Crystal Image through Imaging Innovation

PIXELPLUS

PK9210K

1/2.92” Full HD RGB Bayer CMOS Image Sensor

120dB HDR 30 fps

Ver 0.3

Last update : June 13, 2023

6th Floor, 105, Gwanggyo-ro, Yeongtong-gu, Suwon-si, Gyeonggi-do, 16229, Korea

Tel : 82-31-888-5300, FAX : 82-31-888-5398

General Description

The PK9210K is the 1/2.92” RGB bayer CMOS image sensor (CIS) designed to support 2.0MP at 30 frames per second (fps).

The PK9210K consists of 1960 (H) x 1120 (V) effective pixels with 12 added active pixels on each side and 8 pixels on each side for color interpolation.

The PK9210K has excellent noise performance for low light condition and high dynamic range (HDR) support using by DCG (Dual Conversion Gain) and multi- exposure method up to 120dB. It incorporates on-chip CIS functions such as Defective Pixel Correction (DPC), Purple Fringing Reduction (PFR), exposure control, HDR combine reconstruction, and so on. It enables the PK9210K shows no saturation image in the worst contrast situations.

The PK9210K is suitable for a rear view camera and surround-view camera, home appliances and security applications with excellent image quality with 2.8V/1.8V/1.2V power supply.

Applications

- Rear View Camera
- 360° Surround View Monitoring System (SVM)
- Security
- Home appliances

Features

- Support for display image size 1920 x 1080 with 2.0MP
- Provide for effective image size 1960 x 1120 with added array +24(H), +24(V) and for color processing +16(H), +16(V)
- Support for high dynamic range 120dB with DCG(Dual Conversion Gain) and multi exposure method
- Support for combined RGB bayer output format : DVP 12bit RAW / MIPI 2-lanes RAW
- Functions of image sensor processor functions such as lens shading correction (LSC), DPC, PFR, HDR combination, automatic black level correction, compression, etc
- Programmable frame size, window size, and exposure
- Support for external synchronization capability (Genlock)
- Built-in 2K bit of one-time programmable memory (OTP) for storing the chip information and chip calibration
- Embedded spread spectrum clock generation (SSCG) for EMI avoidance

Key Performance Parameter

Parameter	Typical value
Pixel size	2.8 um x 2.8 um
Effective pixel array	1960(H) x 1120 (V)
Effective image area	5.488 mm x 3.136 mm
Optical format	1/2.92 inch
Input clock frequency	27 [MHz]
Output interface	2-Lane MIPI / DVP Combo
Max. frame rate	HDR 30 fps only
Dark Current	23 e-/sec @60℃
Sensitivity	30.3K e-/Lux. sec
Power supply	HVDD : 1.8~2.8 [V] AVDD : 2.8 [V] DVDD : 1.2 [V]
Power consumption @ 30fps (HVDD = 1.8V / HVDD = 2.8V)	DVP : 204mW / 234mW MIPI : 201mW / 202mW
Standby Power consumption (HVDD = 1.8V / HVDD = 2.8V)	2mW / 2mW
Operating temp.	-40~105 [°C] (Ambient)
Max. dynamic range	120 dB
SNR	44 dB @60℃
Package type	NeoPAC I,
Package size	7.6 x 5.86 mm

Table of Contents

List of tables

List of figures

1. Application system

1.1. System overview

The PK9210K bayer image sensor has a 1/2.92” optical format, 1920 x 1080 display array, 2.8um DCG pixel and HDR sensor intended for low-cost SVM/RVC applications for the automotive market, security/home appliances market.

The PK9210K has the function of output 12-bit compression mapped HDR combined multi-image using the 2- exposure with DCG. The device generates the 3-image capture for acquiring 120dB HDR image. It supports output interfaces such as 2-lane MIPI and 12-bit DVP.

1.2. Ball configuration and assignment (58-Ball)

Figure 1. NeoPAC I Ball Map(Top View)

