RedPitaya

FPGA memory map

Written by	Revision Description	Version	Date
Matej Oblak	Initial	0.1	08/11/13
Matej Oblak	Release1 update	0.2	16/12/13
Matej Oblak	ASG - added burst mode ASG - buffer read pointer readout		Dec. 2014 Jan. 2015
Matej Oblak	AXI master documented		Feb. 2015
Iztok Jeras	Added debounce delay register		Mar. 2015
Iztok Jeras	Added digital loopback, pre trigger status		Apr. 2015
Iztok Jeras	Removed XADC registers GPIO[0] is now R/W		Avg. 2015

Table of Contents

About Document

Red Pitaya HDL design has multiple functions, which are configured by registers. It also uses memory locations to store capture data and generate output signals. All of this are described in this document. Memory location is written in a way that is seen by SW.

FPGA Memory Map

The table describes address space partitioning implemented on FPGA via AXI GP0 interface. All registers have offsets aligned to 4 bytes and are 32 -bit wide. Granularity is 32-bit, meaning that minimum transfer size is 4 bytes. The organization is little-endian .
The memory block is divided into 8 parts. Each part is occupied by individual IP core. Address space of individual application is described in the subsection below. The size of each IP core address space is 4MByte.
For additional informations and better understanding check other documents (schematics, specifications...).

	Start	End	Module Name
CS[0]	0x40000000	0x400FFFFF	Housekeeping
CS[1]	0x40100000	0x401FFFFF	Oscilloscope
CS[2]	0x40200000	0x402FFFFF	Arbitrary signal generator (ASG)
CS[3]	0x40300000	0x403FFFFF	PID controller
CS[4]	0x40400000	0x404FFFFF	Analog mixed signals (AMS)
CS[5]	0x40500000	0x405FFFFF	Daisy chain
CS[6]	0x40600000	0x406FFFFF	FREE
CS[7]	0x40700000	0x407FFFFF	Power test

Red Pitaya Modules

Here are described submodules used in Red Pitaya FPGA logic.

Housekeeping

offset	description	bits	R/W
0x0	**ID**		
	Reserved	*31:4*	*R*
	Design ID 0-prototype0, 1-release1	**3:0**	**R**
0x4	**DNA part1**		
	DNA[31:0]	**31:0**	**R**
0x8	**DNA part2**		
	Reserved	*31:25*	*R*
	DNA[56:32]	**24:0**	**R**
0xC	**Digital Loopback**		
	Reserved	*31:1*	*R*
	digital_loop	**0**	**R/W**
0x10	**Expansion connector direction P**		
	Reserved	*31:8*	*R*
	Direction for P lines 1-out 0-in	**7:0**	**R/W**
0x14	**Expansion connector direction N**		
	Reserved	*31:8*	*R*
	Direction for N lines 1-out 0-in	**7:0**	**R/W**
0x18	**Expansion connector output P**		
	Reserved	*31:8*	*R*
	P pins output	**7:0**	**R/W**
0x1C	**Expansion connector output N**		
	Reserved	*31:8*	*R*
	N pins output	**7:0**	**R/W**
0x20	**Expansion connector input P**		
	Reserved	*31:8*	*R*
	P pins input	**7:0**	**R**
0x24	**Expansion connector input N**		
	Reserved	*31:8*	*R*
	N pins input	**7:0**	**R**
0x30	**LED control**		
	Reserved	*31:8*	*R*

