*The Embedded I/O Company*

TEWS TECHNOLOGIES

# TCP866

## 8 Channel Serial Interface

## RS232 / RS422

## cPCI Module

Version 1.0

## User Manual

Issue 1.2

May 2004


**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

**TEWS TECHNOLOGIES LLC**
1 E. Liberty Street, Sixth Floor
Reno, Nevada 89504 / USA
Phone: +1 (775) 686 6077
Fax : +1 (775) 686 6024
e-mail: usasales@tews.com
www.tews.com

## TCP866-10

8 channel serial RS232, front panel I/O cPCI Module

## TCP866-11

8 channel serial RS232, J2 I/O cPCI Module

## TCP866-20

8 channel serial RS422, front panel I/O cPCI Module

## TCP866-21

8 channel serial RS422, J2 I/O cPCI 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#.



| Issue | Description | Date |
|---|---|---|
| | Preliminary Issue | April 2002 |
| 1.0 | First Issue | May 2002 |
| 1.1 | Technical Specification Upgrade | March 2003 |
| 1.2 | General Revision | May 2004 |

# Table of Contents

# Table of Figures

# 1 Product Description

The TCP866 provides eight channel of high performance serial interface. Four different versions are available. The TCP866-10/-11 provides 8 channels RS232 and the TCP866-20/21 provides 8 channel RS422 interface. Version TCP866-10/-20 supports front panel I/O and version TCP866-11/-21 supports CompactPCI Back I/O (J2 I/O).

The TCP866-10/-11 supports Receive Data (RxD), Transmit Data (TxD), Ready To Send (RTS), Clear To Send (CTS) and GND for each of the eight serial channels. Additionally serial channel 1 and serial channel 2 support Data Set Ready (DSR), Data Terminal Ready (DTR), Data Carrier Detect (DCD) and Ring Detect Indicator (RI).

The TCP866-20/-21 provides RS422 signal levels by differential transmitters and receivers. Transmit Data (TxD+/-) Receive Data (RxD+/-) and GND are provided for each of the eight serial channels. The receiver signal termination (120ohms between RxD+ and RxD-) is provided on the TCP866-20/-21 for each of the eight serial channels.

Each of the eight serial channels of the TPC866 has a 64 byte transmit FIFO and a 64 byte receive FIFO to significantly reduce the overhead required to provide data to and get data from the transmitter and receivers. The FIFO trigger levels are programmable.

Baud rate is individually programmable up to 921.6kBaud for each of the eight channels.

Interrupts are supported. All channels use the PCI interrupt INTA together but for fast interrupt source detection the TCP866 provides a special interrupt status register.

Each RS232/RS422 receiver input and transmitter output is protected against electrostatic discharge (ESD).

