The Embedded I/O Company

TEWS TECHNOLOGIES

TPMC816

Two Independent Channels

Extended CAN Bus PMC Module

Version 2.0

User Manual

Issue 2.0.0

September 2009

TEWS TECHNOLOGIES GmbH

Am Bahnhof 7 25469 Halstenbek, Germany

Phone: +49 (0) 4101 4058 0 Fax: +49 (0) 4101 4058 19

e-mail: info@tews.com www.tews.com

TPMC816-10

2 independent channels extended CAN bus PMC module

TPMC816-11

1 independent channel extended CAN bus PMC module

Style Conventions

Hexadecimal characters are specified with prefix 0x, i.e. 0x029E (that means hexadecimal value 029E).

For signals on hardware products, an ‚Active Low' is represented by the signal name with # following, i.e. IP_RESET#.

Access terms are described as:

W	Write Only
R	Read Only
R/W	Read/Write
R/C	Read/Clear
R/S	Read/Set

Issue	Description	Date
1.0	Module Version 2.0 Exchange PCI – device and physical CAN interface	January 2002
1.1	Change default values for Hot Swap Register	June 2002
1.2	General Revision	December 2002
1.3	Corrections in PCI Target Chip Register Description, Updated Block Diagram for P14 I/O, Added I/O Connection Note	January 2005
1.4	New address TEWS LLC	September 2006
2.0.0	New notation of User Manual Issue	September 2009

Table of Contents

List of Figures

List of Tables

1 Product Description

The TPMC816 is a standard single-width 32 bit PMC module with two complete CAN bus interfaces using two Intel 82527 CAN controllers. Both channels are completely independent and support CAN specification 2.0 part A and B (Standard 11 bit identifier and extended 29 bit identifier).

Each channel provides CAN High Speed and modified RS485 as physical interface. The physical interfaces are optically isolated from the CAN controller and powered by an on board DC/DC converter for each channel. The TPMC816-11 provides one CAN bus channel.

