SHIELD PILOT

Pilot SHIELD

pilot embedded Systems
arcHItecturE for multi-Layer Dependable solutions

SEVEN FRAMEWORK PROGRAMME

Project no: 100204

pSHIELD

pilot embedded **S**ystems arc**HI**tectur**E** for multi-**L**ayer **D**ependable solutions

Instrument type: Capability Project

Priority name: Embedded Systems / Rail Transportation Scenarios

SPD power node technologies prototype report

For the pSHIELD-project

Deliverables D3.3

Partners contributed to the work:

Eurotech, Italy
SESM, Italy
Acorde Seguridad, Spain
CWIN, Norway

Project co-funded by the European Commission within the Seventh Framework Programme (2007-2012)		
	Dissemination Level	
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Document Authors and Approvals

Authors		Date	Signature
Name	**Company**		
Przemyslaw Osocha	SESM		
João Cunha	SESM		
Emilio Bisbiglio	SESM		
Fabio Giovagnini	SESM		
Paolo Azzoni	ETH		
Silvia Mier	AS		
Josef Noll	MAS		
Zahid Iqbal	CWIN		
Reviewed by			
Name	**Company**		
Approved by			
Name	**Company**		

Modification History

Issue	Date	Description
Draft A	17 June 2011	First ToC proposal for comments
Draft B	9 September 2011	Incorporates comments from Draft A review
Issue 1	22 December 2011	Incorporates comments from Draft B review
Issue 2		Incorporates comments from issue 1 review

Contents

Figures

Tables

Glossary

ESs	Embedded Systems
SPD	Security Privacy Dependability
FSK	Frequency-Shift Keying
AFSK	Audio Frequency-Shift Keying
UCS	Use case Scenario
HW	Hardware
SW	Software

This Page is intentionally left blank

Executive Summary

Deliverable D3.3 "SPD power node technologies prototype report" is document covering output from Task 3.2 "Power node" of work package WP3 "SPD Node". The preliminary works description is covered in deliverable D3.1 "SPD node technologies prototype" which comprising output from all tasks of work package WP3. Presented deliverable is a report of works that have been done in Task 3.2. The deliverable presents mainly technical aspects, with prototypes description. More architectural approach could be found in deliverables D2.1.1 "System Requirements and Specification" and D2.3.1 "Preliminary System Architecture Design".

The structure of D3.3 is divided into several chapters presenting after short introduction, the overall pSHIELD SPD Node Layer Architecture, and then describing prepared prototypes from SESM, ETH and CWIN. In the end the power management of power node is presented. The documents ends with short conclusions.

General objectives of WP3 follow:

- Select a representative set of SPD technologies at Node level;
- Develop appropriate composability mechanisms at such level;
- Deliver a SPD node prototype.

WP3 plays important role in designed four layers pSHIELD architecture, representing the basic components of the lower part of the SPD Pervasive System: Node Layer.

Work package 3 works interact with other project tasks, e.g. contribution coming from research and development performed in WP2 and WP4, which are strictly interconnected and interdependent with WP3 and results will be used in WP6. Task 2.1 provides the requirements and specification for a prototypes. Task 2.3 provides definitions of proper interfaces that will allow the nano, micro/personal nodes interoperation with the rest of SHIELD platform. WP4 provides Task 3.3 with SPD features at network level to be implemented at node level.

The aim of deliverable D3.3 is to report solutions selected and implemented in Task 3.2 to fulfill work package goals.

1 Introduction

Work Package 3 covers problematic of 3 different kinds of Intelligent ES Nodes: nano node, micro/personal node and power node. These three node types represent the basic components of pSHIELD architecture, creating Node Layer, one of four layers, beside Network, Middleware and Overlay Layers.

The WP aims at providing SPD intrinsic capabilities at node layer through the creation of an Intelligent ES HW/SW Platform consisting of three different kinds of Intelligent ES Nodes: nano node, micro/personal node and power node. These three node types (which can be considered three node levels of increasing complexity) will represent the basic components of the lower part of the SPD Pervasive System, and will cover the possible requirements of several market areas: from field data acquisition, to transportation, to personal space, to home environment, to public infrastructures, etc.

Objectives of Work package 3 "SPD Node" are: selection of a representation of SPD technologies at Node level, development of appropriate composability mechanisms at node level, and deliver a SPD node prototype.

Aim of this deliverable D3.3 is to present SPD power node technologies prototype report. Prototypes of such SPD technologies were developed, following the composability criteria of the pSHIELD architecture design delivered by WP2.

Nodes definitions

pSHIELD SPD Architecture is composed of four layers:

- Node Layer,
- Network Layer,
- Middleware Layer,
- Overlay Layer.

Node Layer represents the basic components of the lower part of the SPD Pervasive System.

That layer consisting of three different kinds of ES Nodes which can be considered three node levels of increasing complexity:

- Nano Node,
- Micro/Personal Node,
- Power Node.

Nano Node level typically consists of small, mainly wireless sensors, with limited HW and SW resources. Because of their massive distribution in the environment, they could become an interesting target for attacks and hacking.

Micro/Personal Node level consists of devices richer than the Nano Nodes in terms of hardware and software resources, network access capabilities, mobility, interfaces, sensing capabilities etc. The specific functions of a Micro/Personal Node are generally referred to:

- secure network access capabilities,
- monitoring and sensing,
- interfacing.

Power Node level represents, in the pervasive system, the first level of massive data elaboration, with the peculiarity that the computing power is provided directly on the field.

The most powerful kind of nodes is topic of interest in Task 3.2. Power Nodes ESs can provide high performance allowing massive data elaboration on the field.

The goals of Task 3.2 include works on and development of:

- HPC ES used on the field without the limitations of a classical HPC solution like working conditions, energy consumption, dimensions, etc.
- Self contained board that will take care of storage, networking, memory and processing, all devices soldered on board, increasing robustness.
- Development of a new approach for FPGA runtime reconfiguration to increase the nodes dependability. Dependability usually involves HW redundancy and system costs. Solution is use of FPGAs that are intrinsically redundant. They allow runtime reconfiguration during normal operation or fault, and either hardware or software changes. That allows to reduce component count, power consumption, reusing, fault tolerance, etc..
- Alternatives for low power ES nodes with SPD features, take into account the size and power constrains of Power Nodes.

The document presents below results of conducted works.

2 SPD Power Node Layer Architecture

A pSHIELD Node is an embedded system device equipped with several legacy Node Capabilities and with a pSHIELD Node Adapter. A pSHIELD Node is deployed as a hardware/software platform, encompassing intrinsic, innovative SPD functionalities, providing proper services to the other pSHIELD Network and Middleware Adapters to enable the pSHIELD Composability and consequently the desired system SPD.

There are three kinds of pSHIELD Nodes deploying each different configuration of Node Layer SPD functionalities of the pSHIELD framework, and comprising different types of complexity: Nano nodes, Micro/Personal nodes and Power nodes.

- Nano nodes are typically small ES with limited hardware and software resources, such as wireless sensors.
- Micro/Personal nodes are richer in terms of hardware and software resources, network access capabilities, mobility, interfaces, sensing capabilities, etc.
- Power nodes offer high performance computing in one self-contained board offering data storage, networking, memory and (multi-)processing.

This document is concerned with Power Node Architectures.

2.1 Formal conceptual model

Figure 2.1 provides a conceptual model of a pSHIELD Node Layer. Although this is a generic model for all the pSHIELD Node types, it includes some capabilities that are present exclusively in Power Nodes, such as Security and Pivacy, Power Management and Reconfiguration.

These nodes can be implemented in different architectures, providing different functionalities, different SPD compliance levels and different capabilities, depending on the type of node and application field.

Sections 3 to 5 present different implementations of Power Nodes, and section 6 addresses specifically the Power management capabilities for Power Nodes.

Acording to this model, SPD Node architecture is composed of different functional blocks and each one can implement several features of different complexity and performance. The choice of which features to implement depends on the application scenario and SPD Compliance Level required for the system.

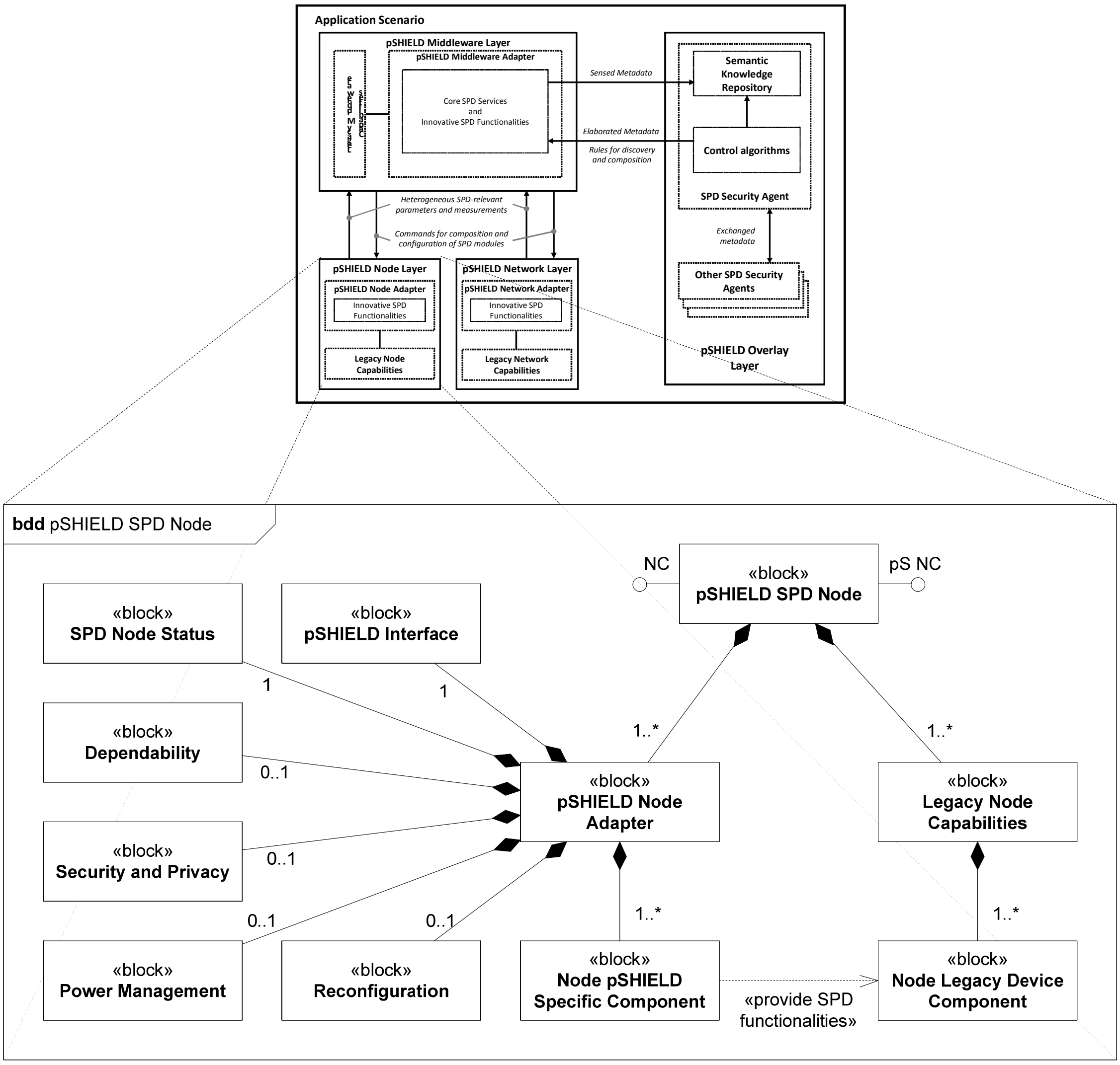

Application Scenario
pSHIELD Middleware Layer
pSHIELD Middleware Adapter
Core SPD Services and Innovative SPD Functionalities
Sensed Metadata
Elaborated Metadata
Rules for discovery and composition
Semantic Knowledge Repository
Control algorithms
SPD Security Agent
Exchanged metadata
Other SPD Security Agents
pSHIELD Overlay Layer
Heterogeneous SPD-relevant parameters and measurements
Commands for composition and configuration of SPD modules
pSHIELD Node Layer
pSHIELD Node Adapter
Innovative SPD Functionalities
Legacy Node Capabilities
pSHIELD Network Layer
pSHIELD Network Adapter
Innovative SPD Functionalities
Legacy Network Capabilities
bdd pSHIELD SPD Node
NC
«block» pSHIELD SPD Node
pS NC
«block» SPD Node Status
«block» pSHIELD Interface
«block» Dependability
«block» Security and Privacy
«block» Power Management
«block» Reconfiguration
«block» pSHIELD Node Adapter
«block» Legacy Node Capabilities
«block» Node pSHIELD Specific Component
«block» Node Legacy Device Component
«provide SPD functionalities»
1
1
0..1
0..1
0..1
0..1
1..*
1..*
1..*
1..*

Figure 1 – Formal conceptual model of pSHIELD SPD Node Layer

The formal conceptual model of a generic pSHIELD Node Layer can be derived from the pSHIELD functional component architecture. The pSHIELD Node Layer is an Embedded System Device (ESD) equipped with several Legacy Node Capabilities and a pSHIELD Node Adapter. The pSHIELD Node Layer is deployed as a hardware/software platform, encompassing intrinsic, innovative SPD functionalities, providing proper services and capabilities to the pSHIELD Middleware Adapters to enable the pSHIELD Composability and consequently the desired system SPD.

The **pSHIELD SPD Node Layer** has two interfaces, one providing pSHIELD Node Capabilities (**pS-NC**) to the pSHIELD Middleware Layer, and another with legacy, technology-dependent, Node Capabilities (**NC**).

The pSHIELD SPD Node is composed of **Legacy Node Capabilities**, which consist of one or more **Legacy Device Components**, such as CPU, I/O Interfaces, Memory, Battery, etc., and a **pSHIELD Node Adapter**, interacting with the legacy ESDs and providing SPD functionalities.

The **pSHIELD Node Adapter** includes a set of Innovative SPD functionalities interoperating with the legacy node capabilities in order to enhance them with the pSHIELD Node Layer SPD enabling technologies. This adapter is in charge to provide (through the pS-NC interface) all the needed information to the pSHIELD Middleware adapter to enable the SPD composability of the Node layer legacy and Node pSHIELD-specific functionalities. Moreover, the pSHIELD Node Adapter translates the technology independent commands, configurations and decisions coming from the pS-NC interface into technology dependent ones and enforce them also to the legacy Node functionalities through the NC interface.

The different Node Layer Innovative SPD Functionalities (i.e. SPD components) are grouped into proper **modules** containing a functional subsets of the Innovative SPD capabilities provided by the pSHIELD Node. In brief, the main modules of a generic **pSHIELD Node Adapter** are:

- **pSHIELD Interface**, which provides a proper interface to the pSHIELD Network.
- **SPD Node Status**, responsible for collecting the status of each individual component, and providing SPD-relevant parameters and measurements to the Middleware Layer. It also checks on system health status for self-recovery, self-reconfiguration and self-adaptation.
- **Reconfiguration**, which performs module or system reconfiguration by demand of the system SPD Node Status or the Middleware.
- **Dependability**, responsible for applying self-dependability at node layer, by detecting problems related to system health status and starting recovery. It is also responsible for collecting checkpoints from the remaining pSHIELD Node Adapter modules, and retrieving this information during system recovery.
- **Security and Privacy**, enforcing system security and privacy at node level, by providing hardware or software encryption, decryption, key generation, firmware protection, etc.
- **Power Management**, module for managing power sources, providing protection against blackouts, etc.
- **Node pSHIELD Specific Components**, which are the innovative SPD functionalities provided to each of the Legacy Device Components, such as status and metrics, checkpoint-recovery,

2.2 Node Layer Interface

Besides providing the access to legacy, technology-dependent, Node Capabilities (NC) from third-party or of-the-shelf device components, the pSHIELD SPD Node Layer has a specific pSHIELD Node Capabilities interface (pS-NC) to the pSHIELD Network and pSHIELD Middleware Layers, allowing SPD composability, Node pSHIELD-specific functionalities, or access to legacy Node capabilities.

2.3 Legacy Capabilities

The pSHIELD SPD Node includes Legacy Node Capabilities, which consist of one or more Legacy Device Components, such as CPU, I/O Interfaces, Memory, Battery, etc. A pSHIELD Node Adapter enhance these legacy capabilities by providing, through the pS-NC interface, all the needed information to the pSHIELD Middleware adapter to enable the SPD composability of the Node layer legacy and Node pSHIELD-specific functionalities. Moreover, the pSHIELD Node Adapter translates the technology independent commands, configurations and decisions coming from the pS-NC interface into technology dependent ones and enforce them also to the legacy Node functionalities through the NC interface.

2.4 Innovative SPD

Depending on the specific implementation of the pSHIELD Node, the pSHIELD Node Adapter may include a set of components providing Innovative SPD capabilities. In brief, the main components of a generic pSHIELD Node Adapter are:

- SPD Node Status, responsible for collecting the status of each individual component, and providing SPD-relevant parameters and measurements to the Middleware Layer. It also checks on system health status for self-recovery, self-reconfiguration and self-adaptation.
- Reconfiguration, which performs module or system reconfiguration by demand of the system SPD Node Status or the Middleware.

- Dependability, responsible for applying self-dependability at node layer, by detecting problems related to system health status, and starting recovery. It is also responsible for collecting checkpoints from the remaining pSHIELD Node Adapter modules, and retrieving this information during system recovery.

- Security and Privacy, enforcing system security and privacy at node level, by providing hardware or software encryption, decryption, key generation, firmware protection, etc.

- Power Management, for managing power sources, providing protection against blackouts, etc.

- Node pSHIELD Specific Components are the innovative SPD functionalities provided to each of the Node Legacy Device Components, such as status and metrics, checkpoint-recovery, etc.

Depending on the type of node, application, technology, etc. each of these modules may be implemented with different pSHIELD SPD functionalities or not implemented at all.

3 FPGA Based Prototype

In many industrial, telecommunications or transportation appliances the transmission of digital information through noisy environments makes use of Frequency-shift keying (FSK) due to its immunity to "adverse environment" conditions (i.e. electromagnetic interference, noise, surge, ground loop/ground plane shift problems), its ability to transmit data across commutators or sparking sources (sliding contacts, slip rings, rolling wheels, etc.), and the use of any two conductor wire, shielded or unshielded. Furthermore, it is employed even in wireless communications, such as in digital cellular communications system (GSM), using Gaussian minimum shift keying (GMSK), a special type of FSK.

This prototype proposes the use uses FSK modulation to transmit data to a SPD Power Node. The transmitted data is encrypted and modulated. The Power Node receives the signal, demodulates it, processes the data, encrypts it and sends to a control center through the pSHIELD Network.

3.1 FSK Demodulator prototype scenario

3.1.1 FSK Modulation

Frequency-shift keying[1] (FSK) is a frequency modulation scheme in which digital information is transmitted through discrete frequency changes of a carrier wave. The simplest FSK is binary FSK (BFSK). BFSK uses a pair of discrete frequencies to transmit binary (0s and 1s) information. With this scheme, the "1" is called the **mark frequency** and the "0" is called the **space frequency**. The time domain of an FSK modulated carrier is illustrated in the figures given below.

