Developer Manual openPOWERLINK Development Kit

Developer Documentation
Version 2

Edition February 2009

Document Number: L-1104e_2

system house for distributed automation

Status/Changes

Status: approved

Date/Version	Chapter	Changes	Editor
12.01.07 / 1.0	all	Draft	D. Krüger
28.02.08 / 1.1	Toolchain	Adjustments to current Linux-BSP	D. Krüger
28.02.08 / 2	all	Rework for new Development Kit	D. Krüger

	EUROPE	NORTH AMERICA
Address:	SYS TEC electronic GmbH August-Bebel-Str. 29 D-07973 Greiz GERMANY	PHYTEC America LLC 203 Parfitt Way SW, Suite G100 Bainbridge Island, WA 98110 USA
Ordering Information:	+49 (3661) 6279-0 info@systec-electronic.com	1 (800) 278-9913 info@phytec.com
Technical Support:	+49 (3661) 6279-0 support@systec-electronic.com	1 (800) 278-9913 support@phytec.com
Fax:	+49 (3661) 62 79 99	1 (206) 780-9135
Web Site:	http://www.systec-electronic.com	http://www.phytec.com

2nd Edition February 2009

Table of Contents

Index of Figures

Index of Tables

1 Document Overview

The document on hand describes the commissioning of the openPOWERLINK Development Kit on the basis of the Development Kit ECUcore-5484 as well as the existent openPOWERLINK demo application.

Initially, **section 4** of this document deals with the **commissioning and configuration** of the openPOWERLINK Development Kit on the basis of the Development Kit ECUcore-5484.

In **section 5** the usage of the ECUcore-5484 is described, including the **login to the system** as well as the **configuration of the file system**.

The **Ethernet POWERLINK functionality** of the ECUcore-5484 is defined in c**hapter 6**.

In **section 7** the diagnostics tool Wireshark is explained.

Section 8 offers a first introduction to the demo application of the ECUcore-5484.

For more information about the openPOWERLINK Protocol Stack please refer to the manuals L-1108 “Introduction into openPOWERLINK Protocol Stack” and L-1098 “Software Manual openPOWERLINK Protocol Stack”.

2 Scope of Delivery

Figure 1: Scope of Delivery

The openPOWERLINK Development Kit consist of the following parts.

Quantum	Part
1	Development Kit ECUcore-5484 as Controlled Node
1	openPOWERLINK Live-CD for X86 processors with Realtek RTL8139 network card as Managing Node.
2	Network cable CAT 5e
1	RS-232 cable for configuration and debugging of ECUcore-5484
1	Power supply 12 VDC, 1.5 A
1	Documentation

Table 1: Scope of Delivery

3 Basic Setup

3.1 Requirements

Using this Development Kit requires a personal computer with

- X86 compatible processor with 1 GHz or higher
- at least 256 MB RAM
- optionally at least 5 GB free space on hard disk for VMware appliance with development environment for ECUcore-5484
- DVD-ROM drive and
- Realtek RTL8139 network controller, e.g.
    - Zyxel FN312
    - Netgear FA311 v2 Rev-D1
    - D-Link DFE-528TX

3.2 Step by Step Instructions

These steps lead you to your own POWERLINK network within a few minutes.

1. Connect the network cable to the RJ45 jack named “EPL1” of the Developmentboard ECUcore-5484 and the RJ45 jack of the Realtek RTL8139 network controller.
2. Ensure that the HEX switches for the node-ID on the Developmentboard ECUcore-5484 are set correctly (S307 = ‘0’, S306 = ‘1’);
3. Connect power supply (24 VDC, 1 A) to the Developmentboard ECUcore-5484 (connector X700).
4. Switch on and boot the personal computer from the openPOWERLINK Live-CD and wait until the welcome screen appears.
5. Start the POWERLINK demo application on the personal computer (e.g. via the icon on the destop).
6. Click on the button “Start EPL Stack”

7. When the POWERLINK network has entered NMT state OPERATIONAL, you can enjoy a running light on your personal computer and the LEDs on the Developmentboard ECUcore-5484.
8. Press the buttons on the Developmentboard ECUcore-5484 to change the direction and mode of the running light.

4 Commissioning of the Development Kit ECUcore-5484

4.1 Electrical Connections

The wall power supply and the Ethernet- and RS232 cables which are necessary for operating the Development Kit ECUcore-5484 are included in the delivery. All connections listed in the table below are the minimum connections to start the Kit.

Connection	Marking on the Demoboard	Remark
Supply voltage	X700 or X701	The wall power supply included in the delivery is intended for direct connection to X700
Ethernet (ETH0)	X500	Interface which facilitates communication to the development system
Ethernet (EPL1)	X501	Interface to the POWERLINK network (is connected to EPL2 via a Hub)
Ethernet (EPL2)	X502	Interface to the POWERLINK network (is connected to EPL1 via a Hub)
RS232 (COM0)	X400 / top	Interface which only allows configuration of the subassembly (e.g. setting the IP-Address) and is not necessary during normal operation
CAN (CAN0)	X400 / bottom	CAN-cable is an optional accessory and is not included in the Kit delivery contents

Table 2: Connections of the Development Kit ECUcore-5484

Figure 2 shows the position of all connections on the baseboard of the Development Kit ECUcore-5484. The power supply for the Kit may optionally as well run over X701 with an external source of 9VDC to 24VDC with 12W at minimum.

