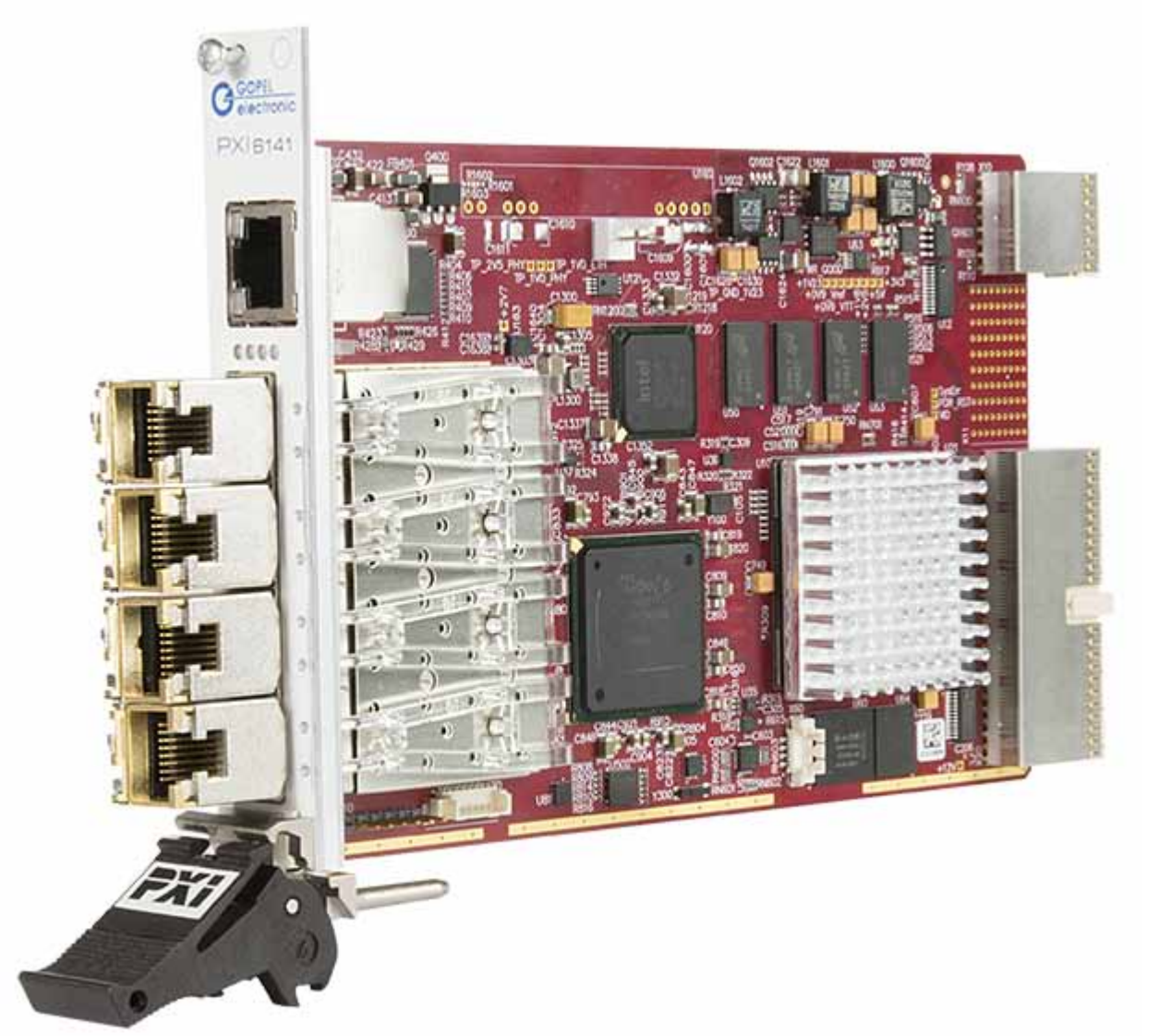
PXI 6141 (Ethernet Board)

User Manual (Translation of Original Docu)

Document Version 1.0



GOPEL electronics

Enjoy Testing

The software described in this manual as well as the manual itself are supplied under license and may be used or copied only in accordance with the terms of the license.
The customer may make one copy of the software for safety purposes.

The contents of the manual is subject to change without prior notice and is supplied for information only.
The hardware and software might be modified also without prior notice due to technical progress.

In case of inaccuracies or errors appearing in this manual, GOEPEL electronic GmbH assumes no liability or responsibility.

Without the prior written permission of GOEPEL electronic GmbH, no part of this documentation may be transmitted, reproduced or stored in a retrieval system in any form or by any means as well as translated into other languages (except as permitted by the license).

GOEPEL electronic GmbH is neither liable for direct damages nor consequential damages from the company's product applications

Contact:

GOEPEL electronic GmbH

49 3641 6896 0 Fon
49 3641 686 944 Fax

ATS_support@goepel.com
Goeschwitzer Str. 58/60 · D-07745 Jena/ Thueringen · www.goepel.com

Issue: July 2016 Printed: 08.07.2016

1 Board Installation

1.1 Hardware Installation

Before beginning with the hardware installation, you have to ensure that your system is switched off and disconnected from the mains supply.

Please refer also to the User Manual of your PXI system for additional installation instructions that possibly have to be followed.

Electro Static Discharge (ESD) can harm your system and destroy electronic components. This can lead to irreparable damage on both the controller board and the system hosting the board as well as to unexpected malfunction of your test system.
Therefore do not touch the board surface or any connector pins and electronic components.

The PXI™ system is to be opened according to its conditions. A free slot is to be selected in your system. Now, the slot cover is to be taken away from the slot selected. To do this, unscrew the fixation screw(s) and remove the cover from the slot.

(If it is necessary to exchange transceiver modules,
pay attention to the general rules to avoid electro static discharging, see the warning above.
Transceiver modules may be removed or mounted with the power switched on, but we recommend not to do that!
Additionally, the right alignment is absolutely required.)