	LEDs 7-0	**7:0**	**R/W**

Oscilloscope

offset	description	bits	R/W
0x0	**Configuration**		
	Reserved	*31:3*	*R*
	Trigger status before acquire ends (0 – pre trigger, 1 – post trigger)	**2**	**R**
	Reset write state machine	**1**	**W**
	Start writing data into memory (ARM trigger).	**0**	**W**
0x4	**Trigger source**		
	Selects trigger source for data capture. When trigger delay is ended value goes to 0.		
	Reserved	*31:4*	*R*
	Trigger source: 1-trig immediately 2-ch A threshold positive edge 3-ch A threshold negative edge 4-ch B threshold positive edge 5-ch B threshold negative edge 6-external trigger positive edge - DIO0_P pin 7-external trigger negative edge 8-arbitrary wave generator application positive edge 9-arbitrary wave generator application negative edge	**3:0**	**R/W**
0x8	**Ch A threshold**		
	Reserved	*31:14*	*R*
	Ch A threshold, makes trigger when ADC value cross this value	**13:0**	**R/W**
0xC	**Ch B threshold**		
	Reserved	*31:14*	*R*
	Ch B threshold, makes trigger when ADC value cross this value	**13:0**	**R/W**
0x10	**Delay after trigger**		
	Number of decimated data after trigger written into memory	**31:0**	**R/W**
0x14	**Data decimation**		
	Decimate input data, uses data average		
	Reserved	*31:17*	*R*
	Data decimation, supports only this values: 1,8, 64,1024,8192,65536. If other value is written data will NOT be correct.	**16:0**	**R/W**
0x18	**Write pointer - current**		
	Reserved	*31:14*	*R*
	Current write pointer	**13:0**	**R**
0x1C	**Write pointer - trigger**		
	Reserved	*31:14*	*R*

	Write pointer at time when trigger arrived	**13:0**	**R**
0x20	**Ch A hysteresis**		
	Reserved	*31:14*	*R*
	Ch A threshold hysteresis. Value must be outside to enable trigger again.	**13:0**	**R/W**
0x24	**Ch B hysteresis**		
	Reserved	*31:14*	*R*
	Ch B threshold hysteresis. Value must be outside to enable trigger again.	**13:0**	**R/W**
0x28	**Other**		
	Reserved	*31:1*	*R*
	Enable signal average at decimation	**0**	**R/W**
0x2C	**PreTrigger Counter**		
	This unsigned counter holds the number of samples captured between the start of acquire and trigger. The value does not overflow, instead it stops incrementing at 0xffffffff.	**31:0**	**R**
0x30	**CH A Equalization filter**		
	Reserved	*31:18*	*R*
	AA Coefficient	**17:0**	**R/W**
0x34	**CH A Equalization filter**		
	Reserved	*31:25*	*R*
	BB Coefficient	**24:0**	**R/W**
0x38	**CH A Equalization filter**		
	Reserved	*31:25*	*R*
	KK Coefficient	**24:0**	**R/W**
0x3C	**CH A Equalization filter**		
	Reserved	*31:25*	*R*
	PP Coefficient	**24:0**	**R/W**
0x40	**CH B Equalization filter**		
	Reserved	*31:18*	*R*
	AA Coefficient	**17:0**	**R/W**
0x44	**CH B Equalization filter**		
	Reserved	*31:25*	*R*
	BB Coefficient	**24:0**	**R/W**
0x48	**CH B Equalization filter**		
	Reserved	*31:25*	*R*
	KK Coefficient	**24:0**	**R/W**
0x4C	**CH B Equalization filter**		

