TPMC815

ARCNET PMC

Version 1.0 Revision A

User Manual

Issue 1.2

25. May 1999

D76815802

TEWS DATEN*TECHNIK* GmbH
Am Bahnhof 7
D-25469 Halstenbek
Germany
Tel.: +49 (0)4101 4058-0
Fax.: +49 (0)4101 4058-19

TPMC815-11
ARCNET Traditional Hybrid Interface (5 Mbps)

TPMC815-21
ARCNET Isolated RS485 Interface (5Mbps)

This manual covers all these products

TEWS DATEN*TECHNIK* GmbH has made any effort to ensure that this manual is accurate and complete. However TEWS DATEN*TECHNIK* GmbH reserves the right to change the product described in this document at any time without notice.

This product has been designed to operate with PMC Mezzanine Card compatible carriers. Connection to incompatible hardware is likely to cause serious damage.

TEWS DATEN*TECHNIK* GmbH is not liable for any damage arising out of the application or use of the device described herein.

Issue	Description	Date
1.2	Additions to Technical Specifications and PCI 9050 PCI Target Chip description.	25. May. 1999
1.1	Issue 1.1	10. Nov. 1998
1.0	First Issue	

Table of Contents

Table of Figures

1 Product Description

The TPMC815 provides a complete **5Mbps ARCNET interface** using the COM20020 controller from SMC. The COM20020 contains the ARCNET controller with transceiver and dual-port RAM.

Two Versions of the TPMC815 are available:

The **TPMC815-11** provides the **traditional isolated hybrid interface**.
The **TPMC815-21** provides an **isolated RS485 differential driver interface**.

The TPMC815 is ideal suited for Industrial / Factory Automation and Automotive Applications.

