

TCPCI120

Hardware

Installation Manual

Issue 1.00 dated 09.11.2007

MSC Tuttlingen GmbH

Rudolf-Diesel-Straße 17
D-78532 Tuttlingen

Tel. +49 7461 925 200

Fax +49 7461 925 268

E-Mail sales@msc-tuttlingen.de

www.msc-ipc.com



Contents

1	TCPCI120 Technical Specification	3
1.1	Tool Monitoring System Overview	3
1.2	The Tool Monitoring 'System' Concept	3
2	PCI-Board Installation.....	4
3	TTBUS Networking.....	8
3.1	TTBUS Wiring	8
4	Profibus Networking.....	9
4.1	Profibus Cycle-Time Considerations.....	9
4.2	The output telegram from the Profibus Master to the TCPCI120	9
4.3	The input telegram from TCPCI120 to the Profibus Master.....	10
4.4	Il file GSD di Profibus TPCI0A0B.GSD;	11

Table of Figures

Fig. 2-1: Software	4
Fig. 2-2: Browse for the Driver Location	5
Fig. 2-3: Hardware Installation.....	5
Fig. 2-4: Driver	6
Fig. 2-5: Device Manager	6
Fig. 2-6: Driver Information.....	7
Fig. 3-1: TTBUS Wiring.....	8

TECHNA-CHECK® is a registered trademark of Techna-Tool Inc.,
Hartland Wisconsin USA.

TCPCI120

Hardware Installation Manual

Article no. 68 36 260

© Copyright MSC Tuttlingen GmbH, 78532 Tuttlingen, 2007

Subject to change without notice.

1 TCPCI120 Technical Specification

Technical Specification

PCI	PCI rev. 2.2 - 32bit 33/66MHz
Operating System	Windows XP
Minimum System	1Ghz Pentium 51 2MB RAM
Profibus-Dp	DPV1 Interface (9pin Dsub)
Proprietary Bus	TTBUS (Phoenix Connector)

1.1 Tool Monitoring System Overview

The main component of the PCI system is a PCI card which will always need to be inserted into a computer. The PCI card could be inserted into a CNC controller which is Windows XP based and has available PCI slots. The card could also be used in a stand alone PC (desktop) or Industrial PC installed in the machine. The TCPCI120 interfaces the Tool Monitoring System to the Machine NC-Controller and to different types of Tool Monitor sensors. The TCPCI120 is thus basically a communication controller; the actual Tool-Monitoring is realized by the application TTMON, running on the PC. TTMON implements a maximum of a 20 channel (spindle) Tool Monitoring System with the capability of monitoring 128 cuts per channel.

1.2 The Tool Monitoring 'System' Concept

The Tool Monitoring System provides a common interface to different types of Tool Monitoring Sensors. Monitoring is based on either Power or Vibration. The NC-controller provides control signals (Start, Learn and Reset), Cut Number and in some cases Measurement Values to the Tool Monitoring System. The Tool Monitoring System returns Alarms in case of Tooling Faults. In this way the details about the operation of the various Tool Monitoring Functions are effectively hidden by the NC-controller. The actual operation of the Tool-Monitoring PC-application TTMON is covered in a separate user manual.

2 PCI-Board Installation

The TCPCI120 Board is a Plug 'n' Play (PnP) board and will be recognized by Windows XP during the first operating system boot after the board has been installed. The TCPCI120 Board uses a Windows WDM driver supplied by PLX Corporation (the designer of the PCI-interface chip). The PLX-supplied PciSdk.inf file provides Windows with the necessary information to install the driver.

When the PC boots and the board is found you should see this:

1. Windows detects the new hardware and generates a "New Hardware Found" message box. Acknowledge the message box.
2. Windows displays the "Add New Hardware" Wizard, which will search for a suitable driver for the board.
3. The board is recognized as a PCI Bridge or Other PCI Bridge Device. Click Next.