Insert the board carefully into the prepared slot. For PXI boards, use the lever at the front plate in order to push in the board finally.

When the board has been inserted properly, it is to be fixed by means of the screw(s) at the front plate.
Now, the board has been installed correctly.

Afterwards, carry out the operations required at the system to make it ready for operation anew.

1.2 Driver Installation

1.2.1 Windows Device Driver

PXI 6141 boards can be operated under Windows® XP as well as under Windows® 7/ 32bit and Windows® 7/ 64bit.

Due to the plug and play capability of Windows®, for every newly recognized hardware component a driver installation is started automatically via the hardware assistant.
The hardware assistant can carry out the installation of the device driver by using the *inf* file contained on the enclosed CD.

It is not absolutely essential to restart the system.

The following step is only required in case you do not use the G-API.

If you want to create your own software for the boards, you possibly need additional files for user specific programming (**.LLB*, **.H*). These files are not automatically copied to the computer and have to be transferred individually from the supplied CD to your development directory.

After Hardware Installation/ Driver Installation, you may check whether the boards have been embedded properly by the system:

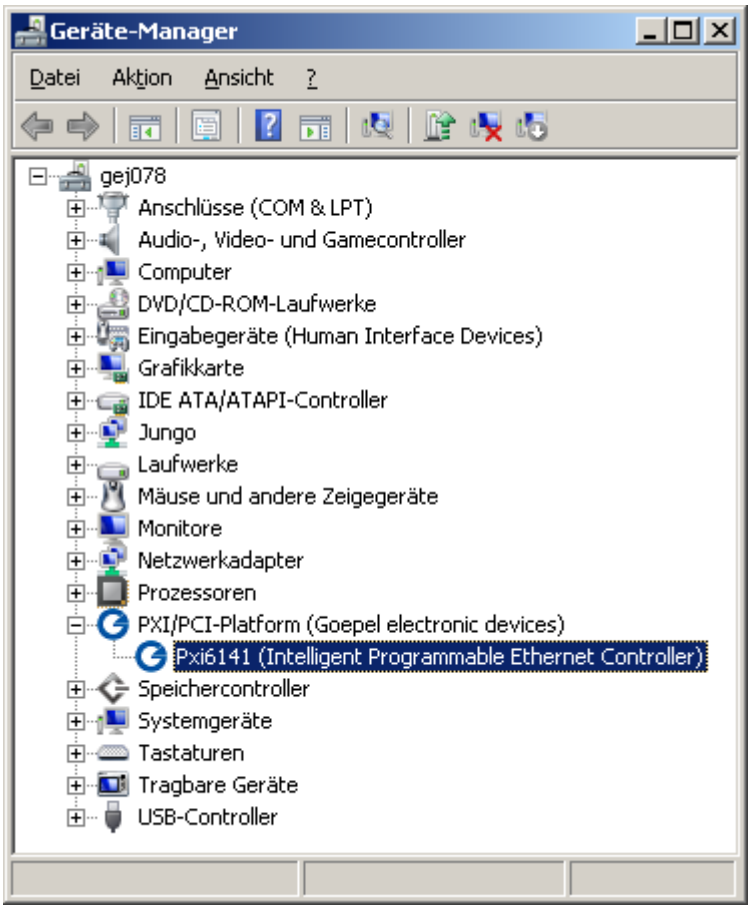


Figure 1-1: Windows

1.2.2 Ethernet

If the Ethernet interface is used for communication with the control PC, there is no driver installation required.

The device can be directly addressed via the IP Address.
This IP Address can be changed by the HardwareExplorer.
The newly set IP Address becomes effective after a restart.:

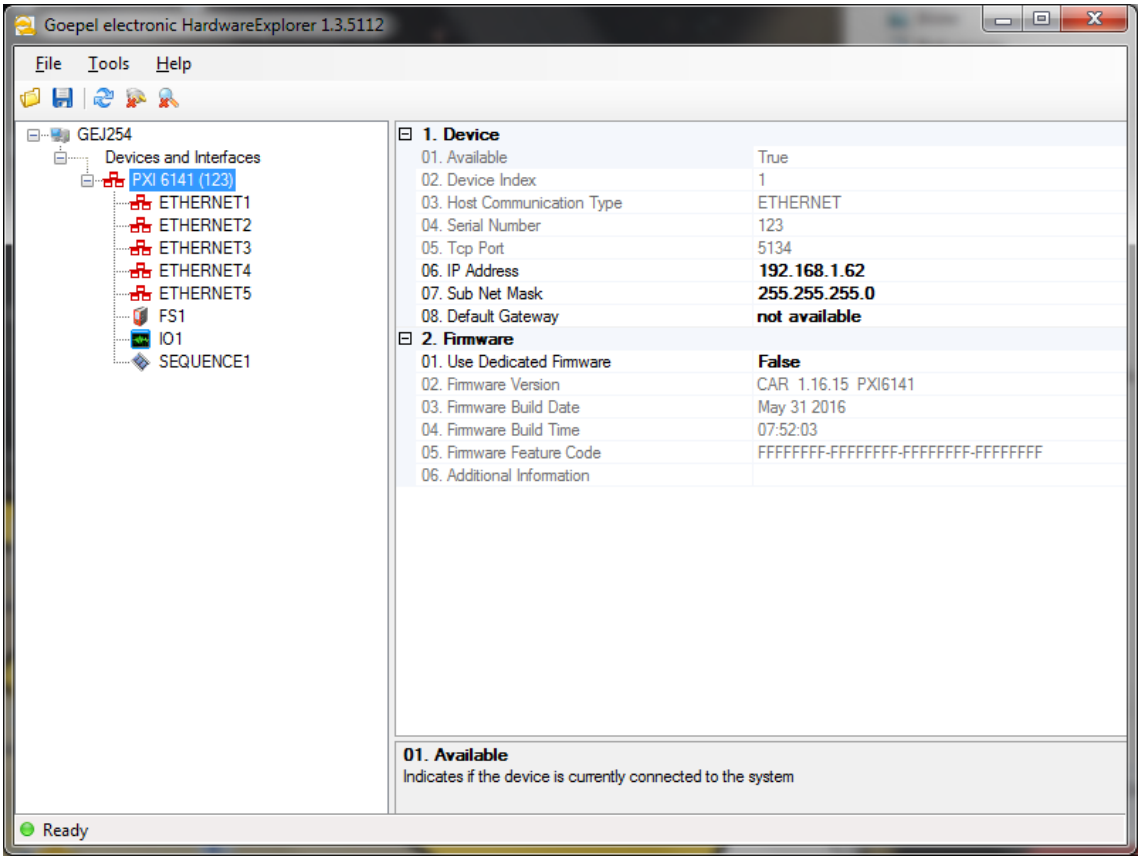


Figure 1-2: IP Address in the GOEPEL electronic HardwareExplorer

1.3 Notes on the Firmware

1.3.1 Firmware Update

In the course of the technical progress, it can be necessary from time to time “to treat” your hardware to a new Firmware release.
Proceed as follows:

- Download the current Firmware Update-File from *genesis.goepel.com*
- Open the GOEPEL HardwareExplorer
- On “Card” (PXI6141) select “Flash Firmware” with the right mouse button
- On “Flash Firmware” select the Update-File with the left mouse button and execute it (e.g. by double clicking)
- After finishing of the progress bar in the “Flashing...” window, press the “OK” button in the following “Success” window with the left mouse button

Please ensure that you use the right Firmware issue when updating the Firmware.
Installing the wrong Firmware issue may result in the loss of functionality and can cause your application to malfunction.

In such events, reinstallation of the correct Firmware issue will restore the functionality.

Find the Interface options supported by the Firmware in the following table:

Software Interface	Hardware Interface (plug-in place)
Ethernet1	ETH0
Ethernet2	SFP0
Ethernet3	SFP1
Ethernet4	SFP2
Ethernet5	SFP3
CAN1	SFP0
CAN2	SFP1
CAN3	SFP2
CAN4	SFP3
LIN1	SFP0
LIN2	SFP1
LIN3	SFP2
LIN4	SFP3
IO1	-
FS1	-
Sequence1	-
UserCode1	-
Net2Run1	-
Net2Run2	-
Net2Run3	-
Net2Run4	-

ETH = Ethernet RJ45
SFP = Small Formfactor pluggable (Module bay)

Please note: Not all interfaces are useable at the same time.
Some interfaces are optional and require specific transceivers or expansion modules to be installed on the card, as well as the corresponding licensing option to be unlocked, in order to use them.

For example, CAN-FD is an additional licensing option.
If this option is not installed on your controller card, all available CAN interfaces will function according to CAN 2.0 specification regardless of the Firmware issue installed.

Please contact our sales department or technical support for questions to the available license and hardware options.

2 Hardware

2.1 Definition

The PXI 6141 GOEPEL electronic GmbH Ethernet board is a programmable, intelligent multibus controller providing various communication interfaces for vehicle network testing and general control applications.

SFP modules can be implemented to use the board for all supplied interfaces.

SFP Transceiver modules - overview:

- Ethernet 10/100/1000Mbit/s
- BroadR-Reach (OABR) 100Mbit/s
- CAN-FD (presently not available, yet)
- LIN (presently not available, yet)
- Reduced Twisted Pair Gigabit Ethernet (RTPGE) (presently not available, yet)

Generally, PXI 6141 controller boards can provide the following features:

- Trace data acquisition on all interfaces with precise hardware time stamps
- Trace data generation in the pcap-ng format
- Cyclic and event-based transmission of Unicast, Multicast and Broadcast Messages
- Diagnostics over Ethernet (DoIP)
- Gateway to CAN-FD and LIN
- Up to 4x Ethernet 10/100/1000Mbit/s
- Up to 4x BroadR-Reach
- Up to 4x CAN-FD
- Up to 4x LIN
- Up to 4x Reduced Twisted Pair Gigabit Ethernet (RTPGE) (as soon as the Transceiver will be available)
- 600MHz Power PC with 512Mb RAM, 256MB Flash
- High flexibility by plug-in Transceiver modules
- PXI 6141 board control via PXI or Ethernet
- The 1 Gbit Ethernet interface at the Front panel is also useable as volume data and debug interface
- Visualization of the operating states by four LEDs arranged at the front panel

2.2 Technical Data

2.2.1 General

The PXI 6141 controller board is a slot-in board developed for the PXI™ bus. PCI eXtensions for Instrumentation (PXI) is a modular instrumentation platform originally introduced in 1997 by National Instruments and now promoted by the PXI Systems Alliance (PXISA).

PXI™ is based on the CompactPCI™ bus, and it offers all of the benefits of the PCI architecture including performance, industry adoption and COTS technology. PXI adds a rugged CompactPCI mechanical form-factor, an industry consortium that defines hardware, electrical, software, power and cooling requirements, leaving nothing to chance. Then PXI™ adds integrated timing and synchronization that is used to route synchronization clocks, and triggers internally. PXI™ is a future-proof technology, and is designed to be simply and quickly reprogrammed as test, measurement, and automation requirements change.

The PXI 6141 controller board operates as PXI slave, therefore the board may be plugged into any desired slot of a PXI chassis (except slot 1). The PCI Plug & Play auto detection mechanism is supported by the PXI 6141 controller board. No jumper configuration is needed for PXI integration.

CompactPCI and PXI products are interchangeable, i.e. they can be used in either CompactPCI or PXI chassis, but installation in the alternate chassis type may eliminate certain clocking and triggering features. So for example you could mount a CompactPCI network interface controller in a PXI rack to provide additional network interface functions to a test stand. Conversely, a PXI module installed in a CompactPCI chassis would not utilize the additional clocking and triggering features of the PXI module.

The PXI 6141 controller board is already prepared for use in a PXI Express hybrid slot, supporting PCI as well as PCI Express.