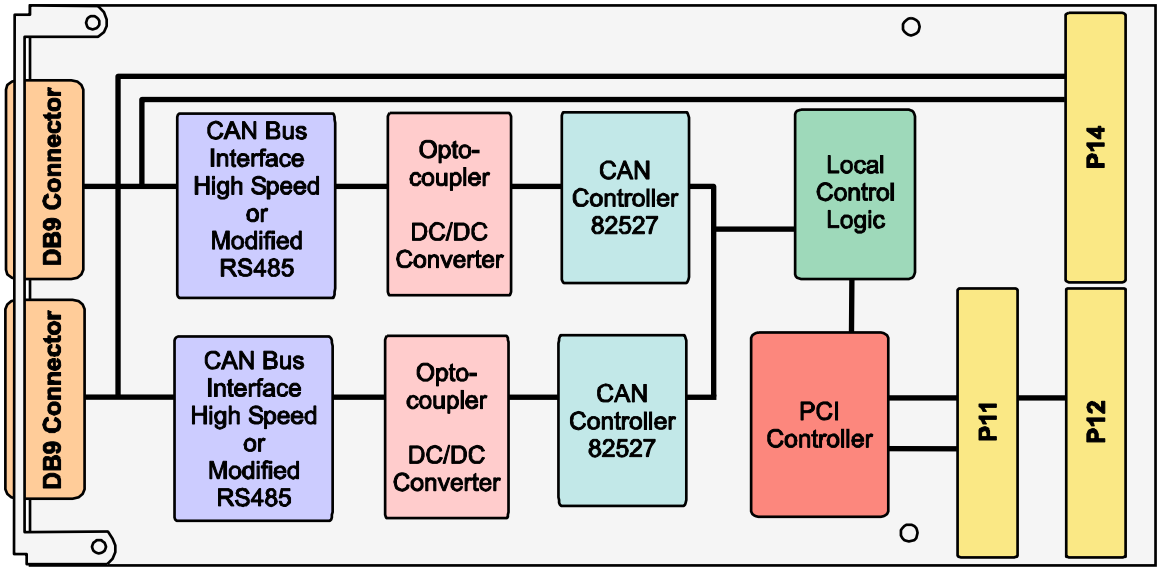

Figure 1-1 : Block Diagram

2 Technical Specification

Logic Interface	PCI Mezzanine Card Interface
PCI Target Chip	PCI9030 (PLX Technology)
PCI I/O Signaling Voltage Keying	+3.3V and +5.0V
Module Specific Data	
CAN Controller	Intel 82527
Number of Channels	TPMC816-10: 2 TPMC816-11: 1
Physical Interface	CAN high speed (according ISO 11898) Modified RS485 Selectable by jumper for each channel
Bus Line Termination	On board 120 ohms, selectable by jumper for each channel
Transfer Rate	Up to 1 Mbps at bus line length of up to 40 m
Isolation	CAN bus isolated by optocouplers
I/O Interface	
Front panel	TPMC816-10: two DB9 male connectors TPMC816-11: one DB9 male connector
P14 Connector	TPMC816-10: 64pol. Mezzanine P14 I/O TPMC816-11: 64pol. Mezzanine P14 I/O
Operating Data	
Power Requirements	170mA typical @ +3.3V DC 550mA typical @ +5V DC (TPMC816-10) 470mA typical @ +5V DC (TPMC816-11)
Temperature Range	Operating: – 40°C to + 85°C Storage: – 55°C to + 125°C
MTBF	TPMC816-10: 346808 h TPMC816-11: 431068 h
Weight	79 g
Size	Singe Size CMC
Humidity	5 – 95 % non-condensing

Table 2-1 : Technical Specification

3 Local Space Addressing

3.1 Local I/O Space

Not used by the TPMC816

3.2 Local Memory Space

The complete register sets of the two CAN controllers of the TPMC816 are accessible in the memory space of the PMC module.

Address range: PCI Base Address 2 for Local Address Space 0 + (0x0000 to 0x01FF)

CAN controller Channel 0 CANCH0: PCI Base Address 2 + (0x0000 to 0x00FF)

CAN controller Channel 1 CANCH1: PCI Base Address 2 + (0x0100 to 0x01FF)

3.3 Memory Address Map

Address	Symbol	Description
CAN Controller Channel 0 CANCH0		
0x0000		Control Register
0x0001		Status Register
0x0002		CPU Interface Register
0x0003		Reserved
0x0004-0x0005		High Speed Read Register
0x0006-0x0007		Global Mask – Standard
0x0008-0x000B		Global Mask – Extended
0x000C-0x000F		Message 15 Mask
0x0010-0x001E		Message 1
0x001F		CLKOUT Register
0x0020-0x002E		Message 2
0x002F		Bus Configuration Register
0x0030-0x003E		Message 3
0x003F		Bit Timing Register 0
0x0040-0x004E		Message 4
0x004F		Bit Timing Register 1
0x0050-0x005E		Message 5
0x005F		Interrupt Register
0x0060-0x006E		Message 6
0x006F		Reserved
0x0070-0x007E		Message 7
0x007F		Reserved

Address	Symbol	Description
0x0080-0x008E		Message 8
0x008F		Reserved
0x0090-0x009E		Message 9
0x009F		P1CONF
0x00A0-0x00AE		Message A
0x00AF		P2CONF
0x00B0-0x00BE		Message B
0x00BF		P1IN
0x00C0-0x00CE		Message C
0x00CF		P2IN
0x00D0-0x00DE		Message D
0x00DF		P1OUT
0x00E0-0x00EE		Message E
0x00EF		P2OUT
0x00F0-0x00FE		Message F
0x00FF		Serial Reset Address

Table 3-1 : Memory Address Map CAN Controller Channel 0 CANCH0

Address	Symbol	Description
CAN Controller Channel 1 CANCH1		
0x0100		Control Register
0x0101		Status Register
0x0102		CPU Interface Register
0x0103		Reserved
0x0104-0x0105		High Speed Read Register
0x0106-0x0107		Global Mask – Standard
0x0108-0x010B		Global Mask – Extended
0x010C-0x010F		Message 15 Mask
0x0110-0x011E		Message 1
0x011F		CLKOUT Register
0x0120-0x012E		Message 2
0x012F		Bus Configuration Register
0x0130-0x013E		Message 3
0x013F		Bit Timing Register 0
0x0140-0x014E		Message 4
0x014F		Bit Timing Register 1
0x0150-0x015E		Message 5
0x015F		Interrupt Register
0x0160-0x016E		Message 6

