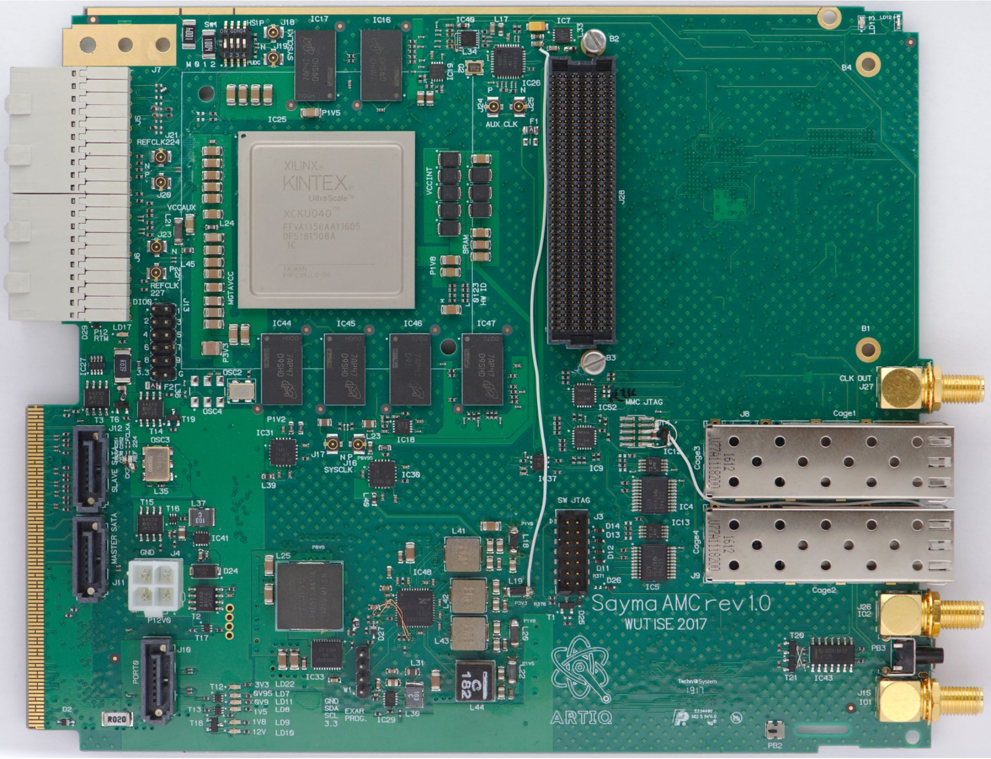
SAYMA AMC

specification



v1.0(11.2017)

Document version:	Preliminary
Issue Date:	March 1, 2018
Written by:	Filip Świtakowski
Approved by:	Greg Kasprowicz
Document title:	SAYMA AMC- specification

Todo list

Contents

1 Glossary

AFE Analogue front-end.

AMC Module or Modul An AMC Module is a mezzanine or modular add-on card that extends the functionality of a Carrier Board. The term is also used to generically refer to the different varieties of Multi-Width and Multi-Height Modules.

BaseMod Base-band input/output mezzanine.

COTS Commercial off-the-shelf. Product which is designed and can be easily purchased.

EEM Eurocard Extension Module is a Sinara standard for low-cost, low-bandwidth peripherals that are controlled by ARTIQ DRTIO.

Fat Pipes Ports 4 though 11 of the AMC Connector constitute the Fat Pipes Region. This Region of Ports is intended for the assignment of multiple Lane interfaces, also called “fat pipes”. Fat Pipe 1 [Ports 4-7], Fat Pipe [Ports 8-11].

FMC FPGA Mezzanine Card

HEPP High Energy Physics. ???

Hot Swap To remove a component (e.g., an AMC Module) from a system (e.g., an AMC Carrier AdvancedTCA Board) and plug in a new one while the power is still on and the system is still operating.

IPMB ntelligent Platform Management Bus. The lowest level hardware management bus as described in the Intelligent Platform Management Bus Communications Protocol Specification.

Management Power or MP The 3.3V power for a Module’s Management function, individually provided to each Slot by the Carrier

MGT Multi-Gigabit Transceiver.

MixMod An up-converting mezzanine, using an analogue IQ mixer to mix the input and output RF signals with a LO supplied by Sayma.

MMC Module Management Controller. The MMC is the required intelligent controller that manages the Module and is interfaced to the Carrier via IPMB-Local.

RFBP RF Backplane.

RTM Rear Transition Module.

Sayma Smart Arbitrary Waveform Generator, providing 8 channels of 1.2 GSPS 16-bit DACs (2.4 GHz DAC clock) and 125 MSPS 16-bit ADCs. It consists of an AMC, providing the high-speed digital logic, and a RTM, holding the data converters and analog components.

Sianra Open-source hardware ecosystem originally designed for use in quantum physics experiments running the ARTIQ control software. It is licensed under CERN OHL v1.2.

uTCA Micro Telecommunications Computing Architecture. MicroTCA is a modular, open standard for building high performance computer systems in a small form factor.

Introduction to ARTIQ Sinara

Sinara is an open-source hardware ecosystem originally designed for use in quantum physics experiments running the ARTIQ control software. It is licensed under CERN OHL v1.2.

Control electronics used in many trapped-ion and other quantum physics experiments suffers from a number of problems. In general, an ad-hoc solution is hastily put together in-house without enough consideration about good design, reproducibility, testing and documentation. This makes those systems unreliable, fragile, and difficult to use and maintain. It also duplicates work in different laboratories. In addition, the performance and features of the existing systems (e.g. regarding pulse shaping abilities) is becoming insufficient for some experiments.

To alleviate those problems, Sinara aims to be:

- high-quality
- simple to use and "turn-key"
- reproducible and open
- flexible and modular
- well tested
- well supported by the ARTIQ control software

Sinara is currently developed by a collaboration including M-Labs, Warsaw University of Technology (WUT), US Army Research Laboratory (ARL), the University of Oxford, the University of Maryland and NIST. The majority of the hardware was designed by WUT. The work was funded by ARL, Duke University, the University of Oxford, and the University of Freiburg.

Following the ARTIQ model, an experiment consists of a core device (master) – typically either a `Metlino` or `Kasli` – controlling multiple slave devices in real time using ARTIQ's distributed real-time IO (DRTIO) protocol. DRTIO provides both gigabit communication links and time distribution over copper cable or optical fibres. It synchronises all device clocks, ensuring they have deterministic phase relationships, and enables nanosecond timing resolution for input and output events across all devices in the experiment. More detailed information about communication between devices and time distribution inside Sinara can be found here.

Sinara uses two main form factors for hardware requiring real-time control: microTCA (uTCA) and Eurocard Extension Modules (EEM). Non real-time hardware is typically connected to the host PC using ethernet.

MicroTCA (uTCA) is Sinara's preferred form factor for high performance hardware with high-speed data converters requiring deterministic phase control, such as the *Sayma* Smart Arbitrary Waveform Generator (SAWG). Information about uTCA hardware, including a list of parts needed to build a Sinara uTCA crate can be found here.

EEMs provide a lower cost, simpler platform than uTCA for hardware that requires real-time control, but not bandwidth or complexity of uTCA hardware.

Extension modules connect to a carrier, such as `Kasli` or the VHDCI carrier, which provides power and DRTIO. They are designed to be mounted either in stand-alone enclosures, or in a rack with a carrier, and connect to the carrier via ribbon cable. More details about the extension module standard can be found here.

uTCA hardware interfaces with the extension modules either directly, using a VHDCI carrier, or indirectly, using a `Kasli` DRTIO slave.

2 uTCA.4 Overview

MicroTCA (uTCA) is Sinara's preferred form-factor for hardware with high-speed data converters requiring deterministic phase control, such as the *Sayma* 2.4 GSPS smart arbitrary waveform generator (SAWG).

uTCA is a modular, open standard originally developed by the telecommunications industry. It allows a single rack master – the Micro TCA Carrier Hub (MCH) – to control multiple slave boards, known as Advanced Mezzanine Cards (AMCs) via a high-speed digital backplane. uTCA chassis and backplanes are available commercially of the shelf (COTS).

We make use one of the most recent extension to the uTCA standard, uTCA.4. Originating in the high-energy and particle physics (HEPP) community, uTCA.4 introduces rear-transition modules (RTMs) along with a second backplane for low-noise RF signals (RFBP). Each RTM connects to an AMC (one RTM per AMC). Typically, the AMCs hold FPGAs and other high-speed digital hardware, communicating with the MCH via gigabit serial links over the AMC backplane. The RTMs hold data converters and other low-noise analog components, controlled by the corresponding AMC. The RFBP provides low-noise clocks and local oscillators (LOs). The RTMs and RFBP are screened from the AMCs to minimise interference from the high-speed digital logic.

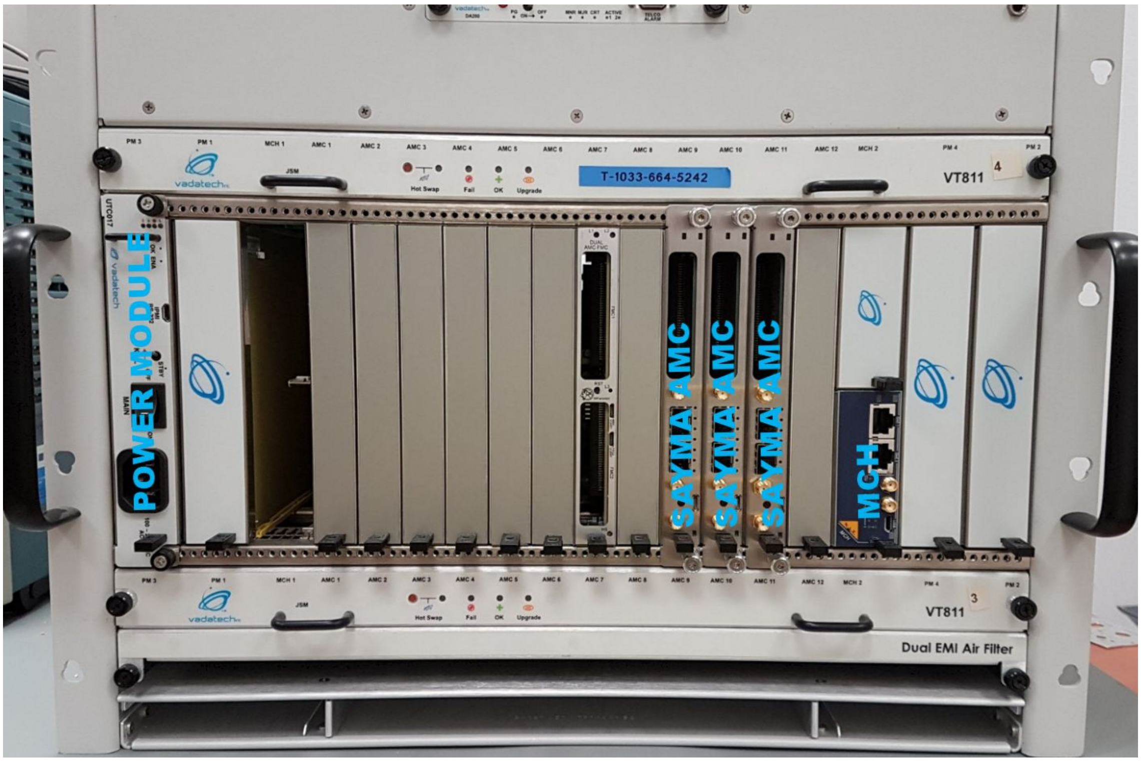


Figure 1: Micro TCA chassis with 3 Sayma AMC modules inserted

New Figure: add photo of Sayma_AMC attached to Sayma_RTM sitting on bench so how they're connected is clear.

(above) Micro TCA chassis with 3 Sayma AMC modules inserted.

Micro TCA chassis with 4 RTM modules inserted. One of them with 4 BaseMod AFE mezzanines installed.

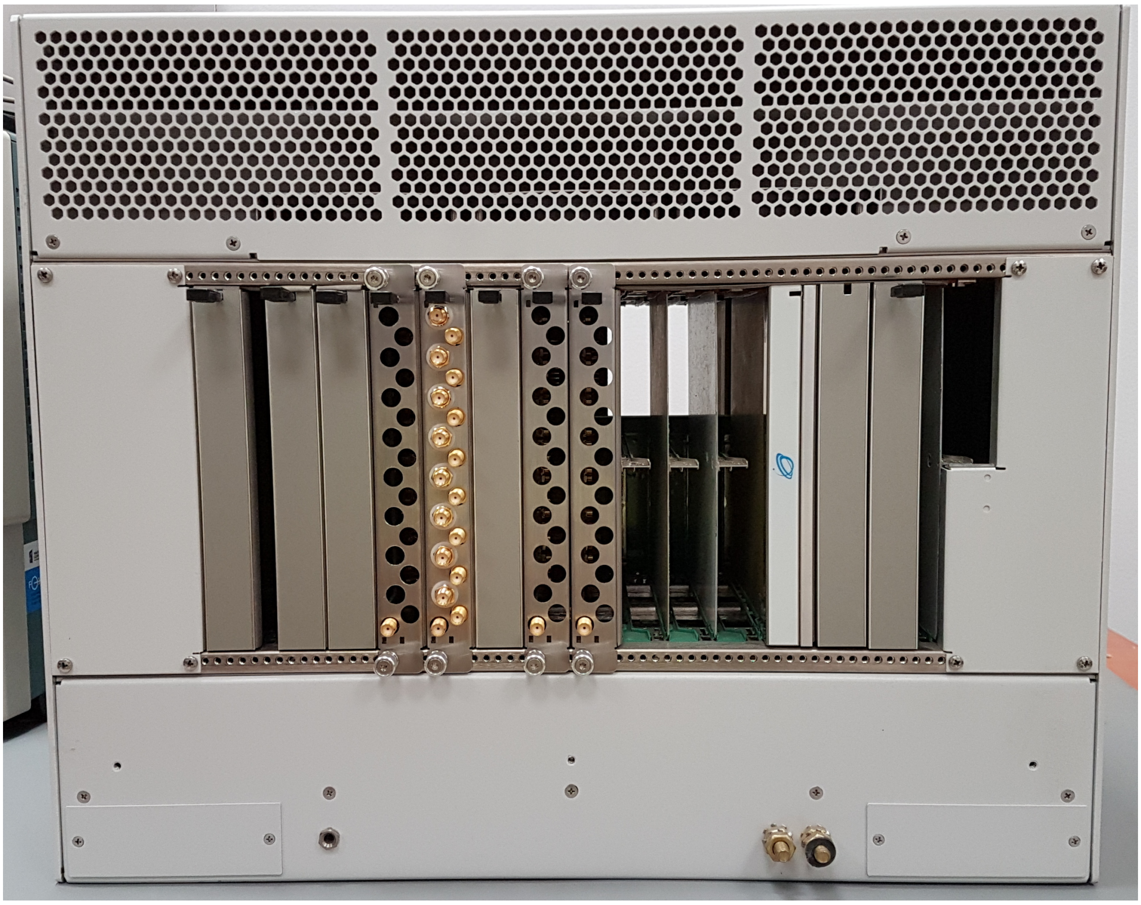


Figure 2: Micro TCA chassis with 4 RTM modules inserted. One of them has 4 BaseMod AFE mezzanines installed.

RF BP datasheet

3 uTCA parts and suppliers

- NAT AC 600D, qty 1
- NAT MCH-Basic v3.5, mid-size front panel, qty 1
- NAT Native-R5, qty 1

4 Schematic / Layout Viewer

Hardware was designed under Mentor Graphics Xpedition Enterprise and Altium Designer CAD tools. Project resources are in two separate folders:

- ARTIQ_EE folder is for designs made with the Mentor Graphics Xpedition Enterprise CAD tool.
- ARTIQ_ALTIUM folder is for designs made with Altium Designer CAD tool.

Read-only access to PCB schematics and layout designs is possible using free tools.
Mentor has a free tool called visECAD Viewer.
Altium has a free tool called Altium Designer viewer

5 Sayma

Sayma is a hardware that supports M-Lab's Smart Arbitrary Waveform Generator(SAWG) gateware. Provide 8 channels of 1.2 GSPS 16-bit DACs (2.4 GHz DAC clock) and 125 MSPS 16-bit ADCs. It consists of an AMC, providing the high-speed digital logic, and a RTM, holding the data converters and analog components.

The design files are located in ARTIQ_EE/PCB_Sayma_AMC and ARTIQ_EE/PCB_Sayma_RTM and, the AMC schematic is here and the RTM schematic is here. The PCBs are double width, mid height AMC module. Sayma AMC

5.1 Features

- May be used in a uTCA rack or stand-alone operation with fibre-based DRTIO link
- Analog input and output front-ends provided by plug-in AFE modules (eg BaseMod) for maximum flexibility.
- Extremely flexible clocking options
- Flexible feedback to SAWG parameters planned. Specification here.

5.2 Key AMC Components

Programmable resources:

- Xilinx Kintex UltraScale – XCKU040-1FFVA-1156C FPGA 20 I/O, 530K Logic Cells
  - speed grade: -1
  - 20 GTH transceivers (Max Preformance 16.3 Gb/s)
- MMC: LPC17762984

Memory:

- 512Mb DDR3 SDRAM (32-bit interface), 800MHz (clock)
- 1Gb DDR3 SDRAM (64-bit interface), 800MHz (clock)
- SPI Flash for FPGA configuration. Accessible by MMC
- SPI Flash for user data storage
- EEPROM with MAC and unique ID

Connectivity:

- 1 high pin count (HPC) FMC slot for single width mezzanine card
- Micro-USB UART connected to FPGA or MMC
- Stand-alone 12V power connector

- MGT (Multi-Gigabit Transceiver) connected to:
  - RTM x16
  - Fat_Pipe1 x2
  - SFP x2
- Port 0 – possibility connected to SATA
- RTM connector compatible with Sayma RTM module

Supply:

- Monitoring of voltage and Power supply for RTM 12V and FMC 12V
- FMC VADJ fixed to 1V8
- Monitoring current of all FMC buses
- Stand-alone power connectore

Clocking:

- Clock recovery Si5324 is a precision clock multiplier andjitter attenuator
- UFL CLK input
- SMA CLK output

Other:

- Temperature, voltage and current monitoring for critical power buses
- Temperature monitoring: FMC1, supply, FPGA core, DDR memory
- JTAG multiplexer (SCANSTA) for FMC access, local JTAG port and remote debug/Chipscope via Ethernet

5.3 Key RTM Components

not sure if it should be here? only in RTM doc?

- **DAC**: AD9154 4-channel high-speed data converter
  - data rate is 1.2 GS/s at 16-bit
  - clock is up to 2.4 GHz (1x, 2x, 4x and 8x interpolating modes)
  - supports mix-mode to emphasize power in 3rd Nyquist Zone
  - interface is 8-lane JESD204B (subclass 1)
  - power consumption is 2.11 W

  - each Sayma has 2 AD9154

- **ADC**: AD9656 is a 4-channel high-speed digitizer
  - data rate is 125 MS/s at 16-bit
  - clock is up to 125 MHz
  - 650 MHz analog bandwidth
  - interface is 8-lane, 8 Gb/s per lane, JESD204B (subclass 1)
  - each Sayma has 2 AD9656

- **clock generation**: (summarized here)
  - Sayma has several distinct clock domains
    - ∗ DAC, JESB204B output clock
    - ∗ ADC, JESD204B input clock
    - ∗ LO for analog mezzanines
  - These clocks may be generated using a low phase noise Clock Mezzanine PCB. A single Clock Mezzanine can be shared by several Sayma in a uTCA crate using [Baikal] PCB and an RTM RF backplane. Alternately, each Sayma can have its own distinct Clock Mezzanine (local generation).

- **clock distribution**
  - HMC7043 SPI 14-Output Fanout Buffer for JESD204B
  - HMC830 SPI fractional-N PLL

- **calibration ADC**: AD7194BCPZ is a 20-bit ADC for monitoring/calibration

6 Product view

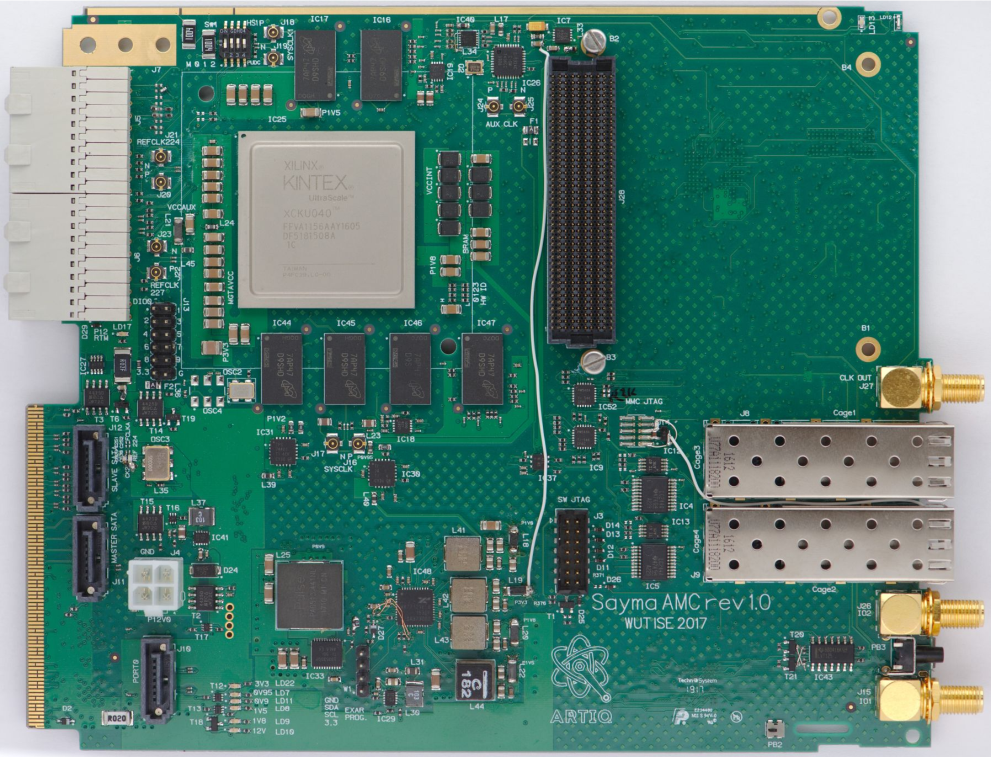


Figure 3: Top view

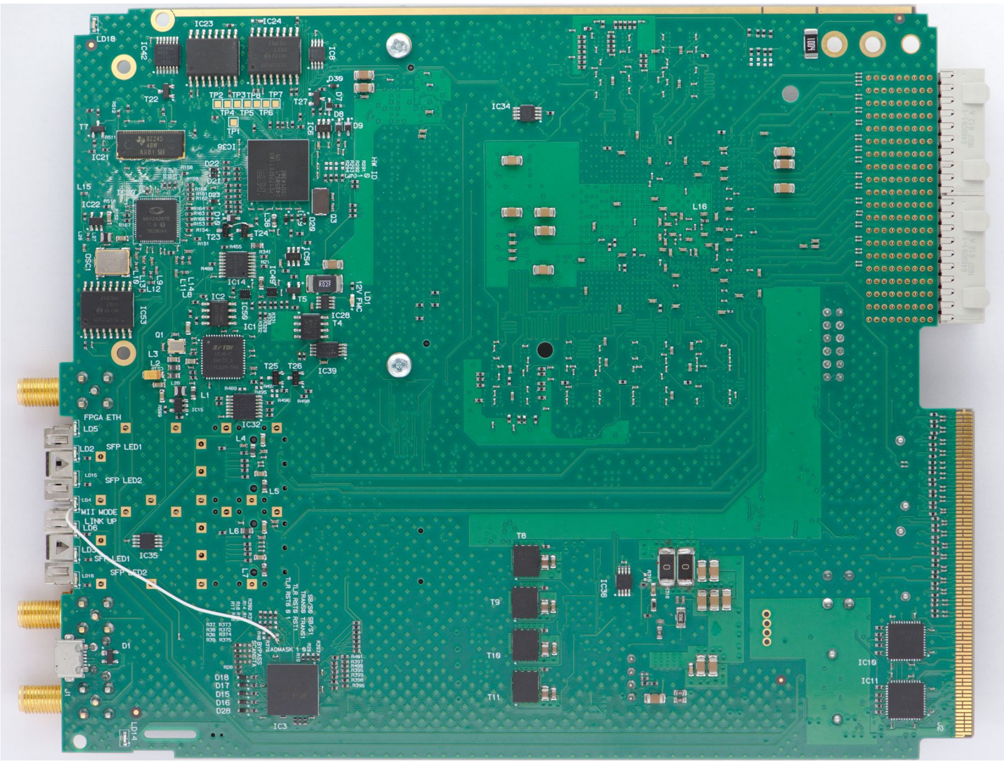


Figure 4: Bottom view

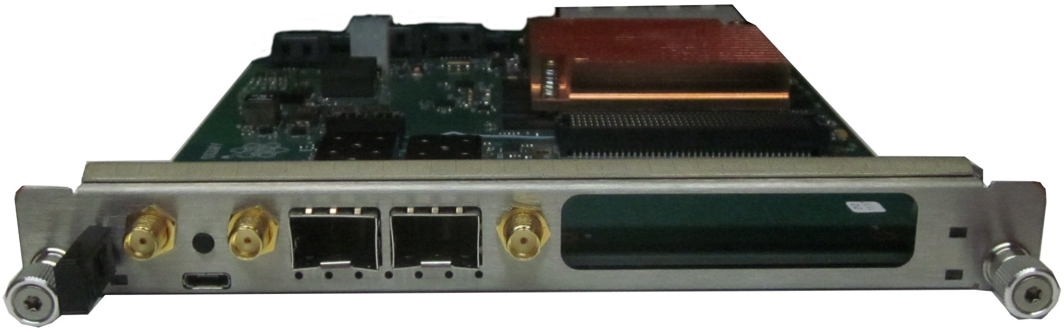


Figure 5: Front view

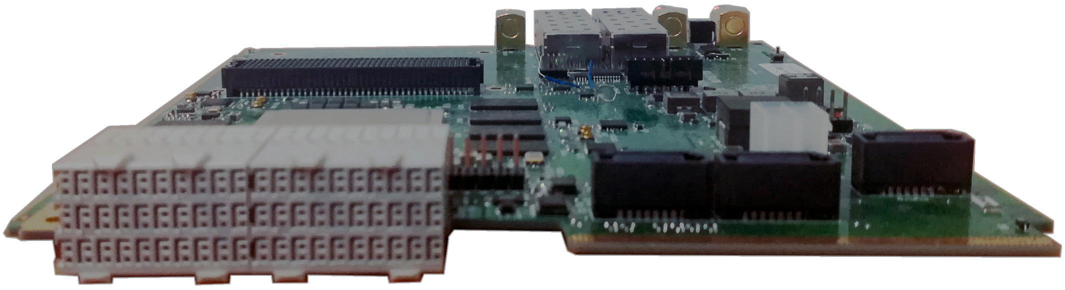


Figure 6: Back view

7 Routing

This section contain general blockm scheme of SAYMA AMC board and I2C map with addresses. General Block Scheme -figure 7 shows more important connections between components. I2C connections with addresses can be found in figure 9. Detailed clocking scheme can be found in next paragraph in figure 10.

