How to integrate and configure the VD56G3, VD66GY, and VD16GZ devices from a hardware and software perspective

Introduction

This document is the user manual for the VD56G3, VD66GY, and the VD16GZ 1.5 megapixel global shutter image sensors. The VD56G3 is a monochrome sensor (1124 x 1364). The VD66GY sensor (1124 x 1364) is available in RGB pattern. The VD16GZ sensor (1124 x 1364) is available in RGB NIR pattern. This user manual should be read with the VD56G3, VD66GY, and VD16GZ datasheets. The aim of this document is to provide further technical details about the VD56G3, VD66GY, and VD16GZ sensors, and to give guidance on how to configure them for different applications.

It is intended that this document primarily covers items not described in the datasheets. Therefore, to avoid repetition, references are made to the VD56G3, VD66GY, and the VD16GZ datasheets. Note that throughout the document we refer to "the device" or "the sensor" in the singular.

In some specific cases we differentiate the VDxxGx from the VDxxGxCCA, which is the fast boot version (no need to add firmware patch).

1 Acronyms and abbreviations

Table 1. Acronyms and abbreviations

Acronym/abbreviation	Definition
ADC	analog-to-digital converter
AE	auto exposure
AEC	automatic exposure control
BRIEF	binary robust independent elementary features
CC	central correction
CCI	camera control interface
CSI	camera serial interface
ECC	error correcting code
EV	exposure value
FAST	feature from accelerated segment test
FW	firmware
FSM	firmware state machine
GPH	group parameter hold
GPIO	general-purpose input/output
I²C	inter-integrated circuit
ISL	intelligent status line
ISP	image signal processor
LSB	least significant byte
MCU	microcontroller unit
MIPI	mobile industry processor interface
MSB	most significant byte
NIR	near infrared
NVM	nonvolatile memory
PCB	printed circuit board
PLL	phase-locked loop
PWM	pulse-width modulation
RC	ring correction
ROI	region of interest
SCL	serial clock line
SDA	serial data line
SW	software
UI	user interface
VT	video timing
VTRAM	video timing memory

2 Application schematic

Information on the application schematic and the layout recommendations are provided in the product datasheets. This section explains the integration of the image sensor into a system from a hardware perspective.

2.1 Power supplies

To power on the device, all the external supplies (VCORE, VDDIO, and VANA) must be properly provided according to the device characteristics described in the electrical characteristics section of the datasheets. Each supply should be in the operating range for a standard application configuration, and compatible with the device power consumption (average and peak current).

Table 2. Operating conditions

Symbol	Parameter	Min.	Typ.	Max.	Unit
VDDIO	Digital IO power supply	1.7	1.8	1.9	V
VANA	Analog power supply	2.65	2.8	2.95	
VCORE	Digital core power supply	1.08	1.15	1.26	
VIO	IO referenced to VDDIO	-0.3	—	VDDIO + 0.3	
Temperature					
T JF	Junction temperature (functional operation)	-30	—	85	°C

2.2 Hardware reset (XSHUTDOWN)

Reset of the device is managed by the XSHUTDOWN pin.

- At low level, the device is in power down state
- At high level, the device is active if all the power supplies are properly set up, and if the CLKIN input clock is present.

See the device datasheets for the electrical specifications of the XSHUTDOWN pin.

2.3 Input clock

To operate correctly, the device needs to be provided with an input clock on the CLKIN pin. Refer to the electrical characteristic section of the datasheets for the constraints of the input clock.

2.4 I²C interface

2.4.1 CCI protocol

The device is controlled via an I²C interface as defined by the MIPI CSI-2 specification (refer to *MIPI Alliance standard for camera serial interface 2).* The device uses CCI protocol over I²C. Details of this interface can be found in the device datasheets.

The value of the pull-up resistors, on the SDA and SCL signals, must be chosen to ensure that the I²C bus constraints are met in fast mode+. If the device has to be used in fast mode, the correct pull-up resistors must be selected to support fast mode and fast mode+. The default configuration of the device is fast mode+. The switch to fast mode is only available in SW_STANDBY state.

2.4.2 Data aligned within registers

Registers larger than 8 bits wide are stored LSB (least significant byte) first, i.e. the LSB is placed in the location with the lowest index (little endian).

Note: This ordering of multibyte registers is contrary to the recommended byte ordering detailed in the MIPI CSI-2 specification which states that multibyte registers should be stored MSB (most significant byte) first.

2.5 MIPI CSI-2 transmitter interface

The device outputs frames through a MIPI CSI-2 with the following characteristics:

- One or two lanes can be connected to the receiver
- Data and clock polarity can be independently inverted
- Lanes can be physically remapped to simplify the PCB layout

These configurations are detailed in Section 16: Output interface.

2.6 GPIO interfaces

The VD56G3, VD66GY, and VD16GZ have eight GPIOs which synchronize signals or act as in/out general purpose pins.

3 Control interface

3.1 Register default values

The register default values depend on the device state. The default values after device reset may differ from the register default values observed once the firmware is in SW_STANDBY state.

Without specific notice, the default values listed in this document are default values observed in the device SW_STANDBY state after the BOOT command is complete.

3.2 Register groups

The registers are grouped together based on their functional purpose.

Table 3. Register group descriptions

Group	Base address	Access	Description
STATUS	0x0000	RO	Device status
COMMAND	0x0200	R/W	Commands to control the device
SENSOR SETTINGS	0x0220	R/W	General settings of the device
STATIC	0x0300	R/W	Static configuration for streaming
DYNAMIC	0x0400	R/W	Dynamic configuration for streaming
CONTEXT 0	0x044C	R/W	Used to control the dynamic settings of the sensor. The sensor can be configured to automatically switch between these settings from frame to frame.
CONTEXT 1	0x0494	R/W	
CONTEXT 2	0x04DC	R/W	
CONTEXT 3	0x0524	R/W	
NVM MIRROR	0x0640	R/W	Buffer for NVM operations
MISCELLANEOUS	0x0900	R/W	Additional configurations
FW PATCH	0x2000	R/W	Buffer to load a Firmware patch

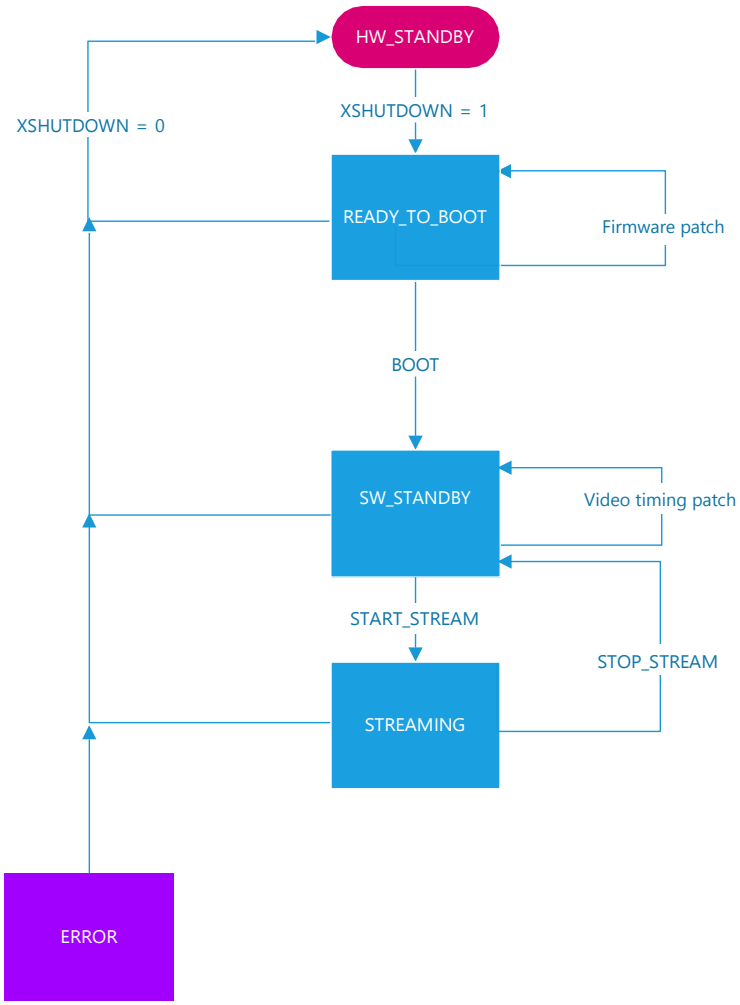
3.3 Register data types

Table 4. Data types description

Data type	Name	Description
UI	Unsigned integer	
SI	Signed integer	Two’s complement notation
C	Coded	This indicates that the value is coded to select one of several functions or modes
FP X.Y	Unsigned fixed point (FP)	Real value stored as unsigned, only for strictly positive value X.Y refers to the number of digits to store respectively the integer and decimal parts 3.75 <=> 0x03C0 in FP 8.8
SFP X.Y	Signed fixed point (SFP)	Real value stored as signed X.Y refers to the number of digits to store respectively the integer and decimal parts -3.75 <=> 0xFC40 in SFP 8.8
B	Bits	Each bit represents a specific function or mode (0 = disable, 1 = enable)

4 Firmware state machine

Figure 1. **Firmware state machine**



The SYSTEM_ FSM register in the STATUS group indicates the state of the device.

5 Steps from device power-up to sensor STREAMING state

This section describes the steps to be followed from device power-up to sensor STREAMING state. A power-up sequence diagram is provided in the product datasheets.

5.1 Device power-up sequence

To power on the device, all the external supplies (VCORE, VDDIO, VANA) must be properly provided according to the device characteristics described in the electrical characteristics section of the datasheets. XSHUTDOWN must stay low and the device should be in HW_STANDBY state.

5.2 HARDWARE_STANDBY to READY_TO_BOOT

The CLKIN signal must be provided before the XSHUTDOWN pin can be set high. See the product datasheets for device signal specifications. The time between the release of the XSHUTDOWN pin and READY-TO-BOOT state depends on the external clock frequency due to firewall activity. The I²C interface is not operational before reaching the READY-TO-BOOT state. If the device answers an I²C request, it means that the READY-TO-BOOT state has been reached.

5.3 READY_TO_BOOT

Once the device is in the READY-TO-BOOT state, the host can:

- Upload a firmware patch. The firmware patch upload procedure is described in Section 10: Firmware patch
- Configure the I²C interface regarding the device address and I²C protocol to be used for read and write operations (see Section 7: I²C interface configuration).

Once these operations are complete, the host must issue a BOOT command to go to SW_STANDBY state.

5.4 SW_STANDBY

From the SW_STANDBY state, the host can perform the following actions:

- Configure the clock and PLL using the UI (see Section 13: Timings).
- Perform all static, dynamic, and context configurations such as synchronization, ISP, output image, exposure, gain, and context chain.
- Perform a thermal sensor read (see Section 17: Temperature)
- Read/write into the NVM memory (see Section 12: NVM).
- Upload a video timing (VT) patch. The VT patch upload procedure is described in Section 11: VT patch

Once these operations have been completed, the host must issue a START_STREAMING command to go to STREAMING state.

5.5 STREAMING

In this state, the sensor streams frames according to the host configuration previously set in SW_STANDBY state.

Alternatively, from STREAMING state, the host can perform the following actions:

- Perform all “dynamic” configurations regarding image streaming through the DYNAMIC or CONTEXT register groups.
- Stop streaming by issuing a STOP_STREAM command. This makes the sensor go back to SW_STANDBY state.

6 Steps to power down device from sensor STREAMING state

This section describes the steps to be followed to power down the device starting from Sensor STREAMING state. The power down steps are summarized as follows:

1. The host issues a STOP_STREAM command. This stops the current streaming and the device goes back to SW_STANDBY state. The sensor finishes emitting its current frame. The exposure of the next frame then moves to SW_STANDBY state. This generates a delay between the STOP_STREAM command and the effective SW_STANDBY state variable. To confirm that the sensor has reached SW_STANDBY state, pool the register SYSTEM_FSM.
2. In SW_STANDBY state, the external clock signal, CLKIN, must be disabled and the XSHUTDOWN pin must be set to low. This sets the device to HW_STANDBY state.
3. Finally, the power supplies (VCORE, VDDIO, and VANA) should be driven down which turns the device off.

See the product datasheets for device signal specifications.

7 I²C interface configuration

The I²C default configuration is fast mode+ (which is compatible with fast mode).

The I²C control interface supports different IDs for read and write operations. By default, these addresses are either loaded from the NVM memory content, or written to a default value by the firmware if the NVM is empty. Note that the NVM memory content is configured during device manufacturing test operations.

Without specific access, the I²C address does not change during device operations. The default I²C address is 0x20 for write operations, and 0x21 for read operations.

Once the device is in SW_STANDBY state, the I²C address can be updated by the host controller. The update sequence is as follows:

1. Ensure that the device is in SW_STANDBY state
2. The host writes the new configuration into the DEVICE_ID bit fields of the DEVICE_I2C_CTRL register.
3. The host issues an I2C_ADDR_UPDATE command

The new address is effective for the next I²C access until HARDWARE_STANDBY or power down.

Table 5. I²C interface configuration

STATIC register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
DEVICE_I2C_CTRL	DEVICE_ID	7:1	UI	—
	MODE	8	C	0x0: FAST_MODE 0x1: FAST_MODE_PLUS

Note: The DEVICE_ID is a 7-bit field. Only give the seven MSB bits. Do not give any read/write bits.

8 Commands

The host sends commands through the UI by writing into different registers (see Table 6. COMMAND registers). Internally, an interrupt is raised to inform the firmware that a command has been issued by the host. Then, the command is processed in the FSM. An interrupt is raised to inform the firmware that a command has been issued, then the command is processed in the FSM. Once a command has been executed, the same register is cleared to indicate that the command has been acknowledged. If a command is issued before the previous command has been acknowledged, the new command is ignored. If the command is not acknowledged, such status is updated in the ERROR_CODE register (see Section 9: Error code).

Table 6. COMMAND registers

COMMAND register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
BOOT	COMMAND	1:0	C	0x1: BOOT 0x2: PATCH_SETUP
SW_STANDBY	COMMAND	3:0	C	0x1: START_STREAM 0x2: NVM_READ 0x3: NVM_PROG 0x4: THSENS_READ 0x5: I2C_ADDR_UPDATE
STREAMING	COMMAND	3:0	C	0x1: STOP_STREAM 0x2: VT_FSYNC_IN_I2C
VTPATCHING	COMMAND	3:0	C	0x1: START_VTRAM_UPDATE 0x2: END_VTRAM_UPDATE

9 Error code

If a command cannot be achieved or an error occurs during STREAMING state, the device goes into ERROR state. In this case, streaming is stopped and the ERROR_CODE register is different from 0x0000. The device must be reset.

