Basic and System Components/Devices – Minimum Requirements – Standardised solutions - Tests KNX System Conformance Testing

BCUs and BIMs

BCUs

9

4

1

Summary

This document contains the description of the standardised Bus coupling units.

This document is part of the KNX Specifications v2.1.

Version 01.02.01 is a KNX Approved Standard.

Document Updates

Version	Date	Modifications
1.1	2003.03	Approved Standard
1.2	2009.06	Readying document for publication in V2.0 of the KNX standard – removal of BCU2
01.02.01	2013.10.23	Editorial updates for the publication of KNX Specifications 2.1.

Filename: 09_04_01 Basic and System Components - BCUs v01.02.01 AS.docx
Version: 01.02.01
Status: Approved Standard
Savedate: 2013.10.23
Number of pages: 60

Contents

1 Preface

In the light of the multi-vendor philosophy within KNX, KNX has opted for a standardization of a number of crucial basic and system components/devices providing standardized interfaces not only to manufacturers but also installers and users. However, it is still possible to design non-standardized solutions.

In the following clauses, the underneath connotation is used:

No.	Abbreviation	Meaning
1	M	Minimum requirements for certification – the 'M' requirements are only a subset of the standardized/optional requirements respectively recommendations – devices not complying to at least these requirements can not be certified
2	O	Optional requirement - when implemented, the KNX requirements shall be met
3	F	Recommendations (free to implement)
4	S	Feature of standardized solutions
5	VI	Visual inspection (test guidelines)

If the names of basic and system components/devices have been standardized respectively exclusively assigned to this type of products (e.g. BCU), non-standardized versions may not bear this same name. For the example given above, the system device would have to be named BAU or Bus Access Unit.

Note: For commercially available basic and system components/devices, consult the KNX Directory of registered/certified solutions.

2 Introduction

The name BCU is reserved for standardised bus access units, as described underneath. The term BCU shall not be used to denote bus access units not complying with the underneath BCU requirements.

A BCU serves for modular mounting of application modules. It may contain the application-program or just serve as bus-interface. Concerning reconfiguration the bus-coupling-unit can be programmed via the bus as well as via the PEI.

3 TP1 BCU1

3.1 Communication requirements

3.1.1 General

- CPU: MC68HC05B6 or compatible type
- Operating Frequency: 2,0 MHz (Crystal Frequency of 4MHz)
- On-Chip RAM: 176 Bytes, (18 Bytes available for user)
- On-Chip EEPROM: 256 Bytes, (230 Bytes available for user)
- 8-Bit A/D-Converter (5 Channels available for user)
- 8-Bit Pulse Length Modulator (PLM)
- Serial Asynchronous Communication-Interface
- Serial Synchronous Communication-Interface
- Watch Dog
- fully implemented PEI (mechanically, electrically and signalling)
- Programming button and LED
- Housing
- EMC basic protection

3.1.2 Profile

The TP1 BCU shall comply with profile “BCU1” in Volume 6

3.1.3 Relation Controller to PEI pins

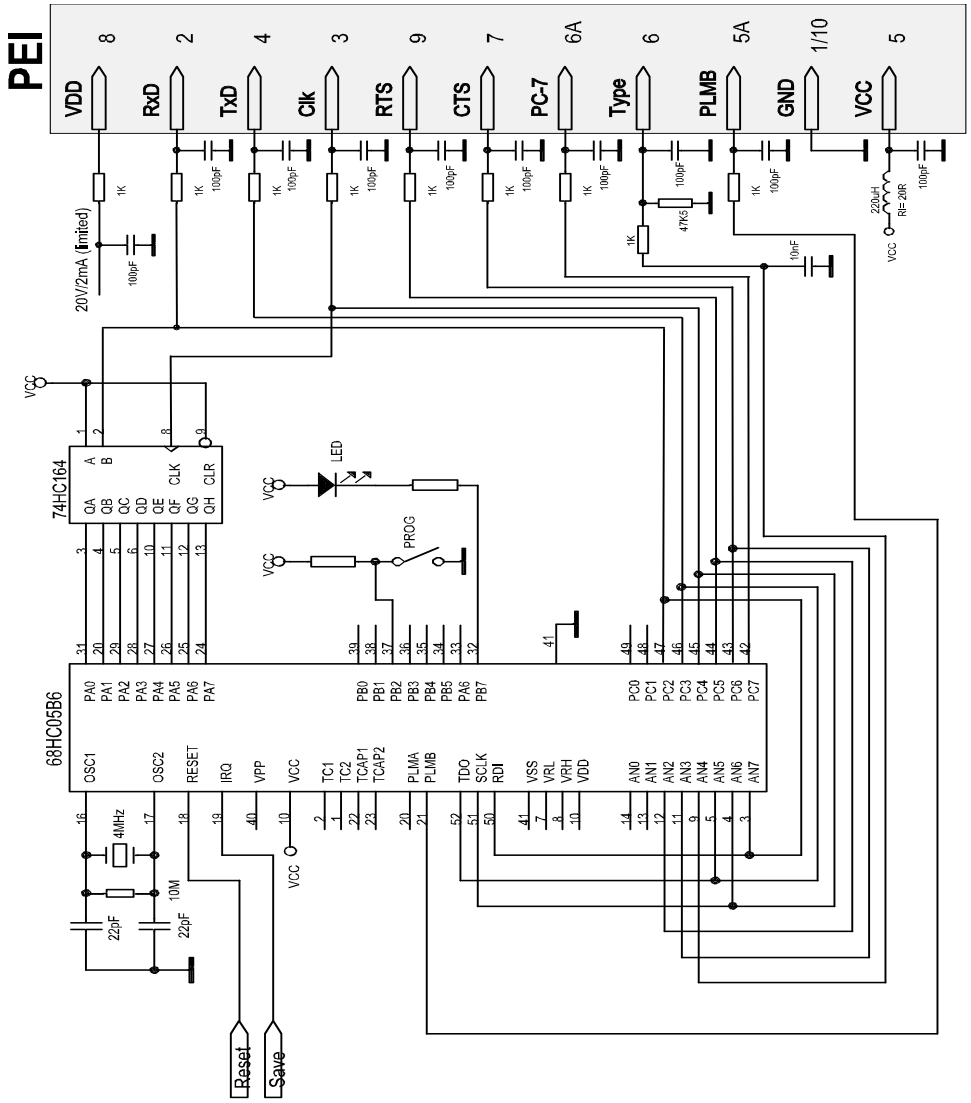

Figure 1: relation controller to PEI pins

3.1.4 TP 1 BCU 1 Software Specification

Three different Mask versions exist from the TP1 BCU 1, Mask version 1.0, 1.1 and Mask version 1.2. All versions are upwards-compatible except the entry point of the system function.

3.1.4.1 Structure of the TP1 BCU 1 System Software

The system software shall consist of two parts: a sequential part and an interrupt-driven part. The interrupt-driven part is responsible for receiving messages from the bus and for the start of the user-save-routine. The sequential part is a large loop, which is shown below.

The TP1 BCU1 cycle time (time needed by the BCU to complete a cycle from the application program to the lower layers and back again) shall be between 3 to 5 ms.

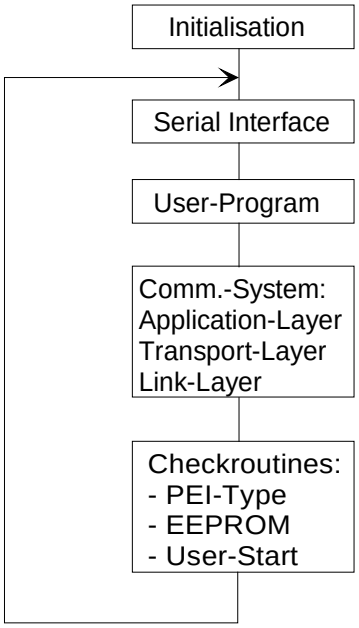

Figure 2: sequential part of TP1 BCU1 system software

3.1.4.2 Addressing

See Volume 6 'BCU1 Profile'.

3.1.4.3 PEI Type Detection

Shall be in accordance with Volume 3/6/2.

3.1.5 Modes of Operation

The TP1 BCU 1 shall have a number of special operation modes, which shall be offered by the system software. After a reset the TP1 BCU1 shall be in normal operation mode. The special modes shall be reached over various operations during start-up or during normal operation.

3.1.5.1 Busmonitor Mode

In busmonitor mode all messages from the Linklayer shall be directly sent to the PEI. The BCU shall be completely inactive on the bus. Each ACK, NACK and BUSY on the bus shall generate a message to the PEI. The busmonitor mode is entered by setting the system state (see paragraph 3.1.10.2.1.2).

3.1.5.2 User-Program-Start

The user-program shall only be started if the EEPROM and the group-object- and association-table-data are ok (i.e. the corresponding error-flags are not set (low active), see 3.1.10.3.12) and the current PEI type matches the required PEI type in EEPROM.

If one of the above conditions is not fulfilled, the user-program shall be stopped.

If later on, the above conditions are fulfilled, the user-program may be restarted.

Start or restart of the user-program shall always cause that

- the user-initialization-routine is called
- the User-RAM is cleared
- the user-routine is periodically called.

Each time before the user-routine is called, the registers (RegB - RegN) shall be cleared.

The Maximum time to stay in user program may not exceed 10 ms.

3.1.5.3 User Program Start-up Inhibit Mode

As long as the programming button is pressed on start-up, the user program shall not be activated.

3.1.6 Initialization

The initialization of the TP1 BCU1 shall include the following steps:

1. wait until save signal is released
2. initialize hardware
3. initialize system software
4. read PEI type
5. check programming-push-button for user inhibit mode
6. wait for start up
7. check checksum
8. start scheduler

3.1.7 Scheduler

The following tasks shall be cyclically called by the scheduler:

- all communication tasks (LL, NL, TL, AL, MG)
- check routines (stack, system state)
- check for user start

3.1.8 Check-Routines

The EEPROM shall be periodically checked (each 256th cycle). In the case of an error, the corresponding runtime-error-flag shall be set (see 3.1.10.3.12).

It shall be possible to read and reset the runtime-error-flags via the bus.

In addition to a correct checksum, the following conditions shall be fulfilled to assume a valid EEPROM-contents:

1. Length of address table < 116
2. PEI-type in EEPROM < 20
3. Pointer to RAM-flags-table in group-objects-table points to User RAM-area.

Other internal consistency checks may also cause a restart of the TP1 BCU1 in the case of error (e.g. stack overflow). The causes for a restart may be manufacturer dependant.

3.1.9 Watchdog

The Watchdog-System shall be automatically started by the system-software. Taking into account the time that may be consumed by interrupt-routines, e.g. bit receive, the watchdog shall be triggered approximately every 1.5ms.

The watchdog shall be triggered just before calling the user program and just after returning from it. In the User-Program itself, the application program writer shall be responsible for properly triggering the watchdog.

3.1.10 Compatibility

Compatibility

Application Programmers using TP1 BCU1 shall be discouraged from using **undocumented features** of the TP1 BCU1 as such developed application programs may not run on updates of the TP1 BCU1 system software!

They shall furthermore be warned not to write complete bytes on Port C of the TP1 BCU1, by means of e.g. the following commands:

sta PORTC ; addr. 02h
stx PORTC

3.1.10.1 ADC Channels

The ADC channel 4 shall be used to detect the hardware PEI type.

The TP1 BCU1 shall allow a coarse estimation of the current bus via the AD-converter channel 1 and the AdcRead-service. The value read can be converted to a voltage value by using the following formula:

Voltage = ADC_Value * 0,15V

3.1.10.2 Memory Map

3.1.10.2.1 RAM

3.1.10.2.1.1 General

RAM Address	Name	Length [bytes]	Comment
0x0050	RegB	1	Register B
0x0051	RegC	1	Register C
0x0052	RegD	1	Register D
0x0053	RegE	1	Register E
0x0054	RegF	1	Register F
0x0055	RegG	1	Register G
0x0056	RegH	1	Register H
0x0057	RegI	1	Register I
0x0058	RegJ	1	Register J
0x0059	RegK	1	Register K
0x005A	RegL	1	Register L
0x005B	RegM	1	Register M
0x005C	RegN	1	Register N
0x005D 0x005F	reserved	3	System Software
0x0060	SystemState	1	State for each layer
0x0061 0x00CD	reserved	113	System Software
0x00CE 0x00DF	UserRAM	18	RAM-Area for Comms-Object- and Application-Data

0x00E0 0x00FF	reserved	32	System Software and Stack Space

Figure 3: Memory Map General

It shall be possible for the application programmer to use all variables called "register" as temporary RAM-storage. The state of the temporary registers shall be changed after RTS of the User Program.

However, some of these variables may also be used by ROM-functions and for parameter passing or as temporary variables (See function descriptions!)

3.1.10.2.1.2 System State

A variable at address $ 60 (TP1 BCU1 RAM) shall be used to specify the operation mode of the TP1 BCU1

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PARITY	DM	UE	SE	ALE	TLE	LLM	PROG

PROG

If this bit is set, the TP1 BCU1 shall be in programming-mode else in normal operation-mode.

LLM

If this bit is set, the link-layer shall be in normal operation-mode else in busmonitor-mode.

TLE

If this bit is set, the transport-layer shall be enabled else disabled.

ALE

If this bit is set, the application-layer shall be enabled else disabled.

SE

If this bit is set, the serial PEI-interface (message-protocol) shall be enabled else disabled.

UE

If this bit is set, the user program shall be enabled, else disabled.

DM

If this bit is set, the TP1 BCU1 shall be in programming-mode (ready to accept its individual address) else in normal operation-mode

PARITY

This bit shall be the parity-bit for this byte. Even parity shall be used.

To activate a layer, a user shall write at address $ 60 in the TP1 BCU1's RAM the legal value:

Mode	Value
Busmonitor	90h
Link-Layer	12h
Transport-Layer	96h
Application-Layer	1Eh
Reset	C0h

After a reset, the Application-Layer shall always be activated.