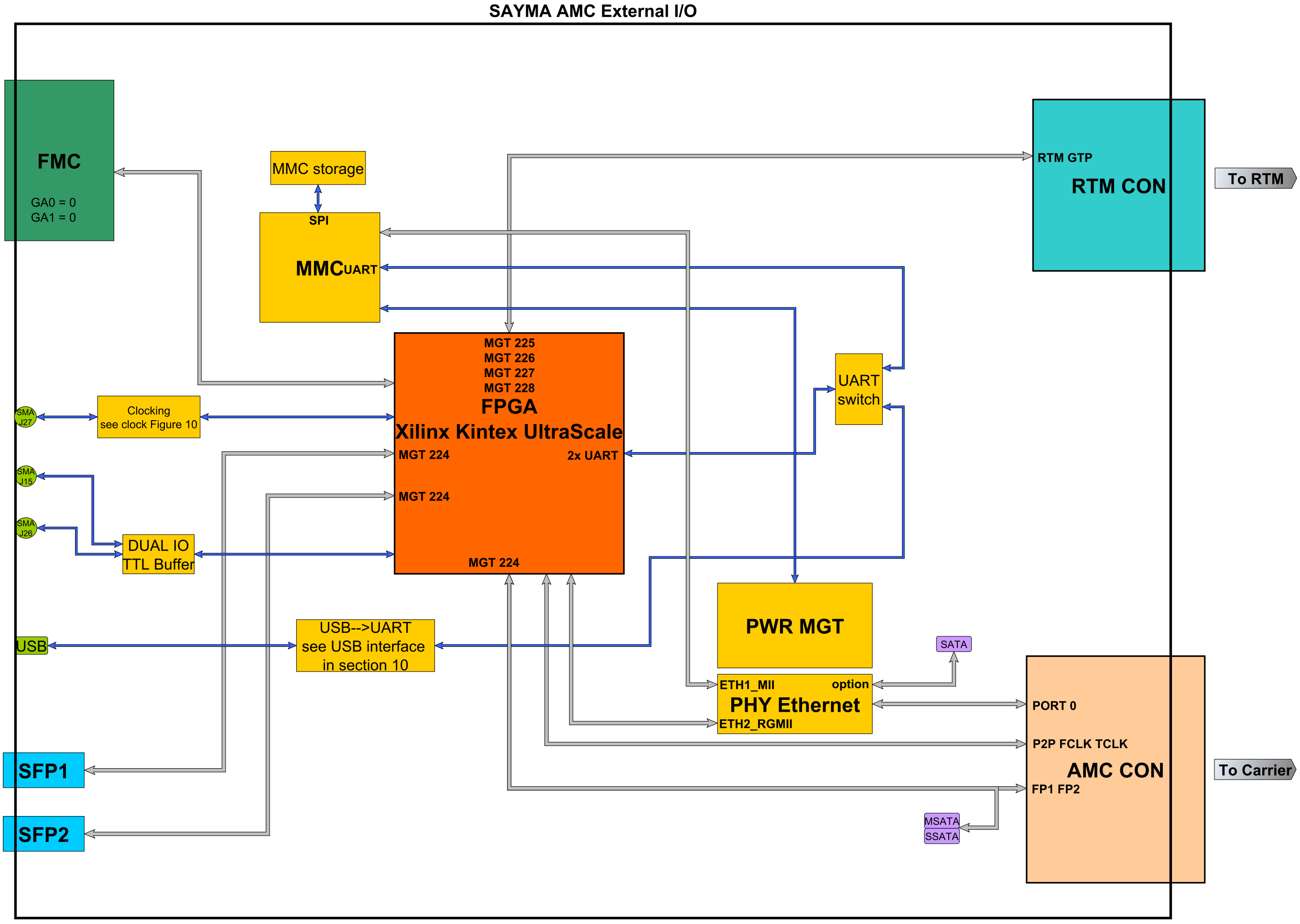


Figure 7: General Block Scheme

5

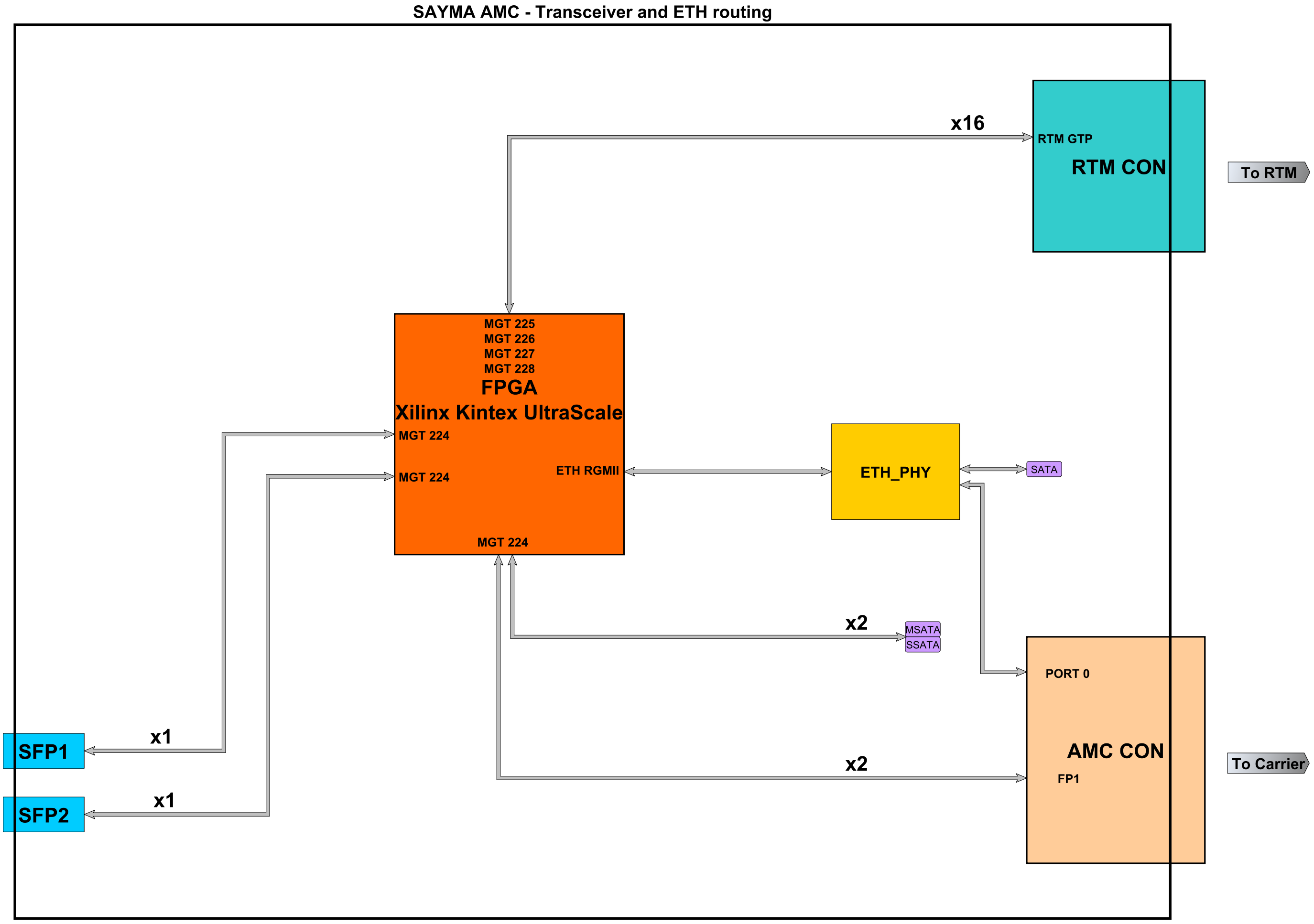


Figure 8: MGT

Transceiver MGT	Direction	Routed to
0_224	TX	SFP1
0_224	RX	SFP1
1_224	TX	SFP2
1_224	RX	SFP2
2_224	TX	FP1 or MASTER SATA
2_224	RX	FP1 or MASTER SATA
3_224	TX	FP1 or SLAVE SATA
3_224	RX	FP1 or SLAVE SATA
0_225	TX	RTM_GTP
0_225	RX	RTM_GTP
1_225	TX	RTM_GTP
1_225	RX	RTM_GTP
2_225	TX	RTM_GTP
2_225	RX	RTM_GTP
3_225	TX	RTM_GTP

3_225	RX	RTM_GTP
0_226	TX	RTM_GTP
0_226	RX	RTM_GTP
1_226	TX	RTM_GTP
1_226	RX	RTM_GTP
2_226	TX	RTM_GTP
2_226	RX	RTM_GTP
3_226	TX	RTM_GTP
3_226	RX	RTM_GTP
0_227	TX	RTM_GTP
0_227	RX	RTM_GTP
1_227	TX	RTM_GTP
1_227	RX	RTM_GTP
2_227	TX	RTM_GTP
2_227	RX	RTM_GTP
3_227	TX	RTM_GTP
4_227	RX	RTM_GTP
0_228	TX	RTM_GTP
0_228	RX	RTM_GTP
1_228	TX	RTM_GTP
1_228	RX	RTM_GTP
2_228	TX	RTM_GTP
2_228	RX	RTM_GTP
3_228	TX	RTM_GTP
4_228	RX	RTM_GTP

The I2C MUX is made from two (TCA9548ARGER) I2C multiplexers. In Sayma AMC there are two main I2C busses: MMC_I2C and FPGA_I2C. Each of them is connected to one multiplexer. Outputs are tied together, so Masters (MMC and FPGA) can acces to any of 7 I2C busses. Addidtionaly MMC has acces to FPGA_I2C and is connected to IPMB through AMC connector.

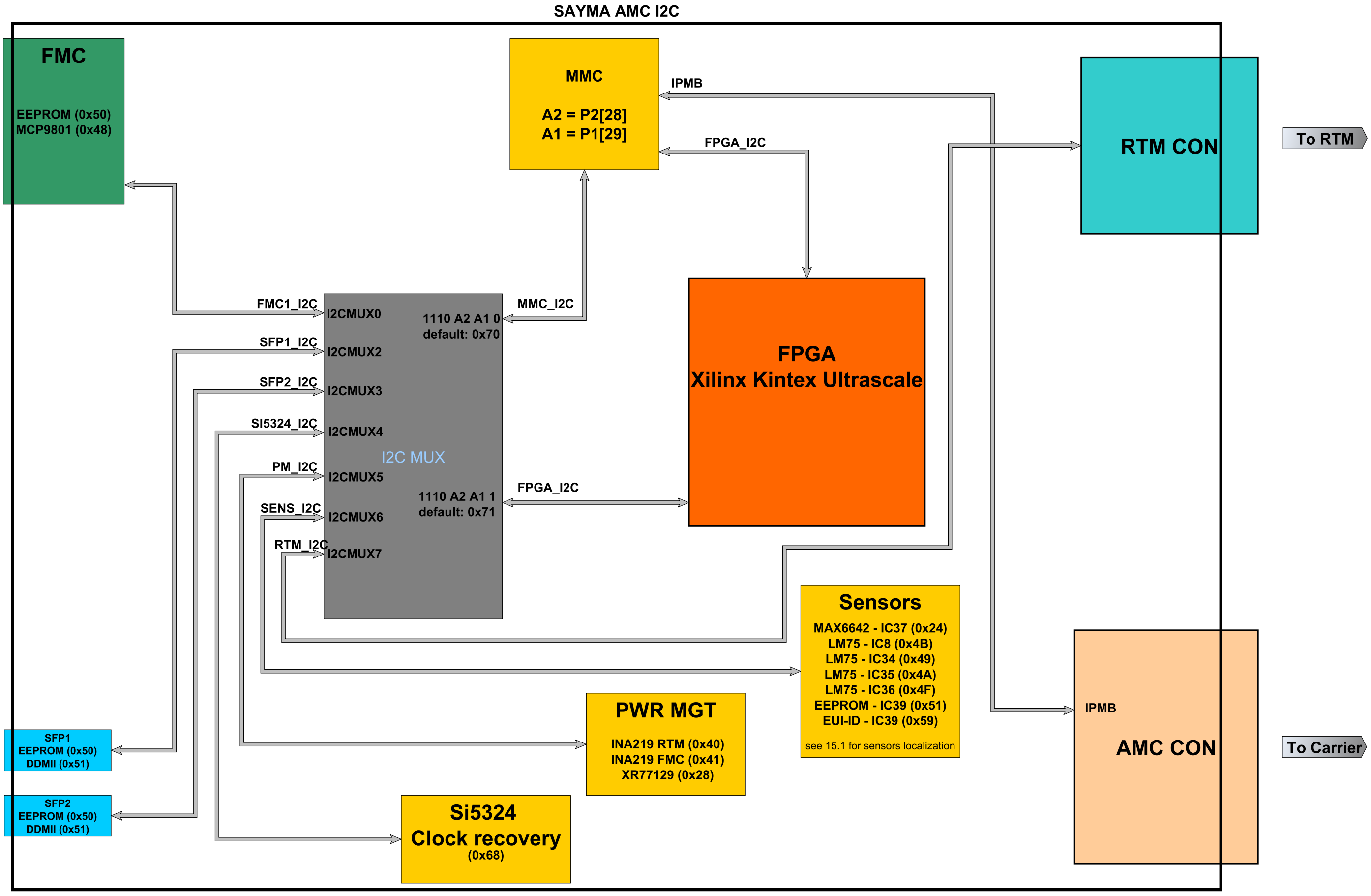


Figure 9: I2C map with addresses in hex

8 Clocking

This section describes how and where clock signals are routed.

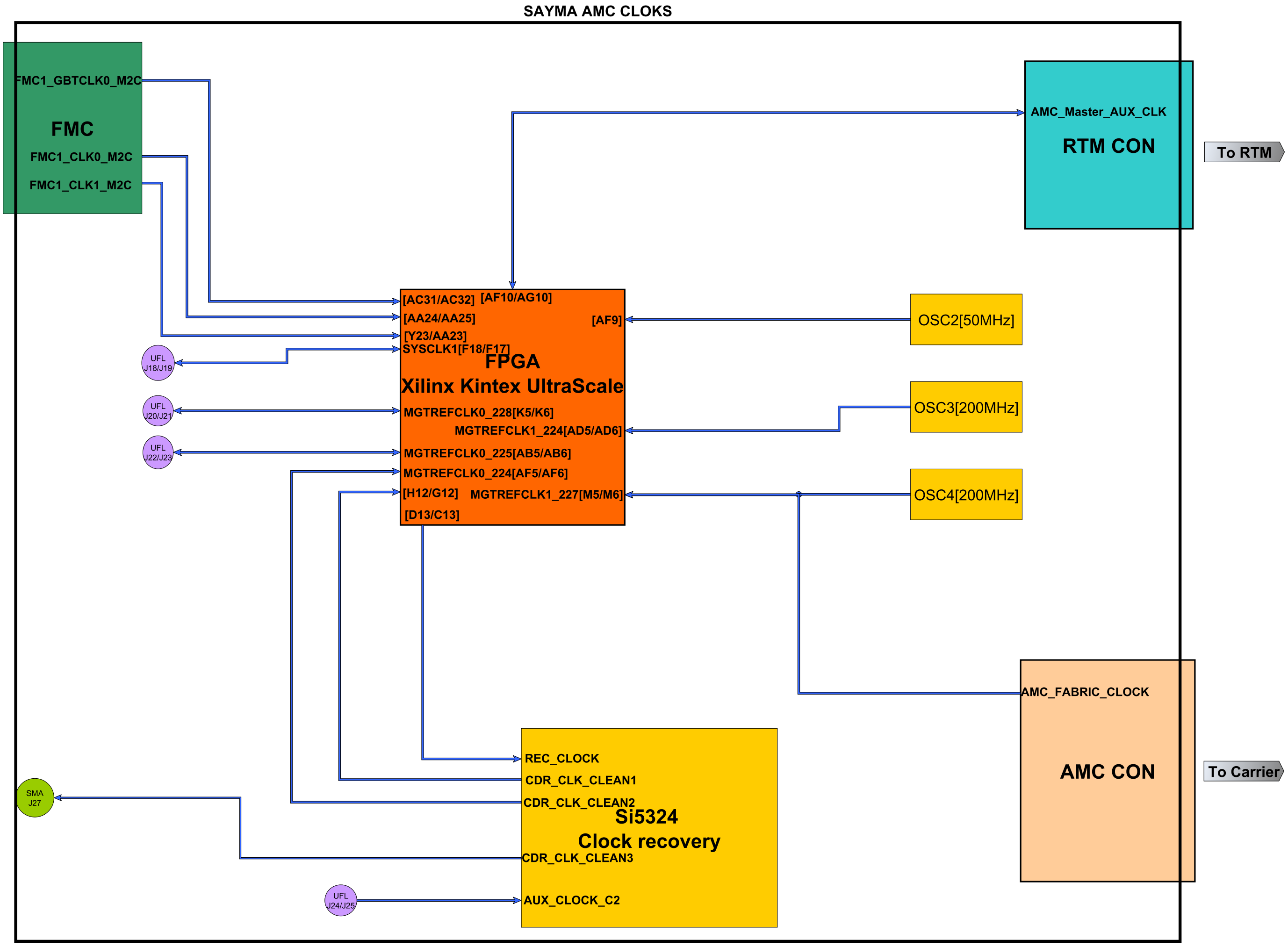


Figure 10: Clocks

- OSC2 - 50MHz main clock source for FPGA resources
- OSC3 - place-holder
- OSC4 dedicated 200MHz clock source to gigabit transceivers

9 Front panel and headers

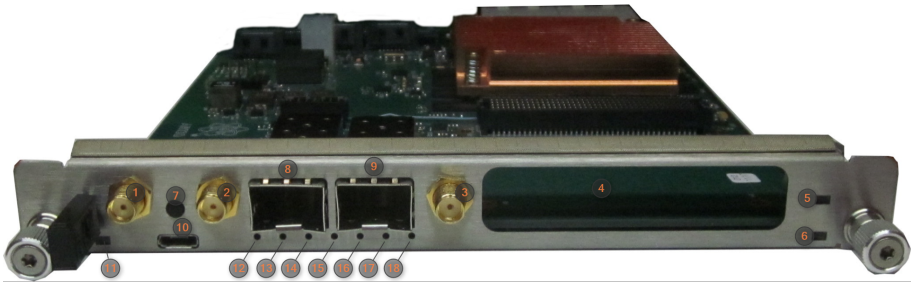


Figure 11: Front view

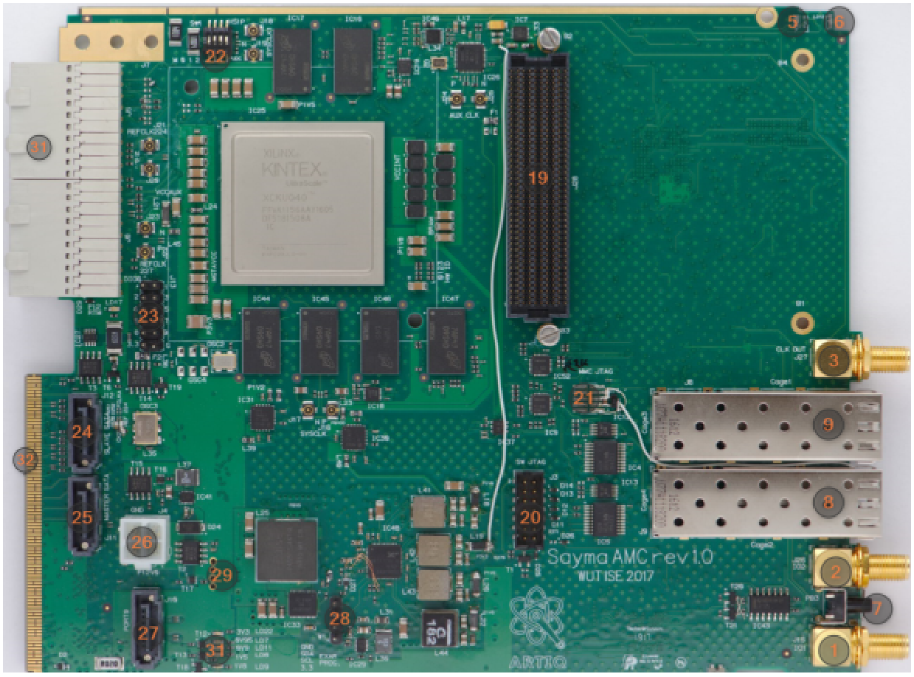


Figure 12: Front view

Call out table		
Call out	Designator	Description
1	J15	GPIO - IO1
2	J26	GPIO - IO2
3	J27	CLK OUT output driven by Si5324 (CDR_CLK_CLEAN3)
4	J28	FMC connector
7	PB3	routed to MMC, allows to flash MMC via USB
8	cage2	SFP Cage
9	cage1	SFP Cage
10	J1	Micro USB –>Serial
19	J28	FMC header

20	J3	JTAG header connected to SCANSTA
21	J14	MMC JTAG
22	SW1	FPGA MODE, see SW1 paragraph
23	J13	Digital IOs connected to FPGA
24	J12	SLAVE SATA
25	J11	MASTER SATA
26	J4	Power In
27	J10	SATA connected to Port 0
28	W1	EXAR I2C header
29	–	P12V0
30	–	Test points
31	J6	RTM Connector
32	J2	AMC Connector

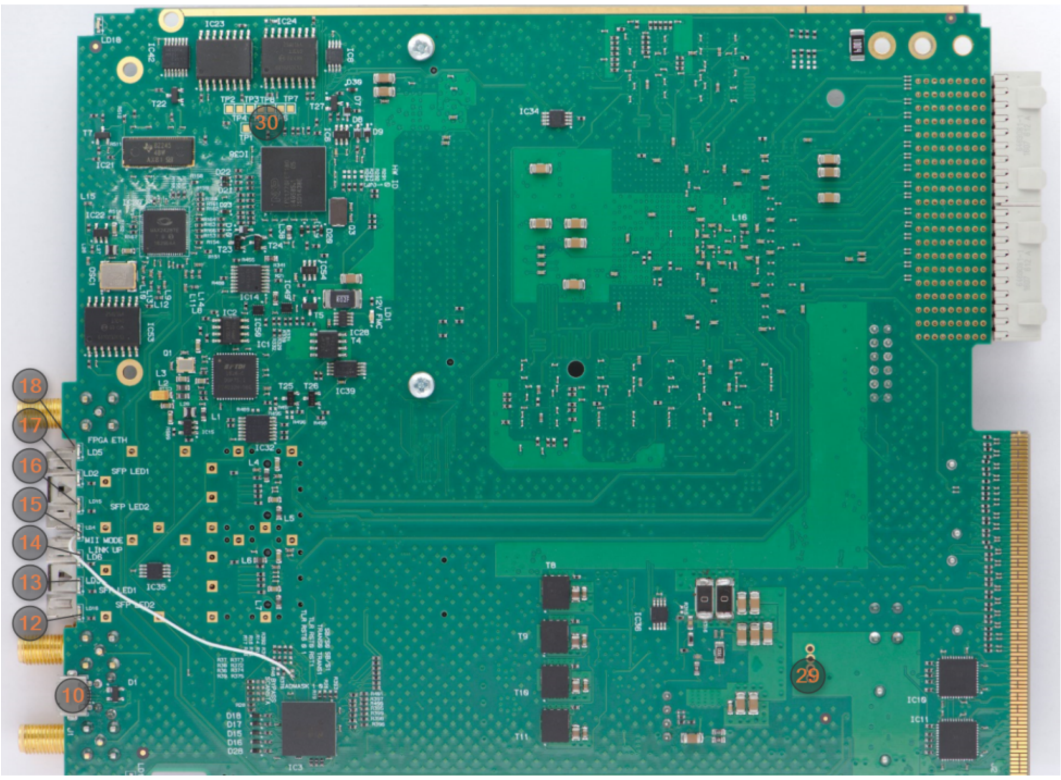


Figure 13: Front view

Callout 5: LD13 is

- off if MMC is ok
- red if MMC is in an error state

Callout 6: LD4 is

- off the board has initialized in crate
- blue if power is cut off and is possible to remove the board.