2.2.2 Dimensions

The PXI 6141 controller board is a 3U standard module and occupies one slot width.

Board dimensions without bracket and handle:

- PXI 6141: 160 mm x 100 mm (L x W)

2.3 Structure and Function

2.3.1 General

The core of the PXI 6141 controller board builds a strong 600MHz AMCC 460EX PowerPC. This dual-issue, superscalar 32bit RISC CPU is based on the Book-E enhanced PowerPC architecture. With features including out-of-order execution, dynamic branch prediction and a highly pipelined double precise floating-point unit, this processor provides the calculation power required for processing complex Residual bus simulation on multiple bus interfaces. Furthermore, the controller comes equipped with a 512MB fast 400MHz DDR2 RAM and 256MB Flash memory, of which over 80% is available for user programs..

The PXI 6141 controller board has been designed as highly flexible multibus controller platform.

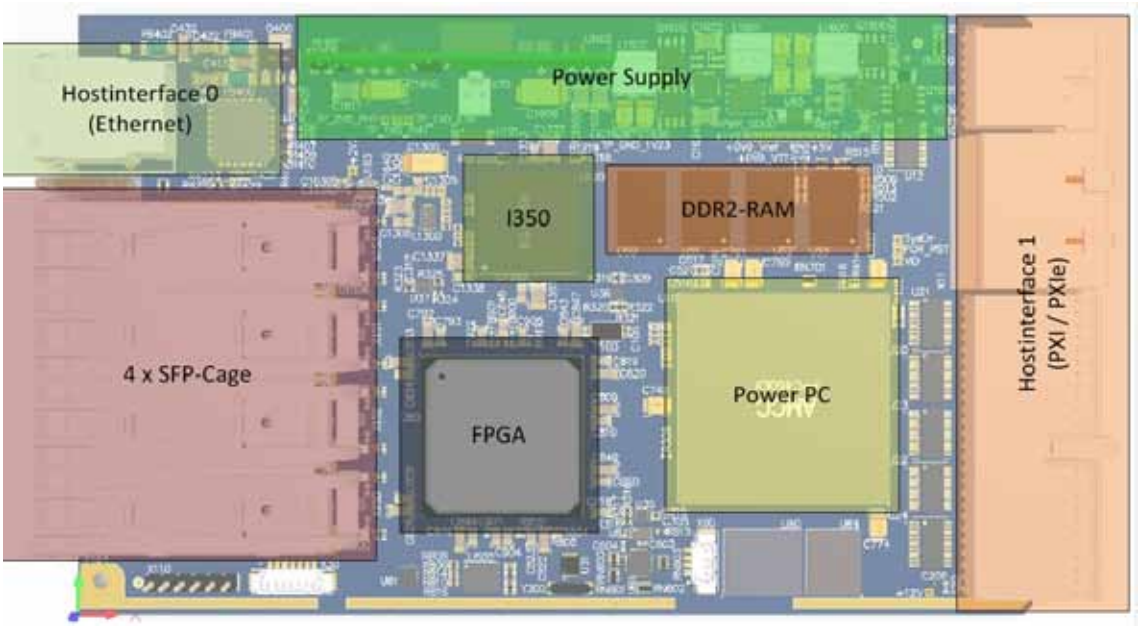


Figure 2-1: PXI 6141 – Schematic Representation

Figure 2-2: PXI 6141 – Interface overview

0	SFP Slot 0
1	SFP Slot 1
2	SFP Slot 2
3	SFP Slot 3
4	Host interface 0 (Ethernet)
5	Host interface 1 (PXI)

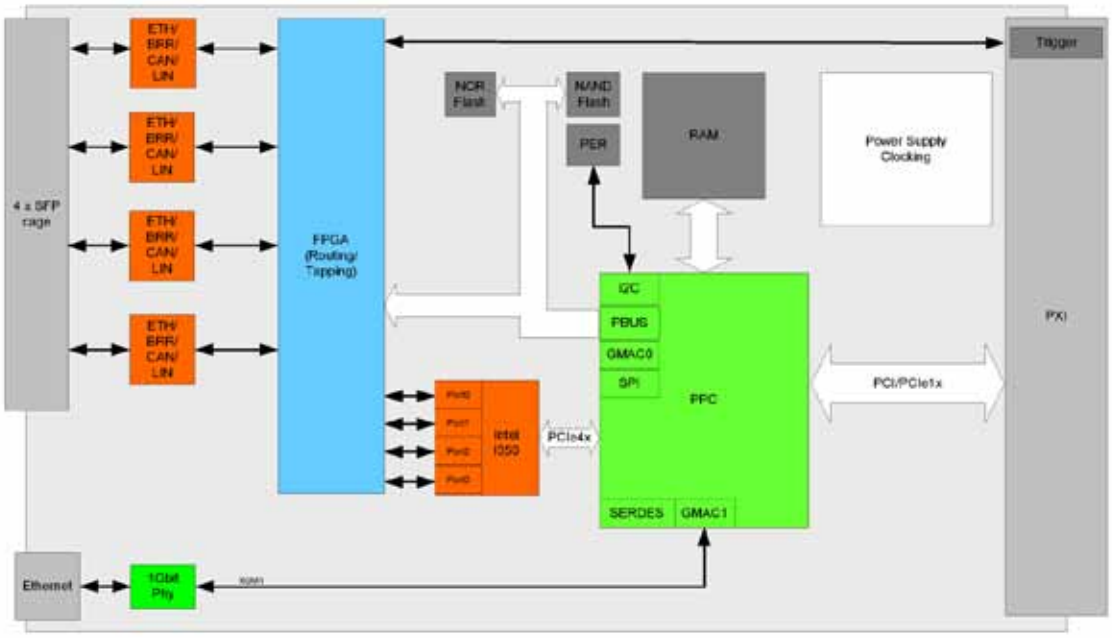


Figure 2-3: PXI 6141 _ Block diagram

2.3.2 Addressing

PXI 6141 boards provide a 1GBit Ethernet and a PXI interface.

