NATIONAL CENTRE FOR NUCLEAR RESEARCH ŚWIERK

Description:	This document describes the design of the RTM Carrier designed for the tests of the ESS LLRF control system
Title:	**Critical Design Report for the RTM Carrier**

Authors:	Jarosław Szewiński
Co Authors:	Marcin Gosk, Dominik Rybka

Version:	0.01
Date:	20.04.2018

1. Introduction

1.1 Concept of the board

Architecture of the ESS LLRF system is based on the MTCA.4 standard, where several devices (actually LO generation and Piezo control) will be implemented as a RTM units. To control those devices an AMC board – the “RTM Carrier” is needed.

In the ESS LLRF system concept, the RTM devices will be the functional units dedicated for particular tasks, such as LO generation or piezo resonance control. AMC boards shall provide FPGA device for data processing and communication. Such architecture allows easy upgrade of the FPGA technology by replacing AMC boards when used technology will become obsolete, without affecting the functional RTMs.

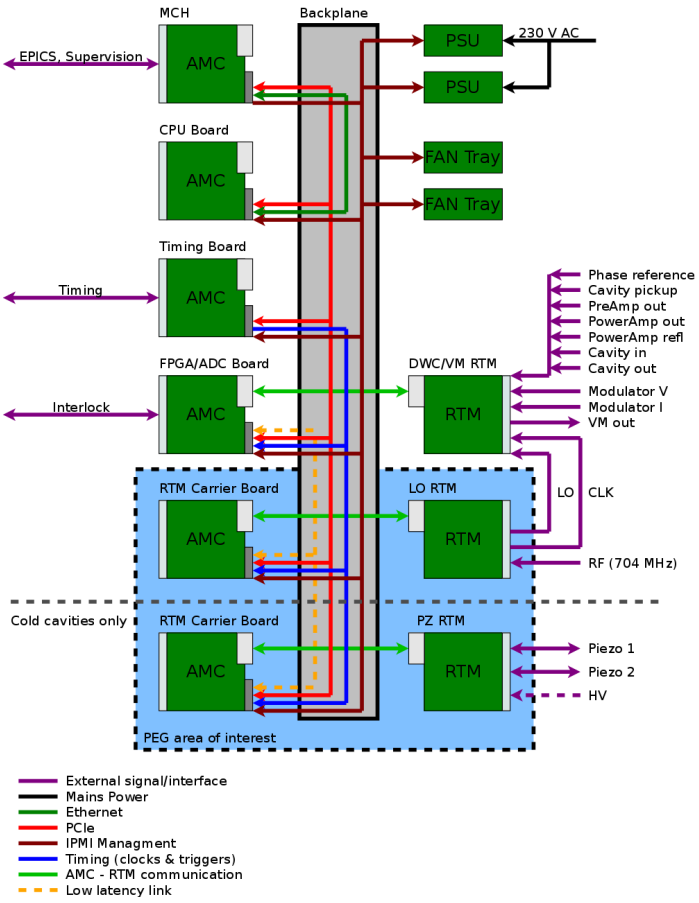

Figure 1.ESS LLRF System

Figure 1 shows where in the ESS LLRF system, RTM Carrier boards are used.

In the described concept, the AMC board for supporting RTMs shall provide minimal functionality that allows RTMs to operate, and this is base of RTM Carrier device concept. The general required

functionality of the board is following:

- Communication with the RTM via ZONE 3
- Providing power to the RTM units
- Communication with the other devices using PCI-Express on the MTCA.4 backplane
- Data processing in the FPGA
- Fulfilling all the requirements for the AMC board defined in the MTCA.4 standard

At the time when architecture of the ESS LLRF system architecture was defined, there was no low cost MTCA.4 board fulfilling listed above goals. Considered alternative DAMC-FMC25 was expensive in comparison with needed functionality, because it had many features not needed in described application. Also it had unsupported in the latest development software (Xilinx Vivado) FPGA chip – Virtex-5.

Designing RTM carrier gives ability to create low cost device that fulfills exact needs of the ESS LLRF system, where estimated savings, in comparison with DAMC-FMC25, will easily cover the development cost.

1.2 Board interfaces

Described device have following interfaces:

- MTCA.4 ZONE 3 – I/O communication with the RTM
- MTCA.4 Backplane connectivity:
  - PCI-Express on AMC ports 4-7
  - Low-Latency Links for direct board-to-board communication, AMC ports 12-15
  - MLVDS clocks, triggers and interlocks on AMC ports 17-20
  - Telecom clocks (TCLKA, TCLKB)
  - MTCA.4 management signals (IPMB, Geographical Address, PS0#, PS1#, ENABLE#, AMC JTAG)
- Front panel:
  - External clock input
  - External clock output
  - Diagnostic connectors (USB-Serial, programming cable connectors)

1.3 RTM Carrier requirements

Based on description of the board concept above, the following requirements can be described:

1. Provide required voltages for the RTM (12V for payload power and 3.3V for management

power)

a) Provide as large as possible amount of power for the RTM (especially for piezo driver)
b) Provide as high as possible current for specified time after start-up (in-rush current) for the RTM (especially for RTM)