Table 7. Error codes

Error code	Error
FW PATCH ERRORS	
0x400	CODE_TOO_LARGE
0x401	TOO_MANY_PATCHES
0x402	TOO_MANY_HOOKS
0x403	BAD_CRC
EXCEPTIONS ERRORS	
0x500	PROTECT
0x501	OPCODE
0x502	GPRSIZE
0x503	PMISALIGN
0x504	POUTOFMEM
0x505	PEXECUTE
0x506	DMISALIGN
0x507	DOUTOFMEM
0x508	DREAD
0x509	DWRITE
0x50A	PSYSERR
0x50B	OVERFLOW
0x50C	UNKNOWN
VTIMING ERRORS	
0xa00	LONG_COARSE_MAX_ERROR
0xa01	LONG_COARSE_MIN_ERROR
0xa02	BAD_FRAME_LENGTH_ERROR
0xa03	ISB_LONG_PIPE_OVERFLOW
0xa04	Y_SIZE_SS_ERROR
0xa05	X_SIZE_SS_ERROR
0xa06	BGISON_LOW
0xa07	BGISON_HIGH
0xa08	TOKEN_NOT_FOUND_ERROR
ISP ERRORS	
0xb00	SDR_FIFO_FULL_ERROR
0xb02	ISLGEN_INVALID_CFG_ERROR
0xb03	ISLGEN_MEMORY_LOCKED_ERROR
0xb04	ISLGEN_MISSED_TRIGGER_ERROR
0xb05	ISLGEN_TOO_MANY_ENTRIES_ERROR

Error code	Error
0xb06	MULTICROP_NO_ROI_ERROR
0xb07	ISB2IDP_LINEBLANKING_ERROR
OIF ERRORS	
0xc00	CSI_LANE_DESYNC_ERROR
0xc01	CSI_PKT_TOO_LONG_ERROR
0xc02	CSI_PKT_TOO_SHORT_ERROR
0xc03	CSI_UNDERFLOW_ERROR
0xc04	MERGER_EXT_SYNC_MISSED_ERROR

10 Firmware patch

A firmware patch must be loaded in the sensor during READY_TO_BOOT state. The size of the current patch is between 9000 and 10000 bytes, but this size may evolve over releases. The firmware patch is loaded using an I²C auto-increment method and the following procedure:

1. Load the binary patch at address 0x2000.
2. Issue the PATCH_SETUP command to activate the code of the patch.
3. The firmware patch version immediately becomes available in the FWPATCH_REVISION register.

The patch number is available in the STATUS register FWPATCH_REVISION.

The VDxxGxCCA does not need the FWPATCH to boot and stream properly.

Table 8. Firmware patch revision register

STATUS register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
FWPATCH_REVISION	MINOR_REVISION	7:0	UI	—
	MAJOR_REVISION	15:8	UI	—

Example (load FWPATCH and boot)

```
Fwpatch= loadbinary(“0227.bin”)// load FWPATCH

Start_add=0x2000 //FWPATCH

If Readbytes(0x0028,1)! =0: //check if FSM STATUS is at READY_TO_BOOT
        Break.

For (i=Stard_add, i<len(Fwpatch), i++):
        Writebyte(i,Fwpatch[i]) //example write byte per byte but can also write in autoincre
ment without limit

Writebyte(0x0200,0x02) // Send command Patch setup to apply the patch
While Readbytes(0x0200,0x00)! =0: //Verify command is sent properly
        Pass

Writebyte(0x0200,0x01) // Send command Patch setup to apply the newpatch
While Readbytes(0x0200,0x00)! =0: //Verify command is sent properly
        Pass
```

11 VT patch

A VT (video timing) patch must be loaded in the sensor during the SW_STANDBY state. The size of the current VT patch is nearly 6000 bytes, but this size may evolve over releases. Note that not all the addresses are contiguous. The VT patch is loaded according to the following procedure:

1. Issue the START_VTRAM_UPDATE command
2. Load the VT patch
3. Issue the END_VTRAM_UPDATE command to activate the code of the VT patch
4. The VT patch ID immediately becomes available in the VTPATCH_ID register

The VDxxGxCCA does not need the VTPATCH to boot and stream properly.

STATUS register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
VTPATCH_ID	VALUE	63:0	UI	—

12 NVM

Several 32-bit words are available in the NVM for customer use. Below is the procedure to program one or more consecutive words. The sensor must be in SW_STANDBY.

1. Write the words in the NVM MIRROR register group (see Section 19.7: NVM MIRROR registers).
2. Set the NVM_START_ADDRESS (refer to the NVM operation address in the table below) and NVM_NB_OF_WORD.
3. Issue the NVM_PROG command (see Section 19.2: COMMAND registers).

An NVM_READ command can be executed to reload values from the NVM in the NVM MIRROR registers. This command also uses the values set in the NVM_START_ADDRESS and NVM_NB_OF_WORD.

The content of the NVM is automatically loaded at boot in the NVM mirror registers (see Section 19.7: NVM MIRROR registers).

The NVM customer area has a specific word: I2C_ADDRESS. This word controls the default I²C address at powerup. This feature is doubly protected as follows:

1. I²C address must be replicated in byte 0, byte 1 and byte 2
2. Security key 0xAA is set in byte 3 (MSB)

If the conditions are wrong, the I²C address is not updated.

The NVM is a one-time programmable memory.

Table 9. NVM registers

NVM register group					
Register name	**NVM operation address**	**Field name**	**Bits**	**Data type**	**Values**
RESERVED 0	0x0000	VALUE	31:0	UI	First reserved word
RESERVED 113	0x01C4	VALUE	31:0	UI	Last reserved word
I2C_ADDRESS	0x01C8	I2C_KEY	31:24	UI	Security key
		I2C_DEVICEID_2	22:16		I²C address Device I²C address
		I2C_DEVICEID_1	14:8		
		I2C_DEVICEID_0	6:0		
CUSTOMER_WORD_0	0x01CC	VALUE	31:0	UI	First word available for customer
CUSTOMER_WORD_1	0x01D0	VALUE	31:0	UI	—
CUSTOMER_WORD_2	0x01D4	VALUE	31:0	UI	—
CUSTOMER_WORD_3	0x01D8	VALUE	31:0	UI	—
CUSTOMER_WORD_4	0x01DC	VALUE	31:0	UI	—
CUSTOMER_WORD_5	0x01E0	VALUE	31:0	UI	—
CUSTOMER_WORD_6	0x01E4	VALUE	31:0	UI	—
CUSTOMER_WORD_7	0x01E8	VALUE	31:0	UI	—
CUSTOMER_WORD_8	0x01EC	VALUE	31:0	UI	—
CUSTOMER_WORD_9	0x01F0	VALUE	31:0	UI	—
CUSTOMER_WORD_10	0x01F4	VALUE	31:0	UI	—
CUSTOMER_WORD_11	0x01F8	VALUE	31:0	UI	Last word available for customer

The NVM STATUS register can be read to check if the write operation went well (refer to table below). Each NVM word is individually protected by the ECC. Therefore, only a single write operation is allowed for each NVM word.

Table 10. **NVM STATUS registers**

STATUS register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
NVM	PROG_OK	0	B	0: False 1: True
	PROG_FAIL	1	B	0: False 1: True
	READ_OK	2	B	0: False 1: True
	READ_FAIL	3	B	0: False 1: True
	ECC_ERROR	5	B	0: NO_ERROR 1: ECC_ERROR
	DRIVER_FSM	7:6	C	0x0: IDLE 0x1: PROG_OP 0x2: READ_OP 0x3: ERROR

Example (NVM write/read)

Write

```
NVM_data=0x00000055 // data to add in NVM
Writebytes(0x080C, NVM_data)// NVM_mirror CUSTOMER_WORD_0

Writebytes(0x22a,0x01CC) //Set the NVM_START_ADDRESS (CUSTOMER_WORD_0)
Writebyte(0x229,0x01) // Set NVM_NB_OF_WORD number of word of 4 bytes
Writebyte(0X201, 0x03) // Update NVM
While Readbytes(0x0201,0x00)! =0: //Verify command is sent properly
        Pass
Readbytes(0x02a,1)//check if upload status ok: 0x01 Fail: 0x02
```

Write new I²C address

```
NVM_data=0xAA151515 // 0xAA: Security key 0x15: New I2C address
Writebytes(0x0808, NVM_data)// set NVM_mirror I2C_ADDRESS

Writebytes(0x22a, 0x01C8) //Set the NVM_START_ADDRESS (I2C_ADDRESS)
Writebyte(0x229,0x01) // Set NVM_NB_OF_WORD number of word of 4 bytes
Writebyte(0X201, 0x03) // Update NVM
While Readbytes(0x0201,0x00)! =0: //Verify command is sent properly
        Pass
Readbytes(0x02a,1)//check if upload status ok: 0x01 Fail: 0x02
```

Read

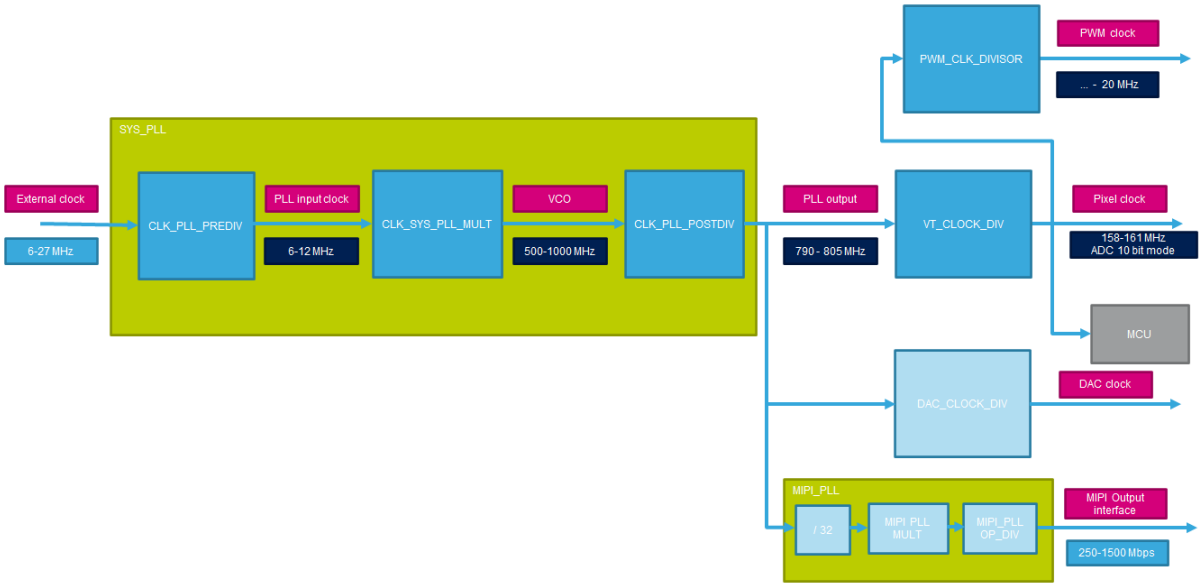
```
Writebytes(0x22a,0x01CC) //Set the NVM_START_ADDRESS (CUSTOMER_WORD_0)
Writebyte(0x229,0x01) // Set NVM_NB_OF_WORD number of word of 4 bytes
Writebyte(0X201, 0x02) // Update NVM mirror register with NVM data
While Readbytes(0x0201,0x00)! =0: //Verify command is sent properly
        Pass
Readbytes(0x02a,1)//check if update status ok: 0x04 Fail: 0x08

Readbytes(0x080C,4) // Read new value of CUSTOMER_WORD_0
```

13 Timings

13.1 Clocks and PLL configuration

Figure 2. Clock tree



The device supports a range of external clock source frequencies from 6 MHz to 27 MHz. It has been designed to run with a pixel clock at 160.8 MHz in ADC 10-bits mode and 201 MHz in ADC 9-bits mode.

The external clock, PLL output, MIPI frequency, and the pixel clock must be configured in SW_STANDBY mode only. To do so, the registers in the table below must be set.

Table 11. Clocks and PLL settings registers

SENSOR SETTINGS register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
EXT_CLOCK	VALUE	31:0	UI	External clock frequency in Hz
CLK_PLL_PREDIV	VALUE	2:0	UI	Pre-PLL clock divider value
CLK_PLL_POSTDIV	VALUE	7:0	UI	Post PLL clock divider value
CLK_SYS_PLL_MULT	VALUE	7:0	UI	PLL multiplier value
VT_CLK_DIV	VALUE	7:0	UI	VT clock divider

These registers must be set according to specific rules:

1. EXT_CLOCK in [6000000, 27000000] (6 MHz, 27 MHz)
2. $\frac{EXT\_CLK}{CLK\_PLL\_PREDIV}$ in [6000000, 12000000] (6 MHz, 12 MHz)
3. CLK_PLL_PREDIV is a discrete set of values [1,2,4]
4. $\frac{EXT\_CLK}{CLK\_PLL\_PREDIV} \times CLK\_SYS\_PLL\_MULT$ in [500000000, 1000000000] (500 MHz, 1 GHz)
5. CLK_SYS_PLL_MULT is in the range 37 to 167
6. $\frac{EXT\_CLK \times CLK\_SYS\_PLL\_MULT}{CLK\_PLL\_PREDIV \times CLK\_PLL\_POSTDIV}$ in [790000000, 805000000] (790 MHz, 805 MHz)
7. $PLL\ output\ clock\ = \frac{EXT\_CLK \times CLK\_SYS\_PLL\_MULT}{CLK\_PLL\_PREDIV \times CLK\_PLL\_POSTDIV}$ must be as close as possible to 804 MHz.

8. $Pixel\ clock = \frac{PLL\ output\ clock}{VT\_CLK\_DIV}$ must be as close as possible to 160.8 MHz in ADC 10-bits mode and to 201 MHz in ADC 9-bits mode.

To check that the correct clocks and PLL settings have been applied by the firmware, check the STATUS registers in the table below during STREAMING state.