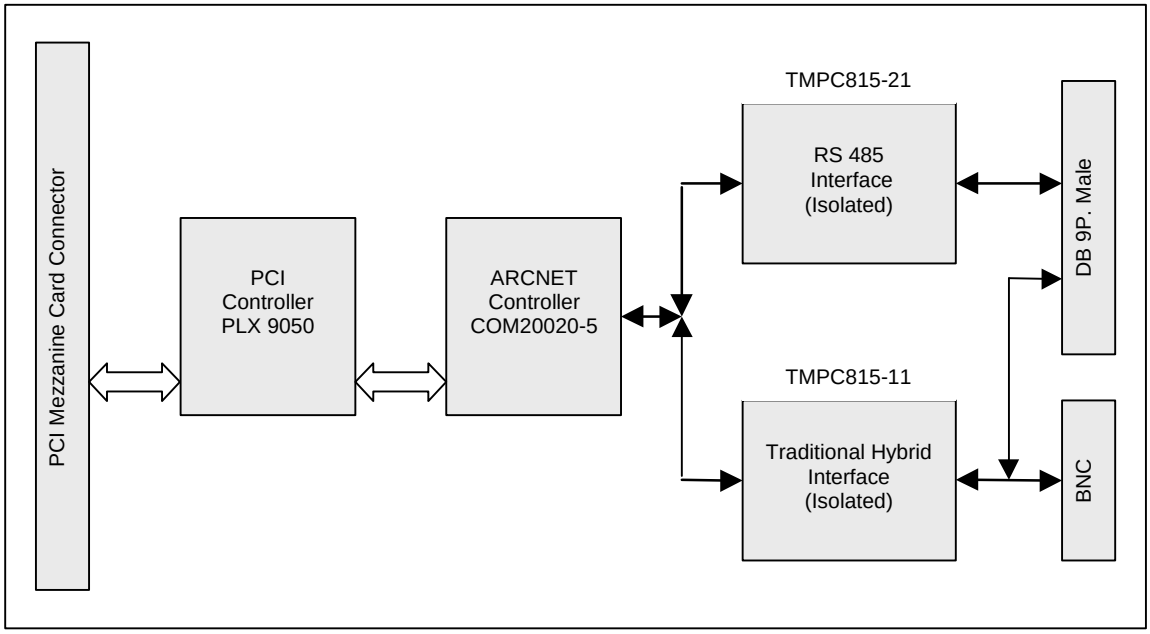

Figure 1-1: TPMC815 Block Diagram

2 Technical Specification

TECHNICAL SPECIFICATION	
INTERFACE	
Logic Interface	PCI Mezzanine Card Interface
I/O Interface	TPMC815-11: DB 9P. Male, BNC 90ohm TPMC815-21: DB 9P. Male
Physical Interface	TPMC815-11: Coax 93ohm or Twisted Pair TPMC815-21: Isolated RS485
Transfer Rates	5Mbps, 2.5Mbps, 1.25Mbps or 625Kbps software selectable
LOCAL DEVICES	
PCI Controller	PCI 9050-1
ARCNET Controller	SMC COM20020-5
PHYSICAL DATA	
Board Size	Single size CMC
Power Requirements	
3.3V	N.A.
5V	270mA (Typ.) (TPMC815-11) 290mA (Typ.) (TPMC815-21)
+12V	N.A.
-12V	0.15A (Typ.) (TPMC815-11) N.A. (TPMC815-21)
Temperature Range	Operating 0°C to 70°C
	Storage –55°C to +150°C
Humidity	5 – 95% non-condensing
MTBF	350.000 h (TPMC815-11) 434.825 h (TPMC815-21)
Weight	72g (TPMC815-11) 72g (TPMC815-21)

Figure 2-1: Technical Specification

3 TPMC815 Local Space Addressing

3.1 Local I/O Space

The complete register set of the ARCNET controller COM20020-5 and the ARCNET ID (SWITCH) register are located in the PCI I/O space.

PCI Base Address: PCI Base Address 2 for Local Address Space 0.

LOCAL I/O SPACE ADDRESS MAP			
ADDRESS OFFSET	**NAME**	**FUNCTION**	**SIZE**
ARCNET Controller			
$ 00	STIMCR	Status/Interrupt Mask	Byte
$ 01	DICOCR	Diagnostic/Command	Byte
$ 02	APTRHI	Address Pointer High	Byte
$ 03	APTLO	Address Pointer Low	Byte
$ 04	DATA	Data Register	Byte
$ 05	---	Reserved	Byte
$ 06	CONFIG	Configuration Register	Byte
$ 07	IDREG	Network ID Register	
Switch Register			
$ 08	SWITCH	Switch Register	Byte

Figure 3-1: Local I/O Space Address Map

3.1.1 ARCNET Controller Registers

The registers STIMCR, DICOCR, APTRHI, APTRLO, DATA, CONFIG and IDREG are implemented in the COM20020-5 ARCNET controller chip. For a detailed description of the usage of these registers please refer the SMC COM20020-5 data-book which is part of the TPMC815-EK engineering kit.

3.1.2 Switch Register

The SWITCH register is a 8 bit read only register. The value is that of the dip-switch, which is located on the TPMC815. The software can read this register to obtain the node-ID of the TPMC815 and then configure the ARCNET controller accordingly.

A switch in position **off** represents a logical **'1'**, a switch in position **on** represents a logical **'0'**. Factory configuration for the dip-switch is all switches in the *off* position.

3.2 Local Memory Space

N.A.

4 Functional Description

Please see the COM20020 Datasheet.