2. Provide data transfer between MTCA.4 backplane and the RTM
   a) Zone 3 I/O configuration compatible with DESY D1.0 Recommendation
      i. All pins on the Zone 3 connector shall be connected to the FPGA
   b) Provide FPGA resources for implementing RTM-specific data exchange algorithms (SPI communication with the RTM, etc.)
   c) Provide access to RTM resources via the PCIe interface on the the MTCA backplane (ports 4-7)
   d) Provide connectivity between RTM and MTCA.4 extensions (MLVDS, ports 17-20)
      i. Provide clocks and triggers from the MTCA.4 backplane to the RTM
      ii. Provide interlock signals from the MTCA backplane to the RTM
      iii. Provide interlock signals from the RTM to the MTCA backplane
3. Provide MTCA management for the RTM
   a) Provide I2C connection from the MMC
   b) Provide support for accessing RTM management resources via I2C: (MTCA required LEDs, Hot-Swap handle, sensors, etc.)
   c) Represent the RTM resources (identification, sensors) in the IPMI records to the MCH
   d) Provide power control for the RTM
   e) Provide standard sensors on the AMC board
4. Provide clocks signals interconnect
   a) Provide MTCA clocks (TCLKA, TCLKB) for the RTM and for the FPGA
   b) Provide External clock source for the RTM and for the FPGA
   c) Provide on-board programmable clock generator for the RTM and for the FPGA
   d) Provide RTM clock for the FPGA
   e) Provide External clock output for monitoring selected on-board clock
   f) Provide local (non-programmable, always enabled) clock for the FPGA
   g) Provide dedicated clock infrastructure for PCIe interface
5. Provide diagnostics and support for out-of-crate board debugging
   a) Provide external power supply connector (12V, optionally 3.3V management power)
   b) Dual channel USB-Serial interface to FPGA and MMC
   c) LEDs on the front panel
   d) JTAG interface for FPGA and for the RTM

6. Board should be low cost
    a) Price goal to be below 1000 Euro in mass production
    b) Board should use latest technology low cost FPGA device

2. RTM Carrier hardware design

Based on conceptual design and requirements described above, the following hardware design presented in fig.2 has been created.

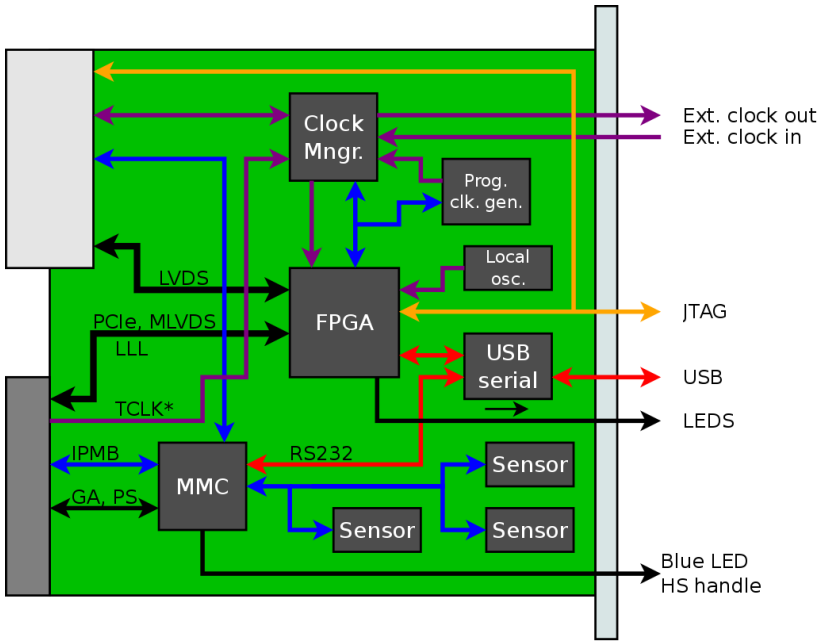

Figure 2. Functional block diagram

2.1 FPGA

The core component of the board is FPGA device. In this project, **Xilinx Artix-7** device has been chosen. This decision was made due to the following reasons:

- **Vendor**: Xilinx devices are used widely in accelerator control, as well as in other areas of experimental physics. Those devices has been used in X-FEL LLRF system, they are proven to be reliable and PEG member has experience with this technology, such as PCB design for Xilinx FPGAs, software tools knowledge, programming techniques, etc.
- **Gigabit transceivers** (MGT, RocketIO) – PCI-Express and Low-Latency Links requires gigabit transceivers, due to this reason other Xilinx low cost FPGA families, such as Spartan-7, can not be used.
- **Low Cost** – Artix-7 is a Xilinx low cost FPGA device with gigabit transceivers
- **Latest architecture** – Artix-7 is a part of Xilinx 7-th Generation FPGA devices (“7 Series”), and in contrast with 6-generation and older devices, it is supported by most recent Xilinx development tools – **Vivado**.

