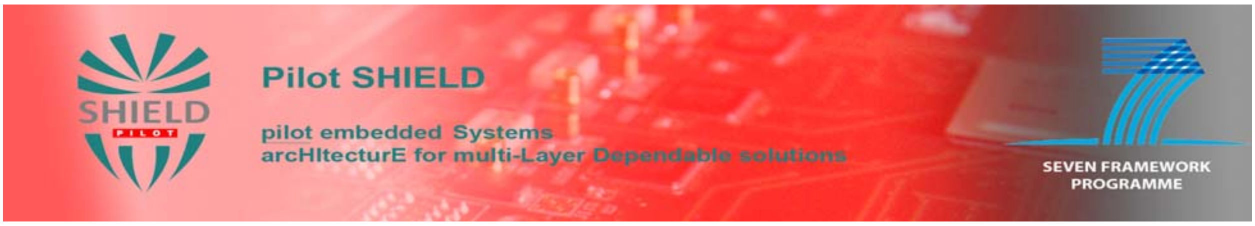

Project no: 100204

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	
PP	Restricted to other programme participants (including the Commission Services)	X
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		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 prototype [SESM]

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 case study proposes the use uses FSK modulation to transmit data between intrusion sensors placed in different cars of a freight train, into an SPD Power Node. The sensors must be equipped with a modulator, and each transmit using different carriers. The Power Node receives the signals, demodulates them, processes the data, encrypts it and sends to a control center through the pSHIELD Network.

2.1 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.

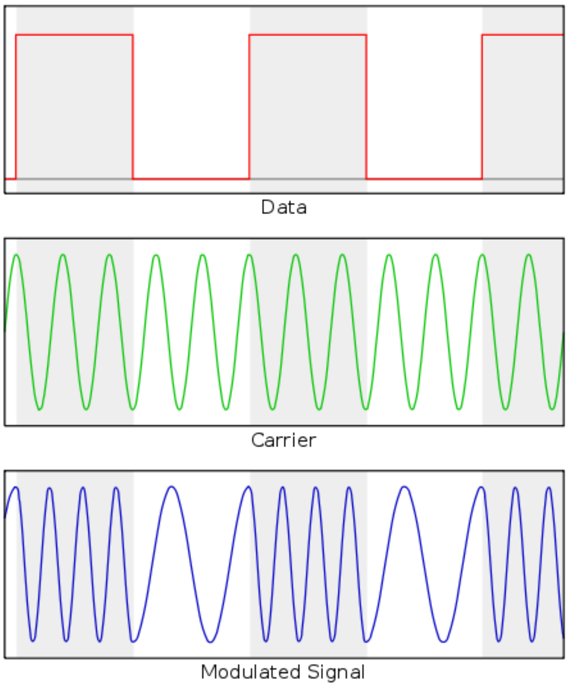

Figure 1 – FSK signal example

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

Audio Frequency-Shift Keying (AFSK) is a modulation technique by which digital data is represented by changes in the frequency (pitch) of an **audio tone**, yielding an encoded signal suitable for transmission via radio or telephone. Even in AFSK modulation, the transmitted audio signal alternates between "mark" tone, which represents a binary “one”, and "space" tone, which represents, instead, a binary “zero”.

AFSK differs from regular frequency-shift keying in performing the modulation at baseband frequencies. In radio applications, the AFSK-modulated signal normally is being used to modulate an RF carrier (using a conventional technique, such as AM or FM) for transmission.

AFSK is not always used for high-speed data communications, since it is far less efficient in both power and bandwidth than most other modulation modes. In addition to its simplicity, however, AFSK has the advantage that encoded signals will pass through AC-coupled links, including most equipment originally designed to carry music or speech.

A-FSK Applications

Most early telephone-line modems used audio frequency-shift keying to send and receive data, up to rates of about **1200 bits per second**. The common **Bell 103** and **Bell 202** modems used this technique. Even today, North American caller ID uses 1200 baud AFSK in the form of the Bell 202 standard. Some early microcomputers used a specific form of AFSK modulation, the Kansas City standard, to store data on audio cassettes. AFSK is still widely used in amateur radio, as it allows data transmission through unmodified voiceband equipment. Radio control gear uses FSK, but calls it FM and PPM instead.

AFSK is also used in the **United States' Emergency Alert System** to transmit warning information. It is used at higher bitrates for Weathercopy used on Weatheradio by **NOAA** in the U.S., and more extensively by **Environment Canada**.

The CHU shortwave radio station in Ottawa, Canada broadcasts an exclusive digital time signal encoded using AFSK modulation.

2.2 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:

- A single proximity sensor, emulated by an FPGA based board, continuously simulating the distance to the closest object.
- 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 with a board with a Xilinx FPGA. 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 PC continuously presents some data on the screen, simulating a distance, modulates this data, and transmits it to the SPD Node. The SPD Node demodulates the signal, 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.

2.3 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.

2.4 A-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.

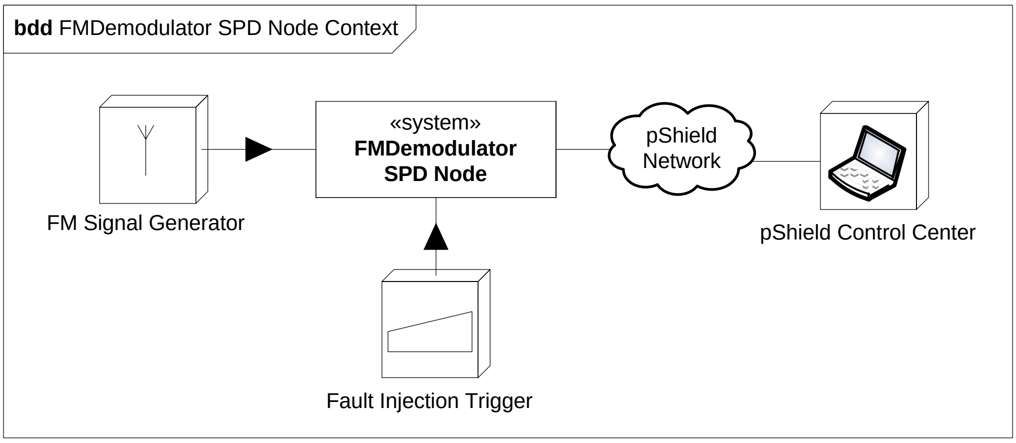

Figure 2 – A-FSK Demodulator demonstration context

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

2.4.1 A-FSK Signal Generator

The **A-FSK signal generator** has been implemented using the audio output channel of a PC audio board. The signal features are contained 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 data. Its parameters are given below in the table.

Table 1 – A-FSK modulated signal parameters

DIGITAL SIGNAL TO TRANSMIT	
A-FSK Rate:	50 Hz
A-FSK MODULATED SIGNAL	
"Space" frequency:	1070 Hz
"Mark" frequency:	1270 Hz
Amplitude:	1 Vpp

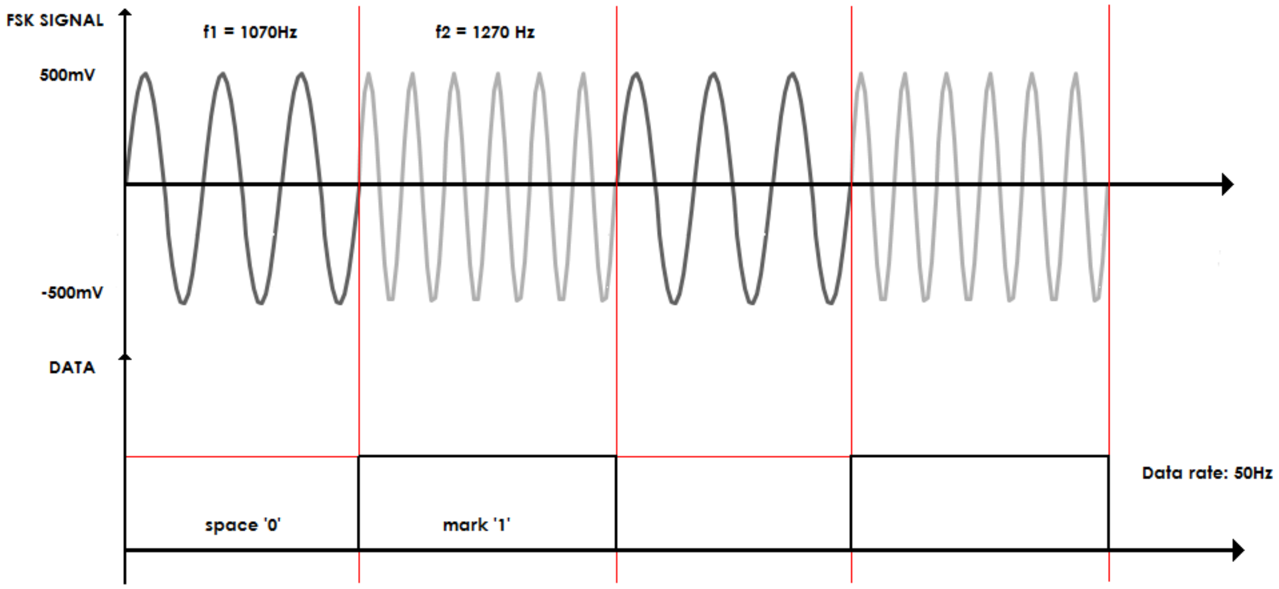

Figure 3 – FSK signal sample

Table 2 – A-FSK audio signal specification

File A-FSK_Signal.wav audio signal specification	
Attributes	8 bit; Mono; 8kHz;
Format	PCM
Duration	60 sec

2.4.2 A-FSK Demodulator SPD Node

A-FSK Demodulator SPD Node receives the data samples sent by Signal Generator block, and performs a digital demodulation of the samples. After that, it analyzes the characteristics of the sampled signal, in order to check if it is compliant with the expected characteristics. If it is so, the A-FSK Demodulator SPD Node sends all the samples analyzed to the pSHIELD Network using the Ethernet communication interface. On the contrary, it rejects the samples sent to pSHIELD network with an information (metric) that highlights the discovered problem, and waits for a information on action to do from the pSHIELD network. In this case the reaction may be as follows.