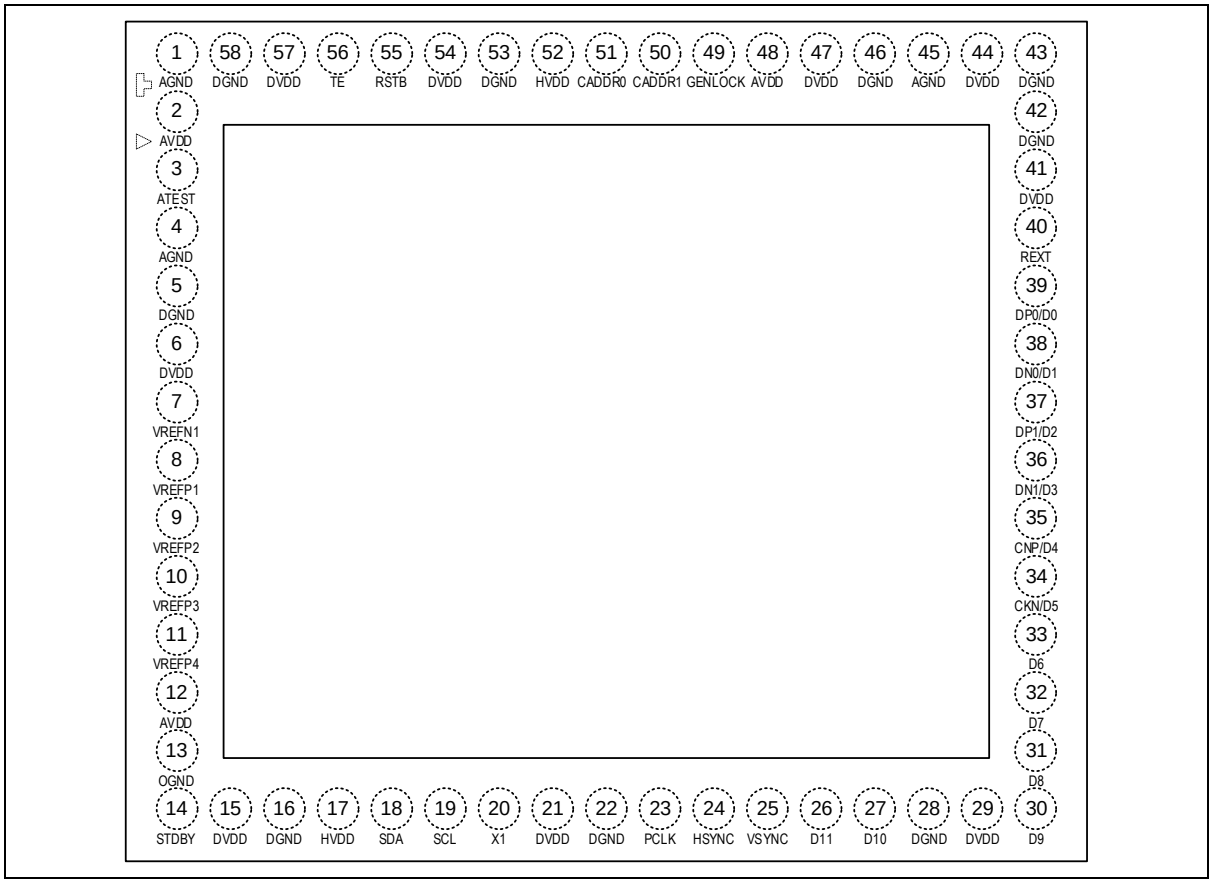


Table 1. Ball Description

Ball	Ball Name	IO	Pull up/ Pull down	Ball Description
1	AGND	P	-	Analog GND
2	AVDD	P	-	Analog VDD 2.8V It should be tied with nearby AGND by 1uF bypass capacitors.
3	ATEST	O	-	Analog test output
4	AGND	P	-	Analog GND

Ball	Ball Name	IO	Pull up/ Pull down	Ball Description
5	DGND	P	-	Digital (Core) GND
6	DVDD	P	-	Digital (Core) VDD 1.2V DC It should be tied with nearby DGND by 1uF bypass capacitors.
7	VREFN1	O	-	VREFN1 output. It should be tied with nearby AGND by 1uF bypass capacitors.
8	VREFP1	O	-	VREFP1 output. It should be tied with nearby AGND by 1uF bypass capacitors.
9	VREFP2	O	-	VREFP2 output. It should be tied with nearby AGND by 1uF bypass capacitors.
10	VREFP3	O	-	VREFP3 output. It should be tied with nearby AGND by 1uF bypass capacitors.
11	VREFP4	O	-	VREFP4 output. It should be tied with nearby AGND by 1uF bypass capacitors.
12	AVDD	P	-	Analog VDD 2.8V It should be tied with nearby AGND by 1uF bypass capacitors.
13	OGND	P	-	Analog GND for OTP
14	STDBY	I	pulldown	Power stdby mode. When Stdby ='1', there's no current flow in any analog circuit branch, neither any beat of digital clock.
15	DVDD	P	-	Digital (Core) VDD 1.2V DC It should be tied with nearby DGND by 1uF bypass capacitors.
16	DGND	P	-	Digital (Core) GND
17	HVDD	P	-	IO VDD 1.8V ~ 2.8V DC It should be tied with nearby DGND by 1uF bypass capacitors.
18	SDA	BIO	pullup	2-wire serial interface clock, SDA line is pulled up to HVDD by off-chip resistor
19	SCL	BIO	pullup	2-wire serial interface data, SCL line is pulled up to HVDD by off-chip resistor
20	X1	I	pulldown	Master clock input pad
21	DVDD	P	-	Digital (Core) VDD 1.2V DC It should be tied with nearby DGND by 1uF bypass capacitors.
22	DGND	P	-	Digital (Core) GND
23	PCLK	O	pulldown	PAD clock Data can be latched by external devices at the rising or falling edge of PCLK
24	HSYNC	O	pulldown	Horizontal synchronization pulse. HSYNC is high (or low) for the horizontal window of interest. It can be programmed to appear or not outside the vertical window of interest.
25	VSYNC	O	pulldown	Vertical sync : Indicates the start of a new frame
26	D11	O	pulldown	Digital Output bit 11
27	D10	O	pulldown	Digital Output bit 10
28	DGND	P	-	Digital (Core) GND
29	DVDD	P	-	Digital (Core) VDD 1.2V DC It should be tied with nearby DGND by 1uF bypass capacitors.
30	D9	O	pulldown	Digital Output bit 9