Fig. 2-1: Software

4. Select Install from a list or specific location (Advanced). Click Next

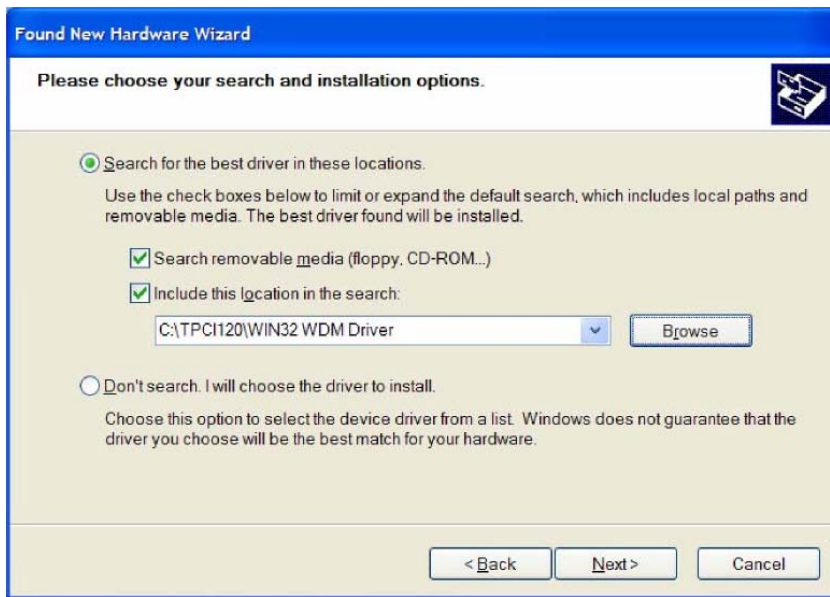


Fig. 2-2: Browse for the Driver Location

5. Browse for the Driver Location. Click Next.



Fig. 2-3: Hardware Installation

6. The driver was submitted to Microsoft for so called Driver Signing. Please Select "Continue Anyway".



Fig. 2-4: Driver

7. Driver installed successfully.

The driver can now be found in device manager as Other devices - PLC Custom (OEM) PCI 9056 Board.

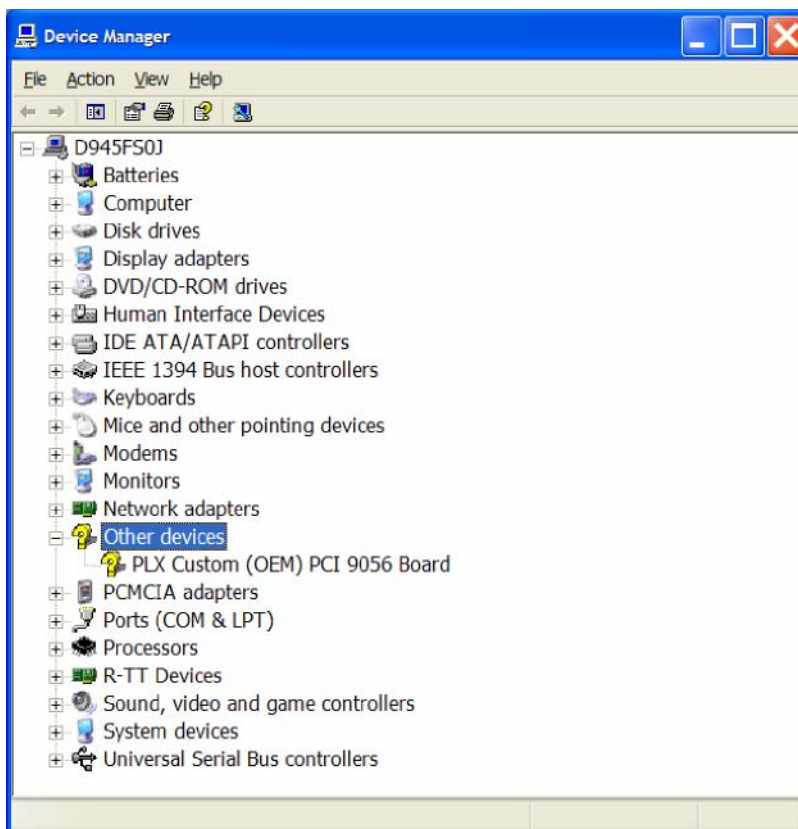


Fig. 2-5: Device Manager

Driver Information:

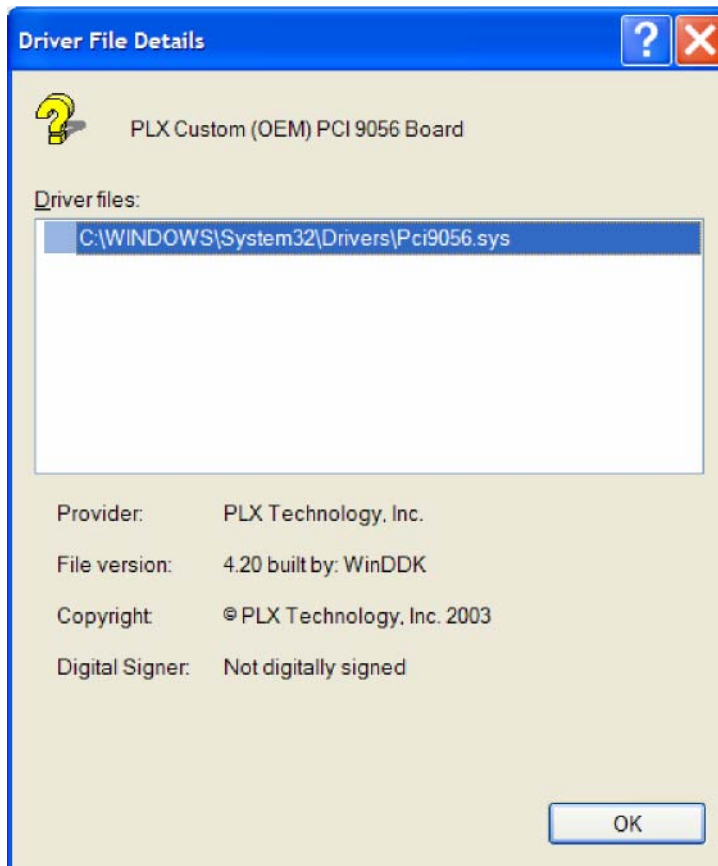


Fig. 2-6: Driver Information

3 TTBUS Networking

The TTBUS is a Proprietary Communication Bus designed to interface multiple Measurement Transducers and/or Digital I/O units to the Tool-Monitoring-System. The TTBUS is based upon traditional RS485 hardware.

Today 3 different TTBUS devices have been developed.

- PWM350T 3-Phase Power Transducer
- VM100T Vibration Sensor Interface Unit
- I/O100T Digital I/O Unit.

In the future other type of sensors may be added.

Detailed specifications of the TTBUS units are found in the appendices of this manual. Each TTBUS unit is assigned a unique address on the network. The address is programmed by 2 BCD switches located on the front of the transducers. The Tool Monitoring System automatically lo-cates transducers on the TTBUS. A Channel Mapping menu in TTMON is used to map the differ-ent transducers to the different channels.

3.1 TTBUS Wiring

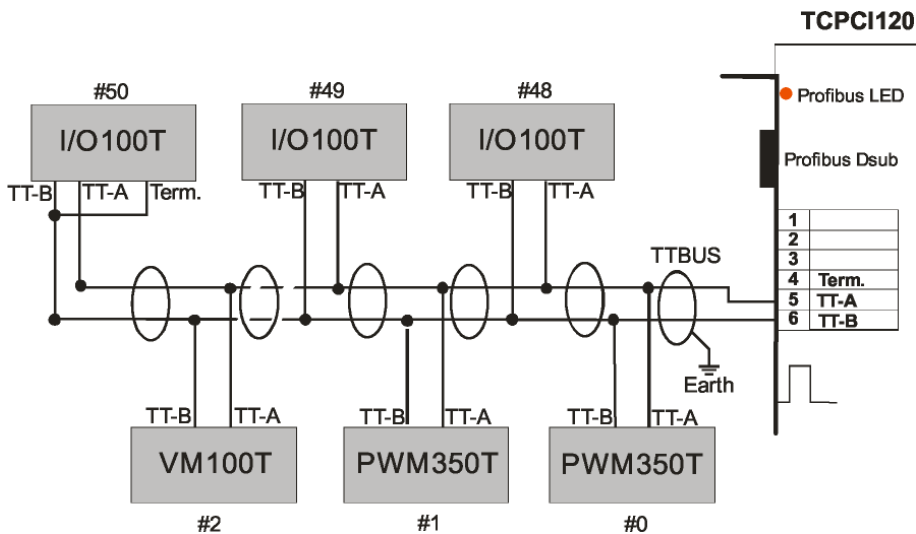


Fig. 3-1: TTBUS Wiring

Important note:

Please use good-quality low-resistance twisted and shielded cable earth-connected at one or both ends for the TTBUS network. Last unit in the TTBUS network chain must be terminated. Termination is possible in all units by adding external wire. Make the stubs as short as possible.

4 Profibus Networking

4.1 Profibus Cycle-Time Considerations

The Profibus cycle time should be a maximum of 10-15 ms (milliseconds) equal to 67-100 profibus telegram transfers per second. To achieve this a Profibus transmission speed (baud rate) of 1Mb and higher is probably required. If this requirement is not met the synchronization from cycle to cycle is affected and also measurement accuracy may be lost (if the measurement value is supplied from the Profibus network).

4.2 The output telegram from the Profibus Master to the TCPCI120

The output telegram length is always 80 bytes - 4 bytes for each channel. Data is always sent for 20 channels no matter how many channels are actually used. Data sent for channels not present should be zero. The purpose of the output telegram is to supply control signals, cut number and possibly measurement value to the TCPCI120 Tool Monitoring System.

Telegram Format:

```
BitFlags#1, Measurement#1, CutNumber#1 ,BitFlags#2, Measurement#2,
CutNumber#2, ..... BitFlags#20, Measurement#20, CutNumber#20
```

Telegram Data-Byte Numbering:

Byte No

0	BitFlags#1	Channel #1 – 8 bits
1, 2	Measurement#1	Channel #1 – 16 bits
3	CutNumber#1	Channel #1 – 8 bits
4	BitFlags#2	Channel #2 – 8 bits
5, 6	Measurement#2	Channel #2 – 16 bits
7	CutNumber#2	Channel #2 – 8 bits
.....		
76	BitFlags#20	Channel #20 – 8 bits
77, 78	Measurement#2 0	Channel #20 – 16 bits
79	CutNumber#2 0	Channel #20 – 8 bits

BitFlags# - b7b6b5bb3b2b1b0

```
#define PROFIBUS_MODE_MASK
```

```
(PROFIBUS_MODE1 | PROFIBUS_MODE2 | PROFIBUS_MODE3) // b6b5b4
```

```
// PROFIBUS - bit_signals -
```

```
Externally Generated Signals - Inputs
```

```
#define START_SIGNAL_ACTIVATED 0x0 1 // b0 = Start Signal
#define LEARN_SIGNAL_ACTIVATED 0x02 // b1 = Learn Signal
#define RESET_ALARM_SIGNAL_ACTIVATED 0x0 4 // b2 = Reset Signal
#define PROFIBUS_SPARE 0x0 8 // b3 = not used
#define PROFIBUS_MODE1 0x1 0 // b4 = Profibus mode
#define PROFIBUS_MODE2 0x2 0 // b3 = Profibus mode
#define PROFIBUS_MODE3 0x4 0 // b2 = Profibus mode
#define PROFIBUS_UNIT_PRESENT 0x8 0 // b1 = Channel Present
```

Measurement# - 2 Byte MSB, LSB

16 bit measurement value 0 - 1000 decimal = 0.0 - 100.0%

CutNumber# - 1 Byte

4.3 The input telegram from TCPCI120 to the Profibus Master

The input telegram (to the Profibus Master) is always 20 bytes long – 1 byte for each channel. Channel #1 is first and Channel #20 is the last byte. The purpose of the inputs is to report Alarms and other status information to the master (NC controller).

