PHYTEC

CANmodul-592

Hardware Manual

Edition January 1999

A product of a PHYTEC Technology Holding company

In this manual are descriptions for copyrighted products that are not explicitly indicated as such. The absence of the trademark (™) and copyright (©) symbols does not imply that a product is not protected. Additionally, registered patents and trademarks are similarly not expressly indicated in this manual.

The information in this document has been carefully checked and is believed to be entirely reliable. However, PHYTEC Meßtechnik GmbH assumes no responsibility for any inaccuracies. PHYTEC Meßtechnik GmbH neither gives any guarantee nor accepts any liability whatsoever for consequential damages resulting from the use of this manual or its associated product. PHYTEC Meßtechnik GmbH reserves the right to alter the information contained herein without prior notification and accepts no responsibility for any damages which might result.

Additionally, PHYTEC Meßtechnik GmbH offers no guarantee nor accepts any liability for damages arising from the improper usage or improper installation of the hardware or software. PHYTEC Meßtechnik GmbH further reserves the right to alter the layout and/or design of the hardware without prior notification and accepts no liability for doing so.

	EUROPE	NORTH AMERICA
Address:	PHYTEC Technologie Holding AG Robert-Koch-Str. 39 D-55129 Mainz GERMANY	PHYTEC America LLC 255 Ericksen Avenue NE Bainbridge Island, WA 98110 USA
Ordering Information:	+49 (800) 0749832 order@phytec.de	1 (800) 278-9913 info@phytec.com
Technical Support:	+49 (6131) 9221-31 support@phytec.de	1 (800) 278-9913 support@phytec.com
Fax:	+49 (6131) 9221-33	1 (206) 780-9135
Web Site:	http://www.phytec.de	http://www.phytec.com

1st Edition January 1999

Index of Figures and Tables

Preface

This CANmodul-592 User's Manual describes the board's design and functions. Precise specifications for the 80C592 microcontrollers can be found in the enclosed microcontroller Data-Sheet/User's Manual. If software is included please also refer to additional documentation for this software.

In this hardware manual and in the attached schematics, low active signals are denoted by a "/" in front of the signal name (i.e.: /RD). A "0" indicates a logic-zero or low-level signal, while a "1" represents a logic-one or high-level signal.

Declaration regarding EMV-Conformity of the PHYTEC CANmodul-592

CE

PHYTEC Single Board Computers (henceforth products) are designed for installation in electrical appliances or as dedicated Evaluation Boards (i.e.: for use as a test and prototype platform for hardware/software development) in laboratory environments.

Attention:
PHYTEC products lacking protective enclosures are subject to damage by ESD and, hence, may only be unpacked, handled or operated in environments in which sufficient precautionary measures have been taken in respect to ESD-dangers. It is also necessary that only appropriately trained personnel (such as electricians and engineers) handle and/or operate these products. Moreover, PHYTEC products should not be operated without protection circuitry if connections to the product's pin header rows are longer than 3 m.

PHYTEC products fulfill the norms of the EMVG-statute only in accordance to the descriptions and rules of usage indicated in this hardware manual (particularly in respect to the pin header row connectors, power connector and serial interface to a host-PC).

Implementation of PHYTEC products into target devices, as well as user modifications and extensions of PHYTEC products, is subject to renewed establishment of conformity to, and certification of, EMV-Statutes. Only after doing so the devices are allowed to be put into circulation.

The CANmodul-592 is one of a series of PHYTEC nano/micro/miniMODULs which can be fitted with different controllers and, hence, offers various functions and configurations. PHYTEC supports all common 8- and 16-bit controllers in two ways:

(1) as the basis for Starter Kits in which user-designed hardware can be implemented on a wrap-field around the controller and

(2) as insert-ready, fully functional micro- and miniMODULS which can be embedded directly into the user's peripheral hardware design.

PHYTEC's microcontroller modules allow engineers to shorten development horizons, reduce design costs and speed project concepts from design to market. Please contact PHYTEC for additional information:

	EUROPE	NORTH AMERICA
Address:	PHYTEC Technologie Holding AG Robert-Koch-Str. 39 D-55129 Mainz GERMANY	PHYTEC America LLC 255 Ericksen Avenue NE Bainbridge Island, WA 98110 USA
Web Site:	http://www.phytec.de	http://www.phytec.com
e-mail:	info@phytec.de	info@phytec.com
Voice:	+49 (6131) 9221-0	+1 (800) 278-9913
Fax:	+49 (6131) 9221-33	+1 (206) 780-9135

1 Introduction to the CANmodul-592

The CANmodul-592 is a credit card-sized Single Board Computer based on the 80C592 microcontrollers from VALVO/PHILIPS. Its universal design allows its insertion in a wide range of embedded applications. All controller signals and ports extend from the controller to standard-width (2.54 mm) pin rows aligning three edges of the board, allowing it to be plugged into a target application like a “big chip”. The CANmodul-592 is nearly pin-compatible to the miniMODUL-535. Hence it can serve as a drop-in replacement for the latter. The canMODUL-592 is pre-configured and equipped with all necessary connectors required for immediate start-up *(refer to Figure 2*).

Precise specifications for the specific controller fitted on the board can be found in the enclosed microcontroller User’s Manual. The descriptions in this manual are based on the 80C592 controllers. No description of compatible microcontroller derivative functions are included, as such functions are not relevant for basic functioning of the CANmodul-592.

The CANmodul-592 offers the following features:

- SBC in credit card-size dimensions (55 x 85 mm) achieved through advanced SMD technology
- fitted with the VALVO/PHILIPS P80C592 8051-compatible controllers
- improved interference safety through multi-layer technology
- controller signals and ports extend to standard-width (2.54 mm.) pins aligning board edges, allowing the board to be plugged into any target application like a “big chip”
- requires a single low power supply 5 V/typ. < 100
- 128 (to 512) kByte Flash on-board (PLCC)
- on-board Flash programming
- no dedicated Flash programming voltage required through use of 5 V Flash devices
- 32 (to 160) kByte RAM on-board (SMD)
- 32 kByte EEPROM (SMD) can also be accommodated on the board
- flexible software-configured address decoding through complex logic device
- bank latches for Flash and RAM integrated in address decoder
- RS-232 interface
- CAN-Interface
- 3 free Chip Select signals for easy connection external peripherals
- operates within a standard range of 0 to 70 degrees C°.