Address	Symbol	Description
0x016F		Reserved
0x0170-0x017E		Message 7
0x017F		Reserved
0x0180-0x018E		Message 8
0x018F		Reserved
0x0190-0x019E		Message 9
0x019F		P1CONF
0x01A0-0x01AE		Message A
0x01AF		P2CONF
0x01B0-0x01BE		Message B
0x01BF		P1IN
0x01C0-0x01CE		Message C
0x01CF		P2IN
0x01D0-0x01DE		Message D
0x01DF		P1OUT
0x01E0-0x01EE		Message E
0x01EF		P2OUT
0x01F0-0x01FE		Message F
0x01FF		Serial Reset Address

Table 3-2 : Memory Address Map CAN Controller Channel 1 CANCH1

4 PCI9030 Target Chip

4.1 PCI Configuration Registers (PCR)

4.1.1 PCI9030 Header

PCI CFG Register Address	Write '0' to all unused (Reserved) bits				PCI writeable	Initial Values (Hex Values)
	31 24	23 16	15 8	7 0		
0x00	Device ID		Vendor ID		N	9050 10B5
0x04	Status		Command		Y	0280 0000
0x08	Class Code			Revision ID	N	028000 0A
0x0C	BIST	Header Type	PCI Latency Timer	Cache Line Size	Y[7:0]	00 00 00 00
0x10	PCI Base Address 0 for MEM Mapped Config. Registers				Y	FFFFFF80
0x14	PCI Base Address 1 for I/O Mapped Config. Registers				Y	FFFFFF81
0x18	PCI Base Address 2 for Local Address Space 0				Y	FFFFFE00 (-10) FFFFFF00 (-11)
0x1C	PCI Base Address 3 for Local Address Space 1				Y	00000000
0x20	PCI Base Address 4 for Local Address Space 2				Y	00000000
0x24	PCI Base Address 5 for Local Address Space 3				Y	00000000
0x28	PCI CardBus Information Structure Pointer				N	00000000
0x2C	Subsystem ID		Subsystem Vendor ID		N	0330 1498
0x30	PCI Base Address for Local Expansion ROM				Y	00000000
0x34	Reserved			New Cap. Ptr.	N	000000 40
0x38	Reserved				N	00000000
0x3C	Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line	Y[7:0]	00 00 01 00
0x40	PM Cap.		PM Nxt Cap.	PM Cap. ID	N	4801 48 01
0x44	PM Data	PM CSR EXT	PM CSR		Y	00 00 0000
0x48	Reserved	HS CSR	HS Nxt Cap.	HS Cap. ID	Y[23:16]	00 020006
0x4C	VPD Address		VPD Nxt Cap.	VPD Cap. ID	Y[31:16]	0000 00 03
0x50	VPD Data				Y	00000000

Table 4-1 : PCI9030 Header

4.1.2 PCI Base Address Initialization

PCI Base Address Initialization is scope of the PCI host software.

PCI9030 PCI Base Address Initialization:

1. Write 0xFFFF_FFFF to the PCI9030 PCI Base Address Register.
2. Read back the PCI9030 PCI Base Address Register.
3. For PCI Base Address Registers 0:5, check bit 0 for PCI Address Space:

   Bit 0 = '0' requires PCI Memory Space mapping

   Bit 0 = '1' requires PCI I/O Space mapping

   For the PCI Expansion ROM Base Address Register, check bit 0 for usage:

   Bit 0 = ‘0’: Expansion ROM not used

   Bit 0 = ‘1’: Expansion ROM used
4. For PCI I/O Space mapping, starting at bit location 2, the first bit set determines the size of the required PCI I/O Space size.

   For PCI Memory Space mapping, starting at bit location 4, the first bit set to '1' determines the size of the required PCI Memory Space size.

   For PCI Expansion ROM mapping, starting at bit location 11, the first bit set to '1' determines the required PCI Expansion ROM size.

   For example, if bit 5 of a PCI Base Address Register is detected as the first bit set to ‘1’, the PCI9030 is requesting a 32 byte space (address bits 4:0 are not part of base address decoding).
5. Determine the base address and write the base address to the PCI9030 PCI Base Address Register. For PCI Memory Space mapping the mapped address region must comply with the definition of bits 3:1 of the PCI9030 PCI Base Address Register.

