The Embedded I/O Company

TEWS TECHNOLOGIES

TPCI868

16 Channel Asynchronous Serial Interface

Version 1.0

User Manual

Issue 1.3

September 2006

D77868800

TEWS TECHNOLOGIES GmbH
Am Bahnhof 7
25469 Halstenbek, Germany
www.tews.com
Phone: +49-(0)4101-4058-0
Fax: +49-(0)4101-4058-19
e-mail: info@tews.com

TEWS TECHNOLOGIES LLC
9190 Double Diamond Parkway,
Suite 127, Reno, NV 89521, USA
www.tews.com
Phone: +1 (775) 850 5830
Fax: +1 (775) 201 0347
e-mail: usasales@tews.com

TPCI868-10

16 channel asynchronous serial interface RS232, front panel I/O

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	First Issue	March 2001
1.1	Changed PCI Configuration (header description)	June 2001
1.2	General Revision	May 2003
1.3	New address TEWS LLC	September 2006

Table of Contents

Table of Figures

1 Product Description

The TPCI868 is a standard 33MHz 32 bit PCI module and provides 16 channels of asynchronous RS232 serial interface. The TPCI868-10 requires 3.3V PCI power supply, but it can operate with 3.3V or 5.0V PCI signaling voltage for I/O.

The module uses a 14.7456MHz crystal oscillator which allows baud rates up to 921.6kbaud. All 16 serial channels can generate interrupts on the PCI bus. The interrupt status information can be accessed via the 16 bit wide interrupt status register.

For the serial interface the signals TxD, RxD, RTS#, CTS# and GND are available on the HD68 front panel connector.