1.1 Block Diagram

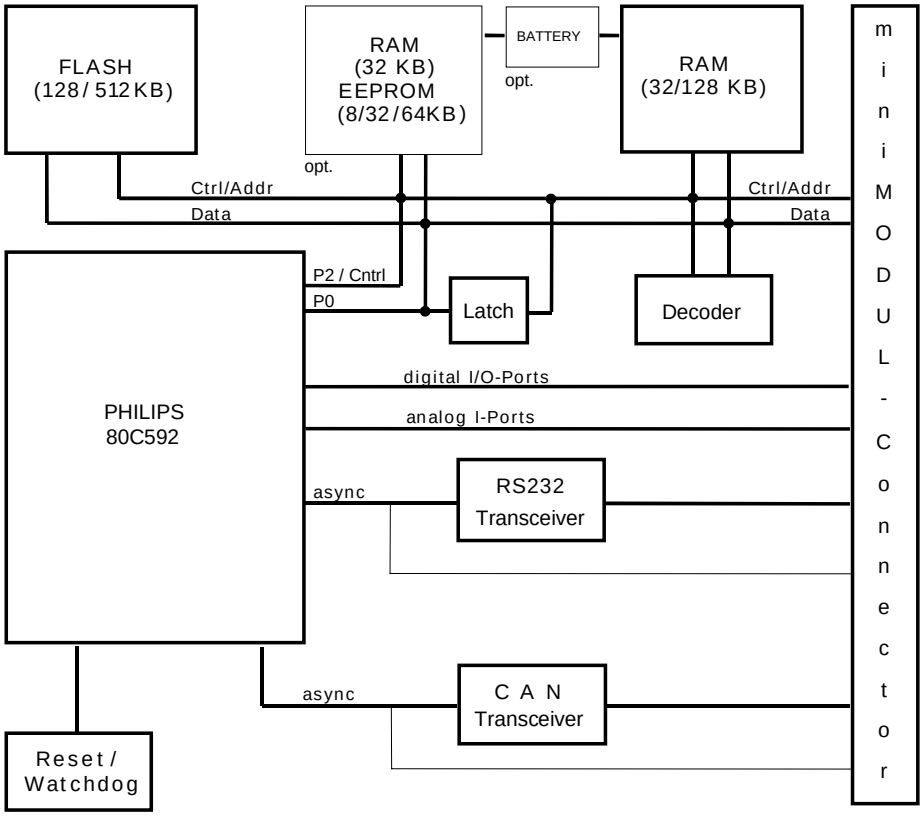

Figure 1: Block Diagram

1.2 CANmodul-592 Overview

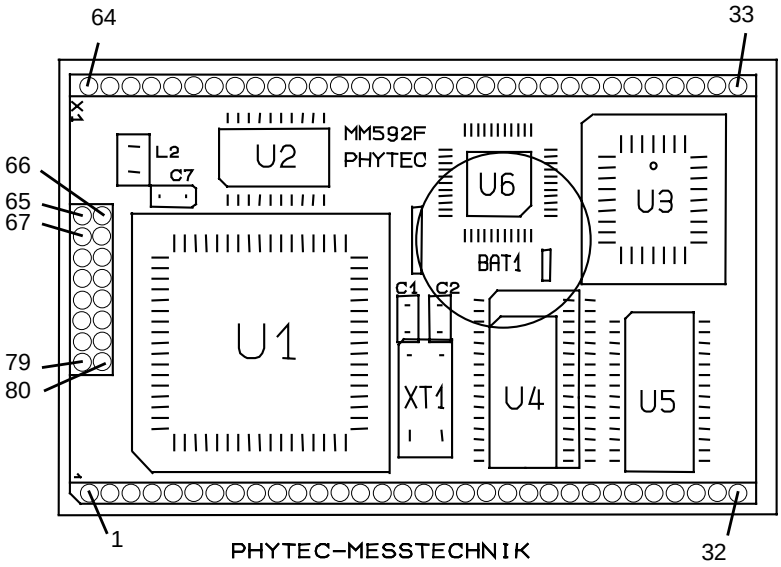

Figure 2: CANmodul-592 Overview (Component Side)

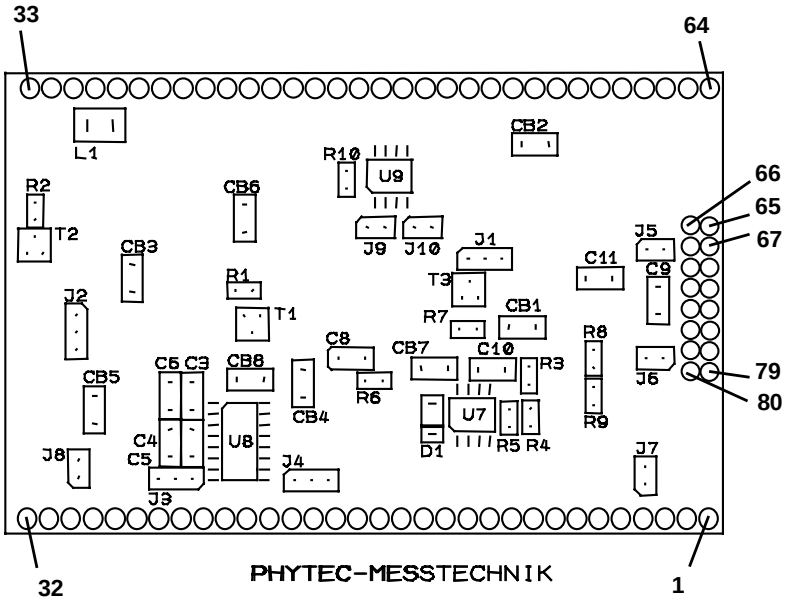

Figure 3: CANmodul-592 Overview (Soldering Side)

2 Pin Description

Please note that all module connections are not to exceed their expressed maximum voltage or current. Maximum signal input values are indicated in the corresponding controller manuals/data sheets. As damage from improper connections varies according to use and application, it is the user's responsibility to take appropriate safety measures to ensure that the module connections are protected from overloading through connected peripherals.

As *Figure 4* indicates, all controller signals extend to standard-width (2.54 mm) pin rows lining three sides the board (referred to as miniMODUL-Connector). This allows the board to be plugged into any target application like a "big chip".

Many of the controller port pins accessible at the edges of the board have been assigned alternate functions that can be activated via software.

Table 1 provides an overview of the pinout of the miniMODUL-Connector, as well as hints about additional functions of some of the port pins. *For further details please refer to the Data Sheet of the P8xC592*.

Attention:
The CANmodul-592 has been reengineered for Flash technology in an manner ensuring the highest possible compatibility to earlier non-Flash fitted versions of the CANmodul-592. However some differences in pinout to earlier versions, as described in *Appendices: Revisions for CANmodul-592*, were unavoidable.