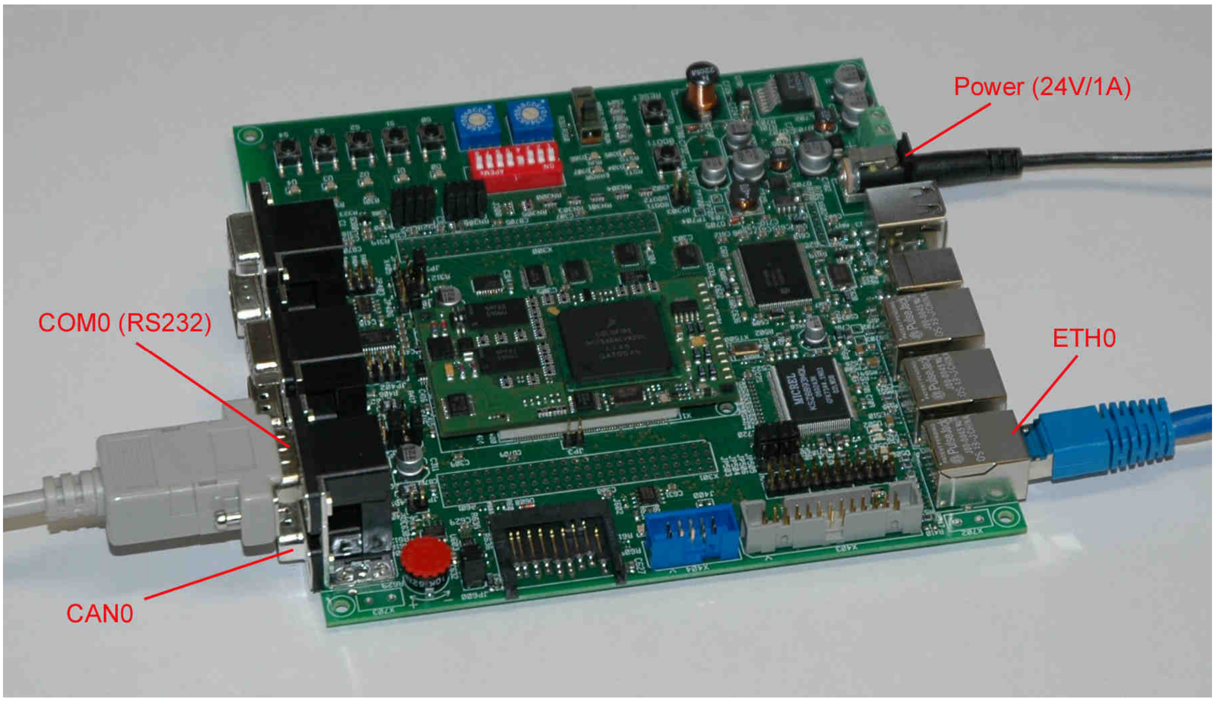


Figure 2: Position of the connections on the baseboard of the Development Kit ECUcore-5484

Advice: For correct commissioning, the Ethernet and RS232 cables have to be connected prior to activating the supply voltage.

4.2 Jumper Configuration

All jumpers on the baseboard must be configurated as follows:

Jumper	Setting	Remark
JP505	2-3	Important for Ethernet POWERLINK
JP506	2-3	Important for Ethernet POWERLINK

Table 3: Jumper configuration

4.3 Configuration of the ECUcore-5484

The configuration of the ECUcore-5484 takes place via a serial interface at a computer. Therefore, any kind of terminal program must be started on the computer (e.g. HyperTerminal, TeraTerm, …) and configuration steps must be taken as follows (also compare picture 2):