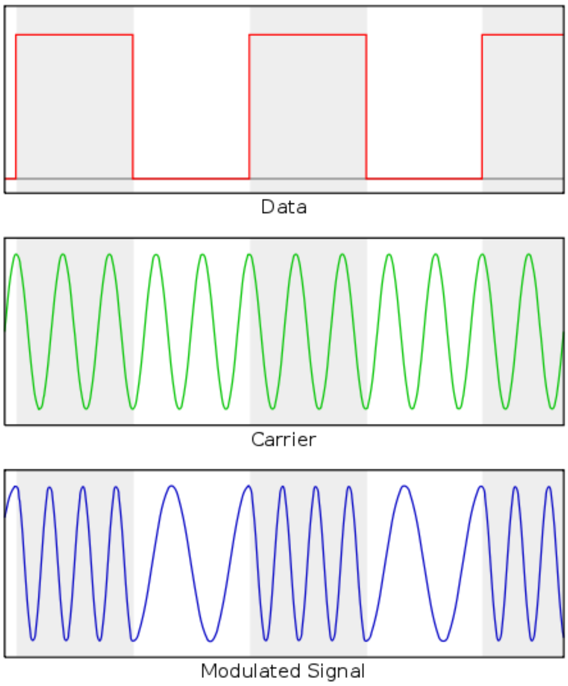


Figure 2 – FSK signal example

[1] After webpage: http://en.wikipedia.org/wiki/Frequency-shift_keying, accessed: 12.07.2011.

3.1.2 Power Node use-case scenario

This scenario demonstrates the Node Layer capabilities, but some Network, Middleware or Overlay functionalities may also be used.

This case study is composed of:

- An FPGA based board, continuously generating sample data.
- A data encryptor, running in the same board, encrypting this data.
- A FSK modulator, modulating the encrypted data into a FSK signal. This is also performed by the same FPGA based board
- A parallel 8 bit wide data bus with the synchronization clock line between the signal generator and the Power Node.
- A SPD Power Node, built using a Xilinx FPGA base board. This Node has a FSK demodulator, a data decryptor, and a web server, presenting the node status, metrics and received data.
- A fault-injector, activated by a pushbutton, able to inject a fault into the FPGA.
- A Control Center, which is a PC with a web browser.
- An Ethernet connection between the SPD Node and the Control Center.

The following scenarios shall be demonstrated:

- The signal generator continuously generates sampled data, encrypts, modulates, and transmits it to the SPD Node. The SPD Node demodulates the signal, decrypts and exposes its data in the web browser. The Control Center PC shows this web page with the same simulated distance.
- While the scenario 1. is executing, a fault is injected into the demodulator. An error is detected and recovered, by a FPGA reconfiguration. Correct data is still presented to the Control Center. The metrics reveal that an error has occurred, and recovery was successful.
- The Modulator switches to a different carrier. The SPD Node detects this error, and the demodulator is automatically reconfigured to this new carrier. On the Control Center, data is still valid. The status reveals a new carrier is being used.

3.1.3 SPD Capabilities Demonstration

This prototype shall demonstrate the following SPD capabilities and functionalities:

- Legacy component adaptation to pSHIELD, by providing SPD functionalities to a legacy FSK Demodulator.
- Dependability, by detecting errors in the demodulator, and tolerating them, through FPGA reprogramming.
- Security, by receiving encrypted data and being able to decrypt it.
- Self-Reconfiguration, by detecting that a different carrier is being used in the FSK signal, and reconfiguring the FPGA for the new carrier.
- Metrics, by collecting and providing data such as the number of messages received, errors detected, etc.
- Composability, by providing discovery and composability information, such as the identification of the modules and its characteristics, that build-up the SPD Node.

- High performance, by demodulating and decrypting in real-time all the received information.

3.2 FSK Demodulator Context

The context application, used to demonstrate how the innovative "SDP Node" could be compliant with security, dependability and privacy requirements of pSHIELD Project is showed in the following figure.

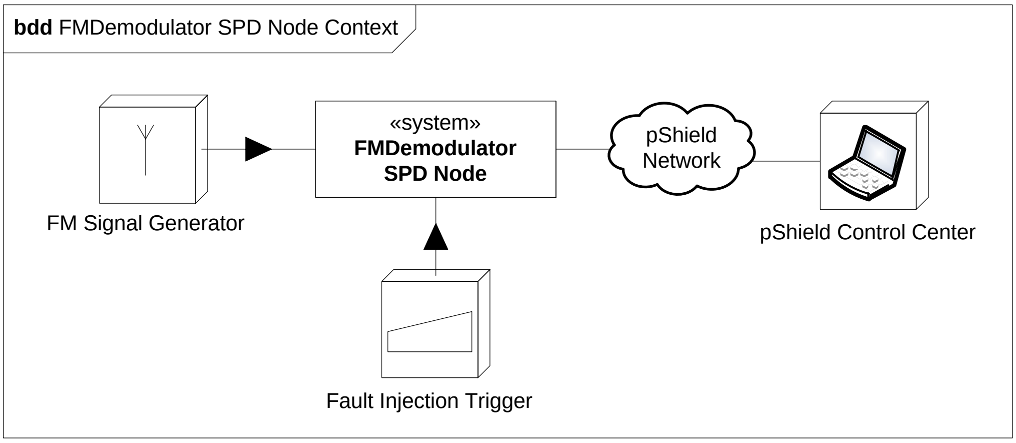


Figure 3 – A-FSK Demodulator demonstration context

The context application refers to an "**FSK Demodulator SPD Node**" which has been implemented and developed according to the general architecture that defines the features of a SPD Node device.

The FPGA Power Node Prototype may be remotely monitored and controlled using web interface.

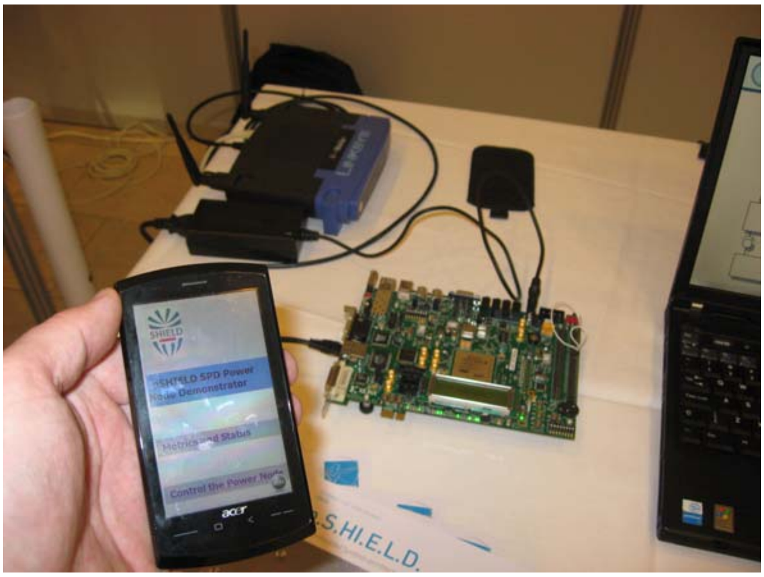


Figure 4 – Running FPGA Power Node Prototype

3.2.1 FSK Modulator

The **FSK Modulator** has been implemented using an ALTERA board. The board is the Altera cycloneIII_3c120_dev development kit.

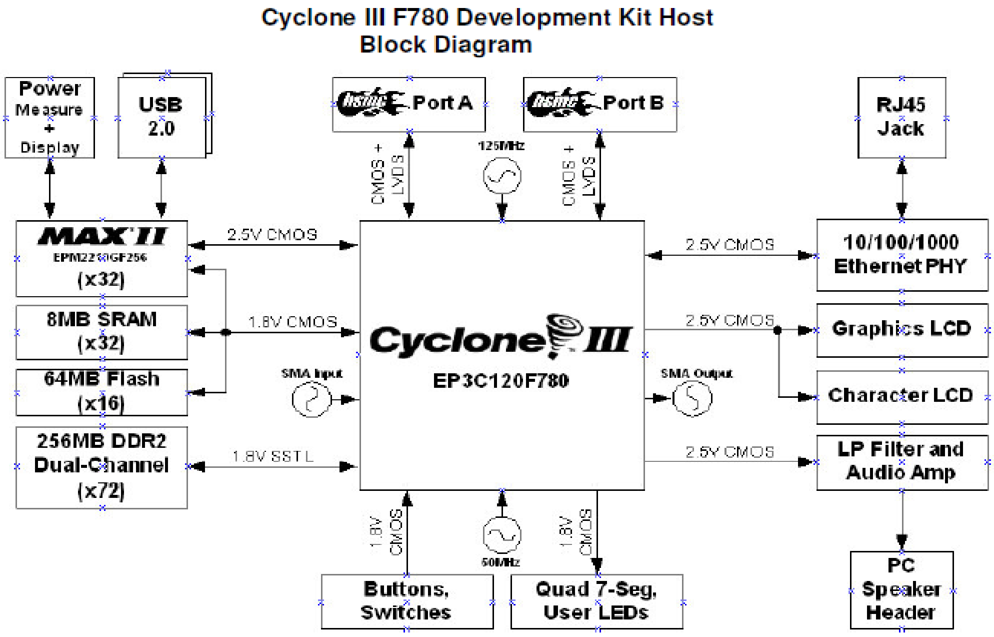


Figure 5 – Block diagram of FSK Modulator hardware structure

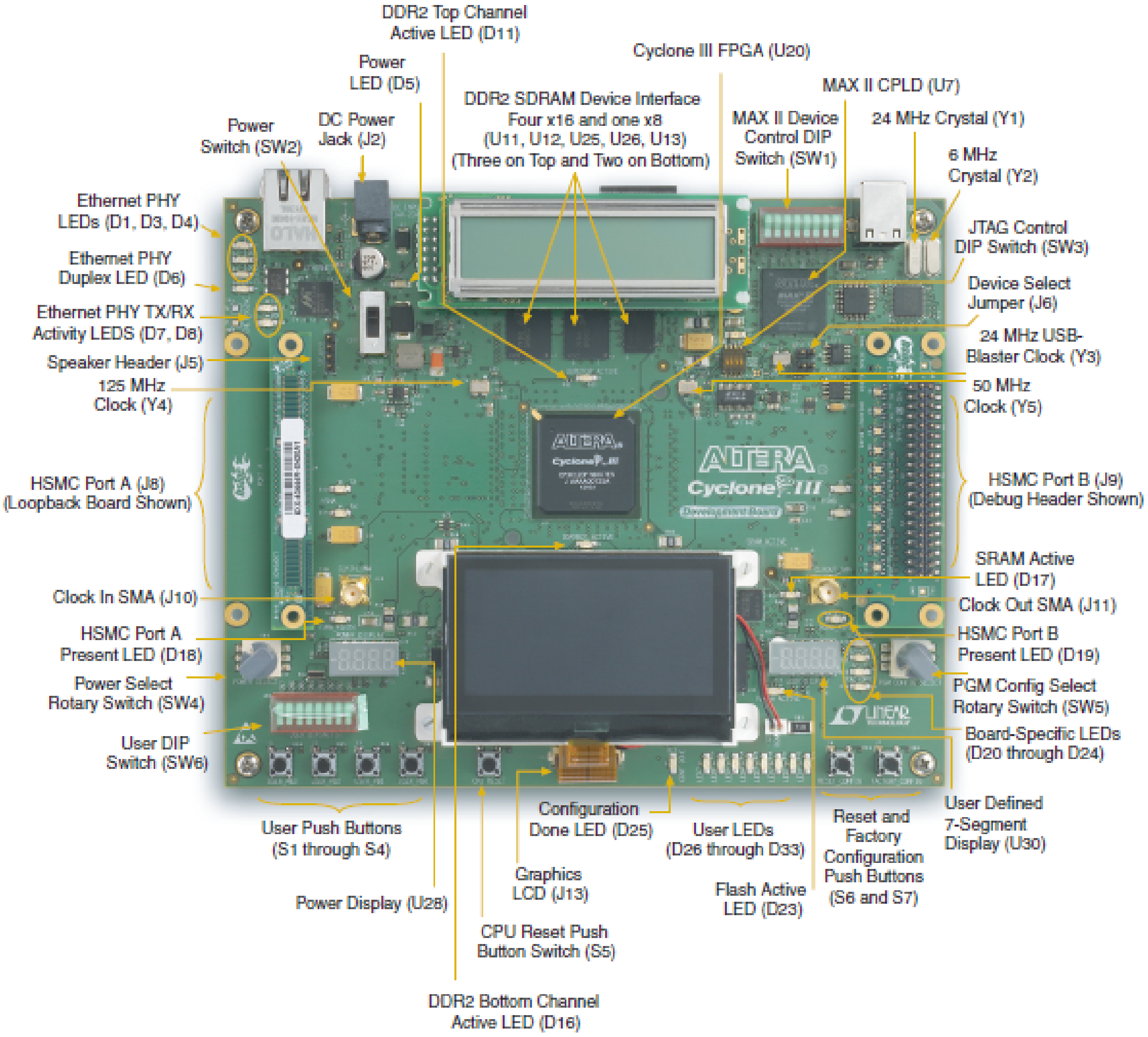

DDR2 Top Channel Active LED (D11)
Cyclone III FPGA (U20)
Power LED (D5)
MAX II CPLD (U7)
DDR2 SDRAM Device Interface Four x16 and one x8 (U11, U12, U25, U26, U13) (Three on Top and Two on Bottom)
MAX II Device Control DIP Switch (SW1)
24 MHz Crystal (Y1)
Power Switch (SW2)
DC Power Jack (J2)
6 MHz Crystal (Y2)
Ethernet PHY LEDs (D1, D3, D4)
JTAG Control DIP Switch (SW3)
Ethernet PHY Duplex LED (D6)
Device Select Jumper (J6)
Ethernet PHY TX/RX Activity LEDS (D7, D8)
24 MHz USB-Blaster Clock (Y3)
Speaker Header (J5)
125 MHz Clock (Y4)
50 MHz Clock (Y5)
HSMC Port A (J8) (Loopback Board Shown)
HSMC Port B (J9) (Debug Header Shown)
SRAM Active LED (D17)
Clock In SMA (J10)
Clock Out SMA (J11)
HSMC Port A Present LED (D18)
HSMC Port B Present LED (D19)
Power Select Rotary Switch (SW4)
PGM Config Select Rotary Switch (SW5)
User DIP Switch (SW6)
Board-Specific LEDs (D20 through D24)
User Push Buttons (S1 through S4)
Configuration Done LED (D25)
User LEDs (D26 through D33)
Reset and Factory Configuration Push Buttons (S6 and S7)
User Defined 7-Segment Display (U30)
Graphics LCD (J13)
Flash Active LED (D23)
Power Display (U28)
CPU Reset Push Button Switch (S5)
DDR2 Bottom Channel Active LED (D16)

Figure 6 – Picture of FSK Modulator hardware board

The previous figures show the block diagram of the FSK Modulator board and also its physical layout. The EP3C120F780 is the selected FPGA, belonging to the Cyclone III Altera FPGA family. This family is a very good compromise between cost and performances.

The 32 bit SRAM and 16 bit NOR flash give the developer a very good support for low resources system prototype, while the 256 Mbytes of DDR2 RAM are the perfect precondition for an operating system full featured system prototype.

The Ethernet interface allows the development of network based system prototype, while the different graphical interfaces (Graphic, characters and Quad 7 Segments) allows the developer to export the output to the external world in many different ways.

Finally the HMSC ports are the perfect method to expand the capability on the board.

3.2.2 FSK Signal Generator

The **FSK signal generator** has been implemented using an ALTERA board. The signal data are saved in a wave file that may be played using a simple wave player software running on the PC.

The generated signal consists of a Audio FSK modulated signal parameters

Table 1 – A-FSK modulated signal parameters

DIGITAL SIGNAL TO TRANSMIT	
FSK Rate:	Variable between 100 Hz and 50 Hz
FSK MODULATED SIGNAL	
“Space” frequency:	968 or 1937 Hz (f1 in figure 3)
“Mark” frequency:	2062 or 1031 Hz (f2 in figure 3)
Amplitude:	1 Vpp
Analog to digital sampling rate	32kHz, or 16kHz

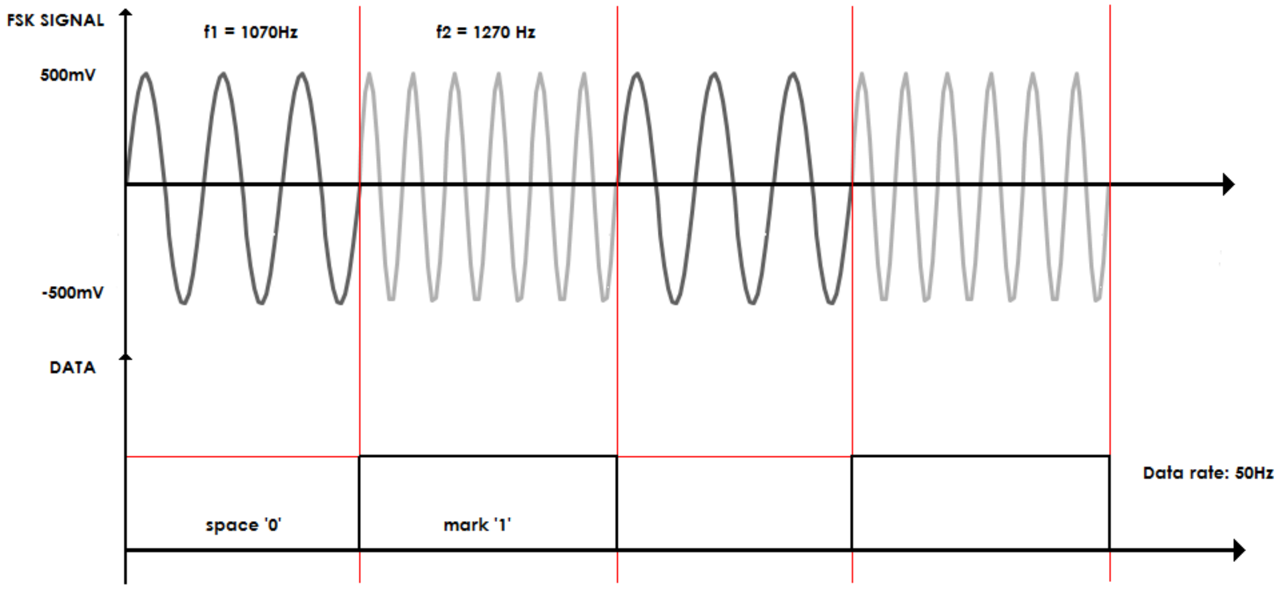


Figure 7 – FSK signal sample

Table 2 – FSK audio signal specification

FSK data sources	
Telnet console characters	8 bit; ASCII coded characters
Proximity sensor sampled data	8 bit; proximity signal level 8 bit quantized
General data file	8 bit characters ASCII coded file

3.2.3 FSK Demodulator

FSK Demodulator SPD Node receives the data samples sent by Signal Generator, and performs a digital demodulation of the samples. After that, it analyses the characteristics of the sampled signal, in order to check if it is compliant with the expected characteristics. If it is so, the FSK Demodulator SPD Node decrypts the signal and provides the sample to the Control Center, through a web page accessed from the pSHIELD Network, using the Ethernet communication interface. If an error occurs, it generates metrics that highlight the discovered problem.

During normal operation, the system sends continuously **metrics data** to the pSHIELD Network. They contain information about the health status of each internal module (HW/SW) of the system. This information will be processed by the pSHIELD network which is responsible to decide what is the action to be done (reconfigure/recovery) based on the obtained results.

3.2.4 Fault Injection Trigger

The Fault Injection Trigger is a mechanism that performs a change of the processing algorithm parameters of the Demodulator block. It is generated with a very simple trigger event (pushing a button). The scope of this block is to inject a fault into Demodulator process, so that it becomes necessary to recovery it.

3.2.5 pSHIELD Control Center

A remote PC, connected to the network via Ethernet is used as pSHIELD Control Center. A web browser running on the PC allows a remote user to:

- receive and store the data samples sent by FSK Demodulator,
- receive and analyze the metrics of the system,
- send the commands (reconfigure/recover) to the system.