Telegram Format:

InputFlags#1, InputsFlags#2 InputFlags#20

InputFlags -

b7b6b5bb3b2b1b0

			= Po Measured or Touched (BK MICRO)
#define ACTIVE_READY	0x01	// b0	
#define SPARE1	0x02	// b1 = bit not used	
#define TOUCHED	0x04	// b2 = Touched	
#define IDLE_ALARM	0x08	// b3 = IDLE_ALARM	
#define BLUNTCOUNT_ALARM	0x10	// b4 = BLUNTCOUNT_ALARM	
#define MISSING_ALARM	0x20	// b5 = MISSING_ALARM	
#define BLUNT_ALARM	0x40	// b6 = BLUNT_ALARM	
#define BREAK_ALARM	0x80	// b7 = BREAK_ALARM	

ACTIVE_READY

This bit is set when the Tool Monitoring becomes active.

Example1: Start Signal has been activated and Idle Power calculated.

Example2: Target tool has been checked for presence – future BK Mikro application.

In some cases cycle time can be saved by waiting for this bit to get activated before the tool feeds towards the target. The alternative (or maybe traditional way) is to introduce a fixed delay large enough until Idle Power has been calculated.

SPARE1

Not used.

TOUCHED

Is used with the Touch-Limit function and set when the tool touches the part – signal reaches the touch-limit.

IDLE_ALARM

Signals the presence of an IDLE_ALARM.

BLUNTCOUNT_ALARM

Signals the presence of a BLUNTCOUNT_ALARM.

MISSING_ALARM

Signals the precense of a MISSING_ALARM.

BLUNT_ALARM

Signals the precense of a BLU NT_ALARM.

BREAK_ALARM

Signals the precense of a BREAK_ALARM.

4.4 Il file GSD di Profibus TPCI0A0B.GSD;

```

; =====
; Techna Tool Inc.
;
; File : TPCI0A0B.GSD
; Revision : 1.0
; Last Modification : 05/09/2005
; =====
;
;#Profibus_DP
; General device information
GSD_Revision          = 1
Vendor_Name           = "Techna Tool Inc."
Model_Name            = "TPCI120"
Revision              = "V1.0"
Ident_Number          = 0x0A0B
Protocol_Ident        = 0          ; 0 = PROFIBUS-DP only
Station_Type          = 0          ; 0 = DP-Slave
FMS_supp              = 0          ; FMS is not supported
Hardware_Release      = "A1"
Software_Release      = "V1.0"

; Supported baudrates
9.6_supp              = 1
19.2_supp             = 1
45.45_supp            = 1
93.75_supp            = 1
187.5_supp           = 1
500_supp              = 1
1.5M_supp             = 1
3M_supp               = 1
6M_supp               = 1
12M_supp              = 1

; MaxTsdr default values for supported baudrates
MaxTsdr_9.6           = 60
MaxTsdr_19.2          = 60
MaxTsdr_45.45         = 60
MaxTsdr_93.75         = 60
MaxTsdr_187.5         = 60
MaxTsdr_500           = 100
MaxTsdr_1.5M          = 150
MaxTsdr_3M            = 250
MaxTsdr_6M            = 450
MaxTsdr_12M           = 800

; General supported features
Redundancy            = 0          ; Redundancy not supported
Repeater_Ctrl_Sig     = 2          ; RTS Signal with TTL level
24V Pins              = 0          ;
Implementation_Type    = "ASIC_solution, VPC3+"

; DP Slave related information
Freeze_Mode_supp      = 0          ; Freeze-Mode not supported
Sync_Mode_supp        = 0          ; Sync.-Mode not supported
Auto_Baud_supp        = 1          ; Automatic baud control supported
Max_Diag_Data_Len     = 6
Set_Slave_Add_supp    = 0          ; Set Slave address not supported
User_Prm_Data_Len     = 05         ;
Min_Slave_Intervall   = 5          ; 500us
Slave_Family           = 1@TT@TPCI

; Modules information
Modular_Station       = 0
Max_Module            = 1
Max_Input_Len         = 20
Max_Output_Len        = 80
Max_Data_Len          = 100

Module = "80 Byte out/ 20 Byte In" 0x2f,0x2f,0x2f,0x2f,0x2f,0x1f,0x13

EndModule

```