- 19200 Baud
- 8 Data-bit
- 1 Stop-bit
- no parity
- no flow control

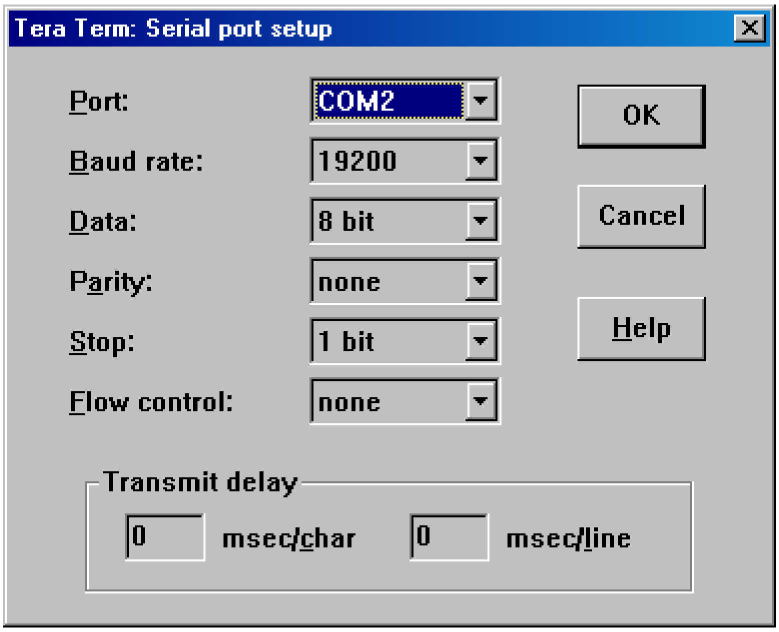


Figure 3: Terminal configuration

For configurating the ECUcore-5484, the DIP-switch 2 on the ECUcore-5484 must first be set to position “Off” (see Figure 4).

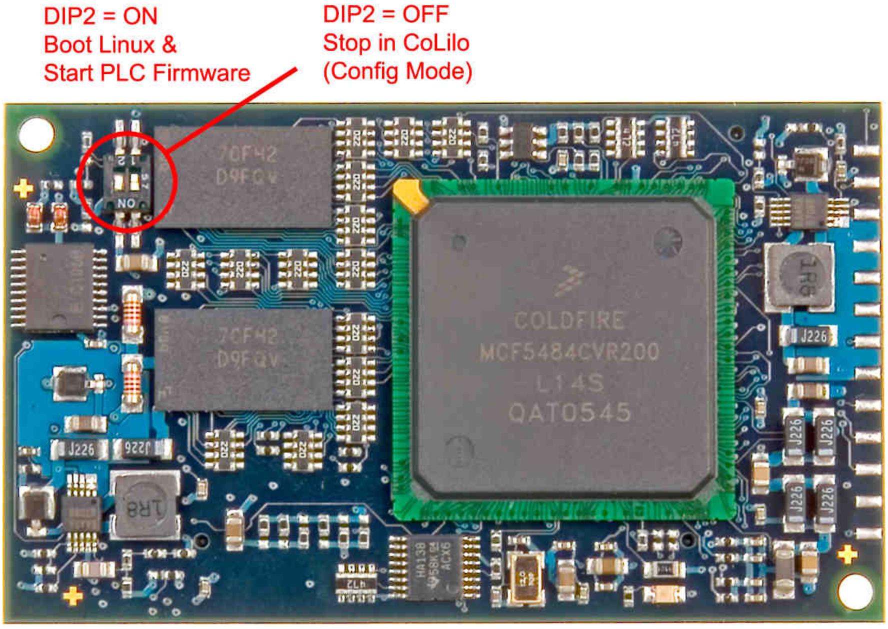


Figure 4: Position of the DIP-switch 2 on the ECUcore-5484

After operating the Reset-pushbutton (pushbutton S303), the Linux-Bootloader “CoLilo” prints its prompt “colilo>”.

By entering “?” at the “CoLilo” command prompt, assistance about all available commands is offered and the current module configuration is shown (see Figure 5).

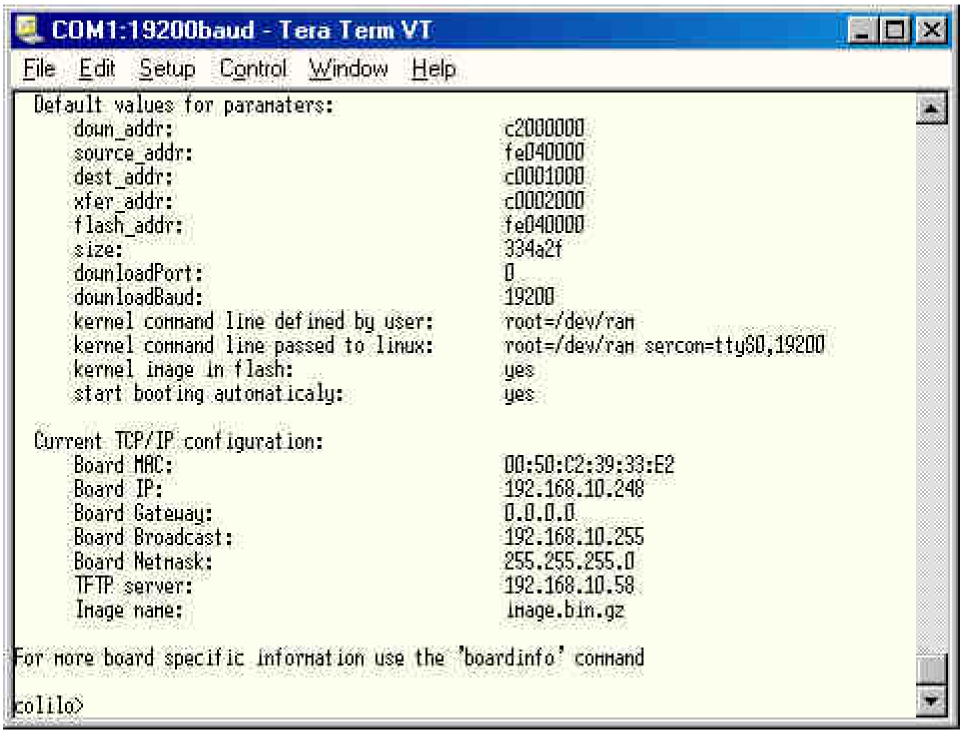


Figure 5: Display of the module configuration in “CoLilo”

The configuration of the ECUcore-5484 takes place at the “CoLilo” command prompt, using the commands listed in Table 4.

