Register file structure : regfile_xgs_ctrl.pdf
Created by jmansill on 2020/04/21 14:33:45
Register file CRC32 : 0x4BE916FE

1. Main Parameters

Register file endianness: little endian
Address bus width: 12 bits
Data bus width: 32 bits

2. Memory Map

Section name	Address(es) / Address Ranges	Register name	Access Type
SYSTEM	0x000	ID	R
	0x030	ACQ_CAP	R
ACQ	0x100	GRAB_CTRL	RW
	0x108	GRAB_STAT	R
	0x110	READOUT_CFG1	RW
	0x114	READOUT_CFG_FRAME_LINE	RW
	0x118	READOUT_CFG2	R
	0x120	READOUT_CFG3	RW
	0x124	READOUT_CFG4	RW
	0x128	EXP_CTRL1	RW
	0x130	EXP_CTRL2	RW
	0x138	EXP_CTRL3	RW
	0x140	TRIGGER_DELAY	RW
	0x148	STROBE_CTRL1	RW
	0x150	STROBE_CTRL2	RW
	0x158	ACQ_SER_CTRL	RW
	0x160	ACQ_SER_ADDATA	RW
	0x168	ACQ_SER_STAT	R
	0x190	SENSOR_CTRL	RW
	0x198	SENSOR_STAT	R
	0x19C	SENSOR_SUBSAMPLING	RW
	0x1A4	SENSOR_GAIN_ANA	RW
	0x1A8	SENSOR_ROI_Y_START	RW
	0x1AC	SENSOR_ROI_Y_SIZE	RW
	0x1B0	SENSOR_ROI2_Y_START	RW
	0x1B4	SENSOR_ROI2_Y_SIZE	RW
	0x1B8	SENSOR_M_LINES	RW
	0x1BC	SENSOR_F_LINES	RW

Section name	Address(es) / Address Ranges	Register name	Access Type
	0x1C0	SENSOR_DP_GR	RW
	0x1C4	SENSOR_DP_GB	RW
	0x1C8	SENSOR_DP_R	RW
	0x1CC	SENSOR_DP_B	RW
	0x1E0	DEBUG_PINS	RW
	0x1E8	TRIGGER_MISSED	RW
	0x1F0	SENSOR_FPS	R
	0x2A0	DEBUG	RW
	0x2A8	DEBUG_CNTR1	R
	0x2B0	DEBUG_CNTR2	R
	0x2B4	DEBUG_CNTR3	R
	0x2B8	EXP_FOT	RW
	0x2C0	ACQ_SFNC	RW

3. Registers definition

Section: SYSTEM

Address Range: [0x000 - 0x030]

ID

Address: section "SYSTEM" base address + 0x000

Description:
Static ID

31	30	29	28	27	26	25	24
StaticID(31:24)							
23	22	21	20	19	18	17	16
StaticID(23:16)							
15	14	13	12	11	10	9	8
StaticID(15:8)							
7	6	5	4	3	2	1	0
StaticID(7:0)							

StaticID (31:0) *RO*	MINUTEs of the build

ACQ_CAP

Address: section "SYSTEM" base address + 0x030

31	30	29	28	27	26	25	24
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
DPC	EXP_FOT	FPN_73	COLOR	CH_LVDS(3:0)			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved			LUT_WIDTH	Reserved		LUT_PALETTE(1:0)	

DPC		
STATIC	This field indicate if the Dead Pixel Corection is implemented in the fpga.	
Value at Reset:	0x1	
Possible Values:	0x0	DPC is not implemented
	0x1	DPC is implemented

EXP_FOT	
STATIC	This field indicate if the Exposure during FOT is implemented in the fpga.
Value at Reset:	0x1

FPN_73	FPN 7.3 correction CAP	
RO	This field indicate if the FPN correction 7.3 is implemented instead the default 5.3 in the FPGA acquisition.	
Possible Values:	0x0	Default 5.3 FPN correction implemented
	0x1	New 7.3 FPN correction implemented

COLOR	
RO	This field indicate if the COLOR path is implemented in the FPGA acquisition.

CH_LVDS (3:0)	
RO	This is the number of LVDS DATA channels connected between the CMOS sensor and the FPGA. This number does not include the LVDS CTROL channel.

LUT_WIDTH		
RO	This is the LUT width implemented in the acquisition core.	
Possible Values:	0x0	LUT 10 to 10 Bits
	0x1	LUT 10 to 8 Bits

LUT_PALETTE (1:0)		
RO	This is the number of LUT palette implemented in the acquisition module	
Possible Values:	0x0	No LUT implemented
	0x1	1 LUT implemented
	0x2	2 LUT implemented

Section: ACQ

Address Range: [0x100 - 0x2CC]

GRAB_CTRL — GRAB ConTRoL Register

Address: section "ACQ" base address + 0x000

Description:

Grag Control Register

31	30	29	28	27	26	25	24
RESET_GRAB	Reserved	GRAB_ROI2_EN	ABORT_GRAB	Reserved			

23	22	21	20	19	18	17	16
Reserved							TRIGGER_OVERLAP_BUFFn

15	14	13	12	11	10	9	8
TRIGGER_OVERLAP	TRIGGER_ACT(2:0)			Reserved	TRIGGER_SRC(2:0)		

7	6	5	4	3	2	1	0
Reserved			GRAB_SS	Reserved		BUFFER_ID	GRAB_CMD

RESET_GRAB		
RW	This register resets the entire python_ctrl.	
Value at Reset:	0x0	
Possible Values:	0x0	Reset not active
	0x1	Reset active

GRAB_ROI2_EN		
RW	Enable the second ROI on the frame (KNS). This register is not DB. 1) No Y overlap is allowed 2) Xsize must be the same for the two ROI for the moment(DMA constraint). 3) EOF and SOF in between the two in-frame ROIs will be masked to the DMA. The DMA will see one frame, with the two ROI inside.	
Value at Reset:	0x0	
Possible Values:	0x0	Dual ROI disable
	0x1	Dual ROI enable

ABORT_GRAB	ABORT GRAB	
WO/AutoClr	This is the grab reset, it will reset all the grab queued.	
Possible Values:	0x0	Normal operation
	0x1	Reset Grab

TRIGGER_OVERLAP_BUFFn		
RW	NOT FULLY VALIDATED. DON'T USE. SET IT TO '0'.	
Value at Reset:	0x0	
Possible Values:	0x0	Buffer the trigger received during the dead window in PET mode and execute
	0x1	The trigger will be ignored during dead window in PET mode.