5 PCI 9050 Target Chip

5.1 PCI Configuration (CFG) Registers

5.1.1 PCI Header of the TPMC815

PCI CFG Register Address	Write '0' to all unused (Reserved) bits				PCI write able	Read after Reset (Hex Value)	Read after **initialization write access** (Hex Value)
	31 24	23 16	15 8	7 0			
00h	Device ID (Target Chip PCI 9050-1)		Vendor ID (PLX – Technology)		N	9050 10B5	9050 10B5
04h	Status		**Command**		Y	0280 0000	0280 0000
08h	Class Code			Revision ID	N	118000 XX	118000 XX
0Ch	BIST	Header Type	PCI Latency Timer	Cache line Size	Y[7:0]	00 00 00 00	00 00 00 00
10h	**PCI Base Address 0 for Memory Mapped Configuration Registers**				Y	00000000	FFFFFF70
14h	**PCI Base Address 1 for I/O Mapped Configuration Registers**				Y	00000001	FFFFFF71
18h	PCI Base Address 2 for Local Address Space 0				Y	00000000	FFFFFFF1
1Ch	PCI Base Address 3 for Local Address Space 1				Y	00000000	00000000
20h	PCI Base Address 4 for Local Address Space 2				Y	00000000	00000000
24h	PCI Base Address 5 for Local Address Space 3				Y	00000000	00000000
28h	Cardbus CIS Pointer				N	00000000	00000000
2Ch	Subsystem ID (TPMC815)		Subsystem Vendor ID (Tews Datentechnik GmbH)		N	032F 1498	032F 1498
30h	PCI Base Address for Local Expansion ROM				Y	00000000	00000000
34h	Reserved				N	00000000	00000000
38h	Reserved				N	00000000	00000000
3Ch	Max_Lat	Min_Gnt	Interrupt Pin	Interrupt Line	Y[7:0]	00 00 00 01	00 00 00 01

Figure 5-1 : PCI Configuration Register Map

- Device - ID and Vendor – ID depends on PCI Chip
  Vendor – ID 10B5 h PLX Technology
  Device – ID 9050 h PCI 9050-1 Target Chip

- Subvendor – ID and Subsystem - ID depends on TEWS Modul
  Subvendor – ID 1498 h Tews Datentechnik GmbH
  Subsystem – ID xxxx h 16 bit, depends on TPMCxxx

Subsystem – ID Example : **TPMC670 => 029E h**

Bit: **15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0**
Bus – Typ Modul – Typ

Bus – Typ
PMC **0000 – 0 h**
PC – MIP 0001 – 1 h
Compact PCI 0010 – 2 h
Standard PCI 0011 – 3 h

Modul – Typ
TPMC670 **1010011110 b – 29E h – 670 d**

5.1.2 PCI Base Address Initialization

PCI host bus-initialization software determines the required address space by an **initialization write access** (writing a value of all ones '1' to a PCI Base Address register) and then reading back the value of the PCI Base Address register. The PCI 9050 (PCI Target chip) returns zeros '0' in don't care address bits, specifying the required address space. The PCI software then maps the local address space into the PCI address space by programming the PCI Base Address register.

After programming the required address spaces the user must set bit 0 (enables I/O accesses) and bit 1 (enables memory accesses) of the command register (Offset 04h) to '1'.

5.1.2.1 I/O Base Address Implementation

1. Write a value of '1' to all bits of the PCI Base Address registers 0 to 5.
2. Check that bit 0 of the register contains a value of '1' (PCI 9050 needs an I/O address space)
3. Starting at bit location 2 of the PCI Base Address register, search for the first bit set to a value of '1'. This bit is the binary size of the total contiguous block of I/O address space needed by the PCI 9050.
   For example, if bit 5 of the PCI Base Address register is detected as the first bit set to '1', the PCI 9050 is requesting a 32 byte block of I/O address space.
4. Write the start address of the requested I/O address space to the PCI Base Address register.

The PCI Base Address 1 for I/O Mapped Configuration Registers (128 byte)
and the PCI Base Address 2 for Local Address Space 0 (16 byte)
are used by the TPMC815 as I/O address space.

5.1.2.2 Memory Base Address Implementation

1. Write a value of '1' to all bits of the PCI Base Address registers 0 to 5.
2. Check that bit 0 of the register contains a value of '0' (PCI 9050 needs an memory address space)
3. Starting at bit location 4 of the PCI Base Address register, search for the first bit set to a value of '1'. This bit is the binary size of the total contiguous block of memory address space needed by the PCI 9050.
   For example, if bit 15 of the PCI Base Address register is detected as the first bit set to '1', the PCI 9050 is requesting a 32 kilobyte block of memory address space.
4. Write the start address of the of the requested memory address block to the PCI Base Address register. This memory address region must not conflict with any other memory space utilized within the system. In addition, it must comply with the definition contained in bits 1 and 2 of this register.