Table 12. Clocks and PLL status registers

STATUS register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
EXT_CLOCK	VALUE	31:0	UI	External clock frequency in Hz
SYSTEM_PLL_CLK	VALUE	31:0	UI	PLL output frequency in Hz
PIXEL_CLK	VALUE	31:0	UI	Pixel clock frequency in Hz
MCU_CLK	VALUE	31:0	UI	MCU clock frequency in Hz
CLK_SYS_PLL_PREDIV	VALUE	2:0	UI	PLL predivider value
CLK_SYS_PLL_MULT	VALUE	7:0	UI	PLL multiplier value
CLK_SYS_PLL_OPDIV	VALUE	7:0	UI	PLL output divider

Use the following table as a reference for clock and PLL configurations.

Table 13. Clocks and PLL configuration example

ADC	External clock (MHz)	6	12	24	25	27
Any	EXT_CLOCK	6000000	12000000	24000000	25000000	27000000
	CLK_PLL_PREDIV	1	2	4	4	4
	CLK_PLL_POSTDIV	1	1	1	1	1
	CLK_SYS_PLL_MULT	134	134	134	128	119
	PLL out clock (MHz)	804	804	804	800	803.25
10 bit	VT_CLK_DIV	5	5	5	5	5
	Pixel clock (MHz)	160.8	160.8	160.8	160	160.65
9 bit	VT_CLK_DIV	4	4	4	4	4
	Pixel clock (MHz)	201	201	201	200	200.81

13.2 Line time

Line time is controlled by the LINE_LENGTH register.

Table 14. Line length register

STATIC register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
LINE_LENGTH	VALUE	15:0	UI	Length of line (in pixels)

Line time is computed using the following equation:

$$Line\ time = \frac{LINE\_LENGTH}{pixel\ clock}$$

The following table provides the minimum LINE_LENGTH according to ADC mode (see Table 19. Video timing control register).

Table 15. Minimum line length

ADC mode	Line length	Line time
10 bits	1236	7.69 µs
9 bits	1360	6.77 µs

Note: For both ADC modes (9 bits mode and 10 bits mode), it is recommended to set the line length at the minimum value. Higher line lengths may decrease performance. Increasing line length reduces the maximum frame rate and pixel performance (due to a higher data retention time). This may allow interoperability with low frequency MIPI receivers.

13.3 Frame rate

Frame rate is controlled by the FRAME_LENGTH register.

Table 16. Frame rate register

CONTEXT register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
FRAME_LENGTH	BIT_RATE	15:0	UI	Length of frame (in lines)

The frame rate is computed as follows:

$$Frame_{rate} = \frac{1}{line_{time} * FRAME\_LENGTH}$$

The minimum FRAME_LENGTH is computed from the resolution of the VT output. The VT output is controlled by the Y_START/Y_END registers (see Section 14.1.1: ROIs) and subsampling factor (see Section 14.1.2: Subsampling and digital binning). Only subsampling has an impact on the minimum frame length.

The minimum FRAME_LENGTH equation must be set as follows:

$$Minimum\ FRAME\_LENGTH > \frac{Y\_END - Y\_START + 1}{subsampling factor} + FRAME\_LENGTH\_OFFSET$$

The FRAME_LENGTH_OFFSET is computed from the current line length and the minimum line length (see Section 13.2: Line time). The FRAME_LENGTH_OFFSET decreases when the line length has been increased. The default value is 69.

$$FRAME\_LENGTH\_OFFSET = 31 + READ\_OFFSET + roundUp\left(20 \times \frac{minimum\ \ line\ \ length}{current\ \ line\ \ length}\right)$$

Part of the frame length offset is linked to the time required by the FW to compute the current frame settings: the read offset. This value is controlled by the READ_OFFSET register (STATIC register group).

Table 17. READ_OFFSET register

STATIC register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
READ_OFFSET	VALUE	15:0	UI	Time to process image settings during interframe

The read offset is set according to the following equation:

$$Read\ offset = roundUp\left(18 \times \frac{minimum\ \ line\ \ length}{current\ \ line\ \ length}\right)$$

The default value of the registers is 18 (0x12).

13.4 Output interface clock

The device output interface clock is generated by a specific PLL block controlled by the firmware. The output interface lane speed is set using the OIF_CSI_BITRATE register. This value is set in megabits per seconds and must be in the range 250 Mbps to 1500 Mbps.

Table 18. Output interface clock register

STATIC register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
OIF_CSI_BITRATE	BIT_RATE	15:0	UI	Output bitrate in Mbps per lane

The output interface should be fast enough to output one line of data in a line time period. It must be set according to the following formula:

$$OIF\_CSI\_BITRATE > \frac{IMAGE\_WIDTH \times BITS\_PER\_PIXEL + 1000}{DATALANE\_NB \times LINE\_TIME}$$

- IMAGE_WIDTH is the image resolution output by the sensor (see Section 14.1.1: ROIs for more details).
- BITS_PER_PIXEL is respectively 8 for RAW8 and 10 for RAW10 (refer to Table 40. Output interface configuration registers).
- DATALANE_NB is either 1 or 2
- LINE_TIME is in microseconds and defined in Section 13.2: Line time

Note: Increasing line length reduces the maximum frame rate and pixel performance (due to a higher data retention time). This may allow interoperability with low frequency MIPI receivers.

13.5 Video timing control

The VT_CTRL register controls the synchronization mode (leader mode or follower mode) and ADC mode (normal or fast readout).

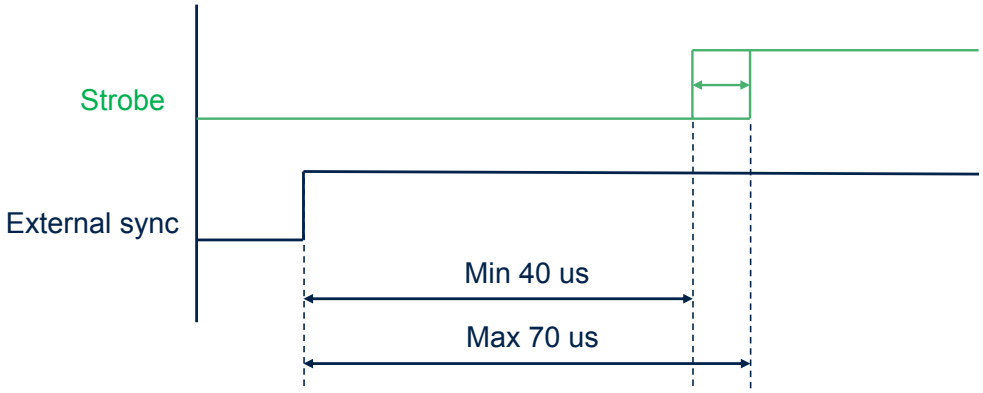
Table 19. Video timing control register

STATIC register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
VT_CTRL	SYNC_MODE	1:0	C	0x0: MASTER 0x1: Follower mode on GPIO 0x2: Follower mode on I²C 0x3: Leader mode synchronization on GPIO (only for VDxxGxCCA)
	ADC_MODE	2	B	0: 10 bits (normal) 1: 9 bits (fast readout)
	RESERVED	7:3	—	0x0

The SYNC_MODE field controls synchronization mode, and sets the device either in leader mode or follower mode. There are two follower modes: External pulse or I²C command.

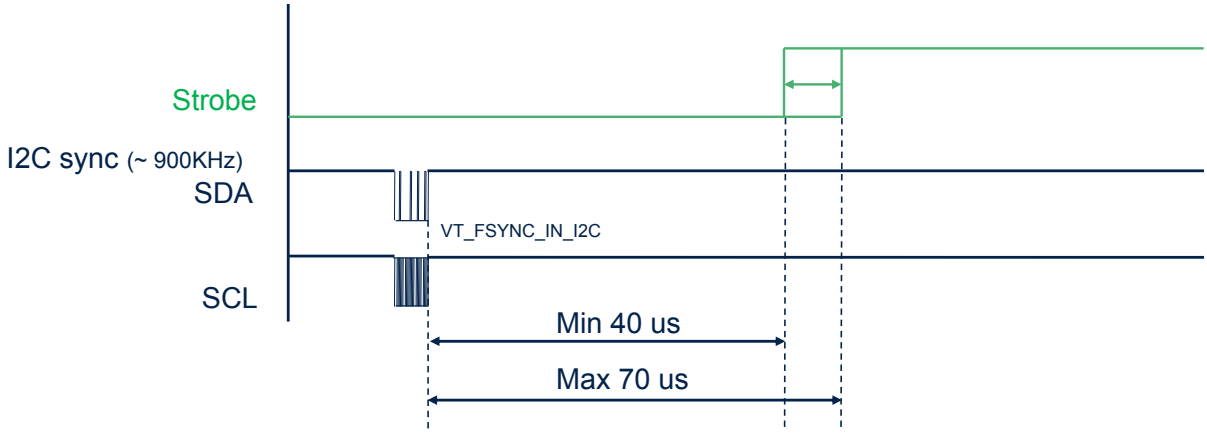
- MASTER
  - The device sends continuous frames
  - The FRAME_LENGTH register gives the frame rate
- EXTERNAL_SYNC (see Figure 3. EXTERNAL_SYNC timing)
  - The device starts to integrate a frame, then sends it each time the EXTERNAL_SYNC input pin is triggered.
  - Only GPIO_0 can be set as the EXTERNAL_SYNC pin
  - The maximum frame rate is defined by the FRAME_LENGTH register
- I2C_SYNC (see Figure 4. I2C_SYNC timing)
  - The device starts to integrate a frame, then sends it at each VT_FSYNC_IN_I2C command. Command is acknowledged once the frame is sent.
  - The maximum frame rate is defined by the FRAME_LENGTH register
- MASTER_EXTERNAL_SYNC (only for VDxxGxCCA)
  - The device starts to integrate a frame, when an EXTERNAL_SYNC is received after that the device sends continuous frames
  - Only GPIO_0 can be set as the EXTERNAL_SYNC pin (GPIO_IN mode)
  - The maximum frame rate is defined by the FRAME_LENGTH register

Figure 3. **EXTERNAL_SYNC timing**



This delay varies from one frame to another

Figure 4. **I2C_SYNC timing**



This delay varies from one frame to another

Figure 5. **MASTER_EXTERNAL_SYNC**

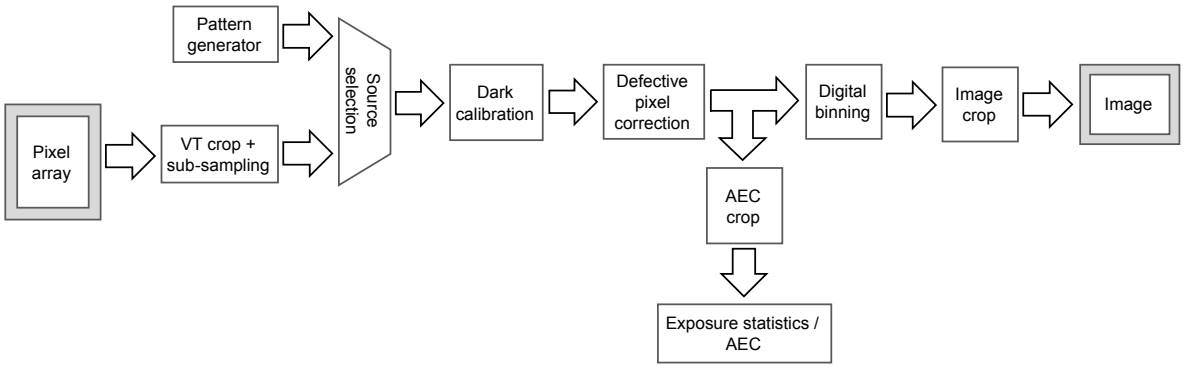
The ADC_MODE field controls the ADC readout mode:

- 10 bits is the standard readout
- 9 bits is the fast readout

Because of a shorter line time, the 9-bit mode enables a higher frame rate, which is compatible with RAW8 format.

14 Video pipe

Figure 6. **Video pipe**



14.1 Resolution and ROIs

The sensor has several features for configuring the resolution of the output image: crop, subsampling and digital binning.

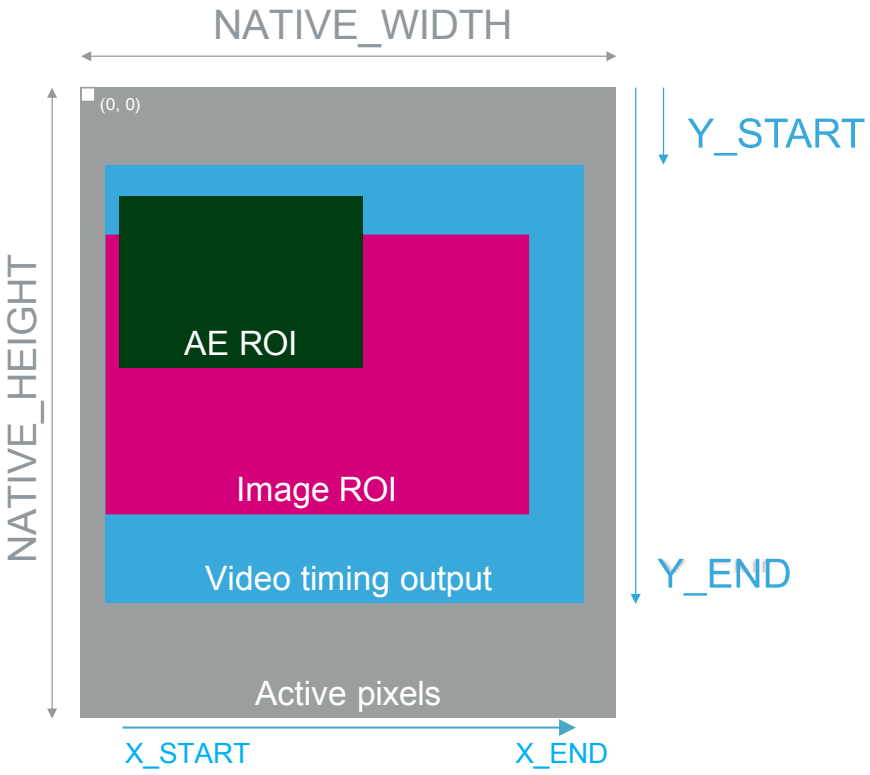
14.1.1 ROIs

There are three crop modules in the video processing pipeline:

- The first one crops the pixel array at the input of the ISP (VT crop)
- The second one crops the image at the output of the ISP (image crop). The image crop defines the image ROI.
- The third one crops at the input of the AEC module (AEC crop) for exposure statistics. The AEC crop defines the AEC ROI (minimum size 48x28).