Ball	Ball Name	IO	Pull up/ Pull down	Ball Description
31	D8	O	pulldown	Digital Output bit 8
32	D7	O	pulldown	Digital Output bit 7
33	D6	O	pulldown	Digital Output bit 6
34	CKN/D5	O	pulldown	MIPI Clock Negative Output / Digital Output bit 5
35	CKP/D4	O	pulldown	MIPI Clock Positive Output / Digital Output bit 4
36	DN1/D3	O	pulldown	MIPI DN1 Output / Digital Output bit 3
37	DP1/D2	O	pulldown	MIPI DP1 Output / Digital Output bit 2
38	DN0/D1	O	pulldown	MIPI DN0 Output / Digital Output bit 1
39	DP0/D0	O	pulldown	MIPI DP0 Output / Digital Output bit 0
40	REXT	O	-	External Resistor for MIPI
41	DVDD	P	-	Digital (Core) VDD 1.2V DC It should be tied with nearby DGND by 1uF bypass capacitors.
42	DGND	P	-	Digital (Core) GND
43	DGND	P	-	PLL GND
44	DVDD	P	-	PLL VDD 1.2V DC It should be tied with nearby DGND_PLL by 1uF bypass capacitors.
45	AGND	P	-	Analog GND
46	DGND	P	-	Digital (Core) GND
47	DVDD	P	-	Digital (Core) VDD 1.2V DC It should be tied with nearby DGND by 1uF bypass capacitors.
48	AVDD	P	-	Analog VDD 2.8V It should be tied with nearby AGND by 1uF bypass capacitors.
49	GENLOCK	BIO	pulldown	External Frame sync input. Slave chip can receive the external frame sync signal from master chip/External Frame sync output. Master chip can output the external frame sync signal through this pad to synchronize all digital outputs of two or more chips
50	CADDR1	I	pullup	Chip address bit 1
51	CADDR0	I	pullup	Chip address bit 0
52	HVDD	P	-	IO VDD 1.8V ~ 2.8V DC It should be tied with nearby DGND by 1uF bypass capacitors.
53	DGND	P	-	IO GND
54	DVDD	P	-	Digital (Core) VDD 1.2V DC It should be tied with nearby DGND by 1uF bypass capacitors.
55	RSTB	I	pullup	System reset must remain low for at least 8 master clocks after power is stabilized. When the sensor is reset, all registers are set to their default values.
56	TE	I	pulldown	Chip test mode enable
57	DVDD	P	-	Digital (Core) VDD 1.2V DC It should be tied with nearby DGND by 1uF bypass capacitors.
58	DGND	P	-	Digital (Core) GND

1.3. Reference Schematic

Figure 2 shows the reference schematic with power supply and signal connection of the PK9210K. The PK9210K chip is powered from AVDD, DVDD, HVDD. The range of each power sources is as follows.