The PCI Base Address 0 for Memory Mapped Configuration Registers (128 byte) is used by the TPMC815 as memory address space.

5.1.2.3 Expansion ROM Base Address Implementation

1. Write a value of '1' to bits 11 through 31 of PCI Base Address Local Expansion ROM register.
2. Starting at bit location 11 of the PCI Base Address Local Expansion ROM register, search upward for the first bit set to a value of '1'. This bit is the binary size of the total contiguous block of memory address space needed by the PCI 9050.
   For example, if bit 16 of the PCI Base Address Local Expansion ROM register is detected as the first bit set, the device is requesting a 64 kilobyte block of memory address space.
3. Write the start address of the requested memory address block to the PCI Base Address Local Expansion ROM register. This memory address region must not conflict with any other memory space utilized within the system.

The Expansion ROM is not used by the TPMC815.

For further information please refer to the PCI 9050 manual which is also part of the TPMC815 Engineering Manual.

5.2 Local Configuration Registers (LCR)

After reset, the Local Configuration Registers (LCRs) are loaded from the onboard EEPROM. The LCRs are accessible in the PCI Base Address 0 Memory Mapped or in the PCI Base Address 1 I/O Mapped Configuration Registers.

Note

Do not change the value of these registers because these values are hardware dependent.

Local Configuration Registers		
PCI (Offset from Local Base Address)	**Register**	**Value**
00h	Local Address Space 0 Range	0FFFFFF1
04h	Local Address Space 1 Range	00000000
08h	Local Address Space 2 Range	00000000
0Ch	Local Address Space 3 Range	00000000
10h	Local Exp. ROM Range	00000000
14h	Local Re-map Register Space 0	00000001
18h	Local Re-map Register Space 1	00000000
1Ch	Local Re-map Register Space 2	00000000
20h	Local Re-map Register Space 3	00000000
24h	Local Re-map Register ROM	00000000
28h	Local Address Space 0 Descriptor	5431F8C0
2Ch	Local Address Space 1 Descriptor	00000000
30h	Local Address Space 2 Descriptor	00000000
34h	Local Address Space 3 Descriptor	00000000
38h	Local Exp. ROM Descriptor	00000000
3Ch	Chip Select 0 Base Address	00000000
40h	Chip Select 1 Base Address	00000000
44h	Chip Select 2 Base Address	00000000
48h	Chip Select 3 Base Address	00000000
4Ch	Interrupt Control/Status	00000041
50h	Miscellaneous Control Register	00780000

Figure 5-2 : Local Configuration Registers

5.3 Target Configuration EEPROM

After reset, the PCI 9050 starts to load the configuration sequence from the onboard EEPROM. This EEPROM contains the following configuration data.

- From 00h to 0Eh : PCI – Configuration
- From 10h to 62h : Local – Configuration
- From 64h to 7Ch : not used
- 7Eh : TPMC variant

TPMC815-11 EEPROM Values

Address	00	02	04	06	08	0A	0C	0E
* 00 *	9050	10B5	1180	0000	029E	1498	0000	0100
* 10 *	0FFF	FFF1	0000	0000	0000	0000	0000	0000
* 20 *	0000	0000	0000	0001	0000	0000	0000	0000
* 30 *	0000	0000	0000	0000	5431	F8C0	0000	0000
* 40 *	0000	0000	0000	0000	0000	0000	0000	0009
* 50 *	0000	0000	0000	0000	0000	0000	0000	0041
* 60 *	0078	0000	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF
* 70 *	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	000B