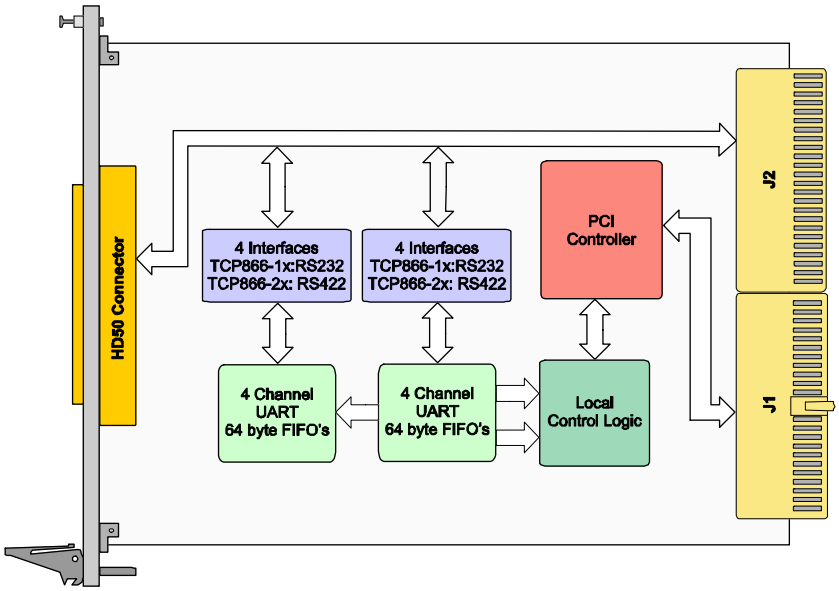


Figure 1-1: Block Diagram

# 2 Technical Specification

| | |
|---|---|
| **Logic Interface** | CompactPCI 3U 32 bit conforming to PICMG 2.0 R3.0 |
| **PCI Target Chip** | PCI9030 (PLX Technology ) |
| **PCI I/O Signaling Voltage Keying** | +3.3V and +5.0V |
| **Module Specific Data** | |
| **Serial Controller** | EXAR ST16C654 Quad Uart |
| **Number of Channels** | 8 |
| **Physical Interface** | TCP866-10/-11 (RS232):<br>TxD, RxD, RTS, CTS, GND<br>additionally for channel 1 and 2 : DTR, DSR, DCD, RI<br>TCP866-20/-21 (RS422):<br>TxD+, TxD-, RxD+, RxD-, GND |
| **Termination** | Only TCP866-20 and TCP866-21<br>120ohms between RxD+ and RxD- of each channel |
| **Transfer Rate** | Each channel programmable up to 921.6 kBaud |
| **FIFO** | 64 byte transmit FIFO and 64 byte receive FIFO per channel |
| **Interrupts** | PCI INTA for all channels and on board interrupt status register |
| **ESD Protection** | ±15kV Human Body Model<br>±8kV IEC 1000-4-2, Contact Discharge (RS232)<br>±15kV IEC 1000-4-2, Air-Gap Discharge (RS232) |
| **I/O Interface** | |
| **Front panel** | TCP866-10: HD50 SCSI-2 type female Connector<br>TCP866-20: HD50 SCSI-2 type female Connector |
| **CompactPCI Back I/O Connector** | TCP866-11: 110pol. CompactPCI back I/O<br><br>TCP866-21: 110pol. CompactPCI back I/O |
| **Operating Data** | |
| **Power Requirements** | 180mA typical @ + 3.3 VDC |
| **Temperature Range** | Operating: – 40°C to + 85°C<br>Storage: – 55°C to + 125°C |
| **MTBF** | TCP866-10: 476774 h<br>TCP866-11: 528127 h<br>TCP866-20: 543470 h<br>TCP866-21: 544476 h |
| **Weight** | 140 g |
| **Size** | Standard 3U CompactPCI |
| **Humidity** | 5 – 95 % non-condensing |

Figure 2-1: Technical Specification

# 3 Local Space Addressing TCP866

The complete register sets of all eight channels are accessible in the I/O space of the TCP866.

## 3.1 Local Memory Space

Not used by the TCP866

## 3.2 Local I/O Space

Address range: PCI Base Address 2 for Local Space 0 + (0x0000 to 0x0048).

| Local I/O SPACE | | |
|---|---|---|
| **Register Name** | **Size** | **Address** |
| Serial Channel 0 | 8 Byte | PCI Base Address 2 + (0x0000 to 0x0007 ) |
| Serial Channel 1 | 8 Byte | PCI Base Address 2 + (0x0008 to 0x000F ) |
| Serial Channel 2 | 8 Byte | PCI Base Address 2 + (0x0010 to 0x0017 ) |
| Serial Channel 3 | 8 Byte | PCI Base Address 2 + (0x0018 to 0x001F ) |
| Serial Channel 4 | 8 Byte | PCI Base Address 2 + (0x0020 to 0x0027 ) |
| Serial Channel 5 | 8 Byte | PCI Base Address 2 + (0x0028 to 0x002F ) |
| Serial Channel 6 | 8 Byte | PCI Base Address 2 + (0x0030 to 0x0037 ) |
| Serial Channel 7 | 8 Byte | PCI Base Address 2 + (0x0038 to 0x003F ) |
| FIFO Ready Register Ch0-Ch3 | Byte | PCI Base Address 2 + 0x0040 |
| FIFO Ready Register Ch4-Ch7 | Byte | PCI Base Address 2 + 0x0044 |
| Interrupt Status Register | Byte | PCI Base Address 2 + 0x0048 |