- **Reconfiguration**: after receiving a reconfiguration command from the pSHIELD network, the system starts an hardware **reconfiguration** of the Demodulator block. This action could be performed if the characteristic of the incoming signal has been changed in order to allow the processing a new type of signal.

- **Recovery**: after receiving a recovery command from the pSHIELD network, the system starts an hardware **recovery** of the Demodulator block. This action will be performed after injection of a fault in to the Demodulator. In this case, even if no changing of the incoming signal has been done, the processing is affected by a fault. So, only a recovery of the Demodulator block is required in order to establish the initial working condition.

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. These 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.

2.4.3 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, e.g. pushing a button. The scope of this block is to inject a fault into Demodulator process, so that it becomes necessary to recovery it.

2.4.4 pSHIELD Control Centre

A remote PC, connected to the network via Ethernet is used as pSHIELD Control Centre. A server/client application running on the PC allows a remote user to:

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

3 FPGA Based Architecture [SESM]

3.1 A-FSK Demodulator Node Layer Description

The A-FSK Demodulator is a proof of concept to demonstrate the SPD paradigm. In fact it implements a simple system managing a data stream with the constraints and suggestions of the SPD paradigm.

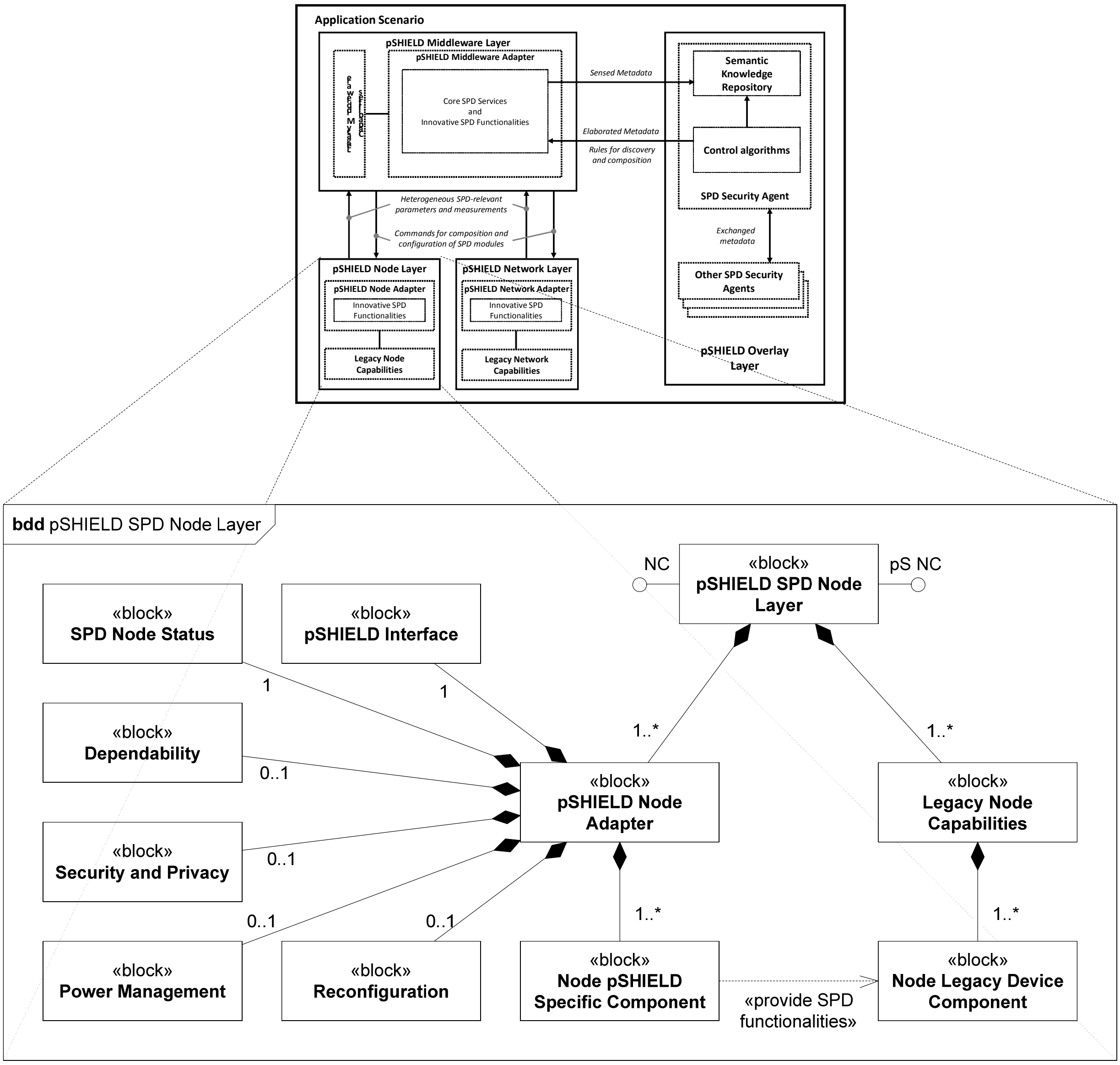

Figure 4 – Generic pSHIELD SPD Node Layer

The demonstrator implements the following features:

- a reconfigurable demodulator,
- a fault injection mechanism,
- a secure connection with the network layer.

The reconfigurable module allows the system to be dependable, and the fault injection mechanism shows how the system can resume its capabilities after a fault.

The secure connection allows the node to be reliable for the network layer and for the important information exchanging and application running at such a level.

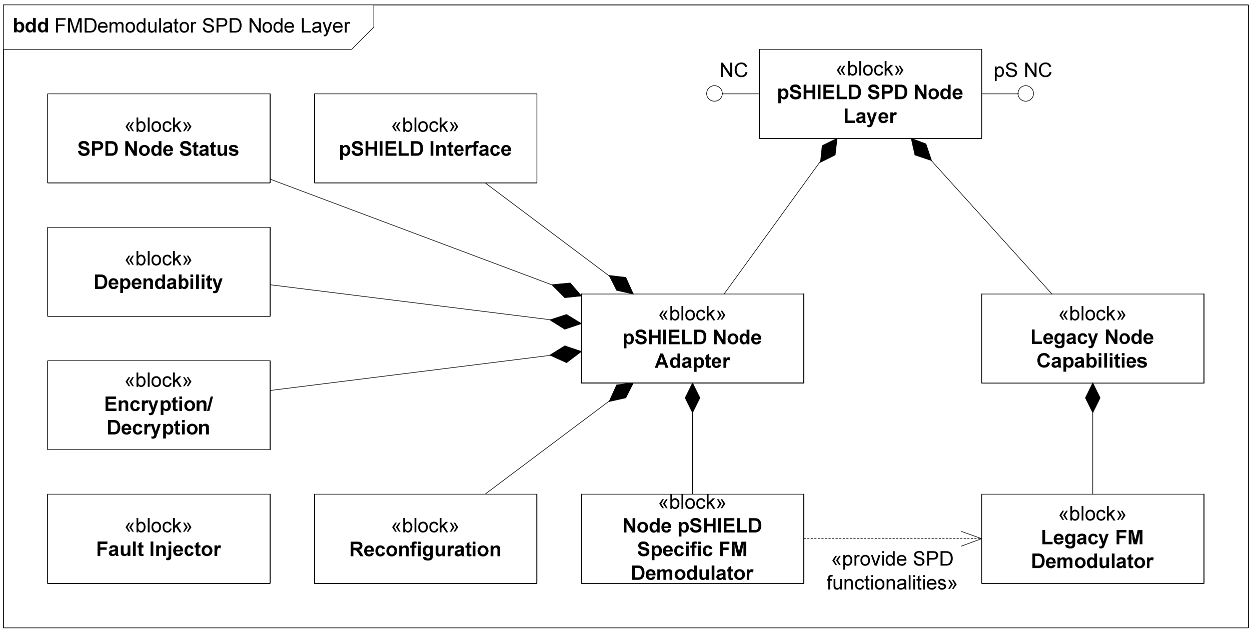
bdd FMDemodulator SPD Node Layer
NC
«block» pSHIELD SPD Node Layer
pS NC
«block» SPD Node Status
«block» pSHIELD Interface
«block» Dependability
«block» pSHIELD Node Adapter
«block» Legacy Node Capabilities
«block» Encryption/ Decryption
«block» Fault Injector
«block» Reconfiguration
«block» Node pSHIELD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

