A FAST INTERLOCK DETECTION SYSTEM FOR HIGH-POWER SWITCH PROTECTION

P. Van Trappen*, S. Uyttenhove, E. Carlier, CERN, Geneva, Switzerland

* pieter.van.trappen@cern.ch

ICALEPCS 2015
International Conference on Accelerator & Large Experimental Physics Control Systems
17-23 OCTOBER 2015 MCEC MELBOURNE

ABSTRACT

Fast pulsed kicker magnet systems are powered by high-voltage and high-current pulse generators with adjustable pulse length and amplitude. To deliver this power, fast high-voltage switches such as thyratrons and GTOs are used to control the fast discharge of pre-stored energy. To protect the machine and the generator itself against internal failures of these switches several types of fast interlocks systems are used at TE-ABT (CERN Technology department, Accelerator Beam Transfer). To get rid of this heterogeneous situation, a modular digital Fast Interlock Detection System (FIDS) has been developed in order to replace the existing fast interlocks systems. In addition to the existing functionality, the FIDS system will offer new functionalities such as extended flexibility, improved modularity, increased surveillance and diagnostics, contemporary communication protocols and automated card parametrization. A Xilinx Zynq®-7000 SoC has been selected for implementation of the required functionalities so that the FPGA (Field Programmable Gate Array) can hold the fast detection and interlocking logic while the ARM® processors allow for a flexible integration in CERN's Front-End Software Architecture (FESA) framework, advanced diagnostics and automated self-parametrization.

SYSTEM DESCRIPTION

A kicker pulse generator generally consists of a stored charge (PFN / PFL) which is discharged into a transmission line and kicker magnet so that the magnetic field deflects the circulating beam. High-voltage tube or semiconductor switches are used to switch that charge. To protect the generators and the machine, the following fast interlock events need to be detected:

1. Normal conduction
2. Missing conduction: a trigger occurs but the magnet is not pulsed
3. Erratic conduction: a spontaneous conduction without trigger
4. Short-circuit: possible in connectors, transmission line or magnet
5. Particular cases depending on the installation, e.g. reverse voltage
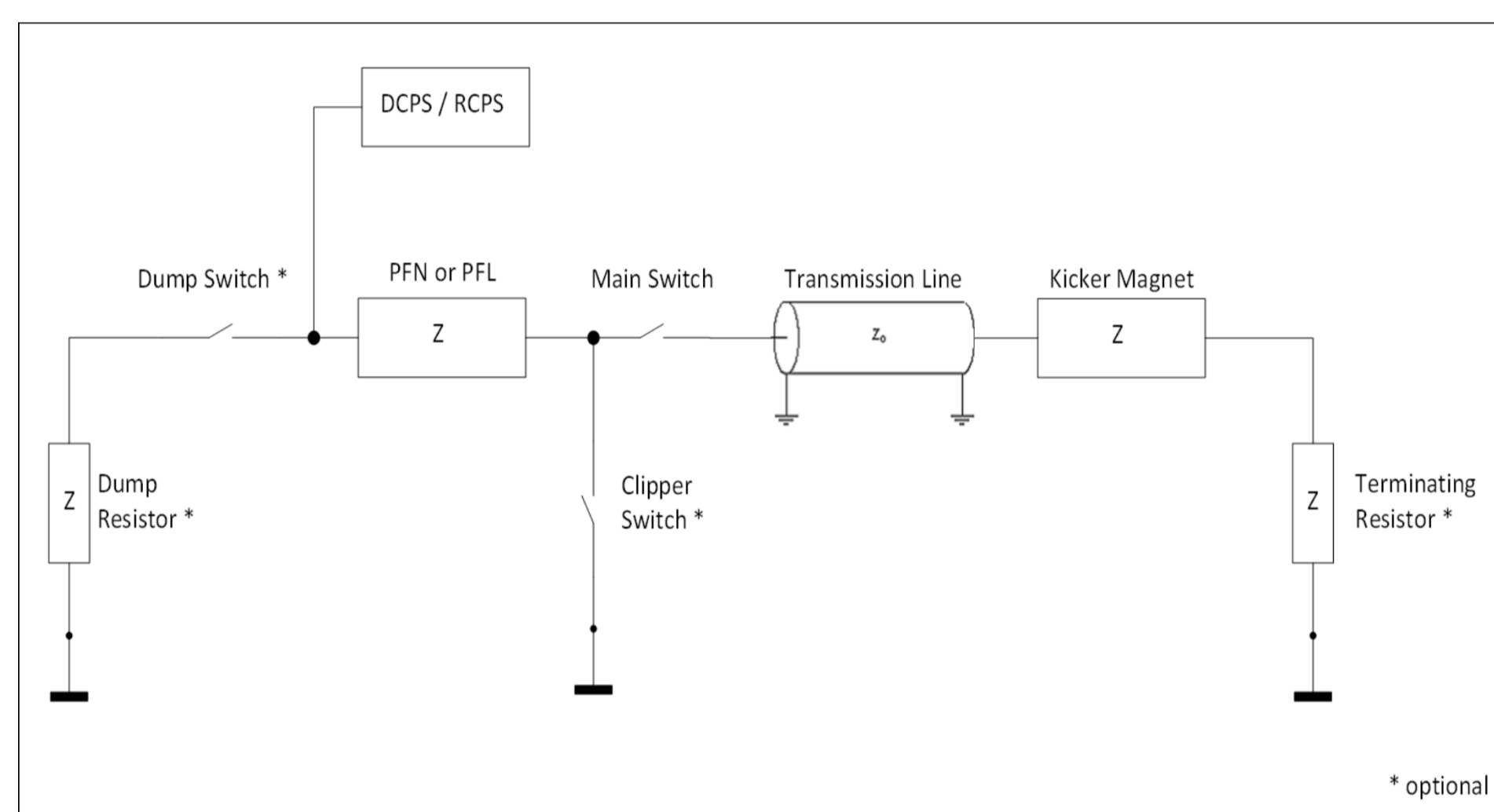
Simplified kicker pulse generator

