

xx7 LIBRARY for Archiving Historical Data in DBs

0	Content		
0.1	Document History	0-2	
0.2	Brands and trademarks	0-2	
1	Description		
1.1	Features.....	1-1	
1.1.1	Abbreviations	1-1	
1.2	Program modules (FB).....	1-2	
1.2.1	HDLOG_INIT	1-2	
1.2.2	HDLOG_ANY.....	1-4	
1.3	COMMAND interface	1-5	
1.4	Data blocks (DB).....	1-6	
1.4.1	CONFIG DB	1-6	
1.4.2	HDLOG-DB	1-8	
1.4.3	Trend value memory (HD DBs).....	1-9	
1.5	Using Saia PG5® Web Editor macros	1-11	
A	Appendix		
A.1	Icons	A-1	
A.2	Contact	A-2	



0.1 Document History

0

Date	Version	Changes	Remarks
EN01	2011-02-24	-	First edition
EN05	2014-04-02	-	updated

0.2 Brands and trademarks

Saia PCD® and Saia PG5®
are registered trademarks of Saia-Burgess Controls AG.

Technical modifications are based on the current state-of-the-art technology.

Saia-Burgess Controls AG, 2009 © All rights reserved.

Published in Switzerland

1 Description

The library for historic data (HD) enables the user to store data cyclically or event-driven in a data block (DB) of SaiaPCD® xx7. The stored data can be read from the controller using a web browser or Saia PCD® web panels via the SaiaPCD® web server and displayed in predefined trend graphs. Data shown in trend graphs in a web browser can be saved on a PC in a csv file.

A Step7 program is required for configuration and recording of trend graphs. The Step7 program contains program and data blocks for configuration and recording of trend graphs.

The visualisation parameters for trend graphs can be set in the Saia PG5® web editor. Trend graphs are assigned in xx7 and in the Saia PG5® web editor by allocation of a unique trend number.

Only the HDLOG DB number has to be configured in the trend macro in the Saia PG5® web editor.

1.1 Features

- Recording a trend graph for the BOOL data type
- Recording a trend graph for any data type (max DWORD)
- Recording value and timestamp
- Each trend recording requires 8 bytes in the trend tracking DB
- The trend tracking addresses are calculated automatically and are stored in the Config DB
- The DBs required for storing trend tracking are generated by the S7 program
- The program can be re-initialised at runtime also with modified parameters (for example more memory for trend tracking 2). In this case all trend tracking DBs are deleted, the trend tracking addresses are re-calculated and finally the necessary trend tracking DBs are generated.

1.1.1 Abbreviations

DB	Step7 Data Block
HD	Historical Data
HDA	Historical Data Recording
GLT	Central Building Control Systems
SBC Pointer	Saia Burgess Controls Pointer

1.2 Program modules (FB)

1.2.1 HDLOG_INIT

If called with INIT = TRUE all trend tracking DBs are deleted, the trend tracking addresses are re-calculated and finally the necessary trend tracking DBs are generated. In CONFIG_DB the configuration parameters for the HDA are entered by the user. These values are used to create and populate HDLOG_DB. Re-initialisation is only possible if the HDA is in a state Stop or Clear.

CALL FBx, Dby

Input parameters:

INIT:	Initialisation
HDLOG DB:	DB with the configuration of the whole HDA
CONFIG_DB:	DB with HDA initialisation values

In-out parameters:

COMMAND:	HDA controls (Run, Stop, Clear)
----------	---------------------------------

Output parameters:

BUF_FULL:	Message buffer full for any trend tracking. The first trend tracking which reaches the full value will trigger a corresponding message
STATE:	Acknowledgement via HDA controllers
RETVAL:	Return value
BUSY:	The order is being processed



We recommend calling HDLOG_INIT in OB100 with INIT = TRUE with synchronous generation of the trend tracking DBs and calculation of the addresses in the HDLOG DB. On re-initialisation in OB1 the order is processed asynchronously and distributed over several OB1 cycles (reduction of cycle time load).

Possible return values for FB's HDLOG_INIT:

RETVAL:

- 0 = HDLOG DB created
- 1 = Message stating that the HDA has not yet been created
- 2 = COMMAND is not plausible
- 3 = HDA has been created
- 4 = Processing mode not defined
- 5 = The HDA must be stopped before the Clear command
- 6 = Size of trend DB is too large
- 7 = Error when creating trend DBs
- 8 = Error when creating trend DBs
- 9 = HDLOG_DB has not been created
- A = HD DB has not been created
- B = Internally no stage is run
- C = Size of HDA is smaller than the sum of trend tracking
- D = Internal error
- E = HDA cannot be initialised in RUN state

1

1.2.2 HDLOG_ANY

Functional module for recording any data type.