3.1.10.3 EEPROM

3.1.10.3.1 General

memory address	name	length	Comment
$0100	OptionReg	1	EEPROM Option Register
$0101-$0103	ManData	3	TP1 BCU1-Manufacturing Data
$0104	Manufact	1	EEPROM-Software Manufacturer
$0105-$0106	DevTyp	2	Device Type Number
$0107	Version	1	Software-Version Number
$0108	CheckLim	1	EEPROM check limit
$0109	PEI_Type expected	1	required PEI-Type by Software
$010A	SyncRate	1	baud rate used for serial synchronous PEI
$010B	PortCDDR	1	Port C Direction Bit Setting for PEI-Type 17
$010C	PortADDR	1	Port A Direction Bit Setting
$010D	RunError	1	Run Time Error Flags
$010E	RouteCnt	1	Routing-count constant
$010F	MxRstCnt	1	INAK-Retransmit-Limit BUSY-Retransmit-Limit
$0110	ConfigDes	1	Configuration Descriptor
$0111	AssocTabPtr	1	Pointer to Association Table
$0112	CommsTabPtr	1	Pointer to Group Object Table
$0113	UsrInitPtr	1	Pointer to USER Initialization Routine
$0114	UsrPrgPtr	1	Pointer to USER Program
$0115	UsrSavPtr	1	Pointer to USER Save Program
$0116-$01FE	UsrEEPROM	233	User EEPROM Start with address table
$01FF	EE_EXOR	1	Checksum

Figure 4: Memory map - EEPROM

3.1.10.3.2 OptionReg

Bits 0 and 1 of this byte are determined by the processor-hardware (see data-sheet).

Bit #	7	6	5	4	3	2	1	0
Meaning	shall all be 1 (reserved for system software)						EE1P (MC68HC05-specific) 1 = EEPROM $ 120-1FF writable 0 = EEPROM $ 120-1FF protected	SEC (MC68HC05-specific)

3.1.10.3.3 ManData

This EEPROM address may contain manufacturing data from the manufacturer of TP1 BCU1.

The supplier of the TP1 BCU1 shall issue a warning to the application developer not to change this data!

3.1.10.3.4 Manufact

This EEPROM address shall contain the manufacturer code of the manufacturer providing the EEPROM-software. A list of the currently assigned manufacturer codes can be obtained from the Association.

3.1.10.3.5 DevTyp

This EEPROM address shall contain a unique code to identify the device type, according to an own device code table of the application programmer. A unique numbering within KNX is preferable.

Device code table (hex):

0000 reserved (esc)

0001 ... (defined by the application programmer)

3.1.10.3.6 Version

This EEPROM address shall contain the version number of the EEPROM-software, formatted as follows:

Bit #	7	6	5	4	3	2	1	0
Meaning	Main Version Number (0-F)				Sub Version Number (0-F)			

3.1.10.3.7 CheckLim

This EEPROM address may contain the EEPROM area to be secured by the checksum routine, whereby:

Check range: 108 to (CheckLim-1)

Legal values: 09h to FFh

Each time a value is written to this area the checksum shall be automatically updated. The checksum shall moreover be periodically checked. If there an error is detected, the corresponding bit in the EEPROM error flags (3.1.10.3.12) shall be set. If the bit is set (=0), the length of the address table shall be set to 1 and the User Program stopped.

If the application programmer stores an object value in EEPROM, then it shall be outside this check-range.

3.1.10.3.8 PEI-Type

This EEPROM address shall contain the mechanical PEI-type required by the application software stored in EEPROM[1].

The TP1 BCU1 shall allow reading the actual PEI-type via the AD-converter channel 4 and the AdcRead-service. The value read can be converted to the actual PEI-type by using the following formula:

$$PEI\_Type = (10 * ADC\_Value + 60)/128$$

If this PEI-type does not match the one read from the PEI, the User-program shall be stopped.

3.1.10.3.9 SyncRate

This EEPROM address shall contain the baud rate used for the serial synchronous PEI in the format and interpretation as documented in the MC68HC05B6-data-sheet (baud rate register).

The crystal-frequency of the TP1 BCU1 shall be 4.0MHz.

3.1.10.3.10 PortCDDR

This EEPROM address may contain the direction bit setting for the port C with the following format: Bit 0-7= PortC 0-7, 0 = Input , 1 = Output.

This value shall be used only if the PEI-Type 17 is used and the EEPROM is assumed to be ok.

Attention : A warning shall be given to application programmers that the use of this feature may be a potential source of incompatibility, of which the use shall therefore be avoided!

3.1.10.3.11 PortADDR

This EEPROM address shall contain the direction bit setting for the port A with the following format: Bit 0-7= PortA 0-7, 0 = Input, 1 = Output.

This value shall be used only if the EEPROM is assumed to be ok. Otherwise the port A shall be set to input.

This value shall be set during the manufacturing process of the application programmer. The ETS uses it to verify whether the application program to be loaded matches the hardware environment.

3.1.10.3.12 RunError

In this set of flags, runtime-errors shall be stored for error analysis purposes.

A flag shall be considered set if the corresponding bit = 0.

The flags shall only be set by the system. They shall be explicitly cleared by a management-tool.

The Run time error flags shall be coded as follows:

Bit #	7	6	5	4	3	2	1	0
Meaning	<reserved for system software> must be 1	SYS3_ERR	SYS2_ERR	OBJ_ERR	STK_OVL	EEPROM_ERR	SYS1_ERR	SYS0_ERR

SYS0_ERR This bit shall indicate a buffer error, i.e. a message buffer was addressed to a non-existing or inactive layer

SYS1_ERR This bit shall indicate a layer-error, i.e. System state has a parity error.

[1] In some cases, the value of the mechanical PEI may be different from the value in EEPROM, e.g. to allow that a user program is only started when the application module is removed (else the external user program runs).

EEPROM_ERR This bit shall indicate that the EEPROM-check has detected an error respectively that the EEPROM-data are assumed to be corrupted. This error shall inhibit user-execution.

STK_OVL This bit shall indicate that a stack overflow was detected.

OBJ_ERR This bit shall indicate that AL has detected an error in the group-object- or association-table, in most cases due to inconsistent EEPROM-data. This error shall inhibit user-execution.

SYS2_ERR This bit shall indicate a transceiver error, i.e. during sending no receive signal was detected.

SYS3_ERR This bit shall indicate that a confirm error, i.e. a T_Data_Group.con was received on an object, of which the status is not “transmitting”.

3.1.10.3.13 RouteCnt

This byte shall contain the constant start value for the routing counter.

The range of legal values is 0-7.

The byte shall be coded as follows:

Bit #	7	6	5	4	3	2	1	0
Meaning	shall be 0 (*)(°)	routing counter			shall be 0 (°)	shall be 0 (°)	shall be 0 (°)	shall be 0 (°)

(*) Mask versions 1.2 shall allow the enabling of the function U_DELMSG by setting this bit to 1. This function shall then be called just before the user main routine.

(°) reserved for system software

3.1.10.3.14 MxRstCnt

This byte shall contain the number of LL retransmissions in case of transmission errors (LL-INAK) and due to busy replies (LL-BUSY).

The Byte shall be coded as follows

Bit #	7	6	5	4	3	2	1	0
Meaning	BUSY-Retransmit-Limit			shall be 0 (°)	shall be 0 (°)	INAK-Retransmit-Limit		

(°): reserved for system software

3.1.10.3.15 ConfigDes

It shall be possible to set a number of optional features of the system software in this EEPROM byte.

Bit #	7	6	5	4

Meaning	shall be 1 (reserved for system software)	U_EVENT.ind-message-generation (see Volume 3/6/3), only active when user is executed and serial PEI is active [0 = enabled, 1 = disabled]	shall be 1 (reserved for system software)	shall be 0 (reserved for system software)
Bit #	3	2	1	0
Meaning	telegram rate limitation(limit at 100H+[UsrInitPtr]-1) 0 = enabled 1 = disabled	CPOL clock phase for serial synchronous interface(see MC68HC05B6 data sheet)	CPHA clock phase for serial synchronous interface(see MC68HC05B6 data sheet)	shall be 1 (deactivating possible PLMA pulse generations)

Notes :

If the telegram-rate-limitation is enabled, the application-layer shall not generate more than the specified number (1-127) of group telegrams in approximately 17 seconds. A side effect of this feature is, that the TP1 BCU1 may not generate group telegrams within the first 17 seconds after restart.

3.1.10.3.16 AssocTabPtr

This byte shall contain the pointer to the association table. The Association Table shall comply with the realization type 1 of Volume 3/5/1.

3.1.10.3.17 CommsTabPtr

This byte shall contain the pointer to the group object table. The Group Object Table shall comply with the realization type 1 of Volume 3/5/1.

3.1.10.3.18 UsrInitPtr

UsrInitPtr shall contain the pointer to the entry point of the USER-initialization-routine in EEPROM. The actual initialization-routine shall start at $ 100+[UsrInitPtr].

The initialization-routine shall be called once at user-start-up-time.

The initialization-routine shall be written as a subroutine, i.e. it shall be terminated by "rts".

3.1.10.3.19 UsrPrgPtr

UsrPrgPtr shall contain a pointer to the entry point of the USER-program in EEPROM. The actual USER-program shall then start at $ 100+[UsrPrgPtr].

The USER-program shall be called periodically if the TP1 BCU1 is in normal operation mode.

The USER-program shall be written as a subroutine, i.e. it shall be terminated by "rts".

3.1.10.3.20 UsrSavPtr

UsrSavPtr shall contain a pointer to the entry point of the USER-Save-program in EEPROM. The actual USER-Save-program shall then start at $ 100+[UsrSavPtr].

The USER-Save-program shall be called if the save-signal is generated due to supply-power-breakdown and the user is active at the same time.

After calling the USER-Save-program the TP1 BCU1 shall be reset.

The USER-Save-program shall be written as a subroutine, i.e. it shall be terminated by "rts".

3.1.10.3.21 AdrTab

This byte shall mark the start of the group address table. The Group Address Table shall comply with the realization type 1 of Volume 3/5/1.

3.1.10.3.22 EE_EXOR

This byte shall contain the checksum over the specified part of the EEPROM.

3.1.10.4 API

The Application Programmer's Interface available in the TP1 BCU1 shall comply with the one specified in Volume 3/6/1.

3.1.10.5 Physical and External External Interface

The Physical External Interface shall comply with the requirements of Volume 3/6/2. The External Message Interface shall comply with the EMI1 of Volume 3/6/3.

3.2 Electrical Safety

No.	Requirements	M
1	The requirements of Volume 4 Part 1 shall be complied with.	M/S

3.3 Environmental conditions

No.	Requirements	M
1	TP1 BCU 1 shall comply with the requirements of Volume 4 Part 1 clause 2.1.2	M/S

3.4 EMC

No.	Requirements	M
1	TP1 BCU 1 shall comply with the requirements of Volume 4 Part 1 clause 2.3	M/S

3.5 Mechanical, Dimensions, Constructional Features

3.5.1 General

A press button switch (toggle function) shall be provided on any mechanical design of the TP1 BCU1 to select or deselect the parameterization mode of the TP1 BCU1.

Programming mode shall allow setting or reading via broadcast of the individual address of the TP1 BCU1. This shall be possible without the knowledge of the individual address of the TP1 BCU1.

In programming mode, the LED on the TP1 BCU1 shall be switched on, else off.

3.5.2 Flush mounted TP1 BCU1

This chapter describes the standardised mechanical features of flush-mounted TP1 BCU1 with all necessary information. The dimensions are given in Figure 5.

This type of TP1 BCU 1 is intended to be inserted into wall-boxes, including means for fixing the TP1 BCU1 (frame) by means of screws. The flush mounted TP1 BCU 1 is delivered as a completely assembled unit without mounting frame.

The position of the programming LED and button in the underneath figure is recommended.

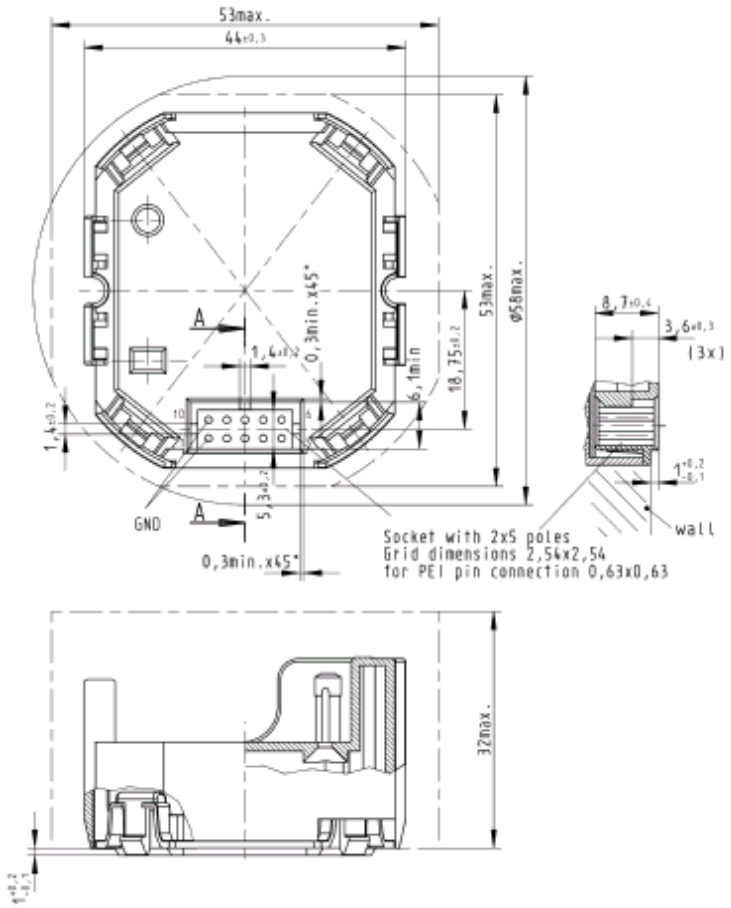

Figure 5: Flush-mounted TP1 BCU1

Warning:

The standardisation of the dimensions of the flush mounted BCU does not - due to the permitted tolerances - in all cases ensure that the PEI of the to be mounted application module fits on the BCU! Therefore it is highly recommended for the design of a BCU to select the dimensions according typical values.