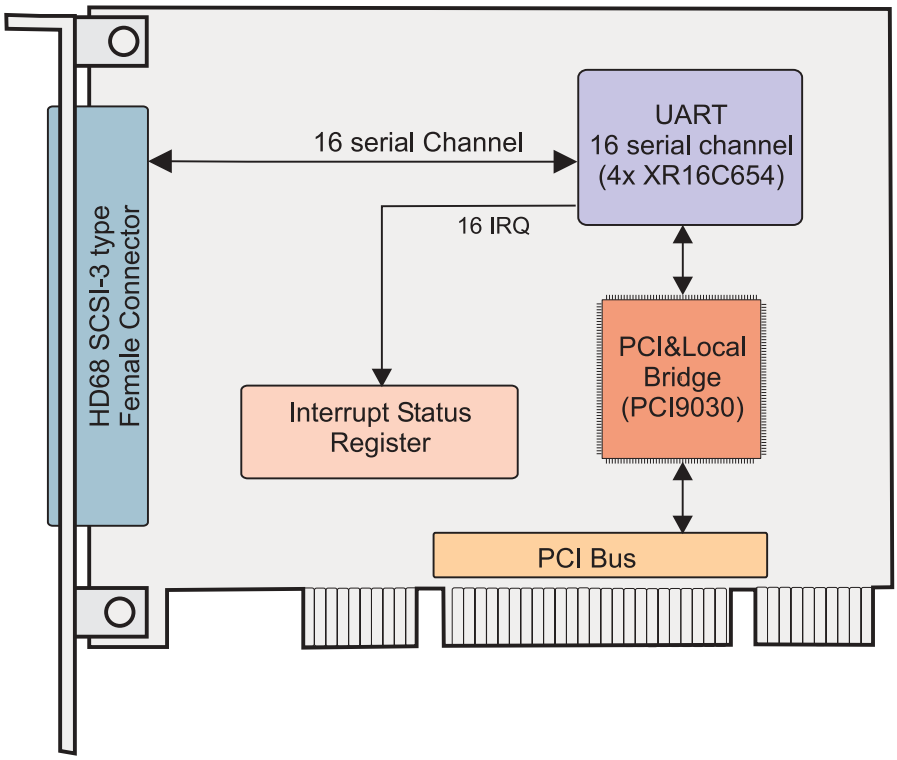

Figure 1-1 : Block Diagram

2 Technical Specification

PMC Interface		
Mechanical Interface	PCI Card Interface	
Electrical Interface	PCI Rev. 2.1 compliant 33MHz / 32bit PCI 3.3V and 5V PCI Signaling Voltage	
On Board Devices		
PCI Target Chip	PCI9030 (PLX Technology)	
UART Type	XR16C654 (Exar)	
I/O Interface		
Number of Channels	16	
Physical Interface	RS232	
Interface Type	Asynchronous serial interface	
Serial Channel I/O Signals	TxD, RxD, RTS#, CTS#	
Maximum Load / Channel @1000kbps	3kohms/250pF	
I/O Connector	HD68 SCSI-3 type connector	
Physical Data		
Power Requirements	210mA typical @ +3.3V DC	
Temperature Range	Operating Storage	0 °C to +70 °C -40°C to +85°C
MTBF	233825 h	
Humidity	5 – 95 % non-condensing	
Weight	75 g	

Figure 2-1 : Technical Specification

3 Local Space Addressing

3.1 PCI9030 Local Space Configuration

The local on board addressable regions are accessed from the PCI side by using the PCI9050 local spaces.

PCI9050 Local Space	PCI9050 PCI Base Address (Offset in PCI Configuration Space)	PCI Space Mapping	Size (Byte)	Port Width (Bit)	Endian Mode	Description
0	0 (0x10)	MEM	128	32	LITTLE	Local Configuration Registers
1	1(0x14)	I/O	128	32	LITTLE	Local Configuration Registers
2	2 (0x18)	MEM	256	8	BIG (lower lane)	Local Address Space 0
3	3 (0x1C)	MEM	16	16	BIG (lower lane)	Local Address Space 1
4	4 (0x20)	-	-	-	-	Local Address Space 2
5	5 (0x24)	-	-	-	-	Local Address Space 3
6	6 (0x30)	-	-	-	-	Local Expansion ROM Space

Figure 3-1 : PCI9030 Local Space Configuration

3.2 Local Address Space 0

PCI Base Address: **PCI9030 PCI Base Address 2 (Offset 0x18 in PCI Configuration Space).**

The TPCI868 uses the Exar XR16C654 UART controller to provide and control the 16 channels.

UART Register Space Contents	PCI Address	Size (bit)
Serial Channel 1 Register Set	PCI Base Address 2 + (0x0000 to 0x0007)	8
Serial Channel 2 Register Set	PCI Base Address 2 + (0x0008 to 0x000F)	8
Serial Channel 3 Register Set	PCI Base Address 2 + (0x0010 to 0x0017)	8
Serial Channel 4 Register Set	PCI Base Address 2 + (0x0018 to 0x001F)	8
Serial Channel 5 Register Set	PCI Base Address 2 + (0x0020 to 0x0027)	8
Serial Channel 6 Register Set	PCI Base Address 2 + (0x0028 to 0x002F)	8
Serial Channel 7 Register Set	PCI Base Address 2 + (0x0030 to 0x0037)	8
Serial Channel 8 Register Set	PCI Base Address 2 + (0x0038 to 0x003F)	8
Serial Channel 9 Register Set	PCI Base Address 2 + (0x0040 to 0x0047)	8
Serial Channel 10 Register Set	PCI Base Address 2 + (0x0048 to 0x004F)	8
Serial Channel 11 Register Set	PCI Base Address 2 + (0x0050 to 0x0057)	8
Serial Channel 12 Register Set	PCI Base Address 2 + (0x0058 to 0x005F)	8
Serial Channel 13 Register Set	PCI Base Address 2 + (0x0060 to 0x0067)	8
Serial Channel 14 Register Set	PCI Base Address 2 + (0x0068 to 0x006F)	8
Serial Channel 15 Register Set	PCI Base Address 2 + (0x0070 to 0x0077)	8
Serial Channel 16 Register Set	PCI Base Address 2 + (0x0078 to 0x007F)	8
FIFO Ready Register CH1-CH4	PCI Base Address 2 + (0x0080)	8
FIFO Ready Register CH5-CH8	PCI Base Address 2 + (0x0084)	8
FIFO Ready Register CH9-CH12	PCI Base Address 2 + (0x0088)	8
FIFO Ready Register CH13-CH16	PCI Base Address 2 + (0x008C)	8

Figure 3-2 : UART Register Space

3.2.1 Serial Channel Register

The UART register space provides a register set for each of the 16 serial channels.