Figure 5 – A-FSK Demodulator SPD Node Layer

3.2 System Architecture

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

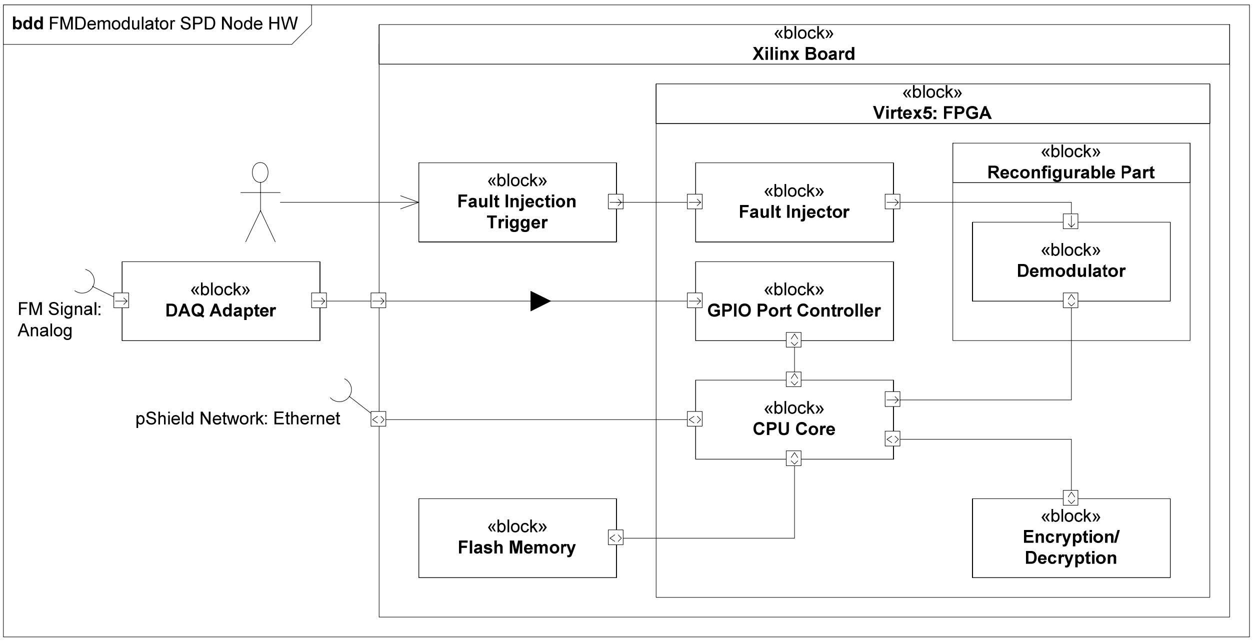

Figure 6 – A-FSK Demodulator hardware architecture

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.

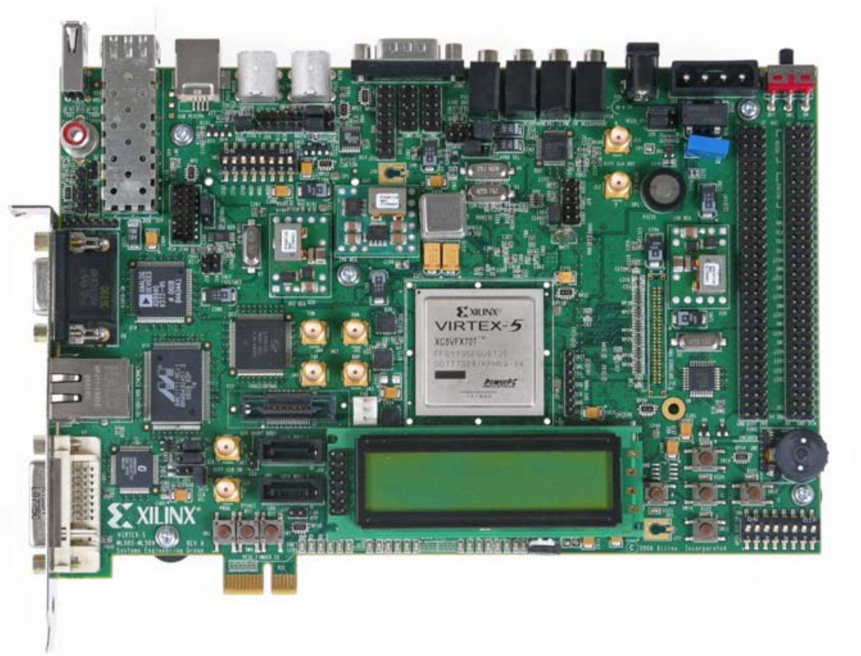

Figure 7 – ML507 printed board assembly (top view)

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

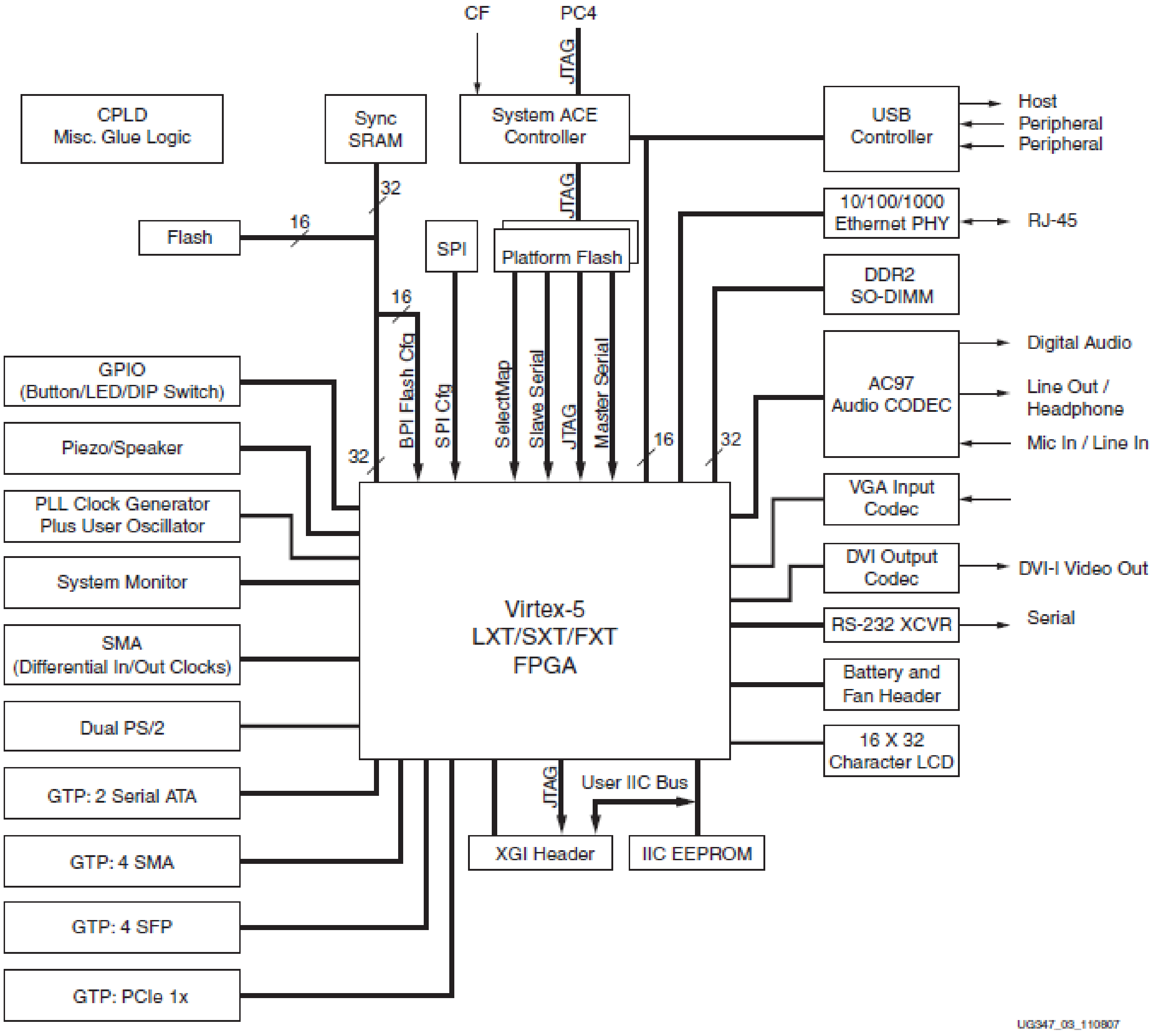

Figure 8 – Block Diagram of Xilinx ML507 Evaluation Platform

3.2.1 Interfacing evaluation board

The **DAQ Adapter** acts principally as an “add-on” module that is able to:

- perform an analog to digital conversion of an incoming A-FSK audio signal;
- feeds the 8-bit digitized data for the ML507 Xilinx board through the general purpose interface available on the board;

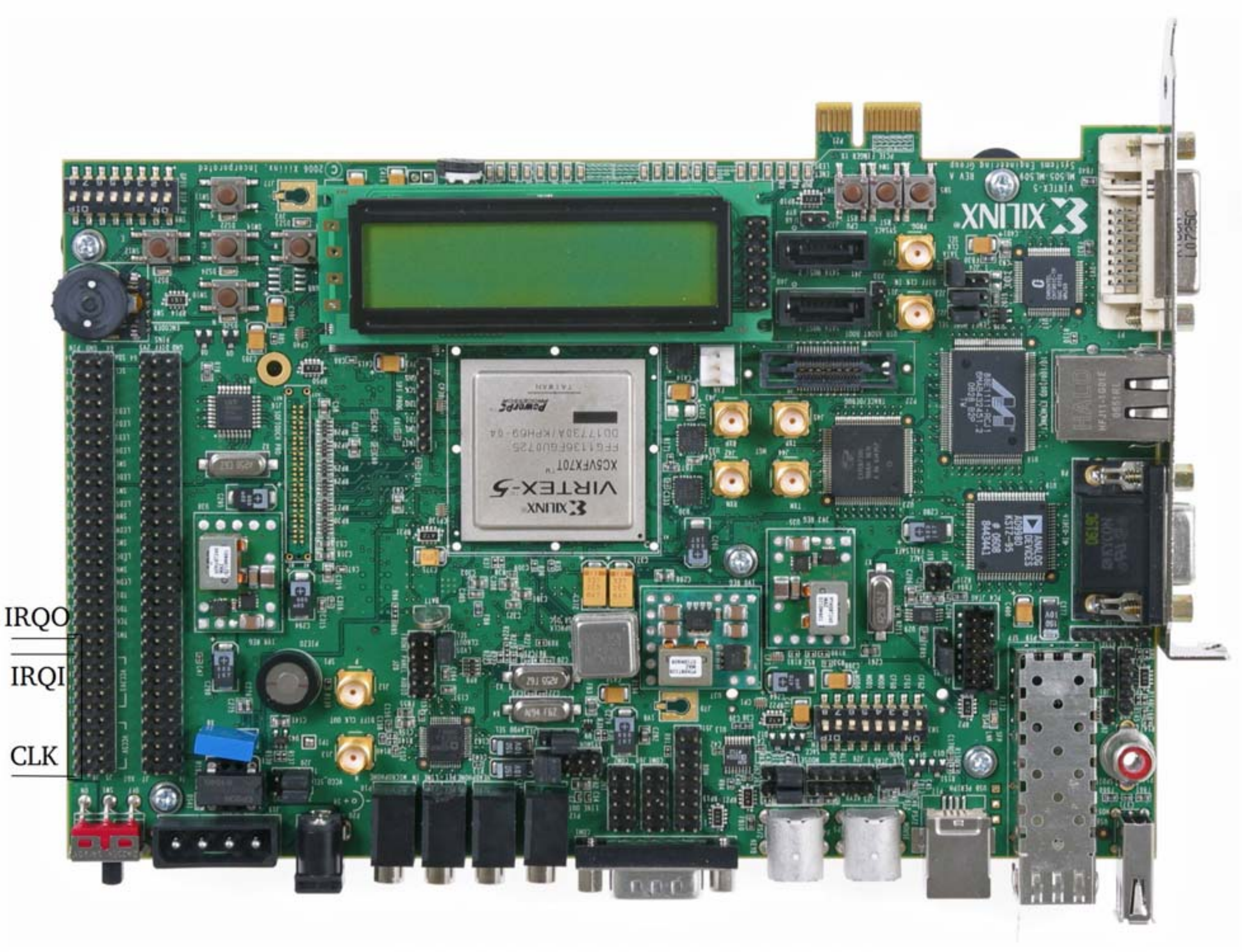

Figure 9 – The evaluation board

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

3.2.2 Demodulator

The demodulator implementation is presented in the following block diagram.

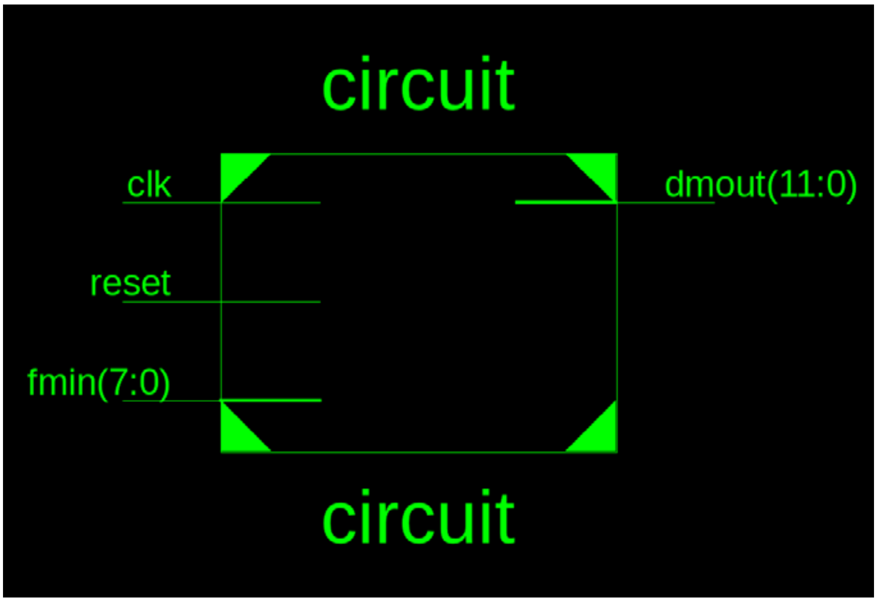

Figure 10 – Demodulator implementation block diagram

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.

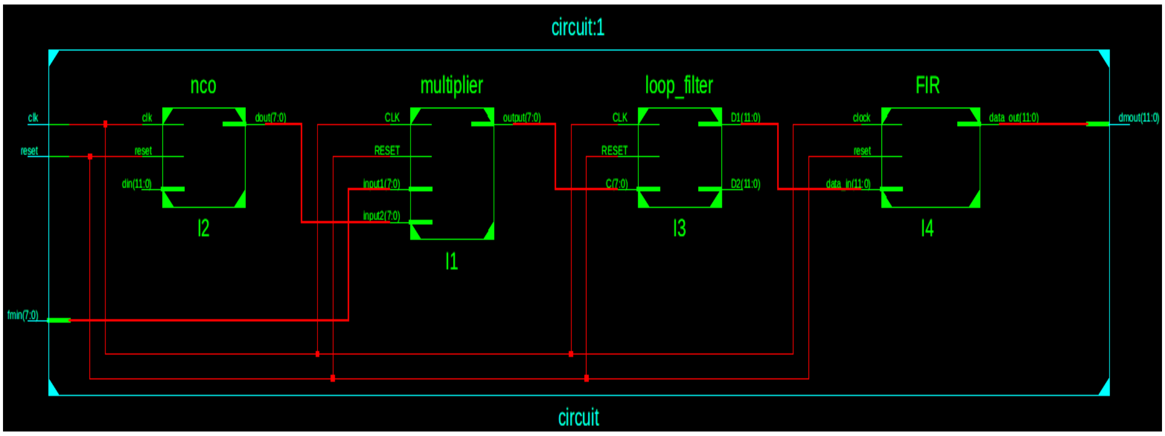

Figure 11 – Detailed demodulator diagram

The fmin byte stream is supposed to be fed by a FIFO input and the fmout word is supposed to fed a FIFO out, being both the FIFOs not properly parts of the demodulator.

3.2.3 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.

3.2.4 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.

3.2.5 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 artifact put into the SoC to show how the system restores its capability after a fault.

3.2.6 Encryption / Decryption

The encryption / decryption system is implemented at software level and only through the network layer.

3.2.7 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.

3.2.8 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.

3.2.9 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

3.3 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 describer as follow:

- IP application
    - reading the data structure to send
    - sending data structure to the network
    - reading data from the stack
    - filling data structure to use inside the application
- File System application
    - Building the application files
    - managing the files for ensuring the correct access
- Encryption application
    - getting clear data
    - encrypting clear data
    - filling data structure to be sent
    - receive encrypted data
    - decrypting data
    - filling private data structure with clear received data
- Reconfiguration application
    - Monitoring the fault injector
    - 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.