Callout 33: LD21 is

- green if FPGA is configured
- off if FPGA is not configured

Callout 11: LD14 is

- green if MMC successfulled operation
- off if MMC unsuccessfuled operation

Callout 14: LD20 is

- green if Link is up
- off if Link is down

Callout 15: LD18 is

- green if Ethernet is in MII mode (mTCA doesn't support 100mbit Ethernet)
- off if Ethernet is in RGMII mode

Callout 12: LD3 is

- red if SFP2 - user defined
- off if SFP2 - user defined

Callout 13: LD6 is

- green if SFP2 - user defined
- off if SFP2 - user defined

Callout 16: LD2 is

- red if SFP1 - user defined
- off if SFP1 - user defined

Callout 17: LD5 is

- green if SFP1 - user defined
- off if SFP1 - user defined

Callout 18: LD19 is

- green if - user defined
- off if - user defined

Power LED table

Call out	Designator	Description	Colour	nominal state	IC	Failure
30	LD22	3V3	Green	on	Power	off
30	LD7	0V95	Green	on	Power	off
30	LD11	0V9	Green	on	Power	off
30	LD8	1V5	Green	on	Power	off
30	LD9	1V8	Green	on	Power	off
30	LD10	12V	Green	on	Power	off

9.1 Headers pinout

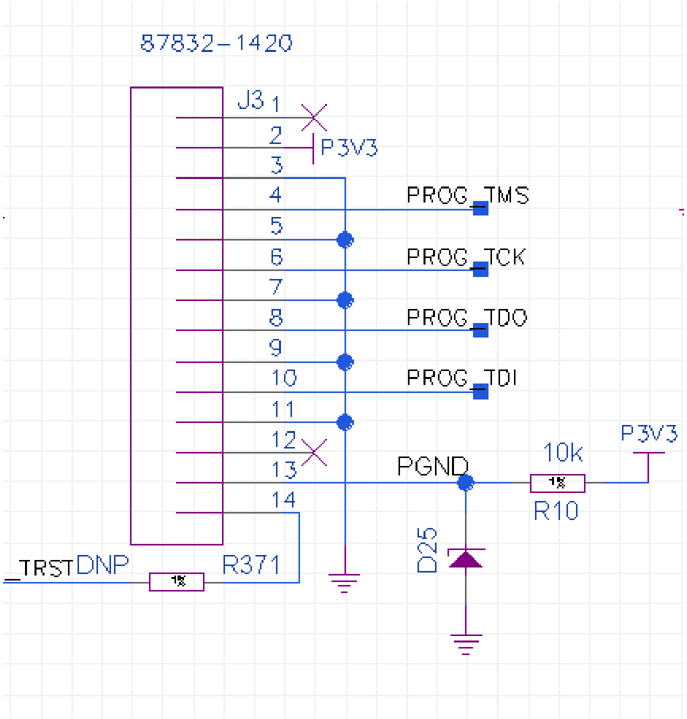


Figure 14: JTAG - Call out 20

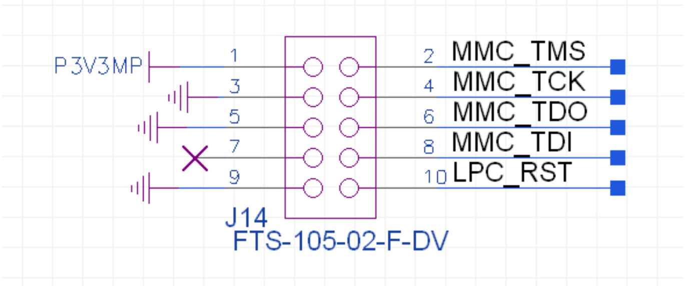


Figure 15: JTAG - Call out 21

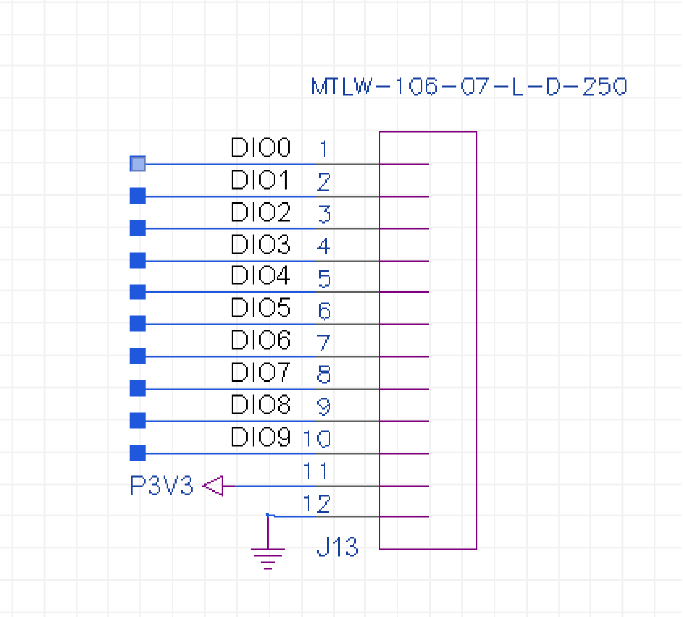


Figure 16: DIO - Call out 23

Tespoints table - Call out 30		
TPx	Sig Name	LPC pin
TP1	MII1_col	C13
TP2	SDCLK	J10
TP3	SDCMD	K14
TP4	SDPWR	K11
TP5	SDDAT0	L14
TP6	SDDAT1	M12
TP7	SDDAT2	N14
TP8	SDDAT3	M11

9.2 Location ICs

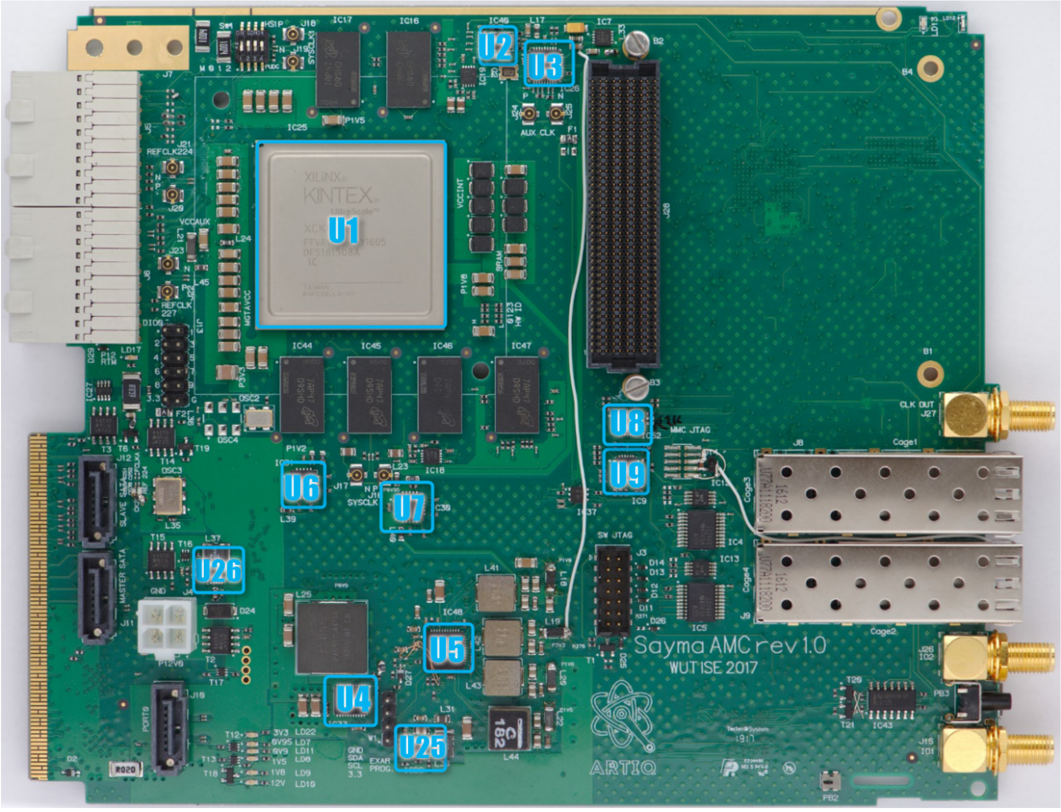


Figure 17: Top

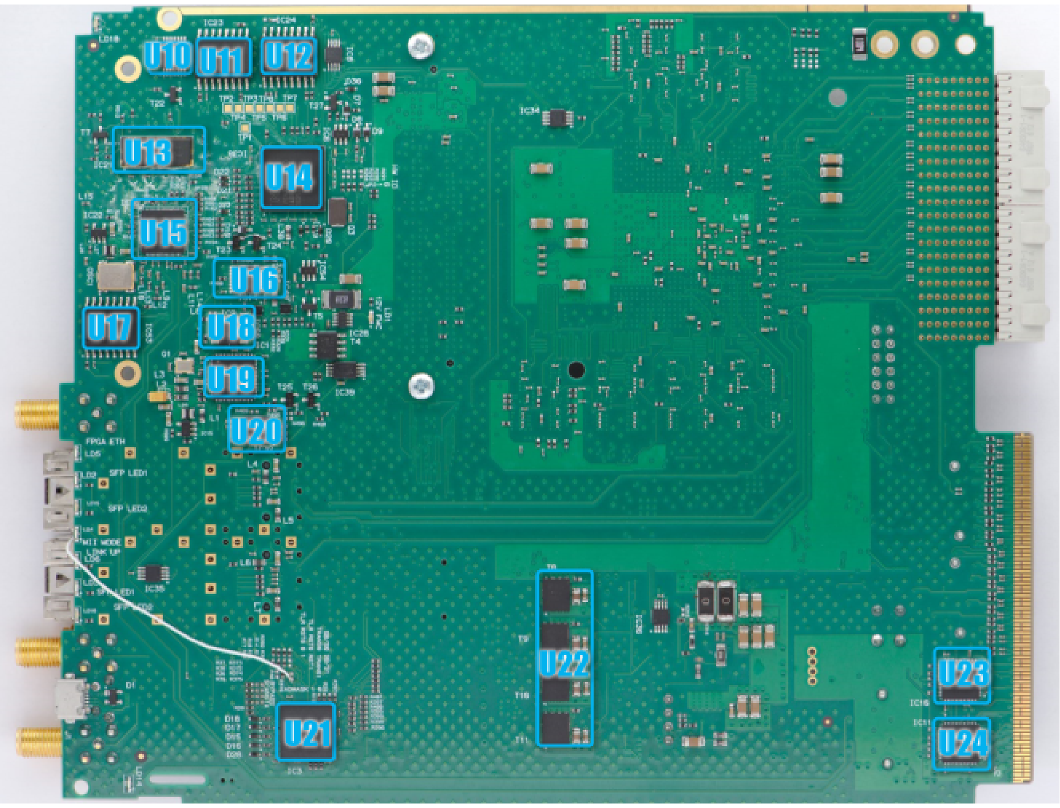


Figure 18: Bot

ICs Location

Ux	IC	Description
U1	Kintex	FPGA
U2	LTC 6957	Low Phase Noise Buffer
U4	TPS53353	P0V9
U5	XR77129	EXAR
U6	TPS 74401	P1V2
U7	TPS 74401	P0V95
U3	SI5324C	Clock recovery
U8	TCA9548	I2C switch - MMC
U9	TCA9548	I2C switch - FPGA
U10	74HC4066PW	Analog switch - Flash update
U11	N25Q256A13ESF40	NOR Flash
U12	N25Q256A13ESF40	NOR Flash
U13	SN74CB3Q32245ZKE	Digital Bus switch - RGMI/MII
U14	LPC1776FET180	MMC
U15	MAX24287ETK+	ETH switch
U16	AN74CBT3257PW	USB console switch
U17	N25Q256A13ESF40	NOR Flash - MMC
U18	M93C46	EEPROM
U19	F4232H-56Q	USB-UART Bridge
U20	74HC4066PW	USB-UART Switch
U21	SCANSTA112SM	SCANSTA JTAG Switch
U22	FDMS7608S	EXAR Transistors
U23	SN65MLVD040RGZT	LVDS transceiver
U24	SN65MLVD040RGZT	LVDS transceiver
U25	TPS62175	P5V0
U26	TPS62175	P3V3

9.3 SW1

SW1 is used to chose configuration mode of the FPGA. Configuration modes define the specifics of how the FPGA will interact with the data source, external control logic. SW1 is tied directly to Bank 0 pf FPGA. All pins have pull up, switching SW1 proceed of connecting to lower potential.

SW1 table

M0	M1	M2	Description
0	0	0	Master Serial Mode
0	0	1	Master Parallel Up
0	1	1	Master Parallel Down
1	0	1	Peripheral mode
1	1	1	Slave Serial mode

10 FMC

- VADJ: 1V8 @ 1A
- FPGA Banks: 47HP and 48HP

The connector is compliant with ANSI/VITA 57.1 FMC-LPC Standard.

FMC1		
FPGA signal	FPGA ball	Signal on the board
IO_L12P_T1U_N10_GC_47	AA24	FMC1_CLK0_M2C_P
IO_L12N_T1U_N11_GC_47	AA25	FMC1_CLK0_M2C_N
IO_L11P_T1U_N8_GC_47	Y23	FMC1_CLK1_M2C_P
IO_L11N_T1U_N9_GC_47	AA23	FMC1_CLK1_M2C_N
IO_L12P_T1U_N10_GC_48	AC31	FMC1_GBTCLK0_M2C_P
IO_L12N_T1U_N11_GC_48	AC32	FMC1_GBTCLK0_M2C_N
IO_L1P_T0L_N0_DBC_48	AE27	FMC1_DP0_M2C_P
IO_L1N_T0L_N1_DBC_48	AF27	FMC1_DP0_M2C_N
IO_L2P_T0L_N2_48	AE28	FMC1_DP0_C2M_P
IO_L2N_T0L_N3_48	AF28	FMC1_DP0_C2M_N
IO_L13P_T2L_N0_GC_QBC_48	AA32	FMC1_LA00_CC_P
IO_L13N_T2L_N1_GC_QBC_48	AB32	FMC1_LA00_CC_N
IO_L14P_T2L_N2_GC_48	AB30	FMC1_LA01_CC_P
IO_L14N_T2L_N3_GC_48	AB31	FMC1_LA01_CC_N
IO_L8P_T1L_N2_AD5P_48	AF33	FMC1_LA02_P
IO_L8N_T1L_N3_AD5N_48	AG34	FMC1_LA02_N
IO_L21P_T3L_N4_AD8P_48	V33	FMC1_LA03_P
IO_L21N_T3L_N5_AD8N_48	W34	FMC1_LA03_N
IO_L7P_T1L_N0_QBC_AD13P_48	AG31	FMC1_LA04_P
IO_L7N_T1L_N1_QBC_AD13N_48	AG32	FMC1_LA04_N
IO_L10P_T1U_N6_QBC_AD4P_48	AE33	FMC1_LA05_P
IO_L10N_T1U_N7_QBC_AD4N_48	AF34	FMC1_LA05_N
IO_L15P_T2L_N4_AD11P_48	AC34	FMC1_LA06_P
IO_L15N_T2L_N5_AD11N_48	AD34	FMC1_LA06_N
IO_L18P_T2U_N10_AD2P_48	AC33	FMC1_LA07_P
IO_L18N_T2U_N11_AD2N_48	AD33	FMC1_LA07_N
IO_L11P_T1U_N8_GC_48	AD30	FMC1_LA08_P
IO_L11N_T1U_N9_GC_48	AD31	FMC1_LA08_N
IO_L9P_T1L_N4_AD12P_48	AE32	FMC1_LA09_P
IO_L9N_T1L_N5_AD12N_48	AF32	FMC1_LA09_N
IO_L17P_T2U_N8_AD10P_48	AA34	FMC1_LA10_P
IO_L17N_T2U_N9_AD10N_48	AB34	FMC1_LA10_N
IO_L16P_T2U_N6_QBC_AD3P_48	AA29	FMC1_LA11_P
IO_L16N_T2U_N7_QBC_AD3N_48	AB29	FMC1_LA11_N
IO_L24P_T3U_N10_48	V31	FMC1_LA12_P
IO_L24N_T3U_N11_48	W31	FMC1_LA12_N
IO_L19P_T3L_N0_DBC_AD9P_48	W33	FMC1_LA13_P
IO_L19N_T3L_N1_DBC_AD9N_48	Y33	FMC1_LA13_N
IO_L23P_T3U_N8_48	U34	FMC1_LA14_P
IO_L23N_T3U_N9_48	V34	FMC1_LA14_N
IO_L22P_T3U_N6_DBC_AD0P_48	Y31	FMC1_LA15_P

IO_L22N_T3U_N7_DBC_AD0N_48	Y32	FMC1_LA15_N
IO_L20P_T3L_N2_AD1P_48	W30	FMC1_LA16_P
IO_L20N_T3L_N3_AD1N_48	Y30	FMC1_LA16_N
IO_L13P_T2L_N0_GC_QBC_47	W23	FMC1_LA17_CC_P
IO_L13N_T2L_N1_GC_QBC_47	W24	FMC1_LA17_CC_N
IO_L14P_T2L_N2_GC_47	W25	FMC1_LA18_CC_P
IO_L14N_T2L_N3_GC_47	Y25	FMC1_LA18_CC_N
IO_L16P_T2U_N6_QBC_AD3P_47	V22	FMC1_LA19_P
IO_L16N_T2U_N7_QBC_AD3N_47	V23	FMC1_LA19_N
IO_L17P_T2U_N8_AD10P_47	T22	FMC1_LA20_P
IO_L17N_T2U_N9_AD10N_47	T23	FMC1_LA20_N
IO_L18P_T2U_N10_AD2P_47	V21	FMC1_LA21_P
IO_L18N_T2U_N11_AD2N_47	W21	FMC1_LA21_N
IO_L15P_T2L_N4_AD11P_47	U21	FMC1_LA22_P
IO_L15N_T2L_N5_AD11N_47	U22	FMC1_LA22_N
IO_L10P_T1U_N6_QBC_AD4P_47	AB21	FMC1_LA23_P
IO_L10N_T1U_N7_QBC_AD4N_47	AC21	FMC1_LA23_N
IO_L8P_T1L_N2_AD5P_47	AC22	FMC1_LA24_P
IO_L8N_T1L_N3_AD5N_47	AC23	FMC1_LA24_N
IO_L9P_T1L_N4_AD12P_47	AA20	FMC1_LA25_P
IO_L9N_T1L_N5_AD12N_47	AB20	FMC1_LA25_N
IO_L7P_T1L_N0_QBC_AD13P_47	AA22	FMC1_LA26_P
IO_L7N_T1L_N1_QBC_AD13N_47	AB22	FMC1_LA26_N
IO_L6P_T0U_N10_AD6P_47	AB25	FMC1_LA27_P
IO_L6N_T0U_N11_AD6N_47	AB26	FMC1_LA27_N
IO_L24P_T3U_N10_47	V26	FMC1_LA28_P
IO_L24N_T3U_N11_47	W26	FMC1_LA28_N
IO_L23P_T3U_N8_47	V29	FMC1_LA29_P
IO_L23N_T3U_N9_47	W29	FMC1_LA29_N
IO_L22P_T3U_N6_DBC_AD0P_47	U26	FMC1_LA30_P
IO_L22N_T3U_N7_DBC_AD0N_47	U27	FMC1_LA30_N
IO_L21P_T3L_N4_AD8P_47	W28	FMC1_LA31_P
IO_L21N_T3L_N5_AD8N_47	Y28	FMC1_LA31_N
IO_L20P_T3L_N2_AD1P_47	U24	FMC1_LA32_P
IO_L20N_T3L_N3_AD1N_47	U25	FMC1_LA32_N
IO_L19P_T3L_N0_DBC_AD9P_47	V27	FMC1_LA33_P
IO_L19N_T3L_N1_DBC_AD9N_47	V28	FMC1_LA33_N
VREF_48	AA30	FMC1_VREF_A_M2C
VREF_47	V24	FMC1_VREF_A_M2C

11 USB-UART

11.1 USB console switch

UART from FPGA is connected through Multiplexer (SN74CB3T3257PW). Selection between MMC and USB is preformed automaticly. When micro-USB is connected, +5V from USB bus switches the multiplexer to pass data from USB to FPGA, after un plugging cable, the switch signal S fall down and the multiplexer conects MMC to FPGA.