Register Set	Register Set Offset
Serial Channel 1	0x0000
Serial Channel 2	0x0008
Serial Channel 3	0x0010
Serial Channel 4	0x0018
Serial Channel 5	0x0020
Serial Channel 6	0x0028
Serial Channel 7	0x0030
Serial Channel 8	0x0038
Serial Channel 9	0x0040
Serial Channel 10	0x0048
Serial Channel 11	0x0050
Serial Channel 12	0x0058
Serial Channel 13	0x0060
Serial Channel 14	0x0068
Serial Channel 15	0x0070
Serial Channel 16	0x0078

Figure 3-3 : Serial Channel Register Set Offset

Each register set provides the following 8 bit registers:

Register Symbol	Register Name
DLL	Baud Rate Divisor LSB
DLM	Baud Rate Divisor MSB
EFR	Enhanced Feature Register
FCR	FIFO Control Register
IER	Interrupt Enable Register
ISR	Interrupt Status Register
LCR	Line Control Register
LSR	Line Status Register
MCR	Modem Control Register
MSR	Modem Status Register
RHR	Receive Holding Register
SCPD	Scratchpad Register
THR	Transmit Holding Register
XON1	Xon-1 Word
XON2	Xon-2 Word
XOFF1	Xoff-1 Word
XOFF2	Xoff-2 Word

Figure 3-4 : Serial Channel Register

The following table provides the register offsets within a register set, access types and access control:

Access Control	Register Offset								Access Type
	0x0	0x1	0x2	0x3	0x4	0x5	0x6	0x7	
LCR Bit 7 = 0	RHR		ISR			LSR	MSR	SCPD	READ
	THR	IER	FCR	LCR	MCR			SCPD	WRITE
LCR Bit 7 = 1	DLL	DLM							READ
	DLL	DLM							WRITE
LCR = 0xBF			EFR		XON1	XON2	XOFF1	XOFF2	READ
			EFR		XON1	XON2	XOFF1	XOFF2	WITE

Figure 3-5 : Serial Channel Register Offset and Access Control

The address for a serial channel *register x* in a register set for *channel y* is:

PCI Base Address 2 (PCI Base Address for the Local Address Space 0)

+ Register Set Offset for *channel y*

+ Register Offset for *register x*.

Addressing example:

The address for the LCR register of serial channel 5 is:

PCI Base Address 2	(PCI Base Address for the Local Address Space 0)
+ 0x20	(Offset of the register set for serial channel 5)
+ 0x3	(Offset of the LCR register within a channel register set)

Because some serial channel registers overlap in address space, further access control must be programmed to access those registers.

Some serial channel registers are write-only or read-only registers.

All serial channel registers are byte sized.

For a description of the serial channel registers please refer to the XR16C654 data sheet which is part of the TPCI868-ED Engineering Documentation.

3.2.2 FIFO Ready Register

There are 4 UART FIFO Ready Registers. Each FIFO Ready Register provides status data for 4 serial channels.

Register	Description	PCI Address
FIFORDY1	FIFO Ready Register CH1-CH4	PCI Base Address 2 + (0x0080)
FIFORDY2	FIFO Ready Register CH5-CH8	PCI Base Address 2 + (0x0084)
FIFORDY3	FIFO Ready Register CH9-CH12	PCI Base Address 2 + (0x0088)
FIFORDY4	FIFO Ready Register CH13-CH16	PCI Base Address 2 + (0x008C)

Figure 3-6 : FIFO Ready Register

The FIFO Ready Registers are byte wide read registers.

The FIFO Ready Registers provide real-time status of the serial channel transmit and receive FIFOs.