After programming the PCI9030 PCI Base Address Registers, the software must enable the PCI9030 for PCI I/O and/or PCI Memory Space access in the PCI9030 PCI Command Register (Offset 0x04). To enable PCI I/O Space access to the PCI9030, set bit 0 to '1'. To enable PCI Memory Space access to the PCI9030, set bit 1 to '1'.

Offset in Config.	Description	Usage
0x10	PCI9030 LCR’s MEM	Used
0x14	PCI9030 LCR’s I/O	Used
0x18	PCI9030 Local Space 0	Used
0x1C	PCI9030 Local Space 1	Not used
0x30	Expansion ROM	Not used

Table 4-2 : PCI9030 PCI Base Address Usage

4.2 Local Configuration Register (LCR)

After reset, the PCI9030 Local Configuration Registers are loaded from the on board serial configuration EEPROM.

The PCI base address for the PCI9030 Local Configuration Registers is PCI9030 PCI Base Address 0 (PCI Memory Space) (Offset 0x10 in the PCI9030 PCI Configuration Register Space) or PCI9030 PCI Base Address 1 (PCI I/O Space) (Offset 0x14 in the PCI9030 PCI Configuration Register Space).

Do not change hardware dependent bit settings in the PCI9030 Local Configuration Registers.

Offset from PCI Base Address	Register	Value	Description
0x00	Local Address Space 0 Range	0x0FFFFE00 (-10) 0x0FFFFF00 (-11)	Memory Space CAN
0x04	Local Address Space 1 Range	0x00000000	Not used
0x08	Local Address Space 2 Range	0x00000000	Not used
0x0C	Local Address Space 3 Range	0x00000000	Not used
0x10	Local Exp. ROM Range	0x00000000	Not used
0x14	Local Re-map Register Space 0	0x00000001	Enabled, Offset 0
0x18	Local Re-map Register Space 1	0x00000000	Not used
0x1C	Local Re-map Register Space 2	0x00000000	Not used
0x20	Local Re-map Register Space 3	0x00000000	Not used
0x24	Local Re-map Register ROM	0x00000000	Not used
0x28	Local Address Space 0 Descriptor	0x5400C042	Local Timing
0x2C	Local Address Space 1 Descriptor	0x00000000	Not used
0x30	Local Address Space 2 Descriptor	0x00000000	Not used
0x34	Local Address Space 3 Descriptor	0x00000000	Not used
0x38	Local Exp. ROM Descriptor	0x00000000	Not used
0x3C	Chip Select 0 Base Address	0x00000081	Chip select CAN0
0x40	Chip Select 1 Base Address	0x00000181	Chip select CAN1
0x44	Chip Select 2 Base Address	0x00000000	Not used
0x48	Chip Select 3 Base Address	0x00000000	Not used
0x4C	Interrupt Control/Status	0x00000049	Interrupt configuration
0x4E	EEPROM Write Protect Boundary	0x00000000	No write protection
0x50	Miscellaneous Control Register	0x00780000	Retry delay = max
0x54	General Purpose I/O Control	0xA4920224	All pins are outputs
0x70	Hidden1 Power Management data select	0x00000000	Not used
0x74	Hidden 2 Power Management data scale	0x00000000	Not used

Table 4-3 : PCI9030 Local Configuration Register

4.3 Configuration EEPROM

After power-on or PCI reset, the PCI9030 loads initial configuration register data from the on board configuration EEPROM.

The configuration EEPROM contains the following configuration data:

- Address 0x00 to 0x27 : PCI9030 PCI Configuration Register Values
- Address 0x28 to 0x87 : PCI9030 Local Configuration Register Values

See the PCI9030 Manual for more information.