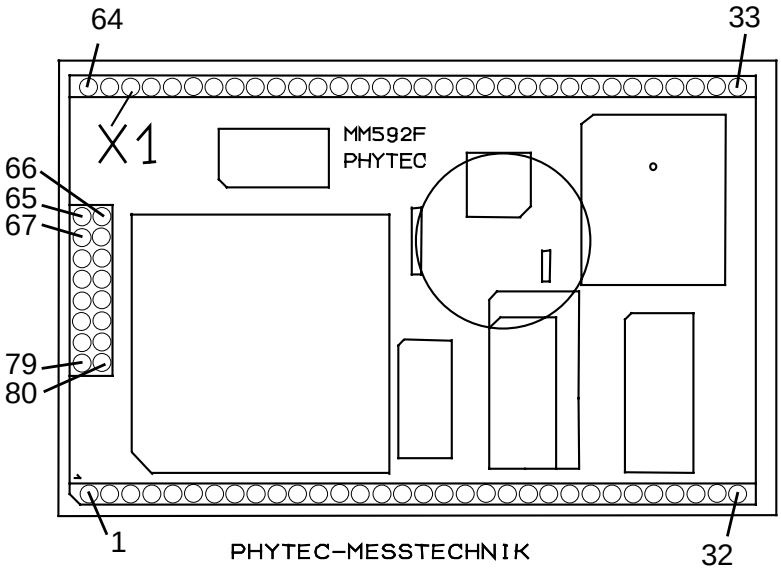

Figure 4: CANmodul-592 Pinout

PIN #	Connection	Comments
1	NC / WDI	optional Watchdog-Input
2...9	P4.0...P4.7	Port 4
10	STADC	ext. Triggerinput of the A/D-converter
11	/PWM0	output 0 for puls-width modulation
12	/PWM1	output 1 for puls-width modulation
13	/EW	watchdog-enable input
14	ALE	addresslatch-enable output
15	RES	reset output port of the controller
16	A13	address bus A13 (High-Byte)
17	A12	address bus A12 (High-Byte)
18...25	P1.0...P1.7	Port 1
26	P3.0 RXD	Port 3.0 or RXD (RS-232)
27	P3.1 TXD	Port 3.1 or TXD (RS-232)
28	P3.2 /INT0	Port 3.2 or /INT0
29	P3.3 /INT1	Port 3.3 or /INT1
30	P3.4 T0	Port 3.4 or Timer 0
31	/RES	reset-input of the module
32	NC / VBAT	optional external battery buffer via Jumper J8
33	VCC	power supply + 5 VDC
34	VPD	battery voltage output

PIN #	Connection	Comments
35	P3.5 T1	Port 3.5 or Timer 1
36...39	A11...A8	address bus (High-Byte)
40...47	A7...A0	address bus (Low-Byte)
48	CANH CRX0	CANH or CRX0 via Jumper J10
49	CANL CRX1	CANL or CRX1 via Jumper J9
50	/PSEN	program-store-enable-output
51	P3.6 /WR	Port 3.6 or /WR-Signal
52	P3.7 /RD	Port 3.7 or /RD-Signal
53	/CS3	pre-decoded Chip-Select-Signal #3
54	/CS2	pre-decoded Chip-Select-Signal #2
55	/CS1	pre-decoded Chip-Select-Signal #1
56...63	D7...D0	data bus (controller port 0)
64	GND	ground 0V.
65	AREF	analog reference voltage + 5 VDC
66,68,70,72, 74,76,78,80	AN7...AN0	8 Analog Inputs
67	AVDD	supply voltage analog circuitry + 5 V=
79	AREF-	analog reference ground 0V
69,71,73,75, 77	AGND	Analog Ground (GND) 0 V

Table 1: Pinout of the miniMODUL-Connector

3 Jumpers

For configuration purposes, the CANmodul-592 has 10 soldering jumpers, some of which have been configured prior to delivery. *Figure 5* illustrates the numbering of the jumper-pads, while *Figure 6* indicates the location of the jumpers on the CANmodul-592. All soldering jumpers on the CANmodul-592 are located on its bottom side.

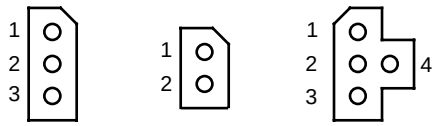

Figure 5: Numbering of the Jumper Pads

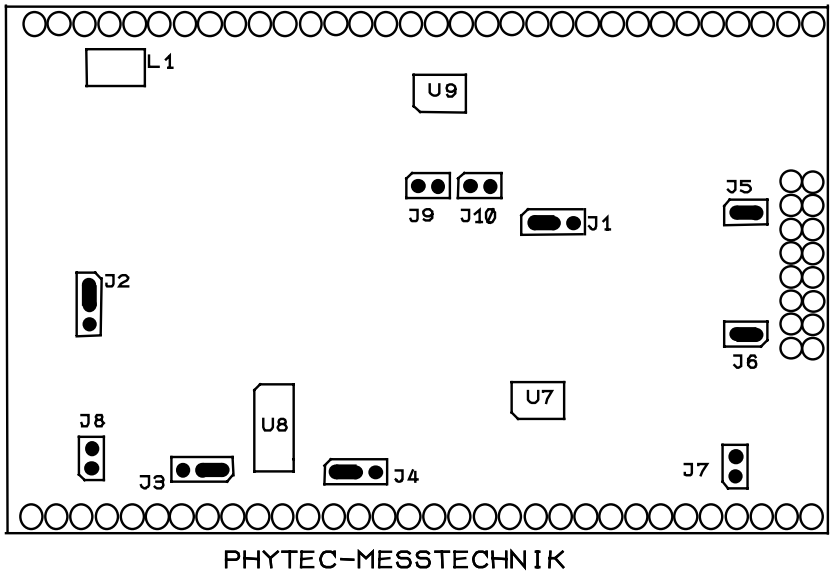

Figure 6: Jumper Location (bottom view)

The jumpers (J = solderable jumper) have the following functions:

	Default-Setting		**Alternate-Setting**	
J1	(1+2)	external ROM/ Flash active	(2+3)	internal ROM/Flash active
J2	(1+2)	no battery buffer	(2+3)	U5 buffered by battery (only for RAM-devices)
J3, J4	(1+2) (1+2)	pins 26 and 27 carry RS-232 signals from the on-board transceiver	(2+3)	pins 26 and 27 carry (2+3) TTL signals from the controller's serial port
J5, J6	(closed) (closed)	V_{AREF} and V_{AGND} derived from supply voltage	(open) (open)	V_{AREF} and V_{AGND} derived from externall voltage source via miniMODUL-Connector
J7	(open)	Watchdog input not available	(closed)	Watchdog input connected to pin 1 of the miniMODUL-Connector
J8	(open)	pin 32 of the miniMODUL-Connector is open	(closed)	VBAT input connected to pin 32 of the miniMODUL-Connector
J9 J10	(open) (open)	pin 49 carries CANL and pin 48 CANH of the on-board transceiver	(closed) (closed)	pin 49 carries CRX1 and pin 48 CRX0 of the controller's CAN-interface

Table 2: Jumper Settings

3.1 Special Features

Jumpers J1, J5, J6, J7, and J8 are used to activate the special features of the particular controller fitted on the module.