TRIGGER_OVERLAP		
RW	This field enables the trigger overlap. In this mode the exposure and the readout of the sensor can be done in parallel for higher framerates.	
Value at Reset:	0x1	
Possible Values:	0x0	Trigger Overlap disable
	0x1	Trigger Overlap enable (default)

TRIGGER_ACT (2:0)	TRIGGER ACTivation	
RW	This is the trigger activation . This register selects the activation of the trigger when the trigger source is set to Hardware Snapshop mode . This register is Double Buffered, so the trigger activation may change from one grab command to another. In activation Level HI/LO with EXPOSURE_MODE register set to Timed, the camera will be triggered in continuous way if the level of the external trigger remains at the LEVEL programmed in this register. In activation Level HI/LO with EXPOSURE_MODE register set to Trigger Width, the Exposure time will be set by the level of the trigger input. The FPGA exposure regsiters will be ignored. The Dual and Triple slope are not supported in the mode.	
Value at Reset:	0x0	
Possible Values:	0x0	Rising edge
	0x1	Falling edge
	0x2	Rising or Falling edge
	0x3	Level HI
	0x4	Level LO
	0x5	RESERVED
	0x6	RESERVED
	0x7	RESERVED

TRIGGER_SRC (2:0)	TRIGGER SouRCe	
RW	This is the trigger source. This register selects the source of the grab trigger. This register is Double Buffered, so the trigger source may change from one grab command to another. TRIGGER_SRC(1) may be seen as a TRIGGER_STATE by the software driver.	
Value at Reset:	0x0	
Possible Values:	0x0	RESERVED
	0x1	Immediate mode (Continuous)
	0x2	Hardware Snapshop mode
	0x3	Software Snapshot mode
	0x4	SFNC mode (auto trig)

GRAB_SS	GRAB Software Snapshot	
WO/AutoClr	This is the software snapshot register when the trigger source selected is Software Snapshot mode.	
Possible Values:	0x0	Idle
	0x1	Start a grab

BUFFER_ID	
RW	This is the ID of the DMA parameters to associate with this grab command.
Value at Reset:	0x0

GRAB_CMD	GRAB CoMmanD	
WO/AutoClr	This is MIL GRAB command. When the trigger source is set to Immediate mode(Continuous), an exposure sequence will be automatically executed. When the trigger source is set to Software Snapshop mode or Hardware Snapshop mode, GRAB_CMD will act as an ARM. The GRAB_CMD will take around 13 clks to reccord the grab parametters to the SPI fifo. The GRAB_CMD_DONE register may be readed to avoid fifo corruption before sending another Grab command instruction.	
Possible Values:	0x0	Idle
	0x1	Start grab command

GRAB_STAT

Address: section "ACQ" base address + 0x008

31	30	29	28	27	26	25	24
GRAB_CMD_DONE	ABORT_PET	ABORT_DELAI	ABORT_DONE	Reserved			TRIGGER_RDY
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved	ABORT_MNGR_STAT(2:0)			TRIG_MNGR_STAT(3:0)			
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Reserved	TIMER_MNGR_STAT(2:0)			GRAB_MNGR_STAT(3:0)			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved	GRAB_FOT	GRAB_READOUT	GRAB_EXPOSURE	Reserved	GRAB_PENDING	GRAB_ACTIVE	GRAB_IDLE

GRAB_CMD_DONE	GRAB CoMmanD DONE	
RO	The GRAB_CMD will take around 13 clks to reccord the grab parametters to the SPI fifo. This register may be readed to avoid fifo corruption before sending another Grab command instruction.	
Possible Values:	0x0	Grab Command in process
	0x1	Grab command idle

ABORT_PET	ABORT during PET	
RO	This is the ABORT PET flag. It is set to '1' when an abort is detected in the PETengin phase of the trigger. It is set back to '0' when ABORT_DONE is set to '1'.	
Possible Values:	0x0	Abort in PET Phase idle
	0x1	Abort in PET Phase active

ABORT_DELAI		
RO	This is the ABORT DELAI flag. It is set to '1' when an abort is detected in the delai phase of the trigger. It is set back to '0' when ABORT_DONE is set to '1'.	
Possible Values:	0x0	Abort in Delai Phase idle
	0x1	Abort in Delai Phase active

ABORT_DONE	ABORT is DONE	
RO	This read-only field indicates the RESET_GRAB command status. If 0, an abort sequence is executing.	
Possible Values:	0x0	Abort sequence not finished yet
	0x1	Abort DONE, or not started (reset value)

TRIGGER_RDY	
RO	

ABORT_MNGR_STAT (2:0)	
RO	DEBUG ABORT MANAGER STATE MACHINE

TRIG_MNGR_STAT (3:0)	
RO	DEBUG TRIGGER MANAGER STATE MACHINE

TIMER_MNGR_STAT (2:0)	
RO	DEBUG TIMER MANAGER STATE MACHINE

GRAB_MNGR_STAT (3:0)	
RO	DEBUG GRAB MANAGER STATE MACHINE

GRAB_FOT	GRAB Field Overhead Time	
RO	This is the sensor FOT (Field Overhead Time).	
Possible Values:	0x0	Not in FOT
	0x1	In FOT

GRAB_READOUT	
RO	This is the sensor readout status. It goes to '1' on the SO_FOT and goes to '0' when the datapath decoder decodes the end of frame.

GRAB_EXPOSURE		
RO	This is the sensor integration status	
Possible Values:	0x0	Idle
	0x1	Integrating

GRAB_PENDING		
RO	Grab pending status. When this register is set to one, a second grab command is queued in the fpga.	
Possible Values:	0x0	No grab pending
	0x1	Grab pending

GRAB_ACTIVE	
RO	Grab active status. When this register is set to one, at least one grab command has been received.

GRAB_IDLE		
RO	GRAB IDLE status. When this register is set to '1', The grab engin is in idle state.	
Possible Values:	0x0	Grab is in proccess
	0x1	Grab is Idle

READOUT_CFG1

Address: section "ACQ" base address + 0x010

31	30	29	28	27	26	25	24
Reserved			FOT_LENGTH_LINE(4:0)				
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							EO_FOT_SEL
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
FOT_LENGTH(15:8)							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
FOT_LENGTH(7:0)							

FOT_LENGTH_LINE (4:0)	Frame Overhead Time LENGTH LINE	
RW	This is the length of the Frame Overhead Time in line_time unit. For debug only for the moment.	
Value at Reset:	0x0	
Possible Values:	Any Value	Any 16 bit value

EO_FOT_SEL	
RW	This selector selects who will generate the EO_FOT in the controller. When select 0, the EO_FOT is the falling edge detection of the monitor FOT. When select 1, the EO_FOT will be generated inside the controller with programmed FOT_LENGTH.
Value at Reset:	0x0

FOT_LENGTH (15:0)	Frame Overhead Time LENGTH	
RW	This is the length of the Frame Overhead Time. This register is defined as number of lines. It is used when EO_FOT_SEL is set to 1.	
Value at Reset:	0x0	
Possible Values:	Any Value	Any 16 bit value

READOUT_CFG_FRAME_LINE

Address: section "ACQ" base address + 0x014

31	30	29	28	27	26	25	24
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
DUMMY_LINES(7:0)							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Reserved			CURR_FRAME_LINES(12:8)				
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CURR_FRAME_LINES(7:0)							

DUMMY_LINES (7:0)	
RW	Number of lines to add in the readout (to debug XGS)
Value at Reset:	0x0

CURR_FRAME_LINES (12:0)	
RO	Current number of lines in the readout calculated by the XGS controller (without FOT).