Each TX and RX channel has its own 64byte FIFO. When any of the 8 TX/RX FIFOs becomes empty/full, the status bit associated with the TX/RX channel is set in the FIFO Ready Register.

Bit	Symbol	Description	Access	Reset Value
7	RXRDY4	Read as '1': the receiver is ready and is filled to a level below the programmed trigger level. Read as '0': the corresponding receive FIFO is filled to a level above the programmed trigger level or a time-out has occurred.	R	
6	RXRDY3			
5	RXRDY2			
4	RXRDY1			
3	TXRDY4	Read as '1': one or more empty locations exist in the corresponding transmit FIFO. Read as '0': the corresponding transmit FIFO is full. The channel will not accept any more transmit data.	R	
2	TXRDY3			
1	TXRDY2			
0	TXRDY1			

Figure 3-7 : FIFORDY1 Register (CH1 - CH4)

Bit	Symbol	Description	Access	Reset Value
7	RXRDY8	Read as '1': the receiver is ready and is filled to a level below the programmed trigger level. Read as '0': the corresponding receive FIFO is filled to a level above the programmed trigger level or a time-out has occurred.	R	
6	RXRDY7			
5	RXRDY6			
4	RXRDY5			
3	TXRDY8	Read as '1': one or more empty locations exist in the corresponding transmit FIFO. Read as '0': the corresponding transmit FIFO is full. The channel will not accept any more transmit data.	R	
2	TXRDY7			
1	TXRDY6			
0	TXRDY5			

Figure 3-8 : FIFORDY2 Register (CH5 - CH8)

Bit	Symbol	Description	Access	Reset Value
7	RXRDY12	Read as '1': the receiver is ready and is filled to a level below the programmed trigger level. Read as '0': the corresponding receive FIFO is filled to a level above the programmed trigger level or a time-out has occurred.	R	
6	RXRDY11			
5	RXRDY10			
4	RXRDY9			
3	TXRDY12	Read as '1': one or more empty locations exist in the corresponding transmit FIFO. Read as '0': the corresponding transmit FIFO is full. The channel will not accept any more transmit data.	R	
2	TXRDY11			
1	TXRDY10			
0	TXRDY9			

Figure 3-9 : FIFORDY3 Register (CH9 - CH12)

Bit	Symbol	Description	Access	Reset Value
7	RXRDY16	Read as '1': the receiver is ready and is filled to a level below the programmed trigger level. Read as '0': the corresponding receive FIFO is filled to a level above the programmed trigger level or a time-out has occurred.	R	
6	RXRDY15			
5	RXRDY14			
4	RXRDY13			
3	TXRDY16	Read as '1': one or more empty locations exist in the corresponding transmit FIFO. Read as '0': the corresponding transmit FIFO is full. The channel will not accept any more transmit data.	R	
2	TXRDY15			
1	TXRDY14			
0	TXRDY13			

Figure 3-10: FIFORDY4 Register (CH13 - CH16)

3.3 Local Address Space 1

PCI Base Address: PCI9030 PCI Base Address 3 (Offset 0x1C in PCI Configuration Space).

Interrupt Status Register Space	PCI Address	Size (Bit)
Interrupt Status Register	PCI Base Address 3+ 0x0000	16

Figure 3-11: Interrupt Status Register Space

Bit	Symbol	Description	Access	Reset Value
15	INT_CH16	0 : No active interrupt request 1 : Active interrupt request	R	
14	INT_CH15			
13	INT_CH14			
12	INT_CH13			
11	INT_CH12			
10	INT_CH11			
9	INT_CH10			
8	INT_CH9			
7	INT_CH8			
6	INT_CH7			
5	INT_CH6			
4	INT_CH5			
3	INT_CH4			
2	INT_CH3			
1	INT_CH2			
0	INT_CH1			

Figure 3-12: Interrupt Status Register

Each of the 16 serial channel interrupts is mapped to the Local Interrupt 1 of the PCI9030 (PCI Target Chip).

The PCI9030 (PCI Target Chip) maps the local interrupts to the PCI INTA interrupt line if PCI interrupts are enabled in the PCI9030 (PCI Target Chip) Interrupt Control Register.