Artix-7 FPGA devices family contains several devices in different packages, where different devices are foot-print compatible. In the described project, **FBG484** footprint for the FPGA has been chosen. This footprint allows assembly of the devices listed in the table 1:

Part Number	Logic Cells	Slices	CLB Flip-Flops	Total Block RAM (Kb)	DSP Slices	Max. single ended IOs (6.6 Gb/s GTPs)
XC7A15T	16,640	2,600	20,800	900	45	250 (4)
XC7A35T	33,280	5,200	41,600	1800	90	250 (4)
XC7A50T	52,160	8,150	65,200	2700	120	250 (4)
XC7A75T	75,520	11,800	94,400	3780	180	285 (4)
XC7A100T	101,440	15,850	126,800	4860	240	285 (4)
XC7A200T	215,360	33,650	269,200	13140	740	285 (4)

Table 1. Available FPGA resources

In the prototype series, the largest device **XC7A200T-1FBG484C** has been mounted for convenience of firmware development. In the later production series smaller and cheaper devices can be mounted if final firmware resource utilization will allow that.

FPGA Configuration

FPGA configuration process is controlled by the Module Management Controller (MMC) device. The following configuration modes are available:

- Master SPI – FPGA loads firmware from SPI flash by itself
- Slave Serial – MMC can disable SPI Flash ans push bitstream to FPGA
- JTAG – fail-safe configuration mode using external programming cable

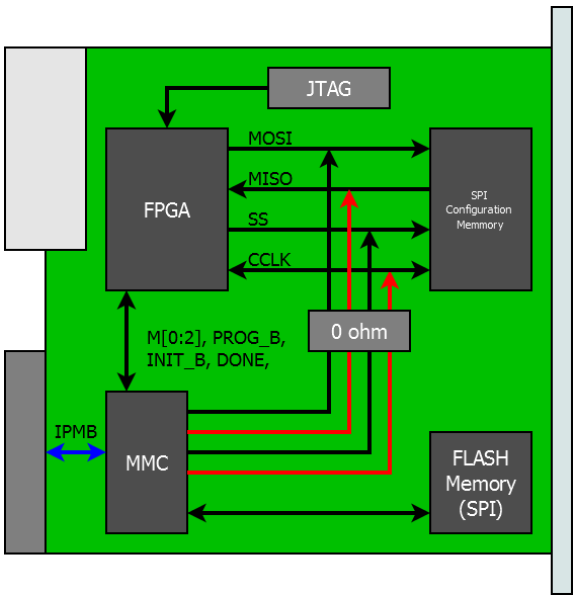

Figure 3. General FPGA configuration scheme

“Master SPI” and “Slave Serial” modes use the same configuration pins: CCLK and MOSI/D0. In Master SPI mode CCLK is driven by FPGA and MOSI/D0 by SPI flash memory, and in Slave Serial mode, both lines will be driven by MMC. To have two different configuration sources, (SPI Flash and MMC), mentioned lines are hard-wired on the PCB to all three devices (FPGA, MMC, SPI Flash). To make proposed solution working, MMC has to control the configuration process, which includes selecting proper configuration mode and enabling or disabling requested devices.
MMC may disable SPI Flash from driving MOSI/D0 by selecting any FPGA configuration mode except SPI and BPI, in such case pulled-up SPI Flash Chip Select (FCS_B) becomes high-impedance. MMC can disable FPGA from driving CCLK by selecting any slave configuration mode. Disabling FPGA allows MMC to update the SPI flash.

Proposed solution use minimal resources (PCB routing) to achieve following goals:

- Allow FPGA to load bitstream directly from SPI flash memory (configuration mode preferred by Vendor)
- Allow FPGA to update by itself the SPI flash during runtime; this allows “remote firmware upgrade” using user data transfer interface – PCI-Express.
- Allow MMC to load bitstream directly to FPGA; MMC may handle several firmware revisions, and select which one shall be used. This also provides the “remote firmware update” via MMC feature
- Allow MMC to update SPI flash
- Allow MMC to readback the SPI flash

In case of unexpected problems, zero-ohm resistors has been place on the configuration lines, allowing selection SPI flash or MMC (slave serial) mode only.

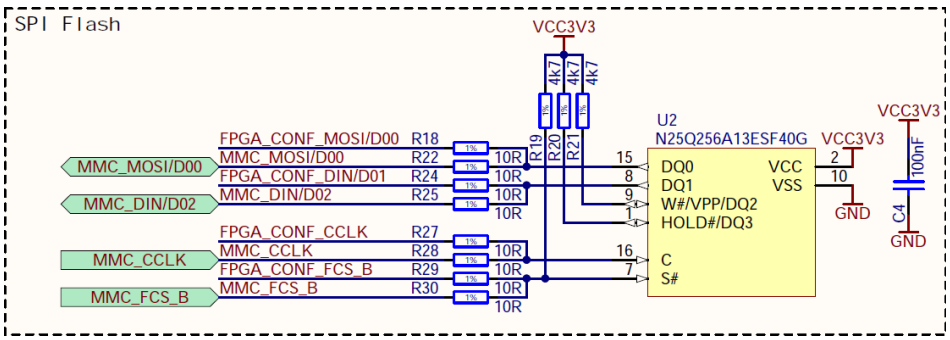

Figure 4. SPI configuration flash memory and signals routing

As a SPI flash memory **N25Q256A13ESF40G** device, has been selected. This is commonly used memory for Xilinx FPGAs configuration, and this particular device allows to keep bitstream for biggest Artrix-7 supported by RTM Carrier. This memory is supported by both ISE and Vivado development tools, and it is compatible witch serial configuration modes of the 7-Series Xilinx FPGAs.