CALL FBx, Dby

Input parameters:

ENABLE:	Record / do not record value
HDLOG_DB:	DB with the configuration of the whole HDA
INSTANCE_DB:	Instance DB
TREND_NUMMER:	Allocation of trend graph
TRIGGER:	Force log entry
TREND:	Recorded value (data type Any)
CLEAR:	Deletes the binary value log files (level-triggered)

Output parameters:

BUF_FULL:	Message buffer full for current trend tracking. 0 = OK 1 = HDA full to ??% (defined via parameter Trend x.Buff_full) 2 = HDA full 100%
LAST_VALUE:	Last recorded value entered in the HDA (DWORD)
LAST_TIME:	Last recorded timestamp
RETVAL:	Return value

The following trend types can be recorded:

- BOOLEAN (Marker, DB, I/O)
- BYTE (Marker, DB, I/O)
- WORD (Marker, DB, I/O)
- DWORD (Marker, DB, I/O)
- Counter
- Timer

Possible return values for FB's HDLOG_ANY:

RETVAL:

0 = Data have been saved

1 = Format and data type do not match

8 = Parameter TREND_NUMMER > number of trend trackings

9 = HDLOG_DB has not been created

A = HD DB has not been created

1.3 COMMAND interface

The HDA is controlled via the input parameter COMMAND and the output parameter of the FB's HDLOG_INIT:

1

COMMAND:

- 0 = OK (written from HDA)
- 1 = RUN (written from GLT)
- 2 = STOP (written from GLT)
- 3 = CLEAR (written from GLT)



A GLT command > 0 is acknowledged by the HDA with 0.

STATE:

- 1 = RUN (written from HDA)
- 2 = STOP (written from HDA)
- 3 = CLEAR (written from HDA)

Rules:

- After a COMMAND is processed, COMMAND = 0 and STATE are set by the HDA according to the state
- Trend tracking can only be deleted when recording is stopped
- After deletion the message BufferFull is reset

1.4 Data blocks (DB)

1.4.1 CONFIG DB

1

When the function HDLOG_INIT is called with INIT = TRUE this DB is evaluated and the HDLOG DB generated correspondingly.

STRUCT

Version:	DWORD := 1002;	//Version number (1002=1.0.02)
KB_Max:	DWORD := DW#16#3C;	//Size of HDA in kB //3Ch = 60 kB
Max DB size:	DWORD := DW#16#14	//Size of trend DB containing the //trend tracking memory. //The max. size is 63 kB. //The smaller this value the more //Trend DBs are generated //14h = 20kB
First_HDA :	DWORD := DW#16#370;	//First trend DB containing the //trend tracking memory //370h = 880
Trends:	DWORD := DW#16#6;	//Number of trend trackings
Trend1.Num:	DWORD := DW#16#1;	//Trend tracking 1
Trend1.Text:	STRING[80]:= 'Temperature';	//Designation of trend tracking
Trend1.kB:	DWORD := DW#16#A;	//Size of trend tracking in kB //Ah = 10 kB
Trend1.Format:	DWORD := DW#16#1;	//Data type of recorded //data value
Trend1MinDifference :	DWORD := DW#16#0;	//Minimum difference for //re-recording
Trend1MinDelay :	DWORD := DW#16#0;	//Minimum delay in seconds //until next recording
Trend1CyclicDelay :	DWORD := DW#16#1;	//Cyclical recording after //x seconds
Trend1BufferType:	DWORD := DW#16#0;	//Buffer type
Trend1BufferVoll:	DWORD := DW#16#50;	//Value in % for message //"Buffer full" //50h = 80%
Trend2.Num:	DWORD := DW#16#2;	//Trend tracking 2

...

END STRUCT



All inputs for format DWORD := DW#16#xx must be entered as HEX values.

The maximum size for the parameter "Max DB Size" is 63 kB.

Format: Declaration of recorded data value for independent evaluation; expressed as a numeric value this can be interpreted as follows:

- 0 = Unknown
- 1 = Boolean, 4 Byte
- 2 = Unsigned, 4 Byte
- 3 = Signed, 4 Byte
- 4 = Not used
- 5 = Float IEEE, 4 Byte
- 6 = 1/10, 4 Byte (1000 ~ 100.0)

1

Buffer Type:

This parameter is used to select either a ring buffer or a simple buffer. Independent of the setting when the full value xx% (parameter "BufferFull") for the trend tracking is reached a message is output in BUF_FULL in order to trigger the corresponding responses.