Figure 3-1: Local Memory Space

### 3.2.1 Register Map TCP866

Each of the eight serial channels of the TCP866 is accessed in the PCI I/O space by two sets of registers. Both register sets have a common register, the Line Control Register (LCR). Bit 7 of the Control Register is used to switch between the two register sets of a channel.

### 3.2.1.1 Register Set of Each Channel

Register Set 1 is accessible only if bit 7 of the Line Control Register (LCR, Address: PCI Base Address 2 + Channel Offset + 0x03) is set to '0'. After reset register Set 1 is accessible.

| I/O SPACE REGISTERS | | | |
|---|---|---|---|
| **PCI Base Address + Channel Offset +** | **READ MODE** | **WRITE MODE** | **SIZE** |
| 0x00 | Receive Holding Register | Transmit Holding Register | Byte |
| 0x01 | Interrupt Enable Register | Interrupt Enable Register | Byte |
| 0x02 | Interrupt Status Register | FIFO Control Register | Byte |
| 0x03 | Line Control Register (LCR) | Line Control Register | Byte |
| 0x04 | Modem Control Register | Modem Control Register | Byte |
| 0x05 | Line Status Register | - | Byte |
| 0x06 | Modem Status Register | - | Byte |
| 0x07 | Scratchpad Register | Scratchpad Register | Byte |

Figure 3-2: Register Set 1

To get access to Register Set 2 of the serial channels bit 7 of Line Control Register must be set to '1'. The Enhance Feature Registers, Xon-1, -2 and Xoff-1, -2 registers are accessible only if LCR is set to 0xBF.

| I/O SPACE REGISTERS | | | |
|---|---|---|---|
| **PCI Base Address + Channel Offset +** | **READ/WRITE** | **Comment** | **SIZE** |
| 0x00 | LSB of Divisor Latch | LCR bit 7 set to '1' | Byte |
| 0x01 | MSB of Divisor Latch | LCR bit 7 set to '1' | Byte |
| 0x02 | Enhanced Feature Register | LCR Register is set to 0xBF | Byte |
| 0x03 | Line Control Register (LCR) | always accessible | Byte |
| 0x04 | Xon-1 Word | LCR Register is set to 0xBF | Byte |
| 0x05 | Xon-2 Word | LCR Register is set to 0xBF | Byte |
| 0x06 | Xoff-1 Word | LCR Register is set to 0xBF | Byte |
| 0x07 | Xoff-2 Word | LCR Register is set to 0xBF | Byte |

Figure 3-3: Register Set 2

## 3.2.2 Special Registers

The TCP866 provides three special register: for fast status detection there are two FIFO Status Registers, one for channel 0 to channel 3 and one for channel 4 to channel 7 and an Interrupt Status Register for all eight channels.

| Register Name | Size | Address |
|---|---|---|
| FIFO Ready Register Ch0-Ch3 | Byte | PCI Base Address 2 + 0x0040 |
| FIFO Ready Register Ch4-Ch7 | Byte | PCI Base Address 2 + 0x0044 |
| Interrupt Status Register | Byte | PCI Base Address 2 + 0x0048 |

Figure 3-4: Special Registers

### 3.2.2.1 FIFO Ready Register Channel 0-3

The FIFO Ready Register provides the real time status of the transmit and receive FIFO's of channel 0 to channel 3. Each TX and RX channel (0-3) has its own 64 byte FIFO. When any of the eight TX/RX FIFO's become empty/full, the status bit associated with the TX/RX function of channel 0-3 is set in the FIFO Ready Register.