JTAG

For development and in case of FPGA troubleshooting, JTAG interface for FPGA is provided. Described design has two JTAG slave devices:

- FPGA device
- RTM Interface (may contain zero, one or more devices in chain)

and two JTAG master interfaces:

- Xilinx JTAG Connector
- AMC JTAG interface

For robust and easy operation, an active JTAG distribution has been designed. Designed solution **does not require any action from user**, such as setting jumpers, to configure JTAG chain properly. By default, master interface for JTAG chain is AMC JTAG interface. Whenever Xilinx programming cable is plugged into front panel FPGA JTAG connector, it takes over the JTAG control using active switch, regardless if the board is in the MTCA chassis or if it is running out-of-crate in standalone mode. It is always safe to connect the Xilinx Programming Cable to the board, but in such case AMC JTAG will be disabled.
JTAG Chain contains FPGA and it is forwarded to the RTM, which may contain additional devices. RTM Carrier detects presence of the RTM unit, and enables/disables forwarding JTAG to the RTM. TDI and TDO lines can the forwarded to RTM or looped-back by an active switch, depending on the RTM presence. TMS and TCK lines are provided to the RTM via buffer, so in case when RTM is not present, signals reflected from high impedance will not affect JTAG driver nor FPGA in the chain. Attaching and detaching RTM units does not require any action to configure JTAG chain.

2.2 Board interfaces

RTM Carrier has following interfaces:

PCI-Express

Selected FPGA devices family has 4 MGT interfaces, and due to requirement of having support for Low-Latency Links (LLL), not all MGT can be used for PCI-Express. As a compromise, 2 MGTs has been assigned for PCI-Express, and 2 for LLL, resulting in having PCI-Express x2 (Gen2) on the AMC ports 4 and 5.

Low-Latencty Links (LLL)

This interface is for fast board-to-board communication over the MTCA backplane. As it was mentioned above, 2 of 4 MGTs has been assigned to LLL. Board to board communication has been foreseen on AMC ports 12-15. But due to the limitation of MGT amount (2), all for ports are covered using two 2:1 switches, having AMC ports 12 and 13 on one MGT, and AMC ports 14 or 15 on another MGT. Communication parameters strictly depends on the FPGA configuration, and it must be compatible with other device in the crate.

Device chosen as a LLL switch is **AD8153ACPZ**, it is I2C controlled single mux/demux for gigabit links. Using those two switches allows to have all LLL Covered on the MTCA backplane, but due to described limitations, not all LLL connections are possible at the same time.

MTCA.4 Zone 3

This is interface for communication with RTMs. Zone 3 is implemented according to DESY D1.0 recommendation, and it is fully covered, providing 96 LVDS pairs and supporting clock signals.

MLVDS

This interface provides general purpose synchronization signals, such as triggers or interlock signals on AMC ports from 17 to 20. Each signal can operate in both directions, board can read it or drive it on the MTCA backplane. For implementation of MLVDS bus, **DS91M040TSQE/NOPB** devices has been used – low cost quad MLVDS transceiver with performance fulfilling MTCA standard requirements.

Telecom clocks (TCLKA, TCLKB)

Board uses available on the MTCA backplane telecom clocks, TCLKA and TCLKB. These signals can be delivered to RTM and FPGA.

MTCA.4 management signals

Except data transfer interfaces such as PCI-Express, board supports management signals available on the MTCA backplane, such as: IPMB bus (I2C), Geographical Address, PS0#, PS1#, ENABLE#, and AMC JTAG.

Clock input and output

Board provides external TTL clock input and TTL clock output on the front panel.

Debug interfaces

For the purpose of diagnostics, board is equipped in several debug interfaces. Board it has dual USB-Serial device, where one channel is connected to the MMC system serial port, and other channel is connected to the FPGA. It provides user ability to create simple communication interface, which is not dependent on other infrastructure, especially MTCA. It can be useful for accessing FPGA in case of MTCA communication problems, or during standalone (out-of-crate) run.

Except the USB transceiver for serial ports, there is also another micro USB connector on the front panel, which is directly connected to the USB interface of the MMC. This second interface may utilize the additional functionality available in MMC, such as acting as dedicated USB device, like processor programming interface, etc.

Except USB, board has on the front panel dedicated JTAG connectors of Xilinx FPGA and RTM (if any), and MMC MCU dedicated JTAG Connector.

2.3 Clock distribution

The clock distribution scheme is shown in the picture below

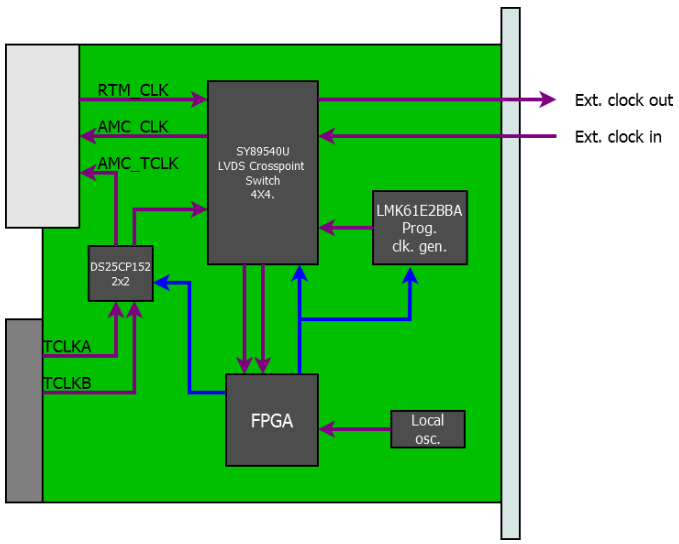

Figure 5. Clock distribution

The main task for the clock distribution is to support clocks foreseen in DESY D1.0 recommendation for the RTM:

- RTM clock input
- RTM output clock
- AMC telecom clock to RTM.