- **Execution out of Internal or External Program Memory J1**

  At the time of delivery Jumper J1 is pre-connected at pads 1+2. This default configuration means that the program stored in the external program memory is executed after a hardware-RESET. In order to allow the execution of a specific controller's internal program memory, the pads 2+3 on jumper J1 must be connected.

  The following configurations are possible:

Code-access	**J1**
External Program Memory	1+2*
Internal Program Memory	2+3

  * = Default-Setting

- **Supply Voltage of the A/D Converter J5 and J6**

  A common supply for the digital and analog circuitry of the controller can be configured with jumpers J5 and J6. To use this option jumpers J5 (analog-Vcc) and J6 (analog-GND) must be closed. The connection for the analog and digital supply voltage should be enabled at a central point in analog circuitry in order to minimize interference. In order to use the CAN-interface of the CANmodul-592 jumpers J5 and J6 must be closed.

Analog supply	**J5**	**J6**
via digital supply	closed*	closed*
external supply	open	open

  * = Default-Setting

- **Watchdog Input J7**

  Closing Jumper J7 renders the Watchdog input from the RESET device at U7 available at pin 1 on the CANmodul-592. This should be done to maintain compatibility to older versions of the CANmodul-592.

CANmodul-592 pin 1	**J7**
Watchdog Input	closed
No Watchdog Input	open*

* = Default-Setting

- **Connecting an External Battery: J8**

  Closing jumper J8 enables attachment of an external battery at pin 32 of the CANmodul-592 for purposes of buffering the RAM contents in the event of a disconnected power supply. When using an external battery, the optional on-board lithium battery must not be installed. Please *refer to section 6,* "*Battery Buffer*" – as well as to the description of Jumper J2 in s*ection 3.2* of this manual.

  Closing Jumper J8 is necessary to maintain compatibility with older versions of the CANmodul-592.

CANmodul-592 pin 32	**J8**
VBAT input connected to pin 32	closed
pin 32 open	open*

* = Default-Setting

3.2 Battery Buffer of U5 J2

Jumper J2 configures the power supply for the memory device installed at U5. If an EEPROM is fitted at U5, a Vcc supply is necessary (i.e., J2 should be closed at 1+2) to prevent discharge of the battery buffer. If U5 is populated with a RAM device, the power should be supplied via VPD (i.e., J2 should be closed at 2+3) in order to preserve data by means of the battery buffer in the absence of a power supply via Vcc.

Device type at U5	**J2**
EEPROM	1+2*
RAM	2+3

* = Default-Setting

Attention:
The battery device on the CANmodul-592 is not appropriate to supply an EEPROM if installed at U5. Therefore jumper J2 has to be closed at 1+2 in order to avoid fast discharge of the battery.

3.3 Serial Interface Configuration J3 and J4

With Jumpers J3 and J4, different serial interface signal levels and signal qualities can be configured at pins 26 and 27 of the CANmodul-592. These pins carry either the TTL-signals of the controller's serial interface or the signals of the on-board RS-232 transceiver. At the time of delivery the RS-232 interface is active by default.

The following signal levels and qualities can be configured:

Signal Quality	**J3**	**J4**
RS-232	1+2*	1+2*
TTL	2+3	2+3

* = Default-Setting

3.4 CAN Interface Configuration J9, J10

The CAN-interface of the 80C592 is available at port P1.6 and the special pins CRX0 and CRX1. These signals extend to the CAN-transceiver populating U9 (PCA82C250 or Si9200), which generates the signals CAN_H (at pin 48) and CAN_L (at pin 49). These signals can be directly connected to a CAN-bus using a dual-wire cable. If no CAN-transceiver populates U9, the controller signals CRX0 and CRX1 are accessable at pins 48 and 49 for use with an external transceiver.

For detailed descriptions of the CAN-interface please refer to appropriate controller User's Manual from PHILIPS, as well as accompanying CAN transceiver data sheet.

The following configurations are possible:

CAN-Transceiver	**J9**	**J10**
external CAN-Transceiver	closed	closed
internal CAN-Transceiver (U9 populated)	open*	open*

* = Default-Setting

4 Memory Models

The CANmodul-592 allows for flexible address decoding which can be adjusted by software to different memory-models. A Hardware-RESET activates a default memory configuration that is suitable for a variety of applications. However, this memory-model can be changed or adjusted at the beginning of a particular application.

Configuration of the memory is done within the address decoder by means of internal registers: two control registers, one address register and one mask register. All named registers are Write-Only-Registers with access to the XDATA-memory of the controller. There are two distinct address areas - selectable by means of the bit IO-SW in control register 1 - by which the registers can be accessed (refer to the description of the bit IO-SW below). Due to a lack of read-access, a copy of all register contents should be maintained within the application. Reserved bits may not be changed during the writing of the register; contents must remain at 0. A Hardware-RESET erases all registers while preserving the configuration of the default memory.

Attention:
In the event that you use the FlashTools – PHYTEC's proprietary firmware allowing convenient on-board Flash-programming - the address FA16 is preset at the start of your application software (refer to the section "*Control Register 1*" below). This is to be noted upon installation of the software copy of the register contents.

The following Figure displays the default memory model:

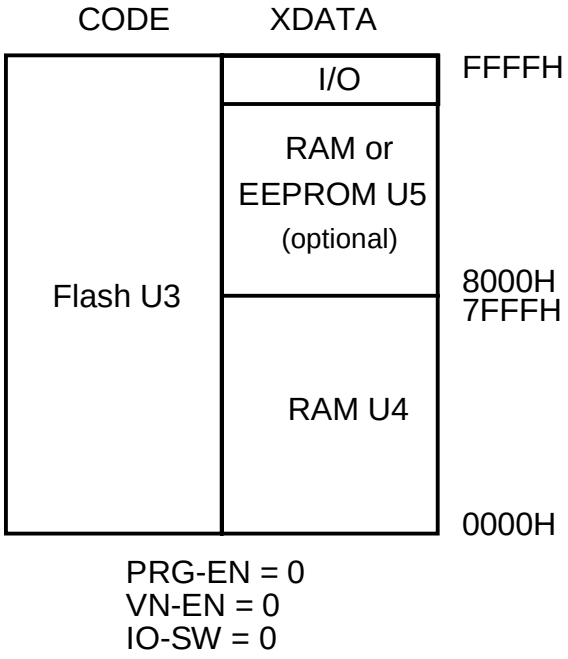

Figure 7: Default Memory Model after Hardware-Reset

It should be noted that the memory blocks U4 and U5 comprise separate 32 kByte memory areas in the XDATA address-area of the controller. The standard module houses a single 32 kByte RAM device at U4. In the event that a 128 kByte RAM device is installed at U4, then blocks of 32 kByte can be accessed and via bank-switching. In the event that memory devices do not populate U4 and U5, then there is no possible access to the corresponding XDATA memories. The corresponding current I/O-area is concentrated in an XDATA-address area in which there is no access to any existing RAM.