3.5.3 Surface and DIN rail mounted TP1 BCU1

The TP1 BCU 1 as surface and DIN rail mounted components are not standardised.

3.6 Electrical Features

3.6.1 PEI Characteristics

Characteristics	Symbol	Min	Max	typical	Unit	Remarks
Supply Output Voltage +5V	V_{CC}	4.7	5.3		V	See *Power Limitation*
Supply Output Voltage +20V	V_{DD}	18	22		V	$_{load}$=2mA V_{BUS}-V_{DD} >1,5V $V_{Bus=}$ 21V ... 30V
Current Limitation	I_{DD} Icc_{peak}	2	5 20		mA	 for Idd=0
Power Limitation			100 [2]		mW	= Icc.5V + Idd.20V
Data Output Voltage Port A, Port B, Port C	V_{OL} V_{OH}	 V_{CC}-0,8	0.4		V	I_{load}=1,6mA I_{load}=0,8mA
Data Output Voltage TDO, SCLK, PLMB	V_{OL} V_{OH}	 V_{CC}-0,8	0.4		V	I_{load}=1,6mA I_{load}=1,6mA
Data Output Voltage Reset	V_{OL}		1,0		V	I_{load}=1,6mA
Data Input Voltage Port A, Port B, Port C Reset,RDI	V_{IL} V_{IH}	0 0.7 V_{CC}	0.2 V_{CC} V_{CC}		V	
Analog Input Voltage Range	V_{AIL} V_{AIH}	0	 VCC		V	
I/O Ports Three-State-Leakage	I_{OZ}		±10		µA	
Input Capacitance	C_{IN}		50		pF	

Figure 6: PEI characteristics

[2] A power consumption of > 50mW reduces time for the saving routine.

3.6.2 Timing PEI

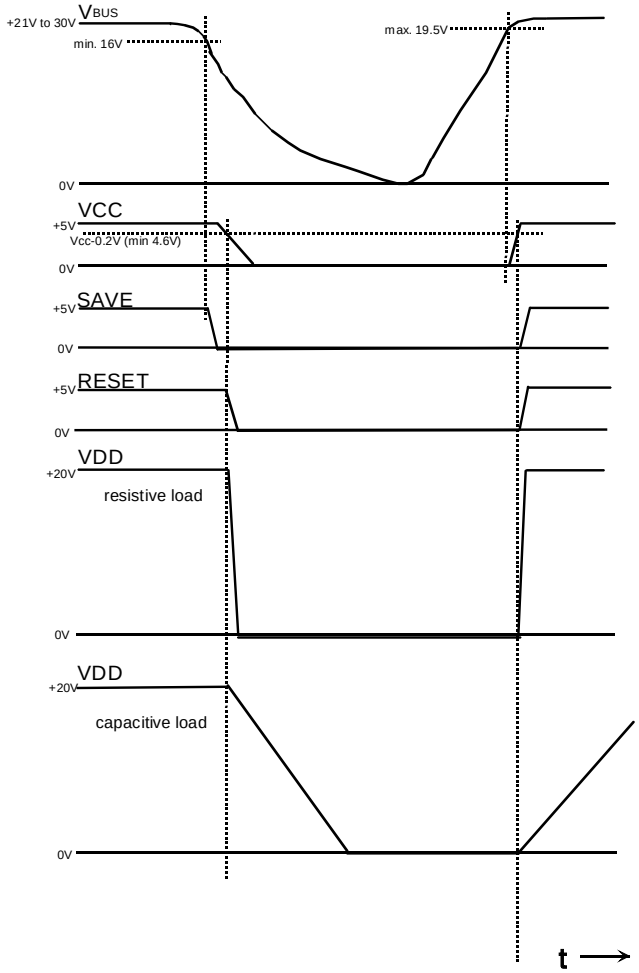

Figure 7: PEI Timing

3.7 Testing

The following test shall be carried out to show compliance:

- communication testing : according Volume 8 System Conformance testing
- Environmental conditions: according Volume 4 Part 3
- EMC: according Volume 4 Part 2
- Mechanical and electrical properties according to this Handbook clause

3.8 Functional Safety

Under Consideration

3.9 Interfaces, Connectors

The TP1 BCU1 shall be connected with the following standard connectors:

- For the connection to the bus : type 5.1 according Part 9/1.
- For the connection to the application module: type 4.1 as described in Part 9/1 with 10 pole PEI

3.10 Marking

The TP1 BCU 1 can be marked by the manufacturer at his discretion.

3.11 Installation

Not applicable

3.12 Symbols

to be completed.

4 PL110 BCU1

4.1 Communication requirements

4.1.1 General

- CPU: MC68HC05B16 or compatible type
- Operating Frequency: 4,0 MHz (Crystal Frequency of 4MHz)
- On-Chip RAM: 352 Bytes, (144 Bytes available for user)
- On-Chip EEPROM: 256 Bytes, (230 Bytes available for user)
- 8-Bit A/D-Converter (5 Channels available for user)
- 8-Bit Pulse Length Modulator (PLM)
- Serial Asynchronous Communication-Interface
- Serial Synchronous Communication-Interface
- Watch Dog
- fully implemented PEI (mechanically, electrically and signalling)
- Programming button and LED
- Housing
- EMC basic protection

4.1.2 Profile

The PL110 BCU shall comply with profile “BCU1” in Volume 6

4.1.3 Relation Controller to PEI pins

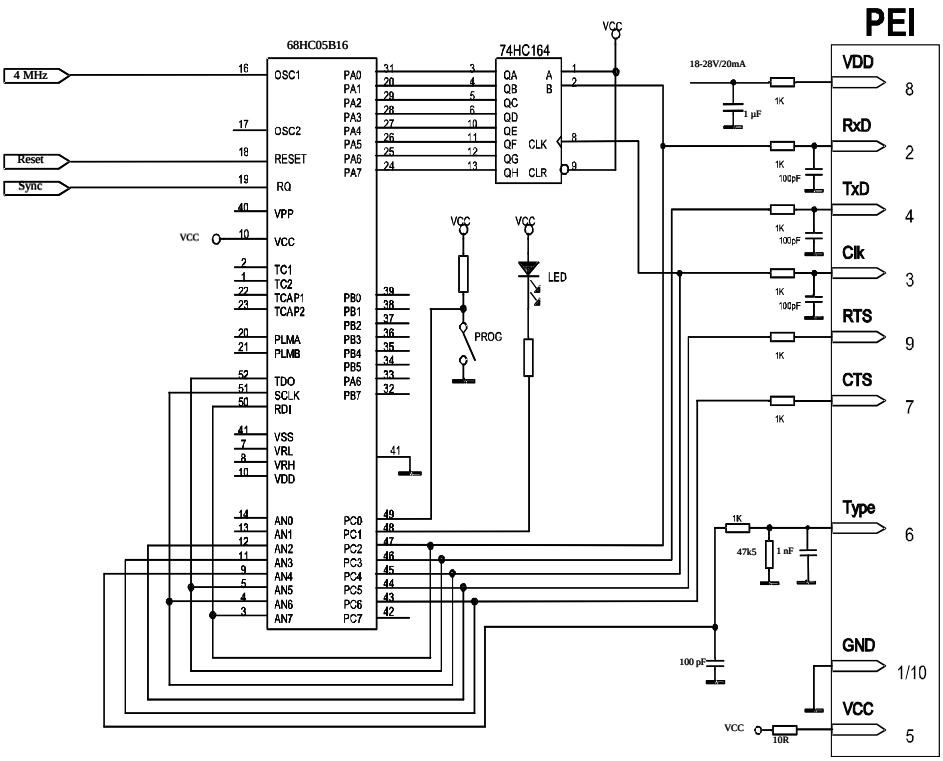

Figure 8: relation controller to PEI pins

4.1.4 PL110 BCU 1 Software Specification

One Mask versions exist from the PL110 BCU 1: Mask version 10.13.

4.1.4.1 Structure of the PL110 BCU 1 System Software

The system software shall consist of two parts: a sequential part and an interrupt-driven part. The interrupt-driven part is responsible for synchronizing and for the start of the user-save-routine. The sequential part is a large loop, which is shown below.

The PL110 BCU1 cycle time (time needed by the BCU to complete a cycle from the application program to the lower layers and back again) shall be between 4 to 6 ms.

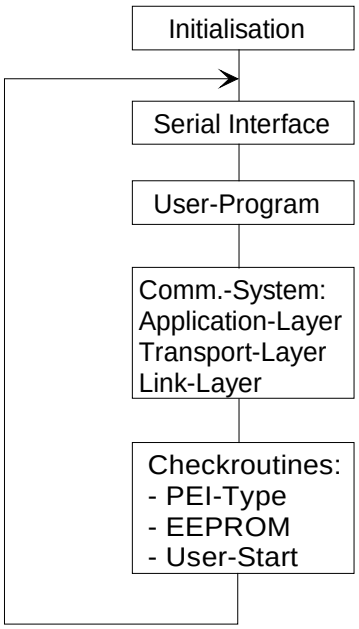

Figure 9: sequential part of PL110 BCU1 system software

4.1.4.2 Addressing

See Volume 6 ‘BCU1 Profile’.

4.1.4.3 PEI Type Detection

Shall be in accordance with Volume 3/6/2.

4.1.5 Modes of Operation

The PL110 BCU 1 shall have a number of special operation modes, which shall be offered by the system software. After a reset the PL110 BCU1 shall be in normal operation mode. The special modes shall be reached over various operations during start-up or during normal operation.

4.1.5.1 Busmonitor Mode

In busmonitor mode all messages from the Linklayer shall be directly sent to the PEI. The BCU shall be completely inactive on the bus. Each ACK and NACK on the bus shall generate a message to the PEI. The busmonitor mode is entered by setting the system state (see paragraph 4.1.10.2.1.2).

4.1.5.2 Extended Busmonitor Mode

If bit 3 in EEPROM $0101 is switched to 0 the PL110 BCU1 switches from L_Busmon.ind to L_Busmon_ext.ind and transmits the domain address as shown below:

1. Length	1 Byte
2. Message Code	1 Byte
3. Status	1 Byte
4. Time Stamp	2 Byte
5. Control Field	1 Byte
6. Domain Address	2 Byte
7. Source Address	2 Byte
8. Destination Addr.	2 Byte
9. Data	1..14 Byte

...

End: Check Sum	1 Byte

4.1.5.3 User-Program-Start

The user-program shall only be started if the EEPROM and the group object- and association-table-data are ok (i.e. the corresponding error-flags are not set (low active), see 4.1.10.3.13) and the current PEI type matches the required PEI type in EEPROM.

If one of the above conditions is not fulfilled, the user-program shall be stopped.

If later on, the above conditions are fulfilled, the user-program may be restarted.

Start or restart of the user-program shall always cause that

- the user-initialization-routine is called
- the User-RAM is cleared
- the user-routine is periodically called.

Each time before the user-routine is called, the registers (RegB - RegN) shall be cleared.

The Maximum time to stay in user program may not exceed 10 ms.

4.1.5.4 User Program Start-up Inhibit Mode

As long as the programming button is pressed on start-up, the user program shall not be activated.

4.1.6 Initialization

The initialization of the PL110 BCU1 shall include the following steps:

1. wait until save signal is released
2. initialize hardware
3. initialize system software
4. read PEI type
5. check programming-push-button for user inhibit mode
6. wait for start-up
7. check checksum
8. start scheduler

4.1.7 Scheduler

The following tasks shall be cyclically called by the scheduler:

- all communication tasks (LL, NL, TL, AL, MG)
- check routines (stack, system state)
- check for user start

4.1.8 Check-Routines

The EEPROM shall be periodically checked (each 256th cycle). In the case of an error, the corresponding runtime-error-flag shall be set (see 4.1.10.3.13)

It shall be possible to read and reset the runtime-error-flags via the bus.

In addition to a correct checksum, the following conditions shall be fulfilled to assume a valid EEPROM-content:

1. Length of address table < 116
2. PEI-type in EEPROM < 20
3. Pointer to RAM-flags-table in group-objects-table points to User RAM-area.

Other internal consistency checks may also cause a restart of the PL110 BCU1 in the case of error (e.g. stack overflow). The causes for a restart may be manufacturer dependant.

4.1.9 Watchdog

The Watchdog-System shall be automatically started by the system-software. Taking into account the time that may be consumed by interrupt-routines, e.g. bit receive, the watchdog shall be triggered approximately every 1.5ms.

The watchdog shall be triggered just before calling the user program and just after returning from it. In the User-Program itself, the application program writer shall be responsible for properly triggering the watchdog.

4.1.10 Compatibility

Compatibility

Application Programmers using PL110 BCU1 shall be discouraged from using **undocumented features** of the PL110 BCU1 as such developed application programs may not run on updates of the PL110 BCU1 system software!
They shall furthermore be warned not to write complete bytes on Port C of the PL110 BCU1, by means of e.g. the following commands:

```
sta PORTC          ; addr. 02h
stx PORTC
```

4.1.10.1 ADC Channels

The ADC channel 4 shall be used to detect the hardware PEI type.

