phyCORE-TC1775

Hardware Manual

Edition April 2003

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 203 Parfitt Way SW, Suite G100 Bainbridge Island, WA 98110 USA
Ordering Information:	+49 (800) 0749832 order@phytec.de	1 (800) 278-9913 sales@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: April 2003

Index of Figures

Index of Tables

Preface

This phyCORE-TC1775 Hardware Manual describes the board's design and functions. Precise specifications for Infineon's TC1775 Tricore microcontroller series controller 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 of Electro Magnetic Conformance of the PHYTEC phyCORE-TC1775

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.

Caution:
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, technicians 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 European Union's Directive for Electro Magnetic Conformance 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, Electro Magnetic Directives. Users should ensure conformance following any modifications to the products as well as implementation of the products into target systems.

The phyCORE-TC1775 is one of a series of PHYTEC Single Board Computers that can be populated with different controllers and, hence, offers various functions and configurations. PHYTEC supports all common 8- and 16-bit as well as selected 32-bit controllers in two ways:

(1) as the basis for Rapid Development Kits which serve as a reference and evaluation platform

(2) as insert-ready, fully functional micro-, mini- and phyCORE OEM modules, 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.

1 Introduction

The phyCORE-TC1775 belongs to PHYTEC's phyCORE Single Board Computer module family. The phyCORE SBCs represent the continuous development of PHYTEC Single Board Computer technology. Like its mini-, micro- and nanoMODUL predecessors, the phyCORE boards integrate all core elements of a microcontroller system on a subminiature board and are designed in a manner that ensures their easy expansion and embedding in peripheral hardware developments.

As independent research indicates that approximately 70 % of all EMI (Electro Magnetic Interference) problems stem from insufficient supply voltage grounding of electronic components in high frequency environments the phyCORE board design features an increased pin package. The increased pin package allows dedication of approximately 20 % of all pin header connectors on the phyCORE boards to Ground. This improves EMI and EMC characteristics and makes it easier to design complex applications meeting EMI and EMC guidelines using phyCORE boards even in high noise environments.

phyCORE boards achieve their small size through modern SMD technology and multi-layer design. In accordance with the complexity of the module, 0402-packaged SMD components and laser-drilled Microvias are used on the boards, providing phyCORE users with access to this cutting edge miniaturization technology for integration into their own design.

The phyCORE-TC1775 is a subminiature (72 x 57 mm) insert-ready Single Board Computer populated with Infineon's TC1775 Tricore microcontroller. Its universal design enables its insertion in a wide range of embedded applications. All controller signals and ports extend from the controller to high-density pitch (0.635 mm) connectors aligning two sides of the board, allowing it to be plugged like a "big chip" into a target application.

Precise specifications for the controller populating the board can be found in the applicable controller User's Manual or Data Sheet. The descriptions in this manual are based on the Infineon TC1775 Tricore microcontroller. No description of compatible microcontroller derivative functions is included, as such functions are not relevant for the basic functioning of the phyCORE-TC1775.

The phyCORE-TC1775 offers the following features:

- subminiature SBC in phyCORE dimensions 72 x 57 mm with two 100-pin high-density (0.635 mm) Molex connectors, enabling it to be plugged like a “big chip” into target application
- Processor: Infineon Tricore TC1775, 16 MHz external clock
- **Internal Components of the phyCORE-TC1775:**
  - High performance 32-bit TriCORE CPU
  - Independent Peripheral Control Processor (PCP)
  - 72 kByte SRAM
  - two synchronous serial interfaces
  - Dual UART
  - Dual CAN 2.0B
  - Serial Data Link Modul (SDLM) according to SAE Class B J1850 specification
  - General Purpose Timer Array (GPTA)
  - General Purpose Timer Unit (GPTU) with three 32-bit timers
  - Dual 16-channel 12-bit A/D converter
  - Multi-purpose I/O signals
- **Memory Configuration**[1]**:**
  - SRAM: 2 MByte maximum
  - Flash-ROM: 4 MByte Burst Mode Flash maximum; 8 MByte Flash maximum
  - I^2C memory: 4 kByte EEPROM (up to 32 kByte) (optional I^2C FRAM (512 Byte), or I^2C SRAM (256 Byte) can be used)
- I^2C Real-Time Clock with calendar and alarm functions
- Ethernet Controller CS8900A 10 Mbit TP
- Dual UART: RS-232 transceiver for both channels (RxD/TxD); TTL level can be configured
- Dual CAN port: 82C251CAN transceiver 82C251 for both channels; TTL level can be configured
- JTAG/Debug port
- Available in standard- (0...+70° C) temperature range

1: Please contact PHYTEC for more information about additional modul configurations.

1.1 Block Diagram

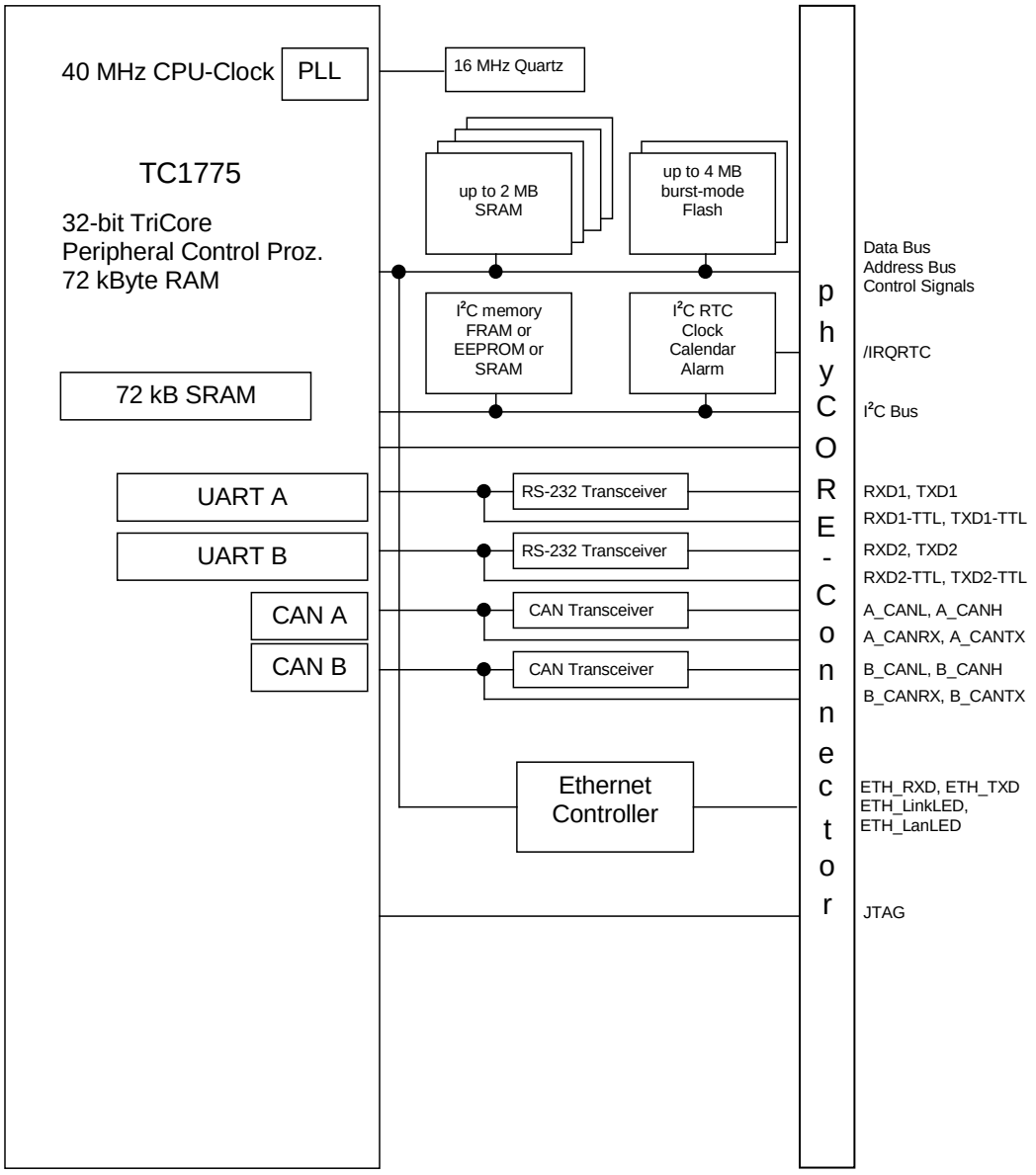


Figure 1: Block Diagram phyCORE-TC1775

1.2 View of the phyCORE-TC1775

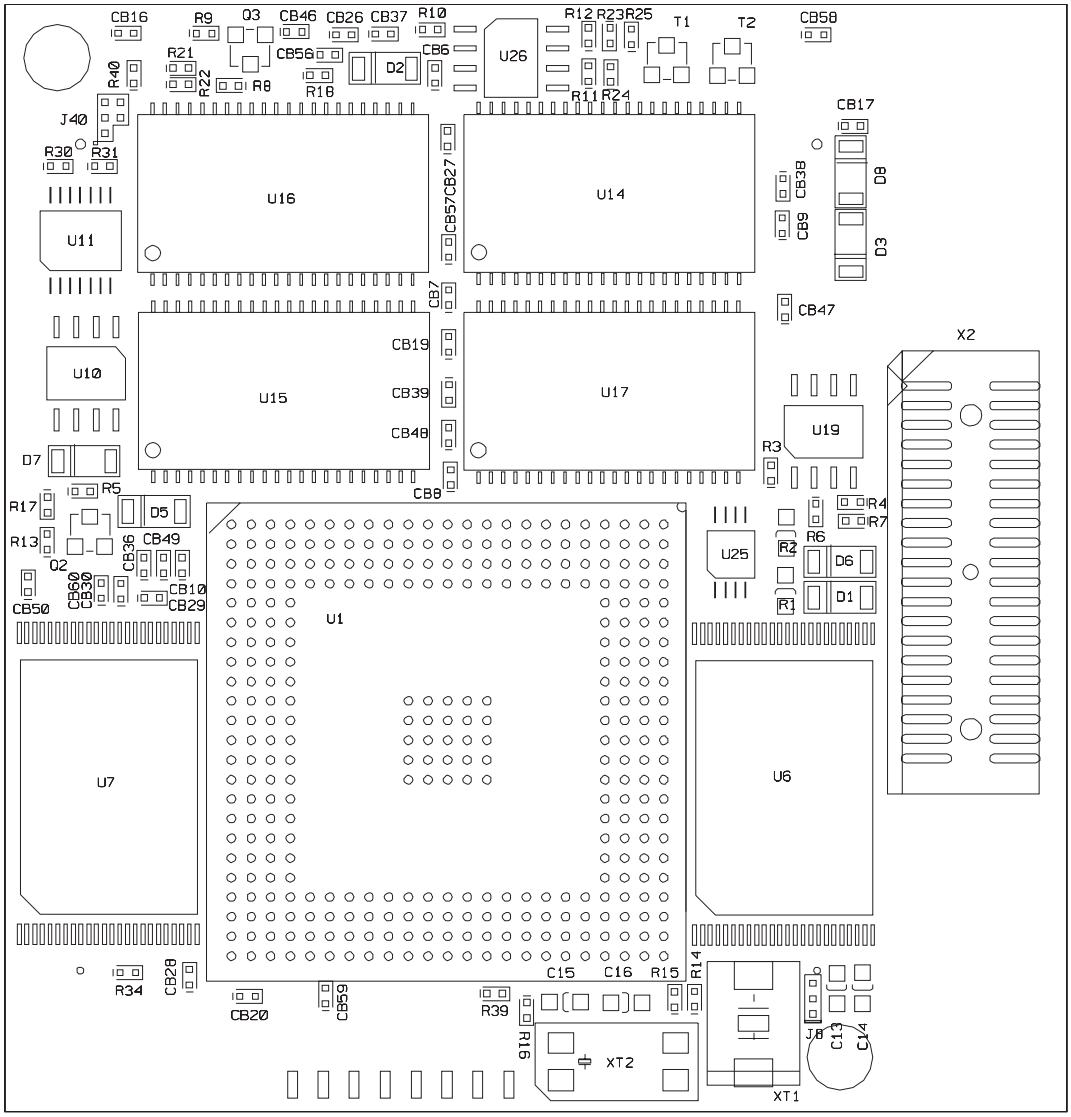


Figure 2: View of the debugCORE-TC1775 (Controller Side)

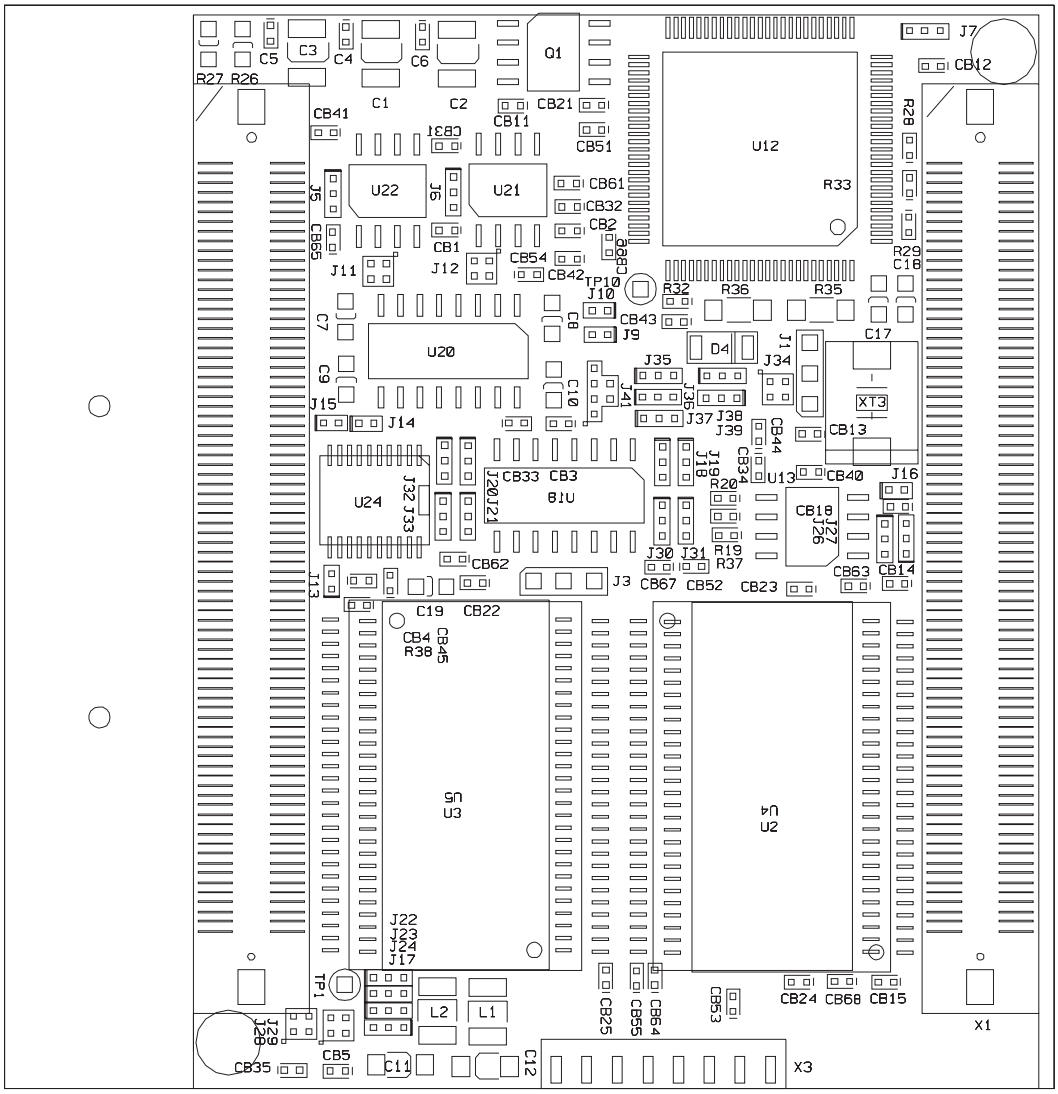


Figure 3: View of the debugCORE-TC1775 (Connector Side)

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 surface mount technology (SMT) connectors (0.635 mm) lining two sides of the module (referred to as phyCORE-connector). This allows the phyCORE-TC1775 to be plugged into any target application like a "big chip".

A new numbering scheme for the pins on the phyCORE-connector has been introduced with the phyCORE specifications. This enables quick and easy identification of desired pins and minimizes errors when matching pins on the phyCORE module with the phyCORE-connector on the appropriate PHYTEC Development Board or in user target circuitry.

The numbering scheme for the phyCORE-connector is based on a two dimensional matrix in which column positions are identified by a letter and row position by a number. Pin 1A, for example, is always located in the upper left hand corner of the matrix. The pin numbering values increase moving down on the board. Lettering of the pin connector rows progresses alphabetically from left to right (*refer to Figure 4*).

The numbered matrix can be aligned with the phyCORE-TC1775 (viewed from above; phyCORE-connector pointing down) or with the socket of the corresponding phyCORE Development Board/user target circuitry. The upper left-hand corner of the numbered matrix (pin 1A) is thus covered with the corner of the phyCORE-TC1775 marked with a white triangle. The numbering scheme is always in relation to the PCB as viewed from above, even if all connector contacts extend to the bottom of the module.

The numbering scheme is thus consistent for both the module's phyCORE-connector as well as mating connectors on the phyCORE Development Board or target hardware, thereby considerably reducing the risk of pin identification errors.

Since the pins are exactly defined according to the numbered matrix previously described, the phyCORE-connector is usually assigned a single designator for its position (X1 for example). In this manner the phyCORE-connector comprises a single, logical unit regardless of the fact that it could consist of more than one physical socketed connector. The location of row 1 on the board is marked by a white triangle on the PCB to allow easy identification.

The following figure (*Figure 4*) illustrates the numbered matrix system. It shows a phyCORE-TC1775 with SMT phyCORE-connectors on its underside.