| FIFORDY1 Register | | | |
|---|---|---|---|
| **Bit Number** | **Bit Symbol** | **Description** | **Access** |
| 7 (MSB) | TXRDY Channel 3 | If FIFO Ready bits for channel 0-3 are read as '0' the corresponding receive FIFO is above the programmed trigger level or a time-out has occurred. If the bits are read as '1' the receiver is ready and is below the programmed trigger level. | R |
| 6 | TXRDY Channel 2 | | |
| 5 | TXRDY Channel 1 | | |
| 4 | TXRDY Channel 0 | | |
| 3 | RXRDY Channel 3 | If FIFO Ready bits for channel 0-3 are read as '0' the corresponding transmit FIFO is full. This channel will not accept any more transmit data. If the bits are read as '1' one or more empty locations exist in the corresponding FIFO. | R |
| 2 | RXRDY Channel 2 | | |
| 1 | RXRDY Channel 1 | | |
| 0 (LSB) | RXRDY Channel 0 | | |

Figure 3-5: FIFORDY1 Register

### 3.2.2.2 FIFO Ready Register Channel 4-7

The FIFO Ready Register provides the real time status of the transmit and receive FIFO's of channel 4 to channel 7. Each TX and RX channel (4-7) has its own 64 byte FIFO. When any of the eight TX/RX FIFO's become empty/full, the status bit associated with the TX/RX function and channel 4-7 is set in the FIFO Ready Register.

| FIFORDY2 Register | | | |
|---|---|---|---|
| **Bit Number** | **Bit Symbol** | **Description** | **Access** |
| 7 (MSB) | TXRDY Channel 7 | If FIFO Ready bits for channel 4-7 are read as '0' the corresponding receive FIFO is above the programmed trigger level or a time-out has occurred. If the bits are read as '1' the receiver is ready and is below the programmed trigger level. | R |
| 6 | TXRDY Channel 6 | | |
| 5 | TXRDY Channel 5 | | |
| 4 | TXRDY Channel 4 | | |
| 3 | RXRDY Channel 7 | If FIFO Ready bits for channel 4-7 are read as '0' the corresponding transmit FIFO is full. This channel will not accept any more transmit data. If the bits are read as '1' one or more empty locations exist in the corresponding FIFO. | R |
| 2 | RXRDY Channel 6 | | |
| 1 | RXRDY Channel 5 | | |
| 0 (LSB) | RXRDY Channel 4 | | |

Figure 3-6: FIFORDY2 Register

### 3.2.2.3 Interrupt Status Register

The Interrupt Status Register of the TCP866 is a byte wide read register located in the PCI I/O space at address (PCI Base Address 1 + 0x48).

The Interrupt Status Register reflects the interrupt status of the eight serial channels.

| Interrupt Status Register | | | |
|---|---|---|---|
| **Bit Number** | **Bit Symbol** | **Description** | **Access** |
| 7 (MSB) | Interrupt Channel 7 | '1' interrupt is pending<br>'0' no interrupt is pending | R |
| 6 | Interrupt Channel 6 | | |
| 5 | Interrupt Channel 5 | | |
| 4 | Interrupt Channel 4 | | |
| 3 | Interrupt Channel 3 | | |
| 2 | Interrupt Channel 2 | | |
| 1 | Interrupt Channel 1 | | |
| 0 (LSB) | Interrupt Channel 0 | | |

Figure 3-7: Interrupt Status Register

Each of the eight serial channels generates interrupts on the local interrupt 1 of the PCI target chip (the local interrupt 1 is routed to the PCI interrupt INTA by the PCI target chip PCI9030).

If the PCI Interrupt Enable of the PCI target chip is disabled (INTCSR bit 6 is set to '0') the Interrupt status Register can be used as a polling register for interrupts of the eight serial controllers. After RESET the PCI Interrupt is enabled.