The PL110 BCU1 shall allow a coarse estimation of the current bus via the AD-converter channel 1 and the AdcRead-service. The value read can be converted to a voltage value by using the following formula:

$$Voltage = ADC\_Value * 0,15V$$

4.1.10.2 Memory Map

4.1.10.2.1 RAM

4.1.10.2.1.1 General

RAM Address	Name	Length [bytes]	Comment
0x0050	RegB	1	Register B
0x0051	RegC	1	Register C
0x0052	RegD	1	Register D
0x0053	RegE	1	Register E
0x0054	RegF	1	Register F
0x0055	RegG	1	Register G
0x0056	RegH	1	Register H
0x0057	RegI	1	Register I
0x0058	RegJ	1	Register J
0x0059	RegK	1	Register K
0x005A	RegL	1	Register L
0x005B	RegM	1	Register M
0x005C	RegN	1	Register N
0x005D 0x005F	reserved	3	System Software
0x0060	SystemState	1	State for each layer
0x0061 0x00CD	reserved	113	System Software
0x00CE 0x00DF	UserRAM	18	RAM-Area for Comms-Object- and Application-Data
0x00E0 0x00FF	reserved	32	System Software and Stack Space
0x0281 0x02FF	UserRAM	126	RAM-Area for Comms-Object- and Application-Data Note: For Comms-Object the Communication Config Byte bit 7 shall be set to “0” an bit 5 shall be set to “1”.

Figure 10: Memory Map General

It shall be possible for the application programmer to use all variables called "register" as temporary RAM-storage. The state of the temporary registers shall be changed after RTS of the User Program.

However, some of these variables may also be used by ROM-functions and for parameter passing or as temporary variables (See function descriptions!)

4.1.10.2.1.2 System State

A variable at address $ 60 (PL110 BCU1 RAM) shall be used to specify the operation mode of the PL110 BCU1

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PARITY	DM	UE	SE	ALE	TLE	LLM	PROG

PROG

If this bit is set, the PL110 BCU1 shall be in programming-mode else in normal operation-mode.

LLM

If this bit is set, the link-layer shall be in normal operation-mode else in busmonitor-mode.

TLE

If this bit is set, the transport-layer shall be enabled else disabled.

ALE

If this bit is set, the application-layer shall be enabled else disabled.

SE

If this bit is set, the serial PEI-interface (message-protocol) shall be enabled else disabled.

UE

If this bit is set, the user program shall be enabled, else disabled.

DM

If this bit is set, the PL110 BCU1 shall be in programming-mode (ready to accept its individual address) else in normal operation-mode

PARITY

This bit shall be the parity-bit for this byte. Even parity shall be used.

To activate a layer, a user shall write at address $ 60 in the PL110 BCU1's RAM the legal value:

Mode	Value
Busmonitor	90h
Link-Layer	12h
Transport-Layer	96h
Application-Layer	1Eh
Reset	C0h

After a reset, the Application-Layer shall always be activated.

4.1.10.3 EEPROM

4.1.10.3.1 General

memory address	name	length	Comment
$0100	OptionReg	1	EEPROM Option Register
$0101	BaseConfig	1	PL110 BCU1-Basic Configuration

$0102-$0103	SysID	2	Domain Address
$0104	Manufact	1	EEPROM-Software Manufacturer
$0105-$0106	DevTyp	2	Device Type Number
$0107	Version	1	Software-Version Number
$0108	CheckLim	1	EEPROM check limit
$0109	PEI_Type expected	1	required PEI-Type by Software
$010A	SyncRate	1	baud rate used for serial synchronous PEI
$010B	PortCDDR	1	Port C Direction Bit Setting for PEI-Type 17
$010C	PortADDR	1	Port A Direction Bit Setting
$010D	RunError	1	Run Time Error Flags
$010E	RouteCnt	1	Routing-count constant
$010F	MxRstCnt	1	INAK-Retransmit-Limit BUSY-Retransmit-Limit
$0110	ConfigDes	1	Configuration Descriptor
$0111	AssocTabPtr	1	Pointer to Association Table
$0112	CommsTabPtr	1	Pointer to Group Object Table
$0113	UsrInitPtr	1	Pointer to USER Initialization Routine
$0114	UsrPrgPtr	1	Pointer to USER Program
$0115	UsrSavPtr	1	Pointer to USER Save Program
$0116-$01FE	UsrEEPROM	233	User EEPROM Start with address table
$01FF	EE_EXOR	1	Checksum

Figure 11: Memory map - EEPROM

4.1.10.3.2 OptionReg

Bits 0 and 1 of this byte are determined by the processor-hardware (see data-sheet).

Bit #	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Meaning	shall all be 1 (reserved for system software)						EE1P (MC68HC05-specific) 1 = EEPROM $ 120-1FF writable 0 = EEPROM $ 120-1FF protected	SEC (MC68HC05-specific)

4.1.10.3.3 BaseConfig

Bit #	7	6	5	4
Meaning	shall all be 1 (reserved for system software)			Repeater present 0 = active 1 = inactive
Bit #	**3**	**2**	**1**	**0**
Meaning	Busmonitor display mode 1 = normal 0 = extended	shall all be 1 (reserved for system software)	Repeater mode[3] 0 = device is repeater 1 = device is not repeater	shall all be 1 (reserved for system software)

Repeater exists: This bit shall indicate whether a repeater is installed within the domain.

Busmonintor display mode: Clearing this bit shall generate extended busmonitor mode telegrams including the Domain Address.

Repeater mode: Clearing this bit shall declare the local device as repeater.

4.1.10.3.4 Domain Address

The domain address range shall start at 0x0001 to 0x00FF. The domain address 0x0000 shall be reserved for system broadcasts and shall not be used.

4.1.10.3.5 Manufact

This EEPROM address shall contain the manufacturer code of the manufacturer providing the EEPROM-software. A list of the currently assigned manufacturer codes can be obtained from the Association.

4.1.10.3.6 DevTyp

This EEPROM address shall contain a unique code to identify the device type, according to an own device code table of the application programmer. A unique numbering within KNX is preferable.

Device code table (hex):

0000 reserved (esc)

0001 ... (defined by the application programmer)

4.1.10.3.7 Version

This EEPROM address shall contain the version number of the EEPROM-software, formatted as follows:

Bit #	7	6	5	4	3	2	1	0
Meaning	Main Version Number (0-F)				Sub Version Number (0-F)			

4.1.10.3.8 CheckLim

This EEPROM address may contain the EEPROM area to be secured by the checksum routine, whereby:

[3] This bit shall only be evaluated if bit 4 is set to zero.

Check range: 108 to (CheckLim-1)

Legal values: 09h to FFh

Each time a value is written to this area the checksum shall be automatically updated. The checksum shall moreover be periodically checked. If there an error is detected, the corresponding bit in the EEPROM error flags (4.1.10.3.13) shall be set. If the bit is set (=0), the length of the address table shall be set to 1 and the User Program stopped.

If the application programmer stores a group object value in EEPROM, then it shall be outside this check-range.

4.1.10.3.9 PEI-Type

This EEPROM address shall contain the mechanical PEI-type required by the application software stored in EEPROM[4].

The PL110 BCU1 shall allow reading the actual PEI-type via the AD-converter channel 4 and the AdcRead-service. The value read can be converted to the actual PEI-type by using the following formula:

$$PEI\_Type = (10 * ADC\_Value + 60)/128$$

If this PEI-type does not match the one read from the PEI, the User-program shall be stopped.

4.1.10.3.10 SyncRate

This EEPROM address shall contain the baud rate used for the serial synchronous PEI in the format and interpretation as documented in the MC68HC05B16-data-sheet (baud rate register).

The crystal-frequency of the PL110 BCU1 shall be 4.0MHz.

4.1.10.3.11 PortCDDR

This EEPROM address may contain the direction bit setting for the port C with the following format : Bit 0-7= PortC 0-7, 0 = Input , 1 = Output.

This value shall be used only if the PEI-Type 17 is used and the EEPROM is assumed to be ok.

Attention : A warning shall be given to application programmers that the use of this feature may be a potential source of incompatibility, of which the use shall therefore be avoided!

4.1.10.3.12 PortADDR

This EEPROM address shall contain the direction bit setting for the port A with the following format: Bit 0-7= PortA 0-7, 0 = Input, 1 = Output.

This value shall be used only if the EEPROM is assumed to be ok. Otherwise the port A shall be set to input.

This value shall be set during the manufacturing process by the application programmer. The ETS uses it to verify whether the application program to be loaded matches the hardware environment.

4.1.10.3.13 RunError

In this set of flags, runtime-errors shall be stored for error analysis purposes.

A flag shall be considered set if the corresponding bit = 0.

The flags shall only be set by the system. They shall be explicitly cleared by a management-tool.

The Run time error flags shall be coded as follows:

[4] In some cases, the value of the mechanical PEI may be different from the value in EEPROM, e.g. to allow that a user program is only started when the application module is removed (else the external user program runs).

Bit #	7	6	5	4	3	2	1	0
Meaning	<reserved for system software> must be 1	SYS3_ERR	SYS2_ERR	OBJ_ERR	STK_OVL	EEPROM_ERR	SYS1_ERR	SYS0_ERR

SYS0_ERR This bit shall indicate a buffer error, i.e. a message buffer was addressed to a non-existing or inactive layer

SYS1_ERR This bit shall indicate a layer-error, i.e. System state has a parity error.

EEPROM_ERR This bit shall indicate that the EEPROM-check has detected an error respectively that the EEPROM-data are assumed to be corrupted. This error shall inhibit user-execution.

STK_OVL This bit shall indicate that a stack overflow was detected.

OBJ_ERR This bit shall indicate that AL has detected an error in the group-object- or association-table, in most cases due to inconsistent EEPROM-data. This error shall inhibit user-execution.

SYS2_ERR This bit shall indicate a transceiver error, i.e. during sending no receive signal was detected.

SYS3_ERR This bit shall indicate that a confirm error, i.e. a T_Data_Group.con was received on an object, of which the status is not “transmitting”.

4.1.10.3.14 RouteCnt

This byte shall contain the constant start value for the routing counter.

The range of legal values is 0-7.

The byte shall be coded as follows:

Bit #	7	6	5	4	3	2	1	0
Meaning	shall be 0 (°)	routing counter			shall be 0 (°)	shall be 0 (°)	shall be 0 (°)	shall be 0 (°)

(°) reserved for system software

4.1.10.3.15 ConfigDes

It shall be possible to set a number of optional features of the system software in this EEPROM byte.

Bit #	7	6	5	4
Meaning	shall be 1 (reserved for system software)	U_EVENT.ind-message-generation (see Volume 3/6/3), only active when user is executed and serial PEI is active [0 = enabled, 1 = disabled]	shall be 1 (reserved for system software)	shall be 0 (reserved for system software)
Bit #	3	2	1	0

Meaning	telegram rate limitation(limit at 100H+[UsrInitPtr]-1) 0 = enabled 1 = disabled	CPOL clock phase for serial synchronous interface (see MC68HC05B6 data sheet)	CPHA clock phase for serial synchronous interface (see MC68HC05B6 data sheet)	shall be 1 (deactivating possible PLMA pulse generations)

Notes :

If the telegram-rate-limitation is enabled, the application-layer shall not generate more than the specified number (1-127) of group telegrams in approximately 17 seconds. A side effect of this feature is, that the Pl110 BCU1 may not generate group telegrams within the first 17 seconds after restart.

4.1.10.3.16 AssocTabPtr

This byte shall contain the pointer to the association table. The Association Table shall comply with the realization type 1 of Volume 3/5/1.

4.1.10.3.17 CommsTabPtr

This byte shall contain the pointer to the group object table. The Group Object Table shall comply with the realization type 1 of Volume 3/5/1.

4.1.10.3.18 UsrInitPtr

UsrInitPtr shall contain the pointer to the entry point of the USER-initialization-routine in EEPROM. The actual initialization-routine shall start at $ 100+[UsrInitPtr].

The initialization-routine shall be called once at user-start-up-time.

The initialization-routine shall be written as a subroutine, i.e. it shall be terminated by "rts".

4.1.10.3.19 UsrPrgPtr

UsrPrgPtr shall contain a pointer to the entry point of the USER-program in EEPROM. The actual USER-program shall then start at $ 100+[UsrPrgPtr].

The USER-program shall be called periodically if the PL110 BCU1 is in normal operation mode.

The USER-program shall be written as a subroutine, i.e. it shall be terminated by "rts".

4.1.10.3.20 UsrSavPtr

UsrSavPtr shall contain a pointer to the entry point of the USER-Save-program in EEPROM. The actual USER-Save-program shall then start at $ 100+[UsrSavPtr].

The USER-Save-program shall be called if the save-signal is generated due to supply-power-breakdown and the user is active at the same time.

After calling the USER-Save-program the PL110 BCU1 shall be reset.

The USER-Save-program shall be written as a subroutine, i.e. it shall be terminated by "rts".

4.1.10.3.21 AdrTab

This byte shall mark the start of the group address table. The Group Address Table shall comply with the realization type 1 of Volume 3/5/1.

4.1.10.3.22 EE_EXOR

This byte shall contain the checksum over the specified part of the EEPROM.

4.1.10.4 API

The Application Programmer's Interface available in the PL110 BCU1 shall comply with the one specified in Volume 3/6/1.

4.1.10.5 Physical and External External Interface

The Physical External Interface shall comply with the requirements of Volume 3/6/2. The External Message Interface shall comply with the EMI1 of Volume 3/6/3.

4.2 Electrical Safety

No.	Requirements	M
1	The requirements of Volume 4 Part 1 shall be complied with.	M/S

4.3 Environmental conditions

No.	Requirements	M
1	PL110 BCU 1 shall comply with the requirements of Volume 4 Part 1 clause 2.1.2 to 1.3.3	M/S

4.4 EMC

No.	Requirements	M
1	PL110 BCU1 shall comply with the requirements of Volume 4 Part 1 clause 2.3	M/S
2	PL110 BCU1 shall comply with EN50065-1, Class 116 devices	M/S

4.5 Mechanical, Dimensions, Constructional Features

4.5.1 General

A press button switch (toggle function) shall be provided on any mechanical design of the PL110 BCU1 to select or deselect the parameterisation mode of the PL110 BCU1.

Programming mode shall allow setting or reading via broadcast of the individual address of the PL110 BCU1. This shall be possible without the knowledge of the individual address of the PL110 BCU1.