If PCI interrupts are disabled in the PCI9030 (PCI Target Chip) Interrupt Control Register, the Interrupt Status Register can be used as a polling register for serial channel interrupts.

After Reset PCI interrupts are enabled in the PCI9030 (PCI Target Chip) Interrupt Control Register.

Each of the 16 serial channel interrupts can be enabled individually by the IER register in the UART register space.

After reset all serial channel interrupts are disabled by the UART's IER register.

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	3364 1498
0x04	Status		Command		Y	0280 0000
0x08	Class Code			Revision ID	N	070200 00
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	FFFFFF00
0x1C	PCI Base Address 3 for Local Address Space 1				Y	FFFFFFF0
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	000A 1498
0x30	PCI Base Address for Local Expansion ROM				Y	00000000
0x34	Reserved			New Cap. Ptr.	N	000000 00
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 00 4C 06
0x4C	VPD Address		VPD Nxt Cap.	VPD Cap. ID	Y[31:16]	0000 00 03
0x50	VPD Data				Y	00000000

Figure 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

Figure 4-2 : PCI9030 PLD 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
0x00	Local Address Space 0 Range	0x0FFF_FF00
0x04	Local Address Space 1 Range	0x0FFF_FFF0
0x08	Local Address Space 2 Range	0x0000_0000
0x0C	Local Address Space 3 Range	0x0000_0000
0x10	Local Exp. ROM Range	0x0000_0000
0x14	Local Re-map Register Space 0	0x0000_0001
0x18	Local Re-map Register Space 1	0x0000_0101
0x1C	Local Re-map Register Space 2	0x0000_0000
0x20	Local Re-map Register Space 3	0x0000_0000
0x24	Local Re-map Register ROM	0x0000_0000
0x28	Local Address Space 0 Descriptor	0x5501_40C0
0x2C	Local Address Space 1 Descriptor	0x5541_40C0
0x30	Local Address Space 2 Descriptor	0x0000_0000
0x34	Local Address Space 3 Descriptor	0x0000_0000
0x38	Local Exp. ROM Descriptor	0x0000_0000
0x3C	Chip Select 0 Base Address	0x0000_0081
0x40	Chip Select 1 Base Address	0x0000_0103
0x44	Chip Select 2 Base Address	0x0000_0000
0x48	Chip Select 3 Base Address	0x0000_0000
0x4C	Interrupt Control/Status	0x0000_0041
0x50	Miscellaneous Control Register	0x0078_0000

Figure 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	0x3364	0x1498	0x0281	0x0000	0x0702	0x000A	0x000A	0x1498
0x10	0x0000	0x0040	0x0000	0x0100	0x4801	0x4801	0x0000	0x0000
0x20	0x0000	0x4C06	0x0000	0x0003	0x0FFF	0xFF00	0x0FFF	0xFFF0
0x30	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000	0x0001
0x40	0x0000	0x0101	0x0000	0x0000	0x0000	0x0000	0x0000	0x0000
0x50	0x5501	0x40C0	0x5541	0x40C0	0x0000	0x0000	0x0000	0x0000
0x60	0x0000	0x0000	0x0000	0x0081	0x0000	0x0103	0x0000	0x0000
0x70	0x0000	0x0000	0x0000	0x0041	0x0078	0x0000	0x0000	0x0001
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

Figure 4-4 : Configuration EEPROM TPCI868-10

Do not change these register values because these values are hardware dependent.

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.

4.5 PCI Interrupt Control/Status Register

The PCI9030 Interrupt Control / Status Register (INTCSR; 4Ch) provides a PCI Interrupt Enable control bit (Bit 6) for enabling or disabling PCI interrupts.

To disable PCI interrupts a host on the PCI bus must clear the PCI Interrupt Enable bit in the Interrupt Control / Status Register.

To enable PCI interrupts, a host on the PCI bus must set the PCI Interrupt Enable bit in the Interrupt Control / Status Register.