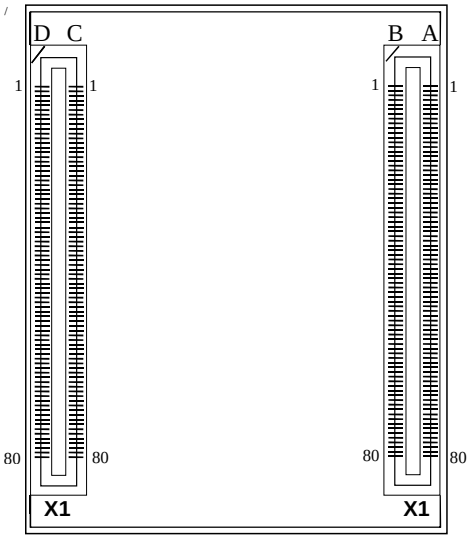


Figure 4: Pinout of the phyCORE-Connector (Connector Side)

Many of the controller port pins accessible at the connectors along 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 phyCORE-connector , as well as descriptions of possible alternative functions. *Please refer to the Infineon TC1775 User's Manual/Data Sheet for details on the functions and features of controller signals and port pins.*

Pin Number	Signal	I/O	Description
Pin Row X1A			
1A	CLKIN	I	Input for optional external clock generator (*refer to J8*)
2A, 7A, 12A, 17A, 22A, 27A, 32A, 37A, 42A, 47A, 52A, 57A, 62A, 67A, 72A, 77A	GND	-	Ground 0 V
3A	P131	I/O	Controller port 13.1, can be configured as external interrupt input Alternative use: General Purpose Timer Unit (GPTU) I/O signal 1
4A	/NMI	I	/NMI Interrupt of the controller
5A	/CS3	O	/CS3 signal used to access Flash of the Ethernet controller (*refer tpo description of J40*)
6A	ALE	O	Address latch enable
8A	/BC0	O	Byte Control signal for data lines D[0..7].
9A, 10A, 11A, 13A, 14A, 15A, 16A, 18A, 24A, 25A, 26A, 28A 48A	A1, A2, A4, A7, A9, A10, A12, A15, A17, A18, A20, A23 A25	I/O	Address lines, are used to access on-board memory Alternative: port 3.9
19A, 20A, 21A, 23A, 29A, 30A, 31A, 33A, 38A, 39A, 40A, 41A, 43A, 44A, 45A, 46A	D1, D2, D4, D7, D9, D10, D12, D15, D18, D19, D20, D22, D25, D27, D28, D30	I/O	Data lines, are used to access on-board memory
34A	/WAIT	I	Microcontroller's wait signal
35A	NC	-	Not connected These contacts should remain unconnected on the target hardware side.
36A	/HLDA	I/O	Hold acknowledge I/O of the processor
49A	/WR	O	Microcontroller's write signal
50A	/CSEMUL	O	Chip Select signal for emulators Alternative: P3.14

Pin Number	Signal	I/O	Description
Pin Row X1A			
51A	/CSOVL	O	Chip Select signal for emulator memory Alternative: P3.15
53A	/CSFPI	I	Chip Select input for external access to the processor-internal FPI bus Alternative: P4.10
54A	/CODE	O	Code fetch status output of the processor Alternative: P4.14
55A	SVM	I/O	Processor supervisor mode Alternative: P4.15
56A, 58A, 59A, 60A, 61A, 63A, 64A, 65A	P1115, P1112, P1110, P119, P117, P114, P112, P111	I/O	I/O port P11 Alternative: I/O signals for General Purpose Timer Arrays (GPTA) IN63/OUT63, IN60/OUT60 IN58/OUT58, IN57/OUT57 IN55/OUT55, IN52/OUT52 IN50/OUT50, IN49/OUT49
66A, 68A, 69A, 70A, 71A, 73A, 74A, 75A	P915, P912, P910, P99, P97, P94, P92, P91	I/O	I/O port P9 Alternative: I/O signals for General Purpose Timer Arrays (GPTA) IN31/OUT31, IN28/OUT28 IN26/OUT26, IN25/OUT25 IN23/OUT23, IN20/OUT20 IN18/OUT18, IN17/OUT17
76A, 78A, 79A, 80A	P815, P812, P810, P89	I/O	I/O port P8 Alternative: I/O signals for General Purpose Timer Arrays (GPTA) IN15/OUT15, IN12/OUT12 IN10/OUT10, IN9/OUT9

Pin Number	Signal	I/O	Description
Pin Row X1B			
1B	CLKOUT	O	TC1775 processor clock
2B	P130	I/O	Processor's port 13.0, can be configured as external interrupt input Alternative: General Purpose Timer Unit (GPTU) I/O signal 0
3B	P132	I	Processor's port 13.2, can be configured as external interrupt input Alternative: General Purpose Timer Unit (GPTU) I/O signal2
4B, 9B, 14B, 19B, 24B, 29B, 34B, 39B, 44B, 49B, 54B, 59B, 64B, 69B, 74B, 79B	GND		Ground 0 V
5B	/CS2	O	Processor's Chip Select signal, is used to access the second on-bord RAM bank
6B	/CS0	O	Processor's Chip Select signal, is used to access the on-bord Flash memory
7B	/RD	O	Processor's read signal
8B, 10B, 11B, 12B, 13B, 15B, 16B, 17B, 23B, 25B, 26B, 27B 47B	A0, A3, A5, A6, A8, A11, A13, A14, A16, A19, A21, A22 A24	I/O	Address lines, are used to access on-board memory Alternative: port 3.8
18B, 20B, 21B, 22B, 28B, 30B, 31B, 32B, 37B, 38B, 40B, 41B, 42B, 43B, 45B, 46B	D0, D3, D5, D6, D8,D11, D13, D14, D16, D17, D21, D23, D24, D26, D29, D31	I/O	Data lines, are used to access on-board memory
33B	/BC1	O	Byte control signal for data lines D[8..15].
35B	/HOLD	I	Processor's hold request input
36B	/BREQ	I/O	Processor's bus request signal Alternative: Port 4.13
48B	/CS1	O	Processor's Chip Select signal, is used to access the first on-bord RAM bank
50B	/ADV	O	Processor's address valid output

Pin Number	Signal	I/O	Description
Pin Row X1B			
51B	/BAA	O	Burst address advance output
52B	/BC2	O	Byte control signal for data lines D[16..23].
53B	/BC3	O	Byte control signal for data lines D[24..31].
55B	/BRKOUT	I	OCDS break output, signal of the processor-internal debug interface
56B, 57B, 58B, 60B, 61B,62B, 63B, 65B	P1114, P1113, P1111, P118, P116, P115, P113, P110	I/O	I/O port P11 Alternative: I/O signals for General Purpose Timer Arrays (GPTA) IN62/OUT62, IN61/OUT61, IN59/OUT59, IN56/OUT56, IN54/OUT54, IN53/OUT53, IN51/OUT51, IN48/OUT48
66B, 67B, 68B, 70B, 71B, 72B, 73B, 75B	P914, P913, P911, P98, P96, P95, P93, P90	I/O	I/O port P9 Alternative: I/O signals for General Purpose Timer Arrays (GPTA) IN30/OUT30, IN29/OUT29, IN27/OUT27, IN24/OUT24, IN22/OUT22, IN21/OUT21, IN19/OUT19, IN16/OUT16
76B, 77B 78B, 80B	P814, P813, P811, P88	I/O	I/O port P8 Alternative: I/O signals for General Purpose Timer Arrays (GPTA) IN14/OUT14, IN13/OUT13 IN11/OUT11, IN8/OUT8

Pin Number	Signal	I/O	Description
Pin Row X1C			
1C, 2C	+2V5_IN	I	Supply voltage +2.5 VDC
3C, 7C, 12C, 17C, 22C, 27C, 32C, 37C, 42C, 47C, 52C, 57C	GND	-	Ground 0 V
4C, 5C	+5 V	I	Supply voltage +5 V=
6C	VBAT	I	Supply voltage for controller-internal stand-by RAMs , the RTC U24 and the serial memory U13; +2.5 VDC
8C,30C	NC	-	Not connected
9C	/BOOT	I	Following a power-on reset (/PORESET) the Boot configuration of the processor is read over the inputs CFG[3..0]. The state of these inputs is determined via the /BOOT signal . /BOOT = low => Boot config. via J30-J33 /BOOT = high => Boot config via J18-J21
10C	/HDRESET	I/O	System's hard-reset signal, /HDRESET is controlled by open-drain drivers
11C	/PORESET	I	Processor's power-on reset, the boot configuration is fetched following a power-on reset
13C, 14C 15C, 16C	P102, P104 P105, P107	I/O	I/O port P10 Alternative: I/O line for General Purpose Timer Arrays (GPTA) IN34/OUT34, IN36/OUT36, IN37/OUT37, IN39/OUT39
18C	CAN_H1	I/O	CANH output of the CAN transceiver for the 2nd CAN intercace
19C	RXD1_TTL	I	Receive line of the 2nd TC1775 UART Alternative: port P12.14. If the alternative function is used, solder jumper J10 must be open in order to disconnect the RS-232 transceiver from the signal
20C	TXD1_TTL	O	Transmit line of the 2nd TC1775 UART Alternative: port P12.5
21C	RxD1	I	RxD input of the RS-232 transceiver for the 2nd serial interface, J10 must be closed to use this interface
23C	TxD1	O	TxD output of the RS-232 transceiver for the 2nd serial interface

Pin Number	Signal	I/O	Description
Pin Row X1C			
24C	RXJ1850	I	RxD input of the SDLM interface Alternative: port P12.10
25C	TXJ1850	O	TxD output of the SDLM interface Alternative: port P12.11
26C	MRST1	I/O	Master transmit / slave receive output / input of the 2nd synchronous serial interface Alternative: port P13.10
28C	MTSR1	I/O	Master receive / slave transmit input / output of the 2nd synchronous serial interface Alternative: port P13.11
31C	SCL	I/O	I^2C clock signal; this signal can be generate by user software via port P8.0 or supplied to the module from an external source, SCL and P8.0 are connected via solder jumper J13
33C	/ETH_LINKLED	O	Link Good signal from the on-board Ethernet controller; solder jumper J39 must be closed at position 1+2 in order to use this function Alternative: port P8.7 ; solder jumper J39 must be closed at position 2+3 in order to use the alternative function
34C	/ETH_LANLED	O	LAN activity LED from the on-board Ethernet controller; solder jumper J38 must be closed at position 2+3 in order to use this function Alternative: port P8.6. solder jumper J38 must be closed at position 1+2 in order to use the alternative function
35C	ETH_RXD-	I	RxD- input of the 10BASE-T receiver from the on-board Ethernet controller
36C	ETH_TXD-	O	TxD- output of the 10BASE-T transmitter from the on-board Ethernet controller
38C	P80	I/O	I/O port P8 Alternative: I/O signal for General Purpose Timer Arrays (GPTA) IN0/OUT0
39C, 40C 41C,43C	P1014, P1013 P1011, P108	I/O	I/O port P10 Alternative: I/O signal for General Purpose Timer Arrays (GPTA) IN46/OUT46, IN45/OUT45, IN43/OUT43, IN40/OUT40

Pin Number	Signal	I/O	Description
Pin Row X1C			
44C, 45C 46C, 48C 49C, 50C 51C, 53C	P514, P513, P511, P58, P56, P55, P53, P50	I/O	I/O port P5 Alternative: Trace output for OCDS debugging Trace14, Trace13 Trace11,Trace8 Trace6, Trace5 Trace3, Trace0
54C	AD1EXTIN0	I	ADC1 external trigger input Alternative: port P12.6
55C	AD1EMUX2	I	Control input 2 of the ADC1 multiplexer
56C	AD1EMUX0	I	Control input 0 of the ADC1 multiplexer
58C	AD0EMUX1	I	Control input 1 of the ADC0 multiplexer
59C	AD0EMUX0	I	Control input 0 of the ADC0 multiplexer
60C,61C 63C, 64C 65C, 66C	AN31, AN29 AN26, AN24 AN23, AN21	I	Analog inputs of ADC1 Alternative: none
70C	Varef1	-	Reference voltage input for ADC1, max. +5 VDC
68C, 69C	AN18, AN16	I	Analog inputs of ADC1 Alternative: none
62C, 67C	Vagnd1	-	Analog Ground 0V for ADC1. Vagnd1 is connected with GND via solder jumper J29 J29 = 3+4
71C, 73C 74C, 75C 76C, 78C 79C, 80C	AN14, AN11 AN9, AN8 AN6, AN3 AN1, AN0	I	Analog inputs of ADC0 Alternative: none
72C, 77C	Vagnd1	-	Analog Ground 0V for ADC0. Vagnd0 is connected with GND via solder jumper J28 J28 = 3+4

Pin Number	Signal	I/O	Description
Pin Row X1D			
1D, 2D	+2.5 V	I	Supply voltage + 2.5 VDC
3D, 9D, 14D, 19D, 24D, 29D, 34D, 39D, 44D, 49D, 54D	GND	-	Ground 0 V
4D, 5D, 6D	+3.3 V	I	Supply voltage + 3.3 V =
7D, 8D	NC	-	Not connected
10D	/RESIN	I	Reset input, controls the system reset /PORESET
11D, 12D 13D, 15D	P100, P101 P103, P106	I/O	I/O port P10 Alternative: I/O line for General Purpose Timer Arrays (GPTA) IN32/OUT32, IN33/OUT33, IN35/OUT35, IN38/OUT38
16D	RXD0_TTL	I	Receive line of first TC1775 UART Alternative: port P12.12 If the alternative function is used, solder jumper J9 must be open in order to disconnect the RS-232 transceiver from the signal
17D	TXD0_TTL	O	Transmit line of first TC1775 UART. Alternative: port P12.13
18D	CAN_L1	I/O	CANL output of the CAN transceiver for the 2nd CAN intercace
20D	CAN_L0	I/O	CANL output of the CAN transceiver for the first CAN intercace
21D	CAN_H0	I/O	CANH output of the CAN transceiver for the first CAN intercace
22D	RxD0	I	RxD input of the RS-232 transceiver for the first serial interface, J9 must be closed to use this interface
23D	TxD0	O	TxD output of the RS-232 transceiver for the first serial interface
25D, 26D	P133, P134	I/O	I/O port P13 (P13.0 – P13.7 are interrupt-capable) Alternative: I/O lines for General Purpose Timer Unit (GPTU) GPT3, GPT4
27D	MRST0	I/O	Master transmit / slave receive output / input of the first synchronous serial interface Alternative: port P13.7
28D	MTSR0	I/O	Master receive / slave transmit input / output of the first synchronous serial interface Alternative: port P13.8

Pin Number	Signal	I/O	Description
Pin Row X1D			
30D	SCLK0	I/O	Clock input/output of the first synchronous serial interface Alternative: port 13.6 or GPTU I/O line 6
31D	P135	I/O	I/O port P13.5 (P13.0 – P13.7 are interrupt-capable) Alternative: I/O lines for General Purpose Timer Unit (GPTU) GPT5
32D	SDA	I/O	Data line of the emulated I^2C bus, port P8.1 is used (*see description for Jumper J14*)
33D	/IRTC	O	RTC interrupt output, /IRTC can be connected with port P3.5 via Jumper J15
35D	ETH_RXD+	I	RxD+ input of the 10BASE-T receiver from the on-board Ethernet controller
36D	ETH_TXD+	O	TxD+ output of the 10BASE-T receiver from the on-board Ethernet controller
37D	P8.1	I/O	I/O port P8.1 Alternative: I/O line for General Purpose Timer Arrays (GPTA) IN1/OUT1
38D, 40D, 41D, 42D	P1015, P1012, P1010, P109	I/O	I/O port P10 Alternative: I/O line for General Purpose Timer Arrays (GPTA) IN47/OUT47, IN44/OUT44, IN42/OUT42, IN41/OUT41
43D, 45D, 46D, 47D, 48D, 50D, 51D, 52D	P515, P512, P510, P59, P57, P54, P52, P51	I/O	I/O port P5 Alternative: Trace output for OCDS Debugging Trace15, Trace12 Trace10,Trace9, Trace7 Trace4, Trace2, Trace1
55D	AD1EMUX1	I	Control input 1 for ADC1 multiplexer
56D	AD0EXTIN1	I	ADC0 external trigger input Alternative: port P12.9
57D	AD0EXTIN	I	ADC0 external trigger input Alternative: port P12.8
58D	AD0EMUX2	I	Control input 2 for ADC0 multiplexer
60D,61D 62D, 63D 65D, 66D 67D, 68D	AN30, AN28 AN27, AN25 AN22, AN20 AN19, AN17	I	Analog inputs of the ADC1 Alternative: none

Pin Number	Signal	I/O	Description
Pin Row X1D			
70D, 71D 72D, 73D 75D, 76D 77D, 78D	AN15, AN13 AN12, AN10 AN7, AN5 AN4, AN2	I	Analog inputs of the ADC0 Alternative: none
59D, 64D 69D	VAGND1	-	Ground 0 V for analog signals of the ADC1. Vagnd1 is connected with GND via solder Jumper J29 = 3+4.
74D, 79D	VAGND0	-	Ground 0 V for analog signals of the ADC0. Vagnd0 is connected with GND via solder Jumper J28 = 3+4.
70D	VAREF0	-	Reference voltage input for ADC0, max. +5 VDC

Table 1: Pinout of the phyCORE-Connector X1

3 Jumpers

For configuration purposes, the phyCORE-TC1775 has 38 solder jumpers, some of which have been installed prior to delivery. *Figure 5* illustrates the numbering of the jumper pads, while *Figure 6* and *Figure 7* indicate the location of the jumpers on the module.