Both interfaces can be used for the communication of the unit with the host PC.

In case of using the Ethernet interface, the board can be controlled via the 192.168.1.62, Port 5134 Default IP Address which can be changed if required
(see also Driver Installation/ Ethernet).

In principle, there are two ways for this:

- HardwareExplorer: Select the device,
  under Device set the required IP Address;
  the new IP Address is effective after restart
- G API Command G_Common_Ethernet_IpAddress_Set;
  the new IP Address is effective after restart

PXI racks do have an own geographical slot addressing of the backplane.
Numbering starts with 1 and can be seen at the cover's front side.
Mount always an embedded controller or an MXI card at slot 1.

A PXI 6141 board can read out this geographical slot address.

2.3.3 Isolation

Electric surges can harm expensive test equipment and lead to unreliable test results. Electric isolation protects against electric surges and can help to suppress dangerous electrical transients. It also eliminates ground loops, responsible for data errors due to ground potential differences.

The signal lines of the Ethernet SFP Modules are decoupled inductively, while the signal lines of the BroadR-Reach Module s are decoupled capacitively.

For the CAN-FD and LIN Modules, there is no isolation of the signal lines available.

2.3.4 Status LEDs

The LEDs arranged at the front panel indicate the current operation state of a PXI 6141 board

Figure 2-4: Status LEDs

The table below describes the meaning of the LED states:

LED 1	LED 2	LED 3	LED 4	Remarks
Permanently ON				Controller does not run (Error!)
Alternately blinking				Bootloader software runs
	blinking			Firmware runs
ON (shortly)				State during execution of a Firmware command
			ON	Ethernet connection established

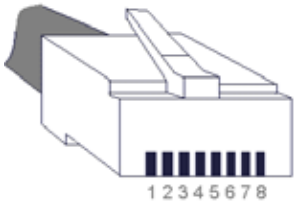
2.3.5 Ethernet

The Ethernet Module is using a Triple-Speed Transceiver in half duplex as well as full duplex operation.
For Data transmission, use an Ethernet cable of “CAT-6” category or higher.

Properties:

- Hot Pluggable Module
- Temperature range: 0°C .. 85°C
- Connection by RJ45 plug
- Data rate: 10/100/1000Mbit/s
- max. cable length up to 100m

Pin assignment of the RJ45 plug:



Ethernet Module:

Pin	Signal
Shield	Ground
1	D1+
2	D1-
3	D2+
4	D3+
5	D3-
6	D2-
7	D4+
8	D4-

2.3.6 BroadR-Reach

In the BroadR-Reach Module, the TJA1100 is used as PHY.
The TJA1100 is an OPEN Alliance BroadR-Reach compliant Ethernet PHY for Automotive Ethernet via an Unshielded Twisted Pair (UTP) cable.

Properties:

- Hot Pluggable Module
- Forward delay BroadR-Reach to 100Tx: 940ns
- Forward delay 100Tx to BroadR-Reach: 280ns
- Temperature range: -40°C .. 100°C
- Connection by RJ45 plug
- Data rate :100Mbit/s Full duplex
- Max. cable length up to 15m
- Unshielded Twisted Pair (UTP) cable as transmission medium
- Capacitive output coupling of the data lines

BroadR-Reach Module:

Pin	Signal
Shield	Ground
1	TRX+
2	TRX-
3	open
4	reserved
5	reserved
6	open
7	open
8	Ground

For simple connection, the D-Sub to RJ45 adaptor can be used:

D-Sub Adaptor cable for BroadR-Reach Pinout:

D-Sub (plug) RJ45

Pin	BroadR-Reach
1	open
2	open
3	Shield
4	TRx+
5	TRx-
6	open
7	open
8	open
9	open

Pin	BroadR-Reach
1	TRx+
2	TRx-
3	open
4	open
5	open
6	open
7	open
8	open

Define by the sliding switch on the rear side whether the Transceiver module is operated in the Master mode or in the Slave mode:

- M = Master Mode
- S = Slave Mode

2.3.7 CAN-FD

The CAN-FD Module is using the TJA1145 as a Transceiver.
The TJA1145 is a high-speed CAN Transceiver.

Properties:

- Hot Pluggable Module
- Temperature range: -40°C .. 100°C
- Connection by RJ45 plug
- Bus termination (120Ω) switchable
- Data rate up to 2Mbit/s
- Suited for 12V and 24V systems
- Operation voltage ($UBAT_{ext}$): 5V .. 30V
- Compatible to ISO 11898-2, ISO 11898-5 and ISO 11898-6

CAN-FD Module:

Pin	Signal
Shield	Ground
1	UBAT ext
2	open
3	CAN H
4	CAN L
5	Wake
6	open
7	open
8	Ground

2.3.8 LIN

The LIN Module is using the TJA1020 as a Transceiver.

Properties:

- Hot Pluggable Module
- Temperature range: -40°C .. 100°C
- Connection by RJ45 plug
- Master mode switchable (1KΩ to $UBAT_{ext}$)
- Baud rate up to 20kBaud
- Operation voltage ($UBAT_{ext}$): 5V .. 27V

LIN Module:

Pin	Signal
Shield	Ground
1	UBAT ext
2	open
3	open
4	open
5	LIN
6	open
7	Wake
8	Ground

2.3.9 FPGA Construction Ethernet/ BroadR-Reach

The following Block diagram shows an overview of the handling of the Ethernet and BroadR-Reach interfaces in the FPGA.

The FPGA has access to all four SFP Modules, all MACs of the PPC resp. I350 as well as to all internal Packet generators.