Features of presented solution:

- Clock from RTM can be delivered to the FPGA and to front panel clock output
- External clock can be delivered to FPGA and to RTM
- Each of 4x4 cross-point switch input can be routed to 3 destinations: FPGA clock capable pin, external output via lvds-to-ttl buffer, or RTM clock input.
- Both telecom clocks can be delivered to FPGA and RTM through 2x2 cross-switch, but one signal cannot go to both receivers - TCLKA may go to RTM and TCLKB to FPGA (through cross-switch), or TCLKB may go to RTM and TCLKA to FPGA (through cross-switch)
- There is programmable clock generator, to have flexible clock source
- FPGA has local always enabled clock source, to bootstrap and configure the rest of clocking infrastructure, such as cross switch or programmable clock generator
  External clocks are connected via lvds to TTL buffers (fig 7.).

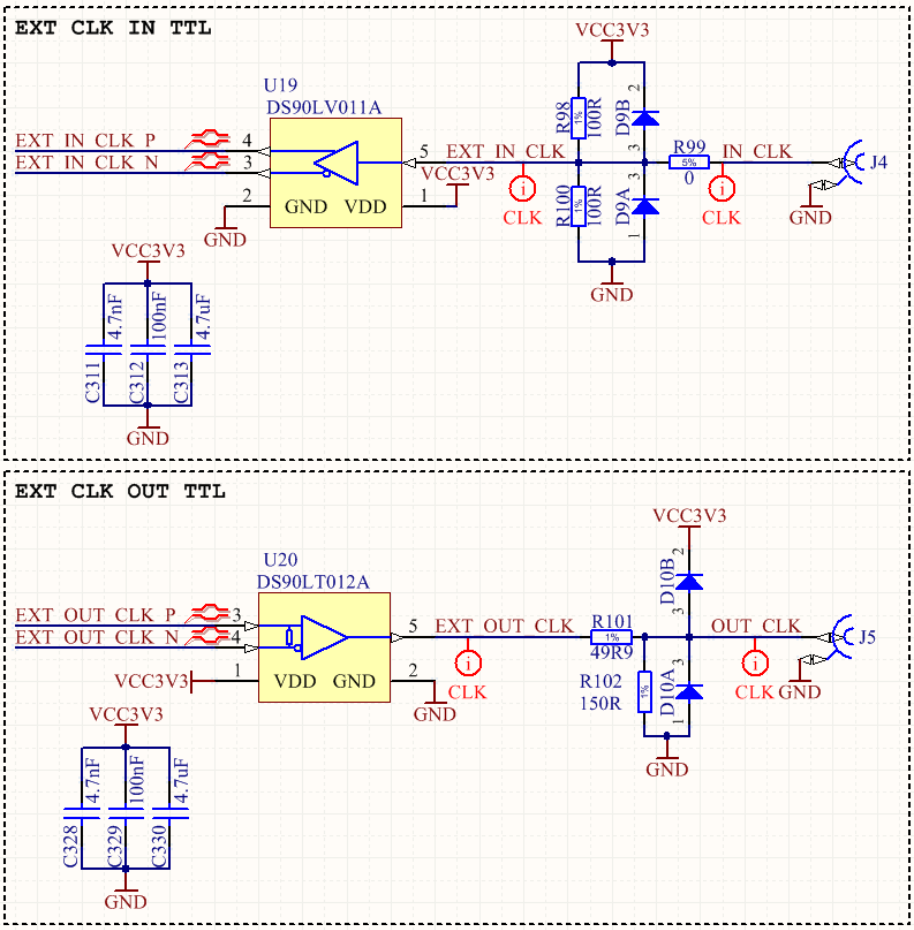

Figure 6. External Clock input and output

Main clock infrastructure components:

- **DS25CP152** - small footprint 2x2 LVDS cross-point switch designed for backplane acquiring signals, it has low additive jitter (max. up to 1 ps), which allows to use this device as clock switch
- **SY89540U** - small footprint 4x4 LVDS cross-point switch designed for high data rates with good channel-to-channel crosstalk performance, it has very low additive jitter (<0.1 ps), which allows to use this device as clock switch
- **LMK61E2BBA-SIAT** - programmable LVDS oscillator witch internal EEPROM. This device has internal power conditioning that provide excellent PSRR. Output frequency is in range from 10 MHz to 900 MHz, footprint is compatible with well known SI598, but this device is much cheaper and has better availability.
- **CFPS-39IB 50.0MHZ** - 50 MHz single ended locked frequency oscillator