3.4 Prototype demonstration results

« Description of the tests performed.»

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

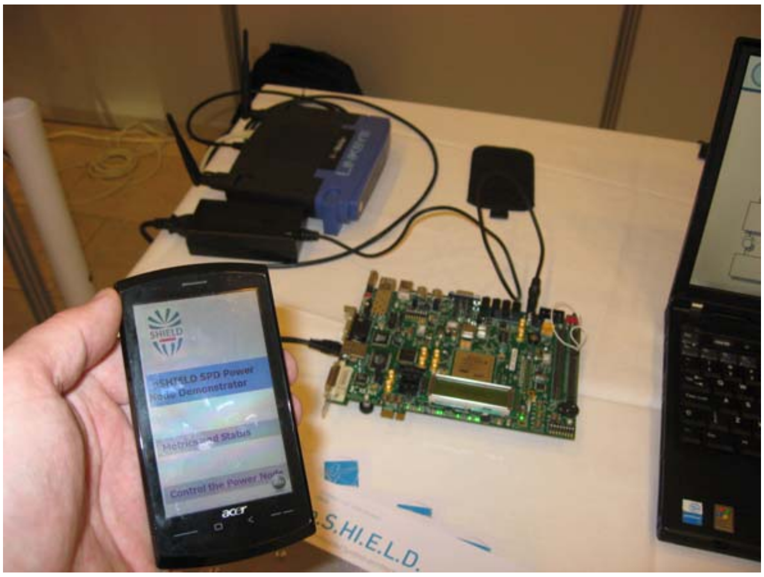

Figure 12 – Running FPGA Power Node Prototype

4 Rugged High Performance Computing Node [ETH]

4.1.1 Power node HW/SW

Power Node will be 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 is required.

The Power Node will be based on a powerful computing architecture: a dual Intel Xeon 5500/5680 series (Quad core CPU) motherboard, with at least 6GB of on-board soldered DDR3 memory and a high data retention 80GB SSD drive. A high speed, high density FPGA device will also be present, providing easy adaptability and implementation of dedicated functions and special algorithms. It will offer a maximum processing power of 80GFlops.

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. The shape of this cold-plate is intended, at this stage of the project, only for descriptive purposes. The final version could be different.

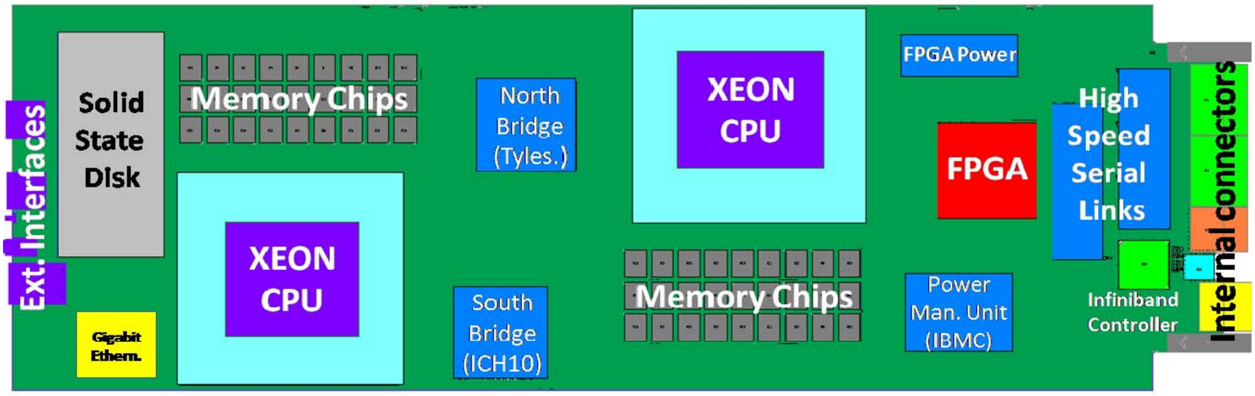

Figure 13 – Power Node board concept, without cooling heat sinks

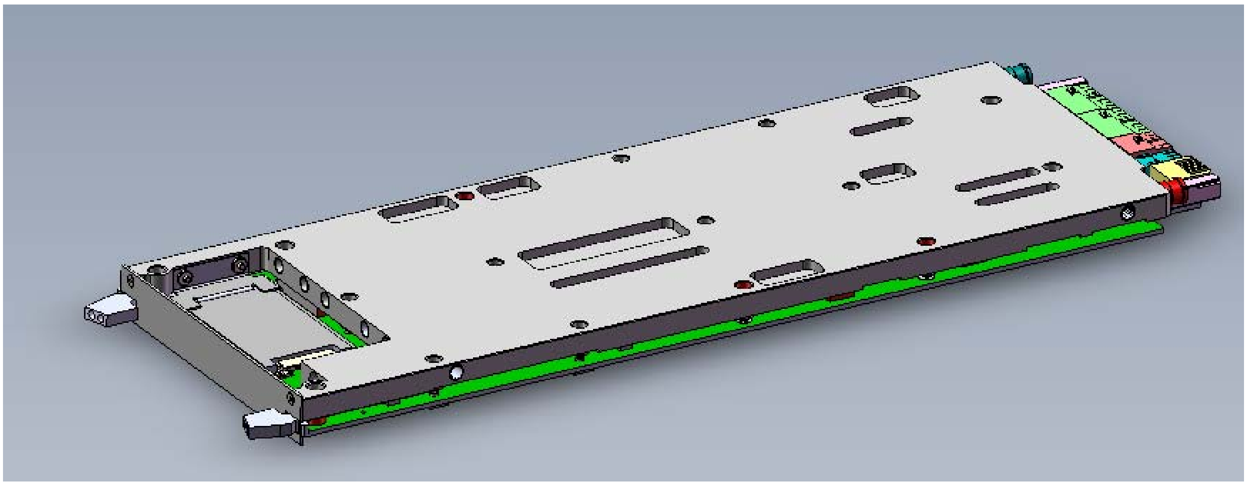

Figure 14 – Power Node board concept, with cooling heat sinks

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

The software development for the Power Node will be 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 will be “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 will be 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)

4.1.1.2 Power Node hardware (Radio, Power, CPU, Interfaces, Sensing, extras (FPGA etc.))

The Power Node is a High Performance Platform based on Nehalem/Wesmare 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 will consist 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) will be 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.

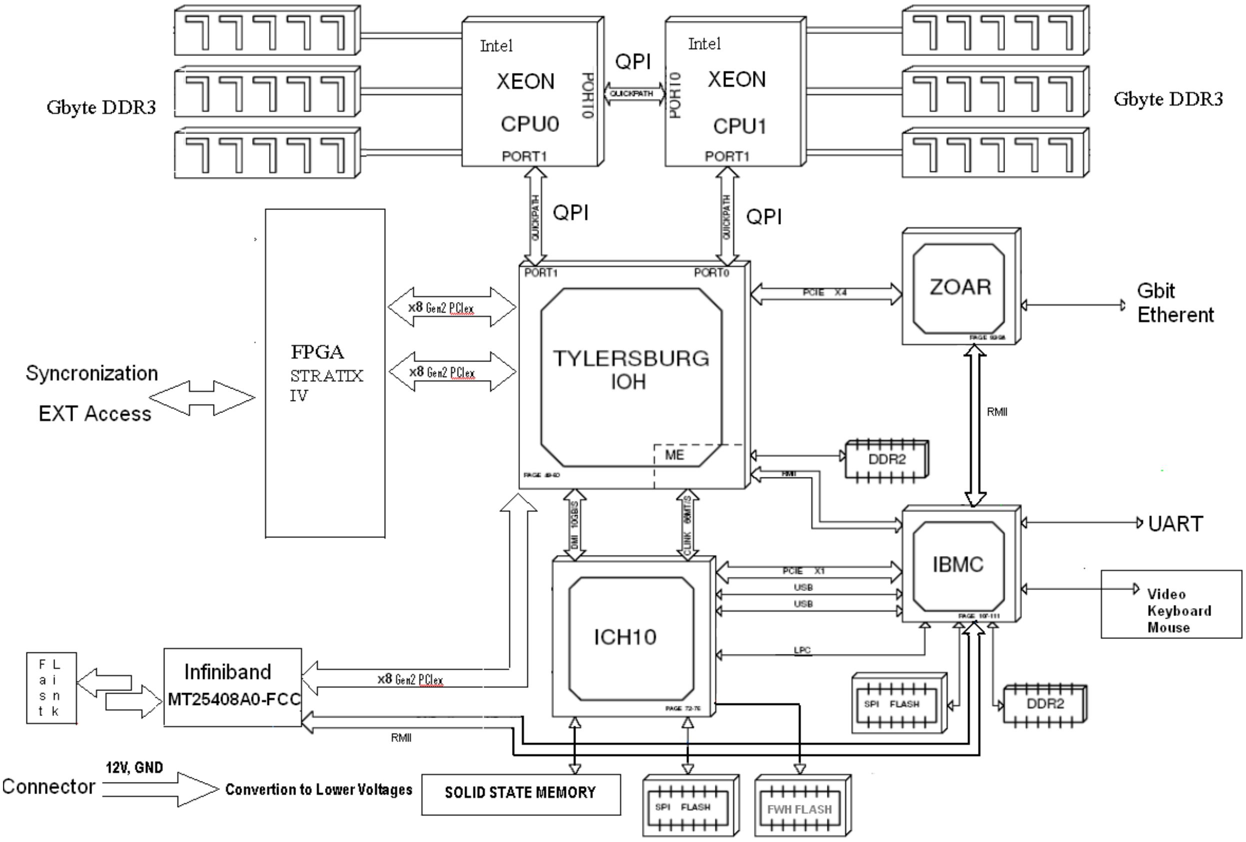

Figure 15 – Power Node architecture: high level description

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
- more in general, the hardware supports the Intel AES-NI technology

The Power Node architecture has been conceived thinking also “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 the FPGA implementation, the torus network processor 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).