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

Figure 4-5 : Interrupt Control/Status Register (INTCSR; 0x4C)

5 Configuration Hints

5.1 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 5-1 : Local Bus Little/Big Endian

Standard use of the TPCI868:

Local Address Space 0	8 bit bus in Big Endian Mode (Lower Lane)
Local Address Space 1	16 bit bus in Big Endian Mode (Lower Lane)
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 TPCI868-ED 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 Functional Description

For a detailed description of the UART functions please refer to the XR16C654 data sheet which is part of the TPCI868-ED Engineering Documentation.

7 Programming Hints

7.1 UART Baud Rate Programming

Each of the 16 serial channels of the TPCI868 provides a programmable Baud Rate Generator. The clock of the 16C654 UART can be divided by any divisor from 1 to 2^{16} – 1. The divisor can be programmed by the UART channel DLM (Divisor MSB) and DLL (Divisor LSB) registers. After reset bit 7 of the UART channels MCR register defaults to '0' and the divisor value is 0xFFFF.

The basic formula of baud rate programming is:

$$\frac{14.7456MHz}{16 * DIVISOR * (1 + 3 * MCR\_BIT7)}$$

Baud Rate MCR Bit 7 = 0	Baud Rate MCR Bit 7 = 1	Divisor	DLM Value	DLL Value
400	100	0x0900	0x09	0x00
600	150	0x0600	0x06	0x00
1200	300	0x0300	0x03	0x00
2400	600	0x0180	0x01	0x80
4800	1200	0x00C0	0x00	0xC0
9600	2400	0x0060	0x00	0x60
19.2K	4800	0x0030	0x00	0x30
38.4K	9600	0x0018	0x00	0x18
57.6K	14.4K	0x0010	0x00	0x10
76.8K	19.2K	0x000C	0x00	0x0C
153.6K	38.4K	0x0006	0x00	0x06
307.2K	76.8K	0x0003	0x00	0x03
460.8K	115.2K	0x0002	0x00	0x02
921.6K	230.4K	0x0001	0x00	0x01

Figure 7-1 : UART Baud Rate Programming

These steps should be used to modify the DLM, DLL registers of an UART channel:

1. Write 0x80 to the LCR register of the UART channel (enable access to the DLM, DLL registers).
2. Program the DLM, DLL registers of the UART channel.
3. Write normal operation byte value to the LCR register of the UART channel.

These steps should be used to modify MCR register bit 7 of an UART channel:

1. Write 0xBF to the LCR register of the UART channel (enable access to EFR register).
2. Set UART channel EFR register bit 4 to '1' (enable modification of MCR register bits 5-7).
3. Modify UART channel MCR register bit 7.
4. Set UART channel EFR register bit 4 to '0' (latch modified MCR register setting).
5. Write normal operation byte value to the LCR register of the UART channel.