The VT crop feeds the processing data path so the output of the VT crop is the input of the ISP. Consequently, both image crop and AEC crop are configured according to the VT crop output (see figure below).

Figure 7. Crops and ROIs



The VT crop is controlled by two registers: Y_START and Y_END. They are respectively the first row and the last row of the pixel array read by the VT. When the orientation is normal, the VT processes the rows from the top of the pixel array to the bottom. Consequently, the reference of these registers is the top first row.

Table 20. VT crop registers

CONTEXT register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
Y_START	VALUE	15:0	UI	First row read by the VT
Y_END	VALUE	15:0	UI	Last row read by the VT
X_START	VALUE	15:0	UI	First column read by the VT
X_END	VALUE	15:0	UI	Last column read by the VT

Two other crop modules in the video pipe enable the selection of two different ROIs: one for the image, another one for the AEC. Both ROIs are strictly independent.

When the orientation is normal, the reference pixel of both ROIs is the top left corner of the VT output (see Figure 8. Image ROI configuration and Figure 9. AEC ROI configuration).

VT crop and image crop are defined in the CONTEXT register groups whereas AEC ROI is defined in the DYNAMIC register group, meaning the AEC ROI is shared across 4 contexts.

Figure 8. Image ROI configuration

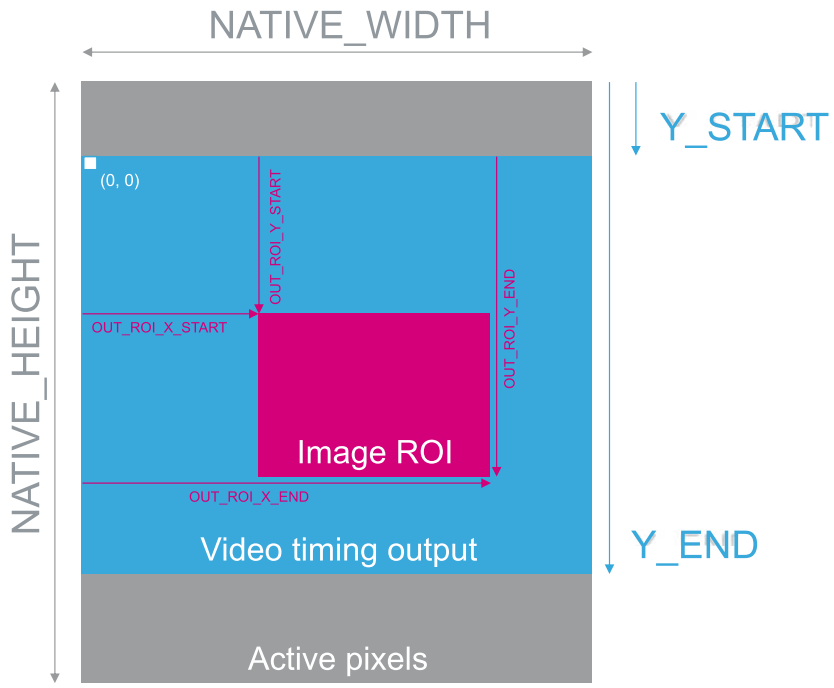


Table 21. Image crop registers

CONTEXT register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
OUT_ROI_X_START	VALUE	15:0	UI	Image ROI
OUT_ROI_X_END	VALUE	15:0	UI	
OUT_ROI_Y_START	VALUE	15:0	UI	
OUT_ROI_Y_END	VALUE	15:0	UI	

The image resolution must comply with MIPI guidelines:

- Both width (OUT_ROI_X_END - OUT_ROI_X_END + 1) and height (OUT_ROI_Y_END - OUT_ROI_Y_END + 1) must be even.
- On top of this, when the output format is RAW10, the width must be a multiple of 4 pixels.
- Additional guidelines apply when subsampling or digital binning is enabled (refer to Table 24. ROI configurations versus decimation factor).

Figure 9. AEC ROI configuration

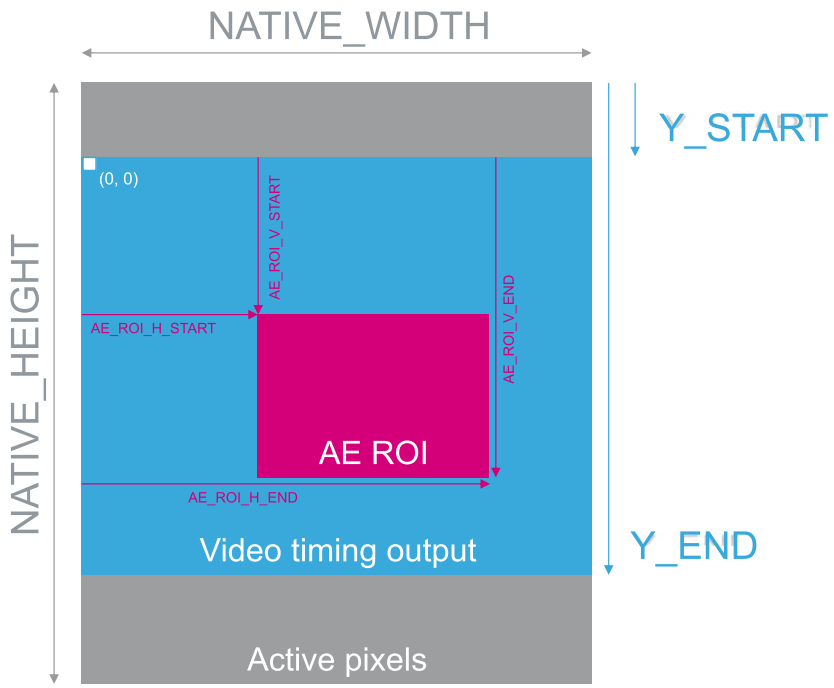


Table 22. AEC crop registers

DYNAMIC register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
AE_ROI_ START_H	VALUE	15:0	UI	AEC ROI
AE_ROI_ END_H	VALUE	15:0	UI	
AE_ROI_ START_V	VALUE	15:0	UI	
AE_ROI_END_V	VALUE	15:0	UI	

OUT_ROI_X_END and AE_ROI_END_H must be lower or equal to the native width (pixel array).
OUT_ROI_Y_END and AE_ROI_END_V must be lower or equal to the VT output width (Y_END – Y_START + 1).

14.1.2 Subsampling and digital binning

On top of the crop operations, the ISP has a subsampling and digital binning module. The available decimation factors are 2 and 4, nevertheless both modules cannot be enabled simultaneously. The decimation factor applies in X and Y directions.

Selection of the decimation method and factor is controlled by the READOUT_CTRL register. This register is part of the CONTEXT register group.

Table 23. Decimation control register

CONTEXT register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
READOUT_CTRL	CFG	2:0	C	0: Normal 1: Digital binning x2 2: Digital binning x4 3: Subsampling x2 4: Subsampling x4

Whatever the decimation factor, the configuration of the ROIs is always based on the normal resolution (x1). However, the decimation operation introduces constraints on ROI configuration. Indeed, some register values must be multiples of powers of two. Refer to table below to determine which is the multiple that applies for each register or couple of registers.

Table 24. ROI configurations versus decimation factor

		RAW8			RAW10		
	Decimation factor	**Normal**	**x2**	**x4**	**Normal**	**x2**	**x4**
Y_START	—	1	2	4	1	2	4
Y_END + 1	—	1	2	4	1	2	4
OUT_ROI_X_END - OUT_ROI_X_START + 1	—	2	4	8	4	8	16
OUT_ROI_Y_END - OUT_ROI_Y_START + 1	—	2	4	8	2	4	8

Refer to Table 40. Output interface configuration registers to select the output format (RAW8/RAW10)

14.1.3 Status registers of the image ROI

Confirmation of the image ROI is available in the STATUS register group. OUT_ROI_X_SIZE and OUT_ROI_Y_SIZE are the final resolution of the image including potentially the digital binning or the subsampling.

$$OUT\_ROI\_X\_SIZE = \frac{OUT\_ROI\_X\_END - OUT\_ROI\_X\_START + 1}{Decimation\ factor}$$

$$OUT\_ROI\_Y\_SIZE = \frac{OUT\_ROI\_Y\_END - OUT\_ROI\_Y\_START + 1}{Decimation\ factor}$$

Table 25. Image status registers

STATUS register group				
Register name	**Field name**	**Bits**	**Data type**	**values**
Y_START	VALUE	15:0	UI	First row read by the VT
Y_END	VALUE	15:0	UI	Last row read by the VT
OUT_ROI_X_START	VALUE	15:0	UI	Image ROI
OUT_ROI_X_END	VALUE	15:0	UI	
OUT_ROI_Y_START	VALUE	15:0	UI	
OUT_ROI_Y_END	VALUE	15:0	UI	
OUT_ROI_X_SIZE	VALUE	15:0	UI	Width of the image
OUT_ROI_Y_SIZE	VALUE	15:0	UI	Height of the image
READOUT_CTRL	CFG	2:0	C	Decimation

14.2 Group parameter hold

The host can prevent the firmware from applying some dynamic parameters (DYNAMIC and CONTEXT registers). This allows the sensor to be updated with a new configuration without timing constraints. To do this, the host controls the GROUP_PARAM_HOLD register (DYNAMIC register group). This register must be updated in STREAMING state.

Table 26. Group parameter hold (GPH) register

DYNAMIC register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
GROUP_PARAM_HOLD	HOLD	0	B	0: Hold disable 1: Hold enable

First, the hold must be enabled, then the host can update the following registers:

Table 27. Registers controlled by GPH

Feature	Register name	Register group
Dark calibration	DARKCAL_PEDESTAL	CONTEXT
Exposure	AE_COMPENSATION	DYNAMIC
	AE_TARGET_PERCENTAGE	
	AE_STEP_PROPORTION	
	AE_LEAK_PROPORTION	
	EXP_MODE	CONTEXT
	MANUAL_ANALOG_GAIN	
	MANUAL_COARSE_EXPOSURE	
	MANUAL_DIGITAL_GAIN	
Image ROI	OUT_ROI_X_START	CONTEXT
	OUT_ROI_X_END	
	OUT_ROI_Y_START	
	OUT_ROI_Y_END	
AEC ROI	AE_ROI_START_H	DYNAMIC
	AE_ROI_START_V	
	AE_ROI_END_H	
	AE_ROI_END_V	
Pattern generator	PATGEN_CTRL	DYNAMIC

Once the new configuration is set, the host disables the hold. Updated registers are then considered by the firmware.

Note: When GPH is enabled, automatic exposure control is frozen.

14.3 Orientation

Device orientation can be set using the Orientation register in the STATIC register group.

Table 28. Orientation register

STATIC register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
ORIENTATION	MODE	15:0	C	0x0: No flip 0x1: X flip 0x2: Y flip 0x3: XY flip

Figure 10. Device orientation

No flip X flip Y flip XY flip

14.4 Pattern generator

The device can generate patterns for debugging purposes. Active pixel values are overridden when the register PATGEN_CTRL is different than 0.

Table 29. Pattern generator register

DYNAMIC register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
PATGEN_CTRL	ENABLE	1:0	C	0x0: BYPASS 0x1: PATGEN
	TYPE	8:4	C	See table below

Table 30. Pattern configuration

Pattern	Value
Solid color	0x01
Vertical color bars	0x02
Horizontal gray scale	0x10
Vertical gray scale	0x11
Diagonal gray scale	0x12
Pseudo random (PN28)	0x13

14.5 Dark calibration

The device embeds an automatic dark calibration mechanism. The dark calibration target value can be set with the dynamic register DARKCAL_PEDESTAL. This is a 10-bit value and is set regardless of the output format (RAW8 or RAW10). However, when the device outputs an 8-bit image, the dark calibration mechanism uses the register value right-shifted by two bits to automatically align the pedestal value according to the output format dynamic.

For instance, with the default value 64 set in the register DARKCAL_PEDESTAL:

- The pedestal is 64 in RAW10
- The pedestal is 16 in RAW8

Table 31. Dark calibration register

STATIC register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
DARKCAL_PEDESTAL	VALUE	7:0	UI	[0:1023]

14.6 Defect correction

The VD56G3 and VD66GY sensors contain an algorithm which automatically corrects defective pixels. Defect correction shall not be used with VD16GZ. Singlet or couplet defective pixels are dynamically corrected on-the-fly by an auto-adaptive algorithm. It uses a 3x3 kernel of pixels from the same color plan that surrounds the pixel to be corrected.

The defect correction block is composed of two consecutive stages:

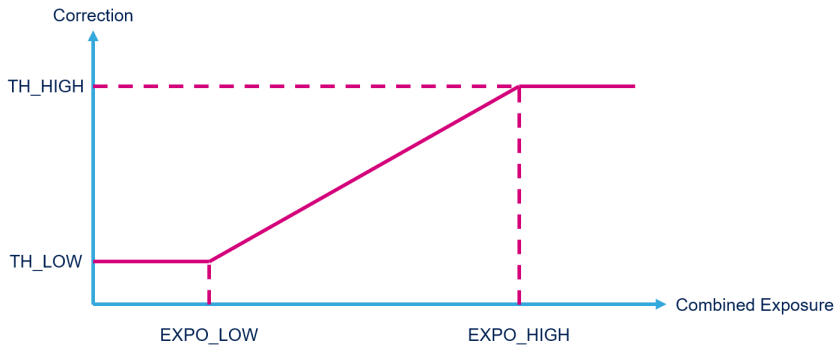
- Ring correction (RC), which can be done temporarily prior to correction of the central pixel.
- Central correction (CC)

Both RC and CC can be activated and tuned to optimize the impact on image quality. The strength of the correction (CC_SIGMA_Y and RC_TH_Y) can be programed relative to the exposure time (CC_SIGMA_X and RC_TH_X). This is achieved by a damper mechanism (the lower the Y value, the stronger the correction). The usual range is 16 down to 4.