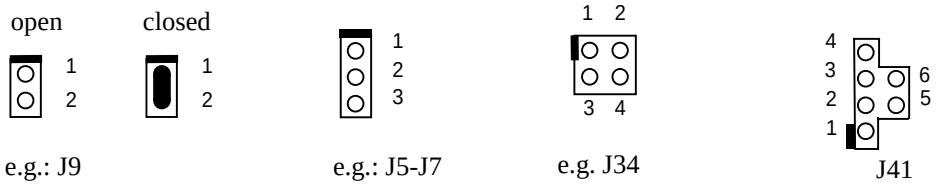


Figure 5: Numbering of the Jumper Pads

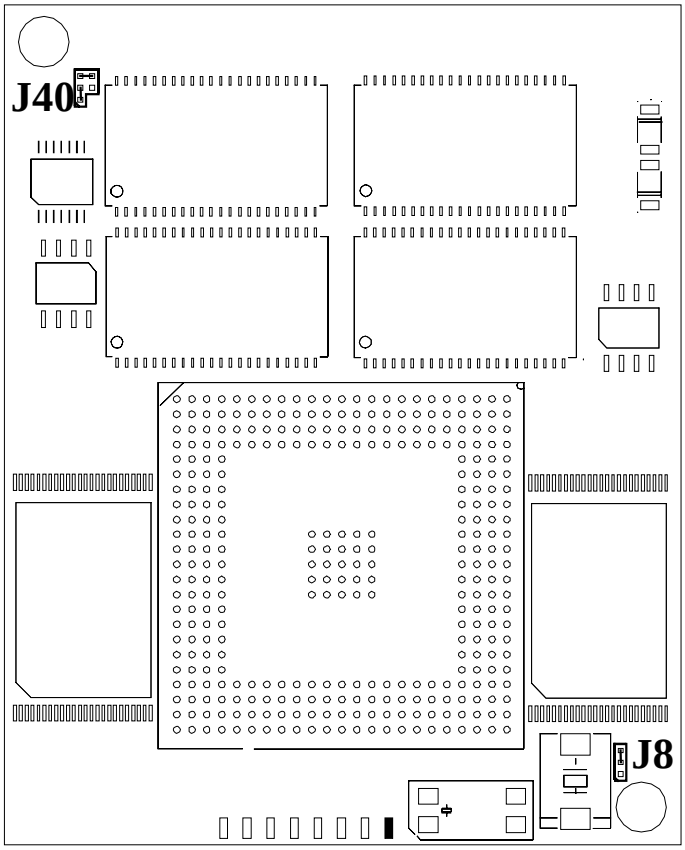


Figure 6: Location of the Jumpers (Controller Side) and Default Settings (phyCORE-TC1775 Standard Version)

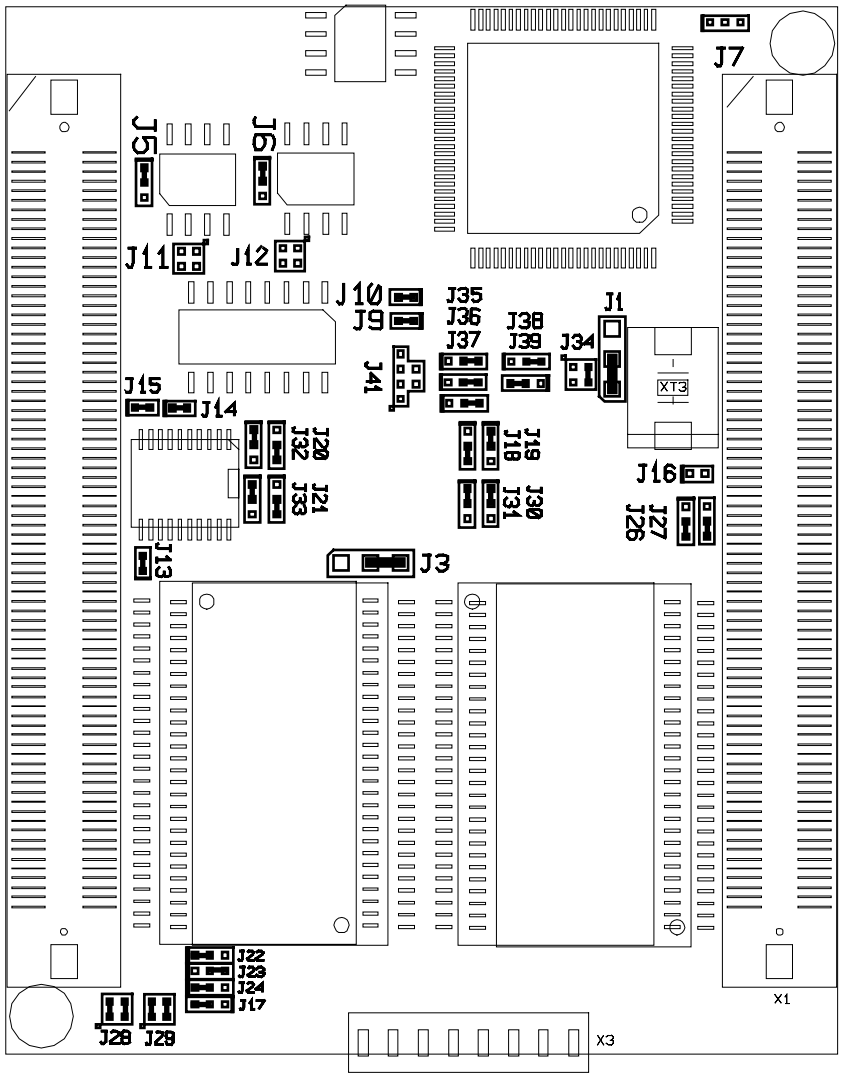


Figure 7: Location of the Jumpers (Connector Side) and Default Settings (phyCORE-TC1775 Standard Version)

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

Jumper	Default	Function
J1		Turns RTC (U24) clock output on and off.
1 + 2	**X**	Clock output is turned off.
2 + 3		Clock output is turned on.
footprint		0R / SMD 0402
J3		Select voltage level at VCC_PERI. VCC_PERI supplies ports 8 - 13 on the TC1775 as well as the RTC U24 and the memory at U13
1 + 2	**X**	VCC_PERI = 5 V
2 + 3		VCC_PERI = 3.3 V
footprint		0R / SMD 0805
J5, J6		Configures the "slope control mode" of the 82C251 CAN transceiver. The signal slope of the CAN signals can be adjusted via corresponding resistors if this jumper is closed. *Please refer to the 82C251 data sheet for details on the resistor selection for slope control.*
1+2	**X**	CAN transceiver in high-speed mode.
footprint		SMD / 0402
J7		Connects the interrupt input on the TC1775 with the interrupt output of the optional Ethernt controller at U12.
1+2		The Ethernet controller releases an NMI.
2+3		The interrupt output of the Ethernet controller extends to port 13.4 on the TC1775.
		NOTE: Do **NOT** configure port 13.4 as output!
open	**X**	Ethernet interrupt not connected to the TC1775.
footprint		0R / SMD 0402
J8		Configures the clock source for the TC1775.
1+2		External clock supply via CLKIN line (module pin 1A).
2+3	**X**	On-board quartz XT1 is used as clock source.
footprint		0R / SMD 0402
J9		Connects the input of the first asynchronous serial interface on the TC1775 (RXD0) with the RS-232 transceiver at U20.
open		RxD0A can be used as port P12.12, no connection to RS-232 transceiver.
closed	**X**	RxD0A used as RS-232 input and conneted to U20.
footprint		0R / SMD 0402

Jumper	Default	Function
J10		Connects the input of the second asynchronous serial interface on the TC1775 (RxD1) with the RS-232 transceiver at U20.
open		RxD1A can be used as port P12.14, no connection to RS-232 transceiver.
closed	**X**	RxD1A used as RS-232 input and conneted to U20.
footprint		0R / SMD 0402
J11, J12		These jumpers connect the TTL_CAN signals with the phyCORE connector pins, for connection to external CAN transceivers, or if CAN outputs are used as standard port pins.
open	**X**	The on-board CAN transceivers U21 and U22 are used.
closed		On-board CAN transceivers not populated.
footprint		0R / SMD 0402
J13, J14		Port pins P8.0 (SCL) and P8.1(SDA) can be used as emulated I^2C interface for communication with the optional on-board RTC at U24 and the EEPROM at U13. These jumpers must be closed when using the I^2C interface.
open		Port 8.0 and port 8.1 freely available.
closed	**X**	Port 8.0 and port 8.1 are sued as I^2C interface.
footprint		0R / SMD 0402
J15		J15 can be used to connect the alarm interrupt output of the optional Real-Time Clock (RTC) at U24 with the interupt-capable port P13.5 on the TC1775.
open		RTC interrupt output is not connected.
closed	**X**	RTC interrupt output is connected with P13.5.
footprint		0R / SMD 0402
J16		Configures the write protection function of the memory device populating U13.
open	**X**	No memory write protection.
closed		Memory write protection for U13 is enabled.
footprint		0R / SMD 0402
J17		The state of the BYPASS signal is latched into the PLL-CLC register of the controller following a power-on-reset. This is used for bypassing the clock generation of the TC1775. *Please refer to the „System. Unit“ Manual for the TC1775 section 3.1.2.5 before changing this jumper setting.*
1+2		*Please refer to the „System. Unit“ Manual for the TC1775 section 3.1.2.5.*
2+3	X	PLL on TC1775 is used.
footprint		0R / SMD 0402

Jumper	Default	Function
J18, J19, J20, J21		Following a power-on-reset, the desired standard boot configuration for the TC1775 configured with these jumpers is latched. The boot configuration selects from which Code memory to fetch instructions and which processing interface to use.
1+2, 1+2, 2+3, 1+2	**X**	Start out of the external memory at address A0000000H (CFG[3:0] = 1101) *Please refer to the TC1775 Manual for alternative settings.*
footprint		0R / SMD 0402
J22, J23, J24		These jumpers configure the PLL factor (N-factor) following a power-on-reset. The system frequency results by the K-divider which can be configured via software, the following equation is used: fsys = (fosc * N-factor) / K-divider
1+2, 2+3 1+2	**X**	N-factor = 10 ; together with the 16 MHz default quartz and a configured K-divider of 4 this results in the as maximum specified system frequency of 40 MHz *Please refer to the TC1775 Manual for alternative settings.*
footprint		0R / SMD 0402
J30, J31, J32, J33		An alternative boot configuration is latched via these jumpers if the /BOOT signal is active. The boot configuration selects from which Code memory to fetch instructions and which processing interface to use.
1+2, 1+2, 1+2, 1+2	**X**	Start out of the internal Boot-ROM. *Please refer to the TC1775 Manual for alternative settings.*
footprint		0R / SMD 0402
J26, J27		J26 and J27 configure the I^2C bus slave address (A2 und A1) of the serial memory at U13. The slave ID for I^2C memory chips is encode in the high-nibble of the address and set to 0xA. The low-nibble is created by A2, A1, A0 and the R/W bit. A0 is connected to GND. Please note that the RTC at U24 is also connected to the I^2C bus. The RTC address is preconfigured in the chip and set to 0xA2/ 0xA3.
1+2, 2+3		A2 = 0, A1 = 0, A0 = 0 (0xA0 / 0xA1)
1+2, 1+2		A2 = 1, A1 = 0, A0 = 0 (0xA8 / 0xA9)
2+3, 2+3		A2 = 0, A1 = 1, A0 = 0 (0xA4 / 0xA5)
2+3, 1+2	**X**	A2 = 1, A1 = 1, A0 = 0 (0xAC / 0xAD) I^2C slave address 0xAC for write operations and 0xAE for read access.
footprint		0R / SMD 0402

Jumper	Default	Function
J28, J29		These Jumpers can be used to connect the first reference voltage inputs for the two on-chip analog to digital converters (ADC) with the 5 V supply voltage of the modul.
1+2 and 3+4	**X**	The reference voltage inputs Varef0 and Varef1 are connected with the 5 V supply of the module.
1+2 and 3+4		
open		External reference voltage sources can be supplied via pins 80D, 79D resp. 70C, 69D of the phyCORE connector.
footprint		0R / SMD 0402
J34, J35, J36, J37, J38, J39		These jumpers configure the routing of port signals P8.2 - P8.7 and the Ethernet signals. Configuration of these jumpers depends on whether the optional Ethernet controller populates the module or not.
3+4	**X**	No Ethernet controller populated.
1+2, 1+2, 1+2, 1+2, 1+2		Port P8.2 – P8.7 is routed to connector pins. P8.2 -> 36D ; P8.3 -> 36C ; P8.4 -> 35D ; P8.5 -> 35C ; P8.6 -> 34C ; P8.7 -> 33C
1+3 and 2+4		Ethernet controller populates U12. The signals from the Ethernet controller are routed to the phyCORE connector pins.
2+3, 2+3		Port P8.2 – P8.7 not available.
2+3		
footprint		0R / SMD 0402
J40		Jumper J40 defines the usage of the controller-internal Chip Select signals. The jumper configuration depends on the populated Flash devices and the optional Ethernet controller. The specified default settings assume a phyCORE module with standard memory configuration.
4+5	X	Standard Flash (U6 and U7) connected with /CS0 /CS3 is freely available
1+2 and 3+4		Burst mode Flash (U4 and U5) connected with /CS0 and Ethernet controller connected with /CS3
2+5 and 3+4		Burst mode Flash (U4 and U5) connected with /CS0 and standard Flash (U6 and U7) connected with /CS3
footprint		0R / SMD 0402

Jumper	Default	Function
J41		The module has been prepared for future expansion with a Boot manager device (U19). If this IC populates the module, J40 is used to configure the signal source that releases a power-on reset and subsequential start with an active /BOOT signal. This IC is in development and not available at this time. We will update the documentation as soon as this IC is available.
open	**X**	U19 not populated
footprint		0R / SMD 0402

Table 2: Jumper Settings

4 Power System and Reset Behavior

Operation of the phyCORE-TC1775 requires three different supply voltages.

Supply voltage 1: +2.5 V
Supply voltage 2: +5 V
Supply voltage 3: +3.3V

Note:
All three supply voltages are required for proper operation of the module. The module must **never** be put into operation whit only one or two of the supply voltages connected to the device!

Circuitry on the module ensures that the power-on sequence as specified in the controller datasheet is met. This means that the 5 V supply is turned on first, followed by the 3.3 V and the 2.5 V as the last supply. If one of the supply voltages fails, all other, and in their value lower, voltages will be turned off.

Note:
The input voltage VBAT will be turned off by this circuitry.

Once all voltages have reached their target level the voltage supervisory circuit keeps the /PORESET reset signal at low level (low is the active level) for additional 200 ms. Then the /PORESET signal switches to high level (inactive) and the controller's boot sequenz starts.

5 Boot Characteristics

Activation of the /PORESET signal results in an asynchronous reset of the controller, which subsequently rereads the configuration for the PLL (phase locked loop) (signals CLKSEL[0..2]), activates the signal /HDRESET and then awaits the occurrance of the following events:

1. The PLL has started and is ready for use.
2. The /PORESET signal has become inactive.

If both of these conditions are met then the controller reads the configuration signals CFG[0..3] as well as /OCDSE and /BRKIN and subsequently decides from which code area the program execution will start.
The /OCDSE and /BRKIN signals are controlled by a debugger, which is attached to the OCDS1 connector, and are ignored for further discussion in this manual (*see System Unit User's Manual for the TC1775, section "Boot Options"*).

Start Address following Power-On-Reset

Selection between two pre-configured start addressses is possible with the help of the /BOOT signal:

/BOOT = 1 (inactive)

Jumper (default)	CFG[0..3]	Boot Source	PC Start Address
JP18 = 2 + 3 JP19 = 1 + 2 JP20 = 2 + 3 JP21 = 2 + 3	1011	external memory (cached)	A00000000H

/BOOT = 0 (active)

Jumper	CFG[3..0]	Boot Source	PC Start Address
JP30 = 1+2 JP31 = 1+2 JP32 = 1+2 JP33 = 1+2	0000b (default)	ASC Bootstrap Loader	BFFFFFFFH
JP30 = 2+3 JP31 = 1+2 JP32 = 1+2 JP33 = 1+2	X001b	SSC Bootstrap Loader	BFFFFFFFH
JP30 = 1+2 JP31 = 2+3 JP32 = 1+2 JP33 = 1+2	X010b	CAN Bootstrap Loader	BFFFFFFFH

If "System Start beginning at address A0000000h" was chosen, then the controller will perform a "blind-read" from address A0000004h, in order to read the "EBU external boot memory configuration" (*see section 12.8 of the TC1775 Systems Units Manual*). This first read access occurs with the fixed standard values, which support the reading of as many different memory devices as possible. The "EBU external boot memory configuration word" that was read must have valid values for the read access to the memory connected to /CS0, so that the subsequent accesses occur with the correct timing. The values relevant for the timing are subsequently copied to the register BUSCON0. From this point on there is a valid configuration for read accesses to the external Flash memory, and the program execution can begin at address A0000000h.

The valid boot memory configuration word for the phyCORE-TC1775 is 0x8280.

6 System Memory

In principle, two different memory models are available. The first memory model is the one that is active after a reset. The run time memory model, in contrast, is configured via software by the application.

6.1 Memory Model following Reset

The internal Chip Select logic provided by the TC1775 controller is used exclusively on the phyCORE-TC1775. Hence the memory model as described in the TC1775 User's Manual is valid after reset.

6.2 Runtime Memory Model

The runtime memory model is configured via software using the internal registers of the TC1775. There is a register set containing a BUSCON and ADDSEL register for each of the controllers Chip Select signals. The values in the Bus Configuration Registers (EBU_BUSCON0-3) inform the processor of how it should access the connected memory devices (wait states, bus width, etc.). The Address Selection Registers (EBU_ADDSEL0-3) define the address range in which the corresponding Chip Select signal is active. The following list shows the default settings for the Chip Select signal assignment. The Chip Select signals /CS0 and /CS3 can be rerouted with the help of Jumper J40 depending on the module's memory configuration (*see section 3*).

/CS0	external on-board burst-mode Flash memory (4 MByte)
/CS1	external on-board SRAM bank 0 (1 MByte)
/CS2	frei (oder SRAM bank 1)
/CS3	10 Mbit Ethernet Controller CS8900A

The runtime memory model is application-dependent. The following table (*Table 3*) shows an example of how such a runtime model can be configured.