8 Pin Assignment – I/O Connector

8.1 TPCI868-10 Front Panel I/O Connector

Pin	Signal	Description	Direction	Level
1	TXD[00]	Transmit Data Serial Channel 1	Output	RS232
2	RTS#[00]	Request To Send Serial Channel 1	Output	RS232
3	TXD[01]	Transmit Data Serial Channel 2	Output	RS232
4	RTS#[01]	Request To Send Serial Channel 2	Output	RS232
5	TXD[02]	Transmit Data Serial Channel 3	Output	RS232
6	RTS#[02]	Request To Send Serial Channel 3	Output	RS232
7	TXD[03]	Transmit Data Serial Channel 4	Output	RS232
8	RTS#[03]	Request To Send Serial Channel 4	Output	RS232
9	GND			
10	TXD[04]	Transmit Data Serial Channel 5	Output	RS232
11	RTS#[04]	Request To Send Serial Channel 5	Output	RS232
12	TXD[05]	Transmit Data Serial Channel 6	Output	RS232
13	RTS#[05]	Request To Send Serial Channel 6	Output	RS232
14	TXD[06]	Transmit Data Serial Channel 7	Output	RS232
15	RTS#[06]	Request To Send Serial Channel 7	Output	RS232
16	TXD[07]	Transmit Data Serial Channel 8	Output	RS232
17	RTS#[07]	Request To Send Serial Channel 8	Output	RS232
18	TXD[08]	Transmit Data Serial Channel 9	Output	RS232
19	RTS#[08]	Request To Send Serial Channel 9	Output	RS232
20	TXD[09]	Transmit Data Serial Channel 10	Output	RS232
21	RTS#[09]	Request To Send Serial Channel 10	Output	RS232
22	TXD[10]	Transmit Data Serial Channel 11	Output	RS232
23	RTS#[10]	Request To Send Serial Channel 11	Output	RS232
24	TXD[11]	Transmit Data Serial Channel 12	Output	RS232
25	RTS#[11]	Request To Send Serial Channel 12	Output	RS232
26	GND			
27	TXD[12]	Transmit Data Serial Channel 13	Output	RS232
28	RTS#[12]	Request To Send Serial Channel 13	Output	RS232
29	TXD[13]	Transmit Data Serial Channel 14	Output	RS232
30	RTS#[13]	Request To Send Serial Channel 14	Output	RS232
31	TXD[14]	Transmit Data Serial Channel 15	Output	RS232
32	RTS#[14]	Request To Send Serial Channel 15	Output	RS232
33	TXD[15]	Transmit Data Serial Channel 16	Output	RS232
34	RTS#[15]	Request To Send Serial Channel 16	Output	RS232
35	RXD[00]	Receive Data Serial Channel 1	Input	RS232
36	CTS#[00]	Clear To Send Serial Channel 1	Input	RS232
37	RXD[01]	Receive Data Serial Channel 2	Input	RS232

Pin	Signal	Description	Direction	Level
38	CTS#[01]	Clear To Send Serial Channel 2	Input	RS232
39	RXD[02]	Receive Data Serial Channel 3	Input	RS232
40	CTS#[02]	Clear To Send Serial Channel 3	Input	RS232
41	RXD[03]	Receive Data Serial Channel 4	Input	RS232
42	CTS#[03]	Clear To Send Serial Channel 4	Input	RS232
43	GND			
44	RXD[04]	Receive Data Serial Channel 5	Input	RS232
45	CTS#[04]	Clear To Send Serial Channel 5	Input	RS232
46	RXD[05]	Receive Data Serial Channel 6	Input	RS232
47	CTS#[05]	Clear To Send Serial Channel 6	Input	RS232
48	RXD[06]	Receive Data Serial Channel 7	Input	RS232
49	CTS#[06]	Clear To Send Serial Channel 7	Input	RS232
50	RXD[07]	Receive Data Serial Channel 8	Input	RS232
51	CTS#[07]	Clear To Send Serial Channel 8	Input	RS232
52	RXD[08]	Receive Data Serial Channel 9	Input	RS232
53	CTS#[08]	Clear To Send Serial Channel 9	Input	RS232
54	RXD[09]	Receive Data Serial Channel 10	Input	RS232
55	CTS#[09]	Clear To Send Serial Channel 10	Input	RS232
56	RXD[10]	Receive Data Serial Channel 11	Input	RS232
57	CTS#[10]	Clear To Send Serial Channel 11	Input	RS232
58	RXD[11]	Receive Data Serial Channel 12	Input	RS232
59	CTS#[11]	Clear To Send Serial Channel 12	Input	RS232
60	GND			
61	RXD[12]	Receive Data Serial Channel 13	Input	RS232
62	CTS#[12]	Clear To Send Serial Channel 13	Input	RS232
63	RXD[13]	Receive Data Serial Channel 14	Input	RS232
64	CTS#[13]	Clear To Send Serial Channel 14	Input	RS232
65	RXD[14]	Receive Data Serial Channel 15	Input	RS232
66	CTS#[14]	Clear To Send Serial Channel 15	Input	RS232
67	RXD[15]	Receive Data Serial Channel 16	Input	RS232
68	CTS#[15]	Clear To Send Serial Channel 16	Input	RS232

Figure 8-1 : HD68 SCSI-3 type front panel connector

The HD68 connector housing is galvanically connected to the front panel.