2.4 Power distribution

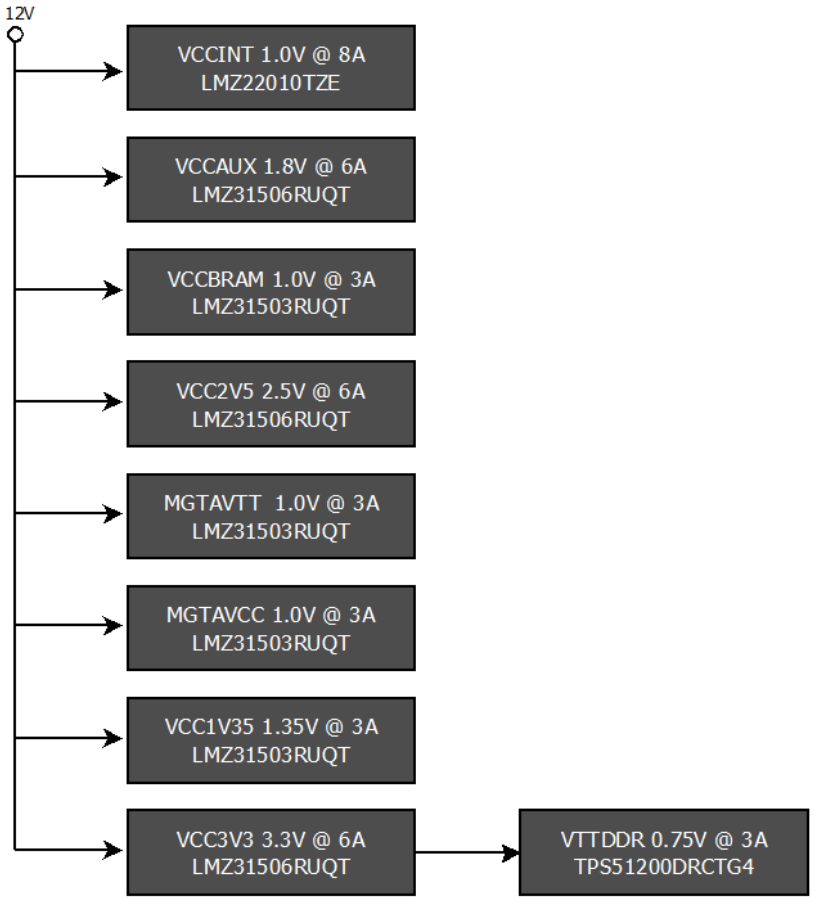

Figure 7. Power distribution scheme

For RTM Carrier, family of **SIMPLESwitcher** devices produced by Texas Instruments (TI) has been chosen. These are power modules with integrated shielded instructors that simplified PCB design and has low EMC emission. Three type of switchers, with output current of 8, 6 and 3 A, are used in design. Each specific FPGA power has own power supply that allows to implement maximum device resources utilization for biggest BGA 484 Artix-7 device. In addition, according to Xilinx documentation, MGT power supplies MGTAVTT and MGTAVCC should have own supplies.

FPGA's banks needs 3 types of supplies:

- 2V5 for LVDS_25 that are compatible with ZONE3 specification,
- 3V3 for bank connected to FPGAs peripherals, JTAG, and configuration memory
- 1V35 for bank connected to DDR3 memory, which allows to implement SSTL I/O standard needed for DDR memory communication

2.5 Management

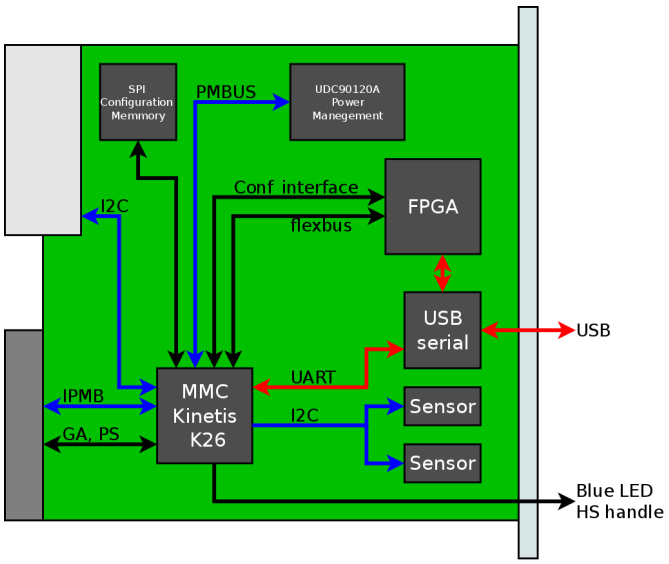

Figure 8. Board Management

MTCA.4 standard requires extensive management resources, to achieve expected availability of the system. RTM Carrier will have MMC (Module Management Controller) based on the Kinetis K-26 ARM microcontroller (**MK26FN2M0VLQ18**). This device has been chosen because it has four I2C interfaces, USB interface that allows to implement serial device, low current consumption, 256KB of RAM memory, and 2MB Flash memory. Also its possible to monitor on-board voltages using internal 16 bits ADC with build in voltage reference.

To simplify board operation, and avoid potential problems with MMC software controlled power start-up sequencing, a hardware power sequencer was placed on the board.

For RTM power control **TPS2459** device has been chosen. This is hot-plug controller for AMC, with digitally controlled inrush current and over-current protection. It has I2C interface, and it is possible to set current limits by software directly from MMC.