- PRI_UART is connected to FPGA, configuration is 1N8 115200 baudrate
- AUX_UART is connected to FPGA, configuration is 1N8 115200 baudrate
- UART1 is connected to MMC, configuration is 1N8 115200 baudrate
- UART4 is connected to MMC, configuration is 1N8 115200 baudrate
- MMC_CONS_PROG is conected to MMC, configuration is 1N8 115200 baudrate

11.2 USB-UART bridge

The USB-UART bridge (FT4232H) requires USB device drivers, vailable free from http://www.ftdichip.com, which are used to make the FT4232H on the Mini Module appear as a four virtual COM ports (VCP). This then allows the user to communicate with the USB interface via a standard PC serial emulation port (TTY).

12 JTAG

Scansta112 is 7-Port Multidrop JTAG Multiplexer. It is used to partition scan chains into managable sizes, or to isolate specific devices onto a separate chain. By default Scansta input signal is from IDC header. AMC JTAG is connected to Master Port on SCANSTA, so it can be used as Master or Slave module. The rest modules (MMC, FPGA, FMC, RTM) are tied to slave SCANSTA outputs.

Tere are two JTAG sources – either AMC connector (JSM module) or USB to JTAG bridge (FTDI chip). There is also onboard JTAG connector (Xilinx type) -J3. Insertion of JTAG programmer probe deactivates the FTDI JTAG connectivity and forces SCANSTA chip set this port as master port. By default SCANSTA selects AMC port as master one.

In Sayma AMC, SCANSTA112 is used in Transparent Sticher Mode. In this mode, the IC can be configured via hardware to skip the addressing protocol needed, sothere is no need to run a SVF configuration file on IMPACT when programming the FPGA bitstream.

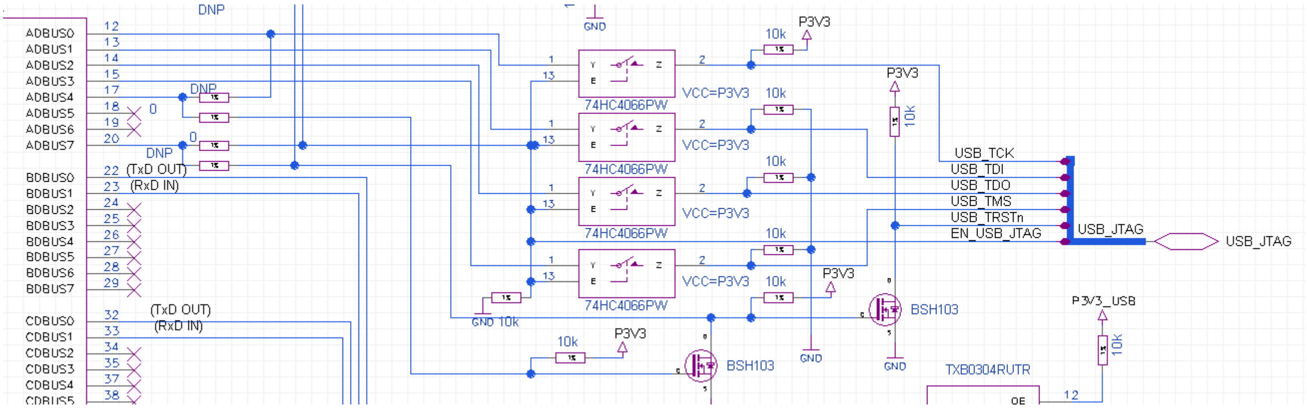


Figure 19: USB–>JTAG

General block scheme of Scansta connections is shown below.

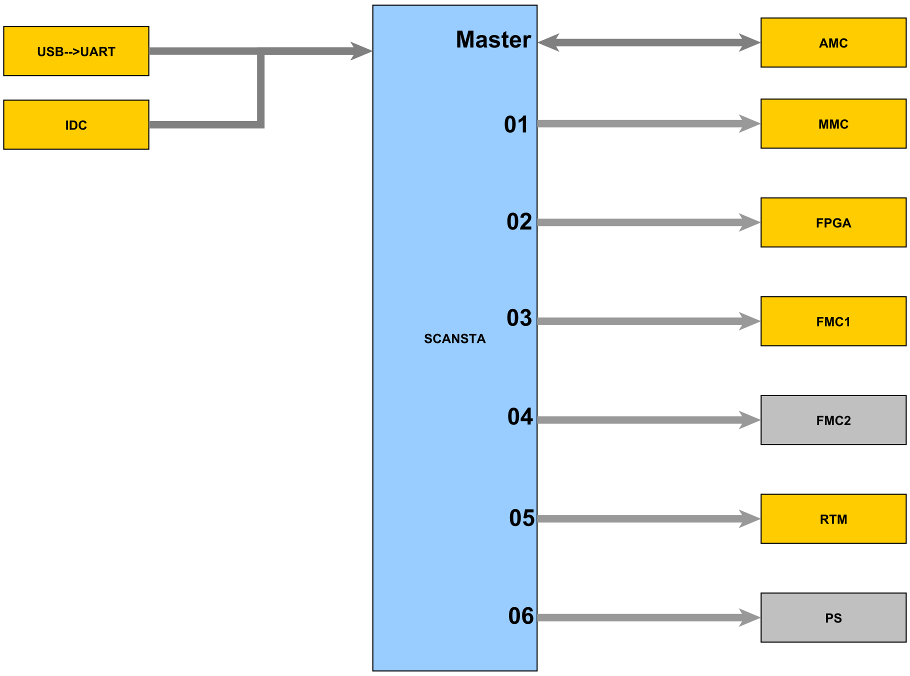


Figure 20: SCANSTA bloch scheme

Note: *The FMC2 and PS is not used.*

Each of JTAG slave devices is connected directly to SCANSTA. SCANSTA allows to connect all devices in chain witn an option to pass one or more devices, intention in Figure 21.

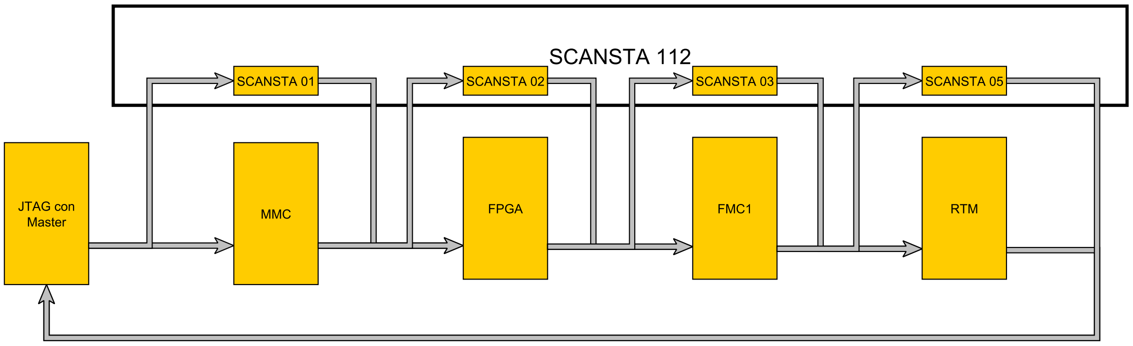


Figure 21: SCANSTA JTAG chain

Simplified instruction if using SCANSTA can be found under: http://www.ti.com/lit/an/snla068c/snla068c.pdf

13 Power

13.1 Power supply

The card can operate as stand alone devide, or plugged into a uTCA crate. While working standalone the power is provided by Molex Connector(39-28-1043) -J4. The pinout is shown in Figure 22.
When the card is inserted to the crate the power is applied from AMC connector -J2, +12V (8 power lines) and +3.3_MP(1 line). At the begening +3.3_MP power up the MMC, then +12 is converted to lower voltages, simplified power map is in figure 24. All on board voltages (exept P3V3_MP) are enabled by MMC. There are two types of power distributors, fixed LDOs and Exar chip - quad channel digital Pulse Width Modulated (PWM) step down (buck) controller. Exar allows for adjust power parameters and for set particular power oorder. Exar firmware monitors current and responds if its too high. Additionaly all LDOs have connected PowerGood outputs so it is possible to read the proper state of all power busses.

TBD Can MMC readout Exar debug information?

TBD voltage noise

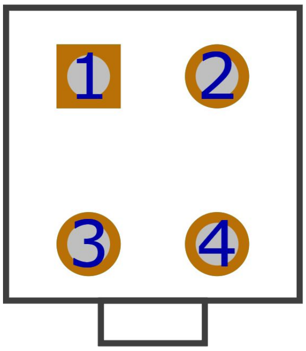


Figure 22: Power connector

GND	GND
+12	+12

Maximum board(AMC+RTM module) power consumption estimate to 3A @ 12V.

Note: *Please note that power consumption mostly depends from FPGA configuration.*

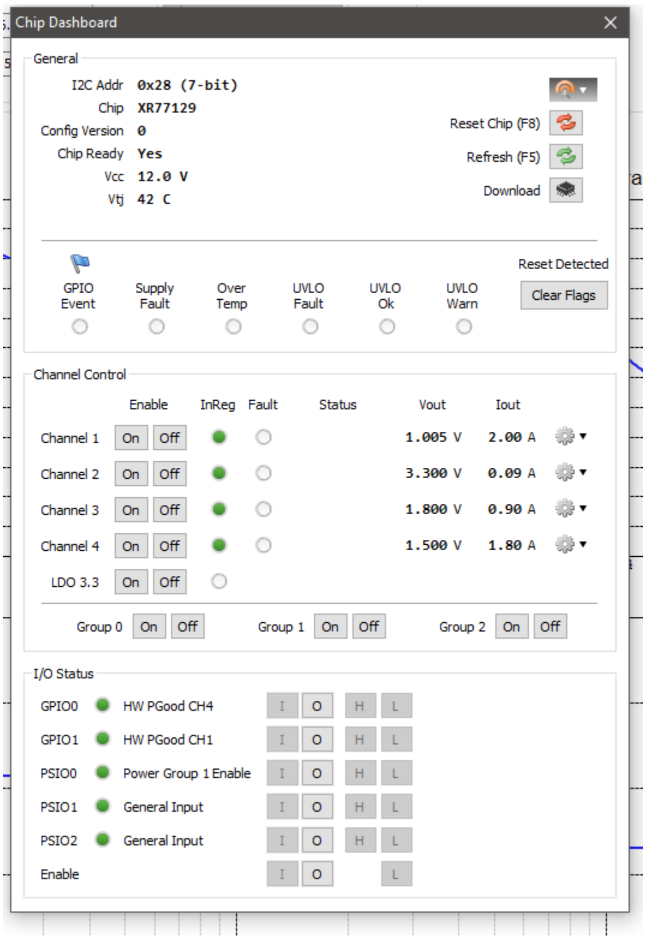
- Input voltage range: 10.8-13.2 [V]

- The board needs active cooling. Approx. 20CFM in 20 C air.

Exar chips are configured via PM_I2C bus (I2CMUX5) or directly by connecting to W1 (call-out 28) header. For proper configuration **Exar Power Architect** in version **5.2-r1** is needed.

Exar Power Archtect 5.2-r1: https://www.exar.com/content/document.ashx?id=21632
Configuration files: https://github.com/m-labs/sinara/tree/master/EXAR_config
Datasheet:https://www.exar.com/ds/xr77129_1a_120514.pdf
Quick Start Guide: https://www.exar.com/files/powerxr/PA5-QSG_110_010614.pdf

Actual voltages and current consumption, temperature can be found in Chip Dashboard. There is also oportunity to adjust settings.



Chip Dashboard

General

I2C Addr 0x28 (7-bit)
Chip XR77129
Config Version 0
Chip Ready Yes
Vcc 12.0 V
Vtj 42 C

Reset Chip (F8)
Refresh (F5)
Download

Reset Detected

GPIO Event	Supply Fault	Over Temp	UVLO Fault	UVLO Ok	UVLO Warn
○	○	○	○	○	○

Clear Flags

Channel Control

	Enable		InReg	Fault	Status	Vout	Iout
Channel 1	On	Off	●	○		1.005 V	2.00 A
Channel 2	On	Off	●	○		3.300 V	0.09 A
Channel 3	On	Off	●	○		1.800 V	0.90 A
Channel 4	On	Off	●	○		1.500 V	1.80 A
LDO 3.3	On	Off	○				

Group 0 On Off   Group 1 On Off   Group 2 On Off

I/O Status

GPIO0	●	HW PGood CH4	I	O	H	L
GPIO1	●	HW PGood CH1	I	O	H	L
PSIO0	●	Power Group 1 Enable	I	O	H	L
PSIO1	●	General Input	I	O	H	L
PSIO2	●	General Input	I	O	H	L
Enable			I	O		L

Figure 23: Chip Dashboard

13.2 Power configuration

13.2.1 Power map

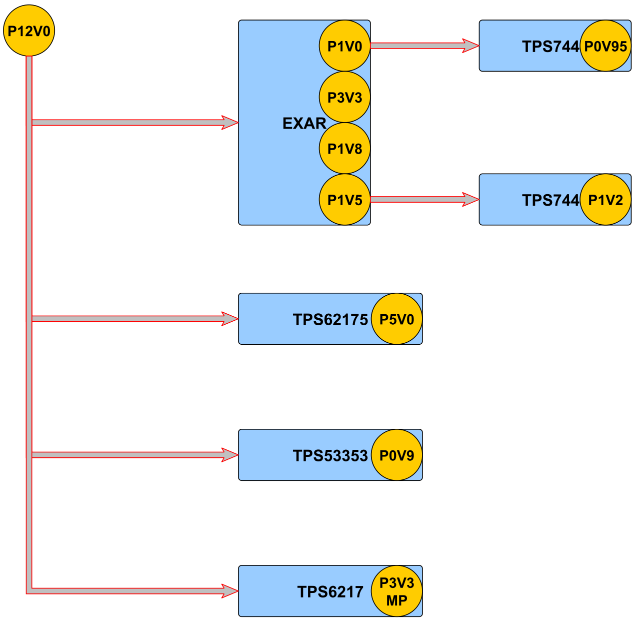


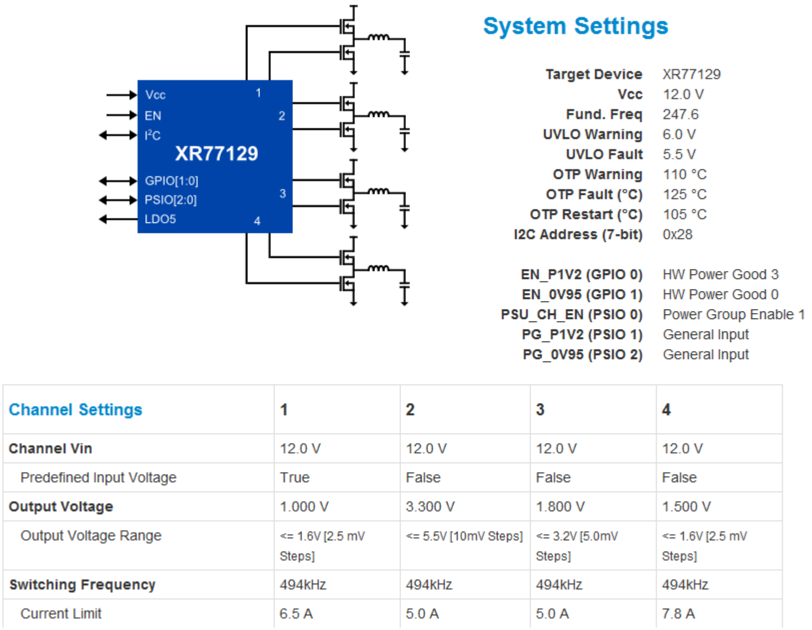
Figure 24: Power map

voltages and currents		
P0V9	0.9V	10A
P0V95	0.95V	31mA
P1V0	1.0V	3A
P1V2	1.2V	0.6A
P1V5	1.5V	7.5A
P1V8	1.8V	1.6A
P3V3	3.3 V	2A
P3V3MP	3.3V	0.18A
P5V0	5.0V	0.5A

Maximum RTM voltages and currents		
P12V0	12V	3A
P3V3MP_RTM	3.3V	30mA

13.2.2 Exar parameters

Exar chip(XR77129) has 4 configurable outputs with configirable current limits. Each channel can be confugured individually. It is possible to set voltage, current limit and power sequencing. In Figure 25 we can see that main power supply is 12V, Under Voltage Lockout(UVLO) is set to 6V, so below this value chip will shutdown all channels. When the temperature rise under Over Temperature Protection (OTP) 105 degrees, the chip will generate warning event and restart.
All 4 channells can be grouped together and will start-up and shut-down in an user defined sequwnce. Selecting none means channels will not be assigned to any group and therefore, will be controlled independently. Group 0 is controlled by ENABLE or PM_I2C command. Group 1 can be controlled by GPIO or by PM_I2C command. By selecting 'Wait PGOOD' next channel will not power up untill current channel reaches the target level. Delay is an additional delay time which postpone after power up one and another channel in group.
In on-going Exar configuration - Figure 26, power sequencing looks like this: After Enable from MMC P1V0 start-up, after it reaches proper value, Exar waits 10ms and turn ramp up another channel in this group - P1V8. In the same order is ramp up last channel in this group. Finally on command 'EN_PSU_CH' P3V3 ramp up.

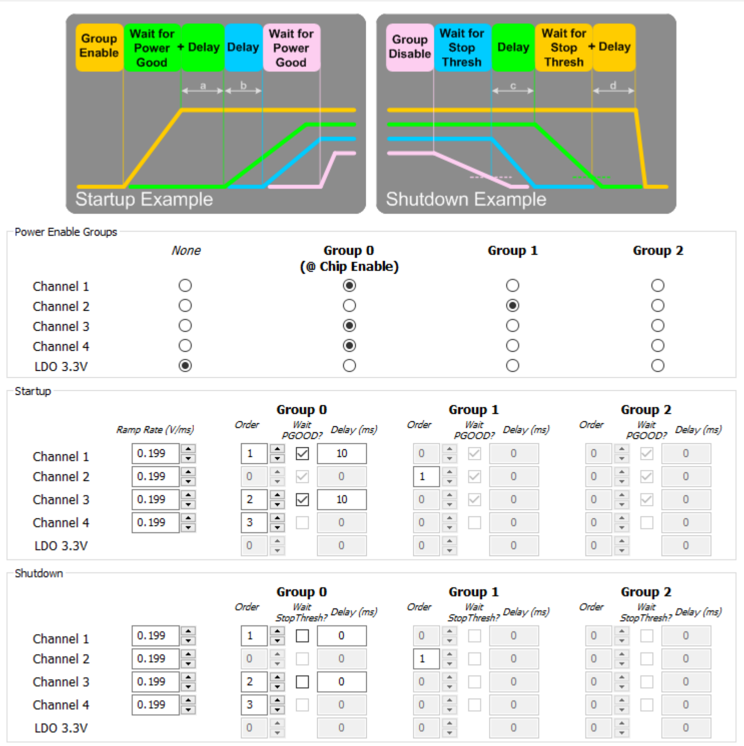


System Settings

Target Device	XR77129
Vcc	12.0 V
Fund. Freq	247.6
UVLO Warning	6.0 V
UVLO Fault	5.5 V
OTP Warning	110 °C
OTP Fault (°C)	125 °C
OTP Restart (°C)	105 °C
I2C Address (7-bit)	0x28
EN_P1V2 (GPIO 0)	HW Power Good 3
EN_0V95 (GPIO 1)	HW Power Good 0
PSU_CH_EN (PSIO 0)	Power Group Enable 1
PG_P1V2 (PSIO 1)	General Input
PG_0V95 (PSIO 2)	General Input

Channel Settings	1	2	3	4
Channel Vin	12.0 V	12.0 V	12.0 V	12.0 V
Predefined Input Voltage	True	False	False	False
Output Voltage	1.000 V	3.300 V	1.800 V	1.500 V
Output Voltage Range	<= 1.6V [2.5 mV Steps]	<= 5.5V [10mV Steps]	<= 3.2V [5.0mV Steps]	<= 1.6V [2.5 mV Steps]
Switching Frequency	494kHz	494kHz	494kHz	494kHz
Current Limit	6.5 A	5.0 A	5.0 A	7.8 A

Figure 25: Exar configuration



Power Enable Groups

	None	Group 0 (@ Chip Enable)	Group 1	Group 2
Channel 1	○	●	○	○
Channel 2	○	○	●	○
Channel 3	○	●	○	○
Channel 4	○	●	○	○
LDO 3.3V	●	○	○	○

Startup

		Group 0			Group 1			Group 2		
	Ramp Rate (V/ms)	*Order*	*Wait PGOOD?*	*Delay (ms)*	*Order*	*Wait PGOOD?*	*Delay (ms)*	*Order*	*Wait PGOOD?*	*Delay (ms)*
Channel 1	0.199	1	☑	10	0	☑	0	0	☑	0
Channel 2	0.199	0	☑	0	1	☑	0	0	☑	0
Channel 3	0.199	2	☑	10	0	☑	0	0	☑	0
Channel 4	0.199	3	□	0	0	□	0	0	□	0
LDO 3.3V		0		0	0		0	0		0

Shutdown

		Group 0			Group 1			Group 2		
		Order	*Wait StopThresh?*	*Delay (ms)*	*Order*	*Wait StopThresh?*	*Delay (ms)*	*Order*	*Wait StopThresh?*	*Delay (ms)*
Channel 1	0.199	1	□	0	0	□	0	0	□	0
Channel 2	0.199	0	□	0	1	□	0	0	□	0
Channel 3	0.199	2	□	0	0	□	0	0	□	0
Channel 4	0.199	3	□	0	0	□	0	0	□	0
LDO 3.3V		0		0	0		0	0		0

Figure 26: Exar power on delays

14 MMC

14.1 MMC steps during booting

- configures CPU, UART from own FLASH
- sets IO port directions
- enables VCCINT PSU
- enables P5V0 PSU (helper PSU)
- enables Exar PSU. It boots from its own EEPROM
- waits 200ms
- configures SCANSTA chip in stitcher mode. If RTM is inserted, it enables its JTAG port
- configures I2C switch base address for master ports - MMC, FPGA
- initializes default RTM power state to off
- initializes Ethernet PHY chip in RGMII mode using pin strap.
- waits 200ms
- initializes I2C controller and chain (switch)
- configures Si5324
- checks if RTM is inserted, if yes, then enables its power, waits 200ms and initializes RTM power supply via RTM_I2C. It also configures Si5324 on RTM
- runs task.

  The task performs following functions:

- checks if FPGA is configured. If not, it keeps Ethernet PHY in reset state. Once FPGA gets configured, it initializes the PHY.
- checks if RTM is unplugged. If not, it switches the power off to make sure it is off during hotplug.

Note:*Configuration CPU, UART does not affect with any changes of LED indicators.*

14.2 Bootstraping

To compile binaries LPCXpresso in newest version is needed.
LPCXpresso User Guide

Another option is to compile under Linux using cmake toolchain in version 4.9.3.

```
cmake & arm−none−eabi−gcc
```

- Header flashing

  The MMC can be upgraded by USB cable and NXP programmer(can be used other programmer but make sure that header shorts pins 3, 5, 9) using Flashmagic or any other software which can talk with NXP bootloader. The tested programmer is LPCLink V2. Flashing using programmer allows to debug.

- USB flashing The MMC can be upgraded using USB and flashmagic software. This option only allows to flash IC, without any debug option. Steps to flash using USB:

  – Set serial console 115200 8n1

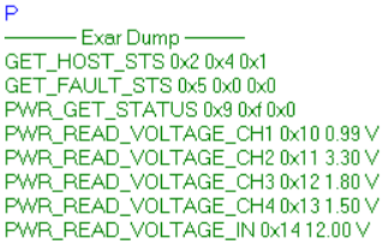
  – Press front-panel button -PB3 to trigger MMD to dump to serial console

  – Set LPC1776, 8MHz oscillator, select hex file and press start

The source code is written in C and can be found on github.
Source code: https://github.com/m-labs/sinara/tree/master/SAYMA_firmware.
pre-compiled binary: https://github.com/m-labs/mmc-firmware/releases

14.3 Exar debugging

In case of chip failure, i.g.overvoltage, overcurrent, etc., there is possibility to check chip status via UART. In this case in UART console, Exar register readout can be done by typing ’P’ character.