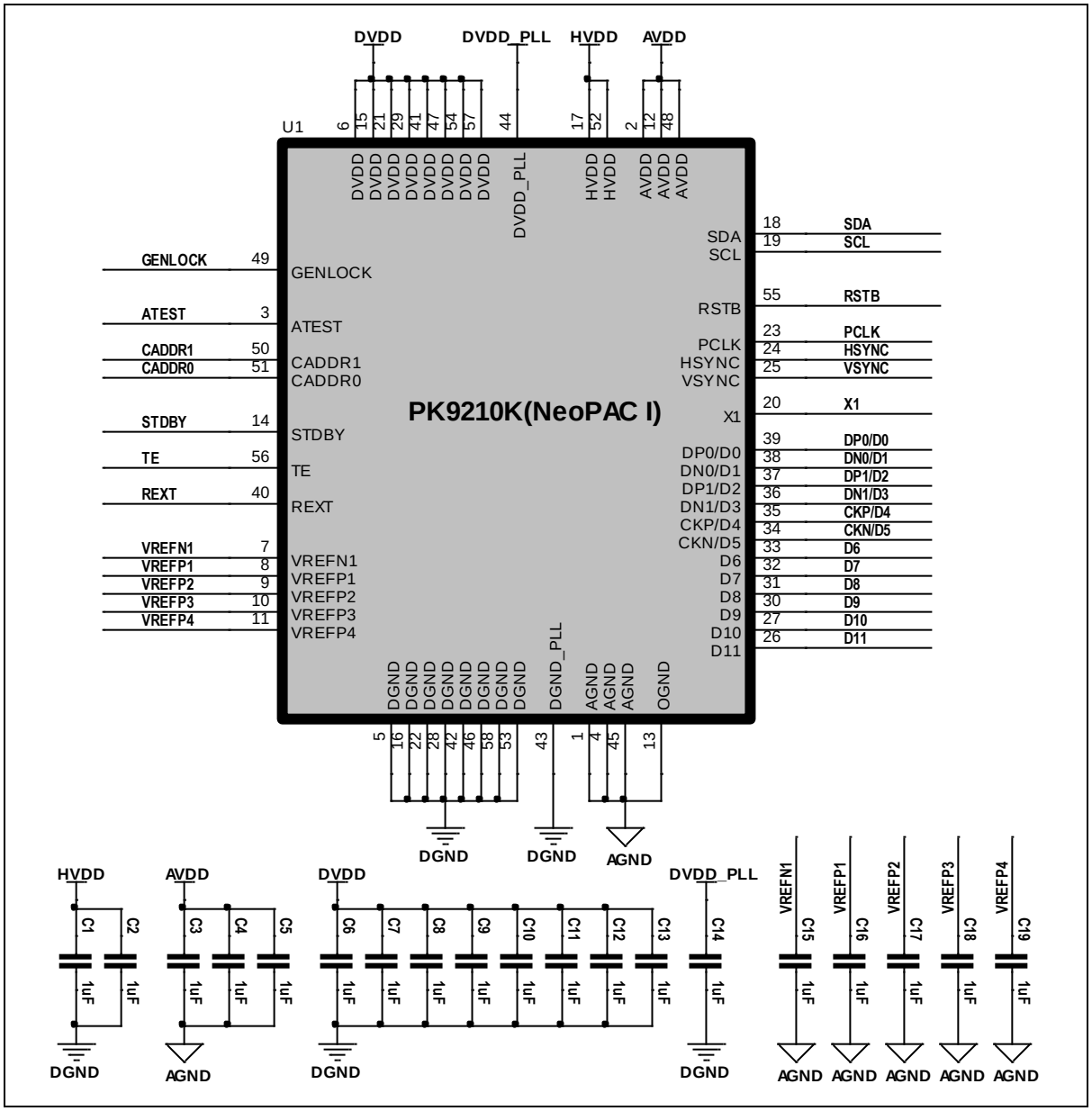
Analog power supply AVDD: 2.8V

Digital power supply DVDD: 1.2V

I/O power supply HVDD: 1.8V/2.8V

(Figure 2 reference schematic is preliminary data)

Figure 2. PK9210K (NeoPAC I) reference schematic



1.4. Power Sequence

When starting the PK9210K chip, it is best to apply all external applied power at the same time, otherwise DVDD must be applied first and then another power shall be applied. Reset must be maintained for 8 clock (X1) after HVDD is applied and then turned off.

Figure 3. PK9210K Power Sequence

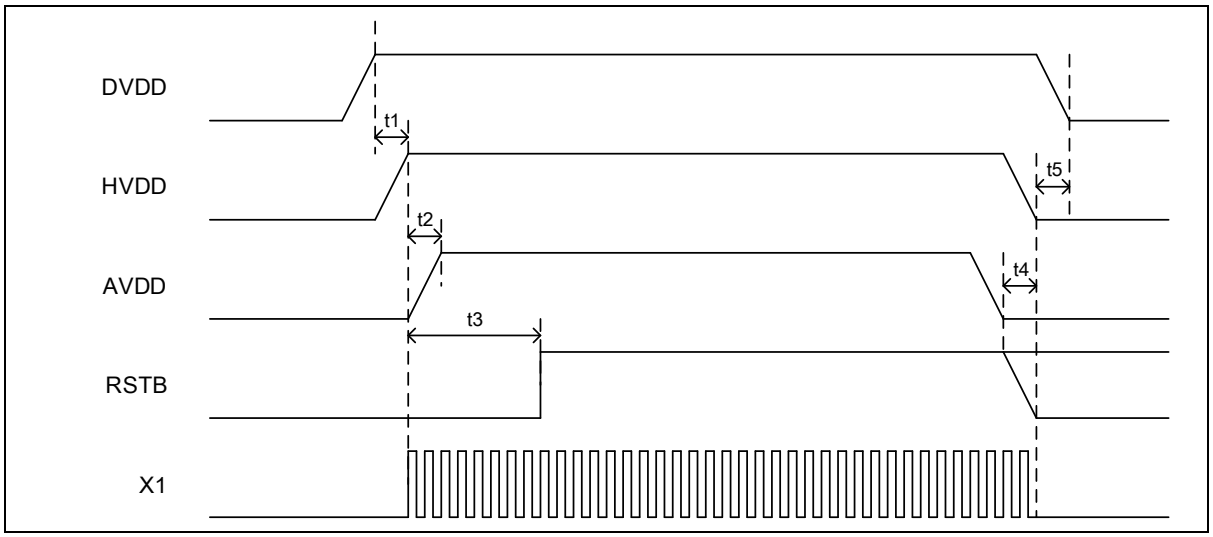


Table 2. Power sequence timing constraint

Symbol	Description	Min	Max	Unit
t1	From DVDD rising to HVDD rising	0	10	ms
t2	From HVDD rising to AVDD	0	10	ms
t3	period of maintaining the reset after DVDD rising	8		X1
t4	From AVDD falling to HVDD falling	0		ms
t5	From HVDD falling to HVDD falling	0		ms

1.5. System Reset

The PK9210K has two methods to reset: hard reset and soft reset. When starting a hard reset, the RSTB must remain low (active low) for a minimum of 8 clocks (X1) to properly reset the sensor. All registers are set to their default values after reset. Figure 4 shows device soft reset by setting “softreset” register through I2C interface. When “softreset” register is set, both async_rstb (asynchronous reset) and sync_rstb(synchronous reset) signals changes from 1 to 0 and holds for 1 clock of SCL. Afterward, async_rstb is set back to 1 while sync_rstb holds 0 for another 16 clocks of pclk for stable reset operation. Therefore, PK9210K requires at least 1 clock of SCL and 16 clocks of pclk to perform a soft reset operation.