Except described devices, RTM carrier will be equipped in following management components:

- 2x I2C on-board temperature sensors **MAX6626RMTT**
- 1x I2C Temperature sensor **SA56004EDP** for readout of FPGA internal temperature sensor
- 1x EEPROM memory **24AA025E48T-I/OT** (typically for board identification data)
- 1x SPI Flash **N25Q256A13ESF40G**, foreseen for additional FPGA firmware storage

List of the devices on the local on-board I2C bus:

- 0x08 - Hot Swap and ORing Controller (U19)

- 0x49 - Temperature Sensor close to power regulators on upper part of PCB (U22)
- 0x4A - Temperature Sensor close to power regulators on bottom part of PCB (U23)
- 0x4C - Temperature Sensor close to hot swap switch right part of PCB and capable to read internal temperature of FPGA (U24)
- 0x50 - EEPROM (U20)

Board will have also USB-serial converter for communication with MMC serial port and/or FPGA.

2.6 Memory

RTM Carrier will have DDR3 memory of capacity 8Gb (1GB – one gigabyte). Memory will be implemented using single **MT41K1G8SN-125:A** device. It has internal organization 1Gx8, This memory has 8-bit data bus width, because this is only way to implement DDR3 interface utilizing single FPGA bank of Artix-7, but is not a problem since it can run with frequency of 300 MHz or more. Devices of this family is supported by latest Vivado design software, availability of this memory on the market has been confirmed.

3. Design Reports; including prototype reports, test & simulations

Fabricated RTM Carrier prototype device has been tested in the following steps:

- General visual inspection
- General electrical test – main power and ground nets has been checked if there is no short connection
- Power-on test – board has been powered using laboratory power supply, with current limit set to low value (hundreds of milliamperes), to avoid damage in case of short connection. Power has been applied first to 3.3V management power and then to 12V payload power. In both cases, current was far below safety limit, which was expected behavior
- Programmable devices response – MMC has been detected by the J-Link programming cable and MCUXpresso IDE, and FPGA has been identified in the JTAG chain using Platform USB cable and Vivado environment
- MMC verification

  Because MMC is responsible for controlling FPGA configuration process, it had to be verified first. For testing MMC, a dedicated test code for micro-controller has been developed. This code has checked and verified proper operation of following functions, on-board devices and interfaces:

  - LEDs – Front Panel LEDs

    - UART via FTDI – serial communication with MMC using console over the USB-Serial interface
    - MTCA H/S Switch
    - Voltage sequencing – turning on and off payload power regulators, mainly for FPGA
    - FPGA Configuration Pins – controlling the FPGA configuration process
    - Built-in analog comparator – required for detecting presence using PS1 and PS0 MTCA signals, if board is plugged in the MTCA Chassis or if it is running standalone out of crate
    - I2C sensors - three temperature sensors on the local on-board I2C bus
    - On-board payload power voltage measurements using 10 ADC channels
    - Identification EEPROM memory on the local I2C bus
    - RTM hot-swap controller placed on the local I2C bus, responsible for enabling power for the attached RTM device and handling different related cases such as in-rush current on startup
    - I2C communication with the RTM Management unit

- RTM Connectivity test-stand
    - Prototype board has been attached to the a RTM board (Vector Modulator RTM)
    - RTM has been powered properly
    - JTAG Chain has been closed properly
    - I2C management communication has been established
- FPGA Verification
    - FPGA Device has been programmed with compiled code with PCI Express core
- MTCA Vitrifaction
    - MMC has been programmed with code supporting minimal implementation of the IPMI, including FRU record describing requires power and describing links on the backplane (PCIe x2)
    - FPGA has been programmed with the firmware containing the PCIe endpoint
    - Board has been powered in the MTCA crate, it has been detected and activated by the MCH
    - In the CPU, which had PCIe Root Complex, lspci command has returned that “Xilinx Memory Device” is present, which confirms, that board is operational in the MTCA.4 crate.

Features and functions, which has not been tested:

- Low Latency Links
- MLVDS bus
- Clock signals distribution

Features and functions where problems has been found:

- Wrong connection of DDR memory chip to FPGA
- MMC JTAG connector pinout not compatible with standard J-Link programmer
- Wrong value of capacitors on the signal lines of FTDI chip
- Wrong footprint (polarization) of LEDs connected to MMC
- Missed pull-up resistors on I2C signal lines between FPGA and programmable clock generator

All problems listed above has been corrected, corrected design is ready for the next production batch.

4. Manufacturing and Verification

4.1 Description of planned test and measurements

General board testing procedure (for mass production units) shall include:

- Visual inspection, if there is no visible damage, if there is no missing components, if soldering looks good, if there are no external bodies that may cause short, etc. - if there is nothing suspicious
- Check with ohmmeter resistance between GND and major power nets (if there is no short)
- Power-on the board with the management power
- Check standby management power consumption on the power supply - if it is in the expected range;
- Check visually if LEDs expected to be active are illuminating.
- Check management power voltage level with multimeter
- Connect MMC programmer and load test firmware for MMC
- Measure management power consumption with loaded test firmware
- Check if result of MMC test firmware operation is as expected
- Power on the payload power;
- Check standby payload power consumption on the power supply - if it is in the expected

range

- Check visually if LEDs expected to be active are illuminating.
- Check voltages derived from payload power if they are correct
- Connect the FPGA programmer and load test FPGA firmware
- Measure payload power consumption with loaded test firmware
- Check if result of FPGA test firmware operation is as expected
- Load operational MMC code on the board
- Power-off the board
- Place board in the MTCA crate
- Power on the crate
- Check if board was properly initialized
- Load test firmeare 2 to FPGA,
- Check connectivity with the CPU via PCIe
- Check connectivity with RTM
- Check the connectivity with test boards using LLL
- Check triggering and receiving signal via MLVDS bus

FAT and SAT

For the factory acceptance test, each board will be tested individually as described above. Additionally, RTM Carrier boards will be tested as a components of the LLRF systems, which will be tested before shipping to ESS.