Fast Interlocks Detection System (FIDS) principle

The PFN voltage is sampled by an ADC and after digital factor multiplication and offset addition this value is used as a comparator reference. The comparators are then used to digitalise (1 bit) analog input signals from voltage and current measurements with a bandwidth of 100 MHz and low latency. Based on the relative time difference between pickup and trigger signals, a fast interlock event is detected and an corrective trigger signal can be issued within 200 ns.

PROTOTYPE TEST RESULTS
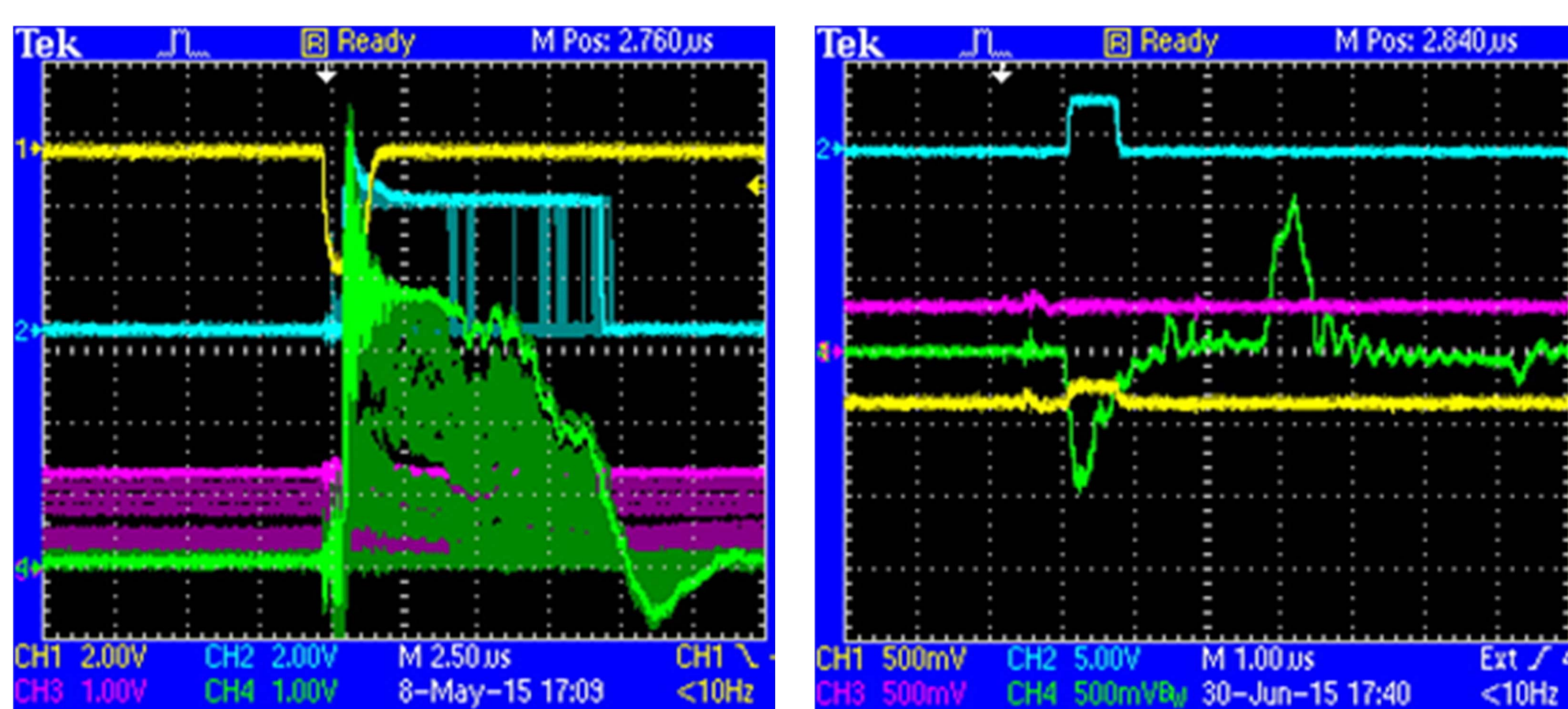
Ch1 (yellow): trigger, Ch2 (blue): comp. output, Ch3 (magenta): comp. ref., Ch4 (green): switch pick-up

