

Feasibility Study

PROFINET IO Device Integration into a FPGA

Version 1.2

July 5, 2007

Softing AG

Richard-Reitzner-Allee 6

D-85540 Haar

Author: Dirk Palme

© 2007 SOFTING AG

No part of these instructions may be reproduced (printed material, photocopies, microfilm or other method) or processed, copied or distributed using electronic systems in any form whatsoever without prior written permission of SOFTING AG.

The producer reserves the right to make changes to the scope of supply as well as changes to technical data, even without prior notice. A great deal of attention was made to the quality and functional integrity in designing, manufacturing and testing the system. However, no liability can be assumed for potential errors that might exist or for their effects. Should you find errors, please inform your distributor of the nature of the errors and the circumstances under which they occur. We will be responsive to all reasonable ideas and will follow up on them, taking measures to improve the product, if necessary.

We call your attention to the fact that the company name and trademark as well as product names are, as a rule, protected by trademark, patent and product brand laws.

All rights reserved.

Table Of Contents

1	Task	2
1.1	Overview	2
1.2	Requirements.....	2
2	Prototypical Portation	3
2.1	Hardware Platform	3
2.2	Operating System and IP Stack	4
2.3	Tool Chain.....	5
3	PROFINET IO Application	5
3.1	STEP 7 Configuration	5
3.2	Device Demo Application	6
3.3	Ethernet Topology.....	7
3.4	Measuring Results	7
3.5	Memory Usage, Number of Used Logic Cells / Memory Blocks.....	9
4	Validation and Outlook.....	10
4.1	Optimization Potential	10
4.2	Cost Considerations.....	10

Abbreviations and Terms

DMIPS Dhystone Million Instructions Per Second

References

[1] Datasheet DBC2C20 Cyclone@II Development Board

1 Task

1.1 Overview

In the area of industrial automation Softing is offering services regarding the integration of communication systems to customer specific hardware and software platforms for device and controller manufacturers. Here, in general, the portable Softing implementations of communication protocols, e.g. for PROFINET IO, PROFIBUS or Foundation Fieldbus are used.

For future device generations Softing has examined the development of a flexible hardware solution, which can be used as a platform for various Real-Time Ethernet protocols. With this type of system an FPGA from Altera is used as a core component. This platform can be adapted to various requirements by configuring one or multiple NIOS-II processors

As a first step, this hardware platform has been used for implementing a prototype PROFINET IO device.

1.2 Requirements

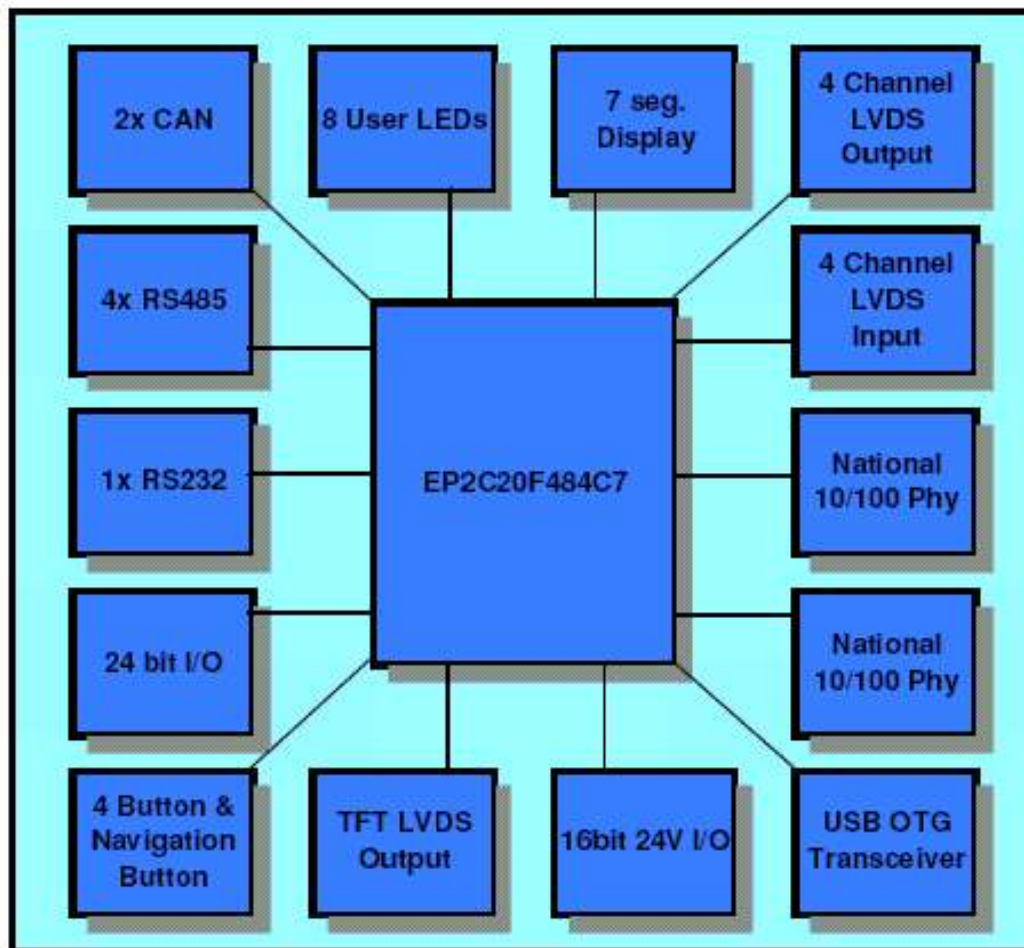
From Softing's point of view the central requirements are:

- Achievement of a stable communication behavior for the transfer of cyclic process data with a cycle of 1ms
- Realization of PROFINET IO Conformance Class B, e.g. integration of redundancy and LLDP into the PROFINET IO stack
- Integration of a 2 Port Switch and a MAC with IEC1588 functionality

2 Prototypical Portation

2.1 Hardware Platform

For the scope of this feasibility study the Altera evaluation board DB C2C35 Cyclone®II Development Board has been used. A detailed description of the hardware platform can be found in [1] (The only significant difference between the used evaluation board DB C2C35 and the described evaluation board DB C2C20 is the number of available logic elements).



Block Diagram of the Evaluation Board (Source: [1])

- For the analysis the following components of the evaluation board have been used:
- FPGA with processor core NIOS-II using configuration f (Processor rate 75MHz, 4KByte Instruction Cache and 2KByte Data Cache) and Ethernet MAC of MoreThanIP
 - National 10/100 PHY
 - 7 Segment Display
 - RAM
 - ROM

The used configuration of the NIOS-II processor core, which is used for the scope of this feasibility study defines a performance class in the lower midfield. When using a maximum cache size for data and instruction cache of 64KByte at 75MHz this is given with a performance of 76 DMIPS. In the highend configuration, which however cannot be reached with a Cyclone II type FPGA, the NIOS-II processor core provides a performance of 185 DMIPS.

The used Ethernet MAC of MoreThanIP allows to send and receive data via DMA. At the moment therefore however only a half duplex mode is supported, while the data transfer via Ethernet is performed in full duplex mode. The send and receive FIFO allows to intermediately store a package of maximum data length after it is read from the memory or sent into the memory.

2.2 Operating System and IP Stack

As operating system the real-time operating system μ C/OS-II is used, which is available with the installation of the NIOS-II development environment. This operating system provides all functionality, which is required for the Softing PROFINET IO implementation. Also for supporting the used hardware configuration all necessary drivers are provided.

The rate of the operating system has been set from 100Hz to 1000Hz for implementing a send and receive cycle of 1ms.

Two modifications have been performed to the Ethernet driver to allow send and receive of Ethernet packages without the IP stack. For receiving Ethernet packages the Ethernet driver interface has been enhanced by a Receive Callback function, which forwards the data directly after the data transfer from the FIFO of the Ethernet MAC to the local receive buffer to the appropriate Callback function. This way the received package first will be forwarded to the PROFINET IO stack and then added to the standard receive queue, if the PROFINET IO stack does not handle it. For sending the synchronous transmit function of the Ethernet driver is used as this is done also by the IP stack.

For the communication of the PROFINET IO implementation via UDP the Interniche IP stack has been used. Amongst others, this stack is distinguished by an optimized package handling without additional copying actions. The option "Zero copying" however has not been used. In contrary to BSD compliant IP stacks the Interniche implementation offers an access capability to UDP packages below the socket interface. Here however a correct handling and return of the receive and send buffer to the UDP level has to be ensured during the portation. The advantage, on the other hand, can be seen in a lesser code size and a reduced resource usage.

2.3 Tool Chain

Together with the NIOS-II development environment a NIOS-II C/C++ project has been created and the standard settings of the compiler and linker has been used. These define that code for the Release Mode is generated and the optimization level `-O2` is used.