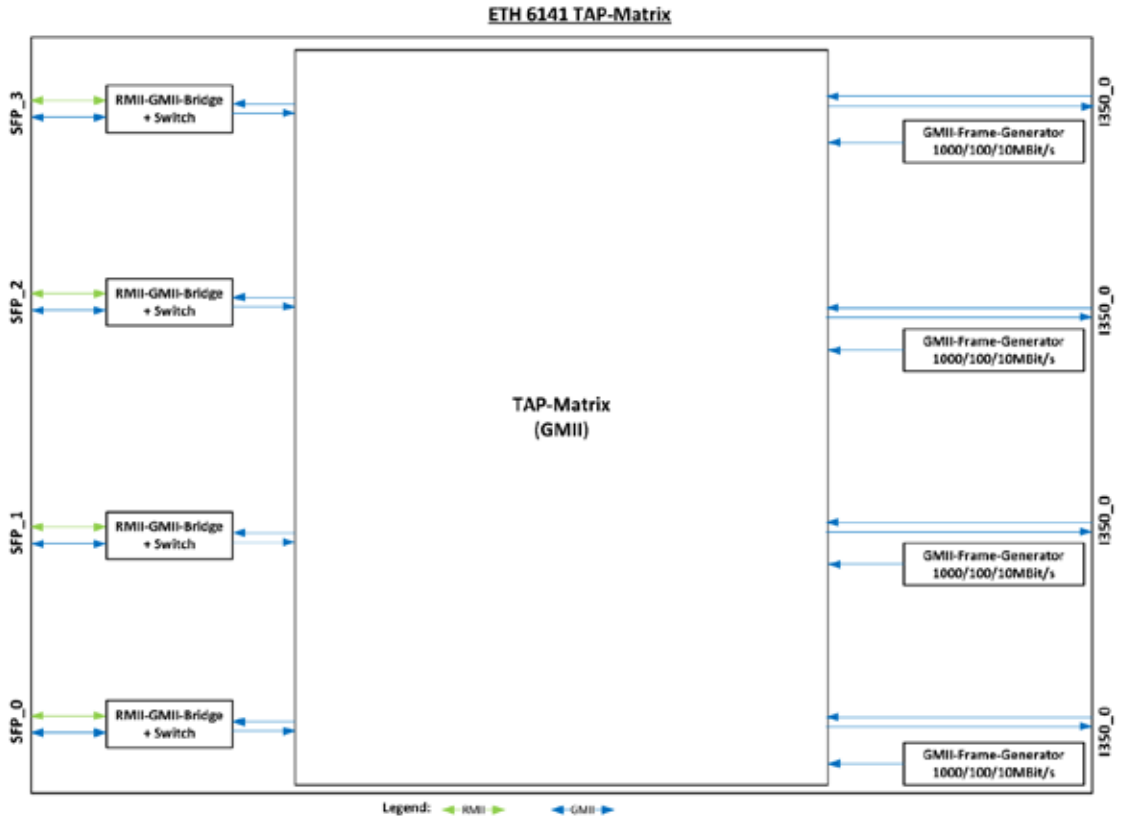


Figure 2-5: FPGA-Structure

2.3.10 Packet Generators

A PXI 6141 board has four freely configurable Packet Generators. Each of this generators has the following features:

- Settable Packet length: 1 .. 4096 bytes
- Settable pause between consecutive Packets: 0ns .. $(2^{64}-1)*8$ns
- Resolution: 8ns
- Output of a settable number of Packets: 1 .. $(2^{32}-1)$
  resp. also continuous output
- Settable Data rate: 10/100/1000 Mbit/s

2.3.11 TAP Matrix

For configuring the TAP Matrix for a joint transmission standard, the SGMII standard of the Ethernet Modules and the RMII standard of the BroadR-Reach Modules are converted to the GMII transmission standard in the FPGA. The TAP Matrix provides the possibility to connect each data source with each data sink.

The following multiplexer is preset to each data sink:

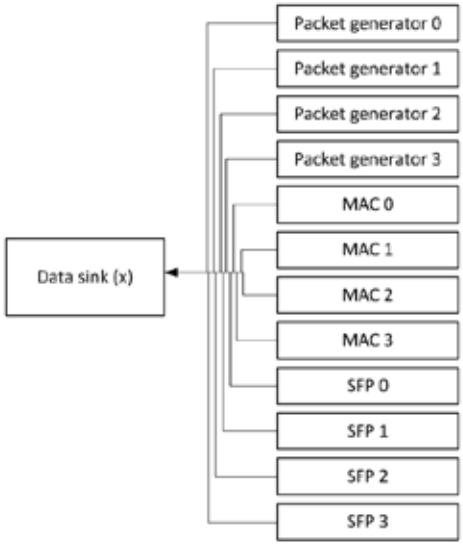


Figure 2-6: Multiplexer

”Data sink (x)“ in this image represents the pluggable Ethernet Modules 0 .. 3, BroadR-Reach Modules 0 .. 3 and the internal MACs 0 .. 3.

For example this TAP Matrix provides the possibility to drag a BroadR-Reach connection through the FPGA with SFP0 and SFP1, but without impacting the communication of both UUTs, and at the same time recording RX data via the internal MACs 0 and 1.

3 Software

To embed PXI 6141 boards in your own applications, there are the following options:

- Programming via G-API
- UserCode Programming
- Programming via DLL Functions (Windows Device Driver)
- Programming with LabVIEW

3.1 G-API Programming

The G-API (GOEPEL-API) is the C-based user interface for GOEPEL electronic hardware under Windows®.
It provides a wide, hardware independent command set for CAN, CAN-FD, LIN/ KLine, MOST, FlexRay, LVDS, SENT, ADIO and Diagnostic services. No matter whether a PXI/ PCI, USB or Ethernet device is used, the commands remain the same.

The hardware abstraction introduced with the G-API gives the test application parallel access to the hardware, allowing one application to access multiple hardware interfaces. As well as multiple applications can access the same hardware interface in parallel.

Another feature introduced by the G-API is the asynchronous hardware access. This means no execution blocking for pending firmware commands. The command acknowledgement is provided via a callback mechanism.