READOUT_CFG2

Address: section "ACQ" base address + 0x018

31	30	29	28	27	26	25	24
Reserved			READOUT_LENGTH(28:24)				
23	22	21	20	19	18	17	16
READOUT_LENGTH(23:16)							
15	14	13	12	11	10	9	8
READOUT_LENGTH(15:8)							
7	6	5	4	3	2	1	0
READOUT_LENGTH(7:0)							

READOUT_LENGTH (28:0)		
RO	This is the readout length register. This register is a register software calculated that gives the readout lenght to the grab engin. This register will depend on the ROI, Subsampling, Binning and LVDS channels used. It is used in the PET engin calculations. In Sys_Clock domain.	
Possible Values:	Any Value	Any 24 bits value

READOUT_CFG3

Address: section "ACQ" base address + 0x020

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							KEEP_OUT_TRIG_ENA
15	14	13	12	11	10	9	8
LINE_TIME(15:8)							
7	6	5	4	3	2	1	0
LINE_TIME(7:0)							

KEEP_OUT_TRIG_ENA	
RW	KEEPOUT zone TRIGger ENAble. When this register is enabled, then the trigger output will be synchronized with the line_int(monitor2) signal from the XGS sensor. To configure this keep out zone, use register READOUT_CFG4.
Value at Reset:	0x0

LINE_TIME (15:0)	LINE TIME	
RW	Line Time Unit is SENSOR Clock Cycles	
Value at Reset:	0x16e	
Possible Values:	Any Value	between 1 and 255

READOUT_CFG4

Address: section "ACQ" base address + 0x024

31	30	29	28	27	26	25	24
KEEP_OUT_TRIG_END(15:8)							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
KEEP_OUT_TRIG_END(7:0)							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
KEEP_OUT_TRIG_START(15:8)							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
KEEP_OUT_TRIG_START(7:0)							

KEEP_OUT_TRIG_END (15:0)	
RW	During the line time, this register indicates the end of the trigger keep-out zone.
Value at Reset:	0x16d

KEEP_OUT_TRIG_START (15:0)	
RW	During the line time, this register indicates the start of the trigger keep-out zone.
Value at Reset:	0x16e

EXP_CTRL1

Address: section "ACQ" base address + 0x028

31	30	29	28	27	26	25	24
Reserved			EXPOSURE_LEV_MODE	EXPOSURE_SS(27:24)			
23	22	21	20	19	18	17	16
EXPOSURE_SS(23:16)							
15	14	13	12	11	10	9	8
EXPOSURE_SS(15:8)							
7	6	5	4	3	2	1	0
EXPOSURE_SS(7:0)							

EXPOSURE_LEV_MODE	EXPOSURE LEVel MODE	
RW	This is the exposure level mode selector. When selecting the TRIGGER ACTIVATION = Level Mode, this register selects the exposure method used. When this register is set to '0' the timed mode is selected; Register EXPOSURE_SS is used for the exposure time. When this register is set to '1' the external trigger width is used for the exposure time.	
Value at Reset:	0x0	
Possible Values:	0x0	Timed Mode
	0x1	Trigger Width

EXPOSURE_SS (27:0)	EXPOSURE Single Slope	
RW	This is the total exposure time in single/dual/triple slope mode. This register is double buffered.	
Value at Reset:	0x0	
Possible Values:	Any Value	Any 28 bits value

EXP_CTRL2

Address: section "ACQ" base address + 0x030

31	30	29	28	27	26	25	24
Reserved				EXPOSURE_DS(27:24)			
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
EXPOSURE_DS(23:16)							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
EXPOSURE_DS(15:8)							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
EXPOSURE_DS(7:0)							

EXPOSURE_DS (27:0)	EXPOSURE Dual	
RW	This is a new 3d profiler feature We will be able to program upto 3 diferent exposure times (using unused multiSlope registers) Then we will be able to sequence those exposure times . Selection is made with input exposure_select.	
Value at Reset:	0x0	
Possible Values:	Any Value	Any 28 bits value

EXP_CTRL3

Address: section "ACQ" base address + 0x038

31	30	29	28	27	26	25	24
Reserved				EXPOSURE_TS(27:24)			
23	22	21	20	19	18	17	16
EXPOSURE_TS(23:16)							
15	14	13	12	11	10	9	8
EXPOSURE_TS(15:8)							
7	6	5	4	3	2	1	0
EXPOSURE_TS(7:0)							

EXPOSURE_TS (27:0)	EXPOSURE Tripple	
RW	This is a new 3d profiler feature We will be able to program upto 3 diferent exposure times (using unused multiSlope registers) Then we will be able to sequence those exposure times . Selection is made with input exposure_select.	
Value at Reset:	0x0	
Possible Values:	Any Value	Any 28 bits value

TRIGGER_DELAY

Address: section "ACQ" base address + 0x040

31	30	29	28	27	26	25	24
Reserved				TRIGGER_DELAY(27:24)			
23	22	21	20	19	18	17	16
TRIGGER_DELAY(23:16)							
15	14	13	12	11	10	9	8
TRIGGER_DELAY(15:8)							
7	6	5	4	3	2	1	0
TRIGGER_DELAY(7:0)							

TRIGGER_DELAY (27:0)	TRIGGER DELAY	
RW	This is the trigger delay. This trigger delay can be applied to HW(Only edge mode), SW and Continuous mode. In HW level mode, the trigger cannot be delayed, since the level time represents the exposure time. This register is double buffered	
Value at Reset:	0x0	
Possible Values:	Any Value	Any 28 bits value

STROBE_CTRL1

Address: section "ACQ" base address + 0x048

31	30	29	28	27	26	25	24
STROBE_E	Reserved		STROBE_POL	STROBE_START(27:24)			
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
STROBE_START(23:16)							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
STROBE_START(15:8)							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
STROBE_START(7:0)							