Interrupts from the eight serial channels can be individual enabled by the ST16C654 serial controller. After RESET all interrupts are disabled.

# 4 PCI9030 Target Chip

## 4.1 PCI Configuration (CFG) Registers

### 4.1.1 PCI Header

| PCI CFG Register Address | Write '0' to all unused (Reserved) bits | | | | | | | | PCI write able | Read after initialization write access (Hex. Value) |
|---|---|---|---|---|---|---|---|---|---|---|
| | 31 | 24 | 23 | 16 | 15 | 8 | 7 | 0 | | |
| 0x00 | Device ID | | | | Vendor ID | | | | N | xyyy 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 Configuration Registers | | | | | | | | Y | FFFFFF80 |
| 0x14 | PCI Base Address 1 for I/O Mapped Configuration Registers | | | | | | | | Y | FFFFFF81 |
| 0x18 | PCI Base Address 2 for Local Address Space 0 | | | | | | | | Y | FFFFFF81 |
| 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 | Cardbus CIS Pointer | | | | | | | | N | 00000000 |
| 0x2C | Subsystem ID | | | | Subsystem Vendor ID | | | | N | x0zz 1498 |
| 0x30 | PCI Base Address for Local Expansion ROM | | | | | | | | Y | 00000000 |
| 0x34 | Reserved | | | | | | Next Cap P. | | N | 000000 40 |
| 0x38 | Reserved | | | | | | | | N | 00000000 |
| 0x3C | Max_Lat | | Min_Gnt | | Interrupt Pin | | Interrupt Line | | Y[7:0] | 00 00 00 01 |
| 0x40 | Power Management Capabilities | | | | Next Cap Pointer | | Capability ID | | Y | 4801 48 01 |
| 0x44 | Data | | PMCSR Bridge Support Extensions | | Power Management Control/Status | | | | Y | 00 00 0000 |
| 0x48 | Reserved | | Control/Status | | Next_Cap Pointer | | Capability ID | | Y[23:16] | 00 02 00 06 |
| 0x4C | VPD Address | | | | Next_Cap Pointer | | Capability ID | | Y[31:16] | 0000 00 03 |
| 0x50 | VPD Data Register | | | | | | | | Y | 00000000 |

Figure 4-1: PCI Configuration Register Map

- Device-ID and Vendor-ID depend on the TEWS Module

  | | | |
  |---|---|---|
  | Device-ID | 0x**xyyy** | TEWS Module Bus Type & Module ID |
  | Vendor-ID | 0x1**498** | TEWS TECHNOLOGIES GmbH |

- Subsystem-ID and Subsystem-Vendor-ID depend on the TEWS-Module

  | | | |
  |---|---|---|
  | Subsystem-ID | 0x**x0zz** | TEWS Module Bus Type & Board Option |
  | Subvendor-ID | 0x**1498** | TEWS TECHNOLOGIES GmbH |

  x TEWS Module Bus Type Coding:

  | | |
  |---|---|
  | PMC | 0x0 |
  | PC-MIP | 0x1 |
  | CompactPCI | 0x2 |
  | Standard PCI | 0x3 |

  0x**yyy** TEWS Module ID

  0x**zz** TEWS Module Board Option

  Example: TCP866-10

  Device-ID: 0x2362

  Subsystem-ID: 0x200A

### 4.1.2 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 PCI9030 (PCI Target chip) returns zero '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 0x04) to '1'.

#### 4.1.2.1 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' (PCI9030 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 PCI9030.

   For example, if bit 5 of the PCI Base Address register is detected as the first bit set to '1', the PCI9030 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 (128 Byte) is used by the TCP866 as I/O address space.**

#### 4.1.2.2 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' (PCI9030 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 PCI9030.

   For example, if bit 15 of the PCI Base Address register is detected as the first bit set to '1', the PCI9030 is requesting a 32 kilobyte block of memory address space.
4. Write the start address 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 TCP866 as memory address space.**

#### 4.1.2.3 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 PCI9030.

   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 TCP866.**

For further information please refer to the PCI9030 manual which is also part of the TCP866-ED Engineering Documentation.