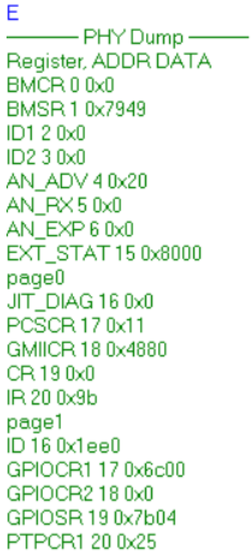


```
P
——— Exar Dump ———
GET_HOST_STS 0x2 0x4 0x1
GET_FAULT_STS 0x5 0x0 0x0
PWR_GET_STATUS 0x9 0xf 0x0
PWR_READ_VOLTAGE_CH1 0x10 0.99 V
PWR_READ_VOLTAGE_CH2 0x11 3.30 V
PWR_READ_VOLTAGE_CH3 0x12 1.80 V
PWR_READ_VOLTAGE_CH4 0x13 1.50 V
PWR_READ_VOLTAGE_IN 0x14 12.00 V
```

Figure 27: Exar register

14.4 PHY debugging

In case of chip failure, there is possibility to check chip status via UART. In this case in UART console, Ethernet PHY content can be read by typing ’E’ character



```
E
———— PHY Dump ————
Register, ADDR DATA
BMCR 0 0x0
BMSR 1 0x7949
ID1 2 0x0
ID2 3 0x0
AN_ADV 4 0x20
AN_RX 5 0x0
AN_EXP 6 0x0
EXT_STAT 15 0x8000
page0
JIT_DIAG 16 0x0
PCSCR 17 0x11
GMIICR 18 0x4880
CR 19 0x0
IR 20 0x9b
page1
ID 16 0x1ee0
GPIOCR1 17 0x6c00
GPIOCR2 18 0x0
GPIOSR 19 0x7b04
PTPCR1 20 0x25
```

Figure 28: Ethernet PHY register

14.5 Ethernet

There is no Ethernet sharing between MMC and FPGA due to speed difference between RGMII and RMII. PHY does not provide link speed translation.

14.6 OpenMMC

OpenMMC Project:https://github.com/lnls-dig/openMMC

TBD

15 Housekeeping Signals

Both MMC and FPGA can acces to any of I2C buses as is in Figure 9. MMC collects all data from all sensors connected to I2C bus. Then the data can be transfered via IPMI to MCH. For now MCH get only information about AMC and RTM to allow power supply.

15.1 sensors

Temperature:

No	Addr.	placement	Type	Accuracy
IC8	0x4B	NOR Flash	LM75	+/- 2
IC34	0x49	FPGA	LM75	+/- 2
IC35	0x4A	Under SFPs	LM75	+/- 2
IC36	0x4F	power section	LM75	+/- 2
IC37	0x24	middle of the board	MAX664A	+/- 1

All temperature sensors are tied tohether to one I2C bus - SENS_I2C.

Current:

No	Addr.	placement	Type	Accuracy
IC27	0x40	RTM_P12V0	INA219	+/- 0.2%
IC28	0x41	FMC_P12V0	INA219	+/- 0.2%

All current sensors are tied tohether to one I2C bus - PM_I2C.

15.2 Safety interlocks

Temperature interlock is available on RTM board only and gets activated after reaching 80 degrees. This is hardware interlock and cannot be deactivated. Dedicated LED (LD15) gets on and RTM power supply is off until the temperature is exceeded

16 Signal tables

In this section the more important signals tables are presented. The total signal table is in Appendix section.

SFP1		
FPGA signal	FPGA ball	Signal on the board
MGTHTXN0_224	AN3	SFP1TX_N
MGTHTXP0_224	AN4	SFP1TX_P
MGTHRXN0_224	AP1	SFP1RX_N
MGTHRXP0_224	AP2	SFP1RX_P
IO_L14N_T2L_N3_GC_64	AG9	SFP1_LED1
IO_T2U_N12_64	AJ10	SFP1_LED2
IO_L3P_T0L_N4_AD15P_64	AM11	SFP1_LOS
IO_L2N_T0L_N3_64	AP13	SFP1_MOD_DEF2
IO_L2P_T0L_N2_64	AN13	SFP1_MOD_DEF1
IO_L3N_T0L_N5_AD15N_64	AN11	SFP1_MOD_DEF0
IO_T0U_N12_64	AK11	SFP1_RATE_SELECT
IO_L1P_T0L_N0_DBC_64	AP11	SFP1_TX_DISABLE
IO_L1N_T0L_N1_DBC_64	AP10	SFP1_TX_FAULT

SFP2		
FPGA signal	FPGA ball	Signal on the board
MGTHTXN1_224	AM5	SFP2TX_N
MGTHTXP1_224	AM6	SFP2TX_P
MGTHRXN1_224	AM1	SFP2RX_N
MGTHRXP1_224	AM2	SFP2RX_P
IO_L8N_T1L_N3_AD5N_64	AJ13	SFP2_LED1
IO_L7P_T1L_N0_QBC_AD13P_64	AE13	SFP2_LED2
IO_L7N_T1L_N1_QBC_AD13N_64	AF13	SFP2_LOS
IO_L5P_T0U_N8_AD14P_64	AK12	SFP2_MOD_DEF2
IO_L6N_T0U_N11_AD6N_64	AL13	SFP2_MOD_DEF1
IO_L6P_T0U_N10_AD6P_64	AK13	SFP2_MOD_DEF0
IO_L5N_T0U_N9_AD14N_64	AL12	SFP2_RATE_SELECT
IO_L4P_T0U_N6_DBC_AD7P_64	AM12	SFP2_TX_DISABLE
IO_L4N_T0U_N7_DBC_AD7N_64	AN12	SFP2_TX_FAULT

AMC		
FP1		
FPGA signal	FPGA ball	Signal on the board
MGTHTXN2_224	AL3	TX4C_N
MGTHTXP2_224	AL4	TX4C_P
MGTHRXN2_224	AK1	RX4_N
MGTHRXP2_224	AK2	RX4_P
MGTHTXN3_224	AK5	TX5C_N

MGTHTXP3_224	AK6	TX5C_P
MGTHRXN3_224	AJ3	RX5_N
MGTHRXP3_224	AJ4	RX5_P
IO_L13N_T2L_N1_GC_QBC_45	AH17	TXC6_N
IO_L13P_T2L_N0_GC_QBC_45	AH18	TXC6_P
IO_L14N_T2L_N3_GC_45	AJ16	RXC6_N
IO_L14P_T2L_N2_GC_45	AH16	RXC6_P
IO_L6N_T0U_N11_AD6N_45	AP15	TXC7_N
IO_L6P_T0U_N10_AD6P_45	AP16	TXC7_P
IO_L7N_T1L_N1_QBC_AD13N_45	AM14	RXC7_N
IO_L7P_T1L_N0_QBC_AD13P_45	AL14	RXC7_P

FP2

FPGA signal	FPGA ball	Signal on the board
IO_L17N_T2U_N9_AD10N_66	K12	TXC8_N
IO_L17P_T2U_N8_AD10P_66	L12	TXC8_P
IO_L18N_T2U_N11_AD2N_66	H13	RXC8_N
IO_L18P_T2U_N10_AD2P_66	J13	RXC8_P
IO_L15P_T2L_N4_AD11P_66	K11	TXC9_N
IO_L15N_T2L_N5_AD11N_66	J11	TXC9_P
IO_L16N_T2U_N7_QBC_AD3N_66	K13	RXC9_N
IO_L16P_T2U_N6_QBC_AD3P_66	L13	RXC9_P
IO_L4N_T0U_N7_DBC_AD7N_45	AN17	TXC10_N
IO_L4P_T0U_N6_DBC_AD7P_45	AN18	TXC10_P
IO_L5N_T0U_N9_AD14N_45	AM15	RXC10_N
IO_L5P_T0U_N8_AD14P_45	AM16	RXC10_P
IO_L2N_T0L_N3_45	AP18	TXC11_N
IO_L2P_T0L_N2_45	AN19	TXC11_P
IO_L3N_T0L_N5_AD15N_45	AN16	RXC11_N
IO_L3P_T0L_N4_AD15P_45	AM17	RXC11_P

P2P

FPGA signal	FPGA ball	Signal on the board
IO_L4N_T0U_N7_DBC_AD7N_47	AC27	TXC12_N
IO_L4P_T0U_N6_DBC_AD7P_47	AC26	TXC12_P
IO_L5N_T0U_N9_AD14N_47	AB27	RXC12_N
IO_L5P_T0U_N8_AD14P_47	AA27	RXC12_P
IO_L2N_T0L_N3_47	AD26	TXC13_N
IO_L2P_T0L_N2_47	AD25	TXC13_P
IO_L3N_T0L_N5_AD15N_47	AC24	RXC13_N
IO_L3P_T0L_N4_AD15P_47	AB24	RXC13_P
IO_L5N_T0U_N9_AD14N_48	AE30	TXC14_N
IO_L5P_T0U_N8_AD14P_48	AD29	TXC14_P
IO_L6N_T0U_N11_AD6N_48	AG30	RXC14_N
IO_L6P_T0U_N10_AD6P_48	AF30	RXC14_P
IO_L3N_T0L_N5_AD15N_48	AD28	TXC15_N
IO_L3P_T0L_N4_AD15P_48	AC28	TXC15_P
IO_L4N_T0U_N7_DBC_AD7N_48	AG29	RXC15_N
IO_L4P_T0U_N6_DBC_AD7P_48	AF29	RXC15_P

RTM

FPGA signal	FPGA ball	Signal on the board
MGTHTXP2_228	C4	GTP1TXC_P
MGTHRXN2_228	B1	GTP1RX_N
MGTHRXP2_228	B2	GTP1RX_P
MGTHTXN1_228	D5	GTP2TXC_N
MGTHTXP1_228	D6	GTP2TXC_P
MGTHRXN1_228	D1	GTP2RX_N
MGTHRXP1_228	D2	GTP2RX_P
MGTHTXN0_228	F5	GTP3TXC_N
MGTHTXP0_228	F6	GTP3TXC_P
MGTHRXN0_228	E3	GTP3RX_N
MGTHRXP0_228	E4	GTP3RX_P
MGTHTXN3_227	G3	GTP4TXC_N
MGTHTXP3_227	G4	GTP4TXC_P
MGTHRXN3_227	F1	GTP4RX_N
MGTHRXP3_227	F2	GTP4RX_P
MGTHTXN2_227	J3	GTP5TXC_N
MGTHTXP2_227	J4	GTP5TXC_P
MGTHRXN2_227	H1	GTP5RX_N
MGTHRXP2_227	H2	GTP5RX_P
MGTHTXN1_227	L3	GTP6TXC_N
MGTHTXP1_227	L4	GTP6TXC_P
MGTHRXN1_227	K1	GTP6RX_N
MGTHRXP1_227	K2	GTP6RX_P
MGTHTXN0_227	N3	GTP7TXC_N
MGTHTXP0_227	N4	GTP7TXC_P
MGTHRXN0_227	M1	GTP7RX_N
MGTHRXP0_227	M2	GTP7RX_P
MGTHTXN3_226	R3	GTP8TXC_N
MGTHTXP3_226	R4	GTP8TXC_P
MGTHRXN3_226	P1	GTP8RX_N
MGTHRXP3_226	P2	GTP8RX_P
MGTHTXN2_226	U3	GTP9TXC_N
MGTHTXP2_226	U4	GTP9TXC_P
MGTHRXN2_226	T1	GTP9RX_N
MGTHRXP2_226	T2	GTP9RX_P
MGTHTXN1_226	W3	GTP10TXC_N
MGTHTXP1_226	W4	GTP10TXC_P
MGTHRXN1_226	V1	GTP10RX_N
MGTHRXP1_226	V2	GTP10RX_P
MGTHTXN0_226	AA3	GTP11TXC_N
MGTHTXP0_226	AA4	GTP11TXC_P
MGTHTXN3_225	AC3	GTP12TXC_N
MGTHTXP3_225	AC4	GTP12TXC_P
MGTHRXN3_225	AB1	GTP12RX_N
MGTHRXP3_225	AB2	GTP12RX_P
MGTHTXN2_225	AE3	GTP13TXC_N
MGTHTXP2_225	AE4	GTP13TXC_P
MGTHRXN2_225	AD1	GTP13RX_N
MGTHRXP2_225	AD2	GTP13RX_P
MGTHTXN1_225	AG3	GTP14TXC_N
MGTHTXP1_225	AG4	GTP14TXC_P

MGTHRXN1_225	AF1	GTP14RX_N
MGTHRXP1_225	AF2	GTP14RX_P
MGTHTXN0_225	AH5	GTP15TXC_N
MGTHTXP0_225	AH6	GTP15TXC_P
MGTHRXN0_225	AH1	GTP15RX_N
MGTHRXP0_225	AH2	GTP15RX_P

17 Factory acceptance testing

TBD

A Total table of sugnals

FPGA signal	FPGA ball	Signal on the board
CCLK_0	AA9	FPGA_CCLK
CFGBVS_0	W7	P3V3
D00_MOSI_0	AC7	QSPI0_IO0
D01_DIN_0	AB7	QSPI0_IO1
D02_0	AA7	QSPI0_IO2
D03_0	Y7	QSPI0_IO3
DONE_0	N7	FPGA_DONE
DXN	Y11	DXN
DXP	Y12	DXP
GND-1	A2	GND
GND-1	T20	GND
GND-10	A32	GND
GND-10	U5	GND
GND-100	AJ1	GND
GND-100	N9	GND
GND-101	AJ2	GND
GND-101	N11	GND
GND-102	AJ5	GND
GND-102	N13	GND
GND-103	AJ7	GND
GND-103	N15	GND
GND-104	AJ17	GND
GND-104	N17	GND
GND-105	AJ27	GND
GND-105	N19	GND
GND-106	AK3	GND
GND-106	N25	GND
GND-107	AK4	GND
GND-107	N28	GND
GND-108	AK7	GND
GND-108	N32	GND
GND-109	AK14	GND
GND-109	P3	GND
GND-11	B3	GND
GND-11	Y8	GND
GND-110	AK24	GND
GND-110	P4	GND
GND-111	AK34	GND
GND-111	P10	GND
GND-112	AL1	GND
GND-112	P12	GND
GND-113	AL5	GND
GND-113	P14	GND
GND-114	AL7	GND
GND-114	P16	GND
GND-115	AL11	GND
GND-115	P18	GND
GND-116	AL21	GND
GND-116	P22	GND

GND-117	AL31	GND
GND-117	P28	GND
GND-118	AM4	GND
GND-118	P30	GND
GND-119	AM7	GND
GND-119	P34	GND
GND-12	B4	GND
GND-12	U13	GND
GND-120	AM8	GND
GND-120	R1	GND
GND-121	AM18	GND
GND-121	R5	GND
GND-122	AM28	GND
GND-122	R9	GND
GND-123	AN1	GND
GND-123	R11	GND
GND-124	AN2	GND
GND-124	R13	GND
GND-125	AN5	GND
GND-125	R15	GND
GND-126	AN7	GND
GND-126	R17	GND
GND-127	AN15	GND
GND-127	R19	GND
GND-128	AN25	GND
GND-128	R28	GND
GND-129	AP3	GND
GND-129	R31	GND
GND-13	B7	GND
GND-13	U15	GND
GND-130	AP4	GND
GND-130	T4	GND
GND-131	AP7	GND
GND-131	T8	GND
GND-132	AP12	GND
GND-132	T10	GND
GND-133	AP22	GND
GND-133	T12	GND
GND-134	AP32	GND
GND-134	T14	GND
GND-135	T16	GND
GND-136	T18	GND
GND-14	B8	GND
GND-14	U17	GND
GND-15	B18	GND
GND-15	U19	GND
GND-16	B28	GND
GND-16	U23	GND
GND-17	B30	GND
GND-17	U30	GND
GND-18	B33	GND
GND-18	U31	GND
GND-19	B34	GND

GND-19	U32	GND
GND-2	A1	GND
GND-2	T26	GND
GND-20	C1	GND
GND-20	U33	GND
GND-21	C5	GND
GND-21	V3	GND
GND-22	C7	GND
GND-22	V4	GND
GND-23	C15	GND
GND-23	V10	GND
GND-24	C25	GND
GND-24	V14	GND
GND-25	C30	GND
GND-25	V16	GND
GND-26	C31	GND
GND-26	V18	GND
GND-27	D4	GND
GND-27	V20	GND
GND-28	D7	GND
GND-28	V30	GND
GND-29	D12	GND
GND-29	W1	GND
GND-3	A5	GND
GND-3	T28	GND
GND-30	D22	GND
GND-30	W5	GND
GND-31	AB8	GND
GND-31	D30	GND
GND-32	D34	GND
GND-32	W13	GND
GND-33	E1	GND
GND-33	W15	GND
GND-34	E2	GND
GND-34	W17	GND
GND-35	E5	GND
GND-35	W19	GND
GND-36	E7	GND
GND-36	W27	GND
GND-37	E9	GND
GND-37	Y4	GND
GND-38	E19	GND
GND-38	Y10	GND
GND-39	E29	GND
GND-39	Y14	GND
GND-4	A6	GND
GND-4	T29	GND
GND-40	E30	GND
GND-40	Y16	GND
GND-41	E32	GND
GND-41	Y18	GND
GND-42	F3	GND
GND-42	Y20	GND

GND-43	F4	GND
GND-43	Y24	GND
GND-44	F7	GND
GND-44	Y34	GND
GND-45	AA1	GND
GND-45	F16	GND
GND-46	AA2	GND
GND-46	F26	GND
GND-47	AA5	GND
GND-47	F28	GND
GND-48	F29	GND
GND-48	P8	GND
GND-49	AA11	GND
GND-49	F33	GND
GND-5	A7	GND
GND-5	T30	GND
GND-50	AA13	GND
GND-50	F34	GND
GND-51	AA15	GND
GND-51	G1	GND
GND-52	AA17	GND
GND-52	G5	GND
GND-53	AA19	GND
GND-53	G7	GND
GND-54	AA21	GND
GND-54	G13	GND
GND-55	AA31	GND
GND-55	G23	GND
GND-56	AB3	GND
GND-56	G28	GND
GND-57	AB4	GND
GND-57	G31	GND
GND-58	AB10	GND
GND-58	H4	GND
GND-59	AB12	GND
GND-59	H7	GND
GND-6	A11	GND
GND-6	T33	GND
GND-60	AB14	GND
GND-60	H10	GND
GND-61	AB16	GND
GND-61	H20	GND
GND-62	AB18	GND
GND-62	H28	GND
GND-63	AB28	GND
GND-63	H30	GND
GND-64	AC1	GND
GND-64	H34	GND
GND-65	AC5	GND
GND-65	J1	GND
GND-66	J2	GND
GND-66	V8	GND
GND-67	AC11	GND

GND-67	J5	GND
GND-68	AC13	GND
GND-68	J7	GND
GND-69	AC15	GND
GND-69	J17	GND
GND-7	A21	GND
GND-7	T34	GND
GND-70	AC17	GND
GND-70	J27	GND
GND-71	AC19	GND
GND-71	J28	GND
GND-72	AC25	GND
GND-72	J32	GND
GND-73	AD4	GND
GND-73	K3	GND
GND-74	AD12	GND
GND-74	K4	GND
GND-75	AD22	GND
GND-75	K14	GND
GND-76	AD32	GND
GND-76	K24	GND
GND-77	AE1	GND
GND-77	K28	GND
GND-78	AE2	GND
GND-78	K30	GND
GND-79	AE5	GND
GND-79	K34	GND
GND-8	A30	GND
GND-8	U1	GND
GND-80	AE7	GND
GND-80	L1	GND
GND-81	AE9	GND
GND-81	L5	GND
GND-82	AE19	GND
GND-82	L11	GND
GND-83	AE29	GND
GND-83	L21	GND
GND-84	AF3	GND
GND-84	L28	GND
GND-85	AF4	GND
GND-85	L31	GND
GND-86	AF7	GND
GND-86	M4	GND
GND-87	AF16	GND
GND-87	M8	GND
GND-88	AF26	GND
GND-88	M10	GND
GND-89	AG1	GND
GND-89	M12	GND
GND-9	A31	GND
GND-9	U2	GND
GND-90	AG5	GND
GND-90	M14	GND

GND-91	AG7	GND
GND-91	M16	GND
GND-92	AG13	GND
GND-92	M18	GND
GND-93	AG23	GND
GND-93	M28	GND
GND-94	AG33	GND
GND-94	M30	GND
GND-95	AH4	GND
GND-95	M33	GND
GND-96	AH7	GND
GND-96	M34	GND
GND-97	AH10	GND
GND-97	N1	GND
GND-98	AH20	GND
GND-98	N2	GND
GND-99	AH30	GND
GND-99	N5	GND
GNDADC	U11	GND
INIT_B_0	V7	FPGA_INIT_B
IO_L10N_T1U_N7_QBC_AD4N_44	AJ25	DDR3_64_DQS1_N
IO_L10N_T1U_N7_QBC_AD4N_45	AL17	DDR3_64_A1
IO_L10N_T1U_N7_QBC_AD4N_46	AP30	DDR3_64_DQS5_N
IO_L10N_T1U_N7_QBC_AD4N_47	AC21	FMC1_LA23_N
IO_L10N_T1U_N7_QBC_AD4N_48	AF34	FMC1_LA05_N
IO_L10N_T1U_N7_QBC_AD4N_64	AE11	DIO6
IO_L10N_T1U_N7_QBC_AD4N_66	J10	SYNCOUT21_N
IO_L10N_T1U_N7_QBC_AD4N_67	A24	DDR3_32_DQS1_N
IO_L10N_T1U_N7_QBC_AD4N_68	D18	DDR3_32_A10
IO_L10N_T1U_N7_QBC_AD4N_A13_D29_65	K23	SMA_IO1
IO_L10P_T1U_N6_QBC_AD4P_44	AH24	DDR3_64_DQS1_P
IO_L10P_T1U_N6_QBC_AD4P_45	AL18	DDR3_64_A10
IO_L10P_T1U_N6_QBC_AD4P_46	AN29	DDR3_64_DQS5_P
IO_L10P_T1U_N6_QBC_AD4P_47	AB21	FMC1_LA23_P
IO_L10P_T1U_N6_QBC_AD4P_48	AE33	FMC1_LA05_P
IO_L10P_T1U_N6_QBC_AD4P_64	AD11	DIO5
IO_L10P_T1U_N6_QBC_AD4P_66	K10	SYNCOUT21_P
IO_L10P_T1U_N6_QBC_AD4P_67	B24	DDR3_32_DQS1_P
IO_L10P_T1U_N6_QBC_AD4P_68	D19	NC
IO_L10P_T1U_N6_QBC_AD4P_A12_D28_65	J22	SI5324_INT_ALM
IO_L11N_T1U_N9_GC_44	AJ24	DDR3_64_DQ8
IO_L11N_T1U_N9_GC_45	AK18	DDR3_64_CAS_N
IO_L11N_T1U_N9_GC_46	AM29	DDR3_64_DQ46
IO_L11N_T1U_N9_GC_47	AA23	FMC1_CLK1_M2C_N
IO_L11N_T1U_N9_GC_48	AD31	FMC1_LA08_N
IO_L11N_T1U_N9_GC_64	AH12	DIO4
IO_L11N_T1U_N9_GC_66	F9	ADC1_SYNC_N
IO_L11N_T1U_N9_GC_67	D25	DDR3_32_DQ8
IO_L11N_T1U_N9_GC_68	D16	DDR3_32_WE_N
IO_L11N_T1U_N9_GC_A11_D27_65	M26	TCKC_D_N
IO_L11P_T1U_N8_GC_44	AJ23	DDR3_64_DQ10
IO_L11P_T1U_N8_GC_45	AJ18	DDR3_64_ODT
IO_L11P_T1U_N8_GC_46	AL29	DDR3_64_DQ44