Address	Offset							
	0x00	0x02	0x04	0x06	0x08	0x0A	0x0C	0x0E
0x00	0x9050	0x10B5	0x0280	0x0000	0x0280	0x000A	0x0330	0x1498
0x10	0x0000	0x0040	0x0000	0x0100	0x4801	0x4801	0x0000	0x0000
0x20	0x0000	0x4C06	0x0000	0x0003	0x0FFF	0xFE00	0x0000	0x0000
0x30	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000	0x0001
0x40	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000
0x50	0x5400	0xC042	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000
0x60	0x0000	0x0000	0x0000	0x0081	0x0000	0x0181	0x0000	0x0000
0x70	0x0000	0x0000	0x0030	0x0049	0x0078	0x0000	0xA492	0x0224
0x80	0x0000	0x0000	0x0000	0x0000	0xFFFF	0xFFFF	0xFFFF	0xFFFF
0x90	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF
0xA0	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF
0xB0	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF
0xC0	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF
0xD0	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF
0xE0	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF
0xF0	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF

Table 4-4 : Configuration EEPROM TPMC816-10

Address	Offset							
	0x00	0x02	0x04	0x06	0x08	0x0A	0x0C	0x0E
0x00	0x9050	0x10B5	0x0280	0x0000	0x0280	0x000A	0x0330	0x1498
0x10	0x0000	0x0040	0x0000	0x0100	0x4801	0x4801	0x0000	0x0000
0x20	0x0000	0x4C06	0x0000	0x0003	0x0FFF	0xFF00	0x0000	0x0000
0x30	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000	0x0001
0x40	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000
0x50	0x5400	0xC042	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000
0x60	0x0000	0x0000	0x0000	0x0081	0x0000	0x0000	0x0000	0x0000
0x70	0x0000	0x0000	0x0030	0x0041	0x0078	0x0000	0xA492	0x0224
0x80	0x0000	0x0000	0x0000	0x0000	0xFFFF	0xFFFF	0xFFFF	0xFFFF
0x90	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF
0xA0	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF
0xB0	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF
0xC0	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF
0xD0	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF
0xE0	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF
0xF0	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF	0xFFFF

Table 4-5 : Configuration EEPROM TPMC816-11

4.4 Local Software Reset

The PCI9030 Local Reset Output LRESETo# is used to reset the on board local logic.

The PCI9030 local reset is active during PCI reset or if the PCI Adapter Software Reset bit is set in the PCI9030 local configuration register CNTRL (offset 0x50).

CNTRL[30] PCI Adapter Software Reset:

Value of 1 resets the PCI9030 and issues a reset to the Local Bus (LRESETo# asserted). The PCI9030 remains in this reset condition until the PCI Host clears this bit. The contents of the PCI9030 PCI and Local Configuration Registers are not reset. The PCI9030 PCI Interface is not reset.

5 Configuration Hints

5.1 PCI Interrupt Control/Status Register (Address 0x4C)

Both CAN controllers generate interrupts at pin INTA# of the PCI bus. The interrupt status can be read at the Interrupt Status Register INTCSR of the PCI Controller PCI9030.

Bit	Description	Access	Reset Value
31:8	unused	R	0
7	Software Interrupt	R/W	0
6	PCI Interrupt Enable	R/W	1
5	CAN Controller 1 Interrupt Status	R	0
4	Local Interrupt 2 Polarity	R/W	0
3	Local Interrupt 2 Enable	R/W	1
2	CAN Controller 0 Interrupt Status	R	0
1	Local Interrupt 1 Polarity	R/W	0
0	Local Interrupt 1 Enable	R/W	1

Table 5-1 : Interrupt Control/Status Register INTCSR (Address 0x4C)

To enable or disable the interrupts use only the CAN controller Control Register bit 1 (PCI Base Address 2 + 0x0000 or PCI Base Address 2 + 0x0100). For more details please refer to the Intel 82527 data sheet which is part of the TPMC816-ED Engineering Documentation.

5.2 Software Reset (Controller and LRESET#)

A host on the PCI bus can set the software reset bit in the Miscellaneous Control Register (CNTRL; 0x50) of the PCI Controller PCI9030 to reset the Controller and assert LRESET# output. The PCI9030 remains in this reset condition until the PCI host clears the software reset bit.

5.3 Big / Little Endian

- PCI – Bus (Little Endian)

Byte 0	AD[7..0]
Byte 1	AD[15..8]
Byte 2	AD[23..16]
Byte 3	AD[31..24]

- Every Local Address Space (0...3) and the Expansion ROM Space can programmed to operate in Big or Little Endian Mode.