The following sections explain the registers of the address decoder for configuration of the memory:

4.1 Control Register 1

Control Register 1 (Address 7C00H / FC00H)							
Bit 7							Bit 0
PRG-EN	IO-SW	RAM-SW	VN-EN	FA18	FA17	FA16[1]	FA15

Bit invalid in programming-model (refer to PRG-EN)
Bit valid only in programming-model (refer to PRG-EN)

PRG-EN: Activates the special Flash-programming memory model (PRG-EN = 1). This configuration is used within FlashTools[2] for Flash-programming. On account of existing restrictions it is either of no or of restricted use in the user's application.
In this model, 32 kByte Flash memory located within the address range 8000H-FFFFH is accessible, as well as 32 kByte RAM within the range 0000H-7FFFH. The Flash memory can only be written in the XDATA-area and can only be read from the CODE-area. The RAM can be read from and written to in the XDATA-area. RAM can also be read from the CODE-area. The address line A15 of the Flash is derived from the Control Register 1 (Bit 0, FA15) only in the programming configuration. In the Runtime execution-configuration (PRG-EN = 0), the address line A15 of the controller leads directly to the Flash device.
The bit IO-SW is also relevant to the programming configuration; whereas the bit VN-EN is not relevant.

1: In the event that you use the FlashTools - a firmware allowing convenient on-board Flash-programming - it should be noted that the address FA16 will be preset at the start of your application software. This is to be noted upon installation of the software copy of the register contents.

2: PHYTEC firmware allowing convenient on-board Flash-programming. Upon delivery of the module, this firmware is already resident in the Flash device.

The following Figure illustrates the programming configuration (the I/O-field is not represented):

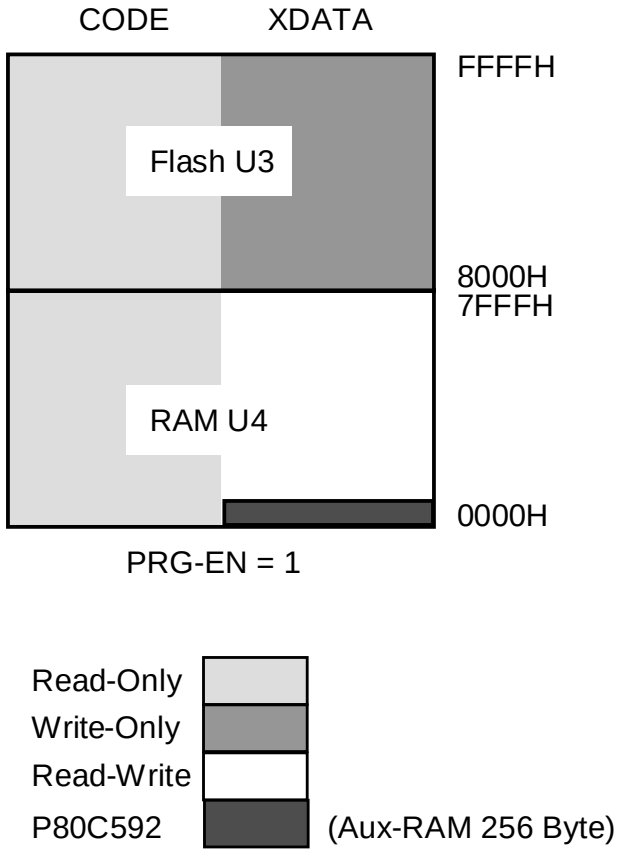

Figure 8: Memory Model for Flash-Programming

IO-SW: By means of this bit, the I/O-area of the module can be selectively mapped either to the upper or to the lower 32 kByte of the address space. After a Hardware-Reset (IO-SW = 0) the I/O-area is located in the address area from FC00H to FFFFH. Following setting of the IO-SW-bit, the I/O-area is located in the address area from 7C00H to 7FFFH.

This I/O-area generally consists of 4 blocks of 256 bytes. In three of these blocks the address decoder provides a pre-coded Chip-Select signal which simplifies the connection of peripheral hardware to the module.

These Chip Select signals are activated by XDATA-access (Read-Write access) to the corresponding address area. The fourth block is reserved for accessing the register internal to the decoder (Write-Only access). Hence, this block is not available for connection of peripheral hardware to the module.
The following diagram illustrates the partitioning of the I/O-area:

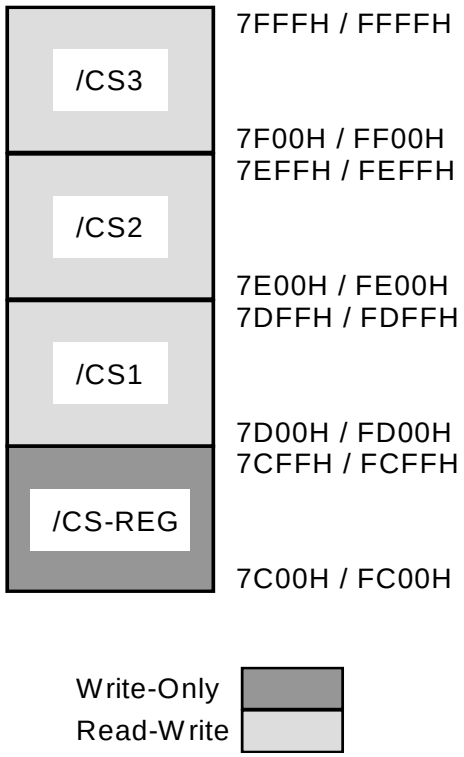

Figure 9: Partitioning of the I/O-Area

As shown above, /CS1 through /CS3 function as the available free Chip Select signals. The signal /CS-REG is solely a signal internal to the decoder, which is necessary in order to access the internal register, and is not available for external devices.

Connection of peripheral devices to /CS-REG should not take place under any circumstances in order to maintain the correct function of the FlashTools[1] for programming of the Flash. The internal register is to occupy only the address ranges 7C00H-7C03H and/or FC00H-FC03H. The rest of the /CS-REG block remains unused and is reserved for future expansion.