The values for Buffer Type are:

- 0 = Ring buffer
- 1 = Stop when 100% of trend tracking is reached

1.4.2 HDLOG-DB

When the function HDLOG_INIT is called with INIT = TRUE the required HDLOG DB is generated automatically by the controller and populated with the corresponding admin data.

The HDLOG_ANY functions evaluate this DB in order to record the corresponding trend trackings.

The SBC Web-HDLOG applet evaluates this DB in order to output and display the corresponding trend trackings.

STRUCT

```

Version :                DWORD := DW#16#1002;    //Version number (1002=1.0.02)
KB_Max :                DWORD := DW#16#3C;      //Size of HDA in kB (parameter of
                                                    //HDLOG_INIT is stored here)
                                                    //3Ch = 60 kB

First_HDA :             DWORD := DW#16#370;     //first data block of trend values
                                                    // 370h = 880

Last_HDA :              DWORD := DW#16#372;     //last data block of trend values
                                                    // 372h = 882

Last_Size :             DWORD := DW#16#1400;    //Number of elements of last DB
                                                    // 1400h = 5120

Trends :                DWORD := DW#16#6;      //Number of trend trackings
Trend1Num :             DWORD := DW#16#1       //Trend tracking 1
Trend1Text :            DWORD := DW#16#00180322; //Text address 1. Trend tracking as SBC pointer;
                                                    //DB802.DBW24
Trend1Par :             DWORD := DW#16#0078035C; //DB address 1. Trend tracking as SBC pointer;
                                                    //DB860.DBW120
Trend1KB :              DWORD := DW#16#A       //Size in kB of 1st trend tracking Ah = 10
Trend2Num :             DWORD := DW#16#2       //Trend tracking 2
Trend2Text :            DWORD := DW#16#008A0322; //Text address 2. Trend tracking as SBC pointer;
                                                    //DB802.DBW138
Trend2Par :             DWORD := DW#16#00BC035C; //DB address 2. Trend tracking as SBC pointer;
                                                    //DB860.DBW188
Trend2KB :              DWORD := DW#16#A;      //Size in kB of 2nd trend tracking Ah = 10
.....
Trend1EnType :          DWORD := DW#16#0;      //Placeholder
Trend1EnAdress :       DWORD := DW#16#0;      //Placeholder
Trend1TrgType :        DWORD := DW#16#0;      //Placeholder
Trend1TrgAdress :     DWORD := DW#16#0;      //Placeholder
Trend1Type :           DWORD := DW#16#0;      //Placeholder
Trend1Adress :         DWORD := DW#16#0;      //Placeholder
Trend1TFormat :        DWORD := DW#16#3;      //Format of value
Trend1MinDifference :  DWORD := DW#16#0;      //Minimum difference for
                                                    //re-recording
Trend1MinDelay :       DWORD := DW#16#0;      //Minimum delay in seconds until
                                                    //next recording
Trend1CyclicDelay :    DWORD := DW#16#1;      //Cyclical recording after
                                                    //x seconds
Trend1LastValue :      DWORD := DW#16#319;     //last recorded value 319h = 793
Trend1LastTime :       DWORD := DW#16#4D5A42C8; //last recorded timestamp
                                                    //4D5A42C8h = 1297760968
Trend1MemStart :       DWORD := DW#16#00000370; //Absolute start address as SBC pointer
                                                    //DB880.DBW0
Trend1MemEnd :         DWORD := DW#16#27F80370; //Absolute end address as SBC pointer
                                                    //DB880.DBW10232
Trend1MemPos :         DWORD := DW#16#09E00370; //Absolute current address as SBC pointer
                                                    //DB880.DBW2528
Trend1BufferType :     DWORD := DW#16#0;      //Buffer type
Trend1Internal :       DWORD := DW#16#1;      //0 = Trend stopped; 1 = Trend started

Trend2EnType :         DWORD := DW#16#0       //Placeholder
Trend2EnAdress :       DWORD := DW#16#0;      //Placeholder
.....
END STRUCT

```

Structure of SBC pointer

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DB address																DB number															



Further information on the timestamp in the following section 1.4.3.