Table 32. Defect correction registers

STATIC register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
DUSTER_CTRL	ENABLE	0	C	0x0: DISABLE 0x1: ENABLE
	DYN_ENABLE	2:1	C	0x0: DISABLE 0x1: ENABLE_SINGLET_COUPLET 0x2: ENABLE_ONLY_COUPLET
	MAP_ENABLE	3	C	0x0: DISABLE 0x1: ENABLE
	RING_ENABLE	4	C	0x0: DISABLE 0x1: ENABLE
DUSTER_DMP_CC_SIGMA_X	EXPO_LOW	15:0	UI	—
	EXPO_HIGH	31:16	UI	—
DUSTER_DMP_CC_SIGMA_Y	TH_LOW	15:0	UI	—
	TH_HIGH	31:16	UI	—
DUSTER_DMP_RC_TH_X	EXPO_LOW	15:0	UI	—
	EXPO_HIGH	31:16	UI	—
DUSTER_DMP_RC_TH_Y	TH_LOW	15:0	UI	—
	TH_HIGH	31:16	UI	—

Figure 11. Damper curve



*Note: The parameters EXPO_LOW and EXPO_HIGH are the product exposure_time (in µs) *AGAIN. This information concerns 16 bit registers only.*

14.7 Exposure

The exposure settings are either automatically controlled by the firmware (AEC), or directly controlled by the host (MANUAL). A third mode is to freeze the current settings of the AEC algorithm. Whatever the mode, analog and digital gains are respectively limited to 4 and 8 by default.

14.7.1 Exposure mode control

Exposure mode is controlled by the EXP_MODE register in the CONTEXT register group. Each context has its own exposure mode.

Table 33. Exposure control register

CONTEXT register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
EXP_MODE	MODE	1:0	C	0x0: Automatic mode 0x1: Freeze AE with current settings 0x2: Manual setting mode

14.7.2 Manual exposure

When manual setting mode is selected, the user takes full control of the exposure settings: integration time and analog and digital gains. Each context has its own manual exposure settings.

- Analog gain is controlled by the MANUAL_ANALOG_GAIN register. The gain is computed with the following formula: $analog\ gain = \frac{32}{32 - MANUAL\_ANALOG\_GAIN}$. The value of the register is in the range [0:28] that is, a gain from 1 to 8.
- Integration time is controlled by the MANUAL_COARSE_EXPOSURE register. The value is expressed in line equivalent.
- Digital gain is controlled by the MANUAL_DIGITAL_GAIN register. This register is coded as a Fixed Point 5.8 and the value is in the range [1.00, 8.00].

Refer to Section 14.7.5: Exposure settings range to set the limits of each parameter.

Table 34. Manual exposure setting registers

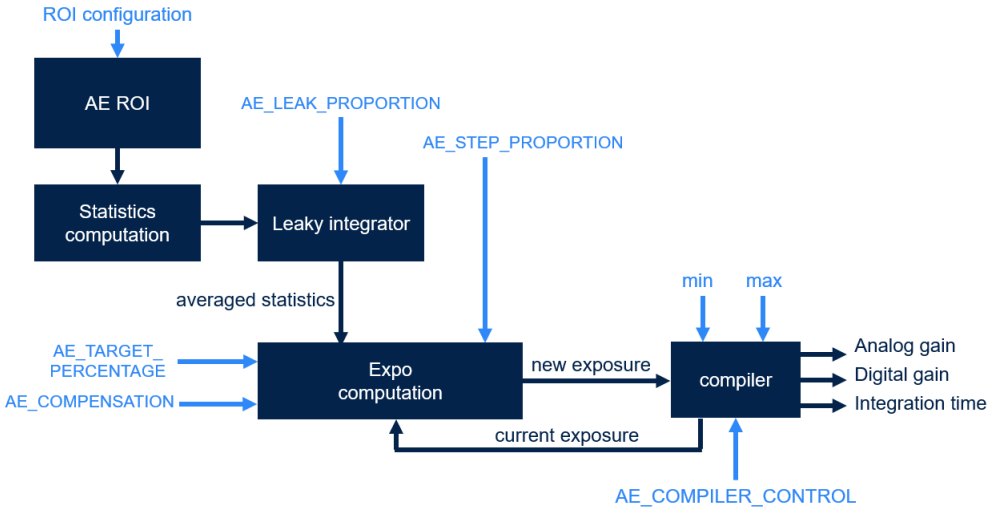
CONTEXT register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
MANUAL_ANALOG_GAIN	VALUE	4:0	C	Analog gain to be set
MANUAL_COARSE_EXPOSURE	VALUE	15:0	UI	Integration time in line to be set
MANUAL_DIGITAL_GAIN	VALUE	12:0	FP 5.8	Digital gain to be set

For the VD56G3 and VD16GZ models, controlling the MANUAL_DIGITAL_GAIN channel is enough to control the digital gain.

For the VD66GY model (RGB Bayer version), all four channels have their own digital gain control. It needs to be configured.

14.7.3 Automatic exposure

Figure 12. **AEC overview**



In automatic mode, the AEC targets a level of luminance as a percent of the saturation level (that is, 100 means saturated image). The exposure can also be adjusted through the compensation factor (expressed in EV) added on top of the target. From the image statistics and the current exposure, the AEC updates gains and the integration time on a frame basis.

- The target is set by the AE_TARGET_PERCENTAGE register
- The compensation factor is controlled by the AE_COMPENSATION register

Image statistics (luminance evaluation) are computed over a configurable area (ROI). Information regarding the configuration of the AE ROI is available in Section 14.1.1: ROIs.

Two AEC modes are available. The mode is selected by bit[0] of the AE_COMPILER_CONTROL register.

- 0 sets the priority to minimum gains. The AEC minimizes both digital and analog gains, then adjusts the integration time.
- 1 sets the priority to flicker free exposure. The AEC prioritizes the integration time in multiples of half the flicker period.

Two frequencies are available for flicker free mode (see above). The frequency is selected by bit[1] of the AE_COMPILER_CONTROL register:

- 0 is 50 Hz, the AEC tries to set the integration time to a multiple of 10 ms
- 1 is 60 Hz, the AEC tries to set the integration time to a multiple of 8.33 ms
- For lower integration times, the flicker free condition is not respected

The main solution to making the sensor insensitive to LED flicker is to adjust the frame length to a multiple of half the flicker period.

Two registers control the convergence loop of the AEC:

- AE_LEAK_PROPORTION register controls reactivity. It is the leak of a leaky integrator fed by the image statistics. The leaky integrator averages the image statistics over time and smoothes the luminance transition.
  - When AE_LEAK_PROPORTION = 0.0, statistics are frozen at the previous value
  - When AE_LEAK_PROPORTION = 1.0, statistics are forced to the current frame statistic
- AE_STEP_PROPORTION register controls the speed. It is the damping factor for the exposure variation.
  - When AE_STEP_PROPORTION = 0.0, the exposure is frozen
  - When AE_STEP_PROPORTION = 0.5, operation is at half speed
  - When AE_STEP_PROPORTION = 1.0, operation is at full speed (not recommended)

Table 35. **Automatic exposure mode registers**

DYNAMIC register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
AE_COMPILER_CONTROL	MODE	0	C	0: Priotity to smallest gains 1: Priority to flicker free
	FLICKER_FREQUENCY	1	C	0: 50 Hz 1: 60 Hz
AE_COMPENSATION	VALUE	15:0	SFP 8.8	Compensation in EV
AE_TARGET_PERCENTAGE	VALUE	15:0	UI	Percentage relative to saturation
AE_STEP_PROPORTION	VALUE	15:0	FP 8.8	Damping factor of the exposure
AE_LEAK_PROPORTION	VALUE	15:0	FP 1.15	Leaky integrator of the statistics

14.7.4 AEC cold start

It is possible to force the initial values of the AEC output. This feature is controlled by bit[0] of the AE_FORCE_COLDSTART register. If the feature is enabled, the AEC applies the following exposure settings at the next start STREAMING command:

- AE_COLDSTART_COARSE_EXPOSURE register
- AE_COLDSTART_ANALOG_GAIN register
- AE_COLDSTART_DIGITAL_GAIN register

If this feature is disabled, the AEC starts with the current values at the next START STREAMING command.

Table 36. **AEC cold start registers**

DYNAMIC register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
AE_FORCE_COLDSTART	VALUE	0	B	0: Disable 1: Enable
AE_COLDSTART_COARSE_EXPOSURE	VALUE	15:0	UI	See manual exposure mode for configuration
AE_COLDSTART_ANALOG_GAIN	VALUE	4:0	UI	
AE_COLDSTART_DIGITAL_GAIN	VALUE	12:0	FP 5.8	

14.7.5 Exposure settings range

The ranges of some exposure settings are set with default values. For instance, the analog and digital gains and the coarse exposure. However, it is possible to modify these default range values. To do so, the following registers must be set in SW_STANDBY state.

Table 37. Exposure settings range

Miscellaneous register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
EXP_COARSE_INTG_MARGIN	VALUE	15:0	UI	Limit maximum exposure coarse value in lines
MIN_DG	VALUE	FP 5.8	UI	Minimum digital gain
MAX_DG	VALUE	FP 5.8	UI	Maximum digital gain
MIN_AG_CODED	VALUE	4:0	UI	Minimum analog gain = 32/(32 - MIN_AG_CODED)
MAX_AG_CODED	VALUE	4:0	UI	Maximum analog gain = 32/(32 - MAX_AG_CODED)

Maximum coarse exposure is set using EXP_COARSE_INTG_MARGIN register, where:

$$MAX\_EXPOSURE\_COARSE = FRAME\_LENGTH - EXP\_COARSE\_INTG\_MARGIN - 7$$
$$EXP\_COARSE\_INTG\_MARGIN \geq 68$$

Analog gain range is set using MIN_AG_CODED and MAX_AG_CODED where:

$$MIN\_AG\_CODED \leq MAX\_AG\_CODED$$

The digital gain range is set using MIN_DG and MAX_DG where:

$$MIN\_DG \leq MAX\_DG$$

These ranges are applied in both auto and manual exposure modes.

The registers MAX and MIN_AG_CODED and MAX and MIN_DG limit AGAIN and DGAIN excursions. The AEC automatically respects and keeps parameters inside these declared limits. In manual exposure mode, the sensor firmware clips the real, applied values to the allowed limits if the parameters configured are above the allowed limits. For good operation, the MAX_AG_CODED must always be equal to or less than 28. MAX_DGAIN must always be equal to or less than 16.0.

Table 38. Gain and exposure range

Parameter	Min.	Max.
AGAIN	0.0	8.0
DGAIN	1.0	16.0
MANUAL_COARSE_EXPOSURE ADC10	21 (1)	—
MANUAL_COARSE_EXPOSURE ADC10 for VDxxGxCCA	5 (2)	—
MANUAL_COARSE_EXPOSURE ADC9	24 (1)	—
MANUAL_COARSE_EXPOSURE ADC9 for VDxxGxCCA	5 (2)	—

1. *The minimum MANUAL_COARSE_EXPOSURE is 161 µs.*
2. *Under 21 lines, an extra delay is added once. Over 40 lines, an extra delay is added once.*

14.7.6 Exposure settings status

Whatever the exposure mode, the settings currently applied are available in the STATUS registers group.

- Exposure control mode is available in EXPOSURE_MODE register
- Coarse exposure time is available in APPLIED_COARSE_EXPOSURE register. The value is expressed in line equivalent.
- Analog gain is available in the APPLIED_ANALOG_GAIN register
- Digital gain is available in the APPLIED_DIGITAL_GAIN register
- Status of the AEC is available in the AE_STATUS register

15 Context management

Four contexts are available to change the settings during STREAMING. The host configures the sequence to switch from one context to another one. The sensor sends CONTEXT_REPEAT_COUNT frames, then switches to the context number stored in NEXT_CONTEXT.

The first frame output by the sensor is based on context 0 parameters. If CONTEXT_REPEAT_COUNT is set to 0, the sensor remains on the current context.

Warning: *The current context parameters (CONTEXT_REPEAT_COUNT and NEXT_CONTEXT) must not be changed during STREAMING.*

The following values are available in the STATUS register group (also see table below).

- FRAME_COUNTER, 16-bit counter, the value goes from 65535 to 1 at rollover
- CONTEXT_FRAME_COUNTER
- CONTEXT_REPEAT_COUNT
- CURRENT_CONTEXT
- NEXT_CONTEXT

Table 39. Context management registers

CONTEXT register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
CONTEXT_REPEAT_COUNT	VALUE	15:0	UI	0 to 65535
NEXT_CONTEXT	VALUE	15:0	C	0, 1, 2, 3

When finishing a context sequence, WAIT_DELAY introduces an additional frame blanking before running the next context. There are two solutions to run infinitely in the same context:

- Set CONTEXT_REPEAT_COUNT = 0 (pre-eminent to any NEXT_CONTEXT value).
- Set CONTEXT_REPEAT_COUNT = N and configure the NEXT_CONTEXT to itself.

In the second solution, a WAIT_DELAY is applied every N frame. In the first solution, a WAIT_DELAY never occurs.

Warning: *The first two frames after start streaming are always context 0, regardless of the context switching sequence.*

16 Output interface

The output interface of the device embeds a two-data lane MIPI DPHY interface. It supports up to 1.5 Gb/s of data rate per lane.

Before streaming, when the sensor is in SW_STANDBY state, set the OIF_CSI_BIT_RATE. The sensor then computes the closest possible MIPI frequency based on its internal constraints. The actual MIPI frequency may differ slightly from the expected MIPI frequency. The actual MIPI frequency can be calculated using the following formula:

$$MIPIfrequency = SYSTEM\_PLL\_CLK \times CLK\_MIPI\_PLL\_MULT / CLK\_MIPI\_PLL\_OPDIV / 32$$

Data are transferred through the CSI2 as follows:

- Status lines coded on RAW8 SMIA format
- RAW10 or RAW8 image data

By default, the MIPI clock is continuous, meaning that the clock lanes are always in high-speed state. It is possible to disable this continuous mode, by setting the register below in SW_STANDBY. In non-continuous mode, the MIPI clock lanes are set in Low power state during the interframe.

Miscellaneous registers					
Register name	**Field name**	**Bits**	**Data type**	**Default**	**Values**
DPHYTX_CTRL	VALUE	[7:0]	CODE	0x1C	0x0C: Disable continuous mode 0x1C: Enable continuous mode

16.1 Configuration

All data (active image and ISL) can be output on different virtual channels and/or data types by using the OIF_VC_CTRL and OIF_XXX_CTRL registers.

Configuration of the output interface is via registers in the STATIC register group (see table below).