RAM-SW: This bit enables exchange of 32 kByte memory areas of the devices installed at U4 and U5. Following a Hardware-Reset (RAM-SW = 0) the RAM U4 is mirrored in the area from 0000H to 7FFFh and the RAM/EEPROM at U5 is addressable from 8000H to FFFH. After setting the RAM-SW bit, the RAM at U4 populates the area from 8000H-FFFFH. Likewise, the RAM/EEPROM at U5 populates the area from 0000H-7FFFH. In the corresponding I/O areas, there is no access to the memory devices.

VN-EN: This bit enables free selection of von-Neumann memory[2] within the address space of the controller. A Reset renders a Harvard[3]-Architecture available as the default configuration. Von Neumann memory is especially useful when programming code is to be downloaded and subsequently run during Runtime, as is the case with a Monitor program. The location of the optional von Neumann memory is defined through the address- and mask registers (see below).

1: PHYTEC firmware allowing convenient on-board Flash-programming. Upon delivery of the module, this firmware is already resident in the Flash device.

2: Memory area in which no difference is made between CODE- and XDATA-access. This means that both accesses use the same physical memory device, usually a RAM.

3: Memory area in which CODE and XDATA-accesses use physical different memory devices. CODE-access typically uses a ROM or Flash device, whereas XDATA-access uses a RAM.

Following a Hardware-Reset (VN-EN = 0) the settings in the address- and mask registers are not released, which means that no von Neumann-memory is available. After setting the bit (VN-EN = 1), the settings in the address- and mask registers are valid and incorporated in access addressing. This bit is only relevant in the Runtime-model (PRG-EN = 0). In the Programming-model (PRG=1) it is unimportant and ignored.

FA[18..15]: The module can be equipped with an optional 512 kByte Flash memory. As the controller's address space is limited to 64 kByte, the remainder of the Flash memory can only be accessed by means of bank memory switching.
In the Runtime-model (PRG-EN = 0), 64 kByte banks can be switched by controlling the high address lines A[18..16] for the Flash through software. For this purpose, register bits FA[18..16] of the address decoder provide a latch to which the desired higher addresses can be written.
Of particular note is the bit FA15, which is solely relevant in the programming-model (PRG-EN = 1). As only 32 kByte of Flash can be accessed in this model, it serves as address line A15 for the Flash memory. In the Runtime-model (PRG-EN = 0) with a 64 kByte Flash memory area, to contrast, the addres line A15 of the controller is attached directly to the Flash.
Bits FA[18..16] are dependent on the hardware configuration of the module and function as described above only if Flash devices of at least 512 kByte are installed on the board.

4.2 Control Register 2

Control Register 2 (Address 7C01H / FC01H)							
Bit 7							Bit 0
N/A[1]	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*	RA16	RA15

RA[16.. 15]: The module can optionally accommodate a 128 kByte RAM device at U4. As the address space at U4 is limited to 32 kByte in the XDATA area of the controller, the remainder of the RAM can only be accessed by means of bank switching.
Four memory banks of 32 kByte banks can be switched by setting the high address-lines A[16..15] through software. For this purpose, register bit RA[16..15] of the address decoder provides a latch to which the desired higher addresses can be written.
The function of this bit is dependent on the hardware configuration of the module and functions, as described above, only in connection with RAM devices of at least 128 kByte at U4.

[1]: N/A: Not Accessible

4.3 Address Register

The address register 7C02H / FC02H functions in conjunction with the mask register (see below) to define the von Neumann[1]- and Harvard[2]-memory in the controller's addressing area. By setting the bit VN-EN in control register 1, the values of the address and the mask register become valid for the definition of the von Neumann and the Harvard addressing space and are used for address decoding (*refer to Control Register 1*). The location of one or more Harvard areas can be configured with both registers. The remaining sections of the addressing area are configured as von-Neumann area in which RAM is accessible through XDATA as well as through CODE.

The mechanism through which the areas are differentiated is based on a comparison of the current address with a predefined address pattern of variable width. If the relevant bit positions of the addresses conform to one another, access occurs according to the Harvard-architecture. In the case of nonconformity, access occurs according to the von-Neumann-architecture.

Address Register (Address 7C02H / FC02H)							
Bit 7							Bit 0
HA15	HA14	HA13	HA12	HA11	HA10	*Res.*[3]	*Res.*

The address register holds the address pattern mentioned above. Each bit of the pattern is compared with the corresponding address line of the controller (HA15 with A15, ..., HA10 with A10). As address lines A15 .. A10 are used to define Harvard addressing space, only Harvard-fields of at least 1 kByte can be configured. Areas smaller than 1 kByte can not be configured.

1: Memory area in which no difference exists between CODE- and XDATA-access. This means that both accesses use the same physical memory device, usually a RAM.

2: Memory area in which CODE and XDATA-accesses use different physical memory devices. Usually CODE-access uses a ROM or Flash device, whereas XDATA-access uses a RAM.

3: Reserved bits are not to be changed, the default value (0) must remain.

4.4 Mask Register

The mask register (addresses 7C03H / FC03H) serves the masking of single bits in the address register (see above). Following a Hardware-Reset, all bits within the address register are relevant. By setting the individual bits in the mask register, all corresponding bits in the address register will no longer be subject to an address comparison.

Mask Register (Address 7C03H / FC03H)							
Bit 7							Bit 0
MA15	MA14	MA13	MA12	MA11	MA10	*Res.*[1]	*Res.*

Please note that in the case of a board populated with a single 32 kByte RAM, the memory area is mirrored within the controller's addressing area. On account of the insufficient utilization of A15 in this configuration, memory accesses to addresses higher than 8000H are reduced to accesses to the memory area from 0000H to 7FFFH. This should be taken into consideration when choosing the memory-model. Otherwise, function failure could result from overlapping access.

[1]: Reserved bits are not to be changed, the default value (0) must remain.