Big Endian		**Little Endian**	
32 Bit		**32 Bit**	
Byte 0	D[31..24]	Byte 0	D[7..0]
Byte 1	D[23..16]	Byte 1	D[15..8]
Byte 2	D[15..8]	Byte 2	D[23..16]
Byte 3	D[7..0]	Byte 3	D[31..24]
16 Bit upper lane		**16 Bit**	
Byte 0	D[31..24]	Byte 0	D[7..0]
Byte 1	D[23..16]	Byte 1	D[15..8]
16 Bit lower lane			
Byte 0	D[15..8]		
Byte 1	D[7..0]		
8 Bit upper lane		**8 Bit**	
Byte 0	D[31..24]	Byte 0	D[7..0]
8 Bit lower lane			
Byte 0	D[7..0]		

Table 5-2 : Local Bus Little/Big Endian

Standard use of the TPMC816:

Local Address Space 0	8 bit Bus in Little Endian Mode
Local Address Space 1	not used
Local Address Space 2	not used
Local Address Space 3	not used
Expansion ROM Space	not used

To change the Endian Mode use the Local Configuration Registers for the corresponding Space. Bit 24 of the according register sets the mode. A value of 1 indicates Big Endian and a value of 0 indicates Little Endian.

For further information please refer to the PCI9030 manual which is also part of the TPMC816 Engineering Documentation.

Use the PCI Base Address 0 + Offset or PCI Base Address 1 + Offset:

Short cut Offset	Name	
LAS0BRD	0x28	Local Address Space 0 Bus Region Description Register
LAS1BRD	0x2C	Local Address Space 0 Bus Region Description Register
LAS2BRD	0x30	Local Address Space 0 Bus Region Description Register
LAS3BRD	0x34	Local Address Space 0 Bus Region Description Register
EROMBRD	0x38	Expansion ROM Bus Region Description Register

You could also use the PCI - Base Address 1 I/O Mapped Configuration Registers.

6 Installation

6.1 CAN bus Termination

Each end of a CAN bus must be terminated by a 120 ohms resistor between the CAN bus lines CAN high and CAN low.

This termination could be activated by installing jumper on the jumper field J4/J5 and J8/J9. Because of split termination there are two jumpers for each channel.

Bus line termination for CAN bus channel 0 active: Jumper J4 + J5 (1-2) installed

Bus line termination for CAN bus channel 1 active: Jumper J8 + J9 (1-2) installed

To set the termination of a CAN bus I/O channel, always use both jumpers for that channel.

6.2 Physical Interface

Select the physical interface by installing jumper on jumper field J1 – J4.

	CAN High Speed	Mod. RS485
Channel 0	Jumper J2 (1-3) Jumper J2 (2-4) Jumper J3 (1-3) Jumper J3 (2-4)	Jumper J2 (3-5) Jumper J2 (4-6) Jumper J3 (3-5) Jumper J3 (4-6)
Channel 1	Jumper J6 (1-3) Jumper J6 (2-4) Jumper J7 (1-3) Jumper J7 (2-4)	Jumper J6 (3-5) Jumper J6 (4-6) Jumper J7 (3-5) Jumper J7 (4-6)

Table 6-1 : Physical Interface Selection

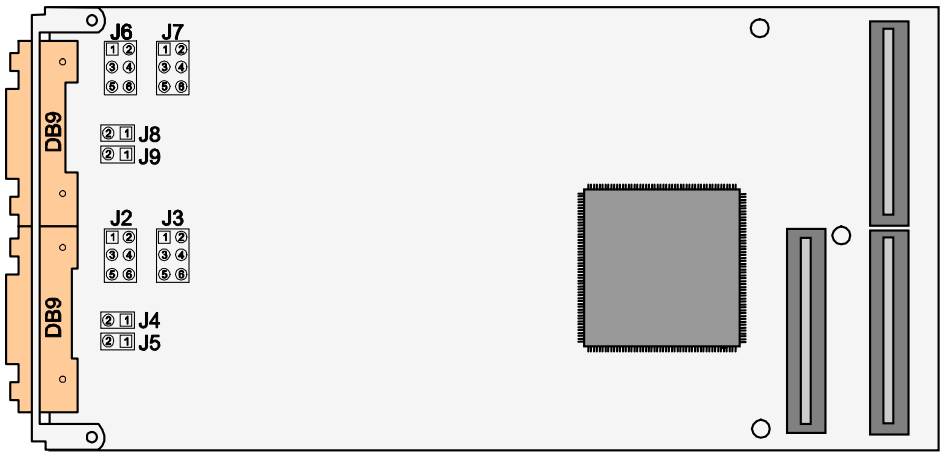

Figure 6-1 : Jumper Location TPMC816

6.3 Typical operating circuit

To connect the TPMC816 into an existing CAN network, link both signal lines (CAN High and CAN Low) and as well the signal ground line (GND).