3.3 FPGA-Based System Architecture

The FSK Demodulator is a proof of concept to demonstrate the SPD paradigm implemented in an.

The following Figure presents the blocks that have been implemented in this demonstrator.

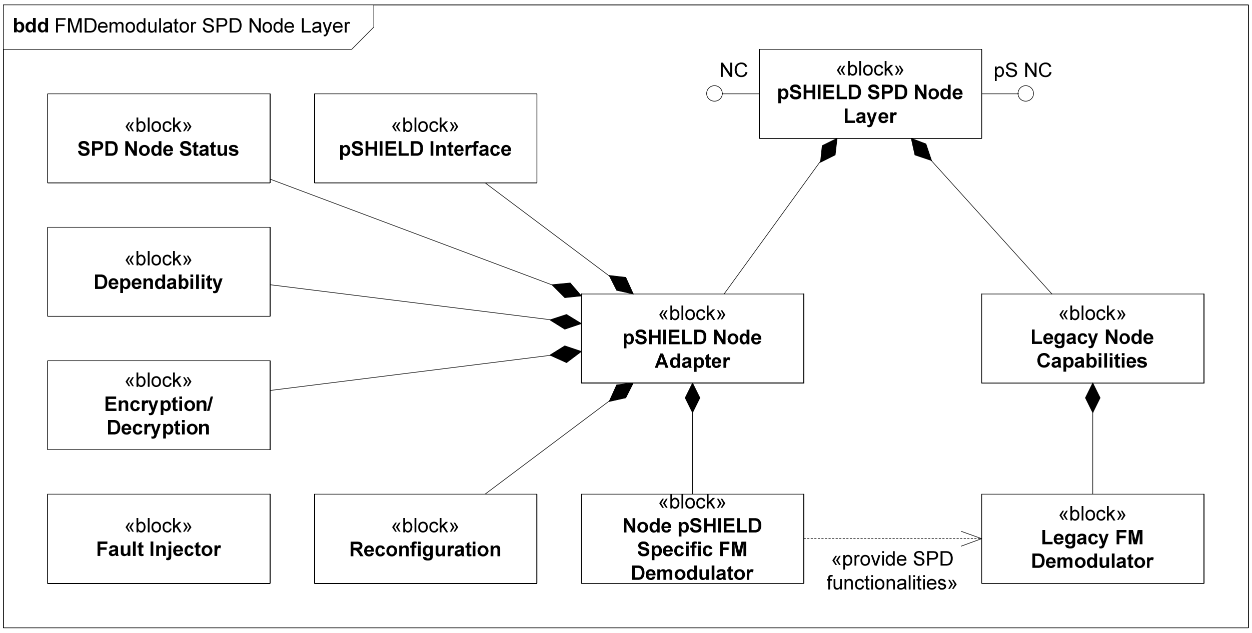

bdd FMDemodulator SPD Node Layer
NC
«block» pSHIELD SPD Node Layer
pS NC
«block» SPD Node Status
«block» pSHIELD Interface
«block» Dependability
«block» pSHIELD Node Adapter
«block» Legacy Node Capabilities
«block» Encryption/ Decryption
«block» Fault Injector
«block» Reconfiguration
«block» Node pSHIELD Specific FM Demodulator
«provide SPD functionalities»
«block» Legacy FM Demodulator

Figure 8 – FSK Demodulator SPD Node Layer

The following figure shows the block diagram of the hardware implementation of the FSK Demodulator SPD Node.

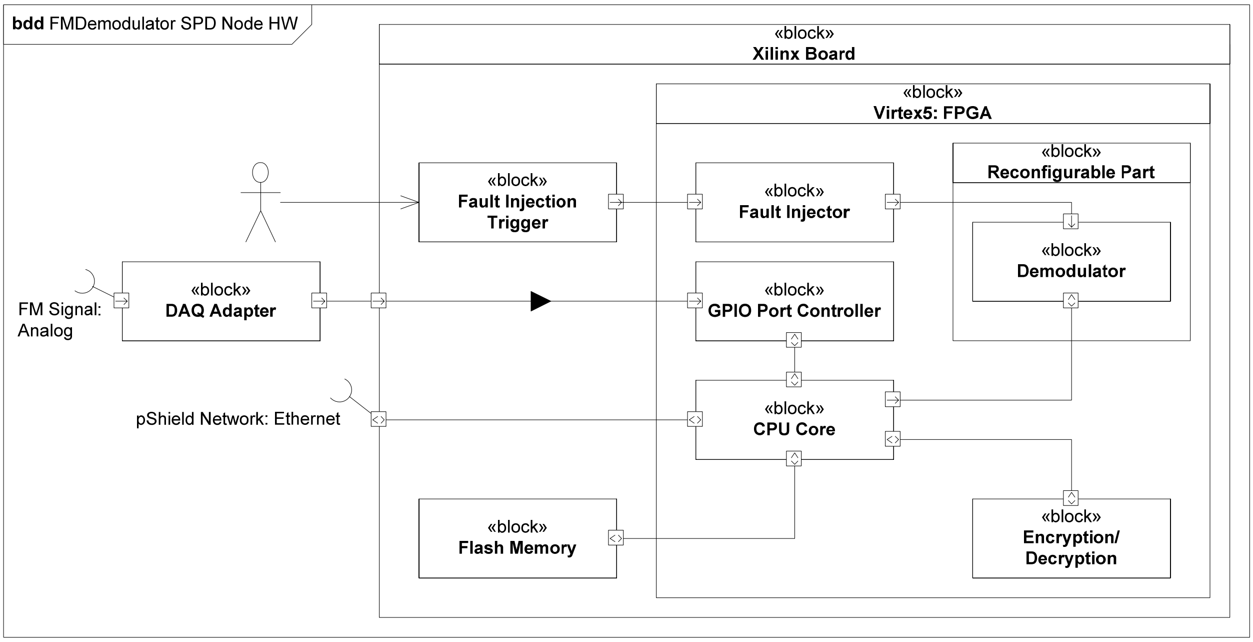

bdd FMDemodulator SPD Node HW
«block» Xilinx Board
«block» Virtex5: FPGA
«block» Reconfigurable Part
«block» Fault Injection Trigger
«block» Fault Injector
«block» Demodulator
FM Signal: Analog
«block» DAQ Adapter
«block» GPIO Port Controller
pShield Network: Ethernet
«block» CPU Core
«block» Flash Memory
«block» Encryption/ Decryption

Figure 9 – A-FSK Demodulator hardware architecture

For our purposes, it has been used a **Xilinx ML507 Evaluation Board,** technical specification is given below.

Features:

- Xilinx Virtex-5 FPGA
    - XC5VFX70T-1FFG1136 (ML507)
- Two Xilinx XCF32P Platform Flash PROMs (32 Mb each) for storing large device configurations

- Xilinx System ACE™ CompactFlash configuration controller with Type I CompactFlash connector
- Xilinx XC95144XL CPLD for glue logic
- 64-bit wide, 256-MB DDR2 small outline DIMM (SODIMM), compatible with EDK supported IP and software drivers
- Clocking
  - Programmable system clock generator chip
  - One open 3.3V clock oscillator socket
  - External clocking via SMAs (two differential pairs)
- General purpose DIP switches (8), LEDs (8), pushbuttons, and rotary encoder
- Expansion header with 32 single-ended I/O, 16 LVDS-capable differential pairs,

14 spare I/Os shared with buttons and LEDs, power, JTAG chain expansion capability, and IIC bus expansion

- Stereo AC97 audio codec with line-in, line-out, 50-mW headphone, microphone-in jacks, SPDIF digital audio jacks, and piezo audio transducer
- RS-232 serial port, DB9 and header for second serial port
- 16-character x 2-line LCD display
- One 8-Kb IIC EEPROM and other IIC capable devices
- PS/2 mouse and keyboard connectors
- Video input/output
  - Video input (VGA)
  - Video output DVI connector (VGA supported with included adapter)
- ZBT synchronous SRAM, 9 Mb on 32-bit data bus with four parity bits
- Intel P30 StrataFlash linear flash chip (32 MB)
- Serial Peripheral Interface (SPI) flash (2 MB)
- 10/100/1000 tri-speed Ethernet PHY transceiver and RJ-45 with support for MII, GMII, RGMII, and SGMII Ethernet PHY interfaces
- USB interface chip with host and peripheral ports
- Rechargeable lithium battery to hold FPGA encryption keys
- JTAG configuration port for use with Parallel Cable III, Parallel Cable IV, or Platform USB download cable
- Onboard power supplies for all necessary voltages
- Temperature and voltage monitoring chip with fan controller
- 5V @ 6A AC adapter
- Power indicator LED
- MII, GMII, RGMII, and SGMII Ethernet PHY Interfaces
- GTP/GTX: SFP (1000Base-X)
- GTP/GTX: SMA (RX and TX Differential Pairs)
- GTP/GTX: SGMII
- GTP/GTX: PCI Express® (PCIe™) edge connector (x1 Endpoint)
- GTP/GTX: SATA (dual host connections) with loopback cable
- GTP/GTX: Clock synthesis ICs
- Mictor trace port
- BDM debug port
- Soft touch port
- System monitor

Top view of the ML507 printed board assembly is shown in the figure below.

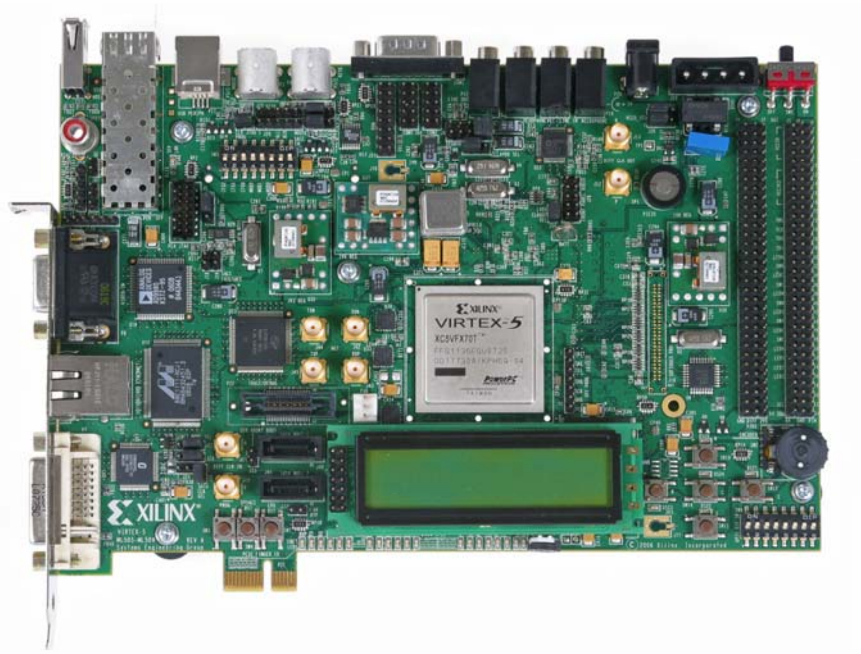


Figure 10 – ML507 printed board assembly (top view)

Block Diagram of Xilinx ML507 Evaluation Platform used for the demonstrator is shown in the next figure.

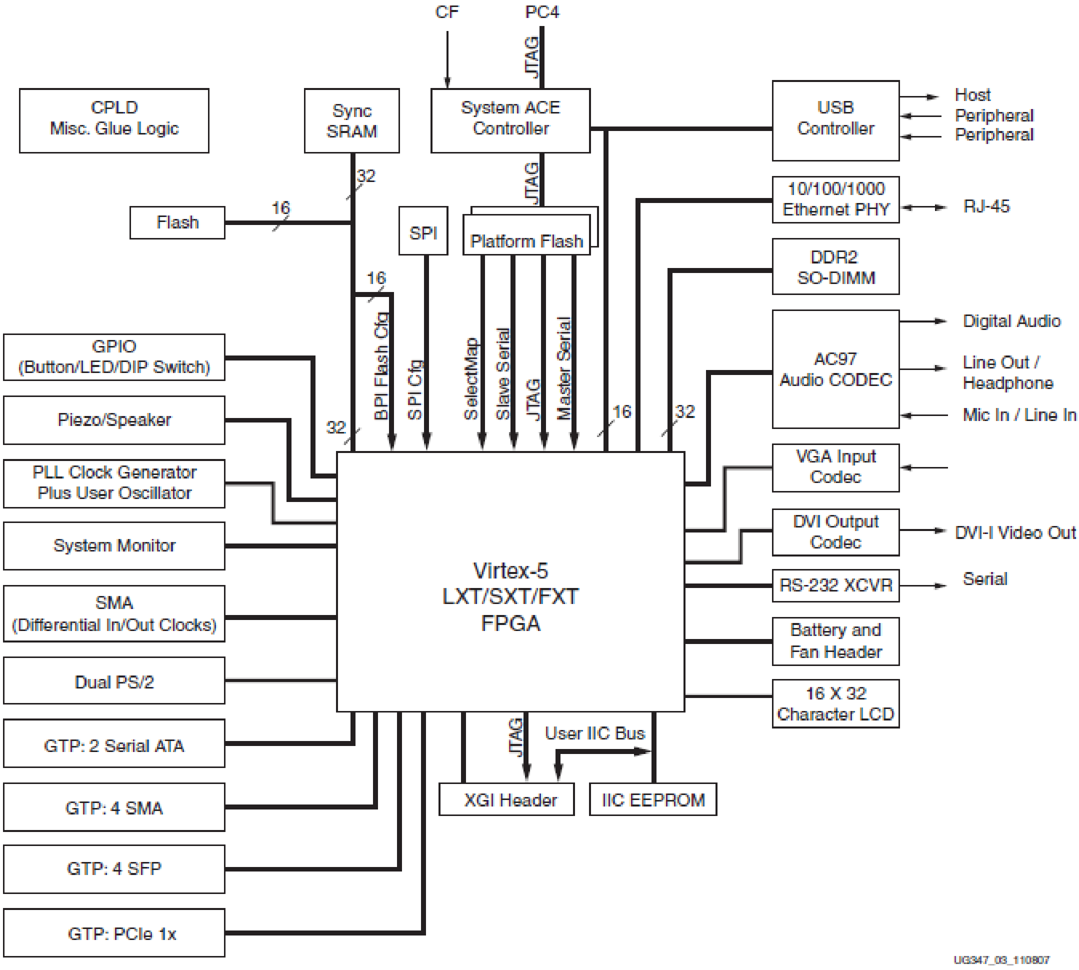


Figure 11 – Block Diagram of Xilinx ML507 Evaluation Platform

3.3.1 Interfacing evaluation board

To interface the FM-Modulator and Demodulator a very simple parallel connection cable is required. It is responsible to:

- Connect the 8 bit wide modulated digital signal;
- Connect the sampling clock being the synchronous clock source to feed the demodulator with digital modulated data samples;

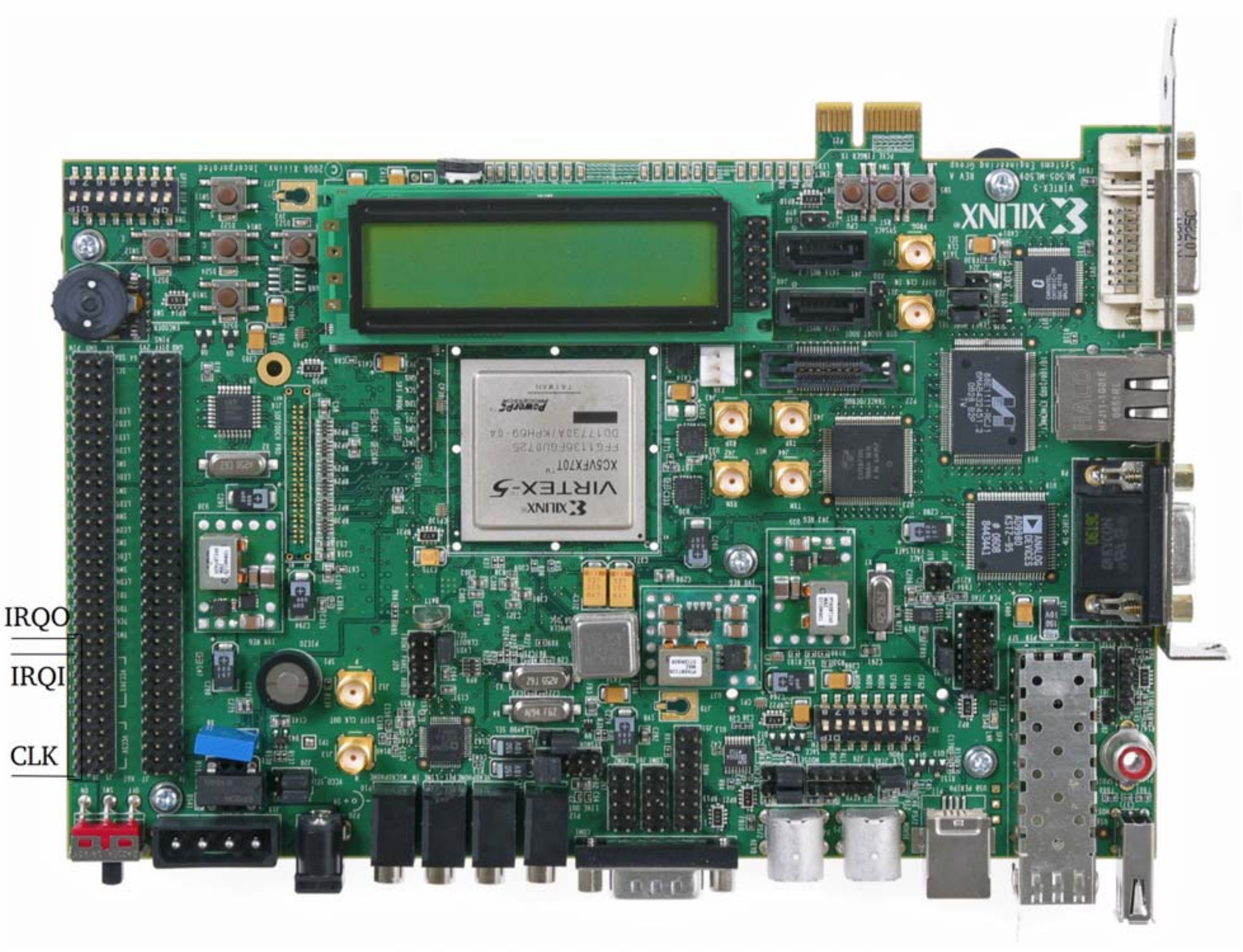


Figure 12 – The evaluation board

The next table shows the pin connection between the DAQ Adapter and ML507 Xilinx board general purpose interface:

Table 3 – Evaluation board pin connection list

Xilinx ML507	Net Name
J6.2	CLK
J6.4	D0
J6.6	D1
J6.8	D2
J6.10	D3
J6.12	D4
J6.14	D5
J6.16	D6
J6.18	D7
J6.20	IRQ IN
J6.22	IRQ OUT

3.3.2 Demodulator

The demodulator implementation is presented in the following block diagram.