The following examples of different combinations of the address- and mask registers illustrate these functions (X = specific bit irrelevant):

Address-Reg.	**Mask-Reg.**	**Comments (only for VN-EN = 1)**
1XXXXX00b	011111**00**b	Harvard 8000H-FFFFH, von Neumann 0000H-7FFFH
0XXXXX00b	011111**00**b	Harvard 0000H-7FFFH, von Neumann 8000H-FFFFH
111111**00**b	000000**00**b	Harvard FC00H-FFFFH, von Neumann 0000H-FBFFH
010X00**00**b	000100**00**b	Harvard 4000H-43FFH and 5000H-53FFH, von Neumann 0000H-3FFFH, 4400H-4FFFH and 5400H-FFFFH
10000000b	00000000b	Harvard 8000H-83FFH, von Neumann 0000H-7FFFH and 8400H-FFFFH
10100X00b	000001**00**b	Harvard A000H-A7FFH, von Neumann 0000H-9FFFH and A800H-FFFFH

Reserved bits without function for address decoding (refer to description of the register)

X = irrelevant (on account of a bit set in the mask register)

The last example in the Table is further illustrated by the following Figure:

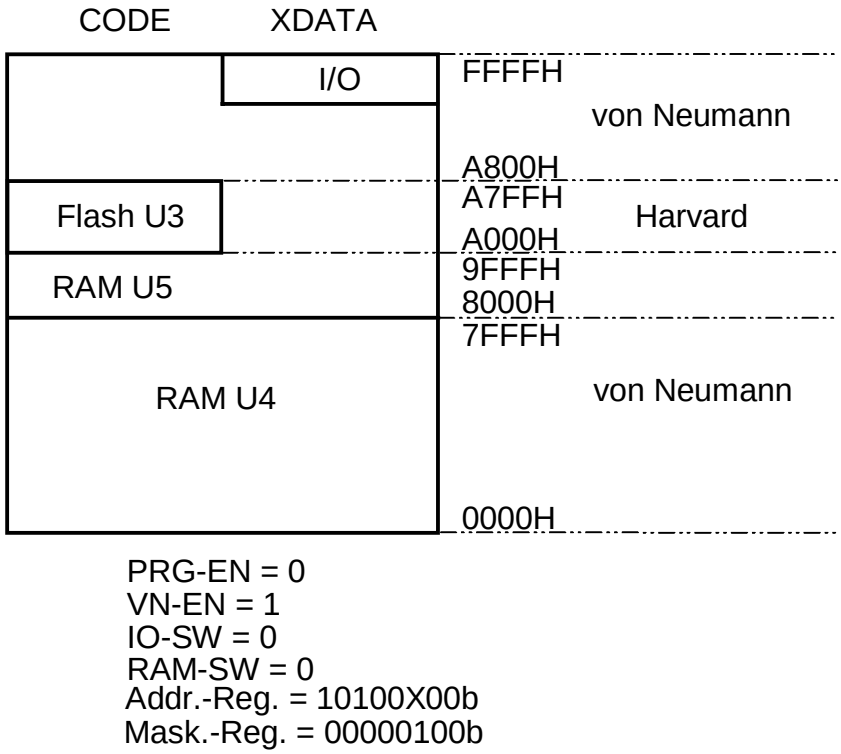

Figure 10: Example of a configurable Memory Model

5 Flash Memory

Flash is a highly functional means of storing non-volatile data. Having the CANmodul-592 equipped with a Flash device makes this modern technique available. The CANmodul-592 can house a Flash device of type 29F010 with two banks of 64 kByte each or of type 29F040 with 8 banks of 64 kByte each.

Use of Flash devices allows incorporation of on-board programming capability. The Flash devices are programmable with 5 V=. Consequently, no dedicated programming voltage is required. A firmware to programm the Flash device (the so-called FlashTools) is pre-installed in the first bank (bank 0) of the Flash device. Hence the total memory available is 64 kByte or 448 kByte (*refer to Figure 11*).

Attention:
Should this software be erased from the Flash device without having a back-up or an equivalent replacement, reprogramming is no longer possible!

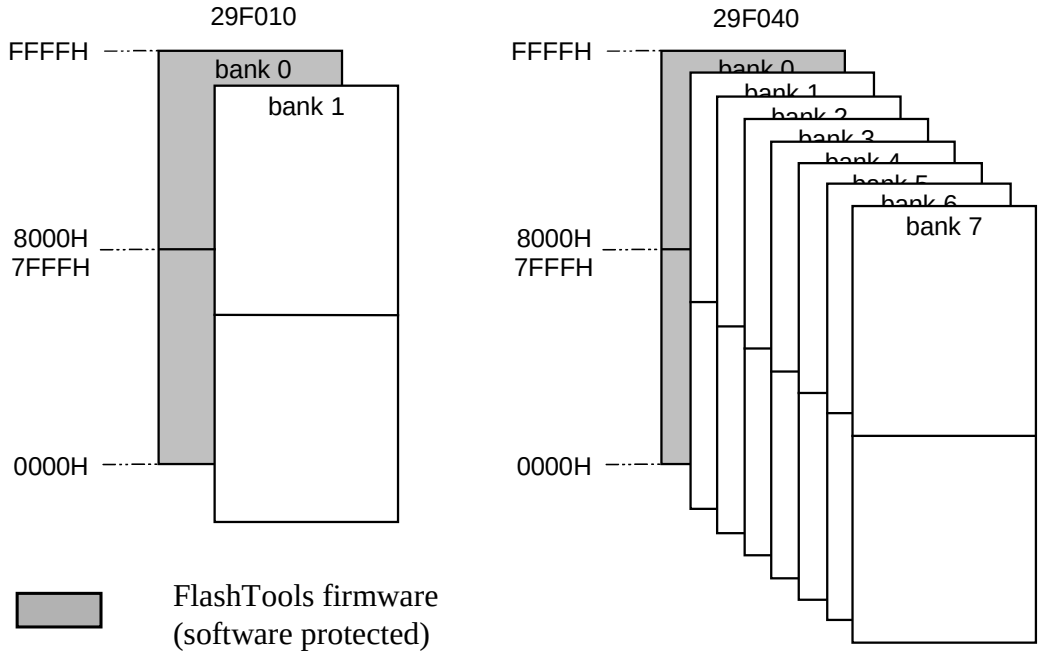

Figure 11: Memory Areas of the Flash Device

Please note that this firmware protects itself against any intentional or accidental erasure or copy-over. As the Flash device's hardware protection mechanism is not utilized, protection is limited to the software level. In the event that you might wish to download your own programming algorithms or tools into the Flash, please ensure that a programming tool remains in the Flash memory.

Use of a Flash device as the only code memory results in no or only a limited usability of the Flash memory as non-volatile memory for data. This is due to the internal structure of the Flash device as, during the Flash-internal programming process, the reading of data from Flash is not possible. Hence, for Flash programming, program execution must be transferred out of Flash (such as into von Neumann RAM). This usually equals the interruption of a "normal" program execution cycle.

As of the printing of this manual, Flash devices generally have a life expectancy of at least 100,000 Erase-/Program-cycles.

6 Battery Buffer

The battery that buffers the memory is not otherwise essential to the functioning of the CANmodul-592. However, this battery buffer embodies an economical and practical means of storing nonvolatile data.