Use the on board CAN termination of the TPMC816 only to terminate both ends of the CAN bus.

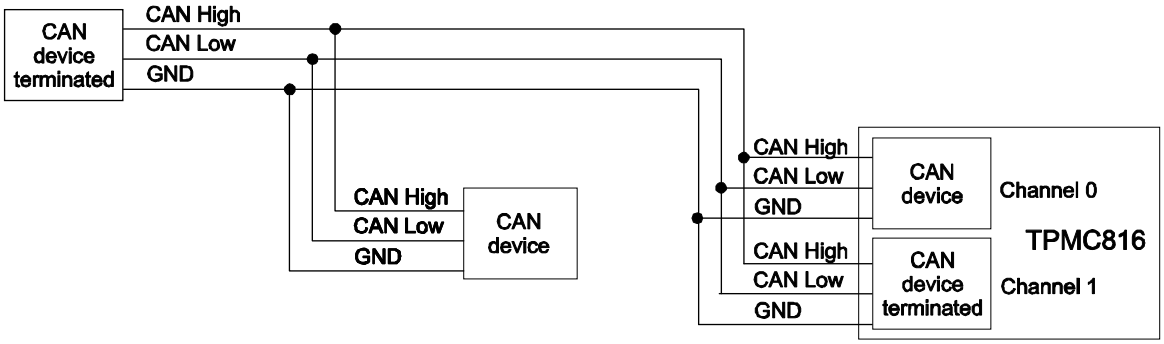

Figure 6-2 : Typical operating circuit

6.4 I/O Connection Note

Please note that on the TPMC816-1x, the P14 back I/O connector is always populated and is connected in parallel to the front I/O connector.
Do not use these modules on carrier boards where P14/J14 is reserved for other system signals but PMC I/O. Ask support for special board options with front I/O only in this case.
For a CAN bus channel, do either use front I/O or back I/O. Do not use both front I/O and back I/O at the same time for a CAN bus channel.

7 Pin Assignment

7.1 Front Panel I/O

CAN high speed (according to ISO11898) or Modified RS485 is selectable by jumper for each channel.

Pin-No.	Function	
1	NC	
2	Low level CAN voltage input/output	Selectable CAN high speed or mod. RS485
3	Ground channel 0	
4	NC	
5	NC	
6	Ground channel 0	
7	High level CAN voltage input/output	Selectable: CAN high speed or mod. RS485
8	NC	
9	NC	

Table 7-1 : DB9 Male Connector Channel 0

Pin-No.	Function	
1	NC	
2	Low level CAN voltage input/output	Selectable: CAN high speed or mod. RS485
3	Ground channel 1	
4	NC	
5	NC	
6	Ground channel 1	
7	High level CAN voltage input/output	Selectable: CAN high speed or mod. RS485
8	NC	
9	NC	

Table 7-2 : DB9 Male Connector Channel 1

7.2 Mezzanine Card Connector P14

Pin-No.	Function	
1	NC	
2	Ground channel 0	
3	Low level CAN voltage input/output channel 0	Selectable: CAN high speed or mod. RS485
4	High level CAN voltage input/output channel 0	Selectable: CAN high speed or mod. RS485
5	Ground channel 0	
6	NC	
7	NC	
8	NC	
9	NC	
10	NC	
11	Ground channel 1	
12	Low level CAN voltage input/output channel 1	Selectable: CAN high speed or mod. RS485
13	High level CAN voltage input/output channel 1	Selectable: CAN high speed or mod. RS485
14	Ground channel 1	
15	NC	
16..32	NC	

Table 7-3 : Mezzanine Card Connector P14

Please note that on the TPMC816-1x, the P14 back I/O connector is always populated and is connected in parallel to the front I/O connectors.
Do not use these modules on carrier boards where P14/J14 is reserved for other system signals but PMC I/O. Ask support for special board options with front I/O only in this case.
For a CAN bus channel, do either use front I/O or back I/O. Do not use both front I/O and back I/O at the same time for a CAN bus channel.