Configuration	command	Remark
MAC-address of the ECUcore-5484	set mac <xx:xx:xx:xx:xx:xx>	The MAC-address is a well-defined module worldwide identification which is assigned by the producer. It should not be modified by the user.
IP-address of the ECUcore-5484	set ip <xxx.xxx.xxx.xxx>	This command sets the local IP-address of the ECUcore-5484. The IP-address is determined by the network administrator.
Net mask	set netmask <xxx.xxx.xxx.xxx>	This command configures the net mask of the ECUcore-5484. The net mask is set by the network administrator.

Gateway-address	set gw <xxx.xxx.xxx.xxx>	This command defines the IP-address of the gateway to be used by the ECUcore-5484. This Gateway-address must be determined by the network administrator. Advice: If ECUcore-5484 and development system are located in the exact same sub net, the definition of the Gateway-address can be left out and "0.0.0.0" can be used instead.

Table 4: "CoLilo" commands for configurating the ECUcore-5484

Modified configurations can again be verified at the "CoLilo" command prompt by entering "?". Current configurations are permanently saved in the flash of the ECUcore-5484 by entering the command

```
config save
```

The modifications are adopted after resetting the ECUcore-5484.

To restart the Linux of the ECUcore-5484 after finishing the configurations, the DIP-switch 2 on the ECUcore-5484 is to be put back into position "On" (compare Figure 4).

After pressing the reset-button (reset S303), Linux starts with the current configuration.

5 How to use the ECUcore-5484

5.1 Logging on to the System

The ECUcore-5484 is delivered with preinstalled Embedded Linux. After booting up the system, the user may log in as “PlcAdmin” with password “Plc123”. Accessing the system is possible via the serial interface (COM0) on the baseboard of the Development Kit ECUcore-5484 which was used for the configuration already. It is essential to use the settings of the serial connection which were described in section 4.3 (19200 Baud, 8 Data-bits, 1 Stop-bit, no parity, no flow control).

As alternative solution to the serial interface, the access to the ECUcore-5484 is as well possible over the Ethernet interface ETH0 (see Figure 2) using a Telnet-Client. The standard Windows Client may be used to log on to the ECUcore-5484. Therefore, the command “telnet” must be called at the IP-address for the ECUcore-5484 determined in chapter 2.3, e.g.

```
telnet 192.168.10.244
```

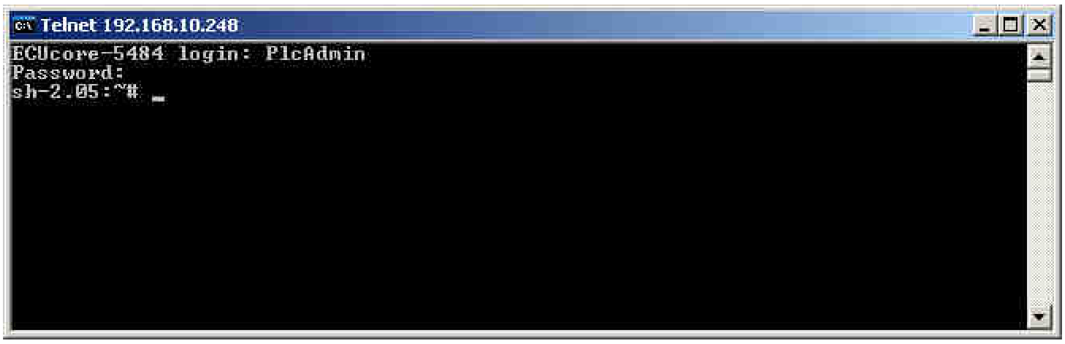


Figure 6: Access to the ECUcore-5484 via Telnet

Figure 6 illustrates the access to the ECUcore-5484 via the Telnet-Client that Windows contains as a standard feature.

5.2 File system of the ECUcore-5484

The preinstalled Embedded Linux on the ECUcore-5484 provides parts of the system memory in form of a file system. As it is common for embedded systems, most of this file system is displayed as “read/only”. This implies that modifications can only be made through a rebuild of the Linux-Image for the ECUcore-5484 on the development computer. The advantage hereby is the resistance of the read/only file system against damages in connection with power supply breakdowns that occur very often in embedded systems. Embedded systems are usually switched off without completing a shutdown command afore.