1.4.3 Trend value memory (HD DBs)

When the function HDLOG_INIT is called with INIT = TRUE the required DBs are generated automatically by the controller. The DBs are numbered sequentially started with First_HDA. The numbers and addresses are stored in HDLOG DB.

All values, such as Input, Output, Marker, Timer, Counter and DBs are recorded with 2 elements of 4 bytes each (total of 8 bytes), the absolute time and date stamp and value.

STRUCT

```

Timestamp1:      TIME := T#0s;           //timestamp
Value1:          DWORD := DW#16#0;      //logged value
Timestamp2:      TIME := T#0s;           //timestamp
Value2:          DWORD := DW#16#0;      //logged value
Timestamp3:      TIME := T#0s;           //timestamp
Value3:          DWORD := DW#16#0;      //logged value
.....
    
```

END STRUCT

Example of recording

```

Timestamp1:      TIME := DW#16#4D5A546D; //timestamp 4D5A546Dh = 1297765485
//1297765485s = 15.02.2011,11:24:45 realtime
Value1:          DWORD := DW#16#320;    //logged value 320h = 794
Timestamp2:      TIME := DW#16#4D5A546E; //timestamp 4D5A546Eh = 1297765486
//1297765486s = 15.02.2011,11:24:46 realtime
Value2:          DWORD := DW#16#319;    //logged value 319h = 793
Timestamp3:      TIME := DW#16#4D5A546F; //timestamp 4D5A546Fh = 1297765487
//1297765487s = 15.02.2011,11:24:47 realtime
Value3:          DWORD := DW#16#318;    //logged value 318h = 792
    
```

Tips:

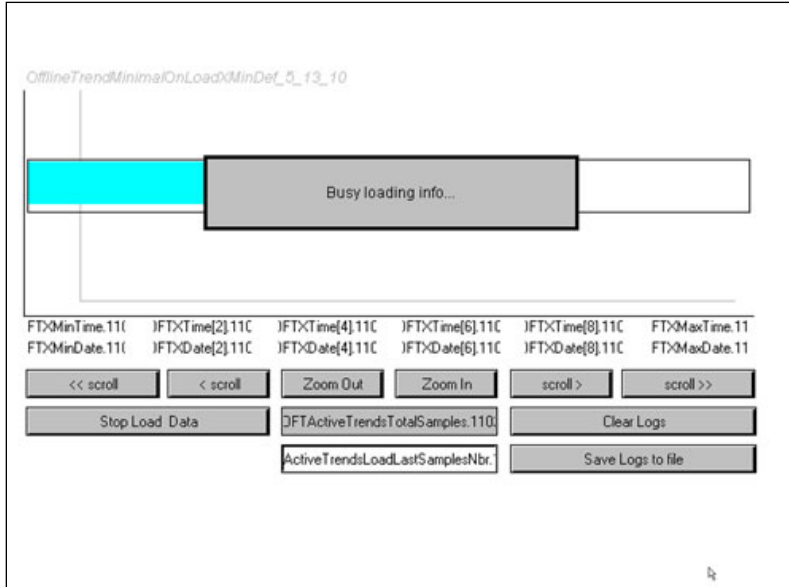
- Bit 31 of the timestamp is expressed as follows:
 - Bit 31 = 0 : The following value is state/value of the element to be recorded
 - Bit 31 = 1 : The following value provides status information
- The status information can be interpreted according to the following numeration:
 - 1 : ShutDown Controller was shut down at this time
 - 2 : Restart Controller was restarted at this time
 - 3 : TrendOn Trend graph was activated via ENABLE
 - 4 : TrendOff Trend graph was deactivated via ENABLE
- The timestamp contains the number of seconds since midnight (00:00:00) on 01.01.1970 (coordinated universal time)
- If a trend has been defined as Signed Format, when recording a byte or word value, the value is modified to a DWORD independent of the prefix