STROBE_E	STROBE Enable	
RW	This register enables the strobe logic. For Nexis 3 systems, to enable STROBE_A signal, STROBE_E and STROBE_A_EN must be enabled. For Nexis 3 systems, to enable STROBE_B signal, STROBE_E and STROBE_B_EN must be enabled. For Nexis 3 systems, STROBE_A and STROBE B can be activated at the same time, in this case the two strobes will be the same as they share the same programmation. This register is double buffered	
Value at Reset:	0x0	
Possible Values:	0x0	Strobe disabled
	0x1	Strobe enabled

STROBE_POL	STROBE POLarity	
RW	This is the strobe polarity at the pin of the FPGA only for GTR systems. For NEXIS3 systems use register ANPUT\IO\IO_OUT_POL\OUTx_POL This register is not double buffered.	
Value at Reset:	0x0	
Possible Values:	0x0	Active high strobe
	0x1	Active low strobe

STROBE_START (27:0)	STROBE START	
RW	This is the strobe start location. This location depends on the Strobe Mode used. In Strobe Mode='0', the start of the strobe is situated during the exposure time. In Strobe Mode='1', the start of the strobe is situated during the trigger delay. This register is double buffered	
Value at Reset:	0x0	
Possible Values:	Any Value	Any 28 bits value

STROBE_CTRL2

Address: section "ACQ" base address + 0x050

31	30	29	28	27	26	25	24
STROBE_MODE	Reserved	STROBE_B_EN	STROBE_A_EN	STROBE_END(27:24)			
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
STROBE_END(23:16)							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
STROBE_END(15:8)							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
STROBE_END(7:0)							

STROBE_MODE	STROBE MODE	
RW	This register selects the location of the Strobe Start. When this register is set to 0, the STROBE_START register is located during the exposure timer. When this register is set to 1, the STROBE_START register is located during the trigger delay timer. In HW level mode the strobe mode must be set to STROBE MODE=0 since the trigger cannot be delayed. This register is double buffered	
Value at Reset:	0x0	
Possible Values:	0x0	Strobe start during exposure
	0x1	Strobe start during trigger delay

STROBE_B_EN	STROBE phase B ENable	
RW	This field enables the generation of STROBE_B signal, for a NEXIS 3 system. This register is double buffered to support back2back mode in nexis systems.	
Value at Reset:	0x0	
Possible Values:	0x0	Enable Strobe B
	0x1	Disable Strobe B

STROBE_A_EN	STROBE phase A ENable	
RW	This field enables the generation of STROBE_A signal(Default strobe), for a NEXIS 3 system. This register is double buffered to support back2back mode in nexis systems.	
Value at Reset:	0x1	
Possible Values:	0x0	Enable Strobe A (default strobe)
	0x1	Disable Strobe A

STROBE_END (27:0)	STROBE END	
RW	This is the strobe end location. This location does not depend on the Strobe Mode used. This register is double buffered	
Value at Reset:	0xfffffff	
Possible Values:	Any Value	Any 28 bits value

Address: section "ACQ" base address + 0x058

31	30	29	28	27	26	25	24
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							SER_RWn
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Reserved						SER_CMD(1:0)	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved			SER_RF_SS	Reserved			SER_WF_SS

SER_RWn	SERial Read/Writen	
RW	This register configures the type of the serial access to the CMOS sensor	
Value at Reset:	0x1	
Possible Values:	0x0	Write access
	0x1	Read access

SER_CMD (1:0)	SERial CoMmand	
RW	This is the type of command sent to the serial fifo. To access the Sensor, write SER_WF_SS=1 with SER_CMD=0x0, with the parametters: SER_WRn, SER_ADD(8:0) and SER_DAT(15:0). To insert a timer between fifo commands, write SER_WF_SS=1 with SER_CMD=0x1, with the parameter: SER_DAT(15:0). The value of the timer inserted is calculated with the following formula: Timer= SER_DAT(15:0)*1024*SYS_PERIOD, SYS_PERIOD is 1/62.5mhz. The granularity of the timer is 16.384us To insert a Stop separator command, write SER_WF_SS=1 with SER_CMD=0x3. When the read logic encounter this command, it will stop read from the fifo until a new SER_RF_SS is received.	
Value at Reset:	0x0	
Possible Values:	0x0	CMOS sensor access COMMAND
	0x1	Insert timer COMMAND
	0x2	STOP separator COMMAND
	0x3	RESERVED

SER_RF_SS	SERial Read Fifo SnapShot	
WO/AutoClr	This is the read fifo snapshot. When the read fifo logic receives this snapshot, it will read all the fifo comands until a STOP separator command is read or Empty fifo is detected.	
Possible Values:	0x0	Idle
	0x1	Start Read FIFO

SER_WF_SS	SERial Write Fifo SnapShot	
WO/AutoClr	When the system toggle this bit, the address, data and command are wrote to the command fifo. This fifo can contain the entire dcf, so the driver will not need to pool the status bit. This is a auto reset bit register, so after the driver write one, the bit will be auto reset to 0. To start the FIFO read logic write '1' to regsiter SER_RF_SS.	
Possible Values:	0x0	Idle
	0x1	Write a command to the FIFO

Address: section "ACQ" base address + 0x060

31	30	29	28	27	26	25	24
SER_DAT(15:8)							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
SER_DAT(7:0)							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Reserved	SER_ADD(14:8)						
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
SER_ADD(7:0)							

SER_DAT (15:0)	SERial interface DATa	
RW	This is the write data to be send to the CMOS sensor by the serial interface, or the config data to a TIMER command or to a POWER sequence command. See register SER_CMD.	
Value at Reset:	0x0	
Possible Values:	Any Value	Any 16 bits value

SER_ADD (14:0)	SERial interface ADDress	
RW	This is the read/write address of the register in the CMOS sensor.	
Value at Reset:	0x0	
Possible Values:	Any Value	Any 9 bits value

Address: section "ACQ" base address + 0x068

31	30	29	28	27	26	25	24
Reserved							SER_FIFO_EMPTY
23	22	21	20	19	18	17	16
Reserved							SER_BUSY
15	14	13	12	11	10	9	8
SER_DAT_R(15:8)							
7	6	5	4	3	2	1	0
SER_DAT_R(7:0)							

SER_FIFO_EMPTY	SERial FIFO EMPTY	
RO	This is the EMPTY flag of the xilinx fifo, when '1' there are no pending operations in the fifo.	

