS2 program is switched to the background. When exiting the graphical project design window, the parameterisation data is saved and the ETS2 "Edit Device" window is displayed again. The time switch program defines when bus telegrams with specific group addresses and specific values are sent on the bus. The parameterisation data is stored in the ETS2 database together with the project. The configured data can be printed out as a list for documentation purposes.

**Time switch program**

15 day routines i.e. switching programs for specific days can be defined in the time switch program. A total of 800 switching times can be assigned to these day routines. Individual days or periods which deviate from the standard week routine are called special days. 100 special days can be defined in total. The periods for the summertime conversion can be defined for up to 5 years in advance.

**Groups**

If several bus telegrams with specific group addresses and values should be sent repeatedly at certain times, they are combined into a group as so-called group members e. g. at 15:00, 16:00, 17:00 and 18:30, the luminaires in Room 1, Room 2 and Room 3 should always be switched on and the blinds in Room 1, Room 2 and Room 3 should always be lowered. The group addresses and values that have been assigned to these functions can now be combined into a group e.g. with the name "Room 1, 2, 3". A group can thus also be called a functional group and the combined group addresses can be called group members.

The group address that acts as a trigger for this group e.g. with the name "Room 1, 2, 3" now needs to be entered in the time switch program in order to send bus telegrams with the group addresses (group members) and values that have been defined for this group. A group address that acts as a trigger is called a group trigger. This saves time during the assignment of parameters and the time switch program is clarified.

A total of 20 groups can be defined. These groups can contain a maximum of 300 group members. The group address that triggers a group is called the group trigger. The group trigger occupies one object. If several group triggers are assigned, each group trigger only occupies one association.

**Parameters**

The parameters of the device are set in the graphical project design program.

**Functions:**

Addresses and objects	250 group addresses 250 objects 250 associations
Time switch program	15 day routines 800 switching times 1 week routine 100 special days Summertime conversion for 5 years in advance
Groups	20 groups 300 group members At least 1 group trigger

**Definition of terms**

**Time switch program:**  
Defines when bus telegrams are sent with specific group addresses and specific values on the bus.

**Day routine:**  
Contains the time switch program of a day from 00:00 to 23:59. Day routines can be activated and deactivated via the time switch program and by bus telegrams.

**Week routine:**  
Defines the assignment of day routines to the days of the week.

**Special days:**  
Individual days or periods which deviate from the standard week routine.

**Group:**

Functional group which is a combination of group addresses that are sent at the same time.

**Group members:**  
Group addresses that belong to a group.

**Group trigger:**  
Group addresses that trigger a group. The type and value of the group addresses determines the triggering of the group e.g. "0" switches ON, "1" switches OFF or value < 230 switches ON, value > 250 switches OFF.