## 4.2 Local Configuration Registers (LCR)

After reset, the Local Configuration Registers (LCRs) are loaded from the on board 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.

**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** | **Description** |
| 0x00 | Local Address Space 0 Range | 0x0FFFFF81 | I/O space 128Byte |
| 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 | 0x50008080 | 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 | 0x00000021 | CS for all 8 serial channel |
| 0x40 | Chip Select 1 Base Address | 0x00000043 | CS for FIFORDY 1 Register |
| 0x44 | Chip Select 2 Base Address | 0x00000047 | CS for FIFORDY 2 Register |
| 0x48 | Chip Select 3 Base Address | 0x0000004B | CS for Interrupt Status Reg. |
| 0x4C | Interrupt Control/Status | 0x00000041 | 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 | 0x02492252 | GPI/O2 = CS2#<br>GPI/O3 = CS3#<br>all other GPI/O = Output ‘0’ |
| 0x70 | Hidden1 Power Management data select | 0x00000000 | not used |
| 0x74 | Hidden 2 Power Management data scale | 0x00000000 | not used |

Figure 4-2: Local Configuration Registers

# 4.3 Target Configuration EEPROM

After reset, the PCI9030 starts to load the configuration sequence from the on board EEPROM.

This EEPROM contains the following configuration data.

- From 0x00 to 0x27 : PCI – Configuration
- From 0x28 to 0x87 : Local – Configuration

| **Address** | 0x**00** | 0x**02** | 0x**04** | 0x**06** | 0x**08** | 0x**0A** | 0x**0C** | 0x**0E** |
|---|---|---|---|---|---|---|---|---|
| 0x**00 *** | 0x2362 | 0x1498 | 0x0280 | 0x0000 | 0x0702 | 0x0000 | 0x20zz | 0x1498 |
| 0x**10 *** | 0x0000 | 0x0040 | 0x0000 | 0x0100 | 0x4801 | 0x4801 | 0x0000 | 0x0000 |
| 0x**20 *** | 0x0000 | 0x4C06 | 0x0000 | 0x0003 | 0x0FFF | 0xFF81 | 0x0000 | 0x0000 |
| 0x**30 *** | 0x0000 | 0x0000 | 0x0000 | 0x0000 | 0x0000 | 0x0000 | 0x0000 | 0x0001 |
| 0x**40 *** | 0x0000 | 0x0000 | 0x0000 | 0x0000 | 0x0000 | 0x0000 | 0x0000 | 0x0000 |
| 0x**50 *** | 0x5000 | 0x8080 | 0x0000 | 0x0000 | 0x0000 | 0x0000 | 0x0000 | 0x0000 |
| 0x**60 *** | 0x0000 | 0x0000 | 0x0000 | 0x0021 | 0x0000 | 0x0043 | 0x0000 | 0x0047 |
| 0x**70 *** | 0x0000 | 0x004B | 0x0030 | 0x0041 | 0x0078 | 0x0000 | 0x0249 | 0x2252 |
| 0x**80 *** | 0x0000 | 0x0000 | 0x0000 | 0x0000 | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |
| 0x**90 *** | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |
| 0x**A0 *** | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |
| 0x**B0 *** | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |
| 0x**C0 *** | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |
| 0x**D0 *** | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |
| 0x**E0 *** | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |
| 0x**F0 *** | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF | 0xFFFF |

Figure 4-3: EEPROM Values TCP866-xx

| 0x**zz** **TEWS Module Board Option** |
|---|

# 5 Configuration Hints

## 5.1 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.

## 5.2 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 TCP866:**

| | |
|---|---|
| Local Address Space 0 | 16 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 TCP866-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 Programming Hints