SER_BUSY	SERial BUSY	
RO	This is the BUSY status of the FIFO read logic. The flag will be set to '1' when the SER_RF_SS is set to '1'. It will be reseted to '0' when the read logic will decode a STOP separator command or when the FIFO will be empty.	
Possible Values:	0x0	FIFO read logic is idle
	0x1	FIFO read logic is runnning

SER_DAT_R (15:0)	SERial interface DATa Read	
RO	This is the data read from CMOS sensor.	
Possible Values:	Any Value	Any 16 bits value

Address: section "ACQ" base address + 0x090

31	30	29	28	27	26	25	24
Reserved							SENSOR_REFRESH_TEMP
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							SENSOR_POWERDOWN
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Reserved							SENSOR_COLOR
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved			SENSOR_REG_UPTATE	Reserved		SENSOR_RESETN	SENSOR_POWERUP

SENSOR_REFRESH_TEMP	SENSOR REFRESH TEMPerature	
WO/AutoClr	This register starts a sensor temperature read on the serial interface of the Python sensor. The temperature value readed will be available on field SENSOR_TEMP when field SENSOR_TEMP_VALID is set to '1'.	
Possible Values:	0x0	Idle
	0x1	Starts a Temperature read on Python SPI interface

SENSOR_POWERDOWN	
WO/AutoClr	After a PowerUp sequence(SESOR_POWERUP_DONE=1), successfull or not, this register can reset the clock oscillator and enable the reset to the sensor. This power down don't do power sequencing.

SENSOR_COLOR	SENSOR COLOR	
RW	This register informs the datapath logic that a color sensor is used. This information is needed for the remapper logic.	
Value at Reset:	0x0	
Possible Values:	0x0	Monochrone sensor
	0x1	Color sensor

SENSOR_REG_UPTATE	SENSOR REGister UPDATE	
RW	By setting this bit to 1, the SENSOR CONTROLLER WILL UPDATE the programed CMOS sensor registers at the beginning of each grab.	
Value at Reset:	0x1	
Possible Values:	0x0	Do not update registers
	0x1	Update registers

SENSOR_RESETN	SENSOR RESET Not	
RW	After a successfull PowerUP sequence, writing this field to '0' reset the Python CMOS sensor.	
Value at Reset:	0x1	
Possible Values:	0x0	Reset the sensor after a successfull powerUP
	0x1	Nothing

SENSOR_POWERUP		
WO/AutoClr	This register Enables the clk oscillator and removes the reset from the sensor.	
Possible Values:	0x0	idle
	0x1	Start the power sequence

Address: section "ACQ" base address + 0x098

31	30	29	28	27	26	25	24
SENSOR_TEMP(7:0)							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
SENSOR_TEMP_VALID	Reserved						SENSOR_POWERDOWN
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Reserved		SENSOR_RESETN	SENSOR_OSC_EN	Reserved			SENSOR_VCC_PG
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved						SENSOR_POWERUP_STAT	SENSOR_POWERUP_DONE

SENSOR_TEMP (7:0)		
RO	This register gives the Temperature of the Python sensor after a SENSOR_REFRESH_TEMP snapshot. The field SENSOR_TEMP_VALID indicates when the SENSOR_TEMP value is valid.	
Possible Values:	Any Value	

SENSOR_TEMP_VALID	SENSOR TEMPerature VALID	
RO	This field indicates that the field SENSOR_TEMP have valid temperature after a SENSOR_REFRESH_TEMP snapshot.	
Possible Values:	0x0	SENSOR_TEMPERATURE register is not valid
	0x1	SENSOR_TEMPERATURE register is valid

SENSOR_POWERDOWN		
RO	This field indicates that the sensor is in powerdown state.	
Possible Values:	0x0	Not in powerdown state
	0x1	Powerdown

SENSOR_RESETN	SENSOR RESET N	
RO	This is the sensor RESETN status.	
Possible Values:	0x0	In reset state
	0x1	Not in reset

SENSOR_OSC_EN	SENSOR OSCILLATOR ENable	
RO	This is the sensor oscillator enable status.	
Possible Values:	0x0	Disable
	0x1	Enable

SENSOR_VCC_PG	SENSOR supply VCC Power Good	
RO	This is the VCC Power Good status (generated by external HW).	
Possible Values:	0x0	Disable
	0x1	Enable

SENSOR_POWERUP_STAT		
RO	When a powerup sequence is finish, this register indicates the result of the POWERUP sequence.	
Possible Values:	0x0	PowerUp sequence fail
	0x1	PowerUp sequence success

SENSOR_POWERUP_DONE		
RO	This register indicates that the POWERUP sequence is finish. Read register SENSOR_POWERUP_STAT to see the result.	
Possible Values:	0x0	PowerUp sequence not started
	0x1	PowerUp sequence finish

SENSOR_SUBSAMPLING

Address: section "ACQ" base address + 0x09C

Description:
SENSOR ADDRESS

31	30	29	28	27	26	25	24
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
reserved1(11:4)							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
reserved1(3:0)				ACTIVE_SUBSAMPLING_Y	reserved0	M_SUBSAMPLING_Y	SUBSAMPLING_X

reserved1 (11:0) *STATIC*		
Value at Reset:	0x0	

ACTIVE_SUBSAMPLING_Y *RW*	Subsampling (Row) for ROI Configurations	
Value at Reset:	0x0	
Possible Values:	0x0	
	0x1	

reserved0 *STATIC*		
Value at Reset:	0x0	
Possible Values:	0x0	Idle
	0x1	Enable

M_SUBSAMPLING_Y *RW*	Subsampling (Row) for M Region	
Value at Reset:	0x0	
Possible Values:	0x0	
	0x1	

SUBSAMPLING_X *RW*	Readout in Column Subsampling Mode	
Value at Reset:	0x0	
Possible Values:	0x0	
	0x1	

SENSOR_GAIN_ANA

Address: section "ACQ" base address + 0x0A4

Description:

SENSOR ADDRESS 204 DEC

31	30	29	28	27	26	25	24
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
reserved1(4:0)					ANALOG_GAIN(2:0)		
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
reserved0(7:0)							

reserved1 (4:0) *STATIC*	
Value at Reset:	0x0

ANALOG_GAIN (2:0) *RW*		
Value at Reset:	0x1	
Possible Values:	0x1	1x
	0x3	2x
	0x7	4x

reserved0 (7:0) *STATIC*	
Value at Reset:	0x0

SENSOR_ROI_Y_START

Address: section "ACQ" base address + 0x0A8

Description:

SENSOR ADDRESS

31	30	29	28	27	26	25	24
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
reserved(5:0)						Y_START(9:8)	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Y_START(7:0)							

reserved (5:0)	
STATIC	
Value at Reset:	0x0

Y_START (9:0)	Y START
RW	Y Start in Kernel size (Kernel is 4 lines)
Value at Reset:	0x0