IO_L11P_T1U_N8_GC_47	Y23	FMC1_CLK1_M2C_P
IO_L11P_T1U_N8_GC_48	AD30	FMC1_LA08_P
IO_L11P_T1U_N8_GC_64	AG12	DIO3
IO_L11P_T1U_N8_GC_66	G9	ADC1_SYNC_P
IO_L11P_T1U_N8_GC_67	E25	DDR3_32_DQ10
IO_L11P_T1U_N8_GC_68	E16	DDR3_32_CAS_N
IO_L11P_T1U_N8_GC_A10_D26_65	M25	TCKC_D_P
IO_L12N_T1U_N11_GC_44	AH23	DDR3_64_DQ12
IO_L12N_T1U_N11_GC_45	AK16	SYSCLK_300_N
IO_L12N_T1U_N11_GC_46	AM30	DDR3_64_DQ42
IO_L12N_T1U_N11_GC_47	AA25	FMC1_CLK0_M2C_N
IO_L12N_T1U_N11_GC_48	AC32	FMC1_GBTCLK0_M2C_N
IO_L12N_T1U_N11_GC_64	AH11	DIO1
IO_L12N_T1U_N11_GC_66	F10	NC
IO_L12N_T1U_N11_GC_67	C24	DDR3_32_DQ12
IO_L12N_T1U_N11_GC_68	E17	NC
IO_L12N_T1U_N11_GC_A09_D25_65	M24	TCKC_C_N
IO_L12P_T1U_N10_GC_44	AH22	DDR3_64_DQ14
IO_L12P_T1U_N10_GC_45	AK17	SYSCLK_300_P
IO_L12P_T1U_N10_GC_46	AL30	DDR3_64_DQ40
IO_L12P_T1U_N10_GC_47	AA24	FMC1_CLK0_M2C_P
IO_L12P_T1U_N10_GC_48	AC31	FMC1_GBTCLK0_M2C_P
IO_L12P_T1U_N10_GC_64	AG11	DIO0
IO_L12P_T1U_N10_GC_66	G10	NC
IO_L12P_T1U_N10_GC_67	D24	DDR3_32_DQ14
IO_L12P_T1U_N10_GC_68	E18	DDR3_32_RAS_N
IO_L12P_T1U_N10_GC_A08_D24_65	N24	TCKC_C_P
IO_L13N_T2L_N1_GC_QBC_44	AK21	HW_ID3
IO_L13N_T2L_N1_GC_QBC_45	AH17	TXC6_N
IO_L13N_T2L_N1_GC_QBC_46	AK30	NC
IO_L13N_T2L_N1_GC_QBC_47	W24	FMC1_LA17_CC_N
IO_L13N_T2L_N1_GC_QBC_48	AB32	FMC1_LA00_CC_N
IO_L13N_T2L_N1_GC_QBC_64	AG10	AMC_MASTER_AUX_CLK_N
IO_L13N_T2L_N1_GC_QBC_66	G11	SYNCOUT11_N
IO_L13N_T2L_N1_GC_QBC_67	C23	NC
IO_L13N_T2L_N1_GC_QBC_68	G16	DDR3_32_BA2
IO_L13N_T2L_N1_GC_QBC_A07_D23_65	N26	TCKC_B_N
IO_L13P_T2L_N0_GC_QBC_44	AJ21	DDR3_64_DM2
IO_L13P_T2L_N0_GC_QBC_45	AH18	TXC6_P
IO_L13P_T2L_N0_GC_QBC_46	AJ29	DDR3_64_DM6
IO_L13P_T2L_N0_GC_QBC_47	W23	FMC1_LA17_CC_P
IO_L13P_T2L_N0_GC_QBC_48	AA32	FMC1_LA00_CC_P
IO_L13P_T2L_N0_GC_QBC_64	AF10	AMC_MASTER_AUX_CLK_P
IO_L13P_T2L_N0_GC_QBC_66	H11	SYNCOUT11_P
IO_L13P_T2L_N0_GC_QBC_67	D23	DDR3_32_DM2
IO_L13P_T2L_N0_GC_QBC_68	G17	NC
IO_L13P_T2L_N0_GC_QBC_A06_D22_65	P26	TCKC_B_P
IO_L14N_T2L_N3_GC_44	AK23	DDR3_64_DQ19
IO_L14N_T2L_N3_GC_45	AJ16	RXC6_N
IO_L14N_T2L_N3_GC_46	AK32	DDR3_64_DQ55
IO_L14N_T2L_N3_GC_47	Y25	FMC1_LA18_CC_N
IO_L14N_T2L_N3_GC_48	AB31	FMC1_LA01_CC_N
IO_L14N_T2L_N3_GC_64	AG9	SFP1_LED1

IO_L14N_T2L_N3_GC_66	G12	CDR_CLK_CLEAN1_N
IO_L14N_T2L_N3_GC_67	E23	DDR3_32_DQ19
IO_L14N_T2L_N3_GC_68	F17	SYSCLK1_300_N
IO_L14N_T2L_N3_GC_A05_D21_65	P25	TCKC_A_N
IO_L14P_T2L_N2_GC_44	AK22	DDR3_64_DQ20
IO_L14P_T2L_N2_GC_45	AH16	RXC6_P
IO_L14P_T2L_N2_GC_46	AK31	DDR3_64_DQ51
IO_L14P_T2L_N2_GC_47	W25	FMC1_LA18_CC_P
IO_L14P_T2L_N2_GC_48	AB30	FMC1_LA01_CC_P
IO_L14P_T2L_N2_GC_64	AF9	CLK_50M
IO_L14P_T2L_N2_GC_66	H12	CDR_CLK_CLEAN1_P
IO_L14P_T2L_N2_GC_67	E22	DDR3_32_DQ20
IO_L14P_T2L_N2_GC_68	F18	SYSCLK1_300_P
IO_L14P_T2L_N2_GC_A04_D20_65	P24	TCKC_A_P
IO_L15N_T2L_N5_AD11N_44	AM20	DDR3_64_DQ18
IO_L15N_T2L_N5_AD11N_45	AG16	DDR3_64_A2
IO_L15N_T2L_N5_AD11N_46	AJ31	DDR3_64_DQ52
IO_L15N_T2L_N5_AD11N_47	U22	FMC1_LA22_N
IO_L15N_T2L_N5_AD11N_48	AD34	FMC1_LA06_N
IO_L15N_T2L_N5_AD11N_64	AF8	NC
IO_L15N_T2L_N5_AD11N_66	J11	TXC9_P
IO_L15N_T2L_N5_AD11N_67	B22	DDR3_32_DQ18
IO_L15N_T2L_N5_AD11N_68	G14	DDR3_32_A13
IO_L15N_T2L_N5_AD11N_A03_D19_65	R27	RGMII2_MDIO
IO_L15P_T2L_N4_AD11P_44	AL20	DDR3_64_DQ22
IO_L15P_T2L_N4_AD11P_45	AG17	DDR3_64_A3
IO_L15P_T2L_N4_AD11P_46	AJ30	DDR3_64_DQ53
IO_L15P_T2L_N4_AD11P_47	U21	FMC1_LA22_P
IO_L15P_T2L_N4_AD11P_48	AC34	FMC1_LA06_P
IO_L15P_T2L_N4_AD11P_64	AE8	MLVDS_FSEN
IO_L15P_T2L_N4_AD11P_66	K11	TXC9_N
IO_L15P_T2L_N4_AD11P_67	B21	DDR3_32_DQ22
IO_L15P_T2L_N4_AD11P_68	G15	DDR3_32_A11
IO_L15P_T2L_N4_AD11P_A02_D18_65	T27	RGMII2_MDC
IO_L16N_T2U_N7_QBC_AD3N_44	AK20	DDR3_64_DQS2_N
IO_L16N_T2U_N7_QBC_AD3N_45	AJ14	DDR3_64_RST_N
IO_L16N_T2U_N7_QBC_AD3N_46	AJ33	DDR3_64_DQS6_N
IO_L16N_T2U_N7_QBC_AD3N_47	V23	FMC1_LA19_N
IO_L16N_T2U_N7_QBC_AD3N_48	AB29	FMC1_LA11_N
IO_L16N_T2U_N7_QBC_AD3N_64	AE10	NC
IO_L16N_T2U_N7_QBC_AD3N_66	K13	RXC9_N
IO_L16N_T2U_N7_QBC_AD3N_67	C22	DDR3_32_DQS2_N
IO_L16N_T2U_N7_QBC_AD3N_68	F19	DDR3_32_ODT
IO_L16N_T2U_N7_QBC_AD3N_A01_D17_65	T25	RGMII2_RX_CLK
IO_L16P_T2U_N6_QBC_AD3P_44	AJ20	DDR3_64_DQS2_P
IO_L16P_T2U_N6_QBC_AD3P_45	AJ15	DDR3_64_A13
IO_L16P_T2U_N6_QBC_AD3P_46	AH33	DDR3_64_DQS6_P
IO_L16P_T2U_N6_QBC_AD3P_47	V22	FMC1_LA19_P
IO_L16P_T2U_N6_QBC_AD3P_48	AA29	FMC1_LA11_P
IO_L16P_T2U_N6_QBC_AD3P_64	AD10	RE_DE_RX_P20
IO_L16P_T2U_N6_QBC_AD3P_66	L13	RXC9_P
IO_L16P_T2U_N6_QBC_AD3P_67	C21	DDR3_32_DQS2_P
IO_L16P_T2U_N6_QBC_AD3P_68	G19	DDR3_32_CS_N

IO_L16P_T2U_N6_QBC_AD3P_A00_D16_65	T24	RGMII2_RX_CTL
IO_L17N_T2U_N9_AD10N_44	AL23	DDR3_64_DQ23
IO_L17N_T2U_N9_AD10N_45	AH19	DDR3_64_RAS_N
IO_L17N_T2U_N9_AD10N_46	AH32	DDR3_64_DQ49
IO_L17N_T2U_N9_AD10N_47	T23	FMC1_LA20_N
IO_L17N_T2U_N9_AD10N_48	AB34	FMC1_LA10_N
IO_L17N_T2U_N9_AD10N_64	AD8	RE_DE_RX_P19
IO_L17N_T2U_N9_AD10N_66	K12	TXC8_N
IO_L17N_T2U_N9_AD10N_67	A20	DDR3_32_DQ23
IO_L17N_T2U_N9_AD10N_68	H16	DDR3_32_A4
IO_L17N_T2U_N9_AD10N_D15_65	R26	RGMII2_RXD3
IO_L17P_T2U_N8_AD10P_44	AL22	DDR3_64_DQ16
IO_L17P_T2U_N8_AD10P_45	AG19	DDR3_64_WE_N
IO_L17P_T2U_N8_AD10P_46	AH31	DDR3_64_DQ48
IO_L17P_T2U_N8_AD10P_47	T22	FMC1_LA20_P
IO_L17P_T2U_N8_AD10P_48	AA34	FMC1_LA10_P
IO_L17P_T2U_N8_AD10P_64	AD9	RE_DE_RX_P18
IO_L17P_T2U_N8_AD10P_66	L12	TXC8_P
IO_L17P_T2U_N8_AD10P_67	B20	DDR3_32_DQ16
IO_L17P_T2U_N8_AD10P_68	H17	DDR3_32_BA1
IO_L17P_T2U_N8_AD10P_D14_65	R25	RGMII2_RXD2
IO_L18N_T2U_N11_AD2N_44	AL25	DDR3_64_DQ17
IO_L18N_T2U_N11_AD2N_45	AG14	DDR3_64_A14
IO_L18N_T2U_N11_AD2N_46	AJ34	DDR3_64_DQ50
IO_L18N_T2U_N11_AD2N_47	W21	FMC1_LA21_N
IO_L18N_T2U_N11_AD2N_48	AD33	FMC1_LA07_N
IO_L18N_T2U_N11_AD2N_64	AH8	RE_DE_RX_P17
IO_L18N_T2U_N11_AD2N_66	H13	RXC8_N
IO_L18N_T2U_N11_AD2N_67	D21	DDR3_32_DQ17
IO_L18N_T2U_N11_AD2N_68	H18	DDR3_32_CKE
IO_L18N_T2U_N11_AD2N_D13_65	P23	RGMII2_RXD1
IO_L18P_T2U_N10_AD2P_44	AL24	DDR3_64_DQ21
IO_L18P_T2U_N10_AD2P_45	AG15	NC
IO_L18P_T2U_N10_AD2P_46	AH34	DDR3_64_DQ54
IO_L18P_T2U_N10_AD2P_47	V21	FMC1_LA21_P
IO_L18P_T2U_N10_AD2P_48	AC33	FMC1_LA07_P
IO_L18P_T2U_N10_AD2P_64	AH9	RE_DE_TX_P20
IO_L18P_T2U_N10_AD2P_66	J13	RXC8_P
IO_L18P_T2U_N10_AD2P_67	D20	DDR3_32_DQ21
IO_L18P_T2U_N10_AD2P_68	H19	NC
IO_L18P_T2U_N10_AD2P_D12_65	R23	RGMII2_RXD0
IO_L19N_T3L_N1_DBC_AD9N_44	AN21	HW_ID1
IO_L19N_T3L_N1_DBC_AD9N_45	AD18	DDR3_64_BA2
IO_L19N_T3L_N1_DBC_AD9N_46	AL33	NC
IO_L19N_T3L_N1_DBC_AD9N_47	V28	FMC1_LA33_N
IO_L19N_T3L_N1_DBC_AD9N_48	Y33	FMC1_LA13_N
IO_L19N_T3L_N1_DBC_AD9N_64	AM10	RE_DE_TX_P19
IO_L19N_T3L_N1_DBC_AD9N_66	D11	NC
IO_L19N_T3L_N1_DBC_AD9N_67	F25	NC
IO_L19N_T3L_N1_DBC_AD9N_68	J14	NC
IO_L19N_T3L_N1_DBC_AD9N_D11_65	M22	RGMII2_TX_CLK
IO_L19P_T3L_N0_DBC_AD9P_44	AM21	DDR3_64_DM3
IO_L19P_T3L_N0_DBC_AD9P_45	AD19	DDR3_64_BA1

IO_L19P_T3L_N0_DBC_AD9P_46	AL32	DDR3_64_DM7
IO_L19P_T3L_N0_DBC_AD9P_47	V27	FMC1_LA33_P
IO_L19P_T3L_N0_DBC_AD9P_48	W33	FMC1_LA13_P
IO_L19P_T3L_N0_DBC_AD9P_64	AL10	RE_DE_TX_P18
IO_L19P_T3L_N0_DBC_AD9P_66	E11	NC
IO_L19P_T3L_N0_DBC_AD9P_67	G24	DDR3_32_DM3
IO_L19P_T3L_N0_DBC_AD9P_68	J15	DDR3_32_A7
IO_L19P_T3L_N0_DBC_AD9P_D10_65	N22	RGMII2_TX_CTL
IO_L1N_T0L_N1_DBC_44	AE21	NC
IO_L1N_T0L_N1_DBC_45	AP14	NC
IO_L1N_T0L_N1_DBC_46	AJ26	NC
IO_L1N_T0L_N1_DBC_47	Y27	NC
IO_L1N_T0L_N1_DBC_48	AF27	FMC1_DP0_M2C_N
IO_L1N_T0L_N1_DBC_64	AP10	SFP1_TX_FAULT
IO_L1N_T0L_N1_DBC_66	E8	RTM_FPGA_GTP_RxC1_N
IO_L1N_T0L_N1_DBC_67	E27	NC
IO_L1N_T0L_N1_DBC_68	A14	NC
IO_L1N_T0L_N1_DBC_RS1_65	G27	USR_UART_N
IO_L1P_T0L_N0_DBC_44	AD21	DDR3_64_DM0
IO_L1P_T0L_N0_DBC_45	AN14	NC
IO_L1P_T0L_N0_DBC_46	AH26	DDR3_64_DM4
IO_L1P_T0L_N0_DBC_47	Y26	NC
IO_L1P_T0L_N0_DBC_48	AE27	FMC1_DP0_M2C_P
IO_L1P_T0L_N0_DBC_64	AP11	SFP1_TX_DISABLE
IO_L1P_T0L_N0_DBC_66	F8	RTM_FPGA_GTP_RxC1_P
IO_L1P_T0L_N0_DBC_67	F27	DDR3_32_DM0
IO_L1P_T0L_N0_DBC_68	B14	NC
IO_L1P_T0L_N0_DBC_RS0_65	H27	USR_UART_P
IO_L20N_T3L_N3_AD1N_44	AN22	DDR3_64_DQ29
IO_L20N_T3L_N3_AD1N_45	AF14	DDR3_64_A8
IO_L20N_T3L_N3_AD1N_46	AP33	DDR3_64_DQ57
IO_L20N_T3L_N3_AD1N_47	U25	FMC1_LA32_N
IO_L20N_T3L_N3_AD1N_48	Y30	FMC1_LA16_N
IO_L20N_T3L_N3_AD1N_64	AP9	RE_DE_TX_P17
IO_L20N_T3L_N3_AD1N_66	B12	RTM_FPGA_LVDS2_N
IO_L20N_T3L_N3_AD1N_67	E21	DDR3_32_DQ29
IO_L20N_T3L_N3_AD1N_68	K17	DDR3_32_A5
IO_L20N_T3L_N3_AD1N_D09_65	P21	RGMII2_TXD3
IO_L20P_T3L_N2_AD1P_44	AM22	DDR3_64_DQ31
IO_L20P_T3L_N2_AD1P_45	AF15	DDR3_64_A9
IO_L20P_T3L_N2_AD1P_46	AN33	DDR3_64_DQ56
IO_L20P_T3L_N2_AD1P_47	U24	FMC1_LA32_P
IO_L20P_T3L_N2_AD1P_48	W30	FMC1_LA16_P
IO_L20P_T3L_N2_AD1P_64	AN9	IO_RX_P20
IO_L20P_T3L_N2_AD1P_66	C12	RTM_FPGA_LVDS2_P
IO_L20P_T3L_N2_AD1P_67	E20	DDR3_32_DQ31
IO_L20P_T3L_N2_AD1P_68	K18	DDR3_32_A3
IO_L20P_T3L_N2_AD1P_D08_65	P20	RGMII2_TXD2
IO_L21N_T3L_N5_AD8N_44	AN24	DDR3_64_DQ26
IO_L21N_T3L_N5_AD8N_45	AF18	DDR3_64_CE0_N
IO_L21N_T3L_N5_AD8N_46	AP31	DDR3_64_DQ59
IO_L21N_T3L_N5_AD8N_47	Y28	FMC1_LA31_N
IO_L21N_T3L_N5_AD8N_48	W34	FMC1_LA03_N

IO_L21N_T3L_N5_AD8N_64	AL9	IO_RX_P19
IO_L21N_T3L_N5_AD8N_66	B11	FPGA_ADC_SYSREF_N
IO_L21N_T3L_N5_AD8N_67	F24	DDR3_32_DQ26
IO_L21N_T3L_N5_AD8N_68	K15	DDR3_32_A8
IO_L21N_T3L_N5_AD8N_D07_65	R22	QSPI1_IO3
IO_L21P_T3L_N4_AD8P_44	AM24	DDR3_64_DQ24
IO_L21P_T3L_N4_AD8P_45	AE18	DDR3_64_A12
IO_L21P_T3L_N4_AD8P_46	AN31	DDR3_64_DQ61
IO_L21P_T3L_N4_AD8P_47	W28	FMC1_LA31_P
IO_L21P_T3L_N4_AD8P_48	V33	FMC1_LA03_P
IO_L21P_T3L_N4_AD8P_64	AK10	IO_RX_P18
IO_L21P_T3L_N4_AD8P_66	C11	FPGA_ADC_SYSREF_P
IO_L21P_T3L_N4_AD8P_67	F23	DDR3_32_DQ24
IO_L21P_T3L_N4_AD8P_68	L15	DDR3_32_A14
IO_L21P_T3L_N4_AD8P_D06_65	R21	QSPI1_IO2
IO_L22N_T3U_N7_DBC_AD0N_44	AP21	DDR3_64_DQS3_N
IO_L22N_T3U_N7_DBC_AD0N_45	AE15	DDR3_64_CK_N
IO_L22N_T3U_N7_DBC_AD0N_46	AP34	DDR3_64_DQS7_N
IO_L22N_T3U_N7_DBC_AD0N_47	U27	FMC1_LA30_N
IO_L22N_T3U_N7_DBC_AD0N_48	Y32	FMC1_LA15_N
IO_L22N_T3U_N7_DBC_AD0N_64	AP8	IO_RX_P17
IO_L22N_T3U_N7_DBC_AD0N_66	E13	NC
IO_L22N_T3U_N7_DBC_AD0N_67	F20	DDR3_32_DQS3_N
IO_L22N_T3U_N7_DBC_AD0N_68	J18	DDR3_32_CK_N
IO_L22N_T3U_N7_DBC_AD0N_D05_65	L20	QSPI1_IO1
IO_L22P_T3U_N6_DBC_AD0P_44	AP20	DDR3_64_DQS3_P
IO_L22P_T3U_N6_DBC_AD0P_45	AE16	DDR3_64_CK_P
IO_L22P_T3U_N6_DBC_AD0P_46	AN34	DDR3_64_DQS7_P
IO_L22P_T3U_N6_DBC_AD0P_47	U26	FMC1_LA30_P
IO_L22P_T3U_N6_DBC_AD0P_48	Y31	FMC1_LA15_P
IO_L22P_T3U_N6_DBC_AD0P_64	AN8	IO_TX_P20
IO_L22P_T3U_N6_DBC_AD0P_66	F13	NC
IO_L22P_T3U_N6_DBC_AD0P_67	G20	DDR3_32_DQS3_P
IO_L22P_T3U_N6_DBC_AD0P_68	J19	DDR3_32_CK_P
IO_L22P_T3U_N6_DBC_AD0P_D04_65	M20	QSPI1_IO0
IO_L23N_T3U_N9_44	AP25	DDR3_64_DQ28
IO_L23N_T3U_N9_45	AF17	DDR3_64_BA0
IO_L23N_T3U_N9_46	AN32	DDR3_64_DQ63
IO_L23N_T3U_N9_47	W29	FMC1_LA29_N
IO_L23N_T3U_N9_48	V34	FMC1_LA14_N
IO_L23N_T3U_N9_64	AJ8	IO_TX_P19
IO_L23N_T3U_N9_66	A12	RTM_FPGA_LVDS1_N
IO_L23N_T3U_N9_67	F22	DDR3_32_DQ28
IO_L23N_T3U_N9_68	J16	DDR3_32_A2
IO_L23N_T3U_N9_I2C_SDA_65	M21	FPGA_I2C_SDA
IO_L23P_T3U_N8_44	AP24	DDR3_64_DQ30
IO_L23P_T3U_N8_45	AE17	DDR3_64_A0
IO_L23P_T3U_N8_46	AM32	DDR3_64_DQ60
IO_L23P_T3U_N8_47	V29	FMC1_LA29_P
IO_L23P_T3U_N8_48	U34	FMC1_LA14_P
IO_L23P_T3U_N8_64	AJ9	IO_TX_P18
IO_L23P_T3U_N8_66	A13	RTM_FPGA_LVDS1_P
IO_L23P_T3U_N8_67	G22	DDR3_32_DQ30