Address Range	Capacity	Periphery	TC1775 Register
0 x A000 0000 0 x A03F FFFF	4 MByte	on-board burst Flash AM29BL162C-65 (/CS0)	EBU_BUSCON0= 0x0002 0C80 EBU_ADDSEL0= 0xA000 0051
0 x B000 0000 0 x B00F FFFF	1 MByte	on-board SRAM (/CS1)	EBU_BUSCON0= 0x00020000 EBU_ADDSEL0= 0xB000 0071
0 x B010 0000 0 x B01F FFFF	1 MByte	/CS2 freely available	EBU_BUSCON0= 0xXXXX XXXX EBU_ADDSEL0= 0xB010 0071
0 x B020 0000 0 x B020 0FFF	4 kByte	on-board CS8900A Ethernet Controller /CS3	EBU_BUSCON0= 0x000107FF EBU_ADDSEL0= 0xB020 00F1

Table 3: Runtime Memory Map

The register values for /CS2 depend on the connected peripheral device. Numbers shown an "X" define the bus interface properties, such as bus width, bus type, wait states etc.

Note:
Table 12-6 in the TC1775 System Units Manual shows possible values for the address ranges. You can find the values in the column "No. of Address Bits compared..." as hexadecimal values. These values can be put directly into bits 4-7 of the corresponding EBU_ADDSEL register.

6.3 Flash Memory

Use of Flash as non-volatile memory on the phyCORE-TC1775 provides an easily reprogrammable means of code storage.

The Flash memory operates in 16-bit mode and has 32-bit organization on the module. The phyCORE-TC1775 offers the option of populating 4 MByte burst-mode Flash at U4 and U5 as well as up to 8 MByte of standard Flash at U6/U7. The setting for Jumper J40 determines which of the controller's Chip Select signal will be used for the respective Flash bank. If J40 is closed at position 3+4 and 2+5, then /CS0 is connected to the Flash memory bank U4/U5. With this configuration this memory bank is active following power-on reset.

Note:
If the Ethernet controller populates the module at U12, only one of two possible Flash memory banks can be used. This is because the Ethernet Controller is using Chip Select signal /CS3.

6.3.1 Burst-Mode Flash (U4, U5)

In order to process code quickly, the TC1775 is equipped with an internal instruction cache and offers the possibility of addressing burst-mode Flash. Burst-mode Flash type AM29BL162CB-65RZI is implemented on the phyCORE-TC1775. This Flash has a standard access time of 65 ns and a burst-mode access time of 18 ns. No external programming voltage is required.

The concept of burst-mode Flash is to shorten the access time to the Flash contents by reducing access cycles. This means that the Flash auto-increments the address independently, whereby address setup times are eliminated. Since in burst-mode entire code blocks with a maximum size of 32 words (applies especially to the AM29BL162CB-65RZI) are read, this method is most effective for coherent code areas.

The burst-mode is activated when the user instructs the TC1775 to perform future code fetches on /CS0 directly with the Instruction Fetch Unit instead of using the internal CPU bus (FPI bus). The TC1775 can be instructed by setting the applicable parameters within controller's registers. Since activation of the burst-mode leads to a short term suspension of the External Bus Unit (EBU), the responsible program code has to first be copied to the internal RAM of the CPU, and then be started and executed there. The exact procedures can be found described in the included program example "burstmode".

Note:
If you would like to use burst-mode, the Flash devices at U4/U5 have to be connected with /CS0 (J40 = 3+4). Furthermore, code fetches are only possible via /CS0 when burst mode is activated.

6.3.2 Standard Flash (U6, U7)

The following standard Flash devices can populate the phyCORE-TC1775:

Type	Capacity	Manufacturer	Device Code	Manufacturing Code
29LV200T/B	256 kByte	AMD	223B/22BF	01
29LV200T/B	256 kByte	Fujitsu	223B/22BF	04
29LV200T/B	256 kByte	ST	0051/0057	20
29LV400T/B	512 kByte	AMD	22B9/22BA	01
29LV400T/B	512 kByte	Fujitsu	22B9/22BA	04
29LV400T/B	512 kByte	ST	00EE/00EF	20
29LV800T/B	1 MByte	AMD	22DA/225B	01
29LV800T/B	1 MByte	Fujitsu	22DA/225B	04
29LV800T/B	1 MByte	ST	00D7/005B	20
29LV160T/B	2 MByte	AMD	22C4/2249	01
29LV160T/B	2 MByte	Fujitsu	22C4/2249	04
29LV160T/B	2 MByte	ST	22C4/2249	20
29DL323TE	4 MByte	Fujitsu	2250	04

Table 4: Flash Memory Types and Manufacturers

These Flash devices are programmable with 3.3 VDC. No dedicated programming voltage is required. Use of Flash memory allows for in-circuit reprogramming of the module.

Jumper J40 (*refer to section 3*) configures the Chip Select signal used to access Flash bank at U6/U7.

Use of a Flash device as the only code memory results in no or only a
As of the printing of this manual, Flash devices generally have a life expectancy of at least 100.000 erase/program cycles.

6.4 Serial Memory, EEPROM/FRAM/SRAM (U13)

The phyCORE-TC1775 features a non-volatile memory with an I^2C interface. This memory can be used for storage of configuration data or operating parameters, that must not be lost in the event of a power interruption. Depending on the module's configuration, this memory can be in the form of an EEPROM, FRAM or SRAM. The available capacity ranges from 512 Byte to 32 kByte.

Since the TC1775 does not provide an on-chip I^2C interface, the protocol has to be generated by software. The processor port pins P80 and P81 are used for emulating the I^2C interface. Solder jumpers J13 and J14 connect these port pins to the applicable SCL and SDA signals.

Table 5 provides an overview of compatible components for U13 at the time this manual was printed.

Type	Capacity	I^2C Clock	Address Pins	Write Cycles	Data Retention	Component	Manuf.
EEPROM	256/512 Byte	400 kHz	A2, A1, A0	1 000 000	100 years	CAT24WC 02/04	Catalyst
	1/ 2 kByte	400 kHz	A2, A1, A0	1 000 000	100 years	CAT24WC 08/16	Catalyst
	4/8 kByte	400 kHz	A2, A1, A0	1 000 000	100 years	CAT24WC 32/64	Catalyst
	32 kByte	1 MHz	A1, A0	100 000	100 years	CAT24WC 256	Catalyst
FRAM	512 Byte	400 kHz	A2, A1	10 Billion	10 years	FM24C04	Ramtron
	8 kByte	1 MHz	A2, A1, A0	10 Billion	10 years	FM24C64	Ramtron
SRAM	256 Byte	100 kHz	A2, A1, A0	-	-	PCF8570	Philips

Table 5: Memory Device Options for U13

It is important to note that the RTC is also connected to the I^2C bus according to standard. The RTC can operate with a bus frequency of up to 400 kHz. Therefore the use of high bus frequencies for accessing the serial memory is not recommended.

The RTC possesses the I^2C bus slave address 0xA2 / 0xA3. The slave address of the serial memory must be selected accordingly using the solder jumpers J26 (A1) and J27 (A2), so that no collision occurs. The address input A0 is hardwired to GND.

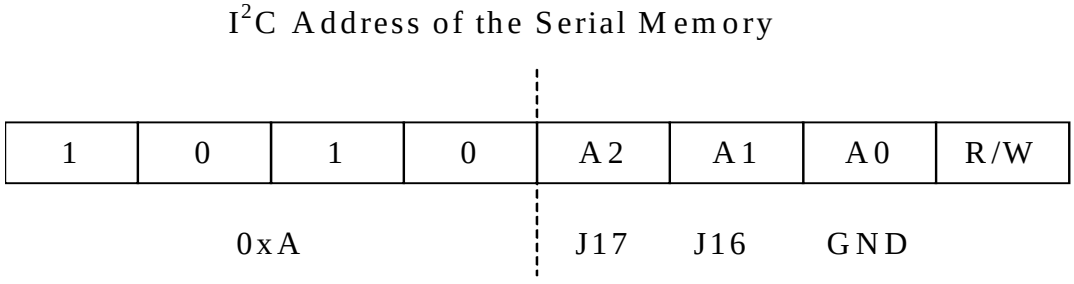


Figure 8: I^2C Slave Address of the Serial Memory

The following configuration options are available:

I^2C Address	**J26 A1**	**J27 A2**
0xA0 / 0xA1	1 + 2	2 + 3
0xA4 / 0xA5	2 + 3	2 + 3
0xA8 / 0xA9	1 + 2	1 + 2
0xAC / 0xAD	2 + 3	1 + 2

Table 6: I^2C Addresses for Serial Memory

Address lines A1 and A2 are not always made available by certain serial memory types. This should be noted when configuring the I^2C bus slave address.

7 Serial Interfaces

7.1 RS-232 Interface

One dual-channel RS-232 transceiver is located on the phyCORE-TC1775 at U20. This device converts the signal levels for the RXD0_TTL and TXD0_TTL lines, as well as those of the second serial interface, RXD1_TTL and TXD1_TTL from TTL level to RS-232 level. The RS-232 interface enables connection of the module to a COM port on a host-PC. In this instance the RxD0 line of the transceiver is connected to the TxD line of the COM port; while the TxD0 line is connected to the RxD line of the COM port. The Ground potential of the phyCORE-TC1775 circuitry needs to be connected to the applicable Ground pin on the COM port as well.

The microcontroller's on-chip UART does not support handshake signal communication. However, depending on user needs, handshake communication can be software emulated using port pins on the microcontroller. Use of an RS-232 signal level in support of hand-shake communication requires use of an external RS-232 transceiver not located on the module.

Furthermore there is the possibility of using the TTL signals of both UART channels externally. These are available on the phyCORE-connector at X1C19, X1C20 (RXD1_TTL, TXD1_TTL) and X1D16, X1D17 (RXD0_TTL, TXD0_TTL). This becomes necessary if galvanic isolation of the interface signals is required. The TTL transceiver outputs of the on-board RS-232 device can be decoupled from the receive signals RXD0_TTL and RXD1_TTL via solder jumpers J19 and J10. This is necessary so that no external transceivers drive signals against the on-board transceiver. The transmit signals TXD0_TTL / TXD1_TTL, in contrast, can be connected parallel to the transceiver inputs, without causing a collision.

7.2 CAN Interface

The phyCORE-TC1775 is designed to house two CAN transceivers at U22 and U21 (either PCA82C251 or Si9200EY). The CAN bus transceiver devices support signal conversion of the CAN transmit (CANTx) and receive (CANRx)lines. The CAN transceiver supports up to 110 nodes on a single CAN bus. Data transmission occurs with differential signals between CANH and CANL. A Ground connection between nodes on a CAN bus is not required, yet is recommended to better protect the network from electromagnetic interference (EMI). In order to ensure proper message transmission via the CAN bus, a 120 Ohm termination resistor must be connected to each end of the CAN bus.

Furthermore, it is required that the CANH and CANL input/output voltages do not exceed the limiting values specified for the corresponding CAN transceiver (for the 82C250 -8 VDC / +18 VDC, on the 82C251 ±40 VDC). If the CAN bus system exceeds these limiting values optical isolation of the CAN signals is required.

For larger CAN bus systems, an external opto-coupler should be implemented to galvanically separate the CAN transceiver and the phyCORE-TC1775. This requires purchasing a module without the on-board CAN transceivers installed. Instead, the TxDCAN0/RxDCAN0 and TxDCAN1/RxDCAN1 signals are routed to the phyCORE-connector at pins X1D21, X1D20 und X1C18, X1D18 with their TTL level. This requires Jumpers J11 and J12 to be closed at positions 1+2 and 3+4 (*refer to section 3 for details*). For connection of the CANTx and CANRx lines to an external transceiver we recommend using a Hewlett Packard HCPL06xx or a Toshiba TLP113 HCPL06xx fast opto-coupler. Parameters for configuring a proper CAN bus system can be found in the DS102 norms from the CiA[1] (CAN in Automation) User and Manufacturer's Interest Group.

---

[1]: CiA: CAN in Automation. Founded in March 1992, CiA provides technical, product and marketing information with the aim of fostering Controller Area Network's image and providing a path for future developments of the CAN protocol.

On-board transceiver configuration:

The transceivers U22 and U21 can be switched to standby with Jumpers J5 and J6 set to position 2+3. Furthermore, the rise time can be configured by connecting a resistor to GND. This results in reduced interference from the CAN bus when using lower baud rates. *For additional information refer to the data sheet for the 82C250/82C251 CAN transceivers.*

7.3 On-Chip Debug Support

The TC1775 offers access to its internal OCDS (OCDS = On-Chip Debug Support) module via an expanded JTAG interface. The JTAG interface enables external access to the system without requiring that some sort of service software (i.e. monitor program) run on the target. Standard cross development systems/debug interfaces, such as the GNU TriCore Development Suite from Hightec offer the possibility of setting breakpoints as well as access to the controller's internal registers via the JTAG interface.

The OCDS1/JTAG interface on the TC1775 extends to a 16-pin connector at X3 located on the edge of the phyCORE module. An external converter (Wiggler, etc.) can be connected at X3, which allows for connectivity of the TC1775 to a host PC.

The phyCORE-TriCORE Develoment Board (article number PCM-993) integrates such a converter, thus allowing direct connectivity with a development computer (*refer to section 15*).

Pin #	Signal	Description
1	TMS	JTAG module state machine control input
2	2V5	Supply voltage for external wiggler
3	TDO	JTAG module serial data output
4	GND	Ground
5	CLKOUT	Clock output
6	GND	Ground
7	TDI	JTAG module serial data input
8	/PORESET	Power-on reset input
9	/TRST	JTAG module reset/enable input
10	/BRKOUT	OCDS break output
11	TCLK	JTAG module clock input
12	GND	Ground
13	/BRKIN	OCDS break output
14	/OCDSE	OCDS enable input
15	NC	not connected
16	NC	not connected

Table 7: OCDS1 Connector X3 Pin Assignment

Note:
Special care must be take when implementing your own external converter in order to ensure the applicable 2.5 VDC supply voltage is applied at pin 2 of connector X3.

8 CS8900A Ethernet Controller

The CS8900A is an IEEE 802.3 Single-Chip Ethernet-Controller that offer a variety of configuration options. The following section only describes the specific characteristics for operating the CS8900A on the phyCORE-TC1775. *Additional technical data for the CS8900A Ethernet controller are available in the corresponding data sheet.*

8.1 I/O and Memory Mode

The phyCORE-TC1775 supports the I/O mode as well as memory mode of the CS8900A controller. These two modes specify different types of access to the internal memory of the Ethernet controller. It is important to note the specific access type when creating or using software drivers. Following a reset the I/O mode is active per default. In this mode access occurs with the help of the PacketPage-Pointer and PacketPage-Data registers. The 4 kBytes of internal memory can be addressed directly in memory mode. Memory mode offers minor advantages in terms of execution speed.

8.2 Addressing and Bus Configuration

Using the Ethernet controller requires Jumper J40 to be closed at position 1+2. This connects the TC1775's Chip Select signal 3 (/CS3) to the Ethernet controller. The TC1775's register Ebu_Buscon3 should be set so that accesses to the CS8900A occur with the correct timing and correct 16-bit bus width. The CS8900A only provides a 16-bit data bus.

Example:

```
EBU_BUSCON3=0x000118c1,     // 16-bit bus, 12 read wait states,
                            // 12 wait states for write
```

The applicable start address of the Ethernet controller in the application is based on the value given in the register Ebu_Addsel3 added to the base address set in the CS8900A controller (default = 300h), for example, B0800300h.

Example:

```
EBU_ADDSEL3=0xB08000F1,   // /CS3 for 4 kByte at B0800000h
```

8.3 Interrupt Control

The Ethernet controller CS8900A offers the possibility of generating an interrupt following the receipt of an Ethernet frame. All of the CS8900A's interrupt outputs are combined on the phyCORE-TC1775 and can be connected to two different interrupt inputs on the TC1775 CPU with the help of Jumper J7 (*refer to section 3, "Jumpers"*).

8.4 Hardware Standby

To conserve power the CS8900A can be switched to hardware power-down mode. To activate this feature the /ETH-Sleep signal must be set to low level. The signal /ETH-Sleep can be connected to port pin P82 on the TC1775 using Jumper J34 (*refer to section 3, "Jumpers"*).

8.5 MAC Address

In a computer network such as a "local area network" (LAN), the MAC (Media Access Control) address is a ***unique*** computer hardware number. For a connection to the Internet, a table is used to convert the assigned IP number to the hardware's MAC address.

In order to guarantee that the MAC address is unique, all addresses are managed in a central location. PHYTEC has acquired a pool of MAC addresses. The MAC address of the phyCORE-TC1775 is located on the bar code sticker attached to the module. This number is a 12-position HEX value. The MAC address has already been programmed into the Ethernet controller's EEPROM, so that the the MAC address is automatically loaded to the Ethernet controller following a reset (*refer to section 8.6*).

8.6 Configuration Data in EEPROM (U10)

The EEPROM (U10) connected to the Ethernet controller can be used to store configuration data that are automatically loaded into the CS8900A following a reset. The EEPROM can be programmed on-board via the Ethernet controller. For detailed programming instructions please refer to the CS8900A data sheet. The EEPROM is pre-programmed with the MAC address at time of delivery (*refer to section 8.5, "MAC Address"*).

8.7 10Base-T Interface

The phyCORE-TC1775 has been designed for use in 10Base-T networks exclusively. The AUI interface of the CS8900A is not available to the user. In order to connect the module to an existing 10Base-T network some external circuitry is required. It is important to note that the CS8900A variant implemented on the phyCORE-TC1775 operates with a 3 V supply voltage (CS8900A-Q3). Please refer to the CS8900A-Q3 documentation for circuitry examples.

If you are using the applicable Development Board for the phyCORE-TC1775 (part number PCM-993), the external circuitry mentioned above is already integrated on the board. In addition, PHYTEC also offers an Ethernet adapter module (part number EAD-001-3V).

9 Real-Time Clock RTC-8564 (U24)

For real-time or time-driven applications, the phyCORE-TC1775 is equipped with an RTC-8564 Real-Time Clock at U24. This RTC device provides the following features:

- Serial input/output bus (I^2C), address 0xA2
- Power consumption
  Bus active (400 kHz): < 1 mA
  Bus inactive, CLKOUT inactive: < 1 µA
- Clock function with four year calendar
- Century bit for year 2000-compliance
- Universal timer with alarm and overflow indication
- 24-hour format
- Automatic word address incrementing
- Programmable alarm, timer and interrupt functions

If the phyCORE-TC1775 is equipped with a battery (VBAT), the Real-Time Clock runs independently of the board's power supply.