Table 40. Output interface configuration registers

STATIC register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
FORMAT_CTRL	OUT_FORMAT	4:0	C	0x8: RAW8 0xa: RAW10
OIF_CTRL	DATALANE_NB	2:0	C	0x1: SINGLE_LANE 0x2: DUAL_LANE
	CLKLANE_SWAP	3	C	0x0: NO_SWAP 0x1: LANE_SWAP
	DATALANE0_MAPPING	5:4	C	0x0: LANE_0 0x1: LANE_1
	DATALANE0_SWAP	6	C	0x0: NO_SWAP 0x1: LANE_SWAP
	DATALANE1_MAPPING	8:7	C	0x0: LANE_0 0x1: LANE_1
	DATALANE1_SWAP	9	C	0x0: NO_SWAP 0x1: LANE_SWAP
OIF_VC_CTRL	ACTIVE_PIX	1:0	UI	Virtual channel of the image
	ISL	3:2	UI	Virtual channel of the embedded status lines
OIF_IMG_CTRL	DATA_TYPE	5:0	UI	Data type of the image
OIF_ISL_CTRL	DATA_TYPE	5:0	UI	Data type of the embedded status lines
ISL_ENABLE	VALUE	0	B	0: Disable status lines 1: Enable status lines
OUTPUT_CTRL	CONTROL	1:0	C	0x1: Image output

16.2 Frame format

Image

The frame is made of two status lines (each line is the same size) followed by the image content.

Figure 13. Standard image content

16.3 Status lines

Status lines are composed of two lines. The line width is the same as the active image output in bytes, with a minimum of 256 bytes. The lines contain status register values for the current frame. Data are coded in SMIA RAW8 format. Content is defined in Section 19.1: STATUS registers.

The default data type of the embedded status lines is 0x12. The default virtual channel of the embedded status lines is 0x0.

Figure 14. Structure of the ISL lines

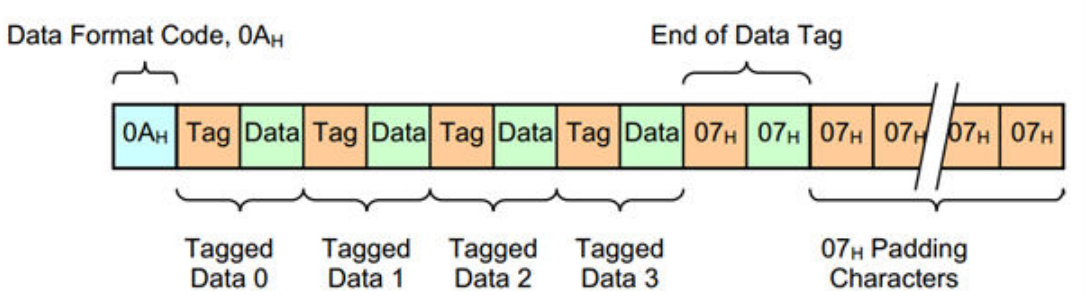


Table 41. List of used SMIA++ tags

Tag	Data byte description
07_H	End of data tdata byte value = 07_H)
AA_H	CCI register index MSB [15:8]
$A5_H$	CCI register index LSB [7:0]
$5A_H$	Auto-increment the CCI index after the data byte - valid data byte The data byte contains valid CCI register data

Although an SMIA encoding rule is used, the ISL structure is frame-to-frame identical. The first tags are used to set the MSB and LSB addresses (0xaa and 0xa5 respectively) of the CCI register. All subsequent tags are auto-increment tags. Consequently, the ISL content may be decoded using an SMIA compatible decoder. More simply, the ISL content can be decoded by reading every other byte of the ISL.

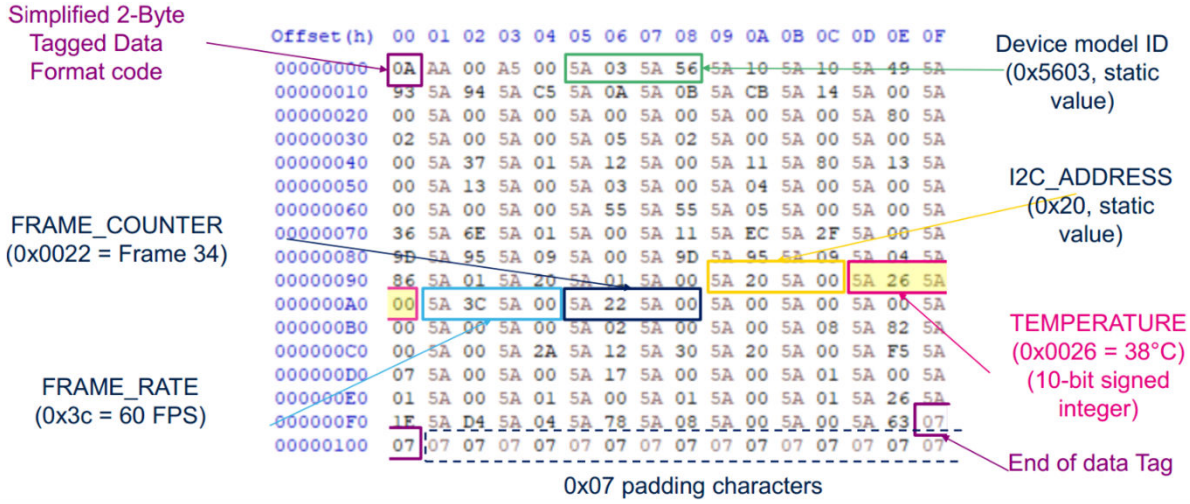
Example for the first ISL line

Status_register_address_N value [0:124] = (content of ISL byte) (2*N + 6).

Example for the second ISL line

Status_register_address_N value [125:249] = (content of ISL byte) (2*(N-125)+6).

Figure 15. **Example of ISL content and informative locations**



16.4 Image data

Image data contain pixel values based on different crop and binning sets. Values can be coded in RAW10 or RAW8 depending on the register settings. Image width and height should be even. For RAW10 data, the image width should be a multiple of 4 pixels.

The data type of the image is set by the OIF_ISL_CTRL register. The default value is 0x2B (RAW10). To comply with CSI-2 specifications, the host must update the OIF_ISL_CTRL register to 0x2A when the output format is RAW8. The default virtual channel of the image is 0x0.

17 Temperature

The device embeds a temperature sensor. Temperature is read at each frame in STREAMING state or in SW_STANDBY state if a THSENS_READ command is sent. The result, coded on 10 bits, is stored in the TEMPERATURE status register.

18 GPIOs

The device has eight configurable GPIOs. GPIO settings are part of the CONTEXT register group (see table below) and consequently they can differ from one context to another. Only GPIO_0 can be used to start frame synchronization that is, mode = 0xA:(VT_SLAVE_MODE).

GPIO settings are part of the CONTEXT register group (see table below) and consequently they can be configured differently from one context to another.

The GPIOs can be configured with the following modes:

- GPIO_IN: Standard input
- FSYNC_OUT: Start of frame
- STROBE_MODE: Envelope of the integration time
- PWM_STROBE: Envelope of the integration time multiplied by the PWM
- PWM: Continuous PWM
- GPIO_OUT: Register control level
- VSYNC_OUT_MODE0: Image readout envelope
- VSYNC_OUT_MODE1: Image readout start
- VSYNC_OUT_MODE2: Image readout end
- VT_SLAVE_MODE: Input to trigger follower mode (only valid for GPIO 0)

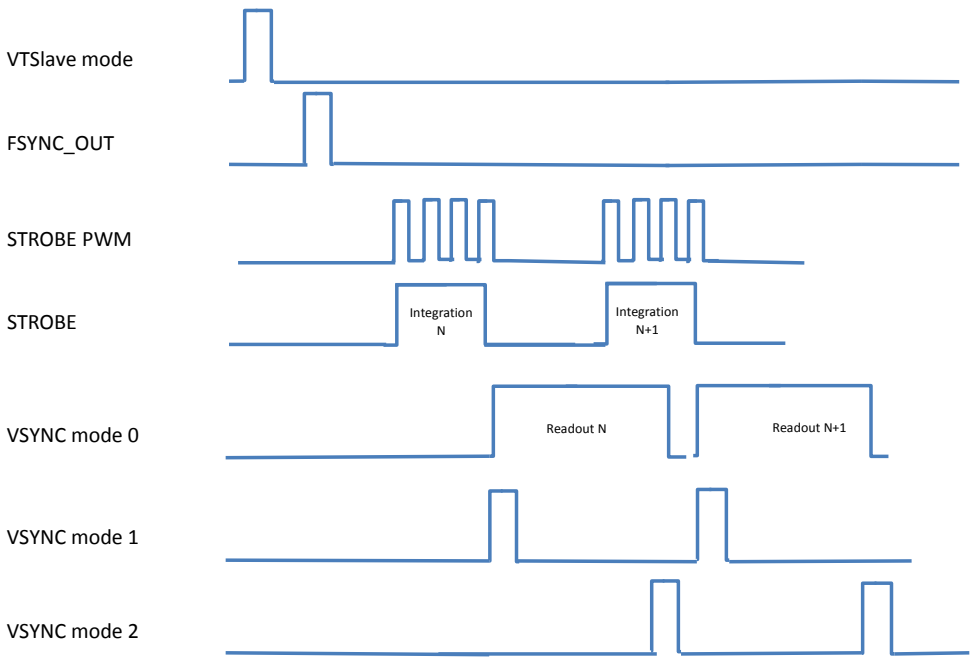
The GPIOs are active only during STREAMING state, although configuration is also possible during SW_STANDBY state.

In VdxxGxCCA the GPIO configuration can be updated by following these steps:

1. Set your GPIO configuration on context 0.
2. Issue the GPIO_UPDATE command.

During a context change, there is a one-frame delay in the application of the GPIO parameters. This means that the GPIO settings applied on the first frame of a new context are actually the GPIO settings from the previous context. The GPIO settings from the current context only apply from the second frame of the current context.

Figure 16. GPIO mechanism



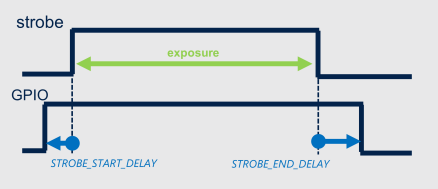
18.1 GPIO out control

When the GPIOs are set to GPIO_OUT, the polarity and value are set with the fields VALUE and POLARITY of the GPIO_X_CTRL register (refer to Section 18.4: GPIO control registers).

18.2 Strobe mode control

When GPIO is set to STROBE_MODE, it is possible to adjust the shift of the rising and falling edges of the output signal relative to the internal strobe. The STROBE_START_DELAY and STROBE_END_DELAY registers respectively control the shift of the rising and falling edges. The values are signed and expressed in line (refer to Section 18: GPIOs). The figure below shows an example with negative STROBE_START_DELAY and positive STROBE_END_DELAY.

Figure 17. Strobe signal control



18.3 PWM mode control

When the GPIOs are set to PWM or PWM_STROBE modes, the PWM frequency and duty cycle are set with the fields DUTY_CYCLE and CLK_DIVISOR of PWM_CTRL register as follows:

$$PWM frequency = \frac{PIXEL\_CLOCK}{(PWM)\ CLK\_DIVISOR + 1}$$

$$PWM \text{ duty cycle} = \frac{16 - DUTY\_CYCLE}{16}$$

The minimum CLK_DIVISOR value should be set to 7.

18.4 GPIO control registers

The following table details the registers used to control the GPIOs.

Table 42. GPIO control registers

CONTEXT register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
GPIO_0_CTRL	MODE	3:0	C	0x0: FSYNC_OUT 0x1: GPIO_IN 0x2: STROBE_MODE 0x3: PWM_STROBE 0x4: PWM 0x5: GPIO_OUT 0x6: VSYNC_OUT_MODE0 0x7: VSYNC_OUT_MODE1 0x8: VSYNC_OUT_MODE2 0xa: VT_SLAVE_MODE
	VALUE	4	C	0x0: GPIO_LOW 0x1: GPIO_HIGH
	POLARITY	5	C	0x0: NO_INVERSION 0x1: INVERTED
GPIO_1_CTRL	Refer to GPIO_0_CTRL			Refer to GPIO_0_CTRL
GPIO_2_CTRL				
GPIO_3_CTRL				
GPIO_4_CTRL				
GPIO_5_CTRL				
GPIO_6_CTRL				
GPIO_7_CTRL				
VSYNC_START_DELAY	VALUE	7:0	SI	[0:127]
VSYNC_END_DELAY	VALUE	7:0	SI	[-128:127]
STROBE_START_DELAY	VALUE	7:0	SI	[-128:127]
STROBE_END_DELAY	VALUE	7:0	SI	[-128:127]
PWM_CTRL	DUTY_CYCLE	3:0	UI	[1:15]
	CLK_DIVISOR	31:16	UI	[7:65535]

18.5 GPIO status registers

When a GPIO is set to GPIO_IN mode, its current value can be retrieved using the GPIO status registers as follows:

Table 43. GPIO status registers

STATUS register group				
Register name	**Field name**	**Bits**	**Data type**	**Values**
GPIO_0_STATUS	VALUE	0	B	Input value of GPIO 0 when set to GPIO_IN
GPIO_1_ STATUS	VALUE	0	B	Input value of GPIO 1 when set to GPIO_IN
GPIO_2_ STATUS	VALUE	0	B	Input value of GPIO 2 when set to GPIO_IN
GPIO_3_ STATUS	VALUE	0	B	Input value of GPIO 3 when set to GPIO_IN
GPIO_4_STATUS	VALUE	0	B	Input value of GPIO 4 when set to GPIO_IN
GPIO_5_ STATUS	VALUE	0	B	Input value of GPIO 5 when set to GPIO_IN
GPIO_6_ STATUS	VALUE	0	B	Input value of GPIO 6 when set to GPIO_IN
GPIO_7_ STATUS	VALUE	0	B	Input value of GPIO 7 when set to GPIO_IN