In programming mode, the LED on the PL110 BCU1 shall be switched on, else off.

4.5.2 Flush mounted PL110 BCU1

This chapter describes the standardised mechanical features of flush-mounted PL110 BCU1 with all necessary information. The dimensions are given in Figure 12 .

This type of PL110 BCU 1 is intended to be inserted into wall-boxes, including means for fixing the PL110 BCU1 (frame) by means of screws. The flush mounted PL110 BCU 1 is delivered as a completely assembled unit without mounting frame.

The position of the programming LED and button in the underneath figure is recommended.

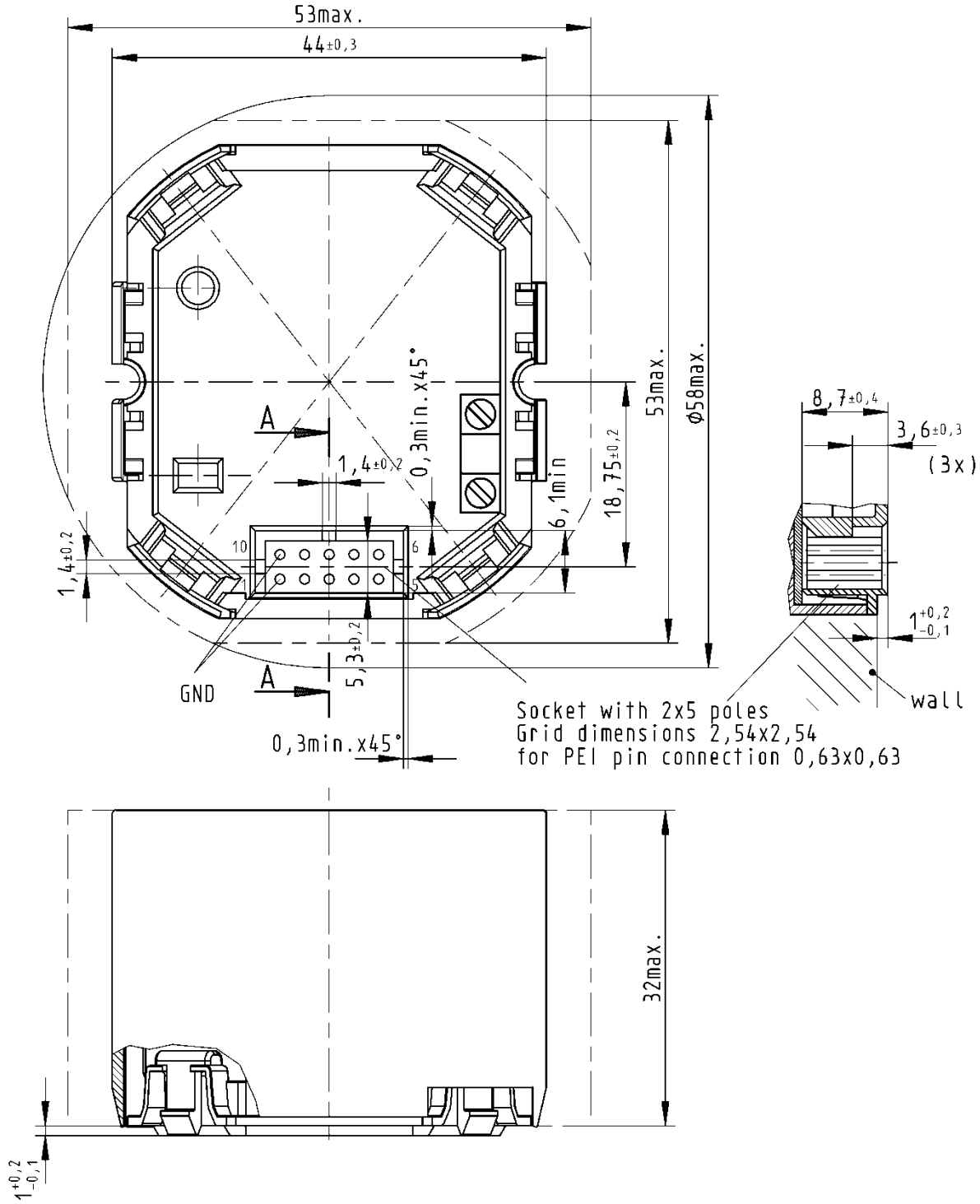

Figure 12: Flush-mounted PL110 BCU1

Warning:

The standardisation of the dimensions of the flush mounted Pl110 BCU1 does not - due to the permitted tolerances - in all cases ensure that the PEI of the to be mounted application module fits on the BCU! Therefore it is highly recommended for the design of a BCU to select the dimensions according typical values.

4.5.3 Surface and DIN rail mounted PL110 BCU1

The PL110 BCU 1 as surface and DIN rail mounted components are not standardized.

4.6 Electrical Features

4.6.1 PEI Characteristics

Characteristics	Symbol	Min	Max	typical	Unit	Remarks
Supply Output Voltage +5V	V_{CC}	4.7	5.3		V	See *Power Limitation*
Supply Output Voltage +20V	V_{DD}	18	28		V	$_{load}$=2mA 18V<V_{DD}<28V
Current Limitation	I_{DD} Icc_{peak}	2	5 20		mA	 for Idd=0
Power Limitation			100 [5]		mW	= Icc.5V + Idd.20V
Data Output Voltage Port A, Port B, Port C	V_{OL} V_{OH}	 V_{CC}-0,8	0.4		V	I_{load}=1,6mA I_{load}=0,8mA
Data Output Voltage TDO, SCLK, PLMB	V_{OL} V_{OH}	 V_{CC}-0,8	0.4		V	I_{load}=1,6mA I_{load}=1,6mA
Data Output Voltage Reset	V_{OL}		1,0		V	I_{load}=1,6mA
Data Input Voltage Port A, Port B, Port C Reset,RDI	V_{IL} V_{IH}	0 0.7 V_{CC}	0.2 V_{CC} V_{CC}		V	
Analog Input Voltage Range	V_{AIL} V_{AIH}	0	 VCC		V	
I/O Ports Three-State-Leakage	I_{OZ}		±10		µA	
Input Capacitance	C_{IN}		50		pF	

Figure 13: PEI characteristics

[5] A power consumption of > 50mW reduces time for the saving routine.

4.6.2 Timing PEI

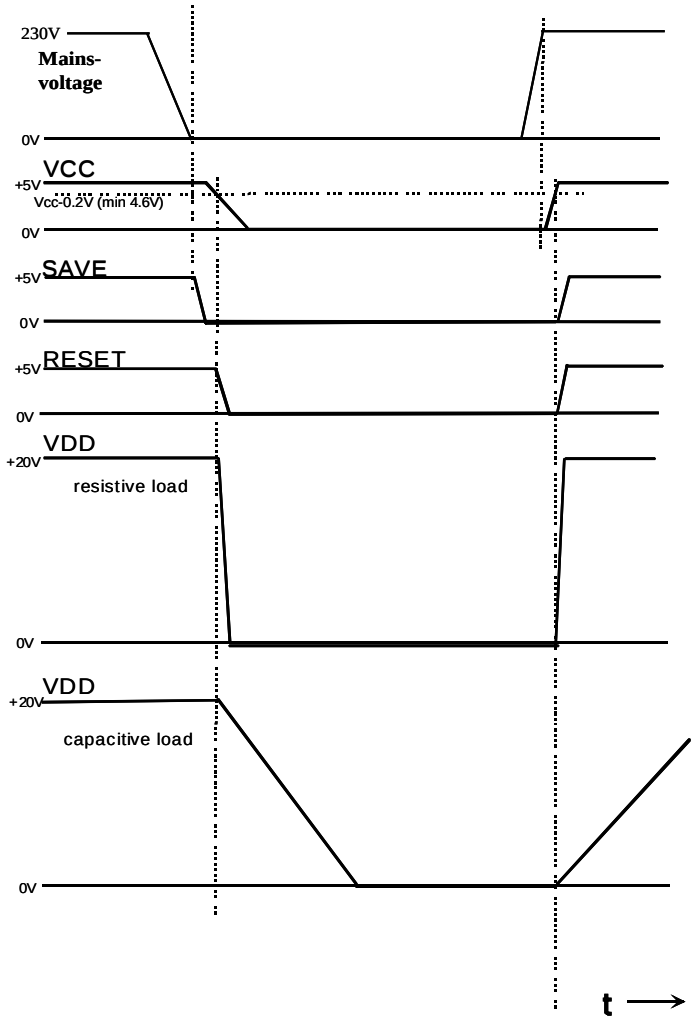

Figure 14: PEI Timing

4.7 Testing

The following test shall be carried out to show compliance:

- communication testing : according Volume 8 System Conformance testing
- Environmental conditions: according Volume 4 Part 3
- EMC: according Volume 4 Part 2
- Mechanical and electrical properties according to this Handbook clause

4.8 Functional Safety

Under Consideration

4.9 Interfaces, Connectors

The PL110 BCU1 shall be connected with the following connectors:

- For the connection to the mains: screwed terminals according EN 60669-2-1.
- For the connection to the application module: type 4.1 as described in Part 9/1 with 10 pole PEI

4.10 Marking

The PL110 BCU 1 can be marked by the manufacturer at his discretion.

According EN 50065-1 the PL110 BCU1 shall be marked with the class of output level, e.g. Cl 116.

4.11 Installation

Not applicable

4.12 Symbols

to be completed.

5 TP1 BCU2

5.1 Communication requirements

5.1.1 Profile

The TP1 BCU2 shall comply with profile “BCU2” in Volume 6

5.1.2 TP 1 BCU 2 Software Specification

5.1.2.1 General

Two different Mask versions exist from the TP1 BCU 2, Mask version 2.0 and 2.1. Both versions are upwards compatible, except the available user RAM. The underneath specification does not include any details on the mask version 2.0: KNX does not recommend the development of a TP1 BCU2 compatible with mask version 2.0.

5.1.2.2 Compatibility

The TP1 BCU 2 shall be fully upward compatible to the TP1 BCU 1 except the entry points of the system functions and some special features listed as follows.

- PLM - Frequency shall be set to 2400 Hz both fast as well as slow;
- due to the crystal frequency of the TP1 BCU2, the absolute time of U_Delay may jitter
- PortD shall not be available at PEI
- Checksum shall only be tested at start-up (compared to the TP1 BCU1 check each 256th cycle)
- BCU2 may contain up to 127 addresses (see EEPROM memory map of TP1 BCU1)
- Address table shall be loaded to allow sending of group telegrams
- when writing the value 00h to address $ 10D, a delay of 1 second shall be taken into account (due to internal EEPROM reconfiguration).
- Access protection (not supported in TP1 BCU1 applications) shall be supported.
- the Error Flags (writing of FFh to 10Dh) shall not be reset without reloading of the application, as this shall activate in the TP1 BCU2 the start-up conditions for the application.
- the start-up time of the application shall depend on the sub network address (provided the address table is loaded).
- Telegram rate limitation shall also be activated during the first 17 seconds.
- It shall be possible to inhibit the start-up of the application in a TP1 BCU2 by briefly pressing the programming button during start-up.
- The sending and the reception of telegrams shall be fully interrupt-driven. Consequently, during sending and receiving the other tasks of the BCU 2 shall be fully functional.
- If the EEPROM is unreadable due to write operations, the BCU 2 shall send a BUSY if it is unable to carry out the address check

When using the TP1 BCU 2 as compatible device for TP1 BCU 1 applications, the BCU1 load procedure (see BCU1 profile) shall be used.

Warning

Application Programmers using TP1 BCU2 shall be discouraged from using **undocumented features** of the TP1 BCU2 as such developed application programs may not run on updates of the TP1 BCU2 system software!
They shall furthermore be warned not to write complete bytes on Port C of the TP1 BCU2, by means of e.g. the following commands:

sta PORTC ; addr. 02h
stx PORTC

5.1.2.3 Structure of the TP1 BCU 2 System Software

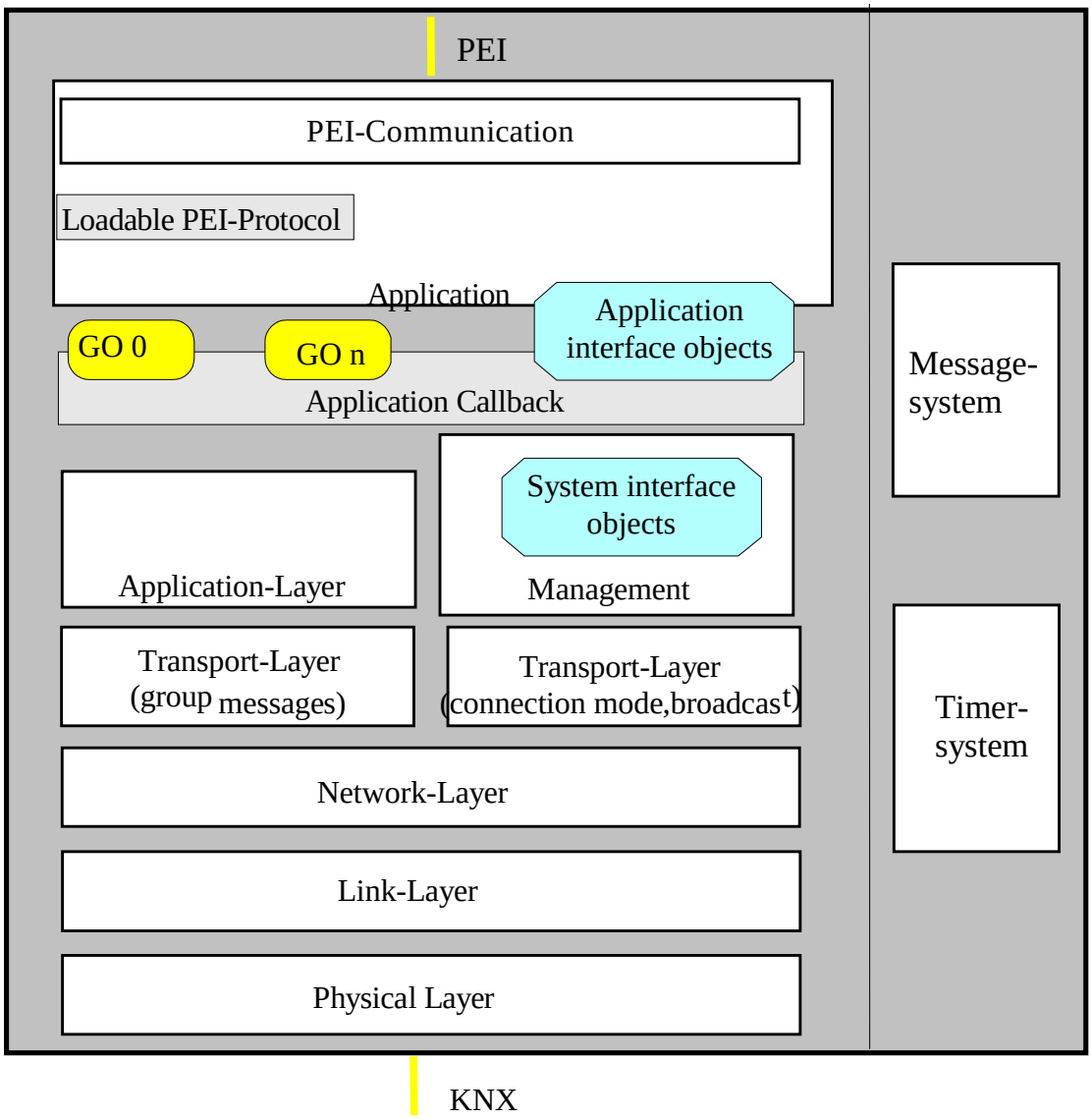

Figure 15: structure of system software

The TP1 BCU2 cycle time (time needed by the BCU to complete a cycle from the application program to the lower layers and back again) shall be between 5 to 7 ms.