With the HardwareExplorer (see also Ethernet), GOEPEL electronic provides an efficient hardware configuration and management tool, offering users an easy way to manage their hardware configurations and identifying specific hardware interfaces by logical names. Using logical interface names in the application saves from rebuilding the application when porting it to another interface or controller board, as the interface can be easily reassigned in the HardwareExplorer.
Furthermore, the HardwareExplorer provides a simple means of testing the interaction between hardware and software by executing the integrated self-tests.

Please consult the G-API documentation for further information.
This documentation and the installation software are located in the *G-API* folder of the supplied “Product Information” CD.

3.2 UserCode Programming

PXI 6141 controller boards can execute user programs direct on their PowerPC processor. This requires the UserCode run-time module being enabled.

The UserCode run-time module is an option for PXI 6141 boards (plus other GOEPEL devices) and requires one license per unit.

Executing programs directly on the PowerPC improves the real-time performance remarkable and frees up PCI bandwidth on the host system.

Therefore GOEPEL electronic has ported and enhanced by additional on-board functionality their C-programming user interface called G-API from Windows® to the QNX Neutrino real-time operating system.
The QNX Neutrino real-time operating system is based on a micro kernel architecture, providing clear separation between the kernel and each individual application.
This allows user applications to run in a separate virtual memory space, which ensures safe test execution and improves reliability.

The UserCode run-time module uses a superset of the G-API commands for Windows® ensuring an easy migration of existing program source code. Additional functions will provide access to event notifications, timer tasks, the FLASH file system and other RT OS resources as well as standard C libraries

The PowerPC processor uses big-endian byte order which must be taken care of when writing or porting code for the UserCode Run-time module. For smooth migration from little to big-endian, a library of conversion macros is provided with the Net2Run IDE development system.

With the Net2Run IDE development system, GOEPEL electronic provides a complete tool chain for creating UserCode programs and for their direct execution on PXI 6141 boards.

The Net2Run IDE development system is based on Eclipse IDE and contains the QNX Neutrino Command Line Tools (CLT), including PowerPC-Compiler, Linker and Debugger.

UserCode programs can be downloaded and debugged direct from Net2Run IDE via an Ethernet connection.

The figure below shows the Net2RunIDE development system:

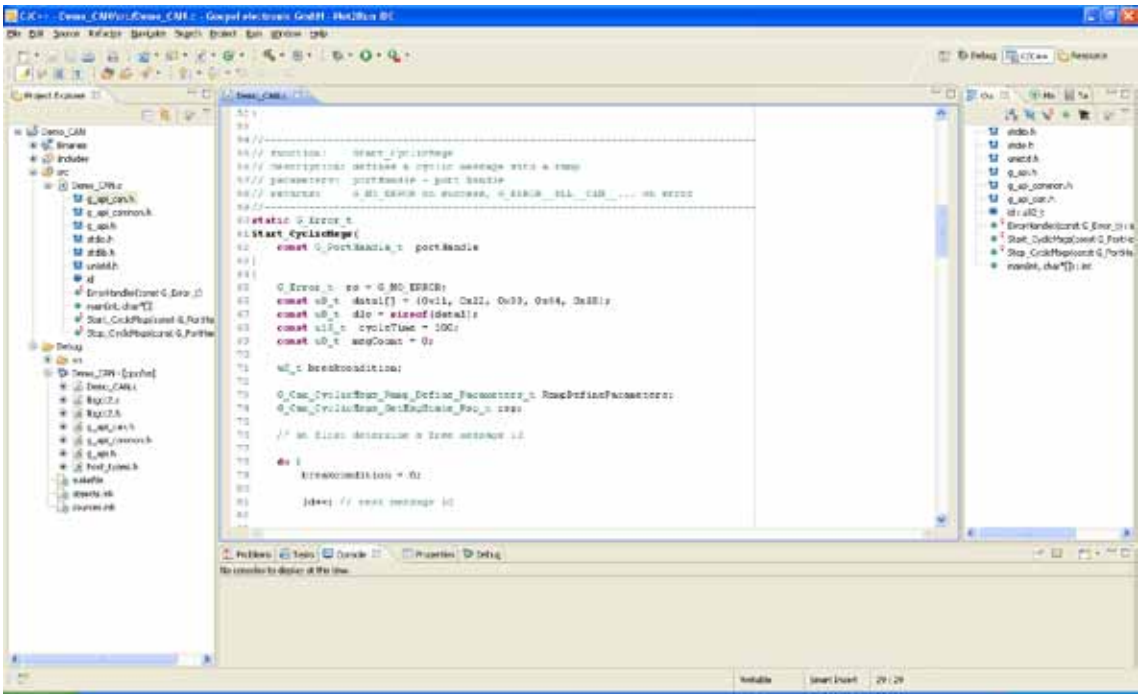


Figure 3-1: Net2Run IDE Window

Please consult the G-API documentation for further information. This documentation and the installation software are located in the *G-API* folder of the supplied “Product Information” CD.

3.3 Programming via DLL Functions

For the used structures, data types and error codes refer to the headers – you find the corresponding files on the supplied CD.

3.3.1 Windows Device Driver

The DLL functions for programming using the Windows® device driver are described in the following chapters:

- System Info
- Transceiver Info
- Write Instruction
- Read Response
- Read Response Block

3.3.1.1 System Info

The Pxi61xx__SystemInfo function is used for the status query of the hardware driver and for query of the board properties.

Format:

S32 Pxi61xx__SystemInfo(t_System_Info *pSystemInfo, U32 LengthInByte);

Parameters:

Pointer, for example pSystemInfo,
to a data structure
(For the structure, see the *Pxi61xx.h* file on the supplied CD)

LengthInByte
Size of the buffer pSystemInfo is pointing to, in bytes

Description:

The Pxi61xx__SystemInfo function returns information regarding the status of the hardware driver.

For this, the address of a pSystemInfo pointer has to be transferred to the function.
Within the function, the structure pSystemInfo is pointing to is filled with the corresponding pieces of information.

3.3.1.2 Transceiver Info

The Pxi61xx__TransceiverInfo returns information regarding the plugged-in transceivers as well as their number.