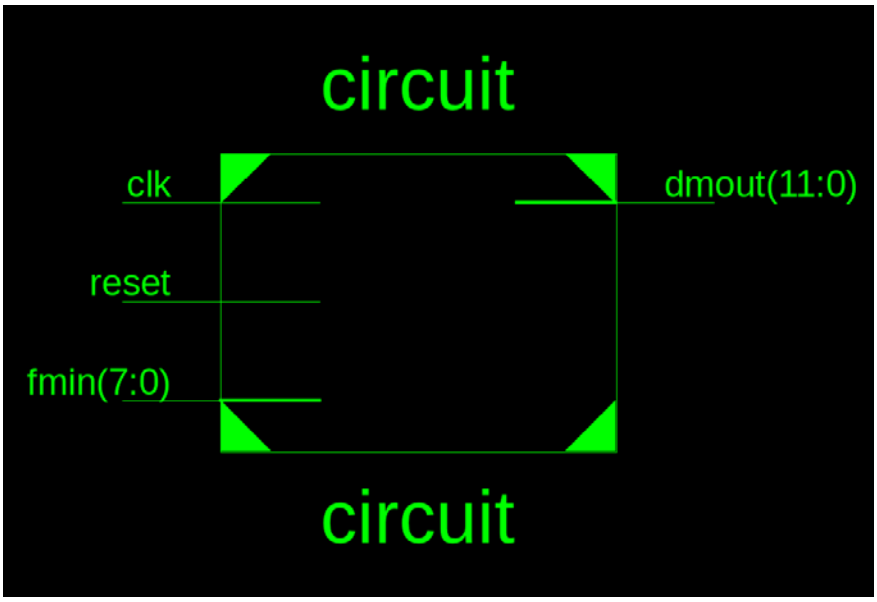


Figure 13 – Demodulator implementation block diagram

The fmin is the byte representing the input modulated sample; the clk is the necessary clock signal to process the incoming modulated byte stream. The dmout word (only 12 bits meaningful) is the demodulated base band signal samples.
A more detailed description of the demodulator is the given by the following diagram.

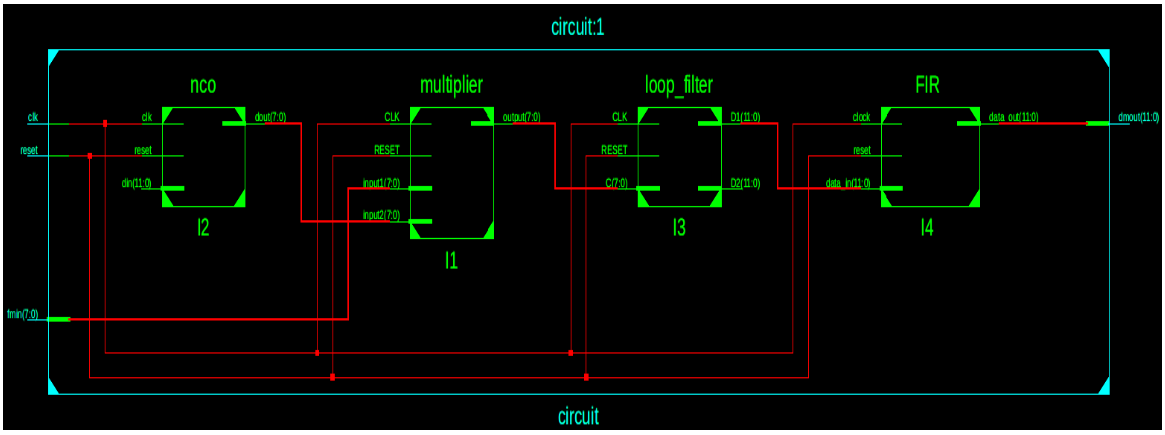


Figure 14 – Detailed demodulator diagram

The fmin byte stream is supposed to be fed by the FM-Modulator data output and clock while the fmout word is supposed to fed a FIFO out, being both the FIFO not properly parts of the demodulator.

3.3.3 CPU Core

The CPU core is a power PC 440. It has the following features:

- PowerPC® 440x5 dual-issue, superscalar 32-bit Documentation,
- 32KB instruction cache, 32KB data cache Example Design,
- Memory Management Unit (MMU),
- Crossbar interconnect with 9 inputs and 2 outputs (128 bits wide), implemented in hardware,

- 128-bit Processor Local Bus (PLB) version 4.6,
- High-speed memory controller interface,
- Auxiliary Processor Unit (APU) controller and interfaces interface for connecting FPU or custom coprocessor.

The cache, the MMU and the FPU are configurable. SO the user can select to use or not the component. Avoiding to use a component (MMU, FPU, Cache) allows to save resources of the FPGA. This aspect is essential for scalability of such a solution.
We note that in the FPGA contained onto ML507 board (used for the demonstrator) PPC440 is hardwired so only a very light wrapper is required. But all the solution is makeable also with pure softcore.

3.3.4 Fault Injection Trigger

The fault injection trigger is a really simple push button; it is polled by the fault injection application and when asserting it generates a failure of the demodulator.

3.3.5 Fault injector

When the fault injector trigger is asserted the demodulator stops to feed the output FIFO. The event of empty output FIFO is the evidence of a malfunctioning of the demodulator. The fault injector is an artefact put into the SoC to show how the system restores its capability after a fault.

3.3.6 Encryption / Decryption

The encryption / decryption system is implemented at software level and only through the network layer.
The Blowfish algorithm is used in both the parts of the system (modulator and demodulator).

3.3.7 Flash Memory

The Flash memory is implemented using a CF card. It is used to store the status of the system and SPD level of the node.

3.3.8 pSHIELD Network: Ethernet interface

The Ethernet interface is built using a MAC softcore and hardware PHY. The PHY present on the board is M88E1111 by Marvell Semiconductor.
The PHY is 10/100/1000 Mbits ready. So the board is Gigabit ready.

3.3.9 Partial Reconfiguration Interface

The partial reconfiguration feature is provided by the ICAP port. This is a proprietary IP core made available by Xilinx Corporation to manage the partial reconfiguration at device driver level. The ICAP port IP core used is hwicap 6.01

3.4 Software Architecture

The demonstration application has been written in C language without the use of any general purpose operating system. This choice has been done to proof the real scalability available on very basic devices, working inside of an SDP distributed system.

The application is essentially an infinite loop managing:

- a very light IP stack
- a FAT32 file system
- A blow fish encrypting algorithm

- a partial reconfiguration device

The application (Main Loop) can be described as follow:

- IP application
    - o reading the data structure to send
    - o sending data structure to the network
    - o reading data from the stack
    - o filling data structure to use inside the application
- File System application
    - o Building the application files
    - o managing the files for ensuring the correct access
- Encryption application
    - o getting clear data
    - o encrypting clear data
    - o filling data structure to be sent
    - o receive encrypted data
    - o decrypting data
    - o filling private data structure with clear received data
- Reconfiguration application
    - o Monitoring the fault injector
    - o If the fault event occurs the reconfiguration application gets the fresh bitstream form the flash and reload it into the partial reconfigurable area to restore the proper work of the demodulator.

3.5 FPGA Partial Reconfiguration

The FPGA partial reconfiguration is a built-in feature of some new FPGA chips. Xilinx FPGAs are leaders of that technology. In particular the partial reconfiguration allows to divide all the FPGA chip in two parts: the static configured part and the dynamic configurable part; the static configuration part is defined at design time ones for all; on the partial reconfigurable part the static part of the system can load each time it likes a partial bit stream implementing a particular hardware function.

The development method to reach such a result needs a particular work flow. Both the parts (static and reconfigurable) needs to be built from the source codes in the form of net lists.

Synthesizing the system will give as a result a full featured bit stream, file allowing to run the default full system configuration, and several partial bit stream files (one for each the different synthesized configuration of the system).

The static part can use the ICAP hardware resource of the FPGA to load from a memory support the required partial bit stream file and store it into the partial reconfigurable region, setting up such a region to work the specified way.

It is easily evident that the partial reconfiguration feature allows the system to be more dependable. In fact if the designers identify a part of the system absolutely vital for the system, they may put such a part into a reconfigurable region and reprogram it just in case of failure of thus part,

3.6 Legacy Devices Integration

The component delivering the SPD compliance are the microprocessor, the FPGA, the ethernet chip and flash memory.

The microprocessor is hardwired into in FPGA so it ensures robustness and reliability. Furthermore it implements the decryption of the demodulated data.

The FPGA hosts several crucial IPs (the demodulator, the FIFO data buffer, the MAC part of the ethernet equipment) and the native feature of dynamic partial reconfiguration, being it the core of dependability feature of the system.

The flash memory allows to store the dynamic information produced by the system and the several required partial bit streams.

3.7 Fault-Injection

The fault-injector subsystem acts on the demodulator. It masks the incoming clock source to the demodulator. This action freezes the demodulated data sending so the main system recognizes the demodulator is failing and reconfigure it.

This is one of the possible techniques to force demodulator failure and we have chosen it because it is relatively simple to add and remove form the project allowing the designer to pass from a development environment to the production one with the minimum risk of undesirable side effect.

3.8 Prototype demonstration results

The prototype works using a proximity sensor able to detect the eventual proximity of not desired object.

The proximity sensor sends its signal to the modulator node, and the modulator node according to its setting encrypts the information, modulates the encrypted data and transmits across a parallel clocked bus.

The demodulator demodulates the data, decrypts the information and calculates the SPD parameters of the service delivered. Furthermore it makes available the proximity information through a web interface.

The FPGA Power Node Prototype may be remotely monitored and controlled using web interface.

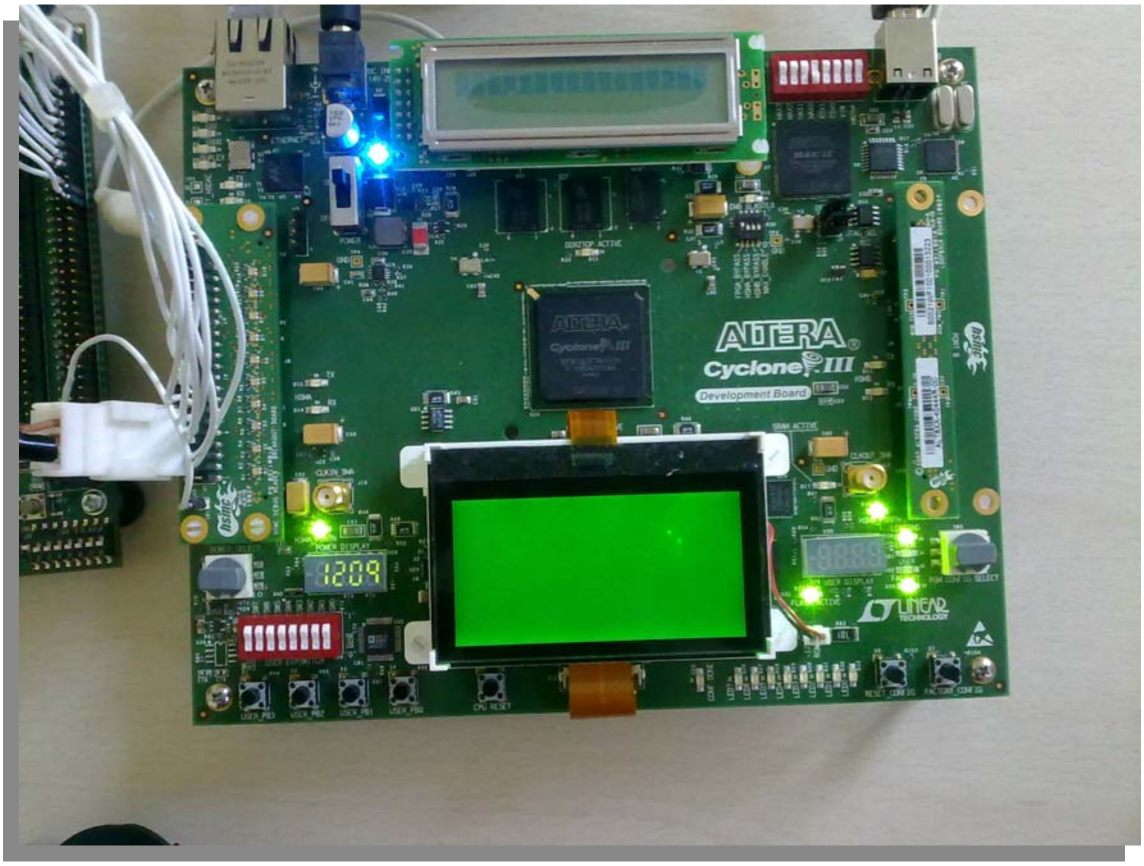


Figure 15 – The FSK Modulator and the connecting bus

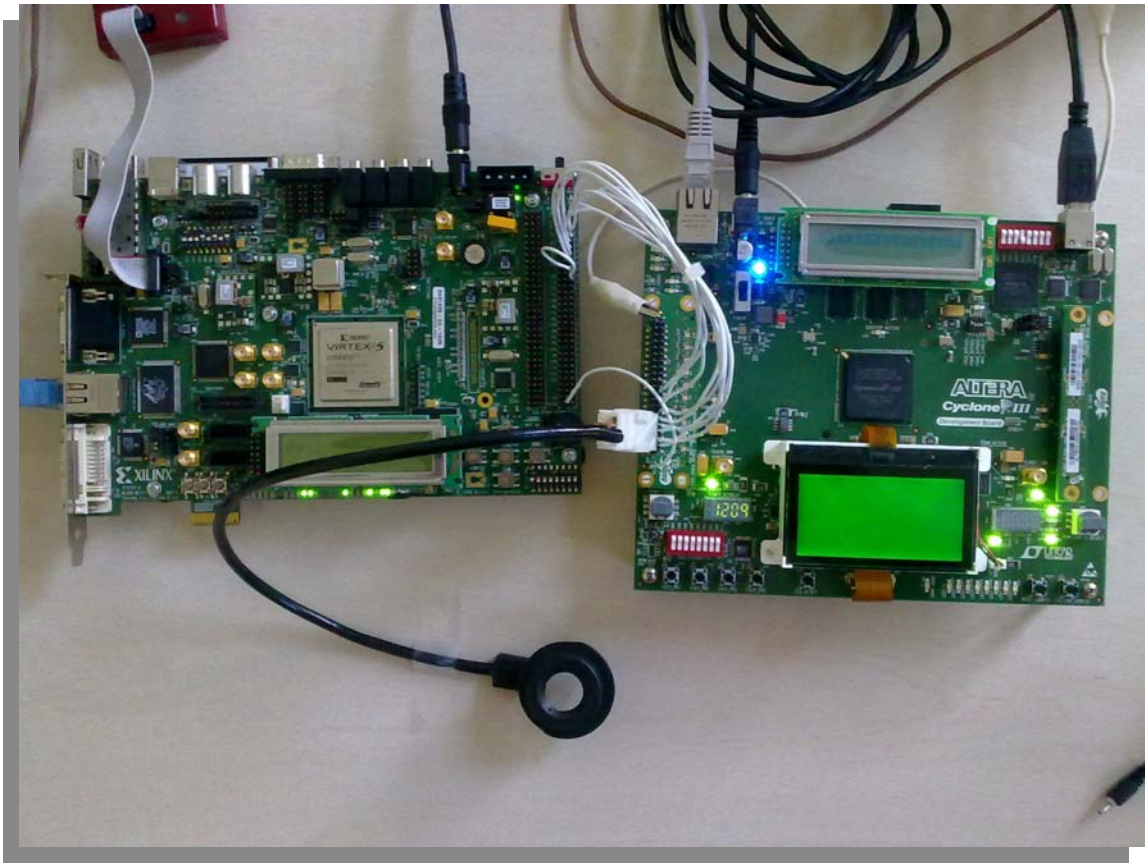
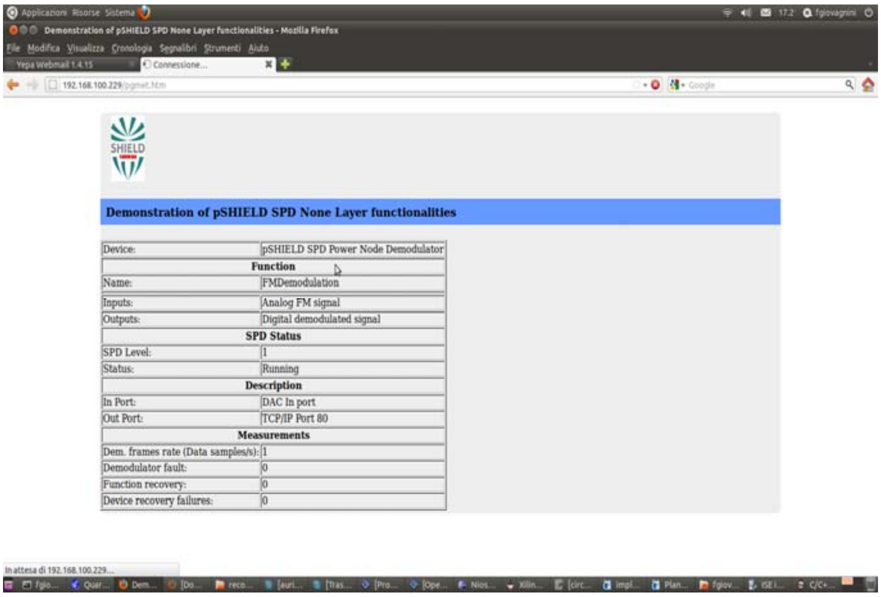


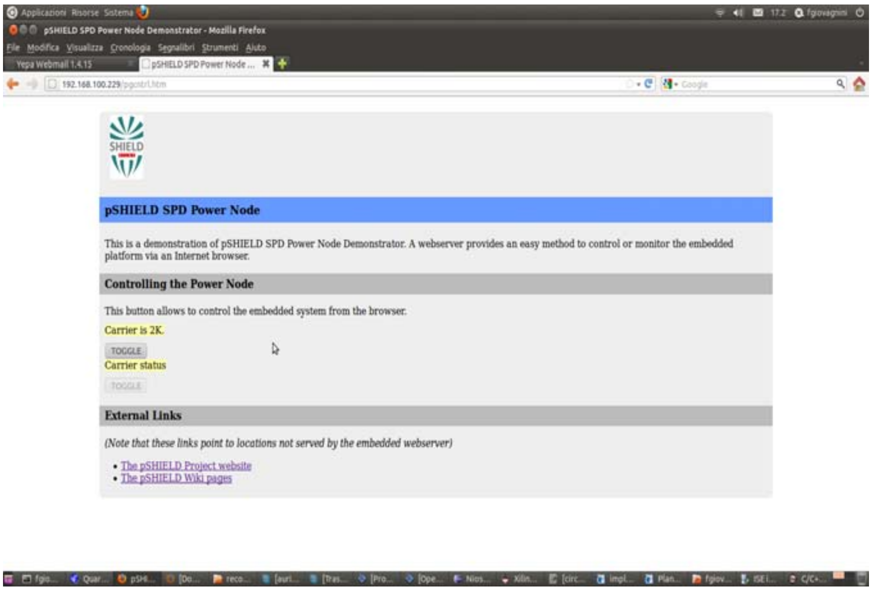
Figure 16 – All the system; the FSK Modulator, the FSK Demodulator and one of the possible proximity sensors


Applicazioni Risorse Sistema
17.2 fgiovagnini
Demonstration of pSHIELD SPD None Layer functionalities - Mozilla Firefox
File Modifica Visualizza Cronologia Segnalibri Strumenti Aiuto
Yepa Webmail 1.4.15
Connessione...
192.168.100.229/pgmet.htm
Google
SHIELD
In attesa di 192.168.100.229...
fgio... Quar... Dem... [Do... reco... [auri... [Tras... [Pro... [Ope... Nios... Xilin... [circ... impl... Plan... fgiov... ISE i... C/C+...

Demonstration of pSHIELD SPD None Layer functionalities

Device:	pSHIELD SPD Power Node Demodulator
Function	
Name:	FMDemodulation
Inputs:	Analog FM signal
Outputs:	Digital demodulated signal
SPD Status	
SPD Level:	1
Status:	Running
Description	
In Port:	DAC In port
Out Port:	TCP/IP Port 80
Measurements	
Dem. frames rate (Data samples/s):	1
Demodulator fault:	0
Function recovery:	0
Device recovery failures:	0

Figure 17 – Control Center web interface - status


Applicazioni Risorse Sistema
pSHIELD SPD Power Node Demonstrator - Mozilla Firefox
File Modifica Visualizza Cronologia Segnalibri Strumenti Aiuto
Yepa Webmail 1.4.15
pSHIELD SPD Power Node ...
192.168.100.229/pgcntrl.htm
Google
SHIELD
TOGGLE
TOGGLE

pSHIELD SPD Power Node

This is a demonstration of pSHIELD SPD Power Node Demonstrator. A webserver provides an easy method to control or monitor the embedded platform via an Internet browser.