Programming the Real-Time Clock is done via the I^2C bus (address 0xA2/0xA3), which is connected to port P80 (SCL) and port P81 (SDA). Solder jumpers J13 and J14 connect the corresponding porcessor pins to the I^2C bus. Since the TC1775 does not provide an on-chip I^2C interface, the protocol has to be generated by software.

The Real-Time Clock also provides an interrupt output that extends to port P13.5 via Jumper J15. An interrupt occurs in case of a clock alarm, timer alarm, timer overflow and event counter alarm. An interrupt must be cleared by software. With the interrupt function, the Real-Time Clock can be utilized in various applications. *For more information on the features of the RTC-8564, refer to the corresponding Data Sheet.*

Note:
After connection of the supply voltage, or after a reset, the Real-Time Clock generates **no** interrupt. The RTC must first be initialized (*see RTC Data Sheet for more information*)

10 Standby Power Supply

In some applications it is desirable to disconnect all supply voltages from the module, but still maintain certain data in the volatile memory. For such cases the phyCORE-TC1775 offers the input pin VBAT (X1C6).

If a voltage of **2.5 V** is supplied over VBAT, then the data is maintained in the RTC, the controller's internal SBRAM and in the memory device mounted at U13, even if all other supply voltages have been turned off.

Connecting a battery via the VBAT input is not mandatory for normal operation of the module, since all devices listed above are supplied with power by the module's operating voltage 2V5_IN.

11 Remote Supervisory Chip (U19)

Space U19 is intended to be populated by a Remote Supervisory Chip[1]. This IC can initiate a boot sequence via a serial interface, such as
RS-232 or CAN. The RSC can start the PHYTEC FlashTools without requiring a manual release of the boot sequence on the phyCORE module applied via a BOOT jumper or button. This enables a remote controlled software update of the on-board Flash device.

This function can be controlled by various interfaces. Solder Jumper J41 configures the remote download source. The Remote Supervisory Chip is under development and not available at this time. Accordingly, Jumper J41 remains open in the default configuration. *Refer to section 3 for details on jumper settings for J41*.

This feature will be available on future phyCORE modules. The following configuration options will then be available:

Download Source	**J41**
no remote download	open*
RxD0	1 + 2
RxD1	2 + 6
CAN0	3 + 5
CAN1	3 + 4

* = Default Setting

Table 8: Remote Download Source

1: This feature is under development and not available at this time.

12 debugCORE-TC1775

The debugCORE-TC1775 is a special debugging version Single Board Computer (SBC) module which is 100 % function-compatible with the phyCORE-TC1775. As opposed to the phyCORE-TC1775, which was developed for use in OEM applications, the debugCORE-TC1775 is used for simple and efficient error detection and debugging using a hardware emulator. To support its debugging function, the debugCORE-TC1775 provides all required connectors for emulator connectivity including a Trace port.

Since the debugCORE is 100 % function-compatible with the phyCORE-TC1775, it can easily be inserted directly into the application in place of the phyCORE-TC1775 for the purpose of hardware debugging.

Please note that when using the Trace port (OCDS2) port 5 of the controllers is no longer available for other functions.

12.1 Components of the debugCORE

As described previously, the debugCORE-TC1775 represents a superset and expansion of the phyCORE-TC1775.

The following components have been added for simple debugging:

- pin header connector (X3), OCDS level 1
- one 50-pin SMD-connectors (X2), OCDS level 2

The following figure shows the positions of the additional components.

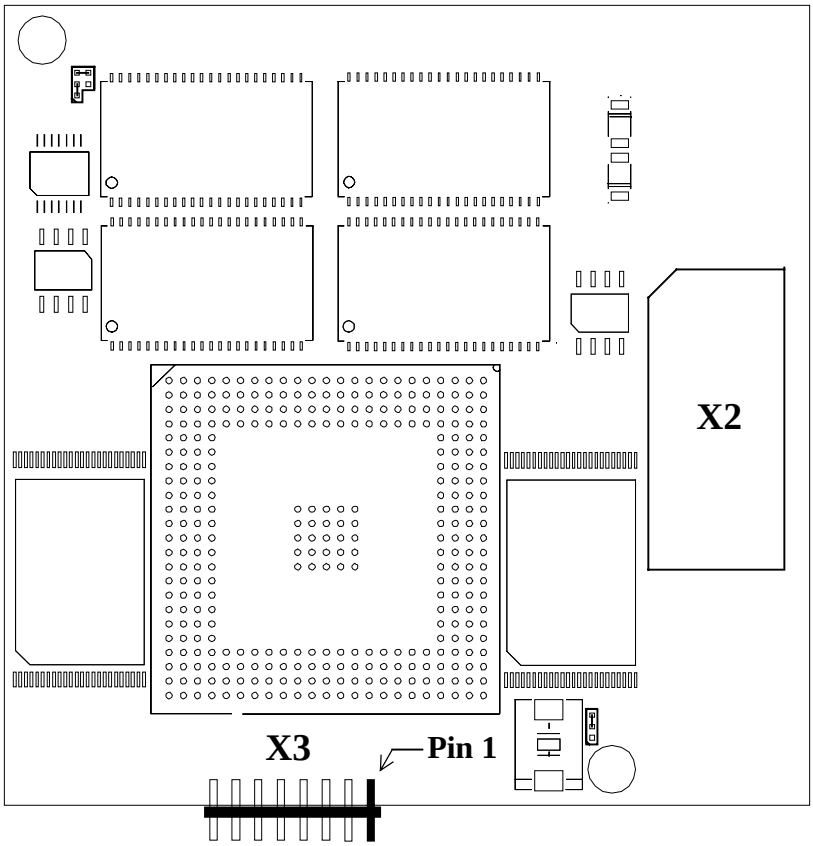


Figure 9: Positions of the Additional Components on the debugCORE-TC1775

12.2 Connecting an Emulator

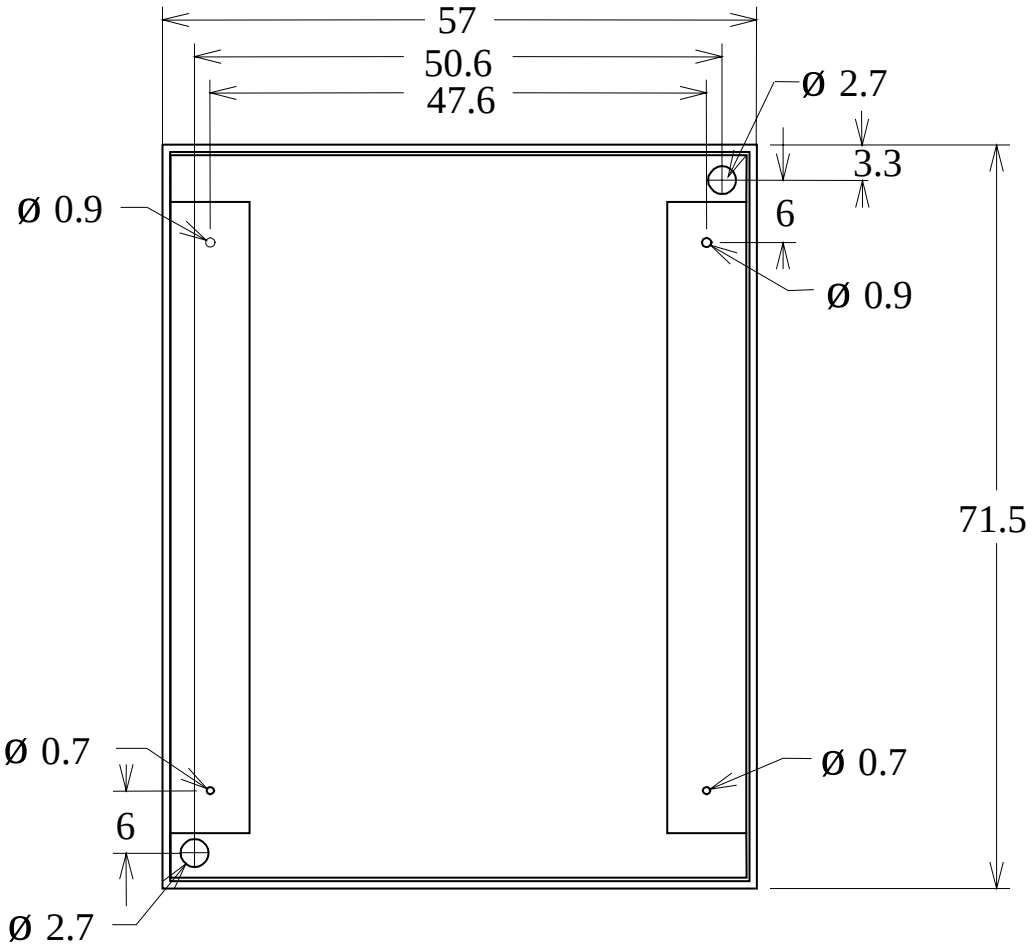
The 2 mm pin header connector at X3 is used for connection of an emulator, which is designed for use with the internal JTAG interface (OCDS level 1) of the TriCore-TC1775 controller. It is then possible to transfer program code to the module and debug this code with the help of standard debug functions such as breakpoints, single step, etc. The signals available at X3 are connected directly from the processor. No jumper configurations are required.

The trace signals (Port 5) of the TC1775 extend to the 40-pin connector at X2. Debugging on a 'deeper' level is possible with the use of these signals, since the trace information provides information about the internal processes of the CPU. The trace port is connected to the emulator in addition to the JTAG port. If your emulator is not equipped with a trace input, it is usually possible to obtain this trace port as an expansion from your emulator manufacturer.

Please note that when using the Trace port (OCDS2) port 5 of the controllers is no longer available for other I/O functions.

13 Technical Specifications

The physical dimensions of the phyCORE-TC1775 are represented in *Figure 10*. The module's profile is ca. 6 mm thick, with a maximum component height of 2.0 mm on the backside of the PCB and approximately 2.8 mm on the front side. The board itself is approximately 1.2 mm thick.



Measurements in mm

Figure 10: *Physical Dimensions*

Preliminary technical specifications:

• Dimensions:	72 mm x 57 mm
• Weight:	approximately 26 g with all optional components mounted on the circuit board
• Storage temperature:	-40°C to +90°C
• Operating temperature:	standard: 0°C to +70°C
• Humidity:	95 % r.F. not condensed
• Operating voltages:	2.5 V + 10 % 3.3 V ± 10 % 5 V ±5 %, VBAT 3 V ±20 %
• Power consumption:	Conditions: 40 MHz clock, 1 MByte RAM , 4 MByte Flash, 20°C
2.5 V voltage	typ. 350 mA
3.3 V voltage	typ. 400 mA
5 V voltage	typ. 200 mA
Battery current draw Real-Time Clock supply	Conditions: VBTA = 2.5 V, 2.5 V voltage=off, 3.3 V voltage=off, 5 V voltage=off, 20°C TBD µA

These specifications describe the standard configuration of the phyCORE-TC1775 as of the printing of this manual.

Connectors on the phyCORE-TC1775:

Manufacturer	Molex
Number of pins per connector row	160 (2 rows of 80 pins each)
Molex type number	52760 (receptacle)

Two different heights are offered for the receptacle sockets that correspond to the connectors populating the underside of the phyCORE-TC1775. The given connector height indicates the distance between the two connected PCBs when the module is mounted on the corresponding carrier board. In order to get the exact spacing, the maximum component height (2 mm) on the underside of the phyCORE must be subtracted.

- Height 6 mm

Manufacturer	Molex
Number of pins per connector row	160 (2 rows of 80 pins each)
Molex type number	55091 (plug)

- Height 10 mm

Manufacturer	Molex
Number of pins per connector row	160 (2 rows of 80 pins each)
Molex type number	53553 (plug)

Please refer to the coresponding data sheets and mechanical specifications provided by Molex (www.molex.com).

14 Hints for Handling the phyCORE-TC1775

Removal of various components, such as the microcontroller and the standard quartz, 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 while desoldering. 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.

Integrating the phyCORE-TC1775 in application circuitry

Successful integration in user target circuitry depends on whether the layout for the GND connections matches those of the phyCORE module. It is recommended that the target application circuitry is equipped with one layer dedicated to carry the GND potential. In any case, be sure to connect all GND pins neighboring signals which are used in the application circuitry. For the supply voltage, there must be contact with at least six of the GND pins neighboring the supply voltage pins.

15 The phyCORE-TC1775 on the phyCORE-TriCORE Development Board

PHYTEC Development Boards are fully equipped with all mechanical and electrical components necessary for the speedy and secure start-up and subsequent communication to and programming of applicable PHYTEC Single Board Computer (SBC) modules. Development Boards are designed for evaluation, testing and prototyping of PHYTEC Single Board Computers in labratory environments prior to their use in customer designed applications.

15.1 Concept of the phyCORE Development Board

The phyCORE-TriCORE Development Board provides a flexible development platform enabling quick and easy start-up and subsequent programming of the phyCORE-TC1775 Single Board Computer module. The Development Board design allows easy connection of additional expansion boards featuring various functions that support fast and convenient prototyping and software evaluation.

This modular development platform concept is depicted in *Figure 11* and includes the following components:

- The actual **Development Board** (1), which offers all essential components and connectors for start-up including: a power socket enabling connection to an **external power adapter** (2) and **serial interfaces** (3) of the SBC module at DB-9 connectors (depending on the module, up to two RS-232 interfaces and up to two RS-485 or CAN interfaces).
- All of the signals from the SBC module mounted on the Development Board extend to two mating receptacle connectors. A strict 1:1 signal assignment is consequently maintained from the phyCORE-connectors on the module to these expansion connectors. Accordingly, the pin assignment of the **expansion bus** (4) depends entirely on the pinout of the SBC module mounted on the Development Board.

- As the physical layout of the expansion bus is standardized across all applicable PHYTEC Development Boards, we are able to offer various **expansion boards** (5) that attach to the Development Board at the expansion bus connectors. These modular expansion boards offer **supplemental I/O functions** (6) as well as peripheral support devices for specific functions offered by the controller populating the **SBC module** (9) mounted on the Development Board.
- All controller and on-board signals provided by the SBC module mounted on the Development Board are broken out 1:1 to the expansion board by means of its **patch field** (7). The required connections between SBC module / Development Board and the expansion board are made using **patch cables** (8) included with the expansion board.

Figure 11 illustrates the modular development platform concept:

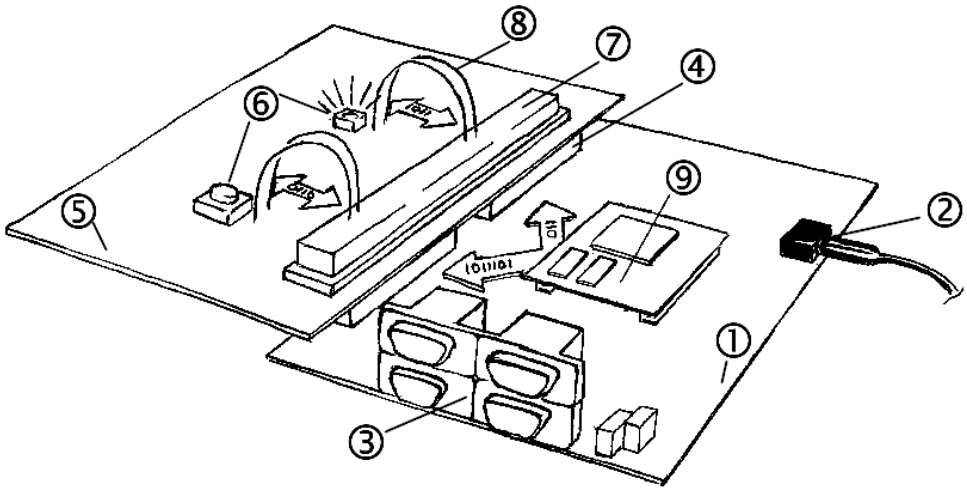


Figure 11: Modular Development and Expansion Board Concept with the phyCORE-TC1775

The following sections contain specific information relevant to the operation of the phyCORE-TC1775 mounted on the phyCORE-TriCORE Development Board. For a general description of the Development Board, please refer to the corresponding Development Board Hardware Manual.

15.2 phyCORE-TriCORE Development Board Connectors and Jumpers

15.2.1 Connectors

As shown in *Figure 12*, the following connectors are available on the phyCORE-TriCORE Development Board:

X1- low-voltage socket for power supply connectivity
X2- mating receptacle for expansion board connectivity
P1- dual DB-9 sockets for serial RS-232 interface connectivity
P2- dual DB-9 connectors for CAN interface connectivity
X4- voltage supply for external devices and subassemblies
X5- GND connector (for connection of GND signal of measuring devices such as an oscilliscope)
X3- phyCORE-connector enabling mounting of applicable phyCORE modules
P3, X6 - connectors for on-board Wiggler
X7 - RJ45 socket for connection to 10 Mbit TP Ethernet cable
XK1 - screw clamp for J1859 bus power supply
U9/U10- space for an optional silicon serial number chip

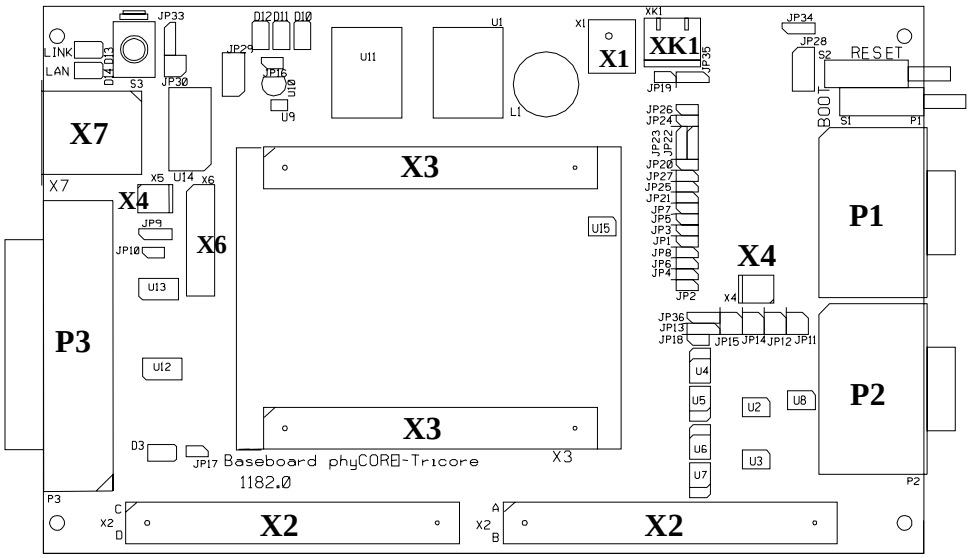


Figure 12: Location of Connectors on the phyCORE-TriCORE Development Board

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 User's Manual/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.