Paths of the file system which are usable for the user during runtime (without rebuild of the Linux-Image for the ECUcore-5484) are listed in Table 5. If parts of the development computer are not already integrated via NFS (compare section 8.3.1), one of both RAM-Disk directories “/var” or “/tmp” should be generally used for tests during the development phase.

Path	Usable size	Description
/var	1000 kByte	RAM-Disk, well suited for the testing of programs, but no persistence in case of power supply breakdown
/mnt/mtd3	1000 - 5632 kByte depending on Linux configuration	Flash-Disk, to persistently store user programs, configuration files, etc.; data persistence in case of power supply breakdown is given
/mnt/nfs		Target for integrating remote directories of other systems via NFS, also compare section 8.3.1.
/tmp	2000 kByte	RAM-Disk, as "/var" it is well suited for the testing of programs, but no data persistence in case of power supply breakdown

Table 5: File system configuration of the ECUcore-5484

Advice: Sizes of the file system paths “/var” and “/tmp” may be modified in the start script “/etc/rc.d/rcS”.

However, this calls for a rebuild of the Linux-Image for the ECUcore-5484.

Using the command “df” (“DiskFree”), it is possible to determine currently configured sizes and available sizes of file system paths. Figure 7 shows an example for the usage of this command.

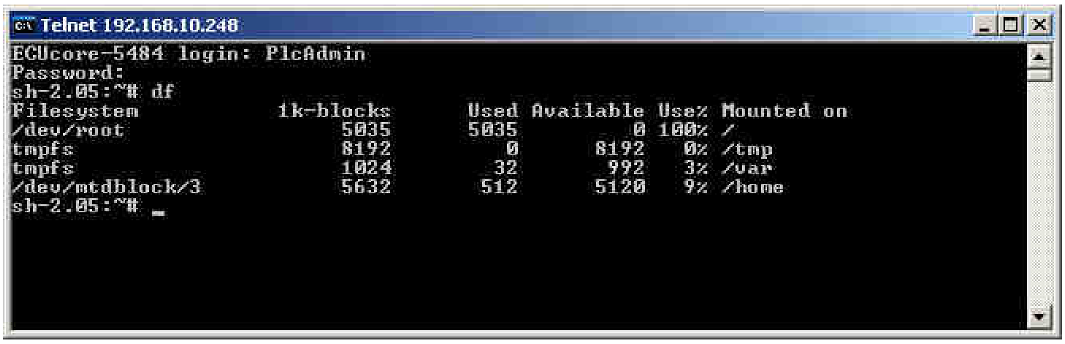


Figure 7: Example for the usage of command “df”

6 Functionality of Ethernet POWERLINK

The openPOWERLINK Development Kit upon delivery is configured, so that the openPOWERLINK demo application starts automatically when the system is booted. The following sections describe how the demo application can be configured and used.

6.1 Configurating the Node-ID

The node-ID, i.e. the POWERLINK address, of the ECUcore-5484 within an Ethernet POWERLINK network is configured via the HEX switches S306 and S307. S306 hereby corresponds to the low-order 4 Bit and S307 to the high-order 4 Bit of the node-ID. The address modification only takes effect after resetting the ECUcore-5484.

The node-ID 0xF0 = 240 configures the ECUcore-5484 to run as Managing Node. Otherwise it runs as Controlled Node.

6.2 Connection to the Ethernet POWERLINK Network

The connection to an Ethernet POWERLINK network is possible via the RJ45 jack X501 and X502 (EPL1 and EPL2). Both jacks are connected to the Ethernet controller of the ECUcore-5484 via an Ethernet hub integrated on the baseboard. For diasgnostic purposes, it is possible to connect a computer to the integrated Ethernet hub and, for example, to monitor and store traffic of the POWERLINK network via Wireshark.

6.3 Digital In- and Outputs

The pushbuttons S0 to S4 on the baseboard are linked as digital inputs in the object directory in object 0x60000 sub-index 0x01 of the type UNSIGNED8.

The LED D0 to D4 on the baseboard are linked as digital outputs in the object directory in object 0x6200 sub-index 0x01 of the type UNSIGNED8.

6.4 Process Data Objects (PDO)

There is a receive PDO (PollRequest of MN) and a transmit PDO (PollResponse), each of the length 1 Byte. Object 0x6200 sub-index 0x01 is mapped in the reception PDO and object 0x6000 sub-index 0x01 in the transmit PDO.

7 Diagnostics Tools

7.1 Wireshark

Wireshark (former Ethereal) supports the analysis of network connections. Since its version 0.99.4, Wireshark includes a dissector for POWERLINK-frames. With it, POWERLINK-frames are displayed easy to understand and not only as hexdump.