On the ESS site, whole LLRF systems will be delivered, so performing SAT of each individual component, such as RTM Carrier board, will not be possible, because it would require disassembly of whole unit. In such case, if particular LLRF systems will pass SAT, it should be assumed that all components has passed SAT as well.

4.2 Standards used for engineering design, construction and verification of the RTM carrier.

Board does not use voltages over 12V, so it shall be safe under general rules, but detailed analysis concerning the regulations which board needs to fulfill will be analyzed withing CE certification process.

4.3 Quality plan

Quality assurance of the RTM Carrier PCB project is based on:

- Continuous in parallel project (schismatics, PCB and libraries) review by other PCB design engineers, skillful in the used technologies, such as Xilinx FPGA.
- Drawing schematics using consistent conventions, such as naming convention, drawing style, etc.
- Project files are stored in Subversion version control (SVN) system repository on the server. This helps in tracking changes, guarantees always safe “step-back” option to the latest correct design version, and provides a form of backup – project is places always in at least 2 places: on engineer’s computer (actual working copy) and on the server (last committed version with all history of changes). Regardless of this, server with SVN repository is backed-up independently by itself.

4.4 Schedule for procurement, manufacturing, testing and delivery.

Actual agreement defines following schedule concerning the RTM Carrier:

Phase	**Beginning**	**End**
Design of the first prototype	01-01-2017	30-09-2017
Production of the first prototypes (2 pieces)	01-10-2017	30-11-2017
Testing of the first prototypes (2 pieces)	01-12-2017	15-04-2018
Production of the M-Beta prototypes (8 pieces)	01-05-2018	31-05-2018
Testing of the M-Beta prototypes (8 pieces)	01-06-2018	30-06-2018
Verification of the reference LLRF system	01-07-2018	31-08-2018
Production of the M-Beta devices (36 pieces)	01-09-2018	30-09-2018
Assembling and Testing M-Beta LLRF Systems (36 systems)	01-10-2018	31-12-2018
Design of the H-Beta prototype	01-05-2019	30-06-2019
Production of the H-Beta and remaining devices	01-07-2019	30-09-2019
Production of the H-Beta devices	01-10-2019	30-03-2020

Table 2. Project schedule

4.5 Risk analysis

Table 3 below presents potential risks and treatment plan

Event	**Cause**	**Impact**	**Treatment plan**

Late change of requirements	Insufficient data during requirements analysis	Late modules delivery.	Use as good as possible estimated requirements, if final ones are not available and build first prototype, expecting that for the next iteration(s), final requirements will be provided.
Delay in prototype design	Insufficient manpower	Delay in board delivery, increased cost	Increase manpower, find new employees
Prototype doesn't meet the requirements	Bad board concept	Delay in board delivery, increased cost	- Identify the reason of the problem - Correct the board concept and selected technology - Correct the prototype design - perform one more design iteration
Problems with components avaiablility	Long time between concept of the board and implementation/assembly	Delay in board delivery, increased cost	- Wait for components if unavailability is temporary - Try to order missing components, even if they are much more expensive - Try to find matching replacement components if possible - Consider which board functionality depends on the missing components and if this is acceptable - look for functional walk-around of the missing feature if possible - Redesign the board, to avoid using missing components
Broken components soldered on prototype	Problems with manufacturing technology, bad/broken components ordered, components stored/handled n the wrong way or	Delay in board delivery, increased cost	- Identify affected units - Order required amount of good components - replace bad components

	Insufficient quality control.		
Broken components soldered on M-Beta version	Problems with manufacturing technology, bad/broken components ordered, components stored/handled n the wrong way or Insufficient quality control.	Delay in board delivery, increased cost	- Identify affected units - Order required amount of good components - replace bad components - send repair team to ESS if affected units has been already shipped
Broken components soldered on H-Beta version	Problems with manufacturing technology, bad/broken components ordered, components stored/handled n the wrong way or Insufficient quality control.	Delay in board delivery, increased cost	- Identify affected units - Order required amount of good components - replace bad components - send repair team to ESS if affected units has been already shipped
Not detected design mistake in final version	- Insufficient quality control - error in test-stand, which covered described mistake - Error in board testing software, which covered this mistake	Increased cost of maintenance. Decreased performance. Reduced functionality.	- Identify mistake severity - Identify functionalities disabled by this mistake - Consider if this mistake could be acceptable (in contrast to full redesign and reproduction cost) - Perform board redesign and fabrication of all boards if there is no other way

Table 3. RTM Carrier project risks