15.2.2 Jumpers on the phyCORE-TriCORE Development Board

Peripheral components of the phyCORE-TriCORE Development Board can be connected to the signals of the phyCORE-TC1775 by setting the applicable jumpers.

The Development Board's peripheral components are configured for use with the phyCORE-TC1775 by means of insertable jumpers. If no jumpers are set, no signals connect to the DB-9 connectors, the control and display units and the CAN transceivers. *Figure 13* illustrates the numbering of the jumper pads, while *Figure 14* indicates the location of the jumpers on the Development Board.

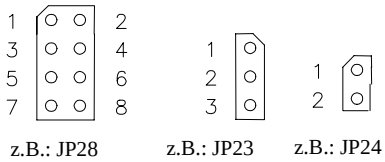


Figure 13: Numbering of Jumper Pads

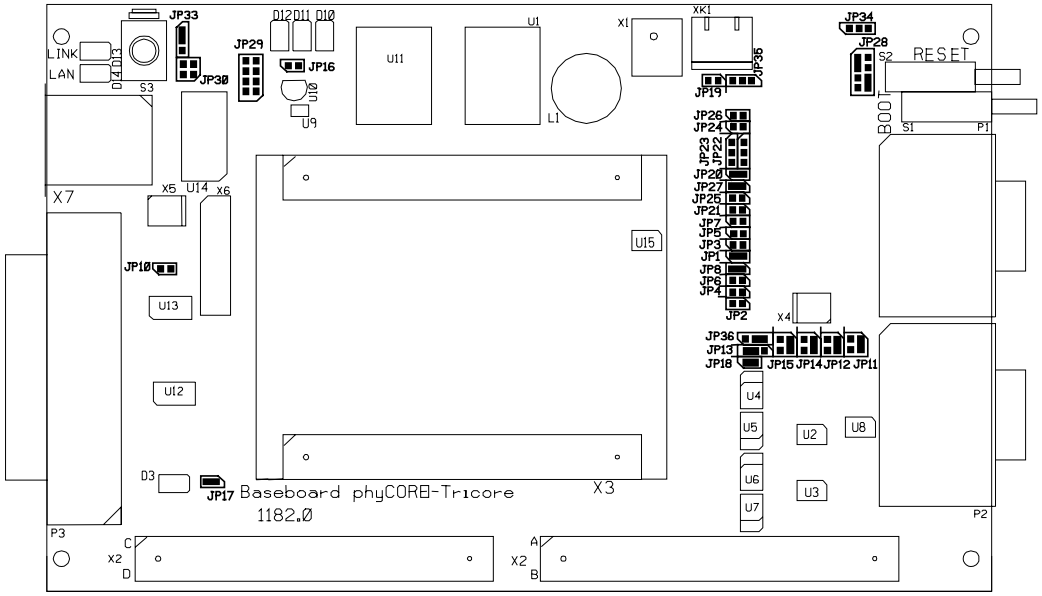


Figure 14: Location of the Jumpers (View of the Component Side)

Figure 15 shows the factory default jumper settings for operation of the phyCORE-TriCORE Development Board with the standard phyCORE-TC1775 (use of two RS-232 interfaces, LED D3, the Boot button etc. on the Development Board). Jumper settings for other functional configurations of the phyCORE-TC1775 module mounted on the Development Board are described in *section 15.3*.

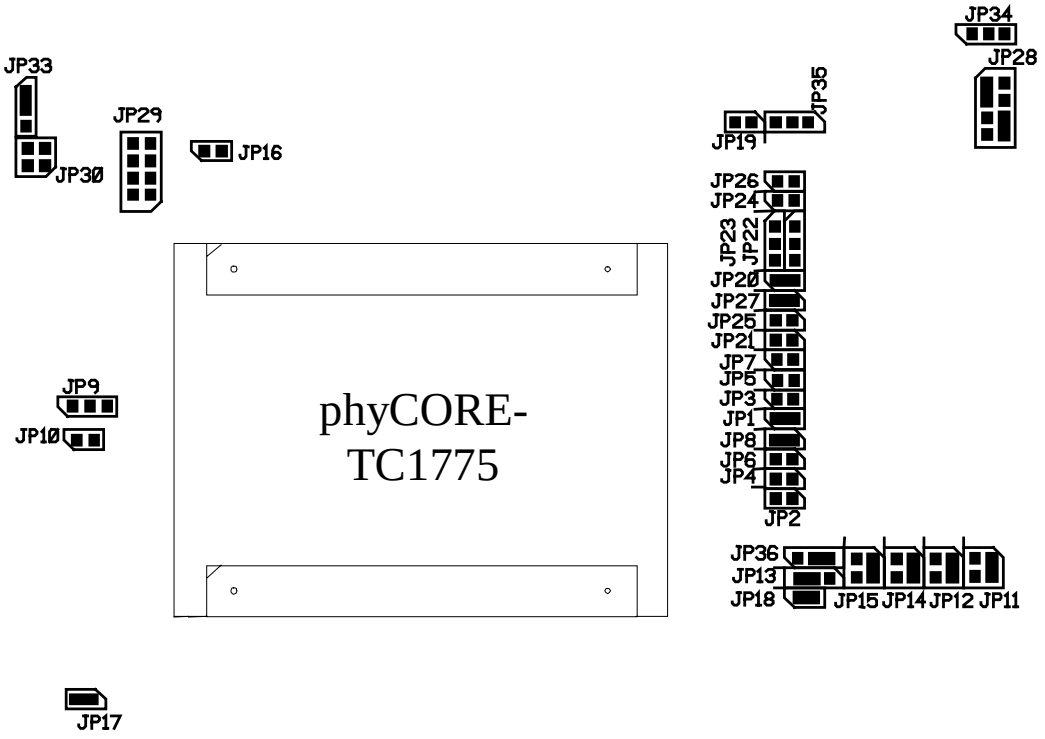


Figure 15: Default Jumper Settings of the phyCORE-TriCORE Development Board with Standard phyCORE-TC1775

15.3 Functional Components on the phyCORE-TriCORE Development Board

This section describes the functional components of the phyCORE-TriCORE Development Board supported by the phyCORE-TC1775 and appropriate jumper settings to activate these components. Depending on the specific configuration of the phyCORE-TC1775 module, alternative jumper settings can be used. These jumper settings are different from the factory default settings as shown in *Figure 15* and enable alternative or additional functions on the phyCORE-TriCORE Development Board depending on user needs.

15.3.1 Power Supply at X1

Caution:
Do not use a laboratory adapter to supply power to the Development Board! Power spikes during power-on could destroy the phyCORE module mounted on the Development Board! Do not change modules or jumper settings while the Development Board is supplied with power!

Permissible input voltage: +/-5 VDC regulated.

The required current load capacity of the power supply depends on the specific configuration of the phyCORE-TC1775 mounted on the Development Board as well as whether an optional expansion board is connected to the Development Board. An adapter with a minimum supply of 1.5 A is recommended.

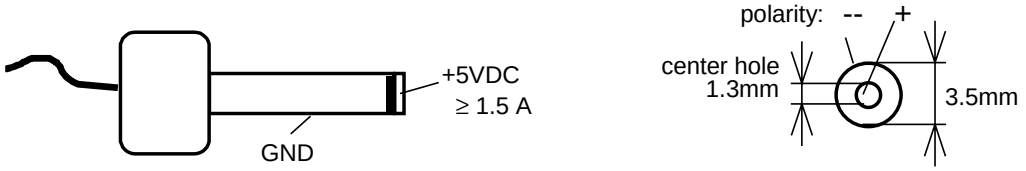


Figure 16: Connecting the Supply Voltage at X1

15.3.2 Activating the Bootstrap Loader

The Infineon Tricore-TC1775 microcontroller contains an on-chip Bootstrap Loader that provides basic communication and programming functions. The combination of this Bootstrap Loader and the corresponding Flash download software installed on the PC allows for Flash programming with application code via an RS-232 interface.

In order to start the on-chip Bootstrap Loader on the phyCORE-TC1775, the configuration inputs CFG0-4 of the microcontroller must provide a certain bit pattern at the time the Reset signal changes from its active to the inactive state. This bit pattern is created by solder jumpers on the module and pass through to the controller with the help of an electronic switch at U18. Applying a low-level signal at pin X1C9 of the phyCORE-TC1775 (via the Boot input) activates the electronic switch.

The phyCORE-TriCORE Development Board provides two different options to activate the on-chip Bootstrap Loader:

1. The Boot button (S1) can be connected to GND via Jumper JP28 which is located next the the Boot and Reset buttons at S1 and S2. This configuration enables start-up of the on-chip Bootstrap Loader if the Boot button is pressed during a hardware reset or power-on.

Jumper	Setting	Description
JP28	1 + 3 and 6 + 8	Boot button (in conjunction with Reset button or connection of the power supply) starts the Bootstrap Loader on the TC1775

Table 9: JP28 Configuration of the Boot Button

2. It is also possible to start the Bootstrap Loader via external signals applied to the DB-9 socket P1A. This requires control of the signal transition on the Reset line via pin 7 while a static low-level is applied to pin 4 for the Boot signal.

Jumper	Setting	Description
JP22	2 + 3	Pin 7 (CTS) of the DB-9 socket P1A as Reset signal for the phyCORE-TC1775
JP23	2 + 3	Pin 4 (DSR) of the DB-9 socket P1A as Boot signal for the phyCORE-TC1775
JP34	1 + 2	Low-level Boot signal connected with the Boot input of the phyCORE-TC1775

Table 10: JP22, JP23, JP34 Configuration of Boot via RS-232

Caution:
When using this function, the following jumper setting is not allowed:

Jumper	Setting	Description
JP34	1 + 2	Jumper setting generates high-level on Boot input of the phyCORE-TC1775

Table 11: Improper Jumper Settings for Boot via RS-232

15.3.3 First Serial Interface at Socket P1A

Socket P1A is the lower socket of the double DB-9 connector at P1. P1A is connected via jumpers to the first serial interface of the phyCORE-TC1775. When connected to a host-PC, the phyCORE-TC1775 can be rendered in Bootstrap mode via signals applied to the socket P1A (*refer to section 15.3.2*).

Jumper	Setting	Description
JP20	closed	Pin 2 of DB-9 socket P1A connected with RS-232 interface signal TxD0 of the phyCORE-TC1775
JP21	open	Pin 9 of DB-9 socket P1A not connected
JP22	open	Pin 7 of DB-9 socket P1A not connected
	2 + 3[1]	Reset input of the module can be controlled via RTS signal from a host-PC
JP23	open	Pin 4 of DB-9 socket P1A not connected
	2 + 3[1]	Boot input of the module can be controlled via DTR signal from a host-PC (**Note**: JP34 must be set to position 1 + 2)
JP24	open	Pin 6 of DB-9 socket P1A not connected
JP25	open	Pin 8 of DB-9 socket P1A not connected
JP26	open	Pin 1 of DB-9 socket P1A not connected
JP27	closed	Pin 3 of DB-9 socket P1A connected with RS-232 interface signal RxD0 from the phyCORE-TC1775

Table 12: Jumper Configuration for the First RS-232 Interface

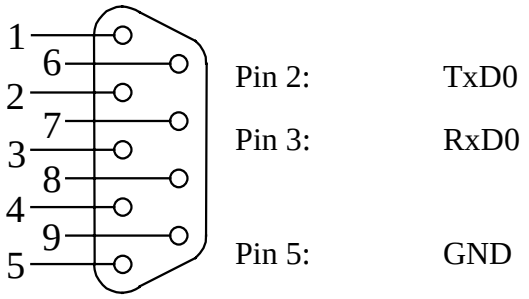


Figure 17: Pin Assignment of the DB-9 Socket P1A as First RS-232 (Front View)

[1]: Alternative jumper configuration for additional features (*refer to section 15.3.2*). Not required for standard communication functions.

Caution:

When using the DB-9 socket P1A as RS-232 interface on the phyCORE-TC1775 the following jumper settings are not functional and could damage the module:

Jumper	Setting	Description
JP21	closed	Pin 9 of DB-9 socket P1A connected with the SCLK0 signal from phyCORE-TC1775
JP22	1 + 2	Pin 7 of DB-9 socket P1A connected with port P13.4 from phyCORE-TC1775
JP23	1 + 2	Pin 4 of DB-9 socket P1A connected with the MRST0 signal from phyCORE-TC1775
JP24	closed	Pin 6 of DB-9 socket P1A connected with the MTSR0 signal from phyCORE-TC1775
JP25	closed	Pin 8 of DB-9 socket P1A connected with port P13.3 from phyCORE-TC1775
JP26	closed	Pin 1 of DB-9 socket P1A connected with port P13.5 from phyCORE-TC1775

Table 13: Improper Jumper Settings for DB-9 Socket P1A as First RS-232

If an RS-232 cable is connected to P1A, the voltage level on the RS-232 lines could destroy the phyCORE-TC1775.

15.3.4 Second Serial Interface at Socket P1B

Socket P1B is the upper socket of the double DB-9 connector at P1. P1B is connected via jumpers to the second serial interface of the phyCORE-TC1775.

Jumper	Setting	Description
JP1	closed	Pin 2 of DB-9 socket P1B connected with RS-232 interface signal TxD1 of the phyCORE-TC1775
JP2	open	Pin 9 of DB-9 socket P1B not connected
JP3	open	Pin 7 of DB-9 socket P1B not connected
JP4	open	Pin 4 of DB-9 socket P1B not connected
JP5	open	Pin 6 of DB-9 socket P1B not connected
JP6	open	Pin 8 of DB-9 socket P1B not connected
JP7	open	Pin 1 of DB-9 socket P1B not connected
JP8	closed	Pin 3 of DB-9 socket P1B connected with RS-232 interface signal RxD1 from the phyCORE-TC1775

Table 14: Jumper Configuration for the Second RS-232 Interface

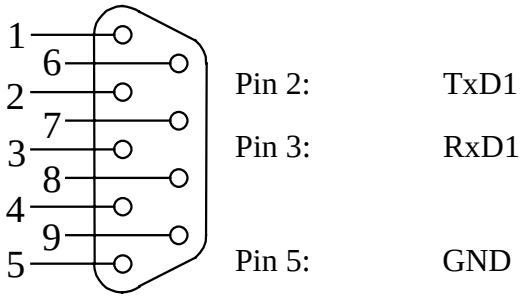


Figure 18: Pin Assignment of the DB-9 Socket P1A as Second RS-232 (Front View)

Caution:

When using the DB-9 socket P1B as RS-232 interface on the phyCORE-TC1775 the following jumper settings are not functional and could damage the module:

Jumper	Setting	Description
JP2	closed	Pin 9 of DB-9 socket P1B connected with the SCLK1 signal from phyCORE-TC1775
JP3	closed	Pin 7 of DB-9 socket P1B connected with the TxJ1850 signal from phyCORE-TC1775
JP4	closed	Pin 4 of DB-9 socket P1B connected with the MRST1 signal from phyCORE-TC1775
JP5	closed	Pin 6 of DB-9 socket P1B connected with the MTSR1 signal from phyCORE-TC1775
JP6	closed	Pin 8 of DB-9 socket P1B connected with the RxJ1850 signal from phyCORE-TC1775
JP7	closed	Pin 1 of DB-9 socket P1B connected with the CLKOUT_RTC signal from phyCORE-TC1775

Table 15: Improper Jumper Settings for DB-9 Socket P1A as Second RS-232

If an RS-232 cable is connected to P1A, the voltage level on the RS-232 lines could destroy the phyCORE-TC1775.

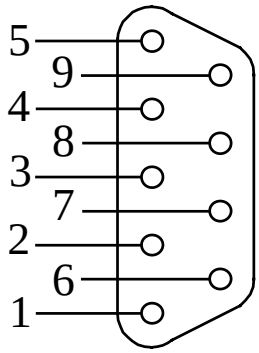
15.3.5 First CAN Interface at Plug P2A

Plug P2A is the lower plug of the double DB-9 connector at P2. P2A is connected to the first CAN interface (CAN0) of the phyCORE-TC1775 via jumpers. Depending on the configuration of the CAN transceivers and their power supply, the following three configurations are possible:

1. CAN transceiver populating the phyCORE-TC1775 is enabled and the CAN signals from the module extend directly to plug P2A.