## 6.1 Baud Rate Programming Formula

Each of the eight serial channels of the TCP866 contains a programmable Baud Rate Generator. The clock of the ST16C654 can divided by any divisor from 1 to $2^{16}$-1. The divisor can be programmed by the LSB and the MSB of the Divisor Latch Register. After RESET the MCR bit 7 of each channel is default to '0' and the value of LSB and MSB of Divisor Latch Register 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 |
|---|---|---|
| 100 | 25 | 2400 |
| 200 | 50 | 1200 |
| 300 | 75 | 0C00 |
| 600 | 150 | 0600 |
| 1200 | 300 | 0300 |
| 2400 | 600 | 0180 |
| 4800 | 1200 | 00C0 |
| 9600 | 2400 | 0060 |
| 19200 | 4800 | 0030 |
| 28800 | 7200 | 0020 |
| 38400 | 9600 | 0018 |
| 76800 | 19200 | 000C |
| 153600 | 38400 | 0006 |
| 230400 | 57600 | 0004 |
| 460800 | 115200 | 0002 |
| 921600 | 230400 | 0001 |

Figure 6-1: Baud Rate Programming Table

# 7 Pin Assignment TCP866

## 7.1 Front Panel I/O

| HD50 SCSI2 type Connector | TCP866-10 RS232 Interface | TCP866-20 RS422 Interface | Comment |
|---|---|---|---|
| **01** | GND | GND | Serial Channel 0 |
| **02** | TXD0 | TXD0- | |
| **03** | RXD0 | TXD0+ | |
| **04** | RTS0 | RXD0- | |
| **05** | CTS0 | RXD0+ | |
| **06** | GND | GND | Serial Channel 1 |
| **07** | TXD1 | TXD1- | |
| **08** | RXD1 | TXD1+ | |
| **09** | RTS1 | RXD1- | |
| **10** | CTS1 | RXD1+ | |
| **11** | GND | GND | Serial Channel 2 |
| **12** | TXD2 | TXD2- | |
| **13** | RXD2 | TXD2+ | |
| **14** | RTS2 | RXD2- | |
| **15** | CTS2 | RXD2+ | |
| **16** | GND | GND | Serial Channel 3 |
| **17** | TXD3 | TXD3- | |
| **18** | RXD3 | TXD3+ | |
| **19** | RTS3 | RXD3- | |
| **20** | CTS3 | RXD3+ | |
| **21** | GND | GND | Serial Channel 4 |
| **22** | TXD4 | TXD4- | |
| **23** | RXD4 | TXD4+ | |
| **24** | RTS4 | RXD4- | |
| **25** | CTS4 | RXD4+ | |
| **26** | GND | GND | Serial Channel 5 |
| **27** | TXD5 | TXD5- | |
| **28** | RXD5 | TXD5+ | |
| **29** | RTS5 | RXD5- | |
| **30** | CTS5 | RXD5+ | |
| **31** | GND | GND | Serial Channel 6 |
| **32** | TXD6 | TXD6- | |
| **33** | RXD6 | TXD6+ | |
| **34** | RTS6 | RXD6- | |
| **35** | CTS6 | RXD6+ | |

| HD50 SCSI2 type Connector | TCP866-10 RS232 Interface | TCP866-20 RS422 Interface | Comment |
|---|---|---|---|
| **36** | GND | GND | Serial Channel 7 |
| **37** | TXD7 | TXD7- | |
| **38** | RXD7 | TXD7+ | |
| **39** | RTS7 | RXD7- | |
| **40** | CTS7 | RXD7+ | |
| **41** | GND | GND | Termination Supply |
| **42** | +5V | +5V- | Termination Supply |
| **43** | CD0 | Not used | Data Carrier Detect CH0 |
| **44** | DTR0 | Not used | Data Terminal Ready CH0 |
| **45** | RI0 | Not used | Ring Indicator CH0 |
| **46** | DSR0 | Not used | Data Set Ready CH0 |
| **47** | CD1 | Not used | Data Carrier Detect CH1 |
| **48** | DTR1 | Not used | Data Terminal Ready CH1 |
| **49** | RI1 | Not used | Ring Indicator CH1 |
| **50** | DSR1 | Not used | Data Set Ready CH1 |