1.5 Using Saia PG5® Web Editor macros

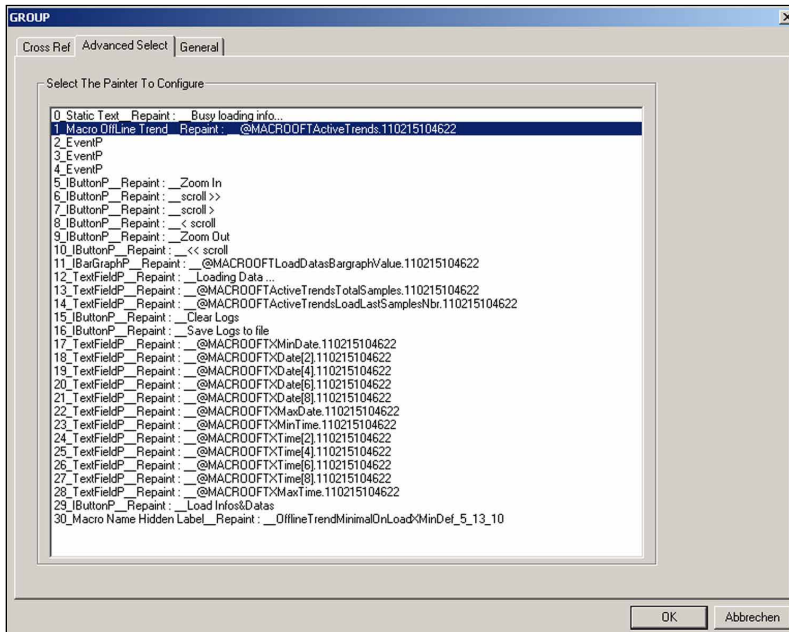
All macros with the name OfflineTrend*.esm contained in the S-Web Editor can be used for displaying trend graphs.

1

After placing the corresponding macro on the web page, the group attributes must be opened by double clicking on the macro.



Double click on the name “Macro Offline Trend...” in the group to open the trend definition page.



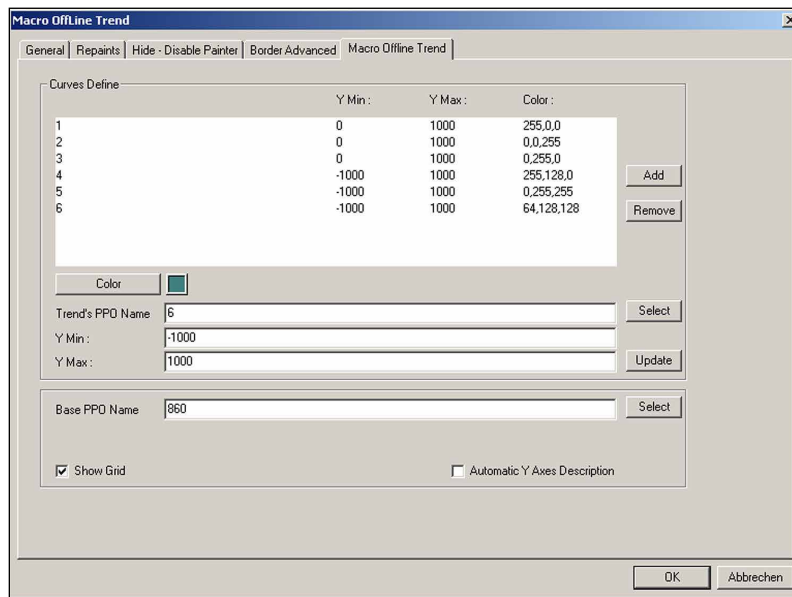
The individual trend graphs can be defined on this page.

The trend number used in the Config DB for the parameter "Trend1.Num" and in the HD-LOG_ANY for the parameter "Trend_Number" must be defined in the field "Trend PPO Name".