Controlling the Power Node

This button allows to control the embedded system from the browser.

Carrier is 2K.

Carrier status

External Links

(Note that these links point to locations not served by the embedded webserver)

- The pSHIELD Project website
- The pSHIELD Wiki pages

Figure 18 – Control Center web interface - control

4 Rugged High Performance Computing Node

Power Node is a rugged embedded system, optimally designed in terms of dimensions, weight, power consumption and capable to work in harsh environmental conditions. The reference application context is defence/aerospace, ground mobile and airborne environments, addressing manned and unmanned applications where reliable high performance computing in an embedded “form factor” is required.

The Power Node is based on a powerful computing architecture: a dual Intel Xeon 5570/5680 series (Quad/Six core CPU) motherboard, with up to 24GB of on-board soldered DDR3 memory and a high data retention 256GB SSD drive. A high speed, high density FPGA device is also present, providing easy adaptability and implementation of dedicated functions and special algorithms. It offer a maximum processing power of 166GFlops.

In the following images the concept of the Power Node is described. The first image illustrates the form factor of the Power Node board and the positioning of the components on the board itself. The second image represents the board covered by a cold-plate that can be air or liquid cooled.

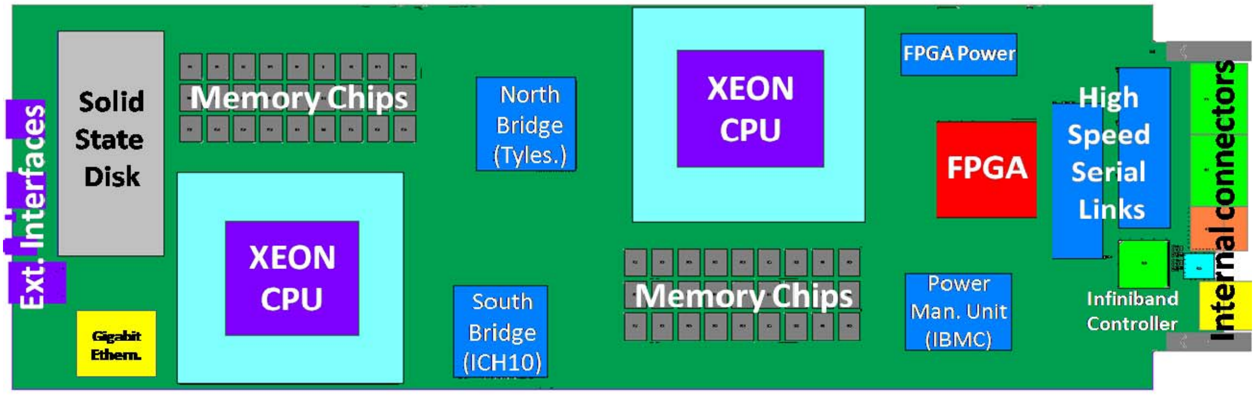


Figure 19 – Power Node board concept, without cooling heat sinks

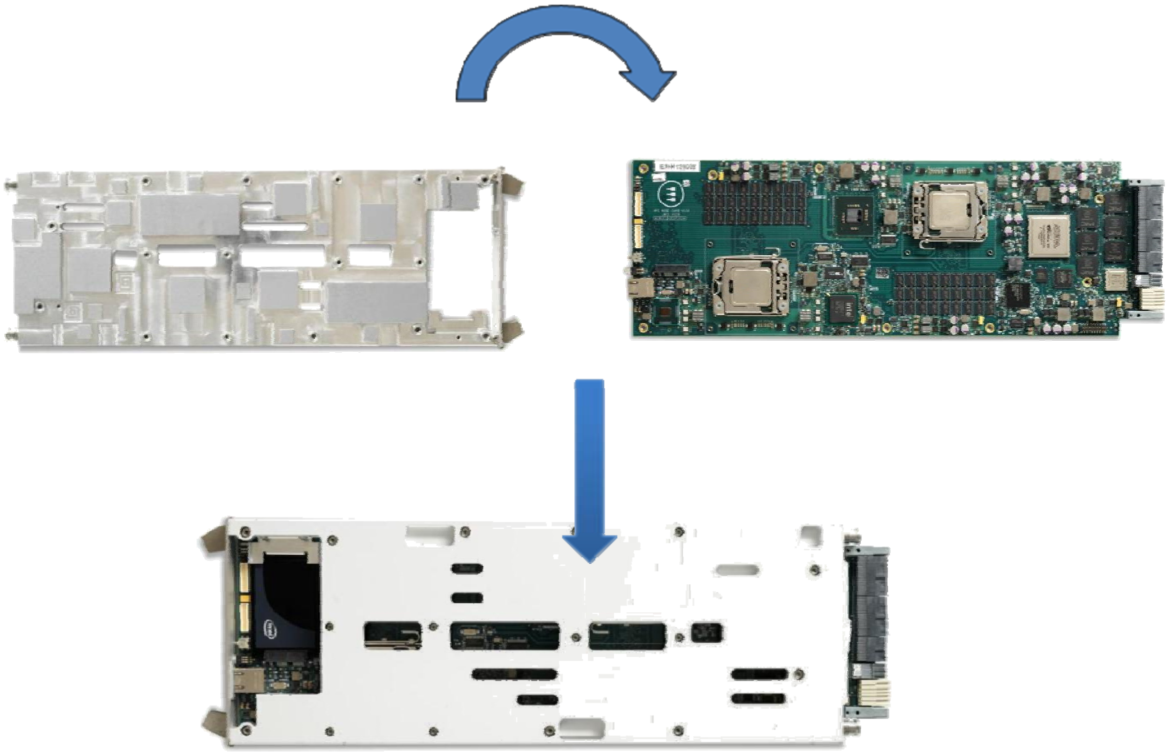


Figure 20 – Power Node board, with cooling cold plate

Figure 21 – Power Node board enclosed inside the rugged chassis (1U) connected the drycooler (1U)

4.1 Power Node software (OS, Protocol stack, Interfaces)

The software development for the Power Node has been mainly devoted to the adaptation of a commonly available Linux Distribution, in order to benefit from the richness of the features of a widely adopted operating system.

Regarding the OS the first choice has been "RedHat Enterprise Linux OS Verison 5.5 x86_64" which needs a license but is very well supported. Alternatively, if an open-source Linux distribution is required, the Power Node can support Linux distribution derived from RedHat, which are available for free but don't have usually an excellent support. In this case, the operating system could be one of the following:

- CentOS,
- Scientific Linux.

In addition to the OS, the porting of device drivers for the Infiniband networking interface and for the IBMC Board Management Controller has been provided.

The design of the FPGA firmware and software is intended to be implemented by the user of Power Node using ALTERA development tools:

- QUARTUS II (http://www.altera.com/products/software/quartus-ii/subscription-edition/qts-se-index.html),
- USB-JTAG programming/debugging tool (http://www.buyaltera.com/scripts/partsearch.dll?Detail&name=544-1775-ND).

As a starting point, many reference design, optimized for the same FPGA used in the Power Node, can be downloaded from Altera website. They reduce time to implement complex interface such as PCIe by means of pre-compiled building block.

To develop end-user applications, the final software development kit will contain the following additional tools:

- Infiniband OFED driver Stack supplied by Mellanox (basically standard OFED stack 1.5.1 pre-compiled). The package contains drivers and libraries for the InfiniBand interface and for the 10Gb Ethernet interface (http://www.mellanox.com/content/pages.php?pg=products_dyn&product_family=26&menu_section=34#tab-three).
- IPMI tools.
- Scientific Computation Libraries from EPEL Repository (they need separate free licensing).
- Intel C/C++ and Fortran Compilers.
- Intel Math Kernel Libraries (All the Mathematic primitives: FFT, Matrix calculations etc).
- Intel Integrated Performance Primitives (these are basically computational accelerators).
- Other Intel Libraries (these are proprietary libraries for example: treading building block).
- The power node supports **compilers** like:
  - Intel Cluster Studio,
  - GNU toolchain,
  - Portland CDK.
- The power node supports **debugger and libraries** like:
  - Totalview,
  - DDT,
  - Intel Trace Analyzer and Collector,
  - Intel VTune,
  - Math Libraries (compatibility of Math libraries is implementation specific):

- Intel MKL.
- IMSL (with ICT requires adaptation).
- NAG.

Finally, from an application point of view, it is the final customer that defines the usage of the system and dictates the applications to be installed. The application types range is extremely wide, going from scientific open source software to commercial off the shelf suites. The design of the Power Node tries to guarantee the highest flexibility of usage possible and the customer may identify some specific utilizations that require a specific software configuration to be installed.

4.2 Power Node hardware (Power, CPU, Interfaces, Sensing, extras)

The Power Node is a High Performance Platform based on Nehalem/Westmare Xeon Intel dual-processor board with Tylesburg chipset; it is equipped with a high density FPGA and a high speed Infiniband controller, moreover there is an Ethernet Gigabit interface. Every component is supervised by a Power Management Controller Unit (IBMC).

The Power Node core architecture consists of two Intel Xeon X5680 or X5570 CPUs, connected via Quick Path Interconnect (QPI), a dedicated low latency and high bandwidth bus capable of up to 6.4GT/s. Three channels of DDR3 memory are connected to each CPU, which integrates a high performance memory controller. The system hub (I/O Bridge) is an Intel 5520 (Tylersburg) chipset and provides connectivity between the CPUs and the rest of the system; each CPU is connected to its Tylersburg with a QPI link. A Mellanox QDR ConnectX2 adapter is connected to the Tylersburg via one x8 PCIe 2.0 link: it provides a high Infiniband compliant connection. The hardware programmable part of the Power Node is represented by an Altera Stratix IV FPGA, which is connected to the Tylersburg with 2 x8 PCIe 2.0 links.
Finally, the peripheral hub (Intel ICH10) is connected to the Tylersburg and provides the following additional peripherals:

- one optional SATA SSD, used to provide local fast and permanent storage,
- one Zoar Gigabit Ethernet adapter,
- 2x external accessible USB ports,
- one Output Video Port,
- one UART for low level debug.

The independent, embedded controller for the Power Management (IBMC) allows the monitoring of each performance parameters, such as temperatures, voltages, etc.. Access to these parameters can be done by the Power Node applications, locally and remotely over the network. The IBMC provides an SNMP interface to the Power Node and allows setting traps for specific events. It can also trigger and monitor the Power-On-Self-Test. In terms of remote control, the embedded IBMC permits the remote configuration of the Power Node through the network and additional remote configurability can be done through the FPGA. The overall architecture of the Power Node is represented in the next figure.

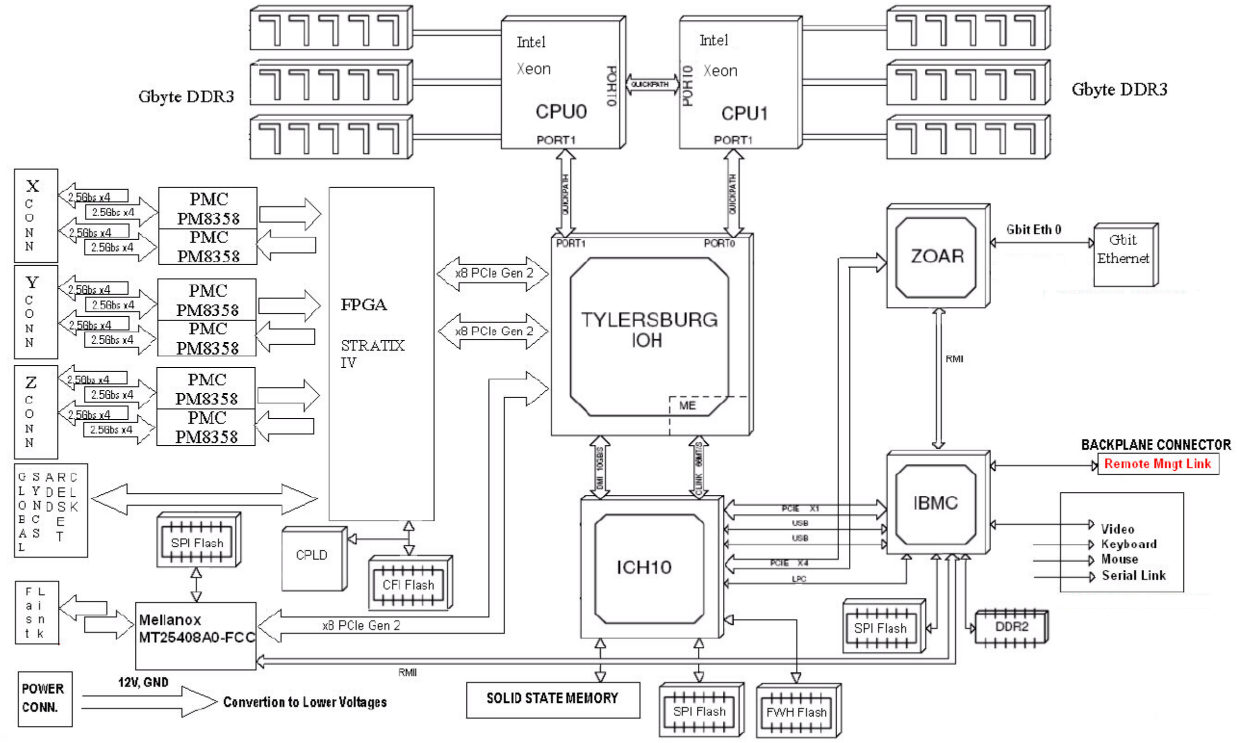


Figure 22 – Power Node architecture: high level description.

The FPGA Processor is responsible for some security aspects. It includes a core logic that monitors the security of the Power Node. Tampering with the node triggers a protection mechanism in the security node that:

- physically disconnects any I/O and network,
- deletes any data resident on the node,
- initiates the physical destruction of the device itself by driving the power supply,
- provides security features such as cryptographic capabilities through a dedicated core embedded in the FPGA,
- and more in general, the hardware supports the Intel AES-NI technology.

The Power Node architecture has been conceived thinking also to “composability”, in order to provide the possibility to build network of Power Nodes depending on the specific requirements of the specific application context. The Infiniband interface allows creating virtual 3D torus networks of Power Nodes, which are very efficient in terms of bandwidth and latency, and are capable of scaling up with no performance penalty. The torus network is managed by a network processor implemented in the FPGA of each Power Node, which interfaces to the system hub through two x8 PCI Express Gen 2 connections, for a total internal bandwidth of 80Gbs. Thanks to this kind of interface the FPGA internal design can operate such as an accelerator for example running real-time cryptographic algorithms to ensure reliable and secure connections.

Alternatively the FPGA can be used to implement a torus network processor which permits standard, ad-hoc and application-dependent collective communications. Finally, the I-O and network interfaces are programmable, in order to permit interfacing the system to multiple network and bus technologies and protocols, increasing in this way the potential scalability of the network.

The possibility to aggregate multiple identical units has an impact also on dependability, providing redundancy. The execution segregation through hardware virtualization allows for protection, monitoring, disabling and replacement of malfunctioning or compromised nodes. Moreover, in case of a fault, redundant hardware provides dependable operations. This is accomplished at the hardware level through duplication of the resource and at a functional level through aggregation of resources (spare Power Nodes).

4.3 Power Node Reconfigurability

The capability of the Power Node to reconfigure itself, at runtime, is offered by the use of “in-system programmable” devices such as an FPGA. This means that according to an environmental request, not only the software libraries can be dynamically loaded, but also the hardware accelerator configuration can be modified at any time. With configuration we intend the hardware logic previously programmed in the FPGA.

As shown in the following image, hardware silicon internal part of the FPGA are based on SRAM Logic Elements which consist of combinational logic attached to memory elements and they can be combined to implement any type of hardware function.

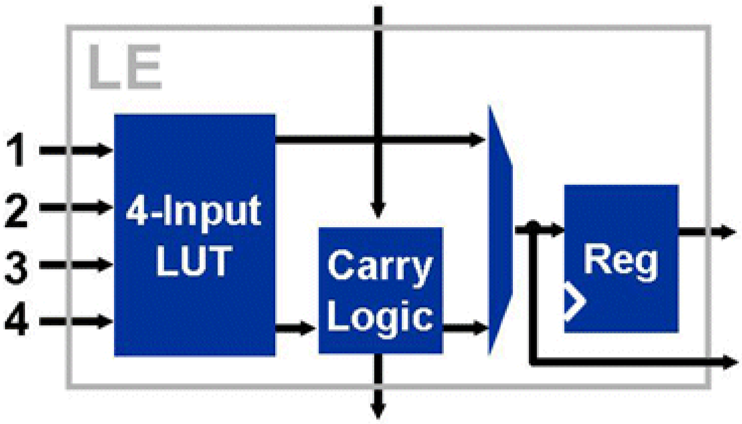


Figure 23 – internal structure of an FPGA logic element.

Complex hardware functionalities can be designed with high level hardware description languages such as VHDL o Verilog or through schematic entry tools provided by development IDE.

Once the design has been completed and synthesized the development tool provide a binary file which can be written to the target device (FPGA) to update the configuration to the newer one.

The standard interface to access configuration registers of the FPGA is the JTAG port and it is used to write on it the binary file produced by the compiler.

The Power Node uses a USB-JTAG converter to grant OS the access to HW reconfiguration. The converter is integrated on the Power Node board. This solution has been adopted on both release of the prototype to simplify and improve the development and debug process. A second solution, that doesn’t require the USB-JTAG converter, could be adopted in future versions of the prototype that will be closer to a final product. The current hardware already allows the implementation of this solution that, in terms of functionalities, is perfectly equivalent to the one adopted. This second solution is based on the direct reconfiguration of the FPGA through the PCi Express bus. A software application is capable to store the FPGA binary images into the Flash memory connected to the FPGA, and chooses the most suitable image depending on the threat identified. In this case, a specific operating system driver must be implemented to control the PCi Express bus and an engine, that acts as a bridge between the bus itself and the flash memory, must be implemented into the FPGA and added to the FPGA application specific logic.