4.1.1.3 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.

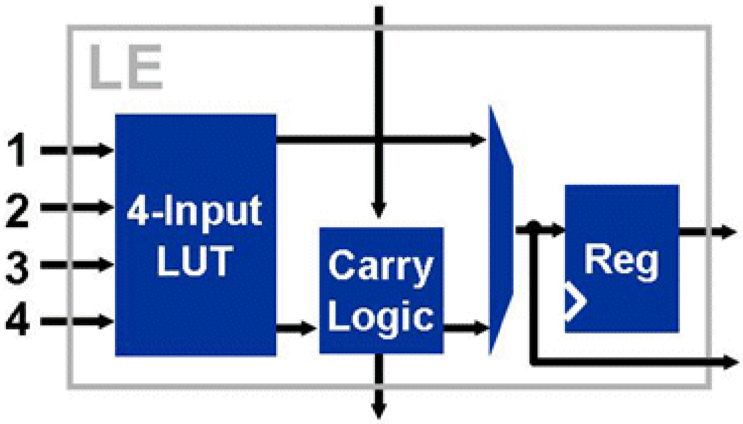

Figure 16 – internal structure of an FPGA logic element

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.

4.1.1.4 Technical Specifications

In terms of technical specifications, the Power Node will feature:

- 2 Intel Xeon 5570/5680 CPUs at 2.93/3.33GHz
- At least 6GB RAM 1333MHz DDR3
- Optional FPGA device, which allows implementation of:
  - ✓ Hardware accelerator features (on board co-processing)
  - ✓ Synchronization network for multi node mode
- Custom processing units
- Optional 80GB 1.8” SATA SSD
- Independent sensor network and monitoring system
- Connectivity via two additional debugging board that will bring the signal on standard connectors for an easier access by the user
- QDR Infiniband port
- LAN 10/100/1000 Interface
- VGA Analog Video output
- 2x USB 2.0 host interface

Physical Specifications:

- Physical dimensions: 166mm h x 25,4mm w x 500mm d
- Weight: 2.2 Kg (with cooling system)

Power Specifications:

- Power consumption: 350W typical (420W Max)
- Power Supply Voltage 12V

5 Non-FPGA Based Power Node [CWIN]

This document tries to highlight the work areas of the consortium members within the scope of this project. We have identified the following areas for SPD related work in pSHIELD:

- Secure firmware developments
- Implementation of security across heterogeneous platforms.

CWIN is prototyping a demonstrator consisting of nano-, micro- and personal nodes, the document also indicates the integration aspects of this demonstrator. The platform currently consists of two different personal nodes, a mobile phone and an embedded Linux platform. The embedded Linux platform connects to a GPS nano-sensor through a wired connection (USB), and has a wireless connection to a Sun SPOT device being able to measure accelerometer, temperature, and light.

The implementation uses the Sun SPOT sensor platform as micro node. Sun SPOT is a useful platform for developing and prototyping application for sensor network and embedded system. Sun SPOT is suitable for application areas such as robotics, surveillance and tracking. The main units are Sunspot devices with embedded sensors and base station. Each Sunspot has a so-called eSPOT with battery, while the base station is not equipped with battery and must be powered from the host computer via an USB cable. The Sunspot does not need to run any operating system, it needs only JVM that runs on bare metal, and executes directly out of ash memory. Stack-boards composed of specific sensors and actuators such as accelerometers, light sensor and temperature sensor.

The integrated sensors include as integrated Sensors:

- Temperature Sensor: Chip-type is ADT7411 sensor that measures temperature with ADC. ADC is integrated into the Demo, and can measure temperatures between -40°C to +125°C.

- Accelerometer sensor: 3-axis accelerometer of the type LIS3L02AQ, designed by ST Micro Systems and is in eDemo Board. This sensor can measure the x-axis, y-axis and z-axis in the direction up and down with the value either ±2G or ±6G. When the Sunspot is at rest, it measures x = y = 0 and z = 1G.

- Light sensor is of the type TPS851, designed by Toshiba. The sensor can measure the voltage between 0.1V (dark) - 4.3V (light), and converts the voltage to the brightness of Luminance (lx) 3.

These sensors are controlled through a VIA EPIA N700 board, which is a compact, low heat, power-efficient Nano-ITX board, ideal for compact industrial PCs and embedded automation devices. The board is integrated with the VIA VX800 media system processor, an all-in-one chipset solution that provides an extensive feature set while using less real state, helps to make the VIA EPIA N700 a superb choice for compact systems. It is based on Nano-ITX form factor (12cm x 12cm. VGA, USB, COM, Compact Flash (CF) and Gigabit network ports are provided on the board to help reduce system foot-print size and eradicate cluttered cabling for improved air-flow and enhanced stability in always-on systems. The VIA VX800 offers an integrated DirectX9 graphics core and excellent hardware accelerated video playback for MPEG-2, WMV9, VC1 video formats. An on-board VGA port is provided along with support for DVI and a multi-configuration 24-bit, dual channel LVDS transmitter, enabling display connection to embedded panels. The VIA EPIA N700 is equipped with a power-efficient 1.5GHz VIA C7, supports up to 2GB of DDR2 system memory and includes two onboard S-ATA connectors, USB 2.0, COM and Gigabit LAN ports. Expansion includes a Mini-PCI slot with an IDE port, additional COM and USB ports and PS/2 support available through pin-headers. The VIA EPIA N700 offers total system stability at extreme temperatures ranging from -20°C to 70°C, an ideal solution for our Norwegian rail use case to meet the extreme weather condition of Norway.

5.1 Potential SPD functionalities in the demonstrator [CWIN]

So far effort was given to integration through the third party an open platform of Telenor Objects. This integration has only limited considerations on security, privacy and dependability (SPD). However the communication with the Telenor Shepherd platform is secured using HTTPS protocol. The following scenarios can illustrate how the above implementation can be extended with the SPD aspects. The first two scenarios address the security aspect and the third scenario deals with the dependability situation.

Scenario 1: Only valid node can be connected to the system (illustrated in fig. 6). In this regard, we may use mobile phone as personal node and can only be integrated with the system if it is authenticated by the system. This will avoid getting communication and data from a fake node.

Scenario 2: Communication to 'new sensor' is only allowed if it is taking place at a pre-defined location. In this regard we assume that the location is a restricted place and only authorised user can access the premise. The system will get the location information from GPS sensor and if the location is same as the 'pre-defined' location, the system can communicate with the 'new sensor' installed on-board. Scenarios 2 can further secure the new sensor integration on top of scenario 1.

Scenario 3: As GPS reception is typically very poor inside the measurement vehicle, we needed to address location through a combined approach of two independent sensor. We extended the set-up with a mobile phone, which allows us position based on three methods: GPS, network and WLAN coverage. Taking into consideration the time-stamp of the equipment will open for a "dependable" positioning solution through a composition of sensor data from the embedded linux platform and the mobile phone.

The realization of the scenarios is subject to further discussion.

5.2 Integration towards Telecom systems [CWIN]

The main focus of the work in this area has been on the integration of the combined sensor platform into the telecom infrastructure of Telenor, the *Shepherd*® Platform.

Telenor, Norway have introduced a platform (named as Shepherd®) for interoperability and integration that supports communication between connected devices (nano and micro nodes) and makes them accessible from anywhere at anytime.

The Shepherd® is a platform for Connected Objects (COs) [32]. This means that any the pluggable component can be connected, and be integrated in Shepherd® platform as a Connected object (CO). Figure below depicts the overview of Shepherd® platform.