	Reserved	*31:25*	*R*
	PP Coefficient	**24:0**	**R/W**
0x50	**CH A AXI lower address**		
	Starting writing address	**31:0**	**R/W**
0x54	**CH A AXI upper address**		
	Address where it jumps to lower	**31:0**	**R/W**
0x58	**CH A AXI delay after trigger**		
	Number of decimated data after trigger written into memory	**31:0**	**R/W**
0x5C	**CH A AXI enable master**		
	Reserved	*31:1*	*R*
	Enable AXI master	**0**	**R/W**
0x60	**CH A AXI write pointer - trigger**		
	Write pointer at time when trigger arrived	**31:0**	**R**
0x64	**CH A AXI write pointer - current**		
	Current write pointer	**31:0**	**R**
0x70	**CH B AXI lower address**		
	Starting writing address	**31:0**	**R/W**
0x74	**CH B AXI upper address**		
	Address where it jumps to lower	**31:0**	**R/W**
0x78	**CH B AXI delay after trigger**		
	Number of decimated data after trigger written into memory	**31:0**	**R/W**
0x7C	**CH B AXI enable master**		
	Reserved	*31:1*	*R*
	Enable AXI master	**0**	**R/W**
0x80	**CH B AXI write pointer - trigger**		
	Write pointer at time when trigger arrived	**31:0**	**R**
0x84	**CH B AXI write pointer - current**		
	Current write pointer	**31:0**	**R**
0x90	**Trigger debouncer time**		
	Number of ADC clock periods trigger is disabled after activation reset value is decimal 62500 or equivalent to 0.5ms	**19:0**	**R/W**
0xA0	**Accumulator data sequence length**		
	Reserved	*31:14*	*R*
0xA4	**Accumulator data offset corection ChA**		
	Reserved	*31:14*	*R*
	signed offset value	**13:0**	**R/W**
0xA8	**Accumulator data offset corection ChB**		
	Reserved	*31:14*	*R*

	signed offset value	**13:0**	**R/W**
0x10000 to 0x1FFFC	**Memory data (16k samples)**		
	Reserved	*31:16*	*R*
	Captured data for ch A	**15:0**	**R**
0x20000 to 0x2FFFC	**Memory data (16k samples)**		
	Reserved	*31:16*	*R*
	Captured data for ch B	**15:0**	**R**

Arbitrary Signal Generator (ASG)

offset	description	bits	R/W
0x0	**Configuration**		
	Reserved	*31:25*	*R*
	ch B external gated repetitions	**24**	**R/W**
	ch B set output to 0	**23**	**R/W**
	ch B SM reset	**22**	**R/W**
	Reserved	**21**	**R/W**
	ch B SM wrap pointer (if disabled starts at address 0)	**20**	**R/W**
	ch B trigger selector: (don't change when SM is active) 1-trig immediately 2-external trigger positive edge - DIO0_P pin 3-external trigger negative edge	**19:16**	**R/W**
	Reserved	*15:9*	*R*
	ch A external gated bursts	**8**	**R/W**
	ch A set output to 0	**7**	**R/W**
	ch A SM reset	**6**	**R/W**
	Reserved	**5**	**R/W**
	ch A SM wrap pointer (if disabled starts at address 0)	**4**	**R/W**
	ch A trigger selector: (don't change when SM is active) 1-trig immediately 2-external trigger positive edge - DIO0_P pin 3-external trigger negative edge	**3:0**	**R/W**
0x4	**Ch A amplitude scale and offset**		
	out = (data*scale)/0x2000 + offset		
	Reserved	*31:30*	*R*
	Amplitude offset	**29:16**	**R/W**
	Reserved	*15:14*	*R*
	Amplitude scale. 0x2000 == multiply by 1. Unsigned	**13:0**	**R/W**
0x8	**Ch A counter wrap**		
	Reserved	*31:30*	*R*
	Value where counter wraps around. Depends on SM wrap setting. If it is 1 new value is get by wrap, if value is 0 counter goes to offset value. 16 bits for decimals.	**29:0**	**R/W**
0xC	**Ch A start offset**		
	Reserved	*31:30*	*R*
	Counter start offset. Start offset when trigger arrives. 16 bits for decimals.	**29:0**	**R/W**