5.1.2.4 Addressing

See Volume 6 ‘BCU2 Profile’.

5.1.2.5 PEI Type Detection

Shall be in accordance with Volume 3/6/2.

5.1.2.6 Modes of Operation

The TP1 BCU 2 shall have a number of special operation modes offered by the system software. After a reset the TP1 BCU2 shall be in normal operation mode. The special modes shall be reached over various operations during start-up or during normal operation.

5.1.2.6.1 Initialization Mode

The TP1 BCU2 shall switch to initialization mode if during start-up the following conditions are fulfilled:

- PEI-Type 20 is detected
- programming push-button is pressed

In initialization mode the following actions shall be carried out

- Individual Address is set to FFFFh
- Address table length is set to 1
- Option register is set to FFh.
- All load state machines are set to UNLOAD
- the whole memory from $ 470 up to $ 4FF is cleared
- the pointer to serial protocol is set to 0000h
- the Interface-Object pointer is set to 0000h
- the Interface-Object count is set to 00h
- all keys are set to FFFFFFFFh
- all default memory control blocks are re-installed.

5.1.2.6.2 Busmonitor Mode

In busmonitor mode all messages from the Link layer shall be directly sent to the PEI. The BCU shall be completely inactive on the bus. Each ACK, NACK and BUSY on the bus shall generate a message to the PEI. The busmonitor mode shall be entered via the PEI switch service or by writing to the system state.

5.1.2.6.3 User program start Mode

The User program shall be started if the loadctrl is loaded, runctrl is ready, if the EEPROM and the group-object- and association-table-data are ok (i.e. the corresponding error-flags are not set (low active)) and the requested PEI-type is equal to the physical PEI-type.

The User save routine shall be called when the save signal is set and the user program is running. After the save routine a reset shall be forced.

If one of the above conditions is not fulfilled, the user-program shall be stopped.

If later on, the above conditions are fulfilled, the user-program may be restarted.

Start or restart of the user-program shall always cause that

- the user-initialization-routine is called
- the User-RAM is cleared
- the user-routine is periodically called.

Each time before the user-routine is called, the registers (RegB - RegN) shall be cleared.

The Maximum time to stay in user program may not exceed 10 ms.

5.1.2.6.4 User Program Start-up Inhibit Mode

It shall be possible to enter the program start-up inhibit mode by pressing the program button briefly during start up. The user runctrl shall then be set to “terminate” and the user shall not start.

5.1.2.7 Initialization

The initialization of the TP1 BCU2 shall include the following steps:

1. wait until save signal is released
2. initialize hardware
3. initialize system software
4. wait until bus is free
5. read PEI type
6. check for Initialization mode
7. check programming-push-button for user inhibit mode
8. wait for start-up
9. check checksums
10. start scheduler

5.1.2.8 Scheduler

The following tasks shall be cyclically called by the scheduler:

- all communication tasks (LL, NL, TL, AL, MG)
- check routines (stack, system state)
- check for user start

5.1.2.9 Check-Routines

When loaded with a BCU2 style application, all available MCBs (memory control blocks) may have their own checksum byte. If this is the case, on start-up the MCBs shall be checked on respective validity and compared to the appropriate checksum byte. In the case of an error, the corresponding runtime-error-flag shall be set and the relevant load state machine set to error.

When loaded with a BCU1 style application, the TP1 BCU2 shall act as specified in the corresponding clause of the TP1 BCU1 chapter and the clause ‘Compatibility’ in this chapter.

It shall be possible to read the runtime-error-flags via the bus. As the error flags are in case of TP1 BCU2 part of the loading mechanism, it shall be avoided to reset them (see section compatibility).

Some other internal check routines may cause the halting of the user program of the TP1 BCU2 in the case of error

a) Pointer to RAM-flags-table in group-objects-table points to User RAM-area.

Some other internal check routines may cause a restart of the TP1 BCU2 in the case of error

b) stack overflow

c) system state parity error

5.1.2.10 Watchdog

The Watchdog-System shall be automatically started by the system-software. Taking into account the time that may be consumed by interrupt-routines, e.g. bit receive, the watchdog shall be triggered by the system software approximately every 40 ms.

5.1.2.11 ADC Channels

The ADC channel 4 shall be used to detect the hardware PEI type.

The TP1 BCU2 shall allow a coarse estimation of the bus current via the AD-converter channel 1 and the AdcRead-service. The value read shall be converted to a voltage value by using the following formula: Voltage = ADC_Value*0.15V

5.1.2.12 Memory Map

5.1.2.12.1 General

Owing to the used micro-controller, page 0, i.e. all addresses below $ 100, shall be bit-manipulable. The TP1 BCU 2 shall have a second user RAM area, called UserRAM2. The start address of the address table shall be located at $ 116. At $ 117 the own Individual address shall be stored. Pointer located in BCU 1 at $ 111 to $ 115 (i.e. pointer to the association, pointer to the user program initialization, pointer to the user program and pointer to the user program save routine) shall be located in the TP1 BCU 2 EEPROM System Area (not accessible by direct memory access). $ 118 to $ 46F may be used for the rest of the address table, the association table, the user program and the user initialization and save routines.

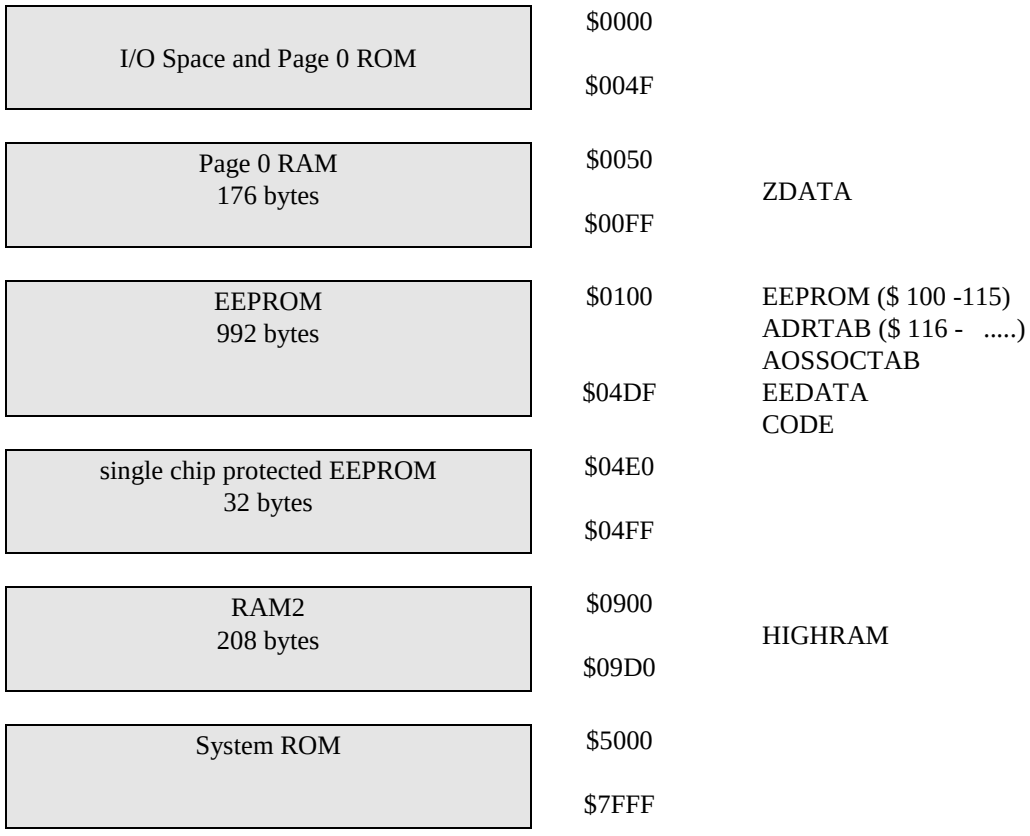

Figure 16: Memory Map

5.1.2.12.2 I/O-Space and Page 0 ROM

This memory area shall never be accessed by direct memory access.

memory address	name	length	comment		
$0000-$001F	I/O-space 1	32	I/O registers		
$0020-$002F	Page 0 ROM	16	$0020	OR_TAB	0000 0001 0000 0010 0000 0100 0000 1000 0001 0000 0010 0000 0100 0000 1000 0000
			$0028	AND_TAB	1111 1110 1111 1101 1111 1011 1111 0111 1110 1111 1101 1111 1011 1111 0111 1111
$0030-$003F	I/O space 2	16	I/O registers		
$0040-$004F	Page 0 ROM	16	reserved for system software		
			$004E	Mask type	($00 = TP-BCU)
			$004F	Mask-Version	($21 = 2.1)

Figure 17: I/O-Space and Page 0 ROM

5.1.2.12.3 Page 0 RAM

memory address	name	length	Comment	Segment
$0050-$005C	Reg B..N	13	RAM: registers reserved for the internal application (i.e. user program)	
$005D-$005F	iReg B..D	3	RAM: registers reserved for the internal application (i.e. user program) in Interrupt-Mode only	
$0060	System status	1	reserved for API 1 – for detailed description see corresponding clause in TP1 BCU1 chapter	
$0061-$00BC	System	90	reserved for system software	
$00BD-$00C3	PEI_Buff	7	reserved for serial protocol on PEI type 10 (only when FT 1.2 communication is used), 12, 16	
$00C4	PEI_Interface	1	reserved for Serial Protocol (if used)	
$00C5	PEI_Info	1	reserved for Serial Protocol (if used)	
$00C6-$00DF	UserRAM	26	RAM area for group object data and internal user application data	ZDATA ($BD-$DF)

$00E0-$00FF	stack	32	reserved for system software stack; also: internal application software stack $ E0 : stack overflow marker	

Figure 18: Page 0 ROM

It shall be possible for the application programmer to use all variables called "register" as temporary RAM-storage.

However, some of these variables may also be used by ROM-functions and for parameter passing or as temporary variables (See function descriptions !)

5.1.2.12.4 RAM 2

memory address	name	length	Comment	Segment
$0900-$0971	MS_Buffer	114	reserved for Message system	
$0972-$0989	UserRAM2	24	RAM area for group object data and internal user application data	HIGHRAM
$098A-$09A2	PEI_RecBuf	25	reserved for Serial Protocol / User	
$09A3-$09BB	PEI_SndBuf	25	reserved for Serial Protocol / User	

Figure 19: RAM2

The Memory areas PEI_Buff, PEI_RecBuf, PEI_SndBuf shall be reserved if PEI-Type 10 (only when using FT 1.2), 12, 16 is used.

If PEI-Type 10 is used with an own Serial Protocol, it shall be possible to use these memory areas.

If PEI-Type 14 is used, the RAM locations 0C8h – 0CDh (PEI_Buff) shall not be available for the Application Program. The RAM locations 0972h –09BBh (UserRAM2, PEI_RecBuf, PEI_SndBuf) shall be available for the Application Program.

5.1.2.12.5 EEPROM

memory address	name	length	comment	Segment
$0100	OptionReg	1	option register always FFh	EEPROM
$0101-$0102	ManData	2	reserved for Manufacturer	
$0103-$0107	ApplicationID	5	In BCU 2 Accessible viaSegmentControlRecord	
$0108	CheckLim	1	reserved for API 1	
$0109	PEI_Type	1	In BCU 2 Accessible via SegmentControlRecord	
$010A	SyncRate	1	baud rate used for serial synchronous PEI types12 and 14	
$010B	PortCDDR	1	Port C direction bit setting PEI type 10, 17	
$010C	PortADDR	1	port A direction bit setting (checked by ETS!)	

$010D	RunError	1	firmware run time error flags	
$010E	RouteCnt	1	routing count constant	
$010F	MxRstCnt	1	INAK retransmit limit BUSY retransmit limit	
$0110	ConfigDes	1	configuration descriptor	
$0111	AssocTabPtr	1		
$0112	CommsTabPtr	1		
$0113	UsrInitPtr	1		
$0114	UsrPrgPtr	1		
$0115	UsrSavPtr	1		
$0116-$046F	UsrEEPROM	858	User EEPROM Start with address table	Segment ADRTAB ASSOCTAB EEDATA CODE
$0470-$04DF	System	112	reserved for system	

Figure 20: EEPROM

Option Reg[6]

The Option register shall always be set to FFh. The BCU2 system software shall always invert values written to or read from the option register. Bits 0 and 1 of this byte are used by the processor hardware.