IO_L23P_T3U_N8_68	K16	DDR3_32_A6
IO_L23P_T3U_N8_I2C_SCLK_65	N21	FPGA_I2C_SCL
IO_L24N_T3U_N11_44	AP23	DDR3_64_DQ27
IO_L24N_T3U_N11_45	AD15	DDR3_64_A6
IO_L24N_T3U_N11_46	AM34	DDR3_64_DQ58
IO_L24N_T3U_N11_47	W26	FMC1_LA28_N
IO_L24N_T3U_N11_48	W31	FMC1_LA12_N
IO_L24N_T3U_N11_64	AL8	PRI_UART_RxD
IO_L24N_T3U_N11_66	C13	REC_CLOCK_N
IO_L24N_T3U_N11_67	G21	DDR3_32_DQ27
IO_L24N_T3U_N11_68	L18	DDR3_32_A12
IO_L24N_T3U_N11_DOUT_CSO_B_65	K21	FPGA_CFG_DOUT
IO_L24P_T3U_N10_44	AN23	DDR3_64_DQ25
IO_L24P_T3U_N10_45	AD16	DDR3_64_A4
IO_L24P_T3U_N10_46	AL34	DDR3_64_DQ62
IO_L24P_T3U_N10_47	V26	FMC1_LA28_P
IO_L24P_T3U_N10_48	V31	FMC1_LA12_P
IO_L24P_T3U_N10_64	AK8	PRI_UART_TxD
IO_L24P_T3U_N10_66	D13	REC_CLOCK_P
IO_L24P_T3U_N10_67	H21	DDR3_32_DQ25
IO_L24P_T3U_N10_68	L19	DDR3_32_BA0
IO_L24P_T3U_N10_EMCCLK_65	K20	RGMII2_TXD0
IO_L2N_T0L_N3_44	AG20	DDR3_64_DQ1
IO_L2N_T0L_N3_45	AP18	TXC11_N
IO_L2N_T0L_N3_46	AM27	DDR3_64_DQ35
IO_L2N_T0L_N3_47	AD26	TXC13_N
IO_L2N_T0L_N3_48	AF28	FMC1_DP0_C2M_N
IO_L2N_T0L_N3_64	AP13	SFP1_MOD_DEF2
IO_L2N_T0L_N3_66	A9	RTM_FPGA_GTP_Tx0_N
IO_L2N_T0L_N3_67	B27	DDR3_32_DQ1
IO_L2N_T0L_N3_68	A18	NC
IO_L2N_T0L_N3_FWE_FCS2_B_65	G26	NC
IO_L2P_T0L_N2_44	AF20	DDR3_64_DQ3
IO_L2P_T0L_N2_45	AN19	TXC11_P
IO_L2P_T0L_N2_46	AM26	DDR3_64_DQ39
IO_L2P_T0L_N2_47	AD25	TXC13_P
IO_L2P_T0L_N2_48	AE28	FMC1_DP0_C2M_P
IO_L2P_T0L_N2_64	AN13	SFP1_MOD_DEF1
IO_L2P_T0L_N2_66	B9	RTM_FPGA_GTP_Tx0_P
IO_L2P_T0L_N2_67	C27	DDR3_32_DQ3
IO_L2P_T0L_N2_68	A19	NC
IO_L2P_T0L_N2_FOE_B_65	G25	NC
IO_L3N_T0L_N5_AD15N_44	AE20	DDR3_64_DQ7
IO_L3N_T0L_N5_AD15N_45	AN16	RXC11_N
IO_L3N_T0L_N5_AD15N_46	AK27	DDR3_64_DQ38
IO_L3N_T0L_N5_AD15N_47	AC24	RXC13_N
IO_L3N_T0L_N5_AD15N_48	AD28	TXC15_N
IO_L3N_T0L_N5_AD15N_64	AN11	SFP1_MOD_DEF0
IO_L3N_T0L_N5_AD15N_66	C8	RTM_FPGA_GTP_Tx1_N
IO_L3N_T0L_N5_AD15N_67	D29	DDR3_32_DQ7
IO_L3N_T0L_N5_AD15N_68	A15	NC
IO_L3N_T0L_N5_AD15N_A27_65	K27	NC
IO_L3P_T0L_N4_AD15P_44	AD20	DDR3_64_DQ5

IO_L3P_T0L_N4_AD15P_45	AM17	RXC11_P
IO_L3P_T0L_N4_AD15P_46	AK26	DDR3_64_DQ33
IO_L3P_T0L_N4_AD15P_47	AB24	RXC13_P
IO_L3P_T0L_N4_AD15P_48	AC28	TXC15_P
IO_L3P_T0L_N4_AD15P_64	AM11	SFP1_LOS
IO_L3P_T0L_N4_AD15P_66	D8	RTM_FPGA_GTP_Tx1_P
IO_L3P_T0L_N4_AD15P_67	E28	DDR3_32_DQ5
IO_L3P_T0L_N4_AD15P_68	B15	NC
IO_L3P_T0L_N4_AD15P_A26_65	K26	NC
IO_L4N_T0U_N7_DBC_AD7N_44	AH21	DDR3_64_DQS0_N
IO_L4N_T0U_N7_DBC_AD7N_45	AN17	TXC10_N
IO_L4N_T0U_N7_DBC_AD7N_46	AL28	DDR3_64_DQS4_N
IO_L4N_T0U_N7_DBC_AD7N_47	AC27	TXC12_N
IO_L4N_T0U_N7_DBC_AD7N_48	AG29	RXC15_N
IO_L4N_T0U_N7_DBC_AD7N_64	AN12	SFP2_TX_FAULT
IO_L4N_T0U_N7_DBC_AD7N_66	A10	FPGA_DAC_SYSREF_N
IO_L4N_T0U_N7_DBC_AD7N_67	A29	DDR3_32_DQS0_N
IO_L4N_T0U_N7_DBC_AD7N_68	B19	NC
IO_L4N_T0U_N7_DBC_AD7N_A25_65	J25	NC
IO_L4P_T0U_N6_DBC_AD7P_44	AG21	DDR3_64_DQS0_P
IO_L4P_T0U_N6_DBC_AD7P_45	AN18	TXC10_P
IO_L4P_T0U_N6_DBC_AD7P_46	AL27	DDR3_64_DQS4_P
IO_L4P_T0U_N6_DBC_AD7P_47	AC26	TXC12_P
IO_L4P_T0U_N6_DBC_AD7P_48	AF29	RXC15_P
IO_L4P_T0U_N6_DBC_AD7P_64	AM12	SFP2_TX_DISABLE
IO_L4P_T0U_N6_DBC_AD7P_66	B10	FPGA_DAC_SYSREF_P
IO_L4P_T0U_N6_DBC_AD7P_67	B29	DDR3_32_DQS0_P
IO_L4P_T0U_N6_DBC_AD7P_68	C19	NC
IO_L4P_T0U_N6_DBC_AD7P_A24_65	J24	MMC_MOSI1
IO_L5N_T0U_N9_AD14N_44	AE23	DDR3_64_DQ0
IO_L5N_T0U_N9_AD14N_45	AM15	RXC10_N
IO_L5N_T0U_N9_AD14N_46	AH28	DDR3_64_DQ32
IO_L5N_T0U_N9_AD14N_47	AB27	RXC12_N
IO_L5N_T0U_N9_AD14N_48	AE30	TXC14_N
IO_L5N_T0U_N9_AD14N_64	AL12	SFP2_RATE_SELECT
IO_L5N_T0U_N9_AD14N_66	C9	RTM_FPGA_GTP_RxC0_N
IO_L5N_T0U_N9_AD14N_67	C28	DDR3_32_DQ0
IO_L5N_T0U_N9_AD14N_68	B16	NC
IO_L5N_T0U_N9_AD14N_A23_65	H26	MMC_MISO1
IO_L5P_T0U_N8_AD14P_44	AE22	DDR3_64_DQ4
IO_L5P_T0U_N8_AD14P_45	AM16	RXC10_P
IO_L5P_T0U_N8_AD14P_46	AH27	DDR3_64_DQ37
IO_L5P_T0U_N8_AD14P_47	AA27	RXC12_P
IO_L5P_T0U_N8_AD14P_48	AD29	TXC14_P
IO_L5P_T0U_N8_AD14P_64	AK12	SFP2_MOD_DEF2
IO_L5P_T0U_N8_AD14P_66	D9	RTM_FPGA_GTP_RxC0_P
IO_L5P_T0U_N8_AD14P_67	D28	DDR3_32_DQ4
IO_L5P_T0U_N8_AD14P_68	B17	NC
IO_L5P_T0U_N8_AD14P_A22_65	J26	MMC_SSEL1
IO_L6N_T0U_N11_AD6N_44	AG22	DDR3_64_DQ6
IO_L6N_T0U_N11_AD6N_45	AP15	TXC7_N
IO_L6N_T0U_N11_AD6N_46	AK28	DDR3_64_DQ34
IO_L6N_T0U_N11_AD6N_47	AB26	FMC1_LA27_N

IO_L6N_T0U_N11_AD6N_48	AG30	RXC14_N
IO_L6N_T0U_N11_AD6N_64	AL13	SFP2_MOD_DEF1
IO_L6N_T0U_N11_AD6N_66	D10	ADC2_SYNC_N
IO_L6N_T0U_N11_AD6N_67	A28	DDR3_32_DQ6
IO_L6N_T0U_N11_AD6N_68	C17	NC
IO_L6N_T0U_N11_AD6N_A21_65	H24	MMC_SCK1
IO_L6P_T0U_N10_AD6P_44	AF22	DDR3_64_DQ2
IO_L6P_T0U_N10_AD6P_45	AP16	TXC7_P
IO_L6P_T0U_N10_AD6P_46	AJ28	DDR3_64_DQ36
IO_L6P_T0U_N10_AD6P_47	AB25	FMC1_LA27_P
IO_L6P_T0U_N10_AD6P_48	AF30	RXC14_P
IO_L6P_T0U_N10_AD6P_64	AK13	SFP2_MOD_DEF0
IO_L6P_T0U_N10_AD6P_66	E10	ADC2_SYNC_P
IO_L6P_T0U_N10_AD6P_67	A27	DDR3_32_DQ2
IO_L6P_T0U_N10_AD6P_68	C18	NC
IO_L6P_T0U_N10_AD6P_A20_65	J23	FPGA_STATUS
IO_L7N_T1L_N1_QBC_AD13N_44	AE26	NC
IO_L7N_T1L_N1_QBC_AD13N_45	AM14	RXC7_N
IO_L7N_T1L_N1_QBC_AD13N_46	AP26	NC
IO_L7N_T1L_N1_QBC_AD13N_47	AB22	FMC1_LA26_N
IO_L7N_T1L_N1_QBC_AD13N_48	AG32	FMC1_LA04_N
IO_L7N_T1L_N1_QBC_AD13N_64	AF13	SFP2_LOS
IO_L7N_T1L_N1_QBC_AD13N_66	K8	SYNCOUT10_N
IO_L7N_T1L_N1_QBC_AD13N_67	D26	NC
IO_L7N_T1L_N1_QBC_AD13N_68	C14	NC
IO_L7N_T1L_N1_QBC_AD13N_A19_65	L27	AUX_UART_RxD
IO_L7P_T1L_N0_QBC_AD13P_44	AE25	DDR3_64_DM1
IO_L7P_T1L_N0_QBC_AD13P_45	AL14	RXC7_P
IO_L7P_T1L_N0_QBC_AD13P_46	AN26	DDR3_64_DM5
IO_L7P_T1L_N0_QBC_AD13P_47	AA22	FMC1_LA26_P
IO_L7P_T1L_N0_QBC_AD13P_48	AG31	FMC1_LA04_P
IO_L7P_T1L_N0_QBC_AD13P_64	AE13	SFP2_LED2
IO_L7P_T1L_N0_QBC_AD13P_66	L8	SYNCOUT10_P
IO_L7P_T1L_N0_QBC_AD13P_67	E26	DDR3_32_DM1
IO_L7P_T1L_N0_QBC_AD13P_68	D14	DDR3_32_A9
IO_L7P_T1L_N0_QBC_AD13P_A18_65	M27	AUX_UART_TxD
IO_L8N_T1L_N3_AD5N_44	AF24	DDR3_64_DQ13
IO_L8N_T1L_N3_AD5N_45	AM19	NC
IO_L8N_T1L_N3_AD5N_46	AP29	DDR3_64_DQ41
IO_L8N_T1L_N3_AD5N_47	AC23	FMC1_LA24_N
IO_L8N_T1L_N3_AD5N_48	AG34	FMC1_LA02_N
IO_L8N_T1L_N3_AD5N_64	AJ13	SFP2_LED1
IO_L8N_T1L_N3_AD5N_66	H9	SYNCOUT20_N
IO_L8N_T1L_N3_AD5N_67	A25	DDR3_32_DQ13
IO_L8N_T1L_N3_AD5N_68	D15	DDR3_32_A1
IO_L8N_T1L_N3_AD5N_A17_65	L24	SI5324_RST
IO_L8P_T1L_N2_AD5P_44	AF23	DDR3_64_DQ11
IO_L8P_T1L_N2_AD5P_45	AL19	DDR3_64_CKE
IO_L8P_T1L_N2_AD5P_46	AP28	DDR3_64_DQ45
IO_L8P_T1L_N2_AD5P_47	AC22	FMC1_LA24_P
IO_L8P_T1L_N2_AD5P_48	AF33	FMC1_LA02_P
IO_L8P_T1L_N2_AD5P_64	AH13	DIO9
IO_L8P_T1L_N2_AD5P_66	J9	SYNCOUT20_P

IO_L8P_T1L_N2_AD5P_67	B25	DDR3_32_DQ11
IO_L8P_T1L_N2_AD5P_68	E15	DDR3_32_A0
IO_L8P_T1L_N2_AD5P_A16_65	L23	SMA_IO2_DIR
IO_L9N_T1L_N5_AD12N_44	AG25	DDR3_64_DQ15
IO_L9N_T1L_N5_AD12N_45	AL15	DDR3_64_A11
IO_L9N_T1L_N5_AD12N_46	AN28	DDR3_64_DQ43
IO_L9N_T1L_N5_AD12N_47	AB20	FMC1_LA25_N
IO_L9N_T1L_N5_AD12N_48	AF32	FMC1_LA09_N
IO_L9N_T1L_N5_AD12N_64	AF12	DIO8
IO_L9N_T1L_N5_AD12N_66	H8	RTM_FPGA_USR_IO_N
IO_L9N_T1L_N5_AD12N_67	B26	DDR3_32_DQ15
IO_L9N_T1L_N5_AD12N_68	F14	DDR3_32_RST_N
IO_L9N_T1L_N5_AD12N_A15_D31_65	K25	SMA_IO1_DIR
IO_L9P_T1L_N4_AD12P_44	AG24	DDR3_64_DQ9
IO_L9P_T1L_N4_AD12P_45	AK15	DDR3_64_A7
IO_L9P_T1L_N4_AD12P_46	AN27	DDR3_64_DQ47
IO_L9P_T1L_N4_AD12P_47	AA20	FMC1_LA25_P
IO_L9P_T1L_N4_AD12P_48	AE32	FMC1_LA09_P
IO_L9P_T1L_N4_AD12P_64	AE12	DIO7
IO_L9P_T1L_N4_AD12P_66	J8	RTM_FPGA_USR_IO_P
IO_L9P_T1L_N4_AD12P_67	C26	DDR3_32_DQ9
IO_L9P_T1L_N4_AD12P_68	F15	NC
IO_L9P_T1L_N4_AD12P_A14_D30_65	L25	SMA_IO2
IO_T0U_N12_64	AK11	SFP1_RATE_SELECT
IO_T0U_N12_A28_65	H23	FPGA_RESETn
IO_T0U_N12_VRP_44	AD24	VRP_44
IO_T0U_N12_VRP_45	AP19	VRP_45
IO_T0U_N12_VRP_46	AG26	VRP_46
IO_T0U_N12_VRP_47	AA28	VRP_47
IO_T0U_N12_VRP_48	AC29	VRP_48
IO_T0U_N12_VRP_66	A8	VRP_68
IO_T0U_N12_VRP_67	C29	VRP_67
IO_T0U_N12_VRP_68	A17	VRP_66
IO_T1U_N12_44	AF25	NC
IO_T1U_N12_45	AJ19	NC
IO_T1U_N12_46	AM31	NC
IO_T1U_N12_47	Y22	NC
IO_T1U_N12_48	AE31	NC
IO_T1U_N12_64	AJ11	DIO2
IO_T1U_N12_66	L9	NC
IO_T1U_N12_67	A23	NC
IO_T1U_N12_68	C16	NC
IO_T1U_N12_PERSTN1_65	N23	NC
IO_T2U_N12_44	AK25	HW_ID2
IO_T2U_N12_45	AH14	DDR3_64_A5
IO_T2U_N12_46	AH29	NC
IO_T2U_N12_47	Y21	NC
IO_T2U_N12_48	AA33	NC
IO_T2U_N12_64	AJ10	SFP1_LED2
IO_T2U_N12_66	F12	NC
IO_T2U_N12_67	A22	NC
IO_T2U_N12_68	H14	NC
IO_T2U_N12_CSI_ADV_B_65	N27	NC

IO_T3U_N12_44	AM25	HW_ID0
IO_T3U_N12_45	AD14	NC
IO_T3U_N12_46	AK33	NC
IO_T3U_N12_47	U29	NC
IO_T3U_N12_48	V32	NC
IO_T3U_N12_64	AM9	IO_TX_P17
IO_T3U_N12_66	E12	NC
IO_T3U_N12_67	H22	NC
IO_T3U_N12_68	L17	NC
IO_T3U_N12_PERSTN0_65	K22	RGMII2_TXD1
M0_0	K7	FPGA_M0
M1_0	L7	FPGA_M1
M2_0	M7	FPGA_M2
MGTAVCC_L-1	F30	GND
MGTAVCC_L-2	H29	GND
MGTAVCC_L-3	J31	GND
MGTAVCC_L-4	N31	GND
MGTAVCC_L-5	P29	GND
MGTAVCC_L-6	E31	GND
MGTAVCC_R-1	C6	MGTAVCC
MGTAVCC_R-10	AE6	MGTAVCC
MGTAVCC_R-11	AJ6	MGTAVCC
MGTAVCC_R-12	AL6	MGTAVCC
MGTAVCC_R-13	AN6	MGTAVCC
MGTAVCC_R-14	AG6	MGTAVCC
MGTAVCC_R-2	E6	MGTAVCC
MGTAVCC_R-3	G6	MGTAVCC
MGTAVCC_R-4	J6	MGTAVCC
MGTAVCC_R-5	L6	MGTAVCC
MGTAVCC_R-6	N6	MGTAVCC
MGTAVCC_R-7	U6	MGTAVCC
MGTAVCC_R-8	W6	MGTAVCC
MGTAVCC_R-9	AC6	MGTAVCC
MGTAVTT_L-1	G32	GND
MGTAVTT_L-2	D33	GND
MGTAVTT_L-3	R32	GND
MGTAVTT_L-4	C32	GND
MGTAVTT_L-5	K33	GND
MGTAVTT_L-6	L32	GND
MGTAVTT_L-7	P33	GND
MGTAVTT_L-8	H33	GND
MGTAVTT_R-1	R2	MGTAVTT
MGTAVTT_R-10	AM3	MGTAVTT
MGTAVTT_R-11	AL2	MGTAVTT
MGTAVTT_R-12	AH3	MGTAVTT
MGTAVTT_R-13	AG2	MGTAVTT
MGTAVTT_R-14	AD3	MGTAVTT
MGTAVTT_R-15	AC2	MGTAVTT
MGTAVTT_R-16	C2	MGTAVTT
MGTAVTT_R-2	T3	MGTAVTT
MGTAVTT_R-3	W2	MGTAVTT
MGTAVTT_R-4	Y3	MGTAVTT
MGTAVTT_R-5	M3	MGTAVTT

MGTAVTT_R-6	L2	MGTAVTT
MGTAVTT_R-7	H3	MGTAVTT
MGTAVTT_R-8	G2	MGTAVTT
MGTAVTT_R-9	D3	MGTAVTT
MGTAVTTRCAL_R	AP6	MGTAVTT
MGTHRXN0_224	AP1	SFP1RX_N
MGTHRXN0_225	AH1	GTP15RX_N
MGTHRXN0_226	Y1	GTP11RX_N
MGTHRXN0_227	M1	GTP7RX_N
MGTHRXN0_228	E3	GTP3RX_N
MGTHRXN1_224	AM1	SFP2RX_N
MGTHRXN1_225	AF1	GTP14RX_N
MGTHRXN1_226	V1	GTP10RX_N
MGTHRXN1_227	K1	GTP6RX_N
MGTHRXN1_228	D1	GTP2RX_N
MGTHRXN2_224	AK1	RX4_N
MGTHRXN2_225	AD1	GTP13RX_N
MGTHRXN2_226	T1	GTP9RX_N
MGTHRXN2_227	H1	GTP5RX_N
MGTHRXN2_228	B1	GTP1RX_N
MGTHRXN3_224	AJ3	RX5_N
MGTHRXN3_225	AB1	GTP12RX_N
MGTHRXN3_226	P1	GTP8RX_N
MGTHRXN3_227	F1	GTP4RX_N
MGTHRXN3_228	A3	GTP0RX_N
MGTHRXP0_224	AP2	SFP1RX_P
MGTHRXP0_225	AH2	GTP15RX_P
MGTHRXP0_226	Y2	GTP11RX_P
MGTHRXP0_227	M2	GTP7RX_P
MGTHRXP0_228	E4	GTP3RX_P
MGTHRXP1_224	AM2	SFP2RX_P
MGTHRXP1_225	AF2	GTP14RX_P
MGTHRXP1_226	V2	GTP10RX_P
MGTHRXP1_227	K2	GTP6RX_P
MGTHRXP1_228	D2	GTP2RX_P
MGTHRXP2_224	AK2	RX4_P
MGTHRXP2_225	AD2	GTP13RX_P
MGTHRXP2_226	T2	GTP9RX_P
MGTHRXP2_227	H2	GTP5RX_P
MGTHRXP2_228	B2	GTP1RX_P
MGTHRXP3_224	AJ4	RX5_P
MGTHRXP3_225	AB2	GTP12RX_P
MGTHRXP3_226	P2	GTP8RX_P
MGTHRXP3_227	F2	GTP4RX_P
MGTHRXP3_228	A4	GTP0RX_P
MGTHTXN0_224	AN3	SFP1TX_N
MGTHTXN0_225	AH5	GTP15TXC_N
MGTHTXN0_226	AA3	GTP11TXC_N
MGTHTXN0_227	N3	GTP7TXC_N
MGTHTXN0_228	F5	GTP3TXC_N
MGTHTXN1_224	AM5	SFP2TX_N
MGTHTXN1_225	AG3	GTP14TXC_N
MGTHTXN1_226	W3	GTP10TXC_N

MGTHTXN1_227	L3	GTP6TXC_N
MGTHTXN1_228	D5	GTP2TXC_N
MGTHTXN2_224	AL3	TX4C_N
MGTHTXN2_225	AE3	GTP13TXC_N
MGTHTXN2_226	U3	GTP9TXC_N
MGTHTXN2_227	J3	GTP5TXC_N
MGTHTXN2_228	C3	GTP1TXC_N
MGTHTXN3_224	AK5	TX5C_N
MGTHTXN3_225	AC3	GTP12TXC_N
MGTHTXN3_226	R3	GTP8TXC_N
MGTHTXN3_227	G3	GTP4TXC_N
MGTHTXN3_228	B5	GTP0TXC_N
MGTHTXP0_224	AN4	SFP1TX_P
MGTHTXP0_225	AH6	GTP15TXC_P
MGTHTXP0_226	AA4	GTP11TXC_P
MGTHTXP0_227	N4	GTP7TXC_P
MGTHTXP0_228	F6	GTP3TXC_P
MGTHTXP1_224	AM6	SFP2TX_P
MGTHTXP1_225	AG4	GTP14TXC_P
MGTHTXP1_226	W4	GTP10TXC_P
MGTHTXP1_227	L4	GTP6TXC_P
MGTHTXP1_228	D6	GTP2TXC_P
MGTHTXP2_224	AL4	TX4C_P
MGTHTXP2_225	AE4	GTP13TXC_P
MGTHTXP2_226	U4	GTP9TXC_P
MGTHTXP2_227	J4	GTP5TXC_P
MGTHTXP2_228	C4	GTP1TXC_P
MGTHTXP3_224	AK6	TX5C_P
MGTHTXP3_225	AC4	GTP12TXC_P
MGTHTXP3_226	R4	GTP8TXC_P
MGTHTXP3_227	G4	GTP4TXC_P
MGTHTXP3_228	B6	GTP0TXC_P
MGTREFCLK0N_224	AF5	CDR_CLK_CLEAN2_N
MGTREFCLK0N_225	AB5	REFCLK227_C_N
MGTREFCLK0N_226	V5	GTP_CLK1_IN_N
MGTREFCLK0N_227	P5	GTP_CLK2_IN_N
MGTREFCLK0N_228	K5	REFCLK224_C_N
MGTREFCLK0P_224	AF6	CDR_CLK_CLEAN2_P
MGTREFCLK0P_225	AB6	REFCLK227_C_P
MGTREFCLK0P_226	V6	GTP_CLK1_IN_P
MGTREFCLK0P_227	P6	GTP_CLK2_IN_P
MGTREFCLK0P_228	K6	REFCLK224_C_P
MGTREFCLK1N_224	AD5	FCLKAC_N
MGTREFCLK1N_225	Y5	NC
MGTREFCLK1N_226	T5	NC
MGTREFCLK1N_227	M5	CLK_RFU_N
MGTREFCLK1N_228	H5	NC
MGTREFCLK1P_224	AD6	FCLKAC_P
MGTREFCLK1P_225	Y6	NC
MGTREFCLK1P_226	T6	NC
MGTREFCLK1P_227	M6	CLK_RFU_P
MGTREFCLK1P_228	H6	NC
MGTRREF_R	AP5	MGTREF