Figure 5-3 : TPMC815-10 EEPROM Values

TPMC815-21 EEPROM Values

Address	00	02	04	06	08	0A	0C	0E
* 00 *	9050	10B5	1180	0000	029E	1498	0000	0100
* 10 *	0FFF	FFF1	0000	0000	0000	0000	0000	0000
* 20 *	0000	0000	0000	0001	0000	0000	0000	0000
* 30 *	0000	0000	0000	0000	5431	F8C0	0000	0000
* 40 *	0000	0000	0000	0000	0000	0000	0000	0009
* 50 *	0000	0000	0000	0000	0000	0000	0000	0041
* 60 *	0078	0000	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF
* 70 *	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	FFFF	0015

Figure 5-4 : TPMC815-21 EEPROM Values

Note

For changing these configuration values, and more details about the EEPROM please refer to the PCI 9050 data-sheet which is part of the TPMC815-EK engineering kit.

6 Programming Hints

6.1 COM20020 Controller

6.1.1 Hardware Initialization

The COM20020 ARCNET controller chip, which is used on the TPMC815, is able to work with different CPU bus-structures. The COM20020 learns the used bus-structure by monitoring the first bus cycles which address the COM20020.

To initialize the COM20020 controller to the right CPU bus-structure the software must do the following accesses to the COM20020 immediately after a system reset:

1. **Write 0x55 to the DICOCR register at byte offset 01**
2. **Read the DICOCR register at byte offset 01**

Note

If the COM20020 is not initialized after a reset with the above listed sequence, the ARCNET PMC will not operate.

6.1.2 Programming

For programming the COM20020 ARCNET controller please refer to the COM20020 datasheet which is part of the TPMC815-EK engineering kit.

7 Configuration Hints

7.1 PCI Interrupt Control/Status

For disabling / enabling PCI interrupts only set bit 6 of the PCI 9050 Interrupt Control/Status Register (INTCSR; $4C) to ‘0’ / ’1’.
Do not change any other bits of this register.

7.2 Software Reset (Controller and LRESET#)

A host on the PCI bus can set the software reset bit in the Miscellaneous Control Register (CNTRL; 50h) of the PCI Controller PCI 9050 to reset the Controller and assert LRESET# output. The PCI 9050 remains in this reset condition until the PCI host clears the software reset bit.

The PCI 9050 LRESET# pin is connected to the reset input of the COM20020 ARCNET Controller.
To (hardware-) reset the COM20020 ARCNET Controller only set bit 30 of the PCI 9050 Miscellaneous Register (CNTRL, $50) to ‘1’.
Do not change any other bits of this register.

7.3 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 be 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]	

Figure 7-1 : Local Bus Little/Big Endian

Standard usage of the TPMC815

Local Address Space 0	8Bit 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 the Endian Mode or to change the lane 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. To change the Lane use the Bit 25.
For further information please refer to the PCI 9050 manual which is also part of the TPMC815 Engineering Manual.

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

Short cut	**Offset**	**Name**
LAS0BRD	28h	Local Address Space 0 Bus Region Description Register
LAS1BRD	2Ch	Local Address Space 0 Bus Region Description Register
LAS2BRD	30h	Local Address Space 0 Bus Region Description Register
LAS3BRD	34h	Local Address Space 0 Bus Region Description Register
EROMBRD	38h	Expansion ROM Bus Region Description Register

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

8 Installation TPMC815

8.1 General Installation Notes

8.1.1 Onboard Switch

Both versions TPMC815-11 and TPMC815-21 are providing a 8 bit dip-switch located on board. This switch can be read by the software at initialization time and it's value can be used to program the node ID of the TPMC's ARCNET controller.
A switch in position **off** represents a logical '1', a switch in position **on** represents a logical '0'.
Factory configuration for the dip-switch is all switches in the *off* position.

8.2 Wiring Hints / Network Cabling

8.2.1 Coax

RG62 is the most popular ARCNET cable. Other cable types may be used, but RG62 offers the best electrical characteristics as well as a low cost. It is first choice for even difficult, electrically noisy environments. It is also simple to install and terminate.