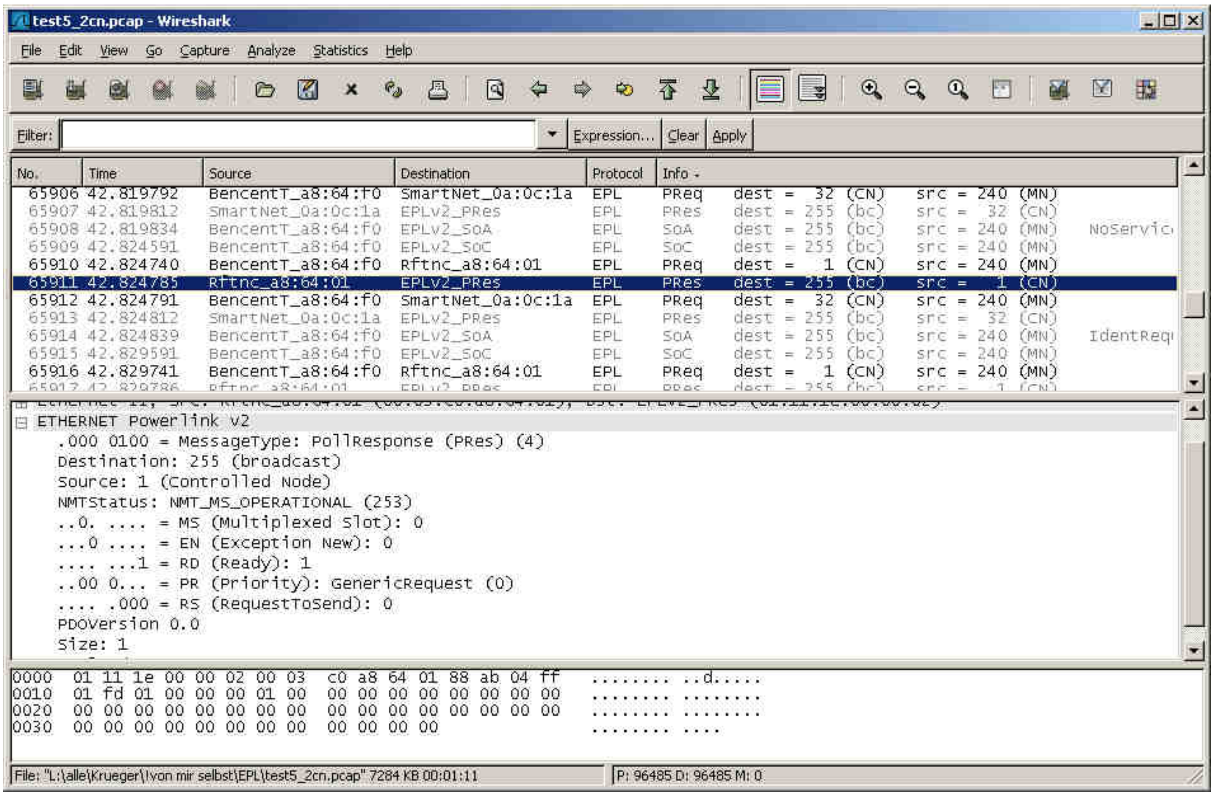


Figure 8: Analysis of a PRes-frame with Wireshark

7.1.1 Download and Installation

The current version of Wireshark can be downloaded at http://www.wireshark.org/. During the installation, WinPcap is installed on Microsoft Windows as well. This driver is necessary to put the network interface card of the computer into the so called Promiscuous Mode. This mode enables the network interface card to pass through every frame received to the operating system or the application – and not only the frames that are addressed to the network interface card.

7.1.2 Premises

For diagnostics using Wireshark, the computer must be connected to the POWERLINK network. Therefore, it is necessary to use an additional network interface card. Only the network monitor driver and perhaps the internet protocol (TCP/IP) are allowed to be bound to this additional network interface card. However, a permanent IP-address should be set for the internet protocol (TCP/IP), e.g. 192.168.100.245.

7.1.3 Recording the Network Traffic

Recording the network traffic is relatively simple. Initially, within the menu item “Capture” -> “Options…” the appropriate network interface card must be chosen. Additionally, among “Display Options” the check boxes “Update list of packets in real time”, “Automatic scrolling in live capture” and “Hide capture info dialog” may be selected.
Now, the recording is ready to be started by pressing the button “Start”. By choosing the menu item “Stop” within the menu “Capture”, the recording can be stopped.

8 POWERLINK Demo Application

8.1 Setting up the Toolchain

For development, Linux BSP for Freescale MCF548x processors is necessary – including the corresponding Toolchain. This is to be unzipped in any directory (e.g. /home/user/projects/LinuxBSP-2.6). Thereby, the Toolchain is located beneath Linux BSP in the directory toolchain (e.g. /home/user/projects/LinuxBSP-2.6/toolchain).