SENSOR_ROI_Y_SIZE

Address: section "ACQ" base address + 0x0AC

Description:
SENSOR ADDRESS

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
reserved(5:0)						Y_SIZE(9:8)	
7	6	5	4	3	2	1	0
Y_SIZE(7:0)							

reserved (5:0)	
STATIC	
Value at Reset:	0x0

Y_SIZE (9:0)	Y SIZE
RW	Y SIZE in Kernel size (Kernel is 4 lines)
Value at Reset:	0x302

SENSOR_ROI2_Y_START

Address: section "ACQ" base address + 0x0B0

Description:

SENSOR ADDRESS

31	30	29	28	27	26	25	24
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
reserved(5:0)						Y_START(9:8)	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Y_START(7:0)							

reserved (5:0)	
STATIC	
Value at Reset:	0x0

Y_START (9:0)	Y START
RW	Y Start in Kernel size (Kernel is 4 lines)
Value at Reset:	0x0

SENSOR_ROI2_Y_SIZE

Address: section "ACQ" base address + 0x0B4

Description:
SENSOR ADDRESS

31	30	29	28	27	26	25	24
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
reserved(5:0)						Y_SIZE(9:8)	
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Y_SIZE(7:0)							

reserved (5:0)	
STATIC	
Value at Reset:	0x0

Y_SIZE (9:0)	Y SIZE
RW	Y SIZE in Kernel size (Kernel is 4 lines)
Value at Reset:	0x302

SENSOR_M_LINES

Address: section "ACQ" base address + 0x0B8

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved	M_SUPPRESSED(4:0)					M_LINES_SENSOR(9:8)	
7	6	5	4	3	2	1	0
M_LINES_SENSOR(7:0)							

M_SUPPRESSED (4:0) *RW*	Suppress the Readout of Initial Lines in the M Region
Value at Reset:	0x0

M_LINES_SENSOR (9:0) *RW*	Number of Lines to Readout from M Region in Context 0 Unit is #lines Total number of Black lines = M_LINES-M_SUPRESSED
Value at Reset:	0x8

SENSOR_F_LINES

Address: section "ACQ" base address + 0x0BC

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
Reserved	F_SUPPRESSED(4:0)					F_LINES_SENSOR(9:8)	
7	6	5	4	3	2	1	0
F_LINES_SENSOR(7:0)							

F_SUPPRESSED (4:0)	
RW	Suppress the Readout of Initial Lines in the F Region , Unit is #lines
Value at Reset:	0x0

F_LINES_SENSOR (9:0)	
RW	Number of Lines to Read from F Region in Context 0 Unit is #lines
Value at Reset:	0x8

SENSOR_DP_GR

Address: section "ACQ" base address + 0x0C0

Description:

Sensor Analog data pedestal for Gr pixels (Black offset)

31	30	29	28	27	26	25	24
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
reserved(3:0)				DP_OFFSET_GR(11:8)			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
DP_OFFSET_GR(7:0)							

reserved (3:0)	
STATIC	
Value at Reset:	0x0

DP_OFFSET_GR (11:0)	
RW	Sensor Analog data pedestal for Gr pixels (Black offset)
Value at Reset:	0x100

SENSOR_DP_GB

Address: section "ACQ" base address + 0x0C4

Description:

Sensor Analog data pedestal for Gb pixels (Black offset)

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							
15	14	13	12	11	10	9	8
reserved(3:0)				DP_OFFSET_GB(11:8)			
7	6	5	4	3	2	1	0
DP_OFFSET_GB(7:0)							

reserved (3:0)	
STATIC	
Value at Reset:	0x0

DP_OFFSET_GB (11:0)	
RW	Sensor Analog data pedestal for Gb pixels (Black offset)
Value at Reset:	0x100

SENSOR_DP_R

Address: section "ACQ" base address + 0x0C8

Description:

Sensor Analog data pedestal for R pixels (Black offset)

31	30	29	28	27	26	25	24
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
reserved(3:0)				DP_OFFSET_R(11:8)			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
DP_OFFSET_R(7:0)							

reserved (3:0)	
STATIC	
Value at Reset:	0x0

DP_OFFSET_R (11:0)	
RW	Sensor Analog data pedestal for R pixels (Black offset)
Value at Reset:	0x100

SENSOR_DP_B

Address: section "ACQ" base address + 0x0CC

Description:

Sensor Analog data pedestal for B pixels (Black offset)

31	30	29	28	27	26	25	24
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
reserved(3:0)				DP_OFFSET_B(11:8)			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
DP_OFFSET_B(7:0)							

reserved (3:0)	
STATIC	
Value at Reset:	0x0

DP_OFFSET_B (11:0)	
RW	Sensor Analog data pedestal for B pixels (Black offset)
Value at Reset:	0x100

DEBUG_PINS

Address: section "ACQ" base address + 0x0E0

31	30	29	28	27	26	25	24
Reserved			Debug3_sel(4:0)				
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved			Debug2_sel(4:0)				
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Reserved			Debug1_sel(4:0)				
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved			Debug0_sel(4:0)				

Debug3_sel (4:0)	

RW	debug_vector(0x0) <= python_monitor0;
	debug_vector(0x1) <= python_monitor1;
	debug_vector(0x2) <= grab_mngr_trig_rdy;
	debug_vector(0x3) <= curr_trig0;
	debug_vector(0x4) <= strobe;
	debug_vector(0x5) <= python_exposure;
	debug_vector(0x6) <= FOT;
	debug_vector(0x7) <= readout;
	debug_vector(0x8) <= readout_stateD;
	debug_vector(0x9) <= ext_trig;
	debug_vector(0xa) <= REGFILE.ACQ.GRAB_CTRL.GRAB_CMD;
	debug_vector(0xb) <= REGFILE.ACQ.GRAB_CTRL.GRAB_SS;
	debug_vector(0xc)<= grab_mngr_trig;
	debug_vector(0xd) <= grab_mngr_trig_rdy;
	debug_vector(0xe) <= grab_pending;
	debug_vector(0xf) <= grab_active;
	debug_vector(0x10) <= DEC_DATA_EN;
	debug_vector(0x11) <= DEC_SOL;
	debug_vector(0x12) <= DEC_SOF;
	debug_vector(0x13) <= DEC_EOL;
	debug_vector(0x14) <= DEC_EOF;
	debug_vector(0x15) <= DEC_CRC;
	debug_vector(0x16) <= DEC_TRAIN;
	debug_vector(0x17) <= fpnprnu_corr_sof;
	debug_vector(0x18) <= fpnprnu_corr_sol;
	debug_vector(0x19) <= fpnprnu_corr_data_val;
	debug_vector(0x1a) <= fpnprnu_corr_eol;
	debug_vector(0x1b) <= fpnprnu_corr_eof;
	debug_vector(0x1c) <= python_ssn_int;
	debug_vector(0x1d) <= debug_lvds(0);
	debug_vector(0x1e) <= debug_lvds(1);
	debug_vector(0x1f) <= 'Z';