3 PROFINET IO Application

3.1 STEP 7 Configuration

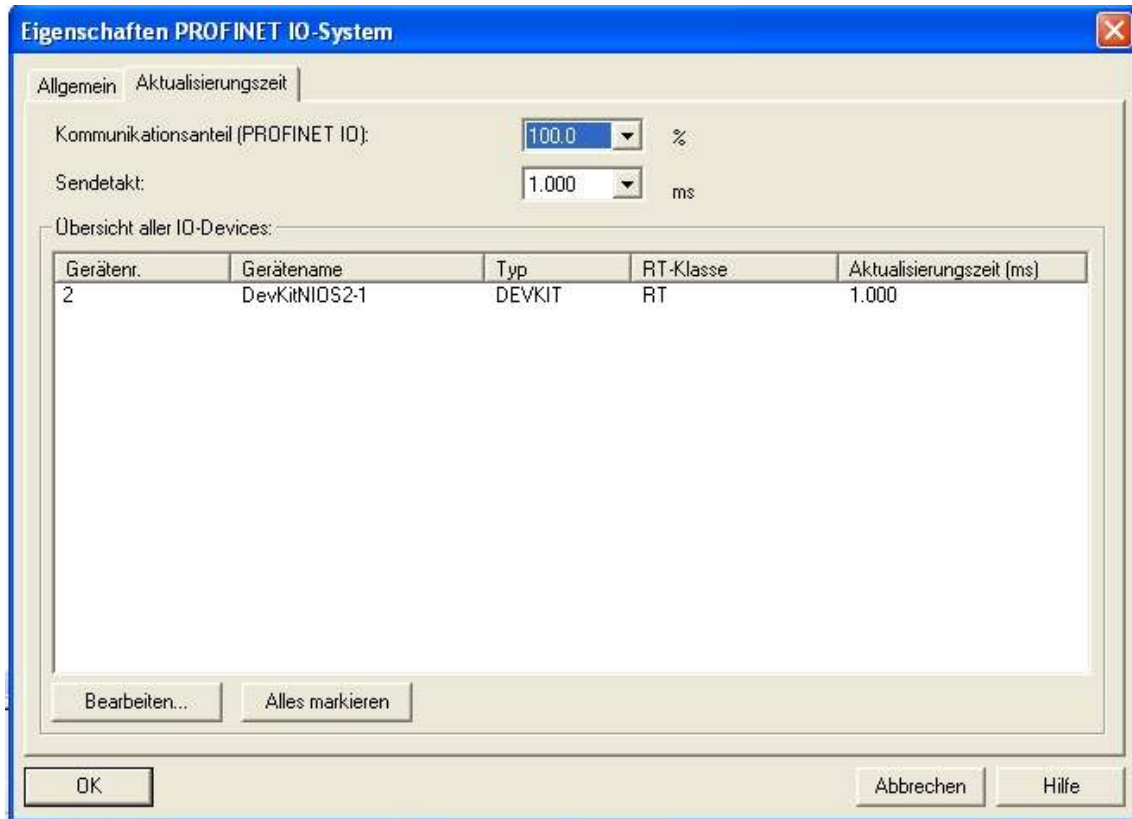
A Siemens PLC has been used for commissioning the evaluation board together with the prototypical portation of the PROFINET IO stack. The S7 317-2 PN/DP had been configured using the engineering tool STEP 7 Version 5.4 SP1. The configuration includes one PROFINET IO device based on the GSDML file as defined by Softing.

Steckplatz	Baugruppe	Bestellnummer	E-Adresse	A-Adresse	Diagnoseadresse
0	DevKitNIOS2-1	PN-DevKit-01			8190*
1	8DI DC24V HF	PN-DevKit-02	0		
2	16DI DC24V ST	PN-DevKit-03	1...2		
3	8DO DC24V HF	PN-DevKit-04		0	
4	16DO DC24V ST	PN-DevKit-05		1...2	
5	2AI 10V	PN-DevKit-06	256...259		
6	2AI 5V	PN-DevKit-07	260...263		
7	2AO 10V	PN-DevKit-08		256...259	
8	2AO 5V	PN-DevKit-09		260...263	
9					
10					

Device Configuration using STEP 7

The PROFINET IO device received a typical module extension of 4 digital Input/Output modules and 4 analog Input/Output modules. Based on this 20Byte of process data including status information are transferred per cyclic connection.

As a next step the setting of the cycle time for the transfer of the process data is set. 100% of the bandwidth has been reserved for PROFINET IO and the sending rate of the PLC has been set to 1ms. For the only configured device an actualization rate of 1ms has been set. The calculation of the monitoring time has been performed by STEP 7 automatically and configured to the triple value of the actualization time.