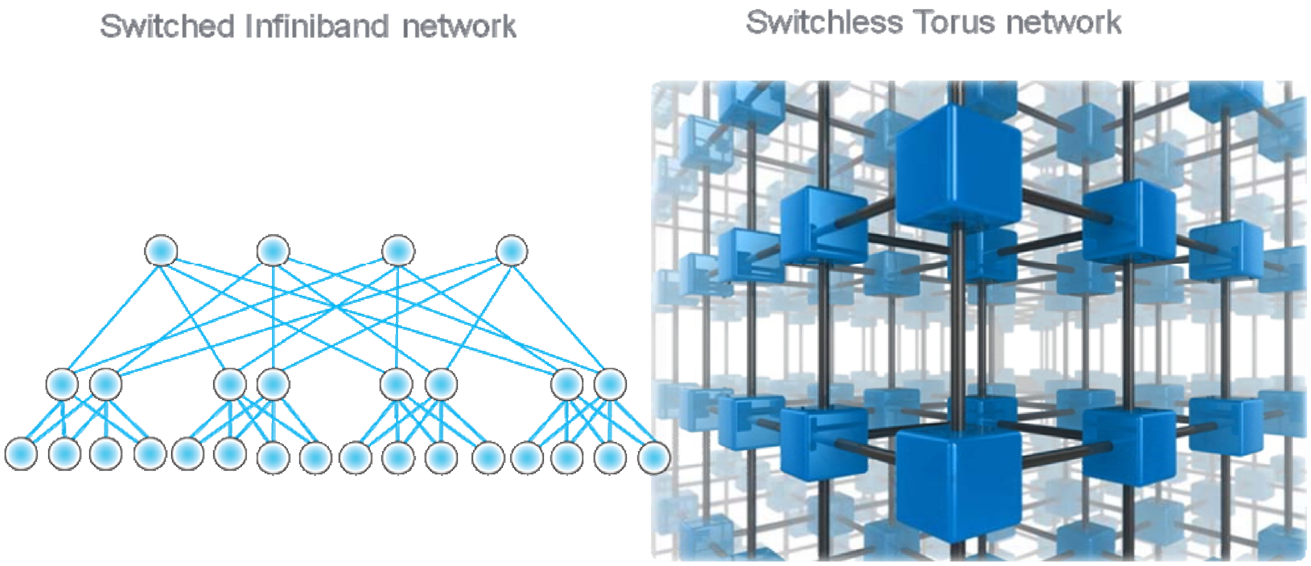
The reconfigurability features offered by the Power Node can be used in a real application scenario as follows:

1. a threat is identified by proper application logic.
2. The application, depending on the threat, decides if a reconfiguration of the FPGA is required.
3. The operating system stops processes that use the current hardware configuration.
4. The application chooses the new configuration capable to face the threat.
5. The selected configuration is written via JTAG to the FPGA.

The operating system starts new processes associated with the new HW configuration.

4.4 Interconnect capability

The power node high performance interconnectivity options are given by network interface which are present on the board, they are briefly summarized by the following pictures where the different topologies and characteristics are described:



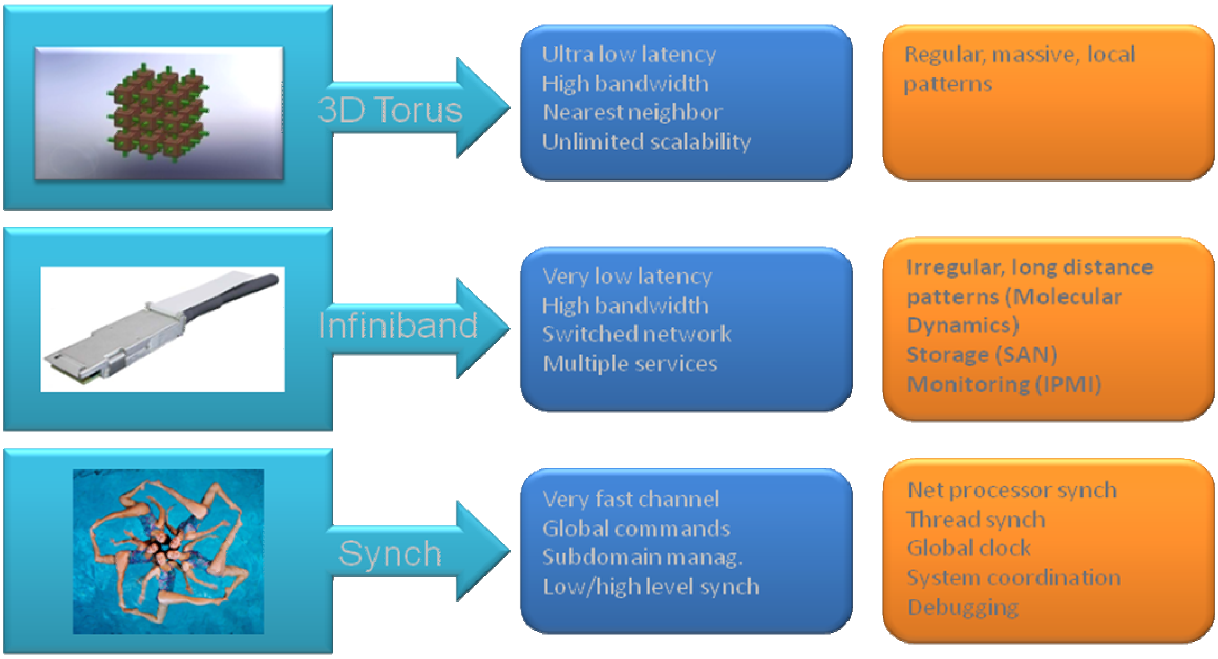


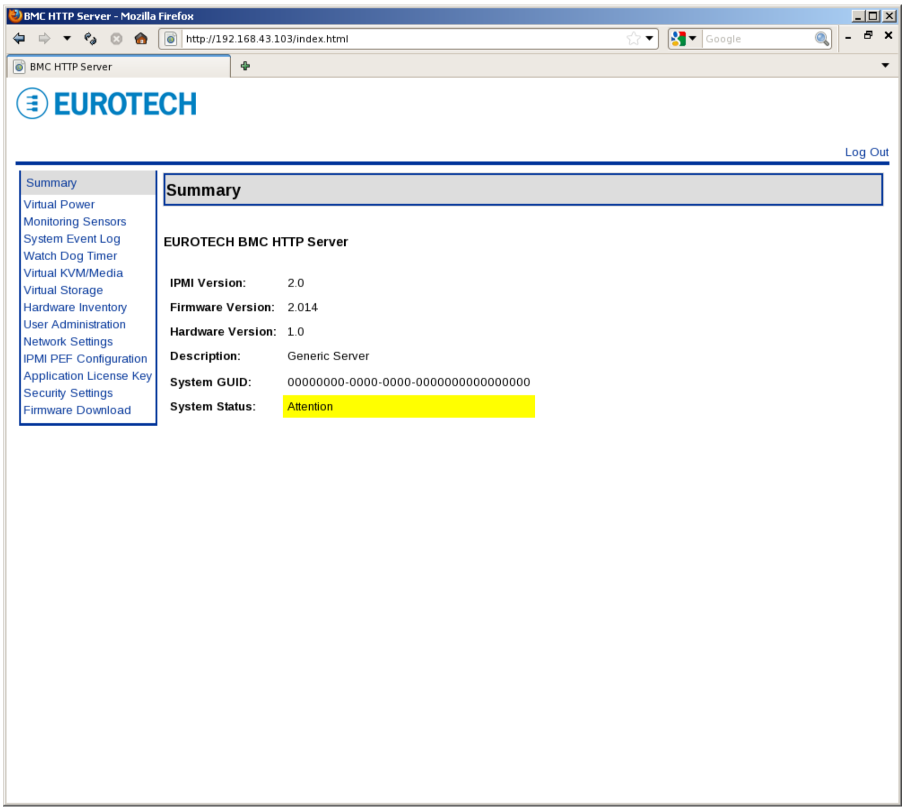
Figure 24 – Interconnectivity capabilities of the Power Node

4.5 Remote management and control

Monitoring and management software is crucial when it comes down to proactive maintenance, problem solving and ultimately availability of the system, for these reasons on the Power Node is present an **IPMI (iBMC based)** management system in order to provide the following services:

- Data logging of integrated sensors,
- Alarm and trigger applications,
- Power control,
- Remote KVM User Interface,
- Bios update and recovery.

All these features are accessible through the standard IPMI connection interface (ethernet, serial, lpc, smbus, kcs etc.).
One of the most powerful interaction channel is provided by the embedded web server interface which is totally independent from the OS running on the system, indeed it is also available when the main CPUs are powered down.
The local console of the Power Node can be accessible remotely through the standard text-only protocol Serial-on-lan (SOL) or via the Graphic interface made available by a Java applet, integrated in the web server, which exports Video Keyboard and Mouse to the remote web browser.

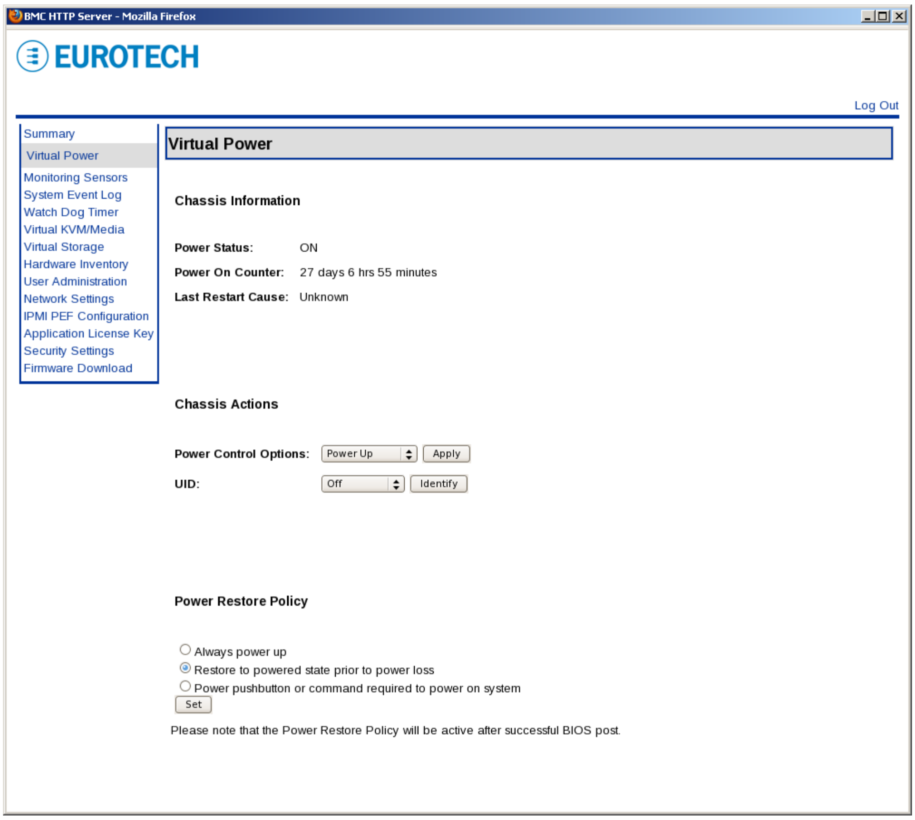


Summary

EUROTECH BMC HTTP Server

IPMI Version:	2.0
Firmware Version:	2.014
Hardware Version:	1.0
Description:	Generic Server
System GUID:	00000000-0000-0000-0000000000000000
System Status:	Attention

Figure 6 – Main page of embedded Web server integrated on the iBMC



Virtual Power

Chassis Information

Power Status:	ON
Power On Counter:	27 days 6 hrs 55 minutes
Last Restart Cause:	Unknown

Chassis Actions

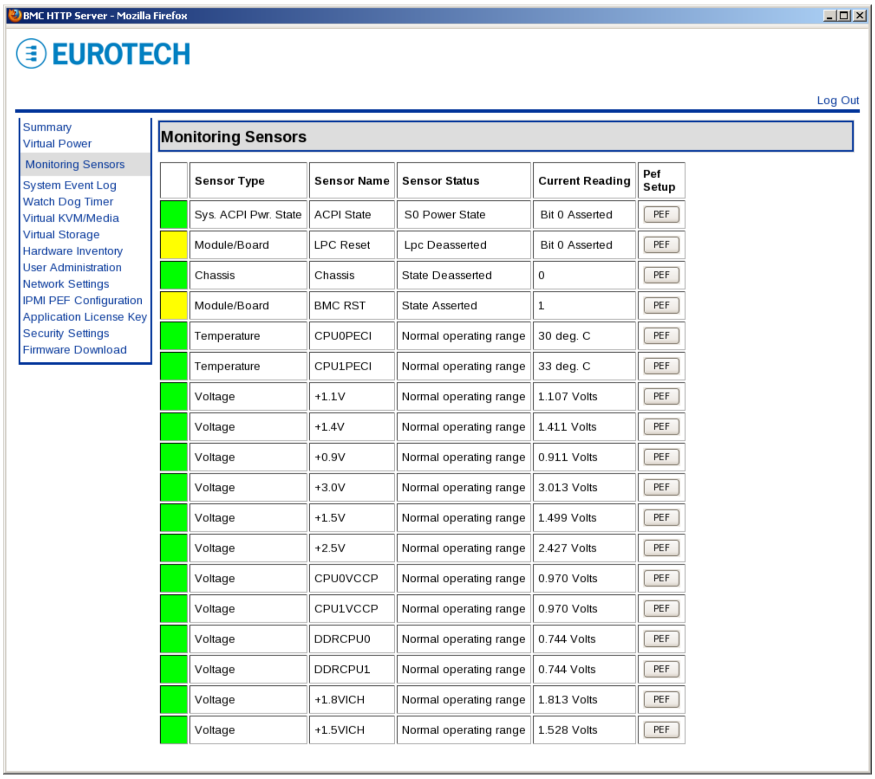
Power Control Options:	Power Up
UID:	Off

Power Restore Policy

○ Always power up
● Restore to powered state prior to power loss
○ Power pushbutton or command required to power on system

Please note that the Power Restore Policy will be active after successful BIOS post.

Figure 7 – Power control and status of the Xeon CPUs



Monitoring Sensors

	Sensor Type	Sensor Name	Sensor Status	Current Reading	Pef Setup
🟩	Sys. ACPI Pwr. State	ACPI State	S0 Power State	Bit 0 Asserted	PEF
🟨	Module/Board	LPC Reset	Lpc Deasserted	Bit 0 Asserted	PEF
🟩	Chassis	Chassis	State Deasserted	0	PEF
🟨	Module/Board	BMC RST	State Asserted	1	PEF
🟩	Temperature	CPU0PECI	Normal operating range	30 deg. C	PEF
🟩	Temperature	CPU1PECI	Normal operating range	33 deg. C	PEF
🟩	Voltage	+1.1V	Normal operating range	1.107 Volts	PEF
🟩	Voltage	+1.4V	Normal operating range	1.411 Volts	PEF
🟩	Voltage	+0.9V	Normal operating range	0.911 Volts	PEF
🟩	Voltage	+3.0V	Normal operating range	3.013 Volts	PEF
🟩	Voltage	+1.5V	Normal operating range	1.499 Volts	PEF
🟩	Voltage	+2.5V	Normal operating range	2.427 Volts	PEF
🟩	Voltage	CPU0VCCP	Normal operating range	0.970 Volts	PEF
🟩	Voltage	CPU1VCCP	Normal operating range	0.970 Volts	PEF
🟩	Voltage	DDRCPU0	Normal operating range	0.744 Volts	PEF
🟩	Voltage	DDRCPU1	Normal operating range	0.744 Volts	PEF
🟩	Voltage	+1.8VICH	Normal operating range	1.813 Volts	PEF
🟩	Voltage	+1.5VICH	Normal operating range	1.528 Volts	PEF

Figure 8 – Monitoring sensors readings (Voltages and Temperatures)

4.6 Cooling System

The intrinsic reliability and safety of the system is mainly due to the fact that there are no spinning discs and no fans for cooling on-board heat generating components. Liquid cooling systems are considered state of the art in removing heat from electronic devices because they permit optimal PUE (Power Usage Effectiveness). This parameter measures how much of the electrical power entering a data center is effectively used for the IT load, that is for making the server work.
PUE is defined as:

$$\mathrm{PUE} = \frac{\mathit{Total\ Facility\ Energy\ Consumption}}{\mathit{IT\ Equipment\ Energy\ Consumption}}$$

Ideal PUE is equal to 1, which is also its theoretical limit: all energy entering the data centre is used to power the IT equipment, so all energy is useful. Average servers have a PUE of 2.13.

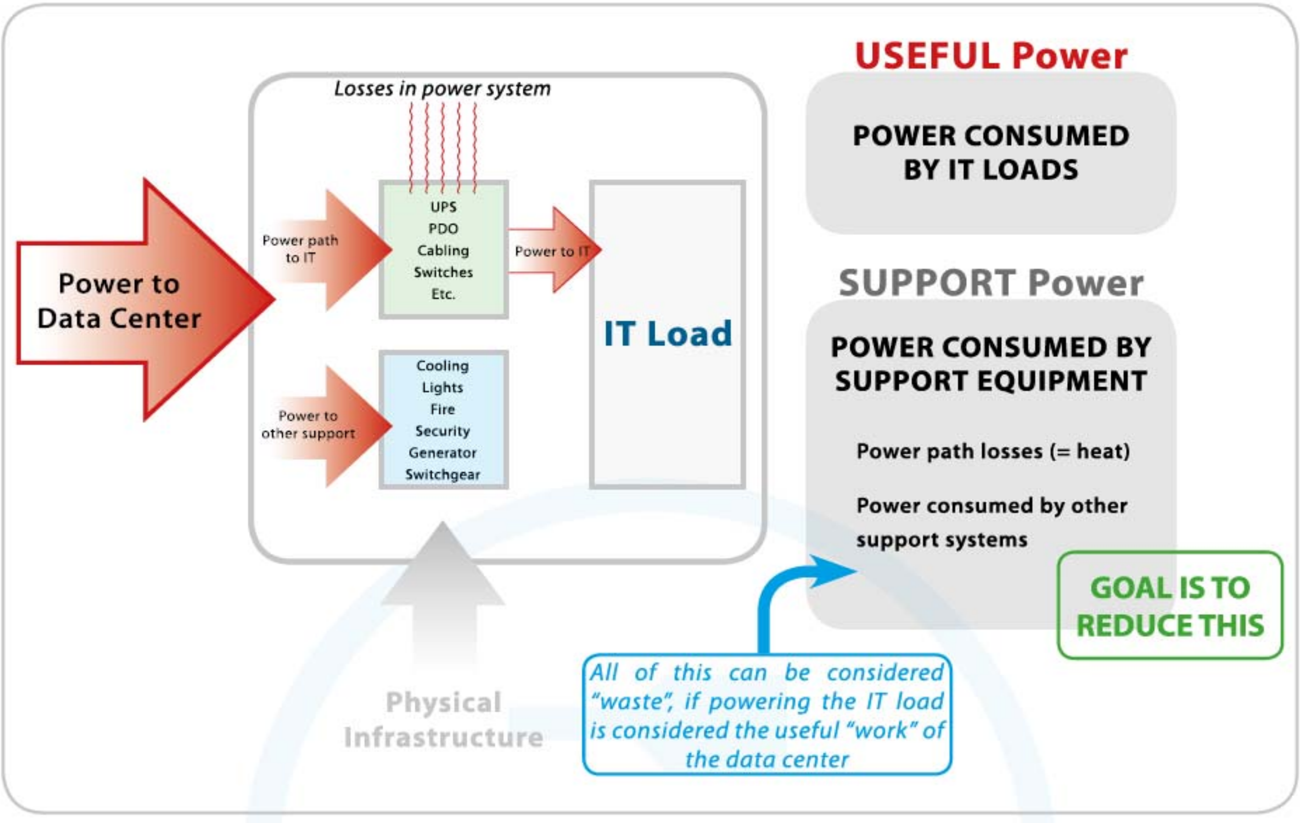

USEFUL Power
POWER CONSUMED BY IT LOADS
Losses in power system
Power to Data Center
Power path to IT
UPS
PDO
Cabling
Switches
Etc.
Power to IT
IT Load
Power to other support
Cooling
Lights
Fire
Security
Generator
Switchgear
SUPPORT Power
POWER CONSUMED BY SUPPORT EQUIPMENT
Power path losses (= heat)
Power consumed by other support systems
GOAL IS TO REDUCE THIS
Physical Infrastructure
All of this can be considered "waste", if powering the IT load is considered the useful "work" of the data center

Figure 9 – The concepts behind the cooling system

With the liquid cooling system used in the implementation of the Power Node the power consumption of the entire system has a better balance compared to e traditional one and it reaches a PUE of 1.5 with the following operative conditions:

- Operating values:
  - Coolant flow: 80-90 lph
  - Pressure: 1-3 PSI
  - T(outlet) – T(inlet) = 3 °C

In the following picture is represented the plot of temperature over time during a repetitive HPL Linpack performance test. From the graph can be seen how peak power doesn’t influence so much the temperature of the coolant, this means that aluminium mass, of the cold plate, acts as a dumper for the heat transfer and this behavior simplifies the design of the external cooler.