19 User interface description

19.1 STATUS registers

Register group range: 0x0000 to 0x00BF

Table 44. STATUS registers

Register address	Register name	Field name	Bits	Data type	Default	Values
0x0000	DEVICE_MODEL_ID	VALUE	15:0	UI	0x5603	Device name
0x0002	DEVICE_REVISION	TOP_DIE	15:8	UI	0x20	Top die mask code (HW)
					0 x 31	Top die mask code (HW) for VDxxGxCCA
		BOTTOM_DIE	7:0	UI	0x10	Bottom die mask code (HW)
0x0004	DEVICE_NUMBER_1	VALUE	31:0	UI	—	Traceability word 1
0x0008	DEVICE_NUMBER_2	VALUE	31:0	UI	—	Traceability word 2
0x000C	DEVICE_NUMBER_3	VALUE	31:0	UI	—	Traceability word 3
0x0010	DEVICE_NUMBER_4	VALUE	31:0	UI	—	Traceability word 4
0x0014	ROM_REVISION	MAJOR_REVISION	11:8	UI	0x3	Major ROM code revision
					0x4	Major ROM code revision for VDxxGxCCA
		MINOR_REVISION	7:4	UI	0x7	Minor ROM code revision
					0x2	Minor ROM code revision for VDxxGxCCA
		MICRO_REVISION	3:0	UI	0x0	Micro ROM revision
0x0018	UI_REVISION	MAJOR_REVISION	15:8	UI	0x04	Major UI revision
		MINOR_REVISION	7:0	UI	0x0	Minor UI revision
0x001A	OPTICAL_REVISION	COLOR_MODE	0	C	VD56G3: 0x00 VD66GY: 0x1 VD16GZ: 0x2	Optical revision
0x001C	ERROR_CODE	VALUE	15:0	UI	0x0	Error code
0x001E	FWPATCH_REVISION	MAJOR_REVISION	15:8	UI	0x0	Major firmware patch revision
		MINOR_REVISION	7:0	UI	0x0	Minor firmware patch revision
0x0020	VTPATCH_ID	VALUE	63:0	UI	0x0011 0011 8011 0011	VTPATCH_ID
0x0028	SYSTEM_FSM	VALUE	7:0	C	0x2	Sensor system FSM status: 0x00: HW_STANDBY 0x01: READY_TO_BOOT 0x02: SW_STANDBY 0x03: STREAMING 0xFF: ERROR

Register address	Register name	Field name	Bits	Data type	Default	Values
0x002A	NVM	DRIVER_FSM	7:6	C	0x0	NVM driver FSM: 0x00: IDLE 0x01: PROG_OP 0x02: READ_OP 0x03: ERROR
		ECC_ERROR	5	UI	0x0	ECC error detection: 0x0: no error 0X1: ECC error
		READ_FAIL	3	UI	0x0	Error during read
		READ_OK	2	UI	0x0	Read with success
		PROG_FAIL	1	UI	0x0	Error during programming
		PROG_OK	0	UI	0x0	Programming with success
0x0034	EXT_CLOCK	VALUE	31:0	UI	0x0	External clock frequency in Hz
0x0038	SYSTEM_PLL_CLK	VALUE	31:0	UI	0x0	PLL output frequency in Hz
0x003C	PIXEL_CLK	VALUE	31:0	UI	0x0	Pixel clock frequency in Hz
0x0040	MCU_CLK	VALUE	31:0	UI	0x0	MCU clock frequency in Hz
0x0044	CLK_SYS_PLL_PREDIV	VALUE	2:0	UI	0x0	PLL predivider value
0x0045	CLK_SYS_PLL_MULT	VALUE	7:0	UI	0x0	PLL multiplier value
0x0046	CLK_SYS_PLL_OPDIV	VALUE	7:0	UI	0x0	PLL output divider
0x0047	CLK_MIPI_PLL_MULT	VALUE	7:0	UI	0x0	PLL MIPI multiplier value
0x0048	CLK_MIPI_PLL_OPDIV	VALUE	7:0	UI	0x0	PLL multiplier value
0x004A	I2C_ADDRESS	DEVICE_ID	7:1	UI	0x10	Device I²C address
0x004C	TEMPERATURE	TH_DEG_INT	9:0	SI	0x0	Temperature in celsius
0x004E	FRAME_RATE	VALUE	15:0	UI	0x0	Frame rate
0x0050	FRAME_COUNTER	VALUE	15:0	UI	0x0	Global frame counter
0x0052	CONTEXT_FRAME_COUNTER	VALUE	15:0	UI	0x0	Context frame counter
0x0054	CONTEXT_REPEAT_COUNT	VALUE	15:0	UI	0x0	Number of frames per context
0x0056	CURRENT_CONTEXT	VALUE	7:0	UI	0x0	Current context
0x0057	NEXT_CONTEXT	VALUE	7:0	UI	0x0	Next context target, if >3 STOP STREAMING occurs after repeat count
0x0058	ORIENTATION	MODE	1:0	C	0x0	Image orientation mode control: 0x0: No flip 0x1: X flip 0x2: Y flip 0x3: XY flip
0x005A	VT_CTRL	ADC_MODE	2	B	0x0	0x0: 10 bits 0x1: 9 bits
		SYNC_MODE	1:0	C	0x0	Video timing synchronization mode: 0x0: MASTER

Register address	Register name	Field name	Bits	Data type	Default	Values
0x005A	VT_CTRL					0x1: STREAMING on EXTSYNC pulses 0x2: STREAMING on I²C pulses 0x3: Master synchronization on GPIO (only for VDxxGxCCA)
0x005B	FORMAT_CTRL	OUT_FORMAT	7:0	UI	0x0	Frame output format control: 0x08: RAW8 0x0A: RAW10
0x005C	OIF_CTRL	DATALANE1_SWAP	9	UI	0x0	Data lane swapping: 0x0: No swapping 0x1: Lane swapping
		DATALANE1_MAPPING	8:7	UI	0	Lane mapping for physical lane 1 (logical lane 0..1)
		DATALANE0_SWAP	6	UI	0x0	Data lane swapping: 0x0: No swapping 0x1: Lane swapping
		DATALANE0_MAPPING	5:4	UI	0x0	Lane mapping for physical lane 0 (logical lane 0..1)
		CLKLANE_SWAP	3	UI	0x0	Clock lane swapping: 0x0: No swapping 0x1: Lane swapping
		DATALANE_NB	2:0	UI	0x0	CSI data lane number: 0x1: single lane 0x2: dual lane
0x005E	OIF_VC_CTRL	ISL	3:2	UI	0x0	Virtual channel selection for ISL
		ACTIVE_PIX	1:0	UI	0x0	Virtual channel selection for image
0x005F	OIF_IMG_CTRL	DATA_TYPE	5:0	UI	0x0	Data type selection aligned with RAW format
0x0060	OIF_ISL_CTRL	DATA_TYPE	5:0	UI	0x0	Data type selection
0x0062	PATGEN_CTRL	TYPE	8:4	C	0x0	0x01: Solid color 0x02: Vertical color bars 0x10: Horizontal gray scale 0x11: Vertical gray scale 0x12: Diagonal gray scale 0x13: Pseudo random (PN28)
		ENABLE	1:0	UI	0x0	0x0: BYPASS 0x1: PATGEN
0x0064	APPLIED_COARSE_EXPOSURE	VALUE	15:0	UI	0x0	Coarse exposure time in line
0x0068	APPLIED_ANALOG_GAIN	VALUE	4:0	C	0x0	Analog gain = 32/(32 - APPLIED_ANALOG_GAIN)
0x006A	APPLIED_DIGITAL_GAIN	VALUE	12:0	FP 5.8	0x0	Digital gain applied
0x0072	AE_MODE	MODE	1:0	C	0x0	Exposure mode:

Register address	Register name	Field name	Bits	Data type	Default	Values
						0x0: Automatic 0x1: Freeze AutoExposure algorithm 0x2: Manual setting
0x0073	AE_STATUS	STATUS	0	UI	0x0	0x0: Unstable 0x1: Stable
0x0074	AE_MEAN_ENERGY	VALUE	15:0	FP 8.8	0x0	Mean energy of input image
0x0076	LINE_LENGTH	VALUE	15:0	UI	0x0	Line length
0x0078	FRAME_LENGTH	VALUE	15:0	UI	0x0	Frame length
0x007A	X_START	VALUE	15:0	UI	0x0	First column read by the VT
0x007C	X_END	VALUE	15:0	UI	0x0	Last column read by the VT
0x007E	Y_START	VALUE	15:0	UI	0x0	First row read by the VT
0x0080	Y_END	VALUE	15:0	UI	0x0	Last row read by the VT
0x0082	X_SIZE	VALUE	15:0	UI	0x0	Image X size (width)
0x0084	Y_SIZE	VALUE	15:0	UI	0x0	Image Y size(height)
0x0086	READOUT_CTRL	CFG	2:0	C	0x0	Readout mode: 0x0: Normal STREAMING 0x1: Digital binning x2 0x2: Digital binning x4 0x3: Subsampling x2 0x4: Subsampling x4
0x0088	WAIT_DELAY	VALUE	15:0	UI	0x0	Wait time before the next frame in addition to blanking in lines
0x008A	OUT_ROI_X_START	VALUE	15:0	UI	0x0	Image ROI
0x008C	OUT_ROI_X_END	VALUE	15:0	UI	0x0	
0x008E	OUT_ROI_Y_START	VALUE	15:0	UI	0x0	
0x0090	OUT_ROI_Y_END	VALUE	15:0	UI	0x0	
0x0092	OUT_ROI_X_SIZE	VALUE	15:0	UI	0x0	Image ROI X size (width)
0x0094	OUT_ROI_Y_SIZE	VALUE	15:0	UI	0x0	Image ROI Y size (height)
0x0096	OUTPUT_CTRL	VALUE	15:0	UI	0x0	Output control: 0x1: Image output only
0x00A7	USER	VALUE	7:0	UI	0x0	Copy of USER value in STREAM_DYNAMICS
0x00B4	GPIO_0_STATUS	VALUE	0	B	0x0	Value of the GPIO when configured as input (GPIO_IN)
0x00B5	GPIO_1_STATUS	VALUE	0	B	0x0	Value of the GPIO when configured as input (GPIO_IN)
0x00B6	GPIO_2_STATUS	VALUE	0	B	0x0	Value of the GPIO when configured as input (GPIO_IN)
0x00B7	GPIO_3_STATUS	VALUE	0	B	0x0	Value of the GPIO when configured as input (GPIO_IN)

Register address	Register name	Field name	Bits	Data type	Default	Values
0x00B8	GPIO_4_STATUS	VALUE	0	B	0x0	Value of the GPIO when configured as input (GPIO_IN)
0x00B9	GPIO_5_STATUS	VALUE	0	B	0x0	Value of the GPIO when configured as input (GPIO_IN)
0x00BA	GPIO_6_STATUS	VALUE	0	B	0x0	Value of the GPIO when configured as input (GPIO_IN)
0x00BB	GPIO_7_STATUS	VALUE	0	B	0x0	Value of the GPIO when configured as input (GPIO_IN)
0x00BF	DARKCAL_PEDESTAL	VALUE	7:0	UI	0x0	Pedestal value

19.2 COMMAND registers

Register group range: 0x0200 to 0x0203

The command registers are the only registers able to trigger an action from the device (see Section 8: Commands.

Table 45. COMMAND registers

Register address	Register name	Field name	Bits	Data type	Default	Values
0x0200	BOOT	COMMAND	1:0	C	0x0	0x0: CMD_ACK 0x1: BOOT 0x2: PATCH_SETUP
0x0201	STBY	COMMAND	3:0	C	0x0	0x0: CMD_ACK 0x1: START_STREAM 0x2: NVM_READ 0x3: NVM_PROG 0x4: THSENS_READ 0x5: I2C_ADDR_UPDATE VdxxGxCCA: 0x6 GPIO_UPDATE
0x0202	STREAMING	COMMAND	3:0	C	0x0	0x0: CMD_ACK 0x1: STOP_STREAM 0x2: VT_FSYNC_IN_I2C
0x0203	VTPATCHING	COMMAND	3:0	C		0x1: START_VTRAM_UPDATE 0x2: END_VTRAM_UPDATE

19.3 SENSOR SETTINGS registers

Register group range: 0x0220 to 0x0230

The SENSOR_SETTINGS registers must be set in the SW_STANDBY state. Any changes made in the STREAMING state are not considered before the next START_STREAM command.

Table 46. SENSOR SETTINGS registers

Register address	Register name	Field name	Bits	Data type	Default	Values
0x0220	EXT_CLOCK	VALUE	31:0	UI	0xb71b00	External clock frequency in Hz
0x0224	CLK_PLL_PREDIV	VALUE	2:0	UI	0x1	Pre-PLL clock divider value
0x0225	CLK_PLL_POSTDIV	VALUE	7:0	UI	0x01	Post PLL clock divider value
0x0226	CLK_SYS_PLL_MULT	VALUE	7:0	UI	0x43	PLL multiplier value
0x0227	VT_CLK_DIV	VALUE	7:0	UI	0x5	VT clock divider
0x0229	NVM_NB_OF_WORDS	VALUE	7:0	UI	0x0	Number of 32-bit words to read/write in burst mode
0x022A	NVM_START_ADDRESS	VALUE	8:0	UI	0x0	Start address in NVM for read/write burst operation
0x0230	DEVICE_I2C_CTRL	MODE	8	C	0x1	I²C mode 0x0: FAST mode 0x1: FAST+ mode
		DEVICE_ID	7:1	UI	0x10	Device I²C address

19.4 STATIC registers

Register group range: 0x0300 to 0x0341

The STATIC registers must be set in SW_STANDBY state. Any changes made in STREAMING state are not considered before the next START_STREAM command.