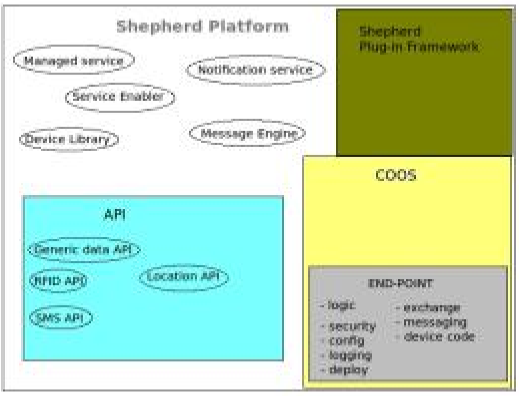

Figure 17 – Details of Telenor's Shepherd platform

The platform offers number of service including:

- Service Management for monitoring, device configuration, SLAs, and supporting.
- Service Enabler has a specific API that allows further access to other modules.
- Message Engine handles and secures the process of message flow, including capturing, processing,

routing and storage of data in an environment.

- Notification services that inform about the status of devices and applications.

- Device library consists of interfaces for tools and services recognition.

Connectivity with Shepherd® Platform

Shepherd® offers two methods for establishing connection. This includes: 1. HTTP Connection API - This mechanism establishes a direct connection to the Shepherd by using the HTTPS protocol. With this method, it requires the development of the HTTP API of the object. Shepherd accepts both methods POST and GET. When the connection is established, the Shepherd sends a response code back to that object as a confirmation of success or failure of reception. To be able to connect to the Shepherd, the "device object" is identified with an Application ID and an Object ID.

Connected Objects Operating System (COOS) - is an open source and has been written in Java. When using the COOS instance, the applications can connect to Shepherd in a secure, reliable and stable manner. In particular, it is important in this respect that eavesdropping by third parties is not possible when using COOS. Reliable in the sense that it is an M2M network, and communication between objects with COOS instance and the Shepherd will not be interrupted or delayed more than necessary. Thereby, ensuring a stable environment for the users and the applications. It requires therefore, developing an application using COOS, so this can apply to device so it can communicate with the Shepherd. From a programming perspective, "Connected Objects

Operating System (COOS) is an application distributed in a container so that it can enable data exchange between the object and the Shepherd. In COOS concept, every component that is integrated, and can be pluggable is called "Connected Object (CO). This means that a COOS instance can have multiple Connected Object, and each COOS instance carries its own distinctive character.

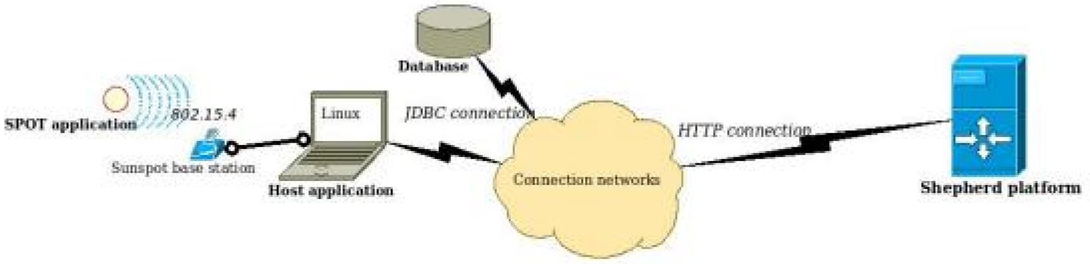

Figure 18 – Set-up of the communication from SunSPOT to the Shepherd platform

Figure given above presents the system overview. It shows the establishment of an intended two-way communication between Sun SPOT sensors and its base station, and also two-way communication between the embedded Linux system and the Shepherd Platform.

A Host application has been developed, that performs broadcasts every fifteen seconds. While, the spot application will detect the broadcasts every thirty seconds. But, it does not transmit the values to the base station after one minute has passed since the last envoy. When the values arrive, these will also be stored in cache. At the same time, the Host application sends out a request to Shepherd for receiving the values. The connection is opened until the application has received confirmation from the Shepherd of receipt. However, the values to be sent to Shepherd, only happens in every five minutes. The SPOT application is also designed to detect spot's battery level prior it using the wireless communication. If spot battery is either lower than -32 or greater than 32, then the MIDlet will be destroyed, terminated and a notification will be send to the concerned actors.

6 Enhanced Power Management [AS]

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 19). 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.

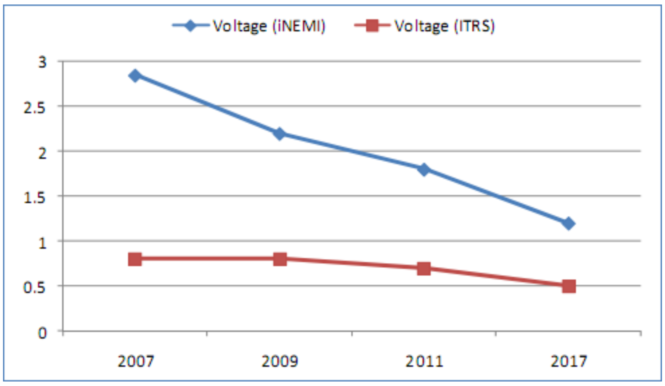

Figure 19 – ITRS-iNEMI (2010) System-to-Chip Supply voltage and threshold voltage trends

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.

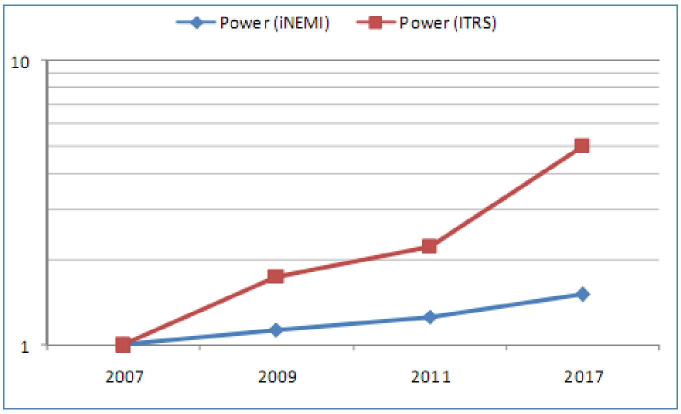

Figure 20 – ITRS-iNEMI (2010) System-to-Chip Power Comparison trends

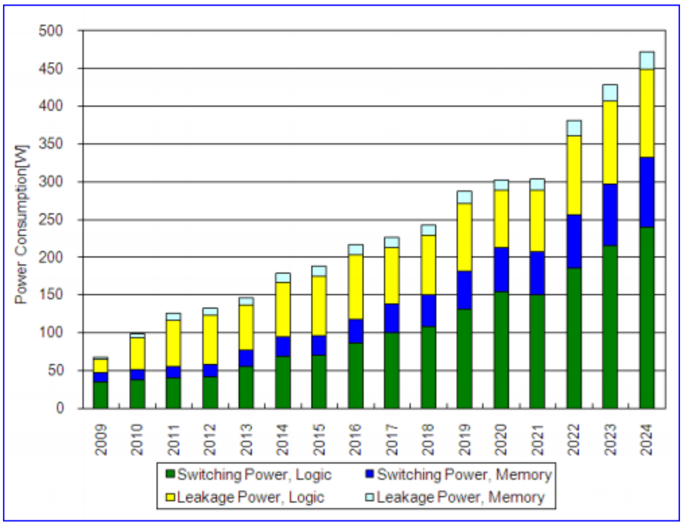

Figure 21 – ITRS projection for SOC Consumer Stationary Power Consumption Trends (2010)

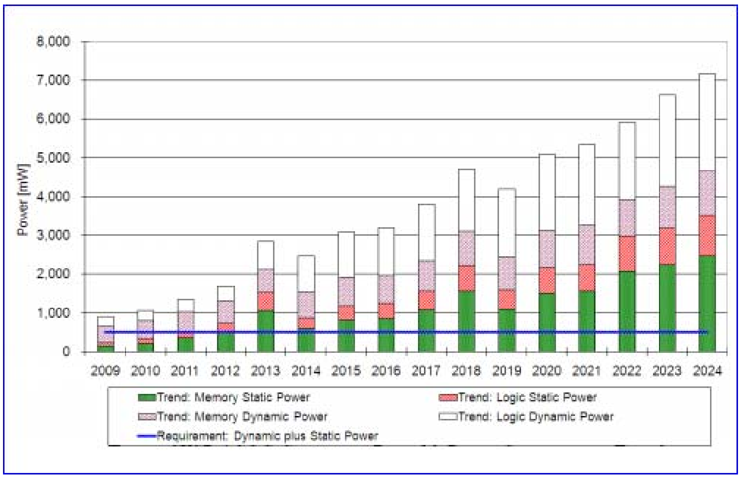

Figure 22 – ITRS projection for SOC Consumer Portable Power Consumption Trends (2010)

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

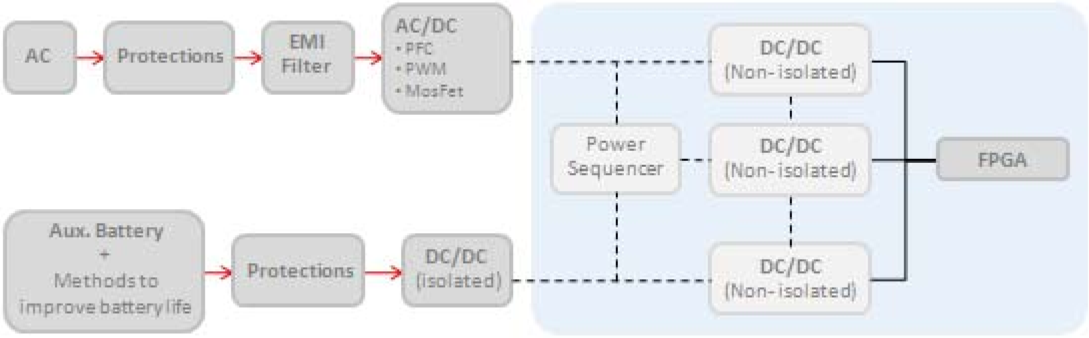

Figure 23 – Power supply components - General design

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.