1

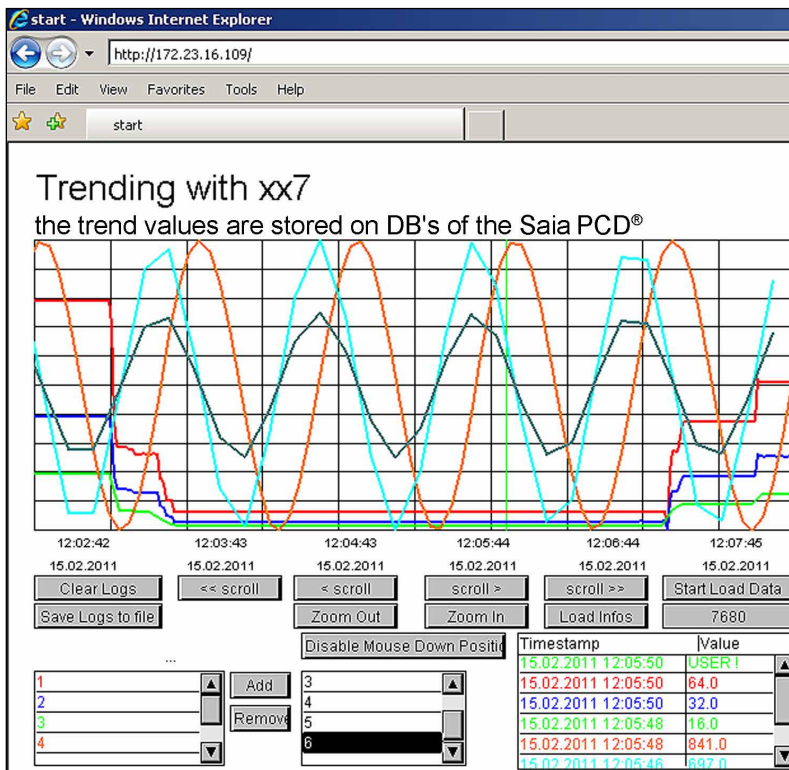
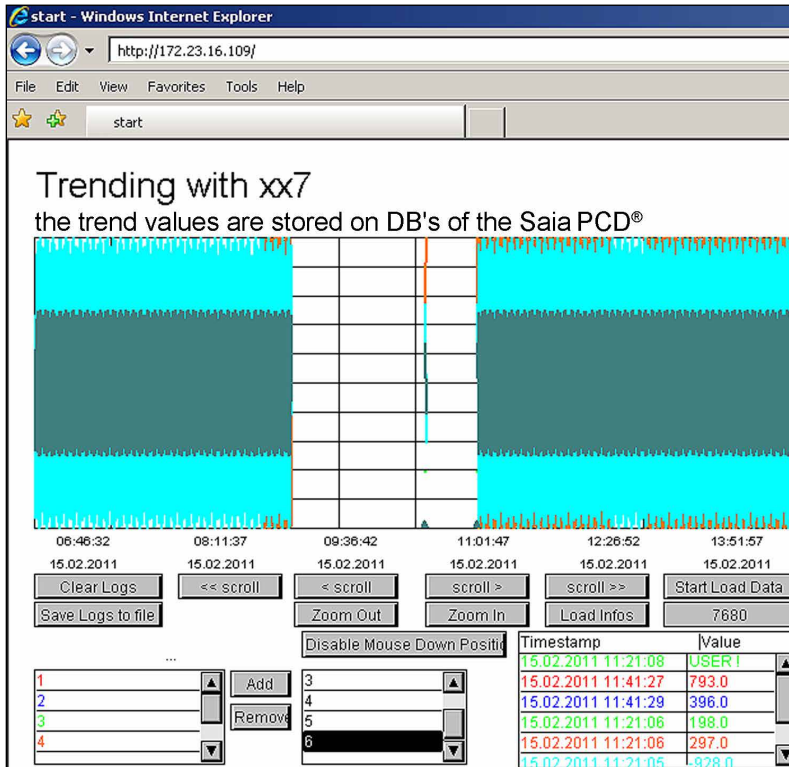
The values saved under this trend number in the Saia PCD® will be recorded in the trend with the Y values defined as "Color" and "Y min" and "Y max".

The DB number used in HDLOG_ANY for the parameter "HDLOG_DB" is entered in the field "Base PPO Name".








After loading the web project to the Saia PCD® the online view of the offline trends will be displayed in the web browser as follows:

1



A Appendix

A.1 Icons

	<p>In manuals, this symbol refers the reader to further information in this manual or other manuals or technical information documents. As a rule there is no direct link to such documents.</p>
	<p>This symbol warns the reader of the risk to components from electrostatic discharges caused by touch. Recommendation: Before coming into contact with electrical components, you should at least touch the Minus of the system (cabinet of PGU connector). It is better to use a grounding wrist strap with its cable permanently attached to the Minus of the system.</p>
	<p>This sign accompanies instructions that must always be followed.</p>
	<p>Explanations beside this sign are valid only for the Saia PCD® Classic series.</p>
	<p>Explanations beside this sign are valid only for the Saia PCD® xx7 series.</p>

A.2 Contact**Saia-Burgess Controls AG**

Bahnhofstrasse 18
3280 Murten
Switzerland

Phone +41 26 672 72 72

Fax..... +41 26 672 74 99

Email support: support@saia-pcd.com

Supportsite: www.sbc-support.com

SBC site: www.saia-pcd.com

International Representatives &

SBC Sales Companies: www.saia-pcd.com/contact

Postal address for returns from customers of the Swiss Sales office**Saia-Burgess Controls AG**

Service Après-Vente
Bahnhofstrasse 18
3280 Murten
Switzerland

A