Table 47. STATIC registers

Register address	Register name	Field name	Bits	Data type	Default	Values
0x0300	LINE_LENGTH	VALUE	15:0	UI	0x4D4	Line length configuration
0x0302	ORIENTATION	MODE	1:0	C	0x0	Image orientation mode control: 0x0: No flip 0x1: X flip 0x2: Y flip 0x3: XY flip
0x0304	X_START	VALUE	15:0	UI	0x0	First column read by the VT
0x0306	X_END	VALUE	15:0	UI	0x0	Last column by the VT
0x309	VT_CTRL	ADC_MODE	2	B	0x0	Video timing mode of the ADC: 0x0: Standard mode 10 bits 0x1: Fast mode 9 bits
		SYNC_MODE	1:0	C	0x0	Video timing synchronization mode: 0x0: MASTER 0x1: STREAMING on EXTSYNC pulses 0x2: STREAMING on I²C pulses
0x030A	FORMAT_CTRL	OUT_FORMAT	4:0	UI	0xA	Frame output format control: 0x8: RAW8 0xA: RAW10
0x030C	OIF_CTRL	DATALANE1_SWAP	9	C	0x0	Data lane swapping: 0x0: No swapping 0x1: Lane swapping
		DATALANE1_MAPPING	8:7	UI	0x1	Lane mapping for physical lane 1 (logical lane 0..1)
		DATALANE0_SWAP	6	C	0x0	Data lane swapping: 0x0: No swapping 0x1: Lane swapping
		DATALANE0_MAPPING	5:4	0x0	0x0	Lane mapping for physical lane 0 (logical lane 0..1)
		CLKLANE_SWAP	3	C	0x0	Clock lane swapping: 0x0: No swapping 0x1: Lane swapping
		DATALANE_NB	2:0	UI	0x2	CSI data lane number: 0x1: Single lane 0x2: Dual lane

Register address	Register name	Field name	Bits	Data type	Default	Values
0x030E	OIF_VC_CTRL	ISL	3:2	UI	0x0	Virtual channel selection for ISL
		ACTIVE_PIX	1:0	UI	0x0	Virtual channel selection is common for output image
0x030F	OIF_IMG_CTRL	DATA_TYPE	5:0	UI	0x2B	Data type selection for RAW image format
0x0310	OIF_ISL_CTRL	DATA_TYPE	5:0	UI	0x12	Data type selection for status line
0x0312	OIF_CSI_BITRATE	ENABLE	15:0	UI	0x03FE	Output bit rate in Mbps
0x0318	DUSTER_CTRL	RING_ENABLE	4	UI	0x1	Duster ring correction control: 0x0: Disable duster ring correction 0x1: Enable duster ring correction
		MAP_ENABLE	3	UI	0x0	RESERVED 0x0: Disable
		DYN_ENABLE	2:1	C	0x1	Duster mode control: 0x0: Disable duster dynamic correction 0x1: Duster corrects singlets and couplets 0x2: Duster corrects only couplets
		ENABLE	0	UI	0x1	Duster mode control: 0x0: Fully disable duster 0x1: Enable duster modules
0x031C	DUSTER_DMP_CC_SIGMA_X	EXPO_HIGH	31:16	UI	0x0050	2 point damper for CC sigma (high exposure value user)
		EXPO_LOW	15:0	UI	0x000A	2 point damper for CC sigma (low exposure value user)
0x0320	DUSTER_DMP_CC_SIGMA_Y	TH_HIGH	31:16	UI	0x0020	2 point damper for CC sigma (maximum threshold value)
		TH_LOW	15:0	UI	0x0010	2 point damper for CC sigma (minimum threshold value)
0x0324	DUSTER_DMP_RC_TH_X	EXPO_HIGH	31:16	UI	0x0050	2 point damper for ring correction (high exposure value user)
		EXPO_LOW	15:0	UI	0x000A	2 point damper for ring correction (low exposure value user)
0x0328	DUSTER_DMP_RC_TH_Y	TH_HIGH	31:16	UI	0x00DC	2 point damper for ring correction (maximum threshold value)
		TH_LOW	15:0	UI	0x00DC	2 point damper for ring correction (minimum threshold value)
0x0333	ISL_ENABLE	ENABLE	0	C	0x1	0x0: Disable 0x1: Enable
0x0334	STATUSLINE_PKT_EQ_ACTIVELINE	ENABLE	0	UI	0x1	0x0: Disable 0x1: Enable

Register address	Register name	Field name	Bits	Data type	Default	Values
0x0335	OUTPUT_CTRL	CONTROL	1:0	C	0x1	Output control: 0x1: Image output only
0x033A	FSYNC_IN_DELAY	VALUE	15:0	UI	0x0	Tunable follower mode delay in lines
0x0340	DARKCAL_CTRL	ENABLE	1:0	C	0x1	DarkCal mode control: 0x0: Disable DarkCal is disabled 0x1: Enable DarkCal in automatic mode

19.5 DYNAMIC registers

Register group range: 0x0400 to 0x0448

The DYNAMIC register settings can be set either in SW_STANDBY or STREAMING state. These settings are common to all contexts. The table below shows the DYNAMIC parameters.

Table 48. DYNAMIC registers

Register address	Register name	Field name	Bits	Data type	Default	Values
0x0400	PATGEN_CTRL	TYPE	9:4	C	0x0	0x00: Solid color 0x08: Vertical color bars 0x20: Horizontal gray scale 0x21: Vertical gray scale 0x22: Diagonal gray scale 0x28: Pseudo random (PN28)
		ENABLE	1:0	UI	0x0	Pattern generator configuration 0x0: Disable 0x1: Enable
0x0415	DARKCAL_PEDESTAL	VALUE	7:0	UI	0x40	Pedestal value
0x0428	AE_FORCE_COLDSTART	VALUE	0	B	0x1	Force the exposure to start from a cold start 0x0: Disable 0x1: Enable
0x042A	AE_COLDSTART_COARSE_EXPOSURE	VALUE	15:0	UI	0x32	Coarse exposure time in line
0x042C	AE_COLDSTART_ANALOG_GAIN	VALUE	4:0	UI	0x0	Cold start analog gain = 32/(32 - AE_COLDSTART_ANALOG_GAIN)
0x042E	AE_COLDSTART_DIGITAL_GAIN	VALUE	12:0	FP 5.8	0x100	Digital gain for cold start
0x0430	AE_COMPILER_CONTROL	FLICKER_FREQUENCY	1	C	0x0	0x0: 50 Hz 0x1: 60 Hz
		MODE	0	C	0x0	0x0: priority to minimum gain 0x1: priority to flicker free
0x0432	AE_ROI_START_H	VALUE	15:0	UI	0x0	0x0: Priority to minimum gain 0x1: Priority to flicker free
0x0434	AE_ROI_START_V	VALUE	15:0	UI	0x0	
0x0436	AE_ROI_END_H	VALUE	15:0	UI	0x463	
0x0438	AE_ROI_END_V	VALUE	15:0	UI	0x553	
0x043A	AE_COMPENSATION	VALUE	15:0	SFP 8.8	0x0	Change the target exposure in stops by increasing or decreasing this value
0x043C	AE_TARGET_PERCENTAGE	VALUE	15:0	UI	0x1B	Exposure target of the AEC algorithm
0x043E	AE_STEP_PROPORTION	VALUE	15:0	FP 8.8	0x8C	Autoexposure step proportion
0x0440	AE_LEAK_PROPORTION	VALUE	15:0	FP 1.15	0x2CCC	Autoexposure leaky proportion
0x0442	USER	VALUE	7:0	UI	0x00	Register available for the host/user

Register address	Register name	Field name	Bits	Data type	Default	Values
0x0444	READ_OFFSET	VALUE	15:0	UI	0x12	READOUT offset
0x0448	GROUP_PARAM_HOLD	VALUE	0	UI	0x0	Hold the application of dynamic parameters

19.6 CONTEXT 0 registers

Register group range: 0x044C to 0x048E

The CONTEXT 0 registers can be set either in SW_STANDBY or STREAMING state.

Table 49. CONTEXT 0 registers

Register address	Register name	Field name	Bits	Data type	Default	Values
0x044C	EXP_MODE	MODE	1:0	C	0x2	Control of exposure mode: 0x0: Automatic 0x1: Freeze AutoExposure algorithm 0x2: Manual setting
0x044D	MANUAL_ANALOG_GAIN	VALUE	4:0	C	0x0	Analog gain = 32/(32 - MANUAL_ANALOG_GAIN)
0x044E	MANUAL_COARSE_EXPOSURE	VALUE	15:0	UI	0x0032	Coarse exposure time in line
0x0450	MANUAL_DIGITAL_GAIN_CH0	VALUE	12:0	FP 5.8	0x0100	Digital gain in manual mode
0x0452	MANUAL_DIGITAL_GAIN_CH1 channel 1	VALUE	12:0	FP 5.8	0x0100	Digital gain in manual mode (VD66GY)
0x0454	MANUAL_DIGITAL_GAIN_CH2 channel 2	VALUE	12:0	FP 5.8	0x0100	Digital gain in manual mode (VD66GY)
0x0456	MANUAL_DIGITAL_GAIN_CH3 channel 3	VALUE	12:0	FP 5.8	0x0100	Digital gain in manual mode (VD66GY)
0x0458	FRAME_LENGTH	VALUE	15:0	UI	0x05C4	Length of the frame in lines
0x045A	Y_START	VALUE	15:0	UI	0x0000	First row read by VT
0x045C	Y_END	VALUE	15:0	UI	0x0553	Last row read by VT
0x045E	OUT_ROI_X_START	VALUE	15:0	UI	0x0000	Definition of image ROI according to VT output
0x0460	OUT_ROI_X_END	VALUE	15:0	UI	0x0463	
0x0462	OUT_ROI_Y_START	VALUE	15:0	UI	0x0000	
0x0464	OUT_ROI_Y_END	VALUE	15:0	UI	0x0553	
0x0467	GPIO_0_CTRL	POLARITY	5	C	0x0	Polarity value: 0: Normal 1: Inverted
		VALUE	4	UI	0x0	Value of the GPIO when configured as GPIO_OUT: 0: Low level 1: High level
		MODE	3:0	C	0x1	GPIO mode: 0x0: FSYNC_OUT 0x1: GPIO_IN 0x2: STROBE_MODE 0x3: PWM_STROBE 0x4: PWM 0x5: GPIO_OUT

Register address	Register name	Field name	Bits	Data type	Default	Values
0x0467	GPIO_0_CTRL					0x6: VSYNC_OUT_MODE0 0x7: VSYNC_OUT_MODE1 0x8: VSYNC_OUT_MODE2 0xA: VT_SLAVE_MODE
0x0468	GPIO_1_CTRL	POLARITY	5	C	0x0	Polarity value: 0: Normal 1: Inverted
		VALUE	4	UI	0x0	Value of the GPIO when configured as GPIO_OUT: 0: Low level 1: High level
		MODE	3:0	C	0x0	GPIO mode: 0x0: FSYNC_OUT 0x1: GPIO_IN 0x2: STROBE_MODE 0x3: PWM_STROBE 0x4: PWM 0x5: GPIO_OUT 0x6: VSYNC_OUT_MODE0 0x7: VSYNC_OUT_MODE1 0x8: VSYNC_OUT_MODE2
0x0469	GPIO_2_CTRL	POLARITY	5	C	0x0	Refer to GPIO_1_CTRL description
		VALUE	4	UI	0x0	
		MODE	3:0	C	0x6	
0x046A	GPIO_3_CTRL	POLARITY	5	C	0x0	Refer to GPIO_1_CTRL description
		VALUE	4	UI	0x0	
		MODE	3:0	C	0x2	
0x046B	GPIO_4_CTRL	POLARITY	5	C	0x0	Refer to GPIO_1_CTRL description
		VALUE	4	UI	0x0	
		MODE	3:0	C	0x4	
0x046C	GPIO_5_CTRL	POLARITY	5	C	0x0	Refer to GPIO_1_CTRL description
		VALUE	4	UI	0x0	
		MODE	3:0	C	0x3	
0x046D	GPIO_6_CTRL	POLARITY	5	C	0x0	Refer to GPIO_1_CTRL description
		VALUE	4	UI	0x0	
		MODE	3:0	C	0x5	
0x046E	GPIO_7_CTRL	POLARITY	5	C	0x0	Refer to GPIO_1_CTRL description
		VALUE	4	UI	0x0	
		MODE	3:0	C	0x0	

Register address	Register name	Field name	Bits	Data type	Default	Values
0x046F	VSYNC_START_DELAY	DELAY_L	7:0	SI	0x0	Delay to apply in line (signed value)
0x0470	VSYNC_END_DELAY	DELAY_L	7:0	SI	0x0	Delay to apply in line (signed value)
0x0471	STROBE_START_DELAY	DELAY_L	7:0	SI	0x0	Delay to apply in line (signed value)
0x0472	STROBE_END_DELAY	DELAY_L	7:0	SI	0x0	Delay to apply in line (signed value)
0x0474	PWM_CTRL	CLK_DIVISOR	31:16	UI	0x0064	Refer to clock tree
		DUTY_CYCLE	3:0	UI	0x7	Duty cycle = (16 - DUTY_CYCLE) / 16
0x047E	READOUT_CTRL	CFG	2:0	C	0x0	Control of readout mode: 0x0: Normal STREAMING 0x1: Digital binning x2 0x2: Digital binning x4 0x3: Subsampling x2 0x4: Subsampling x4
0x048A	WAIT_DELAY	VALUE	15:0	UI	0x0000	Wait time before the first frame of the next context (in addition to blanking in line)
0x048C	CONTEXT_REPEAT_COUNT	VALUE	15:0	UI	0x0000	Number of frames for this context
0x048E	NEXT_CONTEXT	VALUE	15:0	UI	0x1	Next context, if value >3 STOP STREAMING occurs after repeat count

19.7 NVM MIRROR registers

Register group range: 0x0640 to 0x0838

Table 50. NVM MIRROR registers

Register address	Register name	Field name	Bits	Data type	Default	Values
0x0640	RESERVED 0	VALUE	31:0	UI	0x00	First reserved word
0x0804	RESERVED 113	VALUE	31:0	UI	0x00	Last reserved word
0x0808	I2C_ADDRESS	I2C_KEY	31:24	UI	0x00	Security key
		DEVICE_ID_2	22:16			I²C address. Device I²C address
		DEVICE_ID_1	14:8			
		DEVICE_ID_0	6:0			
0x080C	CUSTOMER_WORD_0	VALUE	31:0	UI	0x00	First word available for customer
0x0838	CUSTOMER_WORD_11	VALUE	31:0	UI	0x00	Last word

19.8 Miscellaneous registers

Register group range: 0x0946 to 0x0980

The miscellaneous registers must be set in SW_STANDBY state.

Table 51. Miscellaneous registers

Register address	Register name	Field name	Bits	Data type	Default	Values
0x0946	EXP_COARSE_INTG_MARGIN	VALUE	15:0	UI	0x0022	Limit maximum exposure coarse value in line
0x0956	DPHYTX_CTRL	VALUE	7:0	CODE	0x1C	0x0C: Disable Continuous mode 0x1C: Enable Continuous mode
0x095E	MAX_DG	VALUE	12:0	FP 5.8	0x0800	Maximum digital gain
0x0960	MAX_AG_CODED	VALUE	4:0	C	0x0018	Maximum analog gain = 32/(32 - MAX_AG_CODED)
0x097C	MIN_DG	VALUE	12:0	FP 5.8	0x0100	Minimum digital gain
0x097E	MIN_AG_CODED	VALUE	4:0	C	0x0000	Minimum analog gain = 32/(32 - MIN_AG_CODED)
0x0980	MAX_BRIEF_DESCRIPTORS	VALUE	15:0	UI	0x0800	Maximum of BRIEF corners. All additional corners sent by FAST are dropped

Revision history

Table 52. **Document revision history**

Date	Version	Changes
17-Apr-2024	8	First public release

Contents