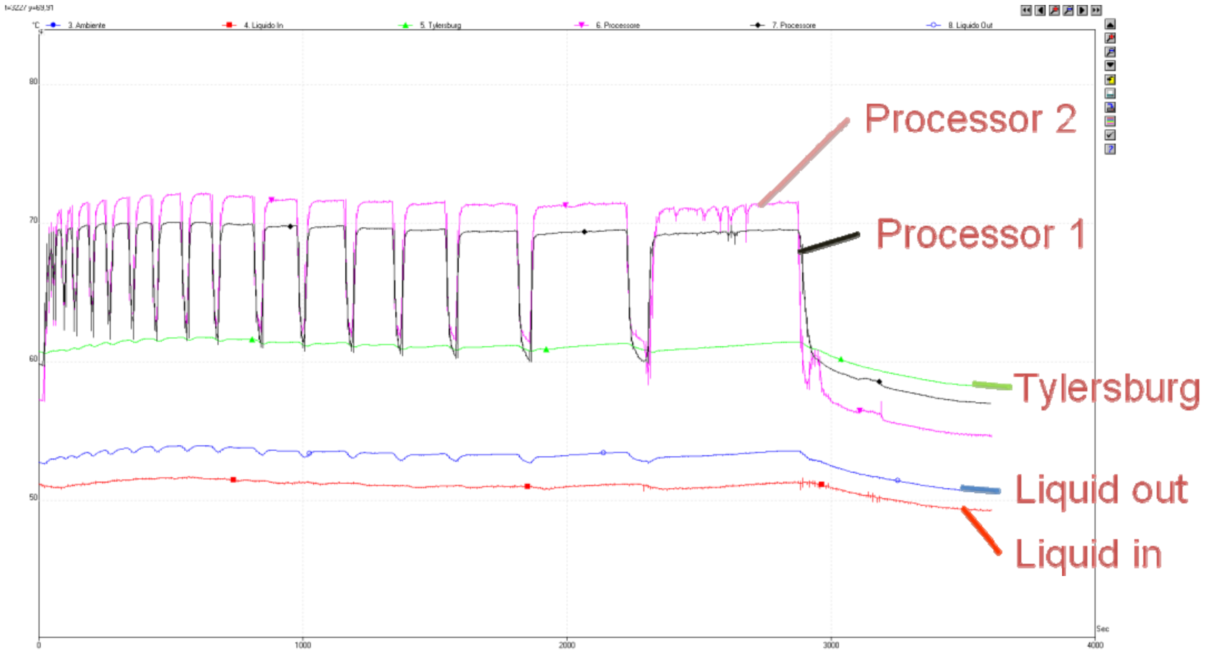


Figure 10 – Temperature measured on most significative parts during High Performance tests

4.7 Technical Specifications

Power Node technical specifications:

- DUAL INTEL XEON 5600 CPU: Higher performance enabled by six core Intel Xeon 5600 processors (up to 3.46 GHz). Up to 167 GFlops per unit.
- MODULAR: Power Node modules can be easily grouped to form 2U or 1U servers fitting 19'' racks.
- ON BOARD DDR3 RAM: 12/24GB of DDR3 RAM soldered on board. No modules required, faster access, better signal integrity.
- SATA SSD: Up to 256 GB high data retention SATA 1.8" SSD drive, vibration/shock proof.
- ENVIRONMENTAL DATA:
  - Designed to meet MIL-STD-810G.
  - Operating Temperature: -40 to +70°C.
  - Storage Temperature: -40ºC to +85ºC.
  - Operating Shock: 40g, 11ms, 3 pos/neg per axis, 18 terminal peak sawtooth pulses (MIL-STD-810G, Met.516).
  - Crash Safety Shock: 75g, 11ms, 2 pos/neg per axis, total 12 sawtooth pulses.
  - Random Vibration: 3 Axes, 1 Hour/Axis.
  - Functional humidity 60%: Up to 95% RH @ 40°C, Non-Condensing.
  - Water Immersion: 1 Meter Submersion, 30 Minutes.
  - Dust Ingress: Designed for Compliance w/Method 510.4, No Dust Ingress.

  - Operational Altitude: Sea Level to 9,144 meters, Storage Altitude: Sea Level to 18,288 meters.
  - Rapid Decompression: 8,000 to 40,000 ft (2,438 to 12,192 meters).
- EXTERNAL INTERFACES: 10/100/1000 ethernet I/F, RS232, USB2.0, VGA output, I2C, up to 2 high speed channels (10GE, IB).
- POWER: 24 VDC Input. Vehicle grade PSU with protection against reverse, overvoltage, surge. Max. Power Dissipation 500W.
- CONNECTORS : D38999 (mil-dtl-38999) hermetically sealed.
- PHYSICAL: Chassis: Aluminum Alloy, Corrosion Resistant. Finish: Anodized per MIL-A-8625, Type III, Class 2.
- EMI/EMC Isolation Designed to meet MIL-STD-461E.
- MTBF Per MIL-HDBK-217 @ 71C:
  - ~ 40,000 hours – Airborne Inhabited Fighter / Ground Mobile.
  - ~120,000 hours – Ground Benign.
- DIMENSIONS: 4,45cm (h) x 22,5cm (w) x 70cm (l);
  - with cooler 8,9cm (h) x 22,5cm (w) x 70cm (l);
  - with cooler for extreme temperature 13,35cm (h) x 22,5cm (w) x 70cm (l).

5 Off-the-shelf power node

The main goal of pSHIELD was to develop prototype technologies for the security, privacy and dependability. Though conventional off-the-shelf processor boards might not be feasible to satisfy all SPD requirements, they were used in pSHIELD as a gateway between the sensors and the telecom M2M (machine-to-machine) platform. Both the off-the-shelf platform “VIA board” and the telecom M2M platform “Shepherd” will be shortly described in this deliverable. As no modifications were performed on these platforms, the reader is deferred to the references for a detailed description.

This type of node supports hich processing power and direct communication with the sensor nodes. The power nodes can manipulate complex ontologies and rules. Four our ptorotypical implementations, we used Linux-based embedded systems.

5.1 Overview VIA EPIA power node

One example of such an embedded system is the VIA EPIA N700 Nano-ITX board, which is integrated with a VIA VX800 media system processor, an all-in-one shipset solution. The VIA EPIA N700 is equipped with a power-efficient 1.5 GHz VIA C7, supports up to 2 GB of DDR2 system memory and includes two onboard SATA connectors, USB 2.0, COM and Gigabit LAN ports. Expansion includes a Mini-PCI slot with an IDE port, additional COM and US ports and PS/2 support available through pin-headers. The implementation uses Ubuntu Linux Kernel 2.6.32-24-generic and Java runtime environment (JRE) 1.6 for development on this embedded system. In the proposed prototype, this system hosts an application facilitates the communication between SPOT sensor and the M2M platform.

Figure 25 - VIA EPIA embedded board

5.2 Prototypical adaptations

Though the embedded platform has the necessary processing capabilities and interfaces, some adaptations had to be performed to fit the real-world requirements on board of the JBV measurement locomotive Roger. These requirements include the *(i)* fully-automatic boot on power-up and *(ii)* the connectivity/re-connectivity of the micro-sensors. A short description is listed here, while details of these implementation are given in Deliverable D6.4.

The adaptations include

- The boot sequence of the VIA EPIA board, starting first the communication to the telecom M2M platform and then the communication with the sensors. The sensor communication would otherwise throw and exception error “not being able to communicate”.
- A pass-by of the manual confirmation to start the Sun SPOT application. The Sun SPOT expects a confirmation through the GUI, which had to be by-passed to allow an automatic operation.
- A recovery action when the Sun SPOT sensor was out of reach. The initial implementation of the Sun SPOT expects that the host is already listening before the sensor starts transmitting the “here I am” message. The “error exception” of “no contact to host” from the sensor had to be escaped, delaying the sensor broadcast message and turning the host into listening mode until successful connection.
- An updated broadcast functionality after boot, being active until the link is established.

While the initial plans for the prototype were the connectivity to the Telenor Objects platform, the real-world requirements of installing the sensor platform on the locomotive caused us to perform tedious software adaptations. The overall functionality is described in section 5.5, after the technical specifications of the VIA board and an overview over the Telenor Objects Shepherd platform.

5.3 Technical specifications

The VIA EPIA N700 Nano-ITX board is referenced in http://www.viaembedded.com/en/products/boards/productDetail.jsp?productLine=1&id=710&tabs=1

The key features include:

- Integrated VIA Chrome9™ HC3 DX9 3D/2D graphics with MPEG-2, WMV9 decoding acceleration
- Supports DDR2 533/667 SDRAM (SODIMM)
- Supports wide temperature from -20 to 70 °C (only for EPIA-N700-10EW)
- Supports one IDE and two SATA
- Supports one PCIe Gigabit Ethernet
- Supports one CF (Compact Flash) Type I
- Supports one VGA port
- Supports four USB 2.0 ports (two as pin headers)
- Onboard LVDS support: one dual channel LVDS panel support
- Supports HD audio

In the pSHIELD railway prototype, the main purpose of this board was to communicate with the sensors and the Telenor Objects platform Shepherd.

5.4 Telecom M2M platform “Shepherd”

For the pSHIELD implementation we used the machine-to-machine platform “Shepherd” from Telenor Objects. This chapter only provides a short introduction, as the platform itself is reasonably documented on the confluence wiki: https://shepherd.onconfluence.com/display/service/Home

The wiki provides a detailed functional description of the Shepherd service platform and is a good introduction to understanding the possibilities and strengths of this managed service enablement service. As the description of the wiki is continuously enhanced, we do not see a reason for doubling information here.

The Shepherd platform is an instance of ETSI TS 102.690 for machine-to-machine platforms providing interoperability and integration of communication between sensors and end-user applications. The type of devices that can be connected range from nano and micro sensors to personal devices such as mobile phones, and make all of theses sensor data being accessible from anywhere at anytime through standardized interfaces. The platform offers a number of services, including:

- Service management for monitoring, device configuration, SLAs, and service support.
- Service enabler that has specific APIs for access to other modules.
- A message engine handling and securing message flow, including capturing, processing, routing and storage of data in tan environment.

- Notification services that inform about the status of devices and applications, and
- A Device Library with interfaces for tools and service recognition.

From the two methods of establishing connection we used the HTTP Connection API, as it was more suited towards our application and easier to implement. The alternative method of using the Connected Objects Operating System (COOS) is based on Java open source tools, and does allow for operations on the raw data before transmitting them over the mobile network.

5.5 Sensor Integration and Prototype

The devices used in the implementation are shown in Figure 26, containing the VIA board, the Sun SPOT host and the sensor, as well as a GPS and an iPhone. The communication unit for 2G/3G mobile communication is outside of the frame of this figure.

The purpose of the prototype is to open for *(i)* detection of unusual conditions such as high temperature of components, vibrations and unexpected movements, and *(ii)* transferring making available such information to different actors, e.g. the train operator, the rail infrastructure owner, and the customer transporting goods on the train. This information should be made available both automatically and in a request/response demand-based mode. The train is equipped with several heterogeneous computing devices such as sensors, actuators, GPS receiver, and a gateway-embedded computer for detection of such anomalies. The devices interact using heterogeneous protocols for the sensing the information in their vicinity and send the information to the gateway. As an intelligent device, the gateway figures out any irregularity, and sends details to the smart train operator.

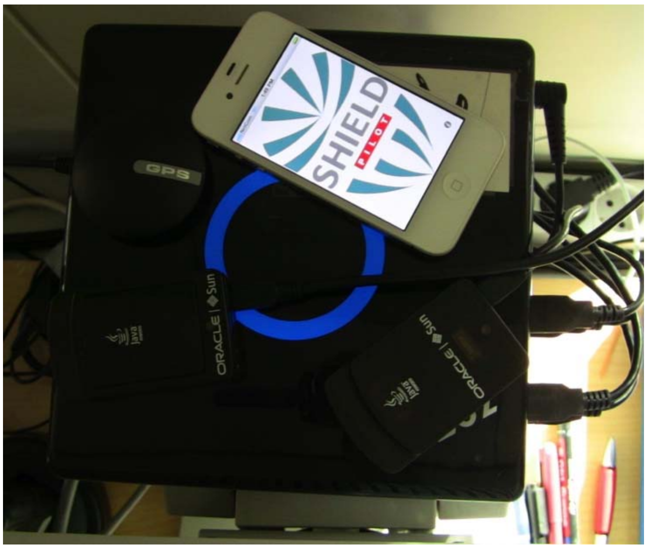


Figure 26 - The embedded platform with the Sun SPOT host, the Sun SPOT sensor and the GPS

Our prototype uses Sun SPOT sensors for temperature, acceleration and light conditions, as well as a GPS and a Smartphone for positioning. The initial prototype considered the aspects of *privacy* and *dependability*, as it handled the data distribution according to semantically defined roles. An extended version of the prototype was foreseen to handle *security*, but due to major programming challenges the encryption of the link to the sensor had to be delayed to the nSHIELD project. Further documentation of the outcome of the prototype can be found in deliverable D6.4.

6 Enhanced Power Management

6.1 Introduction

Power supply design for an ES is one of the critical points in the design process, as requirements are more restrictive as time goes by. Due to the fact that the system is on a single chip, the complete design is compact and the power supply cannot tolerate an exception. It is important to be careful about power supply requirements like initial conditions, transient behavior and the effects of turning-on and turning-off different parts of the circuit.

As time goes by, voltage levels used in ES go down thus complicating the design (see Figure 27). Taking into account the relation between current and voltage (as one increases, the other decreases) higher current requires bigger and more expensive connectors, wires and traces, thus increasing the importance of reaching a compromise between voltage requirements and design costs.

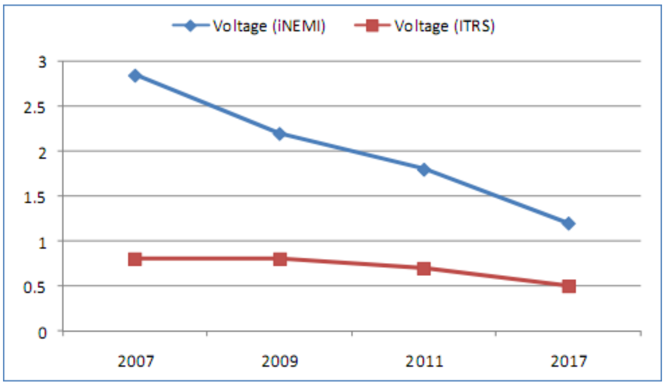


Figure 27 – ITRS-iNEMI (2010) System-to-Chip Supply voltage and threshold voltage trends

Moreover, lower power consumption entails less problems derived from heat dissipation in the final design. This premise improves other parameters, making possible to extend the battery life, increase reliability by reducing the switching current and decrease the packaging cost by reducing the heat dissipation.

Power consumption in ES depends on the number of internal logic transitions and it is proportional to the operating clock frequency. Thus, when increasing the device size the power consumption gets higher. It is common for a large, high-speed design to require several amperes of current.

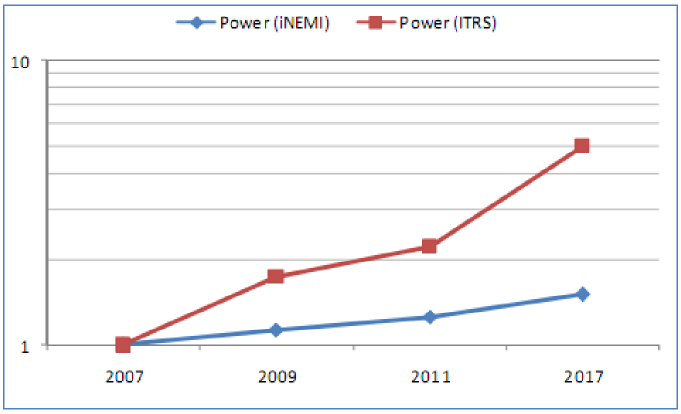


Figure 28 – ITRS-iNEMI (2010) System-to-Chip Power Comparison trends

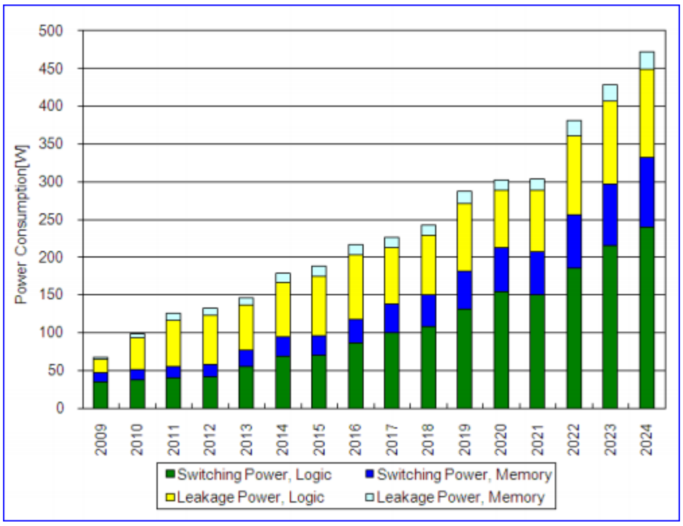


Figure 29 – ITRS projection for SOC Consumer Stationary Power Consumption Trends (2010)

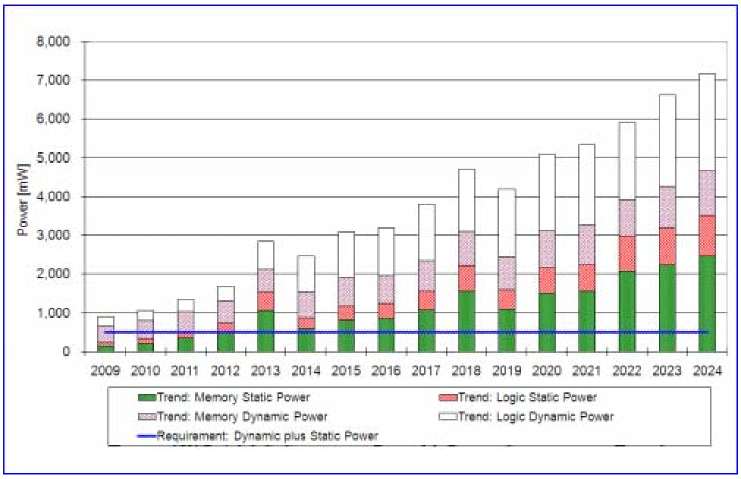


Figure 30 – ITRS projection for SOC Consumer Portable Power Consumption Trends (2010)

Several methods can be used to reduce the power consumption:

- Lower supply voltage: power consumption is proportional to the square of the supply voltage. Therefore, lower supply voltage will reduce power consumption.

- Full custom design: fewer gates will reduce switching activity and thus, lower power consumption.

- Clock gating: unneeded parts of the processor will be prevented from receiving the clock signal. Absence of clock signal will prevent any switching activity and thus, lower power need.

The redundancy in the design of a power supply is not a serious problem but is not desirable since it can add unnecessary cost and complexity to the overall ES design. The task of power estimation is not a trivial one prior to complete the design.

6.2 Power supply components

Power node is the most complex model and offers high performance in terms of computing power. It can be considered as the first level of massive data elaboration but with the peculiarity that the computing power is provided directly on the field.

Power needs in this kind of devices vary substantially from nano and micro nodes. Depending on the installation and the environment, the power source can be different. The main source will be provided by AC Power whereas the secondary one will contemplate other power harvesting methods and energy storage systems.

Special attention will be paid to power supply protections, not only to avoid damages into the system but also to provide a continuous power source. The power consumption of this device is around 350W and, due to its high consumption, the secondary power source will be focused on providing enough autonomy to send an alarm warning.