Figure 4. Soft reset

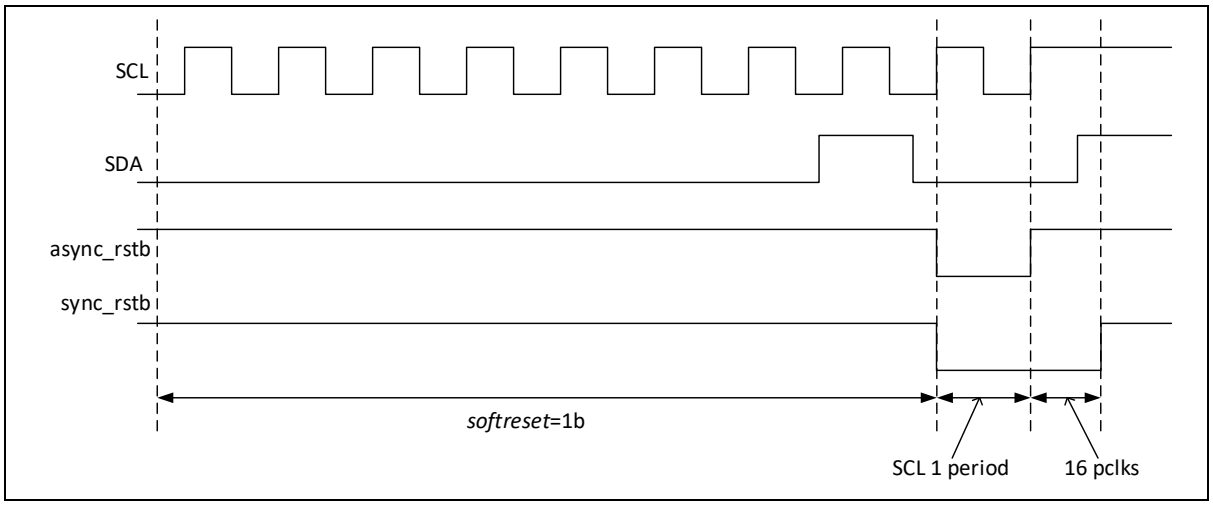


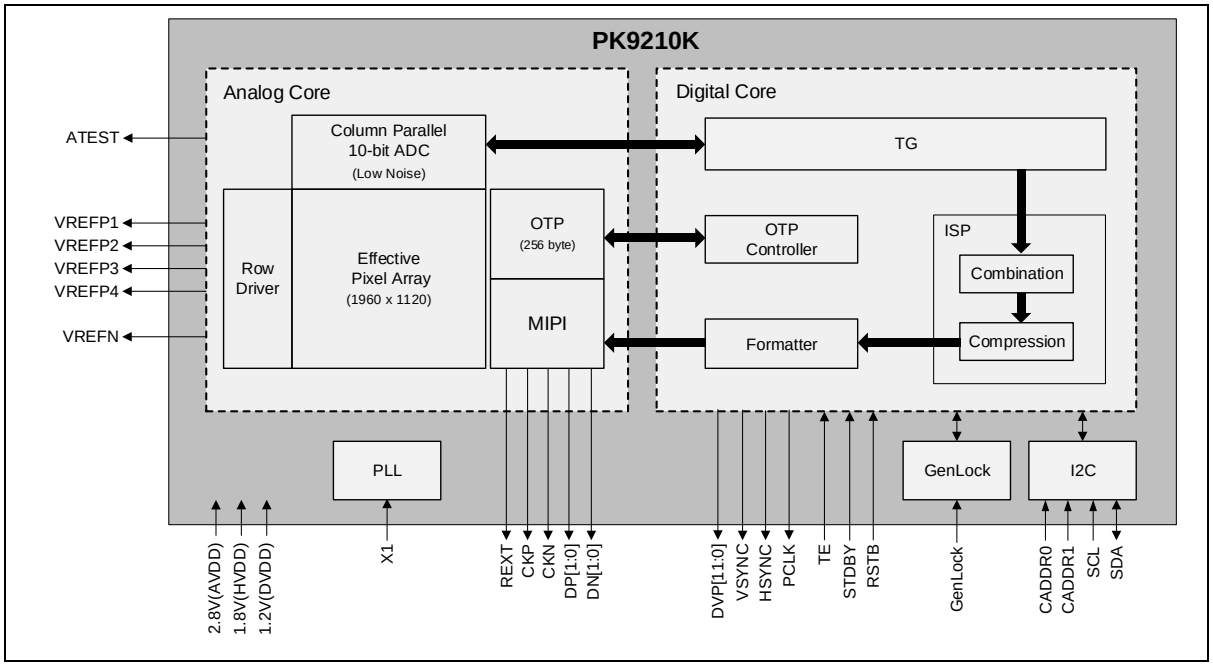
Table 3. Register Table - Soft reset

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
softreset	A	1B	[0]	0x00	RW		Soft reset 1'b0 : disable 1'b1 : enable (after succesful reset value reverts to 0)

2. Chip Architecture

Figure 5 shows the top level block diagram of the PK9210K.