Jumper	Setting	Description
JP11	1 + 2	Pin 7 of DB-9 plug P2A connected with CAN_H0 from on-board transceiver on the phyCORE-TC1775
JP12	1 + 2	Pin 2 of DB-9 plug P2A connected with CAN_L0 from on-board transceiver on the phyCORE-TC1775
JP13	open	No supply voltage to CAN transceiver and opto-coupler on the phyCORE-TriCORE Development Board
JP18	open	No GND potential at CAN transceiver and opto-coupler on the phyCORE-TriCORE Development Board

Table 16: Jumper Configuration for CAN Plug P2A Using the CAN Transceiver on the phyCORE-TC1775



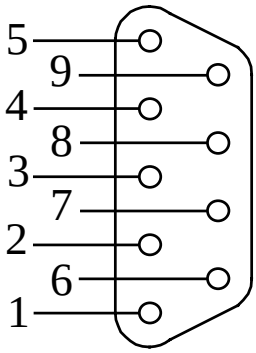
Pin 3:	GND	(Development Board Ground)
Pin 7:	CAN-H0	(not galvanically separated)
Pin 2:	CAN-L0	(not galvanically separated)
Pin 6:	GND	(Development Board Ground)

Figure 19: Pin Assignment of the DB-9 Plug P2A (CAN Transceiver on phyCORE-TC1775, Front View)

2. The CAN transceiver populating the phyCORE-TC1775 is disabled; CAN signals generated by the CAN transceiver (U2) on the Development Board extending to connector P2A **without galvanic seperation:**

Jumper	Setting	Description
JP11	1 + 3 2 + 4	Pin 7 of DB-9 plug P2A connected with CAN-H0 from CAN transceiver U2 on the Development Board
JP12	1 + 3 2 + 4	Pin 2 of DB-9 plug P2A connected with CAN-L0 from CAN transceiver U2 on the Development Board
JP13	2 + 3	Supply voltage for CAN transceiver and opto-coupler derived from local supply circuitry on the phyCORE-TriCORE Development Board
JP18	closed	CAN transceiver and opto-coupler on the Development Board connected with local GND potential

Table 17: Jumper Configuration for CAN Plug P2A Using the CAN Transceiver on the Development Board



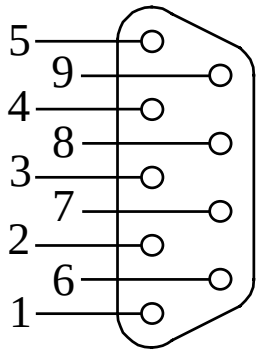
Pin 3:	GND	(Development Board Ground)
Pin 7:	CAN-H0	(not galvanically separated)
Pin 2:	CAN-L0	(not galvanically separated)
Pin 6:	GND	(Development Board Ground

Figure 20: Pin Assignment of the DB-9 Plug P2A (CAN Transceiver on Development Board)

3. The CAN transceiver populating the phyCORE-TC1775 is disabled; CAN signals generated by the CAN transceiver (U2) on the Development Board extend to connector P2A **with galvanic separation.** This configuration requires connection of an external CAN supply voltage of 7 to 13 V. The external power supply must be **only** connected to either P2A **or** P2B.

Jumper	Setting	Description
JP11	1 + 3 2 + 4	Pin 7 of DB-9 plug P2A connected with CAN-H0 from CAN transceiver U2 on the Development Board
JP12	1 + 3 2 + 4	Pin 2 of DB-9 plug P2A connected with CAN-L0 from CAN transceiver U2 on the Development Board
JP13	1 + 2	Supply voltage for CAN transceiver and opto-coupler on the Development Board derived from external source (CAN bus) via on-board voltage regulator
JP18	open	CAN transceiver and opto-coupler on the Development Board disconnected from local GND potential

Table 18: Jumper Configuration for CAN Plug P2A Using the CAN Transceiver on the Development Board with Galvanic Separation



Pin 9: VCAN+

Pin 3: VCAN-
Pin 7: CAN-H0 (galvanically separated)
Pin 2: CAN-L0 (galvanically separated)
Pin 6: VCAN-

Figure 21: Pin Assignment of the DB-9 Plug P2A (CAN Transceiver on Development Board with Galvanic Separation)

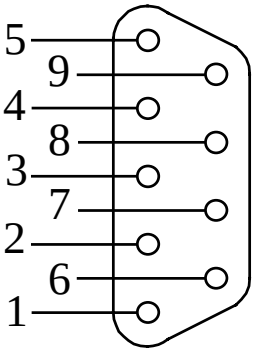
15.3.6 Second CAN Interface at Plug P2B

Plug P2B is the upper plug of the double DB-9 connector at P2. P2B is connected to the second CAN interface (CAN1) of the phyCORE-TC1775 via jumpers. Depending on the configuration of the CAN transceivers and their power supply, the following three configurations are possible:

1. CAN transceiver populating the phyCORE-TC1775 is enabled and the CAN signals from the module extend directly to plug P2B.

Jumper	Setting	Description
JP14	1 + 2	Pin 7 of the DB-9 plug P2B is connected to CAN-H1 from on-board transceiver on the phyCORE module
JP15	1 + 2	Pin 2 of the DB-9 plug P2B is connected to CAN-L1 from on-board transceiver on the phyCORE module
JP13	open	CAN transceiver and opto-coupler on the Development Board disconnected from supply voltage
JP18	open	No GND potential at CAN transceiver and opto-coupler on the phyCORE-TriCORE Development Board
JP36	2 + 3	Pin 3 of the DB-9 plug P2B is connected to GND

Table 19: Jumper Configuration for CAN Plug P2B Using the CAN Transceiver on the phyCORE-TC1775



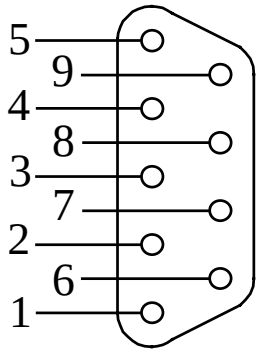
Pin 8:	GND	(Development Board Ground)
Pin 3:	GND	(Development Board Ground)
Pin 7:	CAN-H1	(not galvanically separated)
Pin 2:	CAN-L1	(not galvanically separated)
Pin 6:	GND	(Development Board Ground)

Figure 22: Pin Assignment of the DB-9 Plug P2B (CAN Transceiver on phyCORE-TC1775, Front View)

2. The CAN transceiver populating the phyCORE-TC1775 is disabled; CAN signals generated by the CAN transceiver (U3) on the Development Board extending to connector P2B **without galvanic seperation:**

Jumper	Setting	Description
JP14	1 + 3 2 + 4	Pin 7 of DB-9 plug P2B connected with CAN-H1 from CAN transceiver U3 on the Development Board
JP15	1 + 3 2 + 4	Pin 2 of DB-9 plug P2B connected with CAN-L1 from CAN transceiver U3 on the Development Board
JP13	2 + 3	Supply voltage for CAN transceiver and opto-coupler derived from local supply circuitry on the phyCORE-TriCORE Development Board
JP18	closed	CAN transceiver and opto-coupler on the Development Board connected with local GND potential
JP36	2 + 3	Pin 3 of the DB-9 plug P2B is connected to GND

Table 20: Jumper Configuration for CAN Plug P2B Using the CAN Transceiver on the Development Board



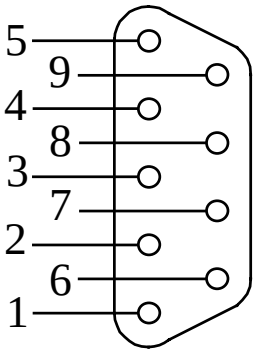
Pin 8:	GND	(Development Board Ground)
Pin 3:	GND	(Development Board Ground)
Pin 7:	CAN-H1	(not galvanically separated)
Pin 2:	CAN-L1	(not galvanically separated)
Pin 6:	GND	(Development Board Ground

Figure 23: Pin Assignment of the DB-9 Plug P2B (CAN Transceiver on Development Board)

3. The CAN transceiver populating the phyCORE-TC1775 is disabled; CAN signals generated by the CAN transceiver (U3) on the Development Board extend to connector P2B **with galvanic separation.** This configuration requires connection of an external CAN supply voltage of 7 to 13 V. The external power supply must be **only** connected to either P2A **or** P2B.

Jumper	Setting	Description
JP14	1 + 3 2 + 4	Pin 7 of DB-9 plug P2B connected with CAN-H1 from CAN transceiver U3 on the Development Board
JP15	1 + 3 2 + 4	Pin 2 of DB-9 plug P2B connected with CAN-L1 from CAN transceiver U3 on the Development Board
JP13	1 + 2	Supply voltage for CAN transceiver and opto-coupler on the Development Board derived from external source (CAN bus) via on-board voltage regulator
JP18	open	CAN transceiver and opto-coupler on the Development Board disconnected from local GND potential
JP36	2 + 3	Pin 3 of the DB-9 plug P2B is connected to GND

Table 21: Jumper Configuration for CAN Plug P2B Using the CAN Transceiver on the Development Board with Galvanic Separation



Pin 9: VCAN+

Pin 8: VCAN-
Pin 3: VCAN-
Pin 7: CAN-H1 (galvanically separated)
Pin 2: CAN-L1 (galvanically separated)
Pin 6: VCAN-

Figure 24: Pin Assignment of the DB-9 Plug P2B (CAN Transceiver on Development Board with Galvanic Separation)

15.3.7 J1850 Interface

The phyCORE-TriCORE Development Board features an HIP7020 transceiver device at U15 which allows direct use of the J1850 interface provided by the TC1775 controller. Jumper JP35 selects the supply voltage source for this transceiver. The transceiver can be supplied by either a 5 V supply voltage or with a supply voltage (6-24 VDC) directed over the screw clamp XK1. Please note that the 5 V supply voltage is not within the transceiver's specifications and should therefore only be used for experimentation purposes.

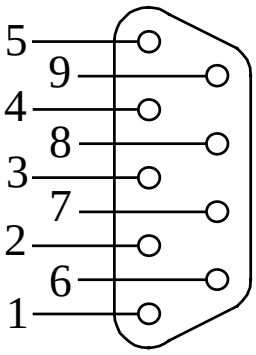
Jumper JP36 must be closed at position 2+3 so that the J1850 bus signal is available on the DB-9 plug P2B.

Caution:
An external supply of the CAN optocoupler via the DB-9 plug is not allowed when using the J1850 bus (*refer to section 7.2*).

For diagnostic purposes, the transceiver HIP7020 offers the possibility of activating a loop-back mode. Closing Jumper JP19 activates this mode. *More detailed information on the loop-back mode can be found in the HIP7020 data sheet.*

Jumper	Setting	Description
JP19	closed	Loop-back mode enabled
JP36	1 + 2	J1850 bus signal available at pin 3 on P2B
JP35	2 + 3	J1850 transceiver supplied via on-board 5 V supply voltage
	1 + 2	J1850 transceiver supplied via voltage applied at XK1 screw clamp (6-24 VDC)
JP18	closed	Pin 6 of DB-9 plug P2B connected with GND

Table 22: Jumper Configuration for Plug P2B with J1850 Interface



Pin 3: J1850

Pin 6: GND

Figure 25: Pin Assignment of the DB-9 Plug P2B for J1850 Interface

Caution:
When using the DB-9 socket P2B as J1850 interface on the phyCORE-TC1775 the following jumper settings are not functional and could damage the module:

Jumper	Setting	Description
JP14	1 + 2 3 + 4	Pin 7 of DB-9 plug P2B connected with the CAN_H1 signal from phyCORE-TC1775
JP19	1 + 2 3 + 4	Pin 2 of DB-9 plug P2B connected with the CAN_L1 signal from phyCORE-TC1775
JP18	open	Pin 6 of DB-9 plug P2B not connected

Table 23: Improper Jumper Settings for DB-9 Plug P2B as J1850

15.3.8 RJ45 Ethernet Connector X7

The phyCORE-TriCORE Development Board provides the Ethernet transformer device at U14 and the RJ45 jack at X7 to enable immediate connection of the phyCORE-TC1775 to an 10 Mbit/s Ethernet network. Two status LEDs for LINK and LAN are provided to display network status. The following jumper settings are required:

Jumper	Setting	Description
JP29	1 + 2 3 + 4 5 + 6 7 + 8	The Ethernet transformer module is connected to the Ethernet signals on the phyCORE-TC1775
JP30	1 + 2 3 + 4	LINK and LAN LEDs are activated

Table 24: JP29, JP30 Configuration of the Ethernet Interface

15.3.9 Programmable LED D3

The phyCORE-TriCORE Development Board offers a programmable LED at D3 for user implementations. This LED can be connected to port pin P10.0 of the phyCORE-TC1775 which is available via signal GPIO0 (JP17 = closed). A low-level at port pin P10.0 causes the LED to illuminate, LED D3 remains off when writing a high-level to P10.0.

Jumper	Setting	Description
JP17	closed	Port pin P10.0 (GPIO0) of the TC1775 controller controls LED D3 on the Development Board

Table 25: JP17 Configuration of the Programmable LED D3

15.3.10 Pin Assignment Summary of the phyCORE, the Expansion Bus and the Patch Field

As described in *section 15.1*, all signals from the phyCORE-TC1775 extend in a strict 1:1 assignment to the Expansion Bus connector X2 on the Development Board. These signals, in turn, are routed in a similar manner to the patch field on an optional expansion board that mounts to the Development Board at X2.

Please note that, depending on the design and size of the expansion board, only a portion of the entire patch field is utilized under certain circumstances. When this is the case, certain signals described in the following section will not be available on the expansion board. However, the pin assignment scheme remains consistent.

A two dimensional numbering matrix similar to the one used for the pin layout of the phyCORE-connector is provided to identify signals on the Expansion Bus connector (X2 on the Development Board) as well as the patch field.

However, the numbering scheme for Expansion Bus connector and patch field matrices differs from that of the phyCORE-connector, as shown in the following two figures:

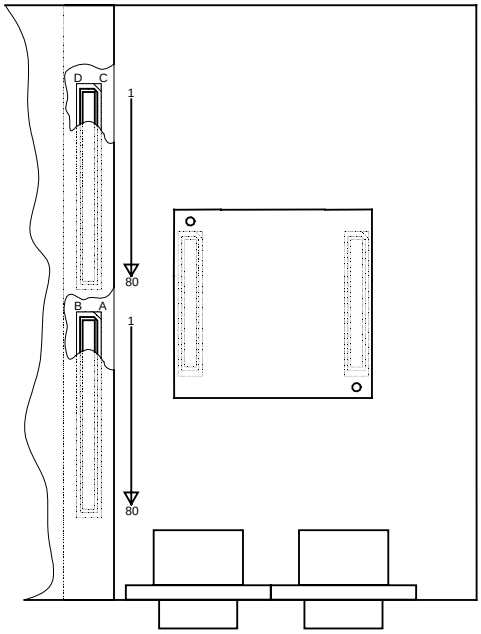


Figure 26: Pin Assignment Scheme of the Expansion Bus

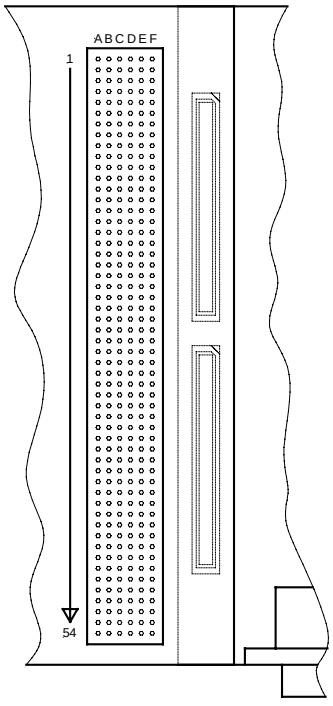


Figure 27: Pin Assignment Scheme of the Patch Field

The pin assignment on the phyCORE-TC1775, in conjunction with the Expansion Bus (X2) on the Development Board and the patch field on an expansion board, is as follows:

Signal	phyCORE-TC1775	Expansion Bus	Patch Field
D0	18B	18B	33F
D1	19A	19A	34A
D2	20A	20A	34E
D3	20B	20B	34B
D4	21A	21A	34D
D5	21B	21B	34F
D6	22B	22B	35A
D7	23A	23A	35E
D8	28B	28B	37C
D9	29A	29A	37E
D10	30A	30A	37B
D11	30B	30B	37F
D12	31A	31A	38A
D13	31B	31B	38C
D14	32B	32B	38E
D15	33A	33A	38B
D16	37B	37B	40A
D17	38A	38A	40E
D18	38B	38B	40B
D19	39A	39A	40D
D20	40A	40A	40F
D21	40B	40B	41A
D22	41A	41A	41E
D23	41B	41B	41B
D24	42B	42B	41F
D25	43A	43A	42A
D26	43B	43B	42C
D27	44A	44A	42E
D28	45A	45A	42B
D29	45B	45B	42F
D30	46A	46A	43A
D31	46B	46B	43C

Table 26: Pin Assignment Data Bus for the phyCORE-TC1775 / Development Board / Expansion Board (Class B Pins Nominal 2.5 V, 3.3 V Tolerant)

Signal	phyCORE-TC1775	Expansion Bus	Patch Field
A0	8B	8B	30B
A1	9A	9A	30D
A2	10A	10A	20F
A3	10B	10B	31A
A4	11A	11A	31E
A5	11B	11B	31B
A6	12B	12B	31F
A7	13A	13A	31A
A8	13B	13B	32C
A9	14A	14A	32E
A10	15A	15A	32B
A11	15B	15B	32F
A12	16A	16A	33A
A13	16B	16B	33C
A14	17B	17B	33E
A15	18A	18A	33B
A16	23B	23B	35B
A17	24A	24A	35D
A18	25A	25A	35F
A19	25B	25B	36A
A20	26A	26A	36E
A21	26B	26B	36B
A22	27B	27B	36F
A23	28A	28A	37A
A24	47B	47B	43E
A25	48A	48A	43B

Table 27: Pin Assignment Address Bus for the phyCORE-TC1775 / Development Board / Expansion Board (Class B Pins Nominal 2.5 V, 3.3 V Tolerant)

Signal	phyCORE-TC1775	Expansion Bus	Patch Field
P3.10/ /CS3	5A	5A	29E
P3.11/ /CS2	5B	5B	29B
P3.12/ /CS1	48B	48B	43F
P3.13/ /CS0	6B	6B	29F
P3.14/ /CSEMUL	50A	50A	44E
P3.15/ /CSOVL	51A	51A	44D
P4.0/ /RD	7B	7B	30A
P4.1/ RD//WR	49A	49A	44A
P4.2/ ALE	6A	6A	29D
P4.3/ /ADV	50B	50B	44B
P4.4/ BC0	8A	8A	30E
P4.5/ BC1	33B	33B	38F
P4.6/ /BC2	52B	52B	45A
P4.7/ BC3	53B	53B	45B
P4.8/ /WAIT//IND	34A	34A	39A
P4.9/ /BAA	51B	51B	44F
P4.10/ /CSFPI	53A	53A	45E
P4.11/ /HOLD	35B	35B	39B
P4.12/ /HLDA	36A	36A	39D
P4.13/ /BREQ	36B	36B	39F
P4.14/ /CODE	54A	54A	45D
P4.15/ SVM	55A	55A	45F
P5.0	53C	53C	18A
P5.1	52D	52D	17F
P5.2	51D	51D	17D
P5.3	51C	51C	17B
P5.4	50D	50D	17E
P5.5	50C	50C	17A
P5.6	49C	49C	16F
P5.7	48D	48D	16B
P5.8	48C	48C	16E
P5.9	47D	47D	16C
P5.10	46D	46D	16A
P5.11	46C	46C	15F
P5.12	45D	45D	15B
P5.13	45C	45C	15E
P5.14	44C	44C	15C
P5.15	43D	43D	15A