There are real alternatives that could provide SPD features since the architecture combines a powerful processor with a FPGA. The power consumption of all these platforms is between 350W and 500W.

Table 4 – Possible models of Power Nodes

Model	Features
OpenVPX Intel Core i7 Dual-Core LDS6520 Module (***pSHIELD power node***)	• 2l Intel Xeon 5570/5680 CPUs at 2.93/3.33GHz, at least 6GB RAM 1333MHz DDR3, • Altera Stratix IV FPGA
OpenVPX Intel Core i7 Dual-Core LDS6520 Module	• Intel Core i7 Arrandale (Westmere-class) dual-core, 2.53 GHz with 40 GFLOPS peak performance, 8 GB of DDR3 SDRAM, NAND flash 4 GB • Altera Stratix® IV EP4SGX180 FPGA
OpenVPX Intel Xeon Dual Quad-Core HDS6600 Module	• Intel 45-nm Nehalem-Class Processor, Quad-core LC5518 Jasper Forest (2 at 1.73 GHz each) with 55 GFLOPS, 12 GB of DDR3-1066 with ECC, 2 GB of NAND flash. • Altera Stratix® IV EP4SGX180 FPGA
CHAMP-FX2	• Dual-core Freescale Power Architecture™ MPC8641 processor, 1GB of DDR2 with ECC, 512MB of Flash. • Two user-programmable Xilinx® Virtex®-5 FPGA (LX110T or LX220T)
AXA-110 Intel Core™ 2 Duo AMC	• Intel Core 2 Duo with 1.5-GHz core frequency. 2 or 4 GB of 64-bit DDR2-400 SDRAM with ECC • Xilinx® Virtex™-5 FPGA
NAMC-QorIQ-P50	• Freescale QorIQ P5020 dual core processor at up to 2.2 GHz, 2-8 GB DDR3 SDRAM at 1.3GHz, 2 GB of NAND Flash. • Xilinx Virtex-6 FPGA

6.2.1 Energy Storage Systems and Power Harvesting Methods

The most common solution for this kind of devices is the installation of a battery backup to provide an uninterruptible power supply (UPS) during a period of time long enough to alert the system in case the primary power source is lost.

There are other alternatives, like installing an UPS based on fuel cells or even provide a combined solution where the fuel cell is used to recharge the batteries and thus, extend the autonomy of the system.

When an AC power source is not available, a solution based on power harvesting methods could be installed. This requires a design of an energy system to feed power nodes through renewable energy sources.

The main limitation of this kind of technologies is the efficiency, since the harvested energy depends on the environmental conditions. This is not a minor issue that should be considered during the design phase.

An autonomous system, which power consumption is around 500W, requires two wind turbines, eight solar panels, two fuel cells and 24 batteries. More relevant features are specified in Table 5.

Table 5 – Autonomous power system – Required components (features)

Model	Features
Airdolphin PRO / Mark-Zero (Wind Turbine)	• Rotor diameter: 1800mm • Tower diameter: 48.6mm • Mass: 18Kg • Start-up Wind Speed: 0m/s (Power-Assist Function) • Peak Instantaneous Power: 2.3kW (20m/s)
Sunpower 290 (Solar Panel)	• Peak Watts/Panel: 290W • Efficiency: 17.8% • Peak Watts/m^2: 178W • Weight: 18.6 Kg • Dimensions (mm): 1559 x 1046 x 46
EFOY Pro 1600 (Fuel Cell)	• Charging capacity: 1560 Wh per day • Dimensions (L x W x H): 433 x 188 x 278 mm • Fuel: Methanol
OPzV Cell 2V 12 OPzV 1200 (Battery)	• Capacity, C10 (1.8 V/cell, 20 °C): 1340Ah • Nominal voltage of battery cell: 2V • Efficiency factor (Ah): 95% • Dimensions (L x W x H): 275x210x669 mm • Weight: 97 Kg

This kind of installation has been calculated to power continuously a system up to 500W and ensures the autonomy of the system during ten days if energy harvesting system fails. Up to 3 different kinds of technologies must be integrated to ensure the autonomy of the system: fuel cells, solar and wind energy.

This solution is not suitable to be installed in all scenarios due to its size. For instance, pilot demonstrator defined in WP6 aims to monitor freight trains transporting hazardous material. A power system like the one needed to provide 500W continuously will need a wagon only to install all batteries and the power node. Solar panels can be integrated in the roof and special attention will be paid to wind turbines since the installation must consider all possible hazard elements that could be present during the route, like tunnels or other facilities where a maximum height is allowed.

6.3 Power Supply Protections

Power nodes also need some protections to avoid damages into the system. These devices are directly plugged into AC power. Therefore, the design of the protection board is different from nano and micro nodes.

Besides the protections against short circuits, overloads, over currents and over voltages, the design includes an EMI filter to bring the electrical noise down to acceptable levels. In either power supplies or electronic equipments, the EMI filter keeps any internally generated noise contained within the device and prevents any external AC line noise from entering the device.

Figure 31 shows the necessary components to convert AC to DC in a secure way.

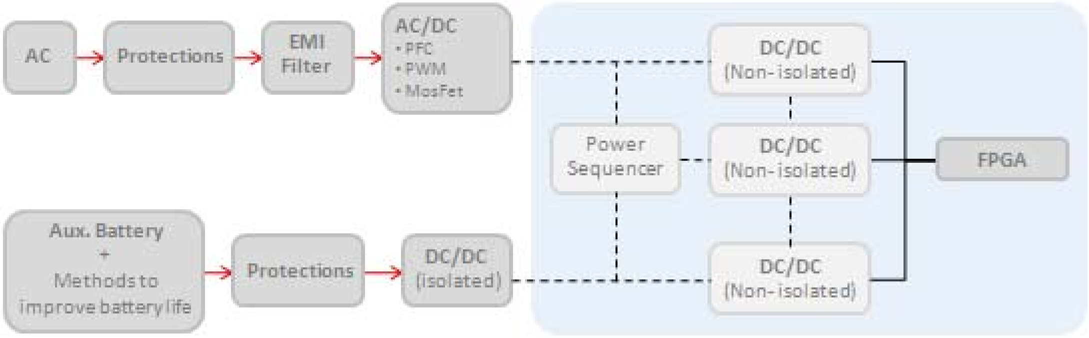


Figure 31 – Power supply components - General design

Two different prototypes have been designed:

- The first one contains ***Thermal Fuse Varistors*** which protect the system against high voltage transients but, even if these devices break down, the system is able to continue working without any protection against these transients.

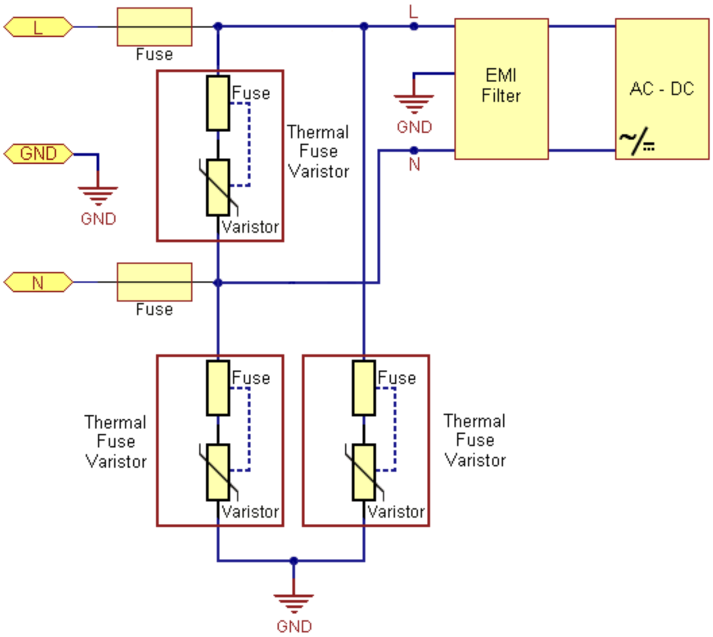


Figure 32 – Protection Board – Thermal Fuse Varistors

- The second one combines ***Varistors*** with a ***Gas Discharge*** to avoid any damage in the system. Unlike the first design, this one disconnects the system from the AC power when the protection board cannot avoid damages against high voltage transients.

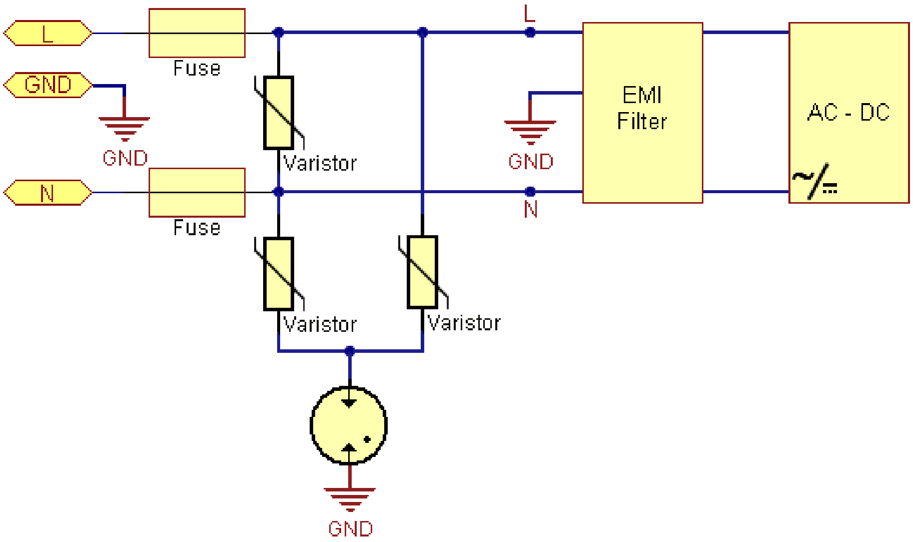


Figure 33 – Protection Board – Varistors and Gas Discharge

6.3.1 Prototypes

Both protection boards have been designed taking into account the normative EN/60950-1. Several tests have been carried out in order to ensure that these protections can avoid damages into the system.

The effect of parameters like leakage current, shock waves, harmonics, ESD or continuous over voltages, have been considered during test phase to check the protection boards.

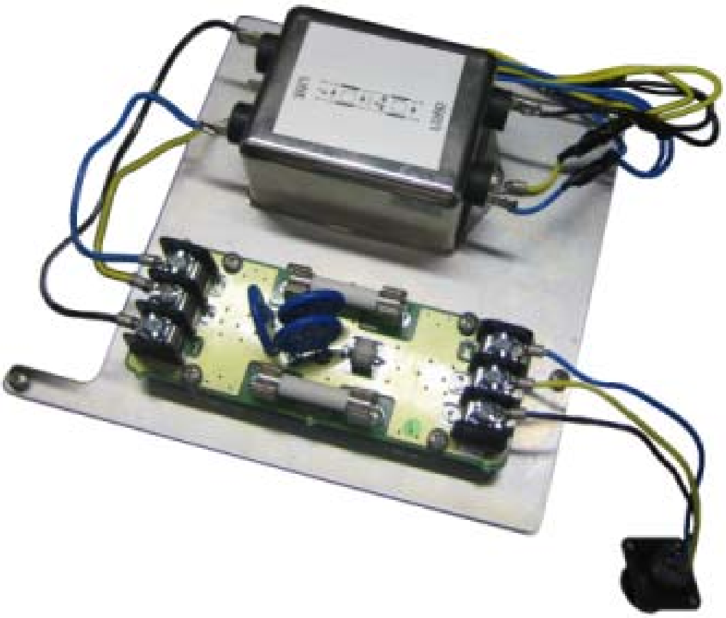


Figure 34 – Protection Board prototype with Varistors and Gas Discharge

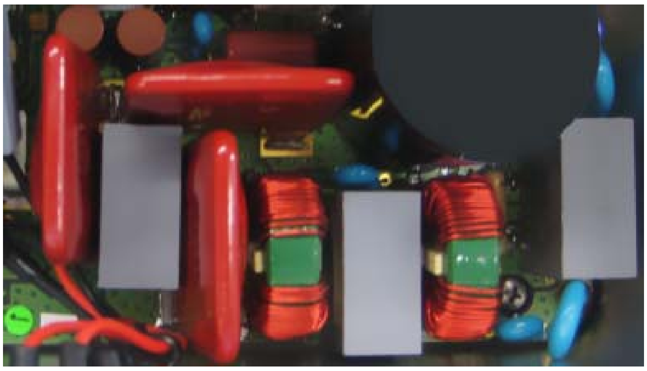


Figure 35 – Protection Board prototype with Thermal Fuse Varistors

Both designs fulfil the specifications defined to achieve SPD features since they are able to protect power nodes against short circuits, overloads, over currents, over voltages and electrical noise.

In all systems, the most critical element is the battery. Different temperatures affect the internal chemical reactions rates, the internal resistance and the efficiency so the run times, charge times and the battery life can vary when the battery operates at different temperatures.

The temperature range of a system is usually defined by the battery. Figure 36 shows the battery performance at different temperatures and defines the recommended temperature range to ensure a proper operation (-10ºC to 50ºC). The other components in the system must be able to work in the range determined by the battery.

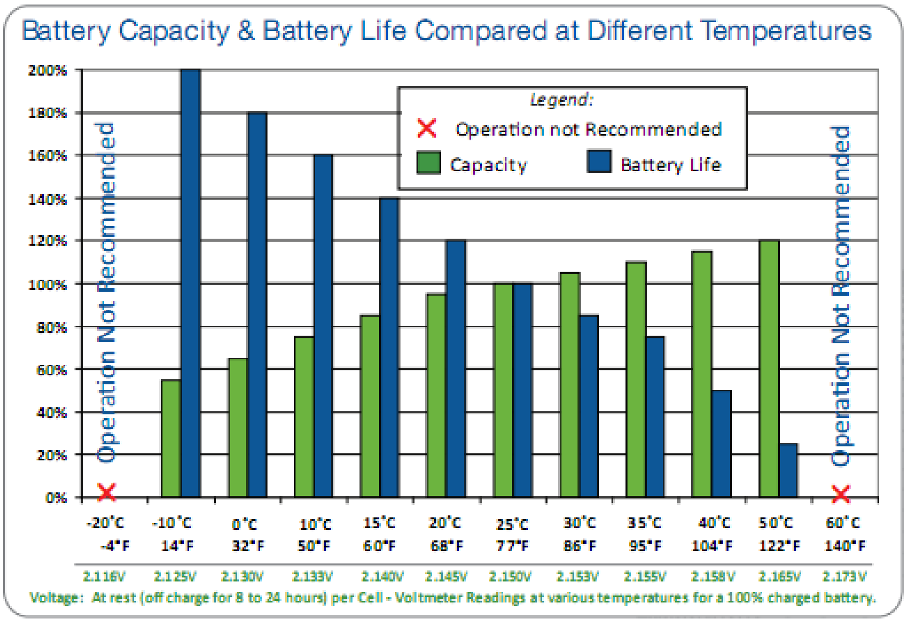


Figure 36 – Temperature effects on battery performance and Life by Discover® and Clean & Green™2

[2] Data provided as representative only. Battery voltage, capacity and life will vary with actual environmental conditions and operator driving habits. Operation above 50˚C / 122°F and below -10˚C / 14°F is not recommended. Temperature: C: Celsius, F: Fahrenheit. Capacity: Operation or available “run time” as a % of base-line capacity established using industry standard testing at 25˚C / 77°F. Battery Life: Expected battery life as a % of base line life established using industry standard testing at 25˚C / 77°F. Voltage: For Discover® Batteries, multiply the voltages shown by 3 for 6-volt batteries, by 4 for 8-volt batteries and by 6 for 12-volt batteries

Temperature tests have been carried out to both protection boards, by means of a climatic chamber. Satisfactory results have been obtained since the protection boards have demonstrated to maintain their behaviour with temperatures up to 65ºC and under -10ºC (the relative humidity has been remained at 90% during all phase tests).

6.4 Monitor Power Supply

The FPGA can monitor the power consumption through a current sense amplifier and can also check if main power source has failed. Thus, the system will have enough time to send an alarm warning before running down the auxiliary battery.

Dynamic reconfigurability is the main advantage of an FPGA since its greater flexibility allows a reduction in the power consumption and reuse the hardware. If a FPGA cannot be used for this implementation, an alternative solution could be provided to monitor power consumption: a microcontroller, and ADC and a current sensor are enough to measure this parameter. Table 6 contains the main features of several components needed to develop a platform to monitor and control power supply consumption.

Table 6 – Components to monitor power supply

Model	Features
MAX4375FEUB High-Side Current-Sense Amplifier	• Current-Sense Amplifier plus Internal Comparator and Bandgap Reference with Improved AccuracyTower diameter: 48.6mm • 50µA Supply Current • Single +2.7V to +28V Operating Supply • Gain +100V/V • Temperature range: -40ºC to +85ºC
ACS714LLCTR-20A-T Hall Effect-Based Linear Current Sensor IC	• Automotive Grade, Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kVRMS Voltage Isolation and a Low-Resistance Current Conductor • SupplyVoltage: 5V (Typ.) • Supply Current: 10mA (Typ.) • Sensitivity: 100mV/A • Temperature range: -40ºC to +150ºC
AD7810 ADC	• 10-Bit ADC with 2.3 µs Conversion Time • Operating Supply Range: 2.7 V to 5.5 V • Low Power Operation: 270 µW at 10 kSPS Throughput Rate 2.7 mW at 100 kSPS Throughput Rate • Temperature range: -40ºC to +105ºC
AD7277 ADC	• Throughput rate: 3 MSPS • Specified for V_{DD} of 2.35 V to

	3.6 V • Power consumption 12.6 mW at 3 MSPS with 3 V supplies • Temperature range -40ºC to +125ºC
AT32UC3A1512 Microcontroller (AVR32)	• CPU: 32-bit AVR • 6 SPI, 1 I2C, 4 UART, 1 SSC, 1 Ethernet, 8ADC Channels (10-bit resolution), 2 DAC (16-bit resolution) • SRAM: 64Kbytes • Flash: 512 Kbytes • Temperature range: -40ºC to +85ºC
C8051F133/131	• High Speed 8051 µC Core • 1 SPI, 1 I2C, 2 UART, 5 Timers, 8ADC Channels (10-bit resolution) • RAM: 8Kbytes + 256bytes • Flash : 64/128 Kbytes • Temperature range: -40ºC to +85ºC

The advantage of using the FPGA instead of a microcontroller is the reaction time in case of failure or anomalous behaviour. The FPGA could turn off any sub-system faster than other microcontrollers. For instance, if power consumption is higher than the one expected, it is easier to avoid damages into the system if a FPGA controls the power supply since the reaction can be almost instantaneous.

7 Conclusions

Deliverable D3.3 represents the Power Node Task 3.3 partners efforts to provide pSHIELD power node layer solutions according to Technical Annex specification, and developed according to the consortium Requirements (D2.1.1), Metrics (D2.2.1) and Architecture (D2.3.1).

This is the document that reports works in WP3 Task 3.2 Power Node, covering wide rage of ES solutions including different kinds of power nodes as well as various mechanisms used in that nodes. A short discussion of the SPD-related features of each power node is found in the respective sections.

References

[1] Technical Annex for ARTEMIS JU pSHIELD project number SP6 100204

[2] D2.1.1 "System Requirements and Specification"

[3] D2.3.1 "Preliminary System Architecture Design"

[4] Deliverable M0.1 "Formalized conceptual models of the key pSHIELD concepts"

[5] D3.1 "SPD node technologies prototype"

[6] D3.2 "SPD nano, micro/personal node technologies prototype report"

[7] D3.4 "SPD self-x and cryptographic technologies prototype report"