Bit	7-2	1	0
Meaning	shall be 1	EEPROT 0 = EEPROM $ 300- $ 4DF protected 1 = EEPROM $ 300- $ 4DFh writable	COPDIS 0 = disable watchdog 1 = enable watchdog

Figure 21: OptionReg

Block	Range	Description
1	$ 100 .. $ 2FF (512 byte)	not protected at all
2	$ 300 .. $ 4DF (480 byte)	selectable via options register
3	$ 4E0 .. $ 4FF (32 byte)	only programmable/ erasable in boot loader/ test mode

5.1.2.12.5.1 ManData

$ 101/$ 102	Product Manufacturer	see explanation below
$ 103/$ 104	AP Manufacturer	see explanation below

[6] values written to the option register shall always be inverted.

$ 105/$ 106	internal application software device type	readable via the property 13 of interface application object
$ 107	internal application software version	

Product Manufacturer

This can be any manufacturer code.

This address location shall coincide with Prop. 0CH, object type 00H (Device-Object).

The manufacturer of the product shall write his Manufacturer's code here before selling the product.

This value will be read by ETS!

This value shall never be overwritten by ETS!

AP Manufacturer

This can be any manufacturer code.

This address location shall be part of the property 0DH (Application Version), Object type 03H (Application Program)

The ETS will write here the manufacturer of the Application Program.

The manufacturer codes as on $101 – $102 respectively $103 – $104 shall be identical.

To assess whether some particular application from the ETS DB may be downloaded, ETS will read by means of an A_PropertyValue_Read Object Index[7] 00H Prop. 0CH (this is stored at $101 – $102) and will compare it with the Application Manufacturer code of the application to be downloaded (ETS database). The download will be refused if these values are not identical[8].

During download, the address location $101-$102 shall not be changed. The address locations $103-$104 will be overwritten with the Application Manufacturer code of the application to be downloaded (ETS database). The ETS will write this data via the segment ctrl record (Application Program Object).

5.1.2.12.5.2 CheckLim

Description see relevant chapter of TP1 BCU1 and clause compatibility in this chapter. This feature shall only be active in case of a loaded BCU1 style application.

5.1.2.12.5.3 PEI type

Description see relevant chapter of TP1 BCU1.

5.1.2.12.5.4 SyncRate

This EEPROM address shall contain the baud rate used for the serial synchronous PEI.

The crystal-frequency of the TP1 BCU2 shall be 4.9152 MHz.

5.1.2.12.5.5 PortCDDR, PortADDR, RunError, RouteCnt, MxRstCnt

Description see relevant chapter of TP1 BCU1.

Note for PortCDDR: this value shall be used only if the PEI-Type 17 or 10 is used and the EEPROM is assumed to be ok.

[7] The ETS assumes that the Device Object (Object type = 00H) has the Object Index 00H.

[8] Except when the code belongs to a pool of compatible manufacturer codes : in this case, both codes may differ if they both belong to the pool.

Attention : A warning shall be given to application programmers that the use of this feature may be a potential source of incompatibility, of which the use shall therefore be avoided!

5.1.2.12.5.6 ConfigDes

Description see relevant chapter of TP1 BCU1 and clause compatibility in this chapter.

5.1.2.12.5.7 UsrInitPtr, UsrPrgPtr, UsrSavPtr

As the TP1 BCU 2 shall have 1 Kbytes EEPROM, pointers of 1 byte will be insufficient to point to values in memory exceeding 1FFh. Consequently, the system software shall copy the maximum value (10 bit corresponding to 1K EEPROM) of each pointer to extended (invisible) 16 bit pointers. This means:

- the memory locations 113 to 115 h shall never be accessed directly by the TP1BCU 2 system software to start the different user routines.
- the value of the pointer to the telegram rate limitation shall be located at $ 113. The actual value is located at 100 + pointer – 1.
- the value of the pointer to the user timer description block shall be located at $ 114. The actual value is located at 100 + pointer – 1.

5.1.2.12.5.8 AdrTab

This byte shall mark the start of the group address table. The Address Table shall comply with the realization type 1 (in compatibility mode) and type 2 (in normal mode) of Volume 3/5/1.

5.1.2.12.5.9 AssocTabPtr

This byte shall contain the pointer to the association table. The Association Table shall comply with the realization type 1 (in compatibility mode) and type 2 (in normal mode) of Volume 3/5/1.

5.1.2.12.5.10 CommsTabPtr

This byte shall contain the pointer to the group object table. The Group Object Table shall comply with the realization type 1 (in compatibility mode) and type 2 (in normal mode) of Volume 3/5/1.

5.1.2.12.6 Single chip protected EEPROM

memory address	name	length	comment
$04E0-$04FF	Protected EEPROM	32	Reserved for system software

5.1.2.12.7 System ROM

memory address	name	length	comment
$5000-$7E7F	System ROM	11904	Reserved for system software
$7E80-$7FDF	Boot loader ROM	352	
$7FE0-$7FEF	Boot loader vectors	16	
$7FF0-$7FFF	System ROM vectors	16	

5.1.2.13 Resources[9]

Note : the maximum number of elements in an array supported by the TP1 BCU2 system software shall be limited to 255 elements.

5.1.2.13.1 Device Object

Property Name	Property ID	Type	Max number of elements	access Read/write
Object ID	PID_OBJECT_TYPE	PDT_UNSIGNED_INT	1	PRIVILEGE_NO / read only
Device Control	PID_DEVICE_CONTROL	PDT_GENERIC1	1	PRIVILEGE_NO / PRIVILEGE_ CONFIGURATION
Service Control	PID_SERVICE_CONTROL	PDT_UNSIGNED_INT	1	PRIVILEGE_NO / PRIVILEGE_ CONFIGURATION
Firmware Revision	PID_FIRMWARE_REVISION	PDT_UNSIGNED_CHAR	1	PRIVILEGE_CONFI-GURATION / read only
Serial Number	PID_SERIAL_NUMBER	PDT_GENERIC6	1	PRIVILEGE_CONFIGU-RATION / read only
Manufacturer ID	PID_MANUFACTURER_ID	PDT_UNSIGNED_INT	1	PRIVILEGE_CONFIGU-RATION / read only
OrderInfo	PID_ORDER_INFO	PDT_CHAR_BLOCK	1	PRIVILEGE_CONFIGU-RATION / read only
PEI-Type	PID_PEI_TYPE	PDT_UNSIGNED_CHAR	1	PRIVILEGE_CONFIGU-RATION / read only
pollgroup settings	PID_POLL_GROUP_SETTIN GS	PDT_POLL_GROUP_SET TINGS	1	PRIVILEGE_NO / PRIVILEGE_CONFIGU-RATION

[9] clause to be transferred to Volume 3/5/1

PortAddr	PID_PORTADDR	PDT_UNSIGNED_CHAR	1	PRIVILEGE_NO / PRIVILEGE_CONFIGU-RATION
Manufacturer Data	PID_MAN_DATA	PDT_GENERIC_04	1	PRIVILEGE_NO/read only

Figure 22: Device Object

PID_OBJECT_TYPE Object Type of the Device Object (00h)

PID_DEVICE_CONTROL Temporary Control field for the Device

Bit	Meaning
00	User stopped
01	A Frame with the own Individual Address was received
02	Verify Mode On
03-07	Reserved

PID_SERVICE_CONTROL Permanent Control field for the Device

Bit	Meaning	Remark
00	User stopped SrvInfo Enable	(1=enable ; 0=disable)
01	Receive Own PhysAddr SrvInfo Enable	(1=enable ; 0=disable)
02	IndividualAddressWriteEnable	(1=enable ; 0=disable)
03-07	Reserved	
08	User layer-Services on EMI Disable	(1=disable ; 0=enable)
09	Link layer-Services on EMI Disable	(1=disable ; 0=enable)
10	Network layer-Services on EMI Disable	(1=disable ; 0=enable)
11	Transport layer Group-Services on EMI Disable	(1=disable ; 0=enable)
12	Switch Service-Services on EMI Disable	(1=disable ; 0=enable)
13	Transport layer Connection Oriented-Services on EMI Disable	(1=disable ; 0=enable)
14	Application layer-Services on EMI Disable	(1=disable ; 0=enable)
15	Management-Services on EMI Disable	(1=disable ; 0=enable)

PID_FIRMWARE_REVISION Revision number of the Firmware starting with 00h

PID_SERIAL_NUMBER Serial number (unsigned char[6])

PID_MANUFCTURER_ID Code assigned by KNX Association to each member or licensee

PID_ORDER_INFO Manufacturer specific Order ID

PID_PEI_TYPE Actual connected PEI-TYPE

PID_POLL_GROUP_SETTINGS 2 byte Polling Group 1Byte Slot number

PollingGroupAddress		reserved		
		Enable		PollingSlotNo
A A A A A A A A A	A A A A A A A A A	E	0 0 0	S S S S

PID_PORTADDR Direction Register for Port A (0=input; 1=output)

PID_MAN_DATA Manufacturer specific data of the device (e.g. production data)

5.1.2.13.2 Address table Object

Property Name	Property ID	Type	Max number of elements	access Read/write
Object ID	PID_OBJECT_TYPE	PDT_UNSIGNED_INT	1	PRIVILEGE_NO / read only
Load Control	PID_LOAD_STATE_CONTROL	PDT_CONTROL	1	PRIVILEGE_NO / PRIVILEGE_SERVICE
Addresstable Pointer	PID_TABLE_REFERENCE	PDT_UNSIGNED_INT	1	PRIVILEGE_NO / read only

Figure 23: Address table Object

PID_OBJECT_TYPE Object type of the Address table (01h)

PID_LOAD_STATE_CONTROL for further Information see Load/State machines

PID_TABLE_REFERENCE Pointer to address table - This pointer points implicitly to the address table

5.1.2.13.3 Association table Object

Property Name	Property ID	Type	Max number of elements	access Read/write
Object ID	PID_OBJECT_TYPE	PDT_UNSIGNED_INT	1	PRIVILEGE_NO / read only
Load Control	PID_LOAD_STATE_CONTROL	PDT_CONTROL	1	PRIVILEGE_NO / PRIVILEGE_SERVICE
Associationtable Pointer	PID_TABLE_REFERENCE	PDT_UNSIGNED_INT	1	PRIVILEGE_NO / read only

Figure 24: Association Table Object

PID_OBJECT_TYPE Object type of the Association table (02h)

PID_LOAD_STATE_CONTROL for further Information see Load/State machines

PID_TABLE_REFERENCE Pointer to memory space of association table. This pointer points implicitly to the BCU 1 association table.

5.1.2.13.4 Application Program Object

Property Name	Property ID	Type	Max number of elements	access Read / Write
Object ID	PID_OBJECT_TYPE	PDT_UNSIGNED_INT	1	PRIVILEGE_NO / read only
Load Control	PID_LOAD_STATE_CONTROL	PDT_CONTROL	1	PRIVILEGE_NO / PRIVILEGE_ SERVICE
Run Control	PID_RUN_STATE_CONTROL	PDT_CONTROL	1	PRIVILEGE_NO / PRIVILEGE_CONFI-GURATION
PEI-Type Required	PID_PEI_TYPE	PDT_UNSIGNED_ CHAR	1	PRIVILEGE_NO / PRIVILEGE_CONFI-GURATION
Application Version	PID_APPLICATION_VER	PDT_SMALL_CHAR_ BLOCK	1	PRIVILEGE_NO / PRIVILEGE_CONFI-GURATION
Group object Table reference	PID_TABLE_REFERENCE	PDT_UNSIGNED_INT	1	PRIVILEGE_NO / read only

Figure 25: Application Program Object

PID_OBJECT_TYPE	Object type of the Program Object (03h)
PID_LOAD_STATE_CONTROL	for further Information see Load/State machines
PID_RUN_STATE_CONTROL	for further Information see Run/State machines
PID_PEI_TYPE	required PEI-Type for User
PID_TABLE_REFERENCE	When using standard callback this pointer shall point to the group object table.

5.1.2.13.5 Application Interface Objects

In order to be able to use application interface Objects, the Interface Object Pointer and Interface Object Count shall be set according Volume 3/5/2 Management Procedures.

Interface-Object Pointer Absolute address of Application interface Objects-Table

Interface-Object Count Number of Application Interface Objects

The Application Interface Object-Table shall contain a 16-Bit pointer for each Application Interface Object

High Byte	Low byte
pointer to Application Interface Object 1	
pointer to Application Interface Object 2	
pointer to Application Interface Object n	

Each Application Interface Object shall at least have an Object type Property.

Accesslevel	Read (4 Bit)	Write (4 Bit)			
Property Count	N (1 Byte)				
Property 1	PID_OBJECT_TYPE(1 Byte)		PUINT (1 Byte)	Obj-ID High (1 Byte)	Obj-ID Low (1 Byte)
Property 2	PID_USER 1		PropCtrl 1	...	...
.....	....				
Property N	PID_USER N		PropCtrl N	...	...

5.1.2.13.6 Attributes of application interface object Properties

All properties shall be accessible via the general Management Procedures.