Figure 5. Chip Architecture



The PK9210K consists of two functional blocks: Analog Core block and Digital Core block.

The Analog Core block consists of pixel array, Analog to Digital Converter (ADC), analog blocks such as row driver related with ADC, One Time Programmable memory (OTP), and MIPI transmitter block. The pixel array are composed of photodiodes and generate electron-hole pairs in proportion to the intensity of incoming light. The generated electron-hole pairs are converted into a voltage using a combination of charge pump and row driver and are transferred to ADC, the voltage is converted into a digital value by applying Correlation Dual Sampling (CDS) operation in ADC and ramp signal blocks, and its digital output value is transferred to digital core block.

The Digital Core consists of Timing Generation (TG), Image Signal Processing (ISP), Formatter and OTP controller blocks. The TG generates a signal for controlling the analog circuit of the Analog Core and receives a 3-image digital value (2-exposure mode & DCG) from the Analog Core. The 3-image value is transferred to the ISP after processing the Black Level Compensation (BLC).

The ISP processes the 3-image input from the TG and ISP block mainly consists of HDR Combination and Compression. The HDR Combination creates an HDR image by combining 3-image differently. The Compression compresses the amount of data that has grown with HDR Combination. The compressed HDR image is delivered to the formatter.

The formatter changes the format of the data to match the interface that outputs the HDR image. PK9210K has DVP and MIPI output interfaces, and the formatter changes the data format differently depending on DVP and MIPI.

The OTP controller may store or read data in the OTP. The OTP controller stores measured data in OTP for correct calculation of defective pixel correction (DPC). It then reads the data stored in OTP and forwards it to the DPC.

The PK9210K supports several modes that can be used to process HDR images. (refer to Table 4) Three images are acquired to generate HDR images. Two images are acquired by changing the exposure, and one image is acquired by changing the DCG.

The PK9210K supports 2-exposure and Dual Conversion Gain (DCG). The brightness of the acquired image varies depending on the time of exposure to light. 2-exposure can control the time when two images are exposed to light differently.

The DCG can use either high conversion gain (HCG) or low conversion gain (LCG). The LCG is used for high bright intensity and HCG is used for low bright intensity. In a bright environment, it can less detect light using LCG and can further express images of bright areas. In a dark environment, it can further detect light using HCG and further express images of dark areas.

Table 4 show the operation modes supported for HDR.

Table 4. HDR operation mode

Operation mode		DCG	Exposure
3HDR	L (Long)	HCG	1-exposure
	S (Short)	LCG	1-exposure
	VS (Very Short)	HCG/LCG	2-exposure

The control of internal functions and output signal timings can be enabled by modifying registers directly through a 2-wire serial interface called I2C. The generator lock (genlock) included in the PK9210K is a function that synchronizes the internal synchronous timing of the master and slave. Finally, the PK9210K supports output interfaces such as 2-lane MIPI and 12-bit DVP.

Table 5 show the operating modes supported and maximum possible frame rates in each mode.

Table 5. Operating mode

No.	Operation mode	Image Structure	Interface	Bit Width	Max. Frame rate [fps]
1	2-exposure with DCG (3HDR)	1920x1080	DVP	12-bit	30
2		1920x1080	MIPI-2	12-bit	30