Ch1 (yellow): comp. ref, Ch2 (blue): comp. output, Ch3 (magenta): PFN ref+, Ch4 (green): switch pick-up

SPS Beam Dump System tests, 50 kV DCPS installation. Pulsed over full dynamic range with scope persistency enabled. The pickup is a thyratron current transformer.

PS Booster Transfer Kicker, -60 kV RCPS installation. The dump switch is traversed by positive and inverse currents which requires a fully bipolar setup. External comparator hysteresis added.

PROTOTYPE

A prototype has been built to verify the functional operation and to test the chosen technologies such as the Xilinx SoC that embeds an FPGA, ARM cores and ADC in one package. A MicroZed™ development board has been used where analog filtering and signal conditioning was added by means of a soldered prototyping board.
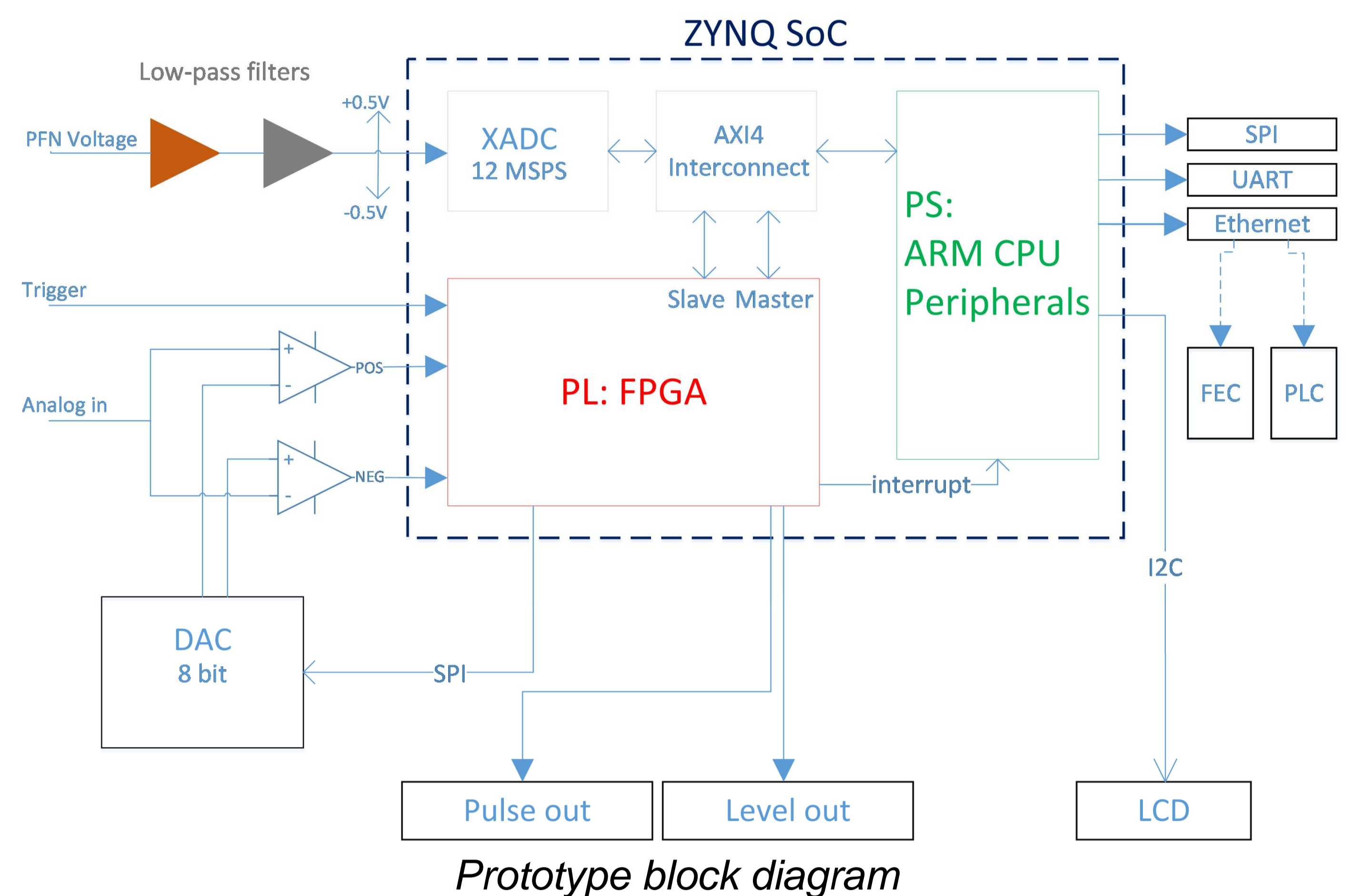
Prototype block diagram