0x10	**Ch A counter step**		
	Reserved	*31:30*	*R*
	Counter step. 16 bits for decimals.	**29:0**	**R/W**
0x14	**Ch A buffer current read pointer**		
	Reserved	*31:16*	*R*
	Read pointer	**15:2**	**R/W**
	Reserved	*1:0*	*R*
0x18	**Ch A number of read cycles in one burst**		
	Reserved	*31:16*	*R*
	Number of repeats of table readout. 0=infinite	**15:0**	**R/W**
0x1C	**Ch A number of burst repetitions**		
	Reserved	*31:16*	*R*
	Number of repetitions. 0=disabled	**15:0**	**R/W**
0x20	**Ch A delay between burst repetitions**		
	Delay between repetitions. Granularity=1us	**31:0**	**R/W**
0x24	**Ch B amplitude scale and offset**		
	out = (data*scale)/0x2000 + offset		
	Reserved	*31:30*	*R*
	Amplitude offset	**29:16**	**R/W**
	Reserved	*15:14*	*R*
	Amplitude scale. 0x2000 == multiply by 1. Unsigned	**13:0**	**R/W**
0x28	**Ch B counter wrap**		
	Reserved	*31:30*	*R*
	Value where counter wraps around. Depends on SM wrap setting. If it is 1 new value is get by wrap, if value is 0 counter goes to offset value. 16 bits for decimals.	**29:0**	**R/W**
0x2C	**Ch B start offset**		
	Reserved	*31:30*	*R*
	Counter start offset. Start offset when trigger arrives. 16 bits for decimals.	**29:0**	**R/W**
0x30	**Ch B counter step**		
	Reserved	*31:30*	*R*
	Counter step. 16 bits for decimals.	**29:0**	**R/W**
0x34	**Ch B buffer current read pointer**		
	Reserved	*31:16*	*R*
	Read pointer	**15:2**	**R/W**
	Reserved	*1:0*	*R*
0x38	**Ch B number of read cycles in one burst**		

	Reserved	*31:16*	*R*
	Number of repeats of table readout. 0=infinite	**15:0**	**R/W**
0x3C	**Ch B number of burst repetitions**		
	Reserved	*31:16*	*R*
	Number of repetitions. 0=disabled	**15:0**	**R/W**
0x40	**Ch B delay between burst repetitions**		
	Delay between repetitions. Granularity=1us	**31:0**	**R/W**
0x10000 to 0x1FFFC	**Ch A memory data (16k samples)**		
	Reserved	*31:14*	*R*
	ch A data	**13:0**	**R/W**
0x20000 to 0x2FFFC	**Ch B memory data (16k samples)**		
	Reserved	*31:14*	*R*
	ch B data	**13:0**	**R/W**

PID Controller

offset	**description**	**bits**	**R/W**
0x0	**Configuration**		
	Reserved	*31:4*	*R*
	PID22 integrator reset	**3**	**R/W**
	PID21 integrator reset	**2**	**R/W**
	PID12 integrator reset	**1**	**R/W**
	PID11 integrator reset	**0**	**R/W**
0x10	**PID11 set point**		
	Reserved	*31:14*	*R*
	PID11 set point	**13:0**	**R/W**
0x14	**PID11 proportional coefficient**		
	Reserved	*31:14*	*R*
	PID11 Kp	**13:0**	**R/W**
0x18	**PID11 integral coefficient**		
	Reserved	*31:14*	*R*
	PID11 Ki	**13:0**	**R/W**
0x1C	**PID11 derivative coefficient**		
	Reserved	*31:14*	*R*
	PID11 Kd	**13:0**	**R/W**
0x20	**PID12 set point**		
	Reserved	*31:14*	*R*
	PID12 set point	**13:0**	**R/W**
0x24	**PID12 proportional coefficient**		
	Reserved	*31:14*	*R*
	PID12 Kp	**13:0**	**R/W**
0x28	**PID12 integral coefficient**		
	Reserved	*31:14*	*R*
	PID12 Ki	**13:0**	**R/W**
0x2C	**PID12 derivative coefficient**		
	Reserved	*31:14*	*R*
	PID12 Kd	**13:0**	**R/W**
0x30	**PID21 set point**		
	Reserved	*31:14*	*R*
	PID21 set point	**13:0**	**R/W**
0x34	**PID21 proportional coefficient**		