Format:

S32 Pxi61xx__TransceiverInfo(t_Transceiver_Properties *pTransceiverProperties, U32 LengthInByte);

Parameters:

Pointer, for example pTransceiverProperties,
to a data structure
(For the structure, see the *Pxi61xx.h* file on the supplied CD)

LengthInByte
Size of the buffer pTransceiverProperties is pointing to, in bytes

Description:

The Pxi61xx__TransceiverInfo function returns information regarding the transceiver properties.
For this, the address of a pTransceiverProperties pointer has to be transferred to the function.
Within the function, the structure pTransceiverProperties is pointing to is filled with the corresponding pieces of information.

3.3.1.3 Write Instruction

The Pxi61xx__WriteInstruction function is for writing a command to the PXI 61xx Controller.

Format:

S32 Pxi61xx__WriteInstruction(U8 *pData, U16 DataLength);

Parameters:

Pointer, for example pData,
to the writing data area,

Consisting of Command header and Command bytes
(At present 4096 bytes per command)

DataLength

Size of the writing area pData is pointing to, in bytes

Description:

The Pxi61xx__WriteInstruction function sends a command to the PXI 61xx controller.

In the header of the structure pData is pointing to, there is the information regarding the PXI 61xx board to be activated by this function.
Therefore this parameter is not to be given separately.

3.3.1.4 Read Response

The Pxi61xx__ReadResponse function is for reading a response from the PXI 61xx controller.

Format:

S32 Pxi61xx__ReadResponse(U8 Device, U8 *pData, U32 *DataLength);

Parameters:

Device

Index of the PXI 61xx board, beginning left with 1

Pointer, for example pData,
to the reading data area,

Consisting of Response header and Response bytes
(At present 4096 bytes per response

DataLength

Parameter value before function call:
Size of the buffer pData is pointing to, in bytes

Parameter value after function call:
Number of bytes actually read

Description:

The Pxi61xx__ReadResponse function reads back the oldest response written by the PXI 61xx controller in the response area.

If several responses have been provided by the controller, but not read, they are not lost but stored in the form of a list.

On calling up, the Pxi61xx__ReadResponse function continues to supply data until this list contains no more entries.

3.3.1.5 Read Response Block

The Pxi61xx__ReadResponseBlock function is for reading all available responses from the PXI 61xx controller.

Format:

S32 Pxi61xx__ReadResponseBlock(U8 Device, U8 *pData, U32 *DataLength, U32 *BlockCounter);

Parameters:

Device

Index of the PXI 61xx board, beginning left with 1

Pointer, for example pData,
to the reading data area,

Consisting of Response header and Response bytes
(At present, max. 4096 bytes per response)

DataLength

Parameter value before function call:
Size of the buffer pData is pointing to, in bytes

Parameter value after function call:
Number of bytes actually read

Pointer, for example BlockCounter

Number of the individual responses contained

Description:

The Pxi61xx__ReadResponseBlock function reads back all responses written by the PXI 61xx controller in the response area.

3.4 Programming with LabVIEW

3.4.1 LabVIEW via the G-API

The supplied CD contains VIs for activating PXI 6141 boards under LabVIEW.

These LabVIEW VIs use the functions of the GOEPEL G-API.

3.4.2 LLB using the Windows Device Driver

The supplied CD contains VIs for activating PXI 6141 boards under LabVIEW.

The functions described in the Windows Device Driver section are used for this.

3.5 Additional Software Interfaces

3.5.1 FS

The Software Interface “FS1” (File System) allows, amongst others, creating, copying, deleting, executing and searching of files on the hardware.
Thus, it allows uniform access to the OnBoard File System.

3.5.2 Net2Run

The Software Interface “Net2Run” (Net2Run1 .. Net2Run4) serves for the creation, configuration and execution of Residual bus simulations. Several bus interfaces for CAN, LIN and FlexRay networks can be simulated simultaneously and continuously.
The “Net2Run” interface supports loading and executing of so-called Residual bus simulation files (*.rbs). These are preconfigured command sequences containing a static Residual bus simulation.
The corresponding files are created by means of the “Net2Run” Configurator Tool.

“Net2Run” is subdivided into several Software modules, strongly leaning to “AUTOSAR”.

The following Software modules do exist:

- COM
- PDU-Router
- CAN-Interface
- LIN-Interface
- FlexRay-Interface
- PDU-Multiplexer
- CAN-NM
- FlexRay-NM

Hence, the routing of PDUs of e.g. CAN1 to CAN2, CAN1 to LIN3 or FlexRay2 to CAN4 is possible (PDU-Gateway). The routing of individual signals can be realized by a COM-Signal-Gateway. In order that several independent Residual bus simulations can be executed on one card (e.g. one Residual bus simulation on CAN1, CAN2, CAN3 and CAN4 each), several “Net2Run” Interfaces do exist (4).

3.5.3 Sequence

The Software Interface “Sequence1” allows recording and playing of Firmware commands as a command sequence, short “Sequence”. A Sequence can also permanently stored under an arbitrary name on the board.
By using its name, this Sequence can be loaded again and played.
The automatic loading of a Sequence after switching on the board e.g. allows the automatic configuration and starting of a Residual bus simulation (in the case the required commands are included in the Sequence).

3.5.4 UserCode

The Software Interface “UserCode1” allows the OnBoard execution of user programs (see also UserCode Programming).
For the communication between OnBoard programs and the Host, Message-FIFOs do exist.
Each side (OnBoard program or Host) can create, write to or read from a Message-FIFO.
Each FIFO can be read and written from both sides. For consistency it is recommended to have a separate FIFO for each direction. So that one side only writes to and the other side only reads from a FIFO.

3.6 Further GOEPEL Software

PROGRESS, Program Generator and myCAR of GOEPEL electronic GmbH are comfortable software programs for testing with GOEPEL hardware.

Please refer to the corresponding User Manual to get more information regarding these programs.

R

S

T

U

W