8.2.2 Twisted Pair and RS485

For a Twisted Pair or RS485 network a twisted pair cable with an impedance of 100Ohm @ 1MHz should be used. It can be AWG 22, 24, 26.

Note

All network signal lines must be terminated at both extremes of the cabling network with a resistor connected between Phase A and Phase B for Twisted Pair – or between Signal + and Signal - for RS485. The resistor value should be equal to the impedance of the used cable.

8.3 Jumper Configuration

The TPMC815-11 must be configured by the jumpers J1 and J2.
J1 sets the termination (93 Ohm) for the coax interface.
J2 selects the desired media coax or twisted pair of the network.

The TPMC815-21 must be configured by the jumpers J3 and J4.
J3 and J4 are setting the termination (120 Ohm) and the DC offset (4K7 pull-up and pull-down).

TPMC815 JUMPER CONFIGURATION		
Jumper	**Installation**	**Function**
TPMC815-11		
J1	Closed	Coax Termination ON (93ohm)
	Open	Coax Termination OFF
J2	1-3 2-4	Coax Cable (BNC Connector)
	3-5 4-6	Twisted Pair Cable (DB 9p.)
TPMC815-21		
J3, J4	Both Closed	RS485 Termination (120ohm) and DC offset (4K7) ON
	Both Open	RS485 Termination (120ohm) and DC offset (4K7) OFF

Figure 8-1: TPMC815 Jumper Configuration

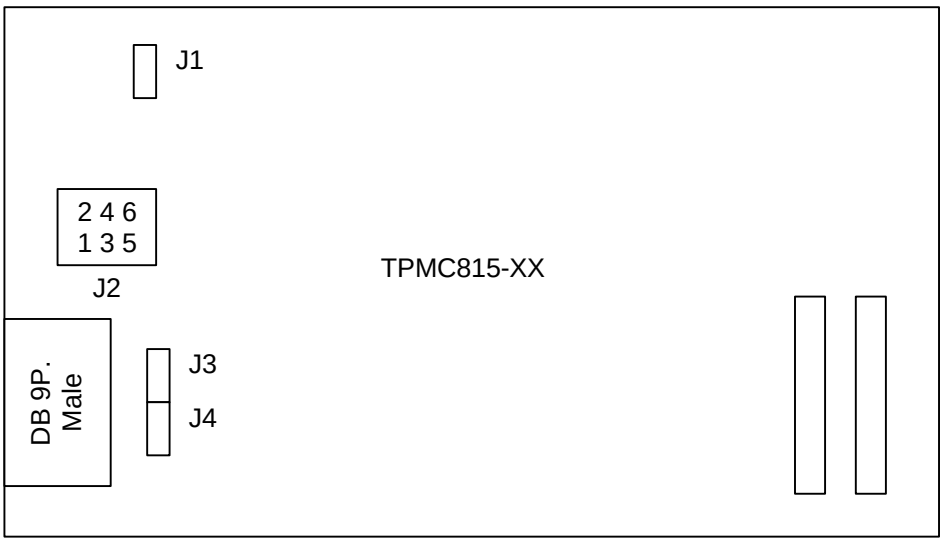

Figure 8-2: Jumper Location

9 Pin Assignment TPMC815

9.1 Front Panel I/O

DB 9P. MALE CONNECTOR		
Pin-No.	**Function**	**Comment**
1	Isolated RS485 Signal –	TPMC815-21 Only
2	Isolated RS485 Signal +	TPMC815-21 Only
3	NC	
4	Twisted Pair Phase B	TPMC815-11 Only
5	Twisted Pair Phase A	TPMC815-11 Only
6	Isolated RS485 GND	TPMC815-21 Only
7	Isolated RS485 GND	TPMC815-21 Only
8	NC	
9	Case Ground	

Figure 9-1: DB 9P. Male Connector

9.2 Mezzanine Card Connector P14

N.A.