The following environment variables must be set within the shell used. Therefore, the following lines can be inserted into, for example, the file "~/.bashrc". The variable M68K_LINUX_BSP_PATH must be adjusted appropriately.

```
M68K_LINUX_BSP_PATH=/home/user/projects/LinuxBSP-2.6
M68K_LINUX_KDIR_PATH=$M68K_LINUX_BSP_PATH/linux-2.6.10
M68K_CC_PREFIX=$M68K_LINUX_BSP_PATH/toolchain/bin/m68k-
   linux-gnu-
M68K_CFLAGS=-malign-int
M68K_LD_LIBRARY_PATH=

export M68K_LINUX_BSP_PATH
export M68K_LINUX_KDIR_PATH
export M68K_CC_PREFIX
export M68K_CFLAGS
export M68K_LD_LIBRARY_PATH
```

The following steps are necessary for the first initialization:

```
cd $M68K_LINUX_BSP_PATH
make fresh
make menuconfig
make
```

Since the standard configuration can be adopted, all three appearing dialogs are to be left via "Exit" and each storing of the configuration is to be confirmed with "Yes". Afterwards, the Toolchain is ready for operation.

8.2 Building the POWERLINK Demo Application

The openPOWERLINK Stack contains demo projects for the ECUcore-5484 within the directory Examples/PLCcore-CF54/Linux/gnu.

At first, you may need to extract the ZIP archive with the source code of the openPOWERLINK Stack. The source code can be found on the openPOWERLINK Live-CD or downloaded from http://openpowerlink.sourceforge.net/.

```
user@host: ~> unzip openPOWERLINK_v1.3.0.zip
user@host: ~> cd openPOWERLINK_v1.3.0
```

8.2.1 Project demo_mn_kernel

This project demonstrates how the openPOWERLINK Stack can be utilized as Linux-kernel module. It is located in the directory Examples/PLCcore-CF54/Linux/gnu/demo_mn_kernel of the openPOWERLINK source code. The demo application "demo_main.c" is placed within the same Linux-kernel module as the openPOWERLINK Stack. This implies the advantage of a direct and efficient connection between the application and the openPOWERLINK Stack.

This demo application provides the functionality described in section 6. It contains an object directory, PDO support, SDO server and client, support of SDO via ASnd and UDP, a virtual Ethernet driver for Linux and the Data Link Layer, and the NMT module for Controlled Nodes and Managing Nodes.

8.2.1.1 Translating the Source Codes

On the development system, the command "make" is to be completed within the above mentioned directory .

```
user@host:Examples/PLCcore-CF54/Linux/gnu/
   demo_mn_kernel> make
```

Therewith, the Linux-kernel module epl.ko is built. As described in section 8.3, this module can be copied onto the ECUcore-5484.

8.2.1.2 Executing the Application

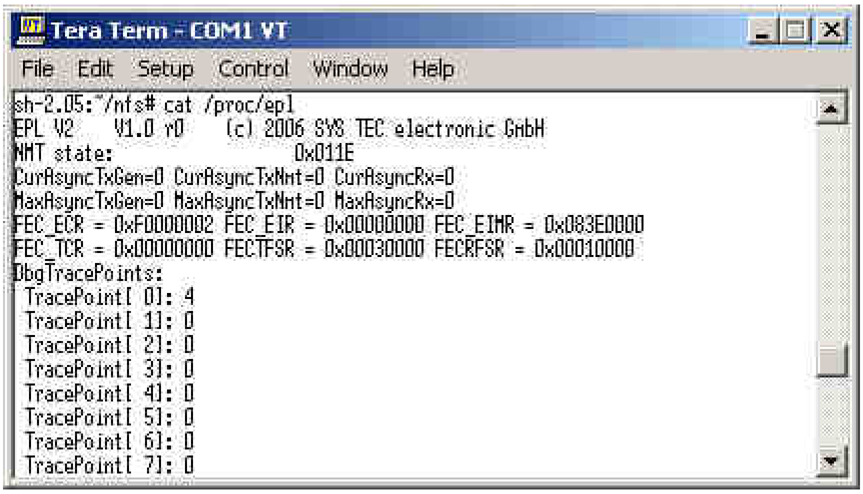
The Linux-kernel module epl.ko requires the Board-support driver cf54drv.ko which is automatically loaded on the delivered ECUcore-5484.

Loading the kernel module takes place via the Linux command “insmod”. Consequently, the openPOWERLINK Stack is initialised, the network interface is configured and the NMT state machine is started.

```
sh-2.05:/nfs# insmod epl.ko
```

After loading the kernel module, the demo application is running. It installs a device “epl” in the proc file system over which status information about the openPOWERLINK Stack may be inquired.

```
sh-2.05:/nfs# cat /proc/epl
```

Picture 9: Part of the output of “cat/proc/epl”

During the unload of the kernel module via “rmmod”, the EPL Stack is stopped. This command is as well necessary to eventually remove an EPL Stack that is already loaded.

```
sh-2.05:/nfs# rmmod epl.ko
```

8.3 Updating the Demo Application on the ECUcore-5484

Provided that the ECUcore-5484 is configured as described in section 5, the EPL demo application on the module can be updated. After rebooting the module (DIP switch 2 positioned at “on”, compare Figure 4), the login as **user “PlcAdmin” with password “Plc123”** is possible.