Table 28: Pin Assignment Port P3, P4, P5 Dedicated Port Pins on the phyCORE-TC1775/ Development Board / Expansion Board (Class B Pins Nominal 2.5 V, 3.3 V Tolerant)

Signal	phyCORE-TC1775	Expansion Bus	Patch Field
P6.0/ AN0	80C	80C	27A
P6.1/ AN1	79C	79C	26F
P6.2/ AN2	78D	78D	26B
P6.3/ AN3	78C	78C	26E
P6.4/ AN4	77D	77D	26C
P6.5/ AN5	76D	76D	26A
P6.6/ AN6	76C	76C	25F
P6.7/ AN7	75D	75D	25B
P6.8/ AN8	75C	75C	25E
P6.9/ AN9	74C	74C	25C
P6.10/ AN10	73D	73D	25A
P6.11/ AN11	73C	73C	24F
P6.12/ AN12	72D	72D	24B
P6.13/ AN13	71D	71D	24E
P6.14/ AN14	71C	71C	24A
P6.15/ AN15	70D	70D	23F
P7.0/ AN16	69C	69C	23B
P7.1/ AN17	68D	68D	23E
P7.2/ AN18	68C	68C	23A
P7.3/ AN19	67D	67D	22F
P7.4/ AN20	66D	66D	22D
P7.5/ AN21	66C	66C	22B
P7.6/ AN22	65D	65D	22E
P7.7/ AN23	65C	65C	22A
P7.8/ AN24	64C	64C	21F
P7.9/ AN25	63D	63D	21B
P7.10/ AN26	63C	63C	21E
P7.11/ AN27	62D	62D	21C
P7.12/ AN28	61D	61D	21A
P7.13/ AN29	61C	61C	20F
P7.14/ AN30	60D	60D	20B
P7.15/ AN31	60C	60C	20E
Varef0	80D	80D	27E
Vagnd0	79D, 77C, 74D, 72C	79D, 77C, 74D, 72C	GND
Varef1	70C	70C	23D
Vagnd1	69D, 67C, 64D, 62C, 59D	69D, 67C, 64D, 62C, 59D	GND

Table 29: Pin Assignment Analog Ports P6, P7 for the phyCORE-TC1775 / Development Board / Expansion Board (Maximum Voltage 5 V)

Signal	phyCORE-TC1775	Expansion Bus	Patch Field
P8.0	38C	38C	13A
P8.1	37D	37D	12F
P8.8	80B	80B	54B
P8.9	80A	80A	54E
P8.10	79A	79A	54A
P8.11	78B	78B	53F
P8.12	78A	78A	53B
P8.13	77B	77B	53E
P8.14	76B	76B	53C
P8.15	76A	76A	53A
P9.0	75B	75B	52F
P9.1	75A	75A	52B
P9.2	74A	74A	52E
P9.3	73B	73B	52C
P9.4	73A	73A	52A
P9.5	72B	72B	51F
P9.6	71B	71B	51B
P9.7	71A	71A	51E
P9.8	70B	70B	51A
P9.9	70A	70A	50F
P9.10	69A	69A	50D
P9.11	68B	68B	50B
P9.12	68A	68A	50E
P9.13	67B	67B	50A
P9.14	66B	66B	49F
P9.15	66A	66A	49D

Table 30: Pin Assignment Analog Ports P8, P9 for the phyCORE-TC1775 / Development Board / Expansion Board (Class A Pins Nominal 3.0 V to 5.25 V)

Signal	phyCORE-TC1775	Expansion Bus	Patch Field
P10.0	11D	11D	4A
P10.1	12D	12D	4B
P10.2	13C	13C	4F
P10.3	13D	13D	5A
P10.4	14C	14C	5C
P10.5	15C	15C	5E
P10.6	15D	15D	5B
P10.7	16C	16C	5F
P10.8	43C	43C	14F
P10.9	42D	42D	14B
P10.10	41D	41D	14E
P10.11	41C	41C	14A
P10.12	40D	40D	13F
P10.13	40C	40C	13D
P10.14	39C	39C	13B
P10.15	38D	38D	13E
P11.0	65B	65B	49B
P11.1	65A	65A	49E
P11.2	64A	64A	49A
P11.3	63B	63B	48F
P11.4	63A	63A	48B
P11.5	62B	62B	48E
P11.6	61B	61B	48C
P11.7	61A	61A	48A
P11.8	60B	60B	47F
P11.9	60A	60A	47B
P11.10	59A	59A	47E
P11.11	58B	58B	47C
P11.12	58A	58A	47A
P11.13	57B	57B	46F
P11.14	56B	56B	46B
P11.15	56A	56A	46E

Table 31: Pin Assignment Analog Ports P10, P11 for the phyCORE-TC1775 / Development Board / Expansion Board (Class A Pins Nominal 3.0 V to 5.25 V)

Signal	phyCORE-TC1775	Expansion Bus	Patch Field
P12.0/ AD0EMUX0	59C	59C	20C
P12.1/ AD0EMUX1	58C	58C	19F
P12.2/ AD0EMUX2	58D	58D	20A
P12.3/ AD1EMUX0	56C	56C	19A
P12.4/ AD1EMUX1	55D	55D	18F
P12.5/ AD1EMUX2	55C	55C	18D
P12.6/ AD1EXTIN0	54C	54C	18B
P12.7/ AD1EXTIN1	53D	53D	18E
P12.8/ AD0EXTIN0	57D	57D	19B
P12.9/ AD0EXTIN1	56D	56D	19E
P12.10/ RxJ1850	24C	24C	8B
P12.11/ TxJ1850	25C	25C	8D
P12.12/ RxD0A_TTL	16D	16D	6A
P12.13/ TxD0A_TTL	17D	17D	6C
P12.14/ RxD1A_TTL	19C	19C	6F
P12.15/ TxD1A_TTL	20C	20C	7A
P13.0	2B	2B	28E
P13.1	3A	3A	28B
P13.2	3B	3B	28F
P13.3	25D	25D	8F
P13.4	26D	26D	9E
P13.5	31D	31D	11A
P13.6/ SCLK0	30D	30D	10B
P13.7/ MRST0	27D	27D	9B
P13.8/ MTSR0	28D	28D	10A
P13.9/ SCLK1	29C	29C	10C
P13.10/ MRST1	26C	26C	9A
P13.11/ MTSR1	28C	28C	9F
/HDRST	10C	10C	3D

Table 32: Pin Assignment Analog Ports P12, P13 for the phyCORE-TC1775 / Development Board / Expansion Board (Class A Pins Nominal 3.0 V to 5.25 V)

Signal	phyCORE-TC1775	Expansion Bus	Patchfeld
CAN_H0	21D		7D
CAN_L0	20D		7E
CAN_H1	18C		6E
CAN_L1	18D		6B
RXD0	22D		7F
TXD0	23D		8E
RXD1	21C		7B
TXD1	23C		8A
ETH_LinkLED	33C		11E
ETH_LanLED	34C		11F
ETH_RXD-	35C		12A
ETH_TXD-	36C		12B
ETH_RXD+	35D		12E
ETH_TXD+	36D		12D

Table 33: Pin Assignment Interface Signals for the phyCORE-TC1775 / Development Board / Expansion Board (Level According to Signal Type)

Signal	phyCORE-TC1775	Expansion Bus	Patch Field
/NMI	4A	4A	29A
/PORST	11C	11C	4E
CLKOUT	1B	1B	28C
CLKIN	1A	1A	28A
/RESIN	10D		3F
/BOOT	9C		3B
CLKOUT_RTC	30C		10E
/IRTC	33D		11B
SDA	32D		11C
SCL	31C		10F

Table 34: Pin Assignment Control & Misc. Signals for the phyCORE-TC1775 / Development Board / Expansion Board

Signal	phyCORE-TC1775	Expansion Bus	Patch Field
2V5_IN	1C, 2C, 1D, 2D	1C, 2C, 1D, 2D	1A, 1C
5V	4C, 5C	4C, 5C	2A, 1B
3V3_IN	4D, 5D	4D, 5D	2C, 1D
CLKIN	1A	1A	28A
VBAT	6C	6C	2B
GND	2A, 7A, 12A, 17A, 22A, 27A, 32A, 37A,42A, 47A, 52A, 57A, 62A, 67A, 72A, 77A, 4B, 9B, 14B, 19B, 24B, 29B, 34B, 39B, 44B, 49B, 54B, 59B, 64B, 69B, 74B, 79B, 3C, 7C, 12C, 17C, 22C, 27C, 32C, 37C, 42C, 47C, 52C, 57C, 3D, 9D, 14D, 19D, 24D, 29D, 34D, 39D, 44D, 49D, 54D	2A, 7A, 12A, 17A, 22A, 27A, 32A, 37A,42A, 47A, 52A, 57A, 62A, 67A, 72A, 77A, 4B, 9B, 14B, 19B, 24B, 29B, 34B, 39B, 44B, 49B, 54B, 59B, 64B, 69B, 74B, 79B, 3C, 7C, 12C, 17C, 22C, 27C, 32C, 37C, 42C, 47C, 52C, 57C, 3D, 9D, 14D, 19D, 24D, 29D, 34D, 39D, 44D, 49D, 54D	3C, 4C, 7C, 8C, 9C, 12C, 13C, 14C, 17C, 18C, 19C, 22C, 23C, 24C, 27C, 29C, 30C, 31C, 34C, 35C, 36C, 39C, 40C, 41C, 44C, 45C, 46C, 49C, 50C, 51C, 54C, 4D, 5D, 6D, 9D, 10D, 11D, 14D, 15D, 16D, 9D, 20D, 21D, 24D, 25D, 26D, 28D, 31D, 32D, 33D, 36D, 37D, 38D, 41D, 42D, 43D, 46D, 47D, 48D, 51D, 52D, 53D, 1E, 2E, 1F

Table 35: Pin Assignment Power Supply for the phyCORE-TC1775 / Development Board / Expansion Board

Signal	phyCORE-TC1775	Expansion Bus	Patch Field
NC	35A, 55B, 8C, 6D, 7D, 8D	35A, 55B, 8C, 6D, 7D, 8D	27B, 27D, 54D, 27F, 54F 39E, 46A, 3E, 2D, 2F, 3A

Table 36: Unused Pins on the phyCORE-TC1775 / Development Board / Expansion Board

15.3.11 DS2401 Silicon Serial Number

Communication to a DS2401 Silicon Serial Number can be implemented in various software applications for the definition of a node address or as copy protection in networked applications. The DS2401 can be soldered on space U10 or U9 on the Development Board, depending on the type of device packaging being used.

The Silicon Serial Number Chip mounted on the phyCORE-TriCORE Development Board can be connected to port pin P10.1 of the TC1775.

Jumper	Setting	Description
JP16	closed	Port pin P10.1 of the TC1775 is used to access the Silicon Serial Number

Table 37: JP19 Jumper Configuration for Silicon Serial Number Chip

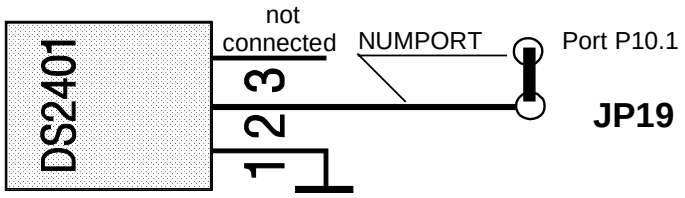


Figure 28: Connecting the DS2401 Silicon Serial Number

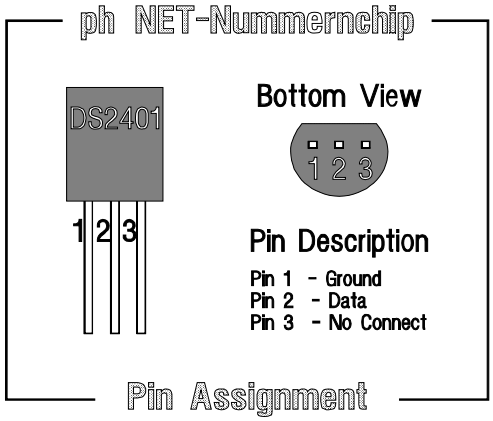


Figure 29: Pin Assignment of the DS2401 Silicon Serial Number

15.3.12 Pin Header Connectors X4 and X5

The pin headers X4 and X5 on the Development Board enable connection of an optional modem power supply. Connector X4 supplies 5 VDC at pin 1 and provides the phyCORE-TriCORE Development Board GND potential at pin 2. The maximum current draw depends on the power adapter used. We recommend the use of modems with less than 250 mA current draw.

16 debugCORE-TC1775 on the Development Board

The debugCORE-TC1775 is a special debugging version Single Board Computer (SBC) module which is 100 % function-compatible with the phyCORE-TC1775. As opposed to the phyCORE-TC1775, which was developed for use in OEM applications, the debugCORE-TC1775 offers an additional 16-pin pin header at X2 in 2 mm spacing and a 40-pin AMP connector at X2. Both connectors can be used for connection to emulator hardware allowing on-board debugging.

The following figure shows the positions of the additional components.

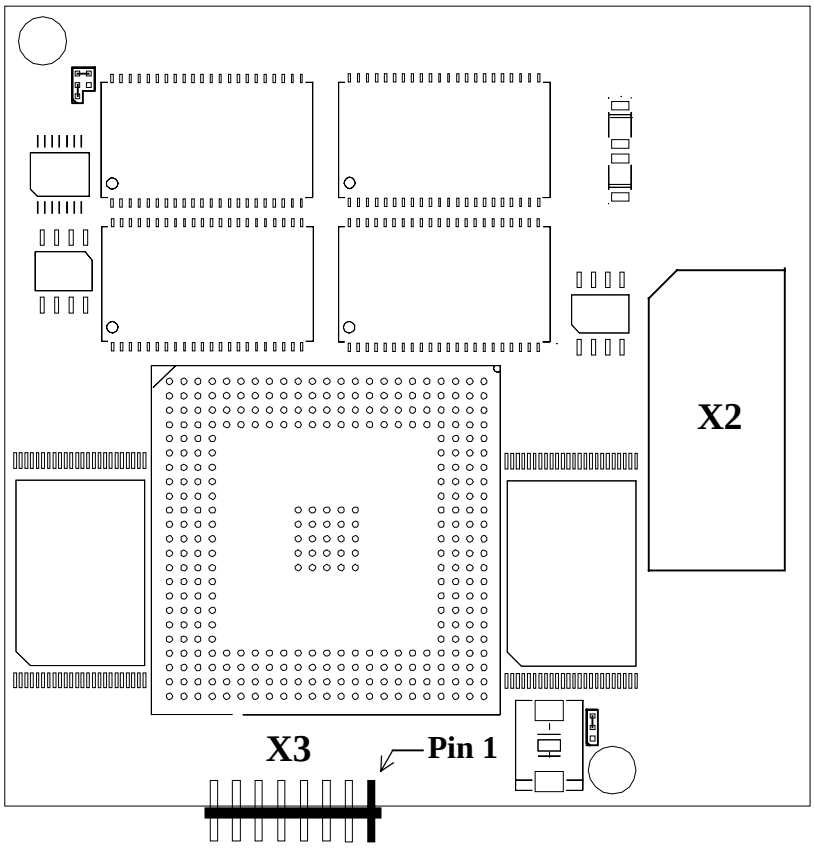


Figure 30: Positions of the Additional Components on the debugCORE-TC1775

1. OCDS1 (JTAG) (X3)

Direct access to the microcontroller is possible over the OCDS (on-chip debugging support) interface. This enables reading and writing of internal and external memory, as well as the setting of breakpoints. A level converter (Wiggler) serves as a connecting unit between the microcontroller and a host PC, that adapts the signals from the parallel PC interface to the microcontroller. A Wiggler of this type is already integrated on the phyCORE-TriCORE Development Board. The connections required between the module and the PC are established as follows:

a) Connect the included ribbon cable (WF040) to the connector X3 on the phyCORE-TC1775 and to pin header X6 on the Development Board. Be sure that the red ribbon cable is connected to pin 1 marked on the corresponding connectors!
b) Connect the parallel interface of your PC to the DB-25 socket (P3) of the phyCORE-TriCORE Development Board using the included parallel cable.

The Wiggler must be activated for a successful connection between the PC and the phyCORE.

Jumper	Setting	Description
JP10	closed	Wiggler activated

Table 38: Activating the Wiggler

A debugger from a third party manufacturer can be used as an alternative to the on-board Wiggler on the phyCORE-TriCORE Development Board.

2. OCDS2 (X2)

This OCDS2 interface provides access to additional information that the TC1775 microcontroller generates during runtime. This supplemental information is made available over the trace signals (port 5) and is used for more in-depth debugging in contrast to standard debugging features available via OCDS1. The trace information is used **in conjunction** with the control signals of the JTAG port and is stored in a special memory area of the debugger.

For debugging according to OCDS2, a debugger from a third party manufacturer is required with built-in trace memory. The steps required for establishing the connections are:

a) Connect the pin header X3 on the phyCORE-TC1775 with the JTAG input (OCDS1) on your debugger.
b) Insert the debugger's trace adapter into the 40-pin socket connector X2 on the phyCORE-TC1775
c) Follow the instructions provided by the debugger manufacturer for establishing a connection to your PC.

17 Revision History

Date	Version numbers	Changes in this manual
01-April-2003	Manual L-538e_1 PCM-010 PCB# 1181.1 PCM-993 PCB# 1182.0	First edition.

Appendice A

A.1 Release Notes

The following paragraph describes the differences between the technical facts provided in this manual and the currently available hardware revisions.

Changes in revision: PCB#1182.0 (phyCORE-TriCORE Development Board):

- The labeling for the expansion bus connector at X2 is incorrect. The letters A and B as well as C and D are in reversed order and should be swapped.

Index

Document: **phyCORE-TC1775**
Document number: **L-538e_1, April 2003**

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-538e_1
Printed in Germany