Gateware and software

The FPGA **gateware** is written in VHDL-2008 and PSL is used for verification. The fastest clock domain is clocked at 250 MHz to measure the pulse length with a 4 ns precision, using embedded DSP48 slices for the adders.

Two **software** approaches have been tested: bare-metal embedded C code and embedded GNU/Linux (PetaLinux). Both approaches were able to act on the FPGA to CPU interrupts to output event information. An AXI4-Lite bus is used internally between the ADC, CPU and FPGA. Embedded peripherals such as UART, I^2C and Gbit Ethernet are used for user interaction.

FULL DESIGN

Parameter	Value
2 FMC LPC slots	VADJ fixed to 2.5V for slot-1 and 1.8V for slot-2
6 analogue inputs	2 MSPS ADC, 1 MHz BW, 12-bit, lemo-00, ±10 V input
4 isolated outputs	9-pin D-Sub male connector
4 outputs	TTL level, rise-time <10 ns
4 high-power outputs	15V, rise-time 20 ns, 1A peak output
user-button	momentary NO push-button
reset-button	momentary NO push-button
8 bi-colour user LEDs	4 FPGA- and 4 CPU-controlled
Serial interface	I2C bus for connecting external LCDs and I/O expanders
JTAG header	Xilinx Platform Cable style
SPI Flash	Bootable, 2x 128 Mbit QSPI
SD card slot	For booting, OS mounting and persistent storage
Fail-safe functionality	FPGA and CPU watchdogs; voltage supervision; pulse output feedback
2 Ethernet RJ45 ports	Ethernet switch; 1 SoC Ethernet MAC; 10/100 Mbps

FIDS Carrier specification

Parameter	Value
Inputs	10 analog comparators
Outputs	8 TTL <10 ns rise time
Number of comparators	20 (2 per input channel)
Comparator references	20 (1 per comparator)
Comparator input bandwidth	1 GHz
Comparator input levels	± 5 V
Comparators to FPGA	LVDS
FMC to carrier interface	Low Pin Count (LPC) connector
Input impedance	High-Z or 50 Ohm
Programmable threshold	DAC with 5mV precision
DAC resolution	12 bit
DAC Sampling rate	166 kHz

FIDS FMC DIO 10i 8o FMC specification

To allow for flexibility and independency regarding form factor and CPU bus the ANSI/VITA 57.1 FPGA Mezzanine Card (FMC) approach has been chosen. A generalised FIDS FMC DIO 10i 8o card with fast comparators is currently being developed. It can be employed on existing and future FMC carriers such as the Open Hardware SVEC.

A FIDS Carrier with two FMC slots is being developed as well to deal with the additional analog inputs and fail-safe functional requirements.

Chosen form factor, example from ohwr.org