MGTVCCAUX_L-1	K29	GND
MGTVCCAUX_L-2	M29	GND
MGTVCCAUX_R-1	AA6	MGTVCCAUX
MGTVCCAUX_R-2	R6	MGTVCCAUX
NC-1	K31	NC
NC-10	M32	NC
NC-11	A34	NC
NC-12	A33	NC
NC-13	P31	NC
NC-14	N33	NC
NC-15	N34	NC
NC-16	P32	NC
NC-17	R29	NC
NC-18	R30	NC
NC-19	T31	NC
NC-2	J33	NC
NC-20	R33	NC
NC-21	R34	NC
NC-22	T32	NC
NC-23	B31	NC
NC-24	C33	NC
NC-25	C34	NC
NC-26	B32	NC
NC-27	J29	NC
NC-28	J30	NC
NC-29	D31	NC
NC-3	J34	NC
NC-30	E33	NC
NC-31	E34	NC
NC-32	D32	NC
NC-33	G29	NC
NC-34	F31	NC
NC-35	F32	NC
NC-36	G30	NC
NC-37	L29	NC
NC-38	L30	NC
NC-39	H31	NC
NC-4	K32	NC
NC-40	G33	NC
NC-41	G34	NC
NC-42	H32	NC
NC-5	N29	NC
NC-6	N30	NC
NC-7	M31	NC
NC-8	L33	NC
NC-9	L34	NC
POR_OVERRIDE	P7	POR_override
PROGRAM_B_0	T7	FPGA_PROG_B
PUDC_B_0	R7	PUDC
RDWR_FCS_B_0	U7	QSPI0_CS_B
TCK_0	AC9	FPGA_TCK
TDI_0	V9	FPGA_TDI
TDO_0	U9	FPGA_TDO

TMS_0	W9	FPGA_TMS
VBATT	AD7	FPGA_VBATT
VCCADC	U12	ADC_VCC
VCCAUX_IO-1	M19	VCCAUX
VCCAUX_IO-10	Y19	VCCAUX
VCCAUX_IO-11	V19	VCCAUX
VCCAUX_IO-2	R20	VCCAUX
VCCAUX_IO-3	P19	VCCAUX
VCCAUX_IO-4	N18	VCCAUX
VCCAUX_IO-5	U20	VCCAUX
VCCAUX_IO-6	T19	VCCAUX
VCCAUX_IO-7	W20	VCCAUX
VCCAUX_IO-8	AB19	VCCAUX
VCCAUX_IO-9	AC18	VCCAUX
VCCAUX-1	AA8	VCCAUX
VCCAUX-2	AC8	VCCAUX
VCCAUX-3	U8	VCCAUX
VCCAUX-4	W8	VCCAUX
VCCBRAM-1	Y17	VCCBRAM
VCCBRAM-2	AB17	VCCBRAM
VCCBRAM-3	V17	VCCBRAM
VCCBRAM-4	AA18	VCCBRAM
VCCINT_IO-1	R18	VCCINT
VCCINT_IO-2	M17	VCCINT
VCCINT_IO-3	P17	VCCINT
VCCINT_IO-4	U18	VCCINT
VCCINT_IO-5	W18	VCCINT
VCCINT-1	T15	VCCINT
VCCINT-10	N14	VCCINT
VCCINT-11	R14	VCCINT
VCCINT-12	N10	VCCINT
VCCINT-13	R10	VCCINT
VCCINT-14	AC16	VCCINT
VCCINT-15	M13	VCCINT
VCCINT-16	R8	VCCINT
VCCINT-17	M9	VCCINT
VCCINT-18	M11	VCCINT
VCCINT-19	M15	VCCINT
VCCINT-2	U14	VCCINT
VCCINT-20	N12	VCCINT
VCCINT-21	P9	VCCINT
VCCINT-22	P13	VCCINT
VCCINT-23	P15	VCCINT
VCCINT-24	R12	VCCINT
VCCINT-25	T9	VCCINT
VCCINT-26	T13	VCCINT
VCCINT-27	U10	VCCINT
VCCINT-28	V13	VCCINT
VCCINT-29	W10	VCCINT
VCCINT-3	V15	VCCINT
VCCINT-30	W14	VCCINT
VCCINT-31	Y15	VCCINT
VCCINT-32	AA12	VCCINT

VCCINT-33	AB11	VCCINT
VCCINT-34	AB13	VCCINT
VCCINT-35	AC10	VCCINT
VCCINT-36	AC14	VCCINT
VCCINT-37	T17	VCCINT
VCCINT-38	N8	VCCINT
VCCINT-39	N16	VCCINT
VCCINT-4	AB15	VCCINT
VCCINT-40	U16	VCCINT
VCCINT-41	W16	VCCINT
VCCINT-42	AA16	VCCINT
VCCINT-43	R16	VCCINT
VCCINT-44	AA14	VCCINT
VCCINT-5	AC12	VCCINT
VCCINT-6	T11	VCCINT
VCCINT-7	AA10	VCCINT
VCCINT-8	Y13	VCCINT
VCCINT-9	P11	VCCINT
VCCO_0-1	Y9	P3V3
VCCO_0-2	AB9	P3V3
VCCO_44-1	AM23	P1V5
VCCO_44-2	AF21	P1V5
VCCO_44-3	AH25	P1V5
VCCO_44-4	AJ22	P1V5
VCCO_44-5	AN20	P1V5
VCCO_44-6	AE24	P1V5
VCCO_45-1	AG18	P1V5
VCCO_45-2	AH15	P1V5
VCCO_45-3	AE14	P1V5
VCCO_45-4	AD17	P1V5
VCCO_45-5	AL16	P1V5
VCCO_45-6	AP17	P1V5
VCCO_45-7	AK19	P1V5
VCCO_46-1	AL26	P1V5
VCCO_46-2	AJ32	P1V5
VCCO_46-3	AK29	P1V5
VCCO_46-4	AM33	P1V5
VCCO_46-5	AP27	P1V5
VCCO_46-6	AN30	P1V5
VCCO_46-7	AG28	P1V5
VCCO_47-1	U28	P1V8
VCCO_47-2	V25	P1V8
VCCO_47-3	AA26	P1V8
VCCO_47-4	T21	P1V8
VCCO_47-5	AB23	P1V8
VCCO_47-6	AC20	P1V8
VCCO_47-7	W22	P1V8
VCCO_48-1	AC30	P1V8
VCCO_48-2	AE34	P1V8
VCCO_48-3	W32	P1V8
VCCO_48-4	Y29	P1V8
VCCO_48-5	AB33	P1V8
VCCO_48-6	AF31	P1V8

VCCO_48-7	AD27	P1V8
VCCO_64-1	AN10	P3V3
VCCO_64-2	AM13	P3V3
VCCO_64-3	AJ12	P3V3
VCCO_64-4	AF11	P3V3
VCCO_64-5	AG8	P3V3
VCCO_64-6	AK9	P3V3
VCCO_65-1	P27	P3V3
VCCO_65-2	J22	P3V3
VCCO_65-3	L26	P3V3
VCCO_65-4	M23	P3V3
VCCO_65-5	N20	P3V3
VCCO_65-6	R24	P3V3
VCCO_65-7	H25	P3V3
VCCO_66-1	C10	P1V8
VCCO_66-2	B13	P1V8
VCCO_66-3	F11	P1V8
VCCO_66-4	J12	P1V8
VCCO_66-5	K9	P1V8
VCCO_66-6	G8	P1V8
VCCO_67-1	B23	P1V5
VCCO_67-2	A26	P1V5
VCCO_67-3	F21	P1V5
VCCO_67-4	D27	P1V5
VCCO_67-5	E24	P1V5
VCCO_67-6	C20	P1V5
VCCO_68-1	K19	P1V5
VCCO_68-2	H15	P1V5
VCCO_68-3	G18	P1V5
VCCO_68-4	E14	P1V5
VCCO_68-5	D17	P1V5
VCCO_68-6	A16	P1V5
VCCO_68-7	L16	P1V5
VN	W11	SYSMON_VN_R
VP	V12	SYSMON_VP_R
VREF_44	AD23	DDR3_64_VREF
VREF_45	AF19	DDR3_64_VREF
VREF_46	AG27	DDR3_64_VREF
VREF_47	V24	FMC1_VREF_A_M2C
VREF_48	AA30	FMC1_VREF_A_M2C
VREF_64	AD13	Vref64
VREF_65	J21	Vref65
VREF_66	L10	DDR32_VREF
VREF_67	J20	DDR32_VREF
VREF_68	L14	Vref68
VREFN	V11	GND
VREFP	W12	Vrefp

B Gateware configurations

mathrm problem

Spline parametrization *This is inherited from the pdq2 documentation*

The method of compression is a polynomial basis spline (B-spline). The data consists of a sequence of knots. Each knot is described by a duration $\Delta t$ and spline coefficients $u_{n}$ up to order $k$. If the knot is evaluated starting at time $t_{0}$, the output $u(t)$ for $t\in[t_{0},t_{0}+\Delta t]$ is $$u(t)=\sum_{n=0}^{k}\frac{u_{n}}{n!}(t-t_{0})^{n}=u_{0}+u_{1}(t-t_{0})+\frac{u_{2}}{2} A sequence of such knots describes a spline waveform. From one discrete time $i$ to the next $i+1$ each accumulator $v_{n,i}$ is incremented by the value of the next higher order accumulator: $$v_{n,i+1}=v_{n,i}+v_{n+1,i}$$ For a cubic spline the mapping between accumulators' initial values $v_{n,0}$ and the polynomial derivatives or spline coefficients $u_{n}$ can be done off-line and must take into consideration the finite time step size $\tau$. The data for each knot is described by the integer duration $T=\Delta t/\tau$ and the initial values $v_{n,0}$. This representation allows both transient large-bandwidth waveforms and slow but smooth large duty cycle waveforms to be described very efficiently.

Waveform parametrization The gateware will support at least 8 independent channels.

Each channel emits waveforms of the general parametrization: $$z=\left(a_1e^{i(f_1 t+p_1)} + a_2e^{i(f_2 t+p_2)}\right)e^{i(f_0t+p_0)}$$ $$o=u+b_0\mathrm{Re}(z)+b_1\mathrm{Im}(z^\prime)$$

- $o$ is the (real valued) output of a channel
- $z$ is the complex-valued output of the "generator" associated with each channel
- $z^\prime$ is the complex-valued output from the generator of each channel's "buddy" channel. Two adjaccent channels form a buddy pair. This enables seamless usage of the complex data path features in DACs, complex (IQ) analog modulation, and yields "four-tone" support on IQ channels for free.
- $u$ and $a$ are 16-bit cubic (third order) spline interpolators
- $p$ are 16-bit constant (zeroth order) spline interpolators
- $f$ are 48-bit linear (first order) interpolators
- $b$ are switches ($1$ or $0$)

Datapath details

- $f_\mathrm{DATA}\geq1\,\mathrm{GHz}$. Exact clock speed is TBD and depends on simultaneously meeting hardware constraints and an integer relationship with the RTIO clock and physics/noise requirements.
- Oscillator $f_0,p_0$ is sampled at $f_{DATA}$

- Interpolators are updated and interpolate at `$f_\mathrm{DATA}/k$` with `$k$` typically 4 or 8 and `$f_\mathrm{DATA}/k \geq 125\,\mathrm{MHz}$`
- Oscillators `$f_{1,2},p_{1,2}$` are sampled at `$f_{DATA}/k$`
- All amplitude summing junctions shall implement saturating summation to prevent wrap-around.
- All amplitude summing junctions shall implement configurable and guaranteed gateware low-high limiters.
- All amplitude summing junctions shall register saturation events.
- To up-sample the data from the `$f_1$`,`$f_2$` oscillators by `$k$` before passing it into the `$f_0$` oscillator, a CIC filter of order TBD shall be implemented for anti-aliasing. CIC filters are linear phase.
- To implement further anti-aliasing, a symmetric (thus linear phase) FIR filter with TBD taps (FPGA DSP resource limits) shall be implemented after the CIC filter.
- All spline interpolators and the total channel output shall be monitored by the ARTIQ channel monitoring infrastructure.
- All spline interpolators shall support ARTIQ injection/override.

Clocking and synchronization

- Timestamps for spline knot scheduling are at least 62 bit wide.
- Spline knots have 16-bit dynamic range in time.
- In order to support slower sweeps with sparser spline knots, the dynamic range of the spline coefficients can be extended using time stretcher. It decelerates the spline evolution/interpolation rate by a factor of `$2^E$`.
- Waveform output shall be with deterministic latency with respect to the RTIO clock:
  - across channels on the same card (to within DAC chip specification)
  - across cards in the same rack (to within DAC chip and intra-rack DRTIO clock sycnchronization)
  - across racks controlled by the same core device (to within DAC chip and DRTIO clock synchronization)
- Each card can be clocked by an internal DAC clock derived from the RTIO clock or by an external DAC clock.
- When an external DAC clock is used, the waveform synchronization is ensured to within one DAC clock cycle (or the limit of the DAC chip whichever is higher) but below that depends on the phase of the external DAC clock.

- All spline knot interpolators can be updated independently (and also simultaneously) of each other.
- All spline interpolator latencies from the internal “RTIO clock reference plane” to the DAC output are matched and deterministic. Channel and board latencies are matched and deterministic (see above).
- Minimum spline knot duration is k/f_DATA.

Phase update modes The phase accumulator of the DDS cores can be updated in multiple different modes during a phase and/or frequency update.

- relative phase update: $q^\prime(t) = q(t^\prime) + (p^\prime - p) + (t - t^\prime) f^\prime$
- absolute phase update: $q^\prime(t) = p^\prime + (t - t^\prime) f^\prime$
- phase coherent update: $q^\prime(t) = p^\prime + (t - T) f^\prime$, where
- q/$q^\prime$: old/new phase accumulator
- p/$p^\prime$: old/new phase offset
- $f^\prime$: new frequency
- $t^\prime$: timestamp of setting new p,f
- T: “origin” timestamp: beginning of experiment, boot of device, or arbitrary
- t: running time

Relative phase updates are called “continuous phase mode” and coherent updates are called “tracking phase mode” by some. Phase coherent updates can be mapped (in software/runtime) to absolute phase updates by transforming $p^\prime \longrightarrow p^\prime + (t^\prime - T) f^\prime$. Since phase coherent updates require large multiplications is is questionable whether they can and should be implemented in gateware.

It is questionable whether phase coherent updates should or even can be supported for sweeping p/f. They can be supported for the modulation inputs (see below).

Modulation by RTIO To each spline interpolator (any of the nine f,p,a,u in the waveform parametrization) a modulation (summarized as e_RTIO) by a separate RTIO channel can be applied.

- The modulation is an additive offset for frequency and phase (f,p) and a multiplicative offset for amplitudes (u,a).
- The modulation is times like any other (non-interpolating) RTIO event, i.e. ≤ 8ns time resolution and has the same value resolution as the spline interpolator it modulates.
- Default values are 0 for frequency and phase modulation (f,p) and 1 for amplitude modulation (u,a).
- Modulation is normalized to full scale.

Modulation by local DSP In addition to RTIO modulation `$e_\mathrm{RTIO}$` there is "local DSP" modulation input to each spline interpolator.

- Same specifications and semantics as the RTIO modulation.

Local DSP A fully reconfigurable local DSP fabric with multiple IIR filters shall be included. The DSP switchyard supports servoing applications of various types.

- See redpid for a rough feature set.

Runtime and kernel interface

- Spline knot sequences can be generated off-line and embedded in ARTIQ experiments.
- Spline knot sequences can be generated at compile time.
- Spline knot sequences can be embedded into ARTIQ experiments and emitted to from the core device to the DRTIO channels during the experiments.
- Spline knot sequences can be computed dynamically on core device.
- Instead of emitting them directly to the DRTIO channel, spline knot sequences can be emitted into a named DMA context which stores the RTIO events in memory (either on the core device or right at the DRTIO channel in the card's DRAM) for later recall.
- Stored, named DMA segments can be replayed by name.
- Given enough slack to transmit DRTIO events and fill the channel FIFOs (from core device or from any DMA source), all boards, all channels, all splines can burst `$\geq128$` knots each at `$\geq125$`MHz (BRAM FIFO limited). This is independent of whether the events are computed dynamically, off-line, embedded, reside in core device DRAM or remote DRAM.
- When sourcing waveforms from core device memory, the sustained aggregated spline knot rate across all interpolators is `$\geq2$`MHz.
- Sourcing from remote DRTIO DMA the spline knot rate per board (aggregated over all channels and all interpolators on that board) is TBD MHz sustained for TBD knots (DRAM limited).
- Supports setting `$e_\mathrm{DRTIO}$` using standard DRTIO events.
- Supports configuring the DAC through RTIO-SPI
- Utility functions shall be made available to users for processing spline waveforms (scaling in value and time, resampling).
- Given a periodically sampled waveform (vector of values) routines shall
  - generate a spline waveform with a fixed knot duration
  - generate a spline waveform with specified knot count and variable knot duration

  - generate a spline waveform with minimal knot count and specified RMS error

- given user-supplied spline waveform routines shall
  - generate a periodically sampled waveform (vector of values) with user specified resolution
  - determine validity (in-range)

Test Cases ARTIQ Python programs demonstrating the following will be provided.

1. Simultaneous generation of two-tone waveforms on 8 DAC channels where `$f_{1}=f_{0}+\Delta$` and `$f_{2}=f_{0}-\Delta$` where `$f_{0}=200$` MHz and `$\Delta=[0,50]$` MHz.
2. Playing a spline knot sequence demonstrating each spline interpolator in turn.
3. Replaying a 128 knot two-tone amplitude sequence from remote DMA.
4. Phase/frequency/amplitude shifting that sequence using `$e_\mathrm{DRTIO}$`.
5. Demonstrate relative and absolute phase mode.
6. Demonstrate deterministic channel alignment to one DAC clock cycle.
7. Demonstrate external and internal clocking.

B.0.1 Sayma SAWG data rate constraints

The fast smart arbitrary waveform channels require a significant amount of logic resources but also necessitate fulfilling several interacting constraints on operating frequencies and clock ratios.

For the DAC channel data rate `$f_\mathrm{DATA}$` on the JESD204B link, the following rules need to be observed.

- `$t_\mathrm{DATA} = t_\mathrm{RTIO\_FINE}$`. DAC samples need to mesh with RTIO timestamps (e.g. RF switches on TTLs and SYS_REF tagging), otherwise DAC timing is not sample-accurate and samples will beat around RTIO timestamps. The RTIO timestamp granularity is a global design variable of an ARTIQ DRTIO fabric instance. The granularity does not need to be 1ns and can easily be altered globally, but it needs to be the same across the entire DRTIO fabric. If e.g. the core device has a coarse clock of 125MHz and the high resolution TTL provide three more bits of resolution, then the fine timestamp granularity needs to be 1ns (or an integer submultiple) everywhere.

- `$t_\mathrm{SLOWDDS}/k = t_\mathrm{FASTDDS} = t_\mathrm{DATA}$` with `$k$` a power of two. The accumulator phasing and datapath parallelization methods that allow generating multiple samples in a single clock cycle only work for powers of two.

- `$t_\mathrm{SLOWDDS}$` can potentially be as low as 4ns on Kintex 7 with speed grade 2 or better, certainly as low as 5ns. The possibility of 4ns fabric timing would need to be explored and verified.
- `$t_\mathrm{SLOWDDS} = m t_\mathrm{RTIO\_FINE}$`: The spline interpolators, RTIO updates, and the slow DDS should mesh with the fine timestamp (e.g. RF switches on TTLs).
- `$t_\mathrm{SLOWDDS} = p t_\mathrm{RTIO}$`: The spline interpolators, RTIO updates, and the slow DDS should mesh with the coarse timestamp (e.g. relative to RF switches on coarse TTLs). `$p$` is a power of two in the current ARTIQ architecture.
- `$f_\mathrm{DATA} \leq 1.09\,\mathrm{GHz}$` or even 1.03,GHz$ for typical DAC and FPGA transciever line rate.

The DAC sample rate `$f_\mathrm{DAC}$` after interpolation and up-sampling from `$f_\mathrm{DATA}$` needs to satisfy:

- `$f_\mathrm{DATA} \leq 2.4\,\mathrm{GHz}$`: Typical DAC sample rate
- `$f_\mathrm{DAC} = q f_\mathrm{DATA}$` with `$q \in \{1, 2, 4, 8\}$`: Available interpolation options

Logic and RAM

- ARTIQ device CPU(s) and miscellaneous logic resources provide a good estimate for the additional logic required to support DRTIO. The kc705-nist_qc2 design occupies 23k LUT and 5Mb BRAM. The pipistrello-nist_qc1 design uses 15k LUT and 1Mb BRAM (on a slightly different architecture).
- parallelized FIR: 4 channel, 4x parallelism, 30 taps: 240 DSP
- parallelized HBF + tricks: 4 channel, 4x parallelism, 30 taps: 120 DSP
- RTIO FIFOs: 4 channel, 128 knots per RTIO channel: 4Mb
- PID, extrapolating from redpid (xc7z010): 2 channel 125MHz ADC/DAC + misc DSP, full servo crossbar matrix: 13 kLUT, 50 DSP

Several design studies were performed for different configurations of the Sayma SAWG channels:

- Sayma initial SAWG on kc705: 2 channel, 8x parallelism, 125MHz: 28k LUT
- Sayma advanced draft SAWG on kc705: 4 channel, 4x parallelism, 200MHz: 33k LUT
- Sayma advanced draft SAWG on kc705: 4 channel, 8x parallelism, 125MHz: 53k LUT
- Sayma advanced draft SAWG on kc705: 8 channel, 8x parallelism, 125MHz: 106k LUT
- Sayma advanced draft SAWG on kcu105: 4 channel 4x parallelism, 200MHz: 33k LUT

Data and sample rates Somewhere in the Sayma docs, we should have a page about clock distribution, giving users an overview of the different constrains that exist for clocking. This section should be merged into that and/or the SAWG docs. The following choices for data rates and lanes appear to be interesting (BW: bandwidth; SSB: single sideband; DSB: dual sideband; “size”: resource usage in units of 13k LUTs per channel):

					f_1,f_2	BW mix
f_DAC	f_DATA	lane rate	lanes	“size”	DSB BW	2nd+3rd
2.4GHz	600MHz	6GHz	8	4	150MHz	300–600–900MHz
2GHz	1000MHz	10GHz	8	8	125MHz	500–1000–1500MHz
1.6GHz	800MHz	8GHz	8	4	200MHz	400–800–1200MHz
300MHz	300MHz	6GHz	4	2	150MHz	150–300–450MHz

For 4 JESD lanes, use DAC “mix mode” (switching up-conversion by f_DAC) to emphasize second Nyquist zone from $f_\mathrm{DAC}/2$ to f_DAC. Zeros at 0Hz and $2\times f_\mathrm{DAC}$.