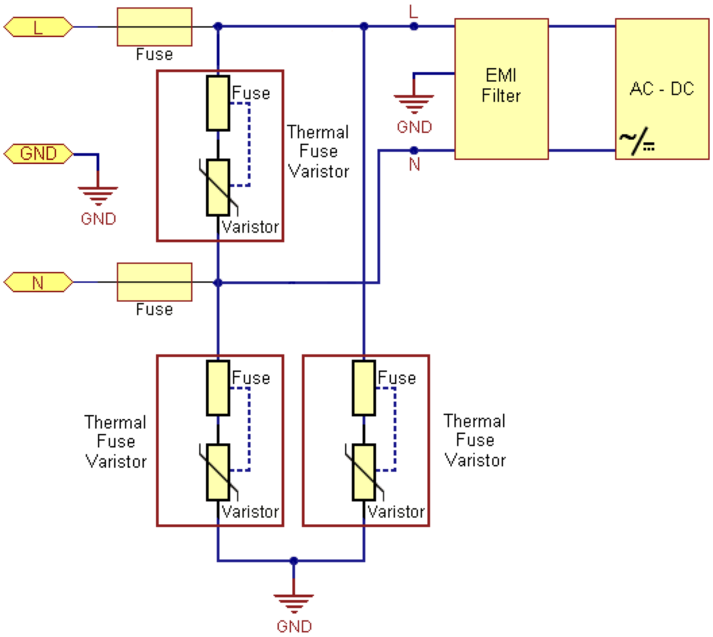

Figure 24 – Protection Board – Thermal Fuse Varistors

- 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.

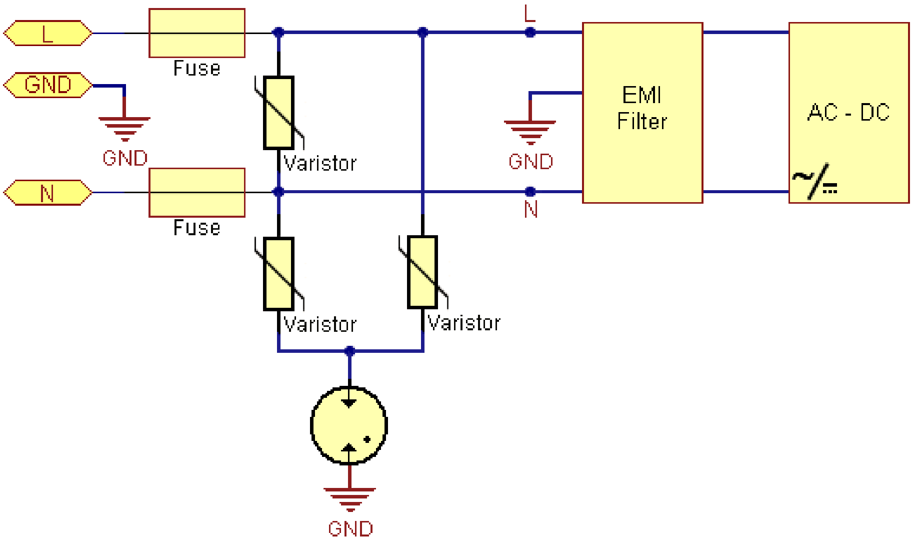

Figure 25 – Protection Board – Varistors and Gas Discharge

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.

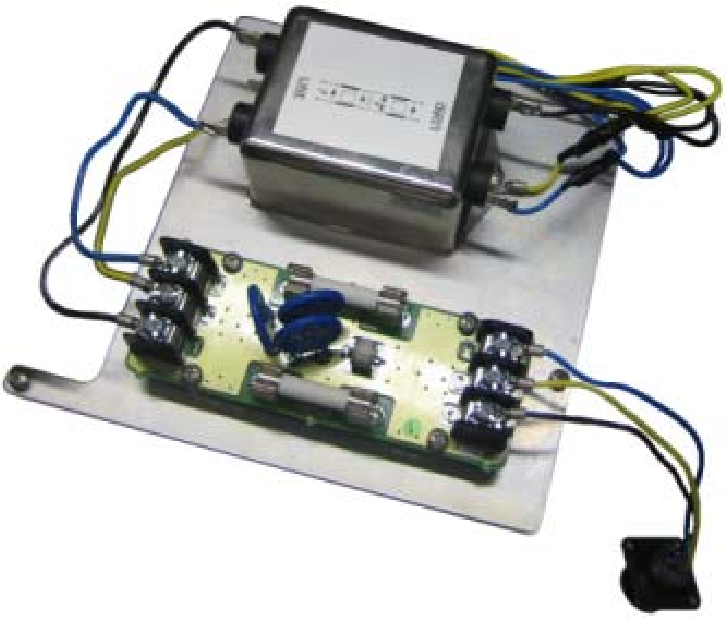

Figure 26 – Protection Board prototype with Varistors and Gas Discharge

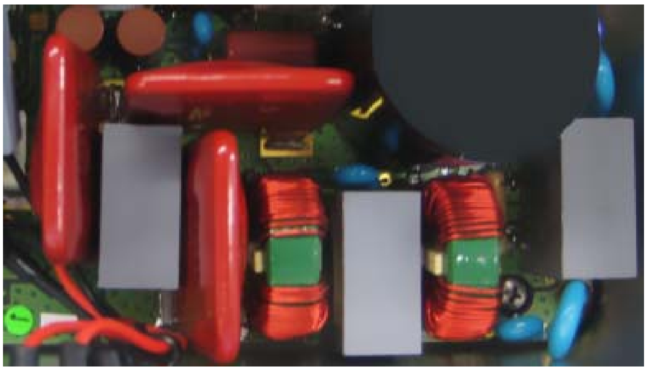

Figure 27 – Protection Board prototype with Thermal Fuse Varistors

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 28 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.

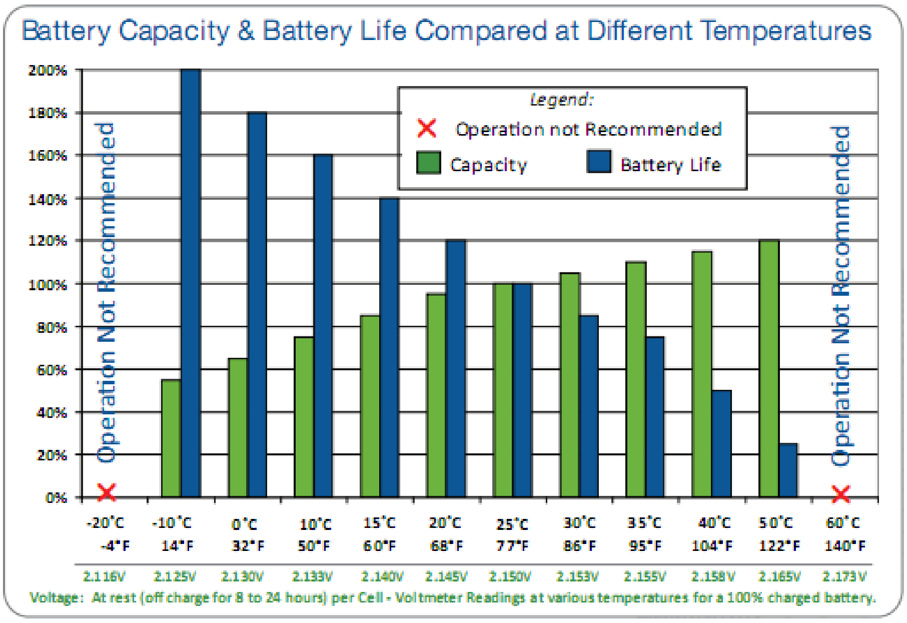

Figure 28 – Temperature effects on battery performance and Life by Discover® and Clean & Green™[2]

[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.

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"