The VBAT-input at pin 32 is intended to connect to an external battery. Optionally a battery can be installed at position BAT1 on the component side of the module. As of the pressing of this manual, a lithium battery is recommended for use with the module as it offers relatively high capacity at low discharge. In the event of a power failure at Vcc, the connected battery via VBAT will buffer the RAM memory blocks.

Attention:
The battery device on the CANmodul-592 is not appropriate to supply an EEPROM if installed at U5. Therefore jumper J2 has to be closed at 1+2 in order to avoid fast discharge of the battery.

Power consumption depends on the components used and memory size. This is typically < 1 µA. per 32 kByte RAM device installed on the CANmodul-592.

For reasons of operating safety, please be advised that despite a battery buffer, changes in the data content within the RAM can occur given disturbances. The battery buffer does not completely remove the danger of data destruction.

7 Technical Specifications

The physical dimensions of the CANmodul-592 are represented in *Figure 12*. The module's profile is ca. 10 mm thick, with a maximum component height of 2.5 mm on the back-side of the PCB and approximately 5.5 mm on the front-side. The board itself is approximately 1.5 mm thick.

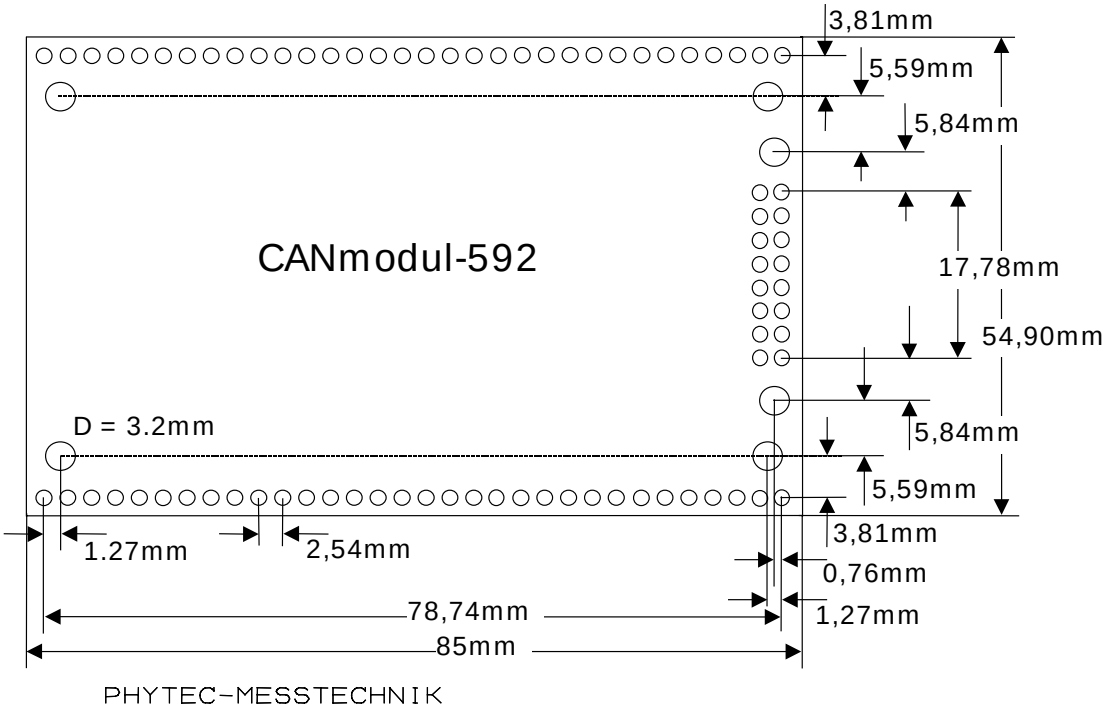

Figure 12: Mechanical Dimensions

Additional specifications:

- Dimensions: 54.9 x 85 mm., ±0,01mm
- Weight: approximately 44 g with 160 kByte RAM device, socketed Flash device and socket controller; ca. 32 g with standard 32 kByte RAM without socket
- Storage temperature: -40°C to +90°C
- Operating temperature: standard 0°C to +70°C, extended -40°C to +85°C condensed
- Operating voltage: 5 V ±5%, VBAT 3 V ±20 %
- Power consumption: maximum 140 mA, typ. 100 mA at 16 MHz oscillator frequency and 128 kByte RAM at +20°C
- Power consumption with battery buffer: 10 µA per RAM-device, typically 1 µA per RAM-device at +20°C

This specifications describe the standard configuration of the CANmodul-592 as of the pressing of this manual.

Please note that utilizing the battery buffer for the RAMs the storage temperature is only 0°C to +70°C.

8 Hints for Handling the Module

When changing controllers please ensure that all components remain free from intrusive damage. Any controller used on the module must be pin-compatible with the P8xC592, and all special hardware features must be compatible with the layout of the board.

Removal of the standard quartz or oscillator is not advisable given the compact nature of the module. Should this nonetheless be necessary, please ensure that the board, as well as surrounding components and sockets, remain undamaged during the unsoldering process. Overheating the board can cause the solder pads to loosen, rendering the module inoperable. Carefully heat neighboring connections in pairs. After a few alternations, components can be removed with the solder-iron tip. Alternatively, a hot air gun can be used to heat and loosen the bonds.

Appendices: Revisions for CANmodul-592

The CANmodul-592 has been updated for Flash memory technology. PHYTEC strives to maintain maximum compatibility between this new and preceding versions of the CANmodul-592. However, small differences in board configuration between Flash and EPROM versions were inevitable. These differences are explained in the Table below:

	CANmodul-592 (Old version: MM-400)	**CANmodul-592** (New version: MM-403)
Pin1	VCC	Vcc extends only to pin 33 in order to improve EMC features. The Watchdog input from U7 can be optionally connected with pin 33 via Jumper J7.
Pin32	GND	GND extends only to pin 64 in order to improve EMC features. The Vbat input can be optionally connected with pin 64 via Jumper J8 for an external battery connection.
U3	applicable to house an OTP/ EPROM (32Kx8/64Kx8) in (P)LCC-socket	applicable to house a Flash (types 29F010/ 29F040 with 128Kx8/ 512Kx8) or OTP (with 128Kx8) in PLCC socket.

Table 3: Revisions

Index

S

T

V

W

Document: CANmodul-592
Document Number: L-376e_1, January 1999

How would you improve this manual?

Did you find any mistakes in this manual? page

Submitted by:
Customer number: _____

Name: _____

Company: _____

Address: _____

_____

Return to:

PHYTEC Technologie Holding AG
Postfach 100403
D-55135 Mainz, Germany
Fax : +49 (6131) 9221-33

Published by PHYTEC

Ordering No. L-376e_1
Printed in Germany