3. Analog Core

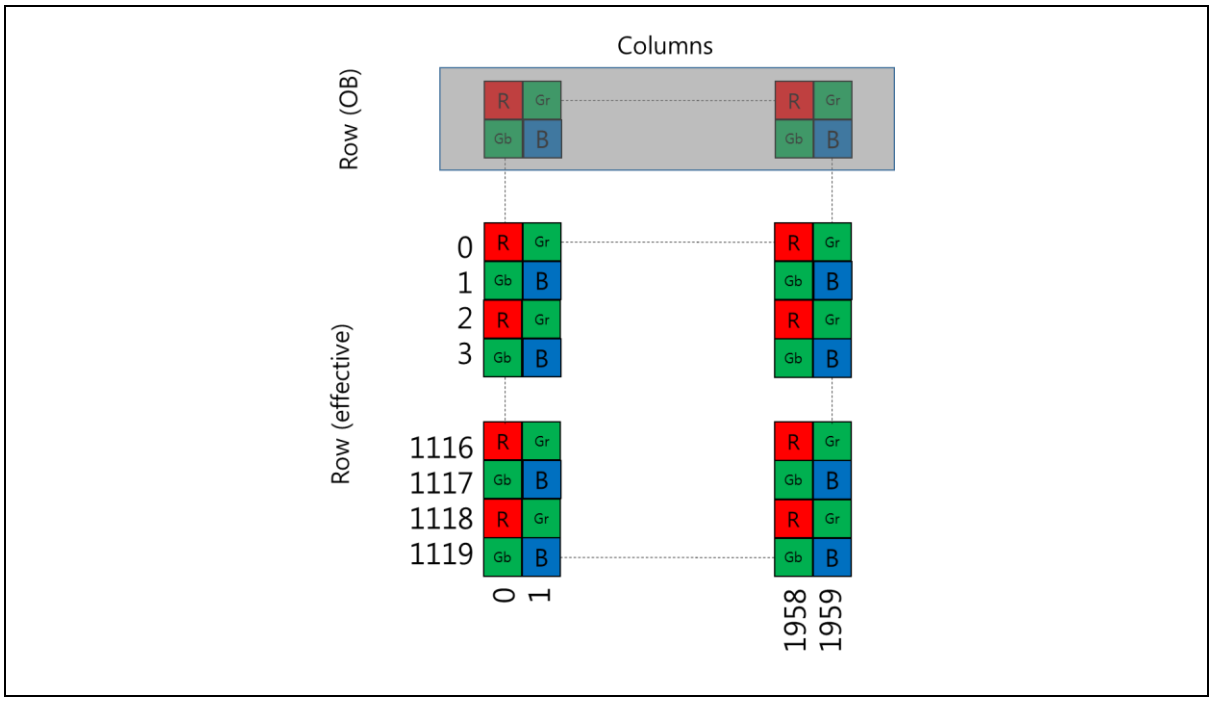
3.1. Pixel Array Structure

The PK9210K sensor has a total effective image array of 1960 columns by 1120 rows (including display image arrays of 1920x1080) covered by color filters Bayer(R, Gr, Gb and B) pattern. Figure 6 shows the pixel array color filter layout. In addition to the effective pixel, Optical block (OB) pixels area embedded to serve as reference pixels for the black level compensation (BLC). The OB Pixels are covered with a metal layer light shield.

The entire readable column is 1944 active column + 8 active border columns (interpolation pixel) + 8 dummy active border columns = 1960 columns

The entire readable row is 1104 active column + 8 active border rows (interpolation pixel) + 8 dummy active border rows = 1120 rows

Figure 6 pixel array structure



3.2. Frame Structure

The frame consists of Row Optical Black Pixel (ROBP) region, Effective Pixel, Crop Window, and Window. ROBP region is an area that does not receive light, and Effective Pixel is an area that receives light. The value of the effective pixel is adjusted based on the ROBP value, and the function is Black Level Compensation (BLC). Crop window determines the input image size for reading pixel data from the Effective Pixel. Window determines the size of the output image. (refer to Figure 7)

The size of the frame is adjustable by "framewidth" and frameheight and must be larger than ROBP region and crop window. Due to the delay in the internal line buffer, the minimum height of the blank height is 15 rows. (refer to Figure 7, Figure 8)

Pixel scanning is performed row by row on entire frame. Frame row counter and frame column counter, which are limited by “framewidth” and frameheight values respectively, are used to indicate the current coordinate of pixel being scanning. The column counter value increases by every pixel clock (pclk). Every time the column counter reaches maximum value, the row counter is increase. Origin point (0, 0) of row and column counter value is in the top left corner of each frame.

The maximum size of Crop window is 1960x1120 pixels. The user cannot change Crop window on the x-axis but can change Crop window in 2-pixel units based on the start and end points on the y-axis. If offsets of start and end point are same, Crop window satisfy image center regardless of vertical mirror. frameheight determines “fheight_a”, “rjump_top”, and “rjump_bot”. (refer to Figure 7, Table 6)