Transmitting the demo application on the ECUcore-5484 can take place using two different possiblities that each include advantages and disadvantages:

NFS: The “Network File System” (NFS) represents the simplest way of starting the demo application – that has been built on the development system – directly on the ECUcore-5484. Therefore, on the development system a directory is mounted into the local file system of the ECUcore-

5484. To start the program, only the appropriate command must then be entered on the ECUcore-5484. The data transmission from the development system to the ECUcore-5484 takes place automatically by NFS, without any other commands of the user. The method NFS is described in section 8.3.1.

FTP: The “File Transfer Protocol” is a standardized and platform independent protocol that is securely established in practice. Both, FTP server and clients are available for different operating systems such as Linux and Windows. In contrast to NFS, FTP enables the transmission of data from a computer using Windows to the ECUcore-5484 (e.g. program updates of a service technician using a Windows laptop). Moreover, FTP allows for a detailled access control through username and password authentication. Disadvantageous for FTP is that a command must be entered for each file transmission – which can be annoying or even forgotten especially during the development phase. Consequently, it could be possible that perhaps an old version of the program is accidentally run. The method FTP is described in section 8.3.2.

8.3.1 Using NFS

Mounting a directory on the development system directly in to the local file system of the ECUcore-5484 represents the simplest way of starting a demo application (that has been built on the development system) on the ECUcore-5484. Therefore, a NFS server must be installed and configured on the development system.

1. Mounting a directory on the ECUcore-5484

To integrate the directory “/tftpboot” of the development system into the local file system of the ECUcore-5484, the command “mount” is to be used as follows:


```
mount  -t nfs  -o nolock  <ip_host>:/tftpboot  /mnt/nfs
```

After the command "mount" is successfully completed, the directory "/tftpboot" of the development system (including any subfolders) is accessible in the local directory "/mnt/nfs" of the ECUcore-5484.

Afterwards, the built demo application (e.g. epl.ko) has to be copied into the directory "/tftpboot" of the development system.

8.3.2 Using FTP

As an alternative to mounting a directory of the development system directly into the local file system of the ECUcore-5484 via NFS, it is possible to transfer data between the development system and the ECUcore-5484 in both directions via FTP.

<u>FTP Download</u>

Downloading files from the development system on the ECUcore-5484 takes place with the help of the command "ftpget". Both parameters, "-u" for username and "-p" for password enable authetication at the host system. The command "ftpget" is written as follows:

```
ftpget -u <username>  -p <password>  <ip_host>
   <local_file>  <remote_file>
```

To transmit the EPL demo application "eplapp" from the directory "/tftpboot" to the ECUcore-5484 via FTP for example, the following command is necessary:

```
ftpget  -u systec  -p systec  192.168.10.56
   /var/epl.ko  /tftpboot/epl.ko
```

<u>FTP Upload</u>

Uploading files from the ECUcore-5484 to the development system takes place with the help of the command “ftpput”. Both parameters, “-u” for username and “-p” for password enable authetication at the host system. The command “ftpput” is written as follows:

```
ftpput -u <username>  -p <password>  <ip_host>
   <remote_dir>  <local_file>
```


Glossary

AMI	Abstract memory interface
ASnd	EPL frame type: Asynchronous Send, which may contain SDO or NMT messages
CAL	Communication Abstraction Layer, internal openPOWERLINK Stack module
CN	Controlled Node, i.e. slave device in the POWERLINK network
DCF	Device configuration file (generated by configuration tools)
DLL	Data Link Layer
DNS	Domain Name System (Internet Protocol)
EPL	Ethernet POWERLINK
EPSG	Ethernet POWERLINK Standardization Group
IP	Internet Protocol
HMI	Human machine interface
MAC	Media Access Control
MN	Managing Node, i.e. master device in the POWERLINK network
NMT	Network Management
node	an arbitrary POWERLINK device. Often an POWERLINK CN
OBD	Object dictionary module
OD	Object dictionary
PDO	Process Data Object
PReq	EPL frame type: Poll Request

PRes	EPL frame type: Poll Response
RPDO	Receive PDO
SDO	Service Data Object
SoA	EPL frame type: Start of Asynchronous
SoC	EPL frame type: Start of Cyclic
TFTP	Trivial File Transfer Protocol
TCP	Transmission Control Protocol
TPDO	Transmit PDO
UDP	User Datagram Protocol

References

EPSG DS 301 Ethernet POWERLINK Communication Profile Specification V1.1.0. Ethernet POWERLINK Standardization Group, 2008

L-1098 Software Manual openPOWERLINK Protocoll Stack. SYS TEC electronic GmbH, 2008

L-1108 Introduction into openPOWERLINK. SYS TEC electronic GmbH, 2008

Document:	openPOWERLINK Development Kit
Document number:	L-1104e_2, Edition February 2009

How would you improve this manual?

___

___

___

Did you find any mistakes in this manual? page

___

___

___

Submitted by:

Customer number: ______________________

Name: ______________________

Company: ______________________

Address: ______________________

Return to:

SYS TEC electronic GmbH
August-Bebel-Str. 29
D-07973 Greiz
GERMANY
Fax : +49 (0) 36 61 / 62 79 99

Published by

Ordering No. L-1104e_2
Printed in Germany