Byte 0	Byte 1				Byte 2		Byte 3
Property-ID	Write (1Bit)	Array (1Bit)	Ptr (1Bit)	Type (5Bit)	Funct. (1Bit)	Pointer High (7Bit)	Pointer low (8Bit)
(1Byte)	Property Control				Variable (16-Bit)		

Descriptions of Attributes:

Property-ID 1 Byte (0-255) Property ID of User Property

Write 1 Bit
- 0 Property is read-only
- 1 Property is read and write enable

Array 1 Bit
- 0 No Array
- 1 Pointer to Array

Pointer 1 Bit
- 0 No Pointer. Value direct in property
- 1 Pointer to Value

Type 5 Bit (See Volume 3/7/3)

Variable/Pointer 16/15 Bit
- Pointer 0 / Array 0 Property Value
- Pointer 1 / Array 0 Pointer to Value / Function
- Pointer 0 / Array 1 Not defined
- Pointer 1 / Array 1 Pointer to Array structure

funct. 1 Bit
- 0 Variable / Array Pointer
- 1 pointer to User function with following parameters
  - Input
    - carry
      - 0 Property Read
      - 1 Property Write
    - X Pointer to message buffer
  - Output
    - carry
      - 0 Read OK
      - 1 Read not OK
    - X Pointer to message buffer
    - acca Message len

A user function shall consist of two parts: one to handle a write (called when carry = 1) and one to handle a read (carry = 0). In case of a PropertyWrite the user function is called twice, once with carry =1 (write) and once with carry = 0 (read) in order to generate the PropertyResponse.

Array structure (in EEPROM)

byte 0	byte 1	byte 2
Max Element Count	Pointer to Array variable	

Array variable (in EEPROM or RAM)

byte 0	byte 1 - m			
Actual Element Count	Element 1	Element 2	...	Element n

5.1.2.14 TP1 BCU2 Access protection

The TP1 BCU 2 shall support 4 access levels. Access level 0, which has the most privileges, shall be required to access the Load State Machine for the Application. The level required for accessing the Load State Machines for Address- and Association-Table shall be access level 1.

Access level 3 shall be reserved for free access without authorization.

Level	Usage
0	Loading Application Programs
1	Loading Address Table, Assoc. Table
2	Application Parameters
3	Reserved for Free Access

5.1.2.15 TP1 BCU2 Callback functionality

The TP1 BCU2 shall offer a possibility to deactivate via user programs the normal AL mechanisms to access group objects and/or interface object properties.

Warning: user programs developed for a TP1 BCU2 supporting the callback functionality will not run on a TP1 BCU2 not supporting this feature.

When the standard callback functionality (i.e. standard group object and/or application interface object property handling) is used, the TaskControl2 of the load state machine shall be written with value 5081h. In case of a user callback, the value shall be equal to the start address of the callback function. This callback function may however also call the standard callback. In the latter case, mutual influencing of user and standard callback functionality shall be carefully avoided.

In case the entire standard AL functionality for group objects and/or interface object properties is replaced by a user callback functionality, this will prevent any further handling of the device by the ETS software.

5.1.2.16 Loadable PEI handler

Next to the standard system PEI communication types (FT 1.2 and handshake per byte), the TP1 BCU2 shall offer the possibility to implement a user PEI Handler as an event handler receiving hardware and software events. If hardware events are received, it shall be possible to call the user handler as an interrupt. The events shall be exchanged via register A. The installation of a user PEI handler is described in Volume 3/5/2 Management procedures.

5.1.2.16.1 Reserved memory for system PEI-Handler (if used by application programmer)

5.1.2.17 PEI_RecBuf

Serial Receive buffer (25 byte) – updates shall be signalled by the system “Serial buffer post” flag

5.1.2.18 PEI_SndBuf

Serial send buffer (25 Byte) – transmission shall be started when system flag “Serial buffer Semaphore” is set.

5.1.2.19 PEI_Interface

Bit	**Meaning**
0	Reserved

1	Serial buffer semaphore: this semaphore shall be used for sending a frame through the PEI. The BCU2 system shall set this bit to indicate to the PEI program that it shall send the contents of PEI_SndBuf to the PEI. The PEI program shall reset this bit to indicate to the system that it is ready to send the next buffer
2-6	Reserved
7	Serial buffer post flag: this flag shall be used for the reception of a frame through the PEI. The PEI program shall set this bit to indicate to the BCU2 system that it has filled PEI_RcvBuf with data originating from the PEI. The system shall reset this bit when it has copied the contents of the PEI_RcvBuf to a message buffer and has sent it to another system module.

5.1.2.20 PEI_Info

Additional data event dependent

If no serial protocol and no PEI Handler is used (i.e. no use of PEI-Type 10,12,14,16 or 20), then it shall be possible that PEI_Info, PEI_Buffer, PEI_RecBuf and PEI_SndBuf are defined by the user.

5.1.2.20.1 Hardware Events

Hardware events shall be derived from hardware interrupts and shall be numbered with fixed codes, their interpretation may depend on the active protocol software.

hardware event description	constant name	event code	PEI_Info
SCI flag RDRF set receive with even parity	rc_even	03h	received data
SCI flag RDRF set receive with odd / none parity	rc_odd/ rc_none	13h	received data
SCI flag TC set	tc	14h	
SCI flag TDRE set	tdre	05h	
SCI flag IDLE	sci_idle	06h	
SPI flag SPIF set	spif	08h	received data
Output Compare A	OCA	10h	Timer Value
Output Compare B	OCB	20h	Timer Value
Input Capture A	ICA	30h	Timer Value
Input Capture B	ICB	40h	Timer Value

Figure 26: hardware events

If one of these hardware events occurs, they shall always be reported to the PEI protocol software. Such hardware events may however be prevented by the user according to the processor's data book.

5.1.2.20.2 Software Events

Software event description	constant name	event specification
Message received	PM_MESSAGE	F1
Cyclic call (call each scheduler cycle one times)	PM_CYCLE	F2
Initialization	PM_INIT	F3
TimerEvents	Userdefined with TS_Seti	

Figure 27: software events

5.1.2.21 API

The Application Programmer's Interface available in the TP1 BCU2 shall comply with the one specified in Chapter 3/6/1.

5.1.2.22 Physical and External Interface

The Physical External Interface shall comply with the requirements of Volume 3/6/2. The External Message Interface shall comply with the EMI1 of Volume 3/6/3.

5.2 Electrical Safety

No.	Requirements	M
1	The requirements of Volume 4 Part 1 shall be complied with.	M/S

5.3 Environmental conditions

No.	Requirements	M
1	BCU2 TP1 shall comply with the requirements of Volume 4 Part 1 clause 2.1.2to 1.3.3	M/S

5.4 EMC

No.	Requirements	M
1	BCU2 TP1 shall comply with the requirements of Volume 4 Part 1 clause 2.3	M/S

5.5 Mechanical, Dimensions, Constructional Features

5.5.1 General

A press button switch (toggle function) shall be provided on any mechanical design of the TP1 BCU2 to select or deselect the parameterization mode of the TP1 BCU2.

Parameterization mode shall allow setting or reading via broadcast the individual address of the TP1 BCU2. This shall be possible without the knowledge of the Individual address of the TP1 BCU2.

In parameterization mode, the LED on the TP1 BCU2 shall be switched on, else off.

5.5.2 Flush mounted BCU2 TP1

This chapter describes the standardised mechanical features of flush-mounted TP1 BCU2 with all necessary information. The dimensions are given in Figure 28.

This type of TP1 BCU 2 is intended to be inserted into wall-boxes including means for fixing the TP1 BCU2 (frame) by means of screws. The flush mounted TP1 BCU 2 is delivered as a completely assembled unit without mounting frame.

The position of the programming LED and button in the underneath figure is recommended.

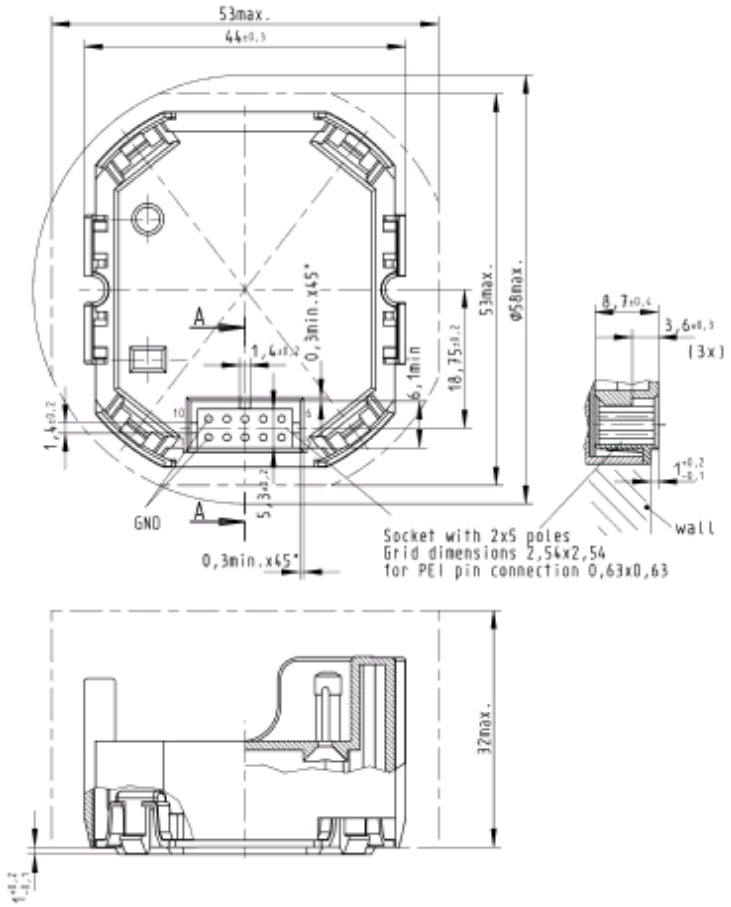

Figure 28: Flush-mounted TP1 BCU2

Warning:

The standardisation of the dimensions of the flush mounted BCU does not - due to the permitted tolerances - in all cases ensure that the PEI of the to be mounted application module fits on the BCU! Therefore it is highly recommended for the design of a BCU to select the dimensions according typical values.

5.5.3 Surface and DIN rail mounted TP1 BCU2

The TP1 BCU 2 as surface and DIN rail mounted components are not standardized.

5.6 Electrical Features

5.6.1 Bus Interface characteristics

Characteristics	Symbol	Min	Max	Typical	Unit	Remarks
Operating Voltage	V_{bus}	21	30		V	

Current consumption	I_{bus}			4	mA	V_{bus} = 30 V
Reset conditions	Vcc	< 4,6			V	Reset generated by transceiver
BCU-Buffer-Time		≥ 60 ms			ms	PEI-Load: 50 mW

Figure 29: TP1 BCU2 Bus interface characteristics

5.6.2 PEI characteristics

Characteristics	Symbol	Min	Max	Unit	Remarks
Supply Output Voltage +5V	VCC	4.7	5.3	V	max. 10mA
Supply Output Voltage +20V	VDD	18	22	V	Load < 5mA VBUS - VDD > 1,5V Vbus = 21V ... 30V
Current Limitation	IDD		5	mA	
Data Output Voltage Port A, Port B, Port C	VOL VOH	 VCC-0,8	0.4	V	Iload = 1,6mA Iload = 0,8mA
Data Output Voltage TDO, SCLK, PLMB	VOL VOH	 VCC-0,8	0.4	V	Iload = 1,6mA Iload = 1,6mA
Data Output Voltage Reset	VOL		1,0	V	Iload = 1,6mA
Data Input Voltage Port A, Port B, Port C Reset, RDI	VIL VIH	0 0.7 VCC	0.2 VCC VCC	V	
Analog Input Voltage Range	VAIL VAIH	0	 VCC	V	
I/O Ports Three-State-Leakage	IOZ		±10	µA	
Input Capacitance	CIN		50	pF	

Figure 30: TP1 BCU2 PEI electrical characteristics

5.6.3 Timing PEI

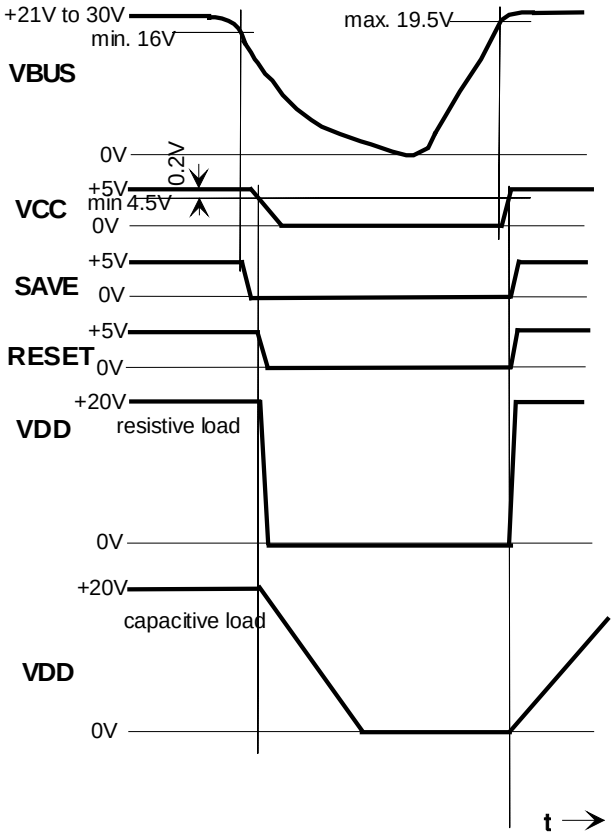

Figure 31: TP1 BCU2 PEI timing

5.7 Testing

The following test shall be carried out to show compliance:

- communication testing : according Volume 8 System Conformance testing
- Environmental conditions: according Volume 4 Part 3
- EMC: according Volume 4 Part 2
- Mechanical and electrical properties according to this Handbook clause

5.8 Functional Safety

Under Consideration

5.9 Interfaces, Connectors

The TP1 BCU2 shall be connected with the following standard connectors:

- For the connection to the bus: type 5.1 according Part 9/1.
- For the connection to the application module: type 4.1 as described in Part 9/1 with 10 pole PEI

5.10 Marking

The BCU2 TP1 can be marked by the manufacturer at his discretion.

5.11 Installation

Not applicable

5.12 Symbols

to be completed.

Page left blank intentionally