Setting the Actualization Time of Cyclic Process Data

3.2 Device Demo Application

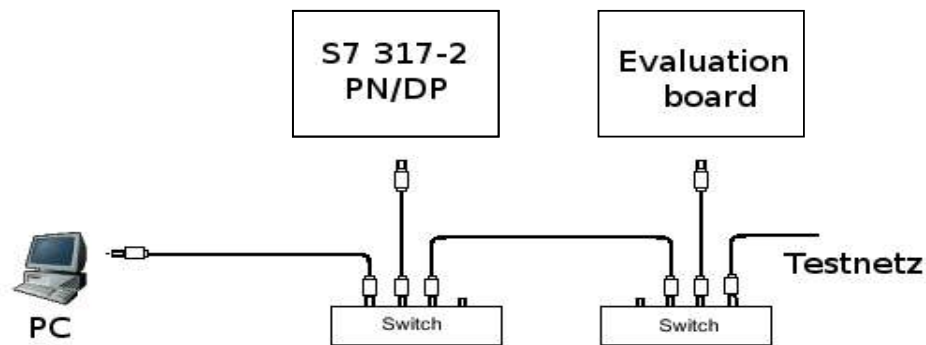
The standard demo application for the Softing PROFINET IO stack has been used as device demo application. This has been compiled via compiled-in header files according to the configuration defined by SETP 7. During the system start-up the callback functions for reading the consumer data and for writing the provider data have been registered at the PROFINET IO stack. These functions however are empty in the demo application, thus, as a result, a data exchange with simulated process values is not supported.

The setup and teardown of a cyclic connection has been monitored. As the PLC is performing a re-initialization of the connection after a connection abortion of a cyclic data exchange, a timer has been implemented, which is incremented with each new setup of the cyclic connection and displayed on a 7 segment display. During a multi-days-test no abortions could be found. A succeeding test including the disconnection and re-connection of the switch connection at once has shown the re-parameterization of the cyclic data.

The execution of alarms has not been implemented.

3.3 Ethernet Topology

The test set-up has been connected to the Softing test network. As PROFINET IO is based on switched Ethernet a negative impact from other network members can be ruled out.



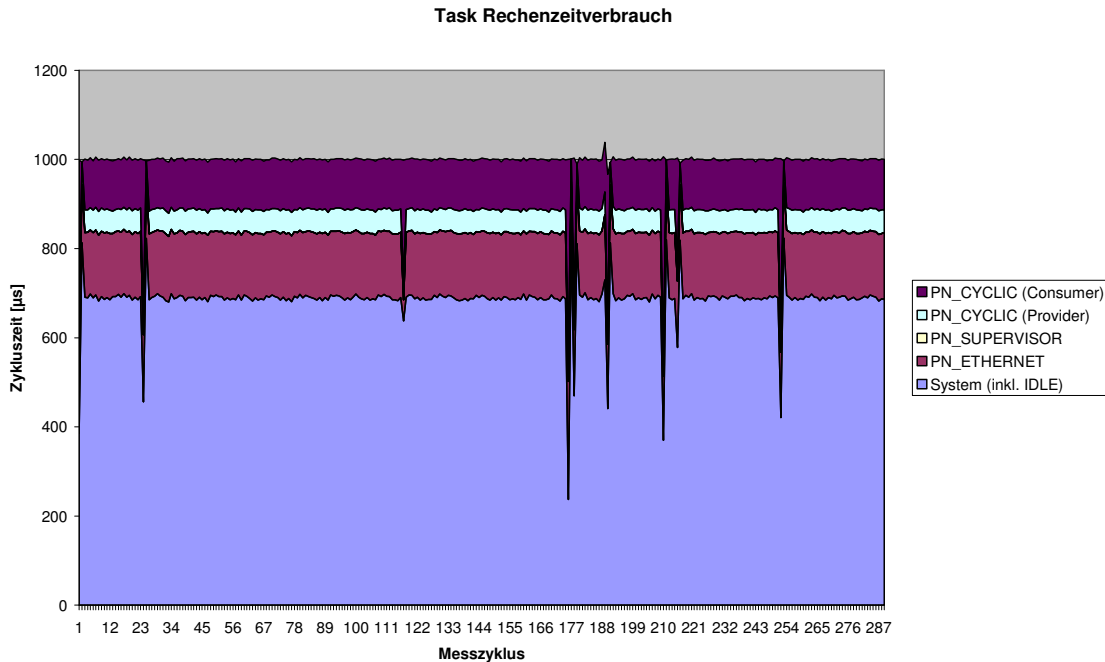
Test Set-up of the PROFINET IO Network

Only a slightly increased number of packages with multicast and broadcast addresses has been found. This number has been in the range of $\leq 5\%$ of the monitored data traffic, however they occurred as bursts. At the PC connected to the mirror port of the switch the PLC S7 317-2 PN/DP has been configured and afterwards the data traffic between the PLC and the evaluation board has been monitored.