	Reserved	*31:14*	*R*
	PID21 Kp	**13:0**	**R/W**
0x38	**PID21 integral coefficient**		
	Reserved	*31:14*	*R*
	PID21 Ki	**13:0**	**R/W**
0x3C	**PID21 derivative coefficient**		
	Reserved	*31:14*	*R*
	PID21 Kd	**13:0**	**R/W**
0x40	**PID22 set point**		
	Reserved	*31:14*	*R*
	PID22 set point	**13:0**	**R/W**
0x44	**PID22 proportional coefficient**		
	Reserved	*31:14*	*R*
	PID22 Kp	**13:0**	**R/W**
0x48	**PID22 integral coefficient**		
	Reserved	*31:14*	*R*
	PID22 Ki	**13:0**	**R/W**
0x4C	**PID22 derivative coefficient**		
	Reserved	*31:14*	*R*
	PID22 Kd	**13:0**	**R/W**

Analog Mixed Signals (AMS)

offset	**description**	**bits**	**R/W**
0x0	**XADC AIF0**		
	Reserved	*31:12*	*R*
	AIF0 value	**11:0**	**R**
0x4	**XADC AIF1**		
	Reserved	*31:12*	*R*
	AIF1 value	**11:0**	**R**
0x8	**XADC AIF2**		
	Reserved	*31:12*	*R*
	AIF2 value	**11:0**	**R**
0xC	**XADC AIF3**		
	Reserved	*31:12*	*R*
	AIF3 value	**11:0**	**R**
0x10	**XADC AIF4**		
	Reserved	*31:12*	*R*
	AIF4 value (5V power supply)	**11:0**	**R**
0x20	**PWM DAC0**		
	Reserved	*31:24*	*R*
	PWM value (100% == 156)	**23:16**	**R/W**
	Bit select for PWM repetition which have value PWM+1	**15:0**	**R/W**
0x24	**PWM DAC1**		
	Reserved	*31:24*	*R*
	PWM value (100% == 156)	**23:16**	**R/W**
	Bit select for PWM repetition which have value PWM+1	**15:0**	**R/W**
0x28	**PWM DAC2**		
	Reserved	*31:24*	*R*
	PWM value (100% == 156)	**23:16**	**R/W**
	Bit select for PWM repetition which have value PWM+1	**15:0**	**R/W**
0x2C	**PWM DAC3**		
	Reserved	*31:24*	*R*
	PWM value (100% == 156)	**23:16**	**R/W**
	Bit select for PWM repetition which have value PWM+1	**15:0**	**R/W**

Daisy Chain

offset	description	bits	R/W
0x0	**Control**		
	Reserved	*31:2*	*R*
	RX enable	**1**	**R/W**
	TX enable	**0**	**R/W**
0x4	**Transmitter data selector**		
	Custom data	**31:16**	**R/W**
	Reserved	*15:8*	*R*
	Data source 0-data is 0 1-user data (from logic) 2-custom data (from this register) 3-training data (0x00FF) 4-transmit received data (loop back) 5-random data (for testing)	**3:0**	**R/W**
0x8	**Receiver training**		
	Reserved	*31:2*	*R*
	Training successful	**1**	**R**
	Enable training	**0**	**R/W**
0xC	**Received data**		
	Received data which is different than 0	**31:16**	**R**
	Received raw data	**15:0**	**R**
0x10	**Testing control**		
	Reserved	*31:1*	*R*
	Reset testing counters (error & data)	**0**	**R/W**
0x14	**Testing error counter**		
	Error increases if received data is not the same as transmitted testing data	**31:0**	**R**
0x18	**Testing data counter**		
	Counter increases when value different as 0 is received	**31:0**	**R**

Power Test

offset	description	bits	R/W
0x0	**Control**		
	Reserved	*31:1*	*R*
	Enable module	**0**	**R/W**