Value at Reset:	0x1f

Debug2_sel (4:0)	

RW	debug_vector(0x0) <= python_monitor0;
	debug_vector(0x1) <= python_monitor1;
	debug_vector(0x2) <= grab_mngr_trig_rdy;
	debug_vector(0x3) <= curr_trig0;
	debug_vector(0x4) <= strobe;
	debug_vector(0x5) <= python_exposure;
	debug_vector(0x6) <= FOT;
	debug_vector(0x7) <= readout;
	debug_vector(0x8) <= readout_stateD;
	debug_vector(0x9) <= ext_trig;
	debug_vector(0xa) <= REGFILE.ACQ.GRAB_CTRL.GRAB_CMD;
	debug_vector(0xb) <= REGFILE.ACQ.GRAB_CTRL.GRAB_SS;
	debug_vector(0xc)<= grab_mngr_trig;
	debug_vector(0xd) <= grab_mngr_trig_rdy;
	debug_vector(0xe) <= grab_pending;
	debug_vector(0xf) <= grab_active;
	debug_vector(0x10) <= DEC_DATA_EN;
	debug_vector(0x11) <= DEC_SOL;
	debug_vector(0x12) <= DEC_SOF;
	debug_vector(0x13) <= DEC_EOL;
	debug_vector(0x14) <= DEC_EOF;
	debug_vector(0x15) <= DEC_CRC;
	debug_vector(0x16) <= DEC_TRAIN;
	debug_vector(0x17) <= fpnprnu_corr_sof;
	debug_vector(0x18) <= fpnprnu_corr_sol;
	debug_vector(0x19) <= fpnprnu_corr_data_val;
	debug_vector(0x1a) <= fpnprnu_corr_eol;
	debug_vector(0x1b) <= fpnprnu_corr_eof;
	debug_vector(0x1c) <= python_ssn_int;
	debug_vector(0x1d) <= debug_lvds(0);
	debug_vector(0x1e) <= debug_lvds(1);
	debug_vector(0x1f) <= 'Z';

Value at Reset:	0x1f

Debug1_sel (4:0)	

RW	debug_vector(0x0) <= python_monitor0;
	debug_vector(0x1) <= python_monitor1;
	debug_vector(0x2) <= grab_mngr_trig_rdy;
	debug_vector(0x3) <= curr_trig0;
	debug_vector(0x4) <= strobe;
	debug_vector(0x5) <= python_exposure;
	debug_vector(0x6) <= FOT;
	debug_vector(0x7) <= readout;
	debug_vector(0x8) <= readout_stateD;
	debug_vector(0x9) <= ext_trig;
	debug_vector(0xa) <= REGFILE.ACQ.GRAB_CTRL.GRAB_CMD;
	debug_vector(0xb) <= REGFILE.ACQ.GRAB_CTRL.GRAB_SS;
	debug_vector(0xc)<= grab_mngr_trig;
	debug_vector(0xd) <= grab_mngr_trig_rdy;
	debug_vector(0xe) <= grab_pending;
	debug_vector(0xf) <= grab_active;
	debug_vector(0x10) <= DEC_DATA_EN;
	debug_vector(0x11) <= DEC_SOL;
	debug_vector(0x12) <= DEC_SOF;
	debug_vector(0x13) <= DEC_EOL;
	debug_vector(0x14) <= DEC_EOF;
	debug_vector(0x15) <= DEC_CRC;
	debug_vector(0x16) <= DEC_TRAIN;
	debug_vector(0x17) <= fpnprnu_corr_sof;
	debug_vector(0x18) <= fpnprnu_corr_sol;
	debug_vector(0x19) <= fpnprnu_corr_data_val;
	debug_vector(0x1a) <= fpnprnu_corr_eol;
	debug_vector(0x1b) <= fpnprnu_corr_eof;
	debug_vector(0x1c) <= python_ssn_int;
	debug_vector(0x1d) <= debug_lvds(0);
	debug_vector(0x1e) <= debug_lvds(1);
	debug_vector(0x1f) <= 'Z';

Value at Reset:	0x1f

Debug0_sel (4:0)	

RW	debug_vector(0x0) <= python_monitor0;
	debug_vector(0x1) <= python_monitor1;
	debug_vector(0x2) <= grab_mngr_trig_rdy;
	debug_vector(0x3) <= curr_trig0;
	debug_vector(0x4) <= strobe;
	debug_vector(0x5) <= python_exposure;
	debug_vector(0x6) <= FOT;
	debug_vector(0x7) <= readout;
	debug_vector(0x8) <= readout_stateD;
	debug_vector(0x9) <= ext_trig;
	debug_vector(0xa) <= REGFILE.ACQ.GRAB_CTRL.GRAB_CMD;
	debug_vector(0xb) <= REGFILE.ACQ.GRAB_CTRL.GRAB_SS;
	debug_vector(0xc)<= grab_mngr_trig;
	debug_vector(0xd) <= grab_mngr_trig_rdy;
	debug_vector(0xe) <= grab_pending;
	debug_vector(0xf) <= grab_active;
	debug_vector(0x10) <= DEC_DATA_EN;
	debug_vector(0x11) <= DEC_SOL;
	debug_vector(0x12) <= DEC_SOF;
	debug_vector(0x13) <= DEC_EOL;
	debug_vector(0x14) <= DEC_EOF;
	debug_vector(0x15) <= DEC_CRC;
	debug_vector(0x16) <= DEC_TRAIN;
	debug_vector(0x17) <= fpnprnu_corr_sof;
	debug_vector(0x18) <= fpnprnu_corr_sol;
	debug_vector(0x19) <= fpnprnu_corr_data_val;
	debug_vector(0x1a) <= fpnprnu_corr_eol;
	debug_vector(0x1b) <= fpnprnu_corr_eof;
	debug_vector(0x1c) <= python_ssn_int;
	debug_vector(0x1d) <= debug_lvds(0);
	debug_vector(0x1e) <= debug_lvds(1);
	debug_vector(0x1f) <= 'Z';

Value at Reset:	0x1f

TRIGGER_MISSED

Address: section "ACQ" base address + 0x0E8

31	30	29	28	27	26	25	24
Reserved			TRIGGER_MISSED_RST	Reserved			
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
TRIGGER_MISSED_CNTR(15:8)							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
TRIGGER_MISSED_CNTR(7:0)							

TRIGGER_MISSED_RST	TRIGGER MISSED ReSeT	
WO/AutoClr	This is the trigger missed reset.	
Possible Values:	0x1	Reset the Trigger counter reset

TRIGGER_MISSED_CNTR (15:0)	TRIGGER MISSED CouNTeR	
RO	This is the number of trigger missed detected.	
Possible Values:	Any Value	

SENSOR_FPS

Address: section "ACQ" base address + 0x0F0

31	30	29	28	27	26	25	24
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
SENSOR_FPS(15:8)							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
SENSOR_FPS(7:0)							

SENSOR_FPS (15:0)	SENSOR Frame Per Second
RO	This is the number of frames received in 1 second interval. This register can count up to 64k frame/s. This counter counts on SO_FOT event.