3.4 Measuring Results

For determining the processor usage of various tasks for a certain application case it would be necessary for the used real-time operating system to support the run-time measurement by appropriate functions. This would allow a detailed overview about all tasks and interrupt functions of the system. Like also $\mu\text{C}/\text{OS-II}$ most operating systems however do not support this functionality.

Thus for measuring the task timing the mechanisms of the PROFINET IO stack implementation have been used, which are somewhat more imprecise. Here the measurement can only be performed for those tasks, which are part of the PROFINET IO stack. Interrupts of the tasks used by the PROFINET IO stack by system tasks and interrupts with higher priorities cannot be recognized and thus increase the time of the interrupted task of the PROFINET IO stack. In addition the available idle times cannot be determined. For the measurement a high resolution hardware timer with 1ms resolution has been used.



After the configuration and parameterization of the PROFINET IO device by the PLC (including the set-up of the cyclic connection) the timing measurement of the tasks has been activated in the PROFINET IO stack of the evaluation board using manufacturer specific DCP services. As soon as the the cyclic data exchange has been established the following tasks are running in the PROFINET IO stack:

- Both tasks for handling the input and output data (Consumer und Provider),
- Supervisor task for monitoring the system status and
- Ethernet task for sending and receiving the cyclic and acyclic Ethernet packages

Based on the above mentioned statements regarding the measurement method the results shown in the diagram can be interpreted as follows:

- (1) A cycle time of 1ms can be reached using the tested hardware configuration.
- (2) Peak load situations primarily are caused by multicast and broadcast packages, as these usually are analyzed by different protocol levels (DCP, ARP, IP, or UDP).
- (3) The system load by the implementation of the PROFINET IO stack is about 30-35%. About 15% are caused by the handling of cyclic data (Consumer and Provider tasks) and about 15% are caused by receiving and sending Ethernet packages (Ethernet task).
- (4) Sporadic raise of the processor time consumption by the PROFINET IO implementation are based on interrupts by system tasks and interrupts of higher priorities.

3.5 Memory Usage, Number of Used Logic Cells / Memory Blocks

- (1) Code size, including application, operating system and IP stack: 600KByte
- (2) Data memory for static and dynamic variables, without application, operating system and IP stack: about 300KByte
- (3) The PROFINET IO code has been executed from the slower SD-RAM.
- (4) The NIOS-II FPGA type F is using 8,900 logic cells as well as 37 M4Ks memory.
- (5) The used switch in the 3 port configuration is using 4,900 logic cells as well as 10 M4Ks memory.
- (6) The MACs with IEC588 functionality are using 4,400 logic cells as well as 21 M4Ks memory each.

4 Validation and Outlook

The prototypical portation of the Softing PROFINET IO stack could fulfill the requirements regarding the hardware and software configuration constraints as mentioned above. Devices with an application of a similar size as the PROFINET IO demo application may use one NIOS-II processor for the Softing PROFINET IO stack as well as the application. Applications with a larger functionality respectively with higher requirements regarding minimal latency times require a more powerful configuration of the processor core or a separate processor. In the later case the PROFINET IO stack is executed on one processor, while the application is running on a second processor. Here the communication between both processors is performed via a DPRAM respectively Shared RAM and interrupts. Independently from the used hardware configuration (one versus two processor cores) there is no difference regarding the interface between the PROFINET IO stack and the application.

4.1 Optimization Potential

- The NIOS-II processor has a Fast Interrupt Mode available, which allows to nearly shorten the interrupt latency times by one half.
- Support of the hardware supported conversion of data formats (Little- und Big-Endian)
- Depending on the used hardware fast S-RAM memory can be used. Alternatively also the cache sized of the SD-RAM can be doubled.

The used PROFINET IO stack covers the Conformance Class A functionality. The stack extensions for supporting Conformance Class B primarily will increase the code and data memory size. In addition also the computing time of the PROFINET IO stack tasks will increase. While this increase of the computing time probably will be in the range of $\leq 10\%$, the increase of the memory consumption will be in the range of about 50%.

4.2 Cost Considerations

In the range of this investigation also the cost side for architectures have been examined for a production batch of 1,000 units each:

<u>Architecture</u>	<u>Used FPGA</u>	<u>Costs PROFINET IO Interface</u>
FPGA with 2 x NIOS CPU and integrated switch	Cyclone 3C40	about EUR 36.-
FPGA with 2 x NIOS CPU and external switch	Cyclone 3C16	about EUR 42.-
FPGA with 1 x NIOS CPU and 1 Ethernet Port	Cyclone 3C16	about EUR 25.-