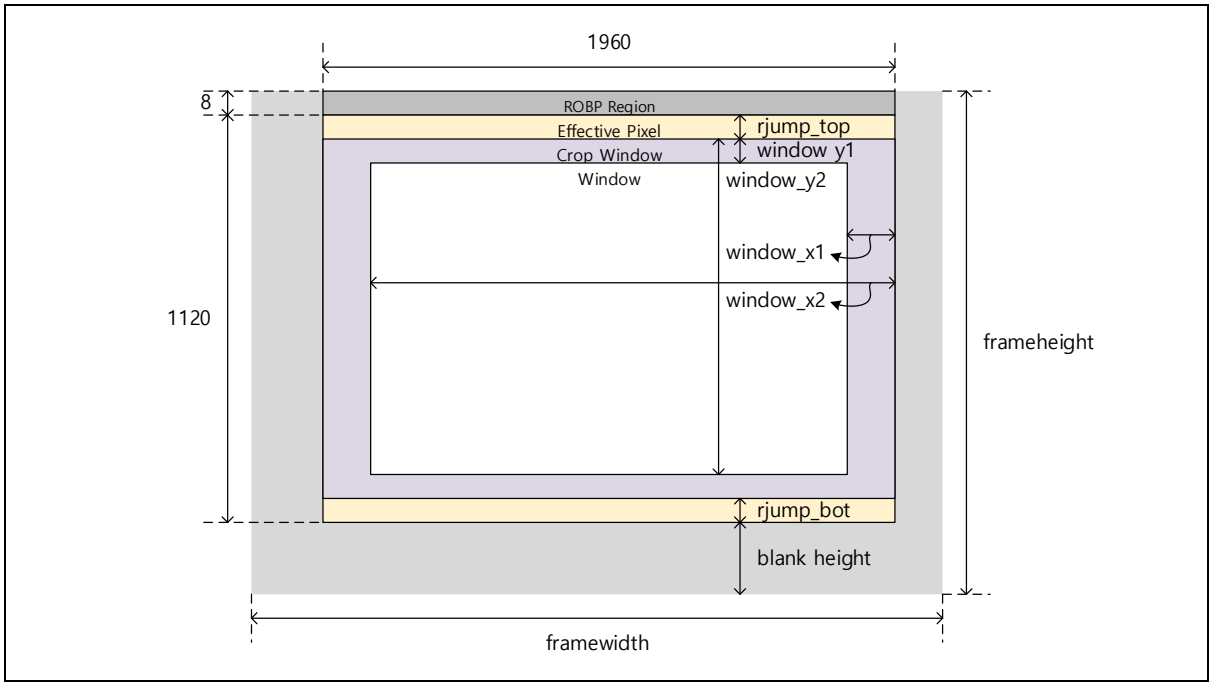
➔ frameheight = “fheigtht_a” – (“rjump_top” + “rjump_bot”)

The size of Crop window is adjusted to "rjump_top" and "rjump_bot". As the value of the "rjump_top" register increases, the starting point of Crop window increases. As the value of the "rjump_bot" register increases, the end point of Crop window decreases. (refer to Figure 7)

The maximum size of Window is determined by Crop window and is 1960x1120 pixels. The x-axis and y-axis can be changed in 2-pixel units to prevent bayer pattern order from reversing. (refer to Figure 7)

The size of Window is adjusted to "window_x1", "window_x2", "window_y1" and "window_y2". As the "window_x1" register value increases, the starting point of Window increases on the x-axis. As the "window"x2" register value increases, the end point of Window increases on the x-axis. As the value of the "window_y1" register increases, the starting point of Window increases on the y-axis. As the value of the "window_y2" register increases, the end point of Window increases on the y-axis. (refer to Figure 7).

Figure 7. Frame structure



If registers are default values, the frame structure behaves as shown below.(refer to Figure 8)

Figure 8. Default frame structure

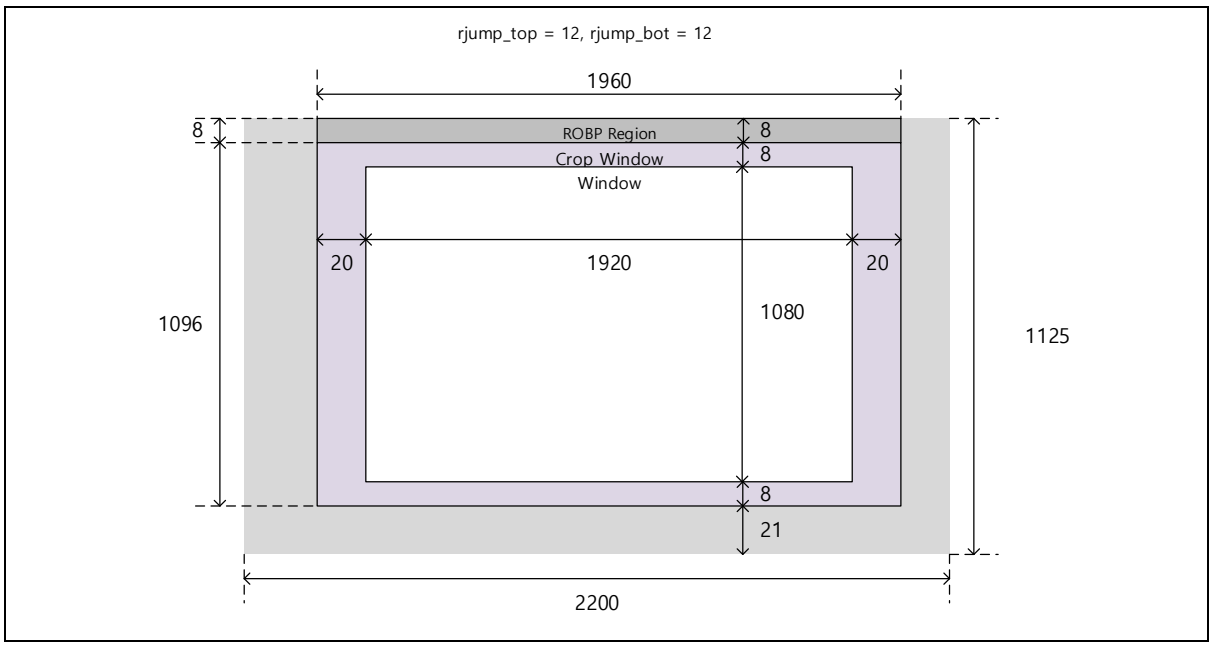