Figure 7-1: Front Panel I/O

# 7.2 CompactPCI Back I/O

## 7.2.1 J2 I/O of TCP866-11

| Pos. | F | E | D | C | B | A |
|---|---|---|---|---|---|---|
| **22** | GND | not used | not used | not used | not used | not used |
| **21** | GND | not used | not used | not used | not used | not used |
| **20** | GND | not used | not used | not used | not used | not used |
| **19** | GND | not used | not used | not used | not used | not used |
| **18** | GND | not used | not used | not used | not used | not used |
| **17** | GND | not used | not used | not used | not used | not used |
| **16** | GND | not used | not used | not used | not used | not used |
| **15** | GND | not used | not used | not used | not used | not used |
| **14** | GND | +5V | +5V | +3,3V | +3,3V | +3,3V |
| **13** | GND | GND | TXD0 | RXD0 | RTS0 | CTS0 |
| **12** | GND | GND | TXD1 | RXD1 | RTS1 | CTS1 |
| **11** | GND | GND | TXD2 | RXD2 | RTS2 | CTS2 |
| **10** | GND | GND | TXD3 | RXD3 | RTS3 | CTS3 |
| **9** | GND | GND | TXD4 | RXD4 | RTS4 | CTS4 |
| **8** | GND | GND | TXD5 | RXD5 | RTS5 | CTS5 |
| **7** | GND | GND | TXD6 | RXD6 | RTS6 | CTS6 |
| **6** | GND | GND | TXD7 | RXD7 | RTS7 | CTS7 |
| **5** | GND | GND | +5V | CD0 | DTR0 | RI0 |
| **4** | GND | DSR0 | CD1 | DTR1 | RI1 | DSR1 |
| **3** | GND | not used | not used | not used | not used | not used |
| **2** | GND | not used | not used | not used | not used | not used |
| **1** | GND | not used | not used | not used | not used | VI/O |

Figure 7-2: J2 I/O of TCP866-11

### 7.2.2 J2 I/O of TCP866-21

| Pos. | F | E | D | C | B | A |
|---|---|---|---|---|---|---|
| **22** | GND | not used | not used | not used | not used | not used |
| **21** | GND | not used | not used | not used | not used | not used |
| **20** | GND | not used | not used | not used | not used | not used |
| **19** | GND | not used | not used | not used | not used | not used |
| **18** | GND | not used | not used | not used | not used | not used |
| **17** | GND | not used | not used | not used | not used | not used |
| **16** | GND | not used | not used | not used | not used | not used |
| **15** | GND | not used | not used | not used | not used | not used |
| **14** | GND | +5V | +5V | +3,3V | +3,3V | +3,3V |
| **13** | GND | GND | TXD0- | TXD0+ | RXD0- | RXD0+ |
| **12** | GND | GND | TXD1- | TXD1+ | RXD1- | RXD1+ |
| **11** | GND | GND | TXD2- | TXD2+ | RXD2- | RXD2+ |
| **10** | GND | GND | TXD3- | TXD3+ | RXD3- | RXD3+ |
| **9** | GND | GND | TXD4- | TXD4+ | RXD4- | RXD4+ |
| **8** | GND | GND | TXD5- | TXD5+ | RXD5- | RXD5+ |
| **7** | GND | GND | TXD6- | TXD6+ | RXD6- | RXD6+ |
| **6** | GND | GND | TXD7- | TXD7+ | RXD7- | RXD7+ |
| **5** | GND | GND | +5V | not used | not used | not used |
| **4** | GND | not used | not used | not used | not used | not used |
| **3** | GND | not used | not used | not used | not used | not used |
| **2** | GND | not used | not used | not used | not used | not used |
| **1** | GND | not used | not used | not used | not used | VI/O |

Figure 7-3: J2 I/O of TCP866-21