DEBUG

Address: section "ACQ" base address + 0x1A0

31	30	29	28	27	26	25	24
Reserved			DEBUG_RST_CNTR	Reserved		TEST_MODE_PIX_START(9:8)	
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
TEST_MODE_PIX_START(7:0)							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Reserved						TEST_MOVE	TEST_MODE
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
LED_STAT_CLHS(1:0)		LED_STAT_CTRL(1:0)		Reserved	LED_TEST_COLOR(1:0)		LED_TEST

DEBUG_RST_CNTR		
RW	This register clears the debug cntrs	
Value at Reset:	0x1	
Possible Values:	0x0	
	0x1	Reset counters

TEST_MODE_PIX_START (9:0)	
RW	This register defines the value of the first pixel in the frame when the TEST_MODE is activated. In 8 bits mode only 8-MSB bits of the register is used.
Value at Reset:	0x0

TEST_MOVE		
RW	This field when in TEST_MODE=1, makes the ramp move. The first pixel of the frame is incremented by one each frame.	
Value at Reset:	0x0	
Possible Values:	0x0	Static test ramp
	0x1	The test ramp moves

TEST_MODE		
RW	This field set the FPGA in test mode. The fpga will send a programmable ramp to the host using the syncs receveived from the sensor. The generated ramp can move when set with the field TEST_MOVE. In color mode (LVDSx1), the ramp pixel is repeated 3 times to generate a B&W ramp in RGB24 mode.	
Value at Reset:	0x0	
Possible Values:	0x0	Normal acquisition data from sensor
	0x1	Test mode, a ramp is generated.

LED_STAT_CLHS (1:0)	
RO	

LED_STAT_CTRL (1:0)	
RO	

LED_TEST_COLOR (1:0)		
RW		
Value at Reset:	0x0	
Possible Values:	0x0	The LED is OFF
	0x1	The LED is GREEN
	0x2	The LED is RED
	0x3	The LED is ORANGE

LED_TEST		
RW	This register will put the LED status in test mode. The test mode is controlled by LED_TEST_COLOR	
Value at Reset:	0x0	
Possible Values:	0x0	The LED is in user mode.
	0x1	The LED is in test mode.

DEBUG_CNTR1

Address: section "ACQ" base address + 0x1A8

31	30	29	28	27	26	25	24
EOF_CNTR(31:24)							
23	22	21	20	19	18	17	16
EOF_CNTR(23:16)							
15	14	13	12	11	10	9	8
EOF_CNTR(15:8)							
7	6	5	4	3	2	1	0
EOF_CNTR(7:0)							

EOF_CNTR (31:0)	
RO	This is the EOF CNTR. This feature is enabled by setting register regfile.ACQ.DEBUG.DEBUG_RST_CNTR to 0.

DEBUG_CNTR2

Address: section "ACQ" base address + 0x1B0

31	30	29	28	27	26	25	24
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Reserved				EOL_CNTR(11:8)			
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
EOL_CNTR(7:0)							

EOL_CNTR (11:0)	
RO	This is the EOL CNTR. This feature is enabled by setting register regfile.ACQ.DEBUG.DEBUG_RST_CNTR to 0.

DEBUG_CNTR3

Address: section "ACQ" base address + 0x1B4

<table>
<tr><td>31</td><td>30</td><td>29</td><td>28</td><td>27</td><td>26</td><td>25</td><td>24</td></tr>
<tr><td colspan="4">Reserved</td><td colspan="4">SENSOR_FRAME_DURATION(27:24)</td></tr>
<tr><td>23</td><td>22</td><td>21</td><td>20</td><td>19</td><td>18</td><td>17</td><td>16</td></tr>
<tr><td colspan="8">SENSOR_FRAME_DURATION(23:16)</td></tr>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr>
<tr><td colspan="8">SENSOR_FRAME_DURATION(15:8)</td></tr>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="8">SENSOR_FRAME_DURATION(7:0)</td></tr>
</table>

<table>
<tr><td>SENSOR_FRAME_DURATION (27:0)
RO</td><td colspan="2">This is the time between the last 2 EOF received(in sys clock domain). This register can count up to 4.29 seconds. It can be used to predict sensor framerate or to verify sync between 3D profiler heads.
This feature is enabled by setting register regfile.ACQ.DEBUG.DEBUG_RST_CNTR to 0.</td></tr>
<tr><td>Possible Values:</td><td>Any Value</td><td>Any 28 bits value</td></tr>
</table>

EXP_FOT

Address: section "ACQ" base address + 0x1B8

31	30	29	28	27	26	25	24
Reserved							
23	22	21	20	19	18	17	16
Reserved							EXP_FOT
15	14	13	12	11	10	9	8
Reserved				EXP_FOT_TIME(11:8)			
7	6	5	4	3	2	1	0
EXP_FOT_TIME(7:0)							

EXP_FOT	EXPosure during FOT	
RW	When set to '1' this register, the output exposure and strobe signals will take into account the exposure in the FOT of the frame. This timing must be programmed in register EXP_FOT_TIME. This timing must be calculated from the OnSemi setting files .	
Value at Reset:	0x1	
Possible Values:	0x0	Disable exposure during FOT in output exposure signal and Strobe
	0x1	Enable exposure during FOT in output exposure signal and Strobe

EXP_FOT_TIME (11:0)	EXPosure during FOT TIME
RW	This is the time of the exposure during the FOT. This timing must be calculated from the OnSemi setting files . From DCF v1.2, for all LVDS modes : P5000 & P2000 EXP_FOT=40.666us, program value 0x9ee P1300 & P500 & P300 EXP_FOT=27.333us, program value 0x6ac
Value at Reset:	0x9ee

ACQ_SFNC

Address: section "ACQ" base address + 0x1C0

31	30	29	28	27	26	25	24
Reserved							
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved							
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**
Reserved							
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved							RELOAD_GRAB_PARAMS

RELOAD_GRAB_PARAMS		
RW	This register is not used for the moment. It may be used in the future to reload the exposure time	
Value at Reset:	0x1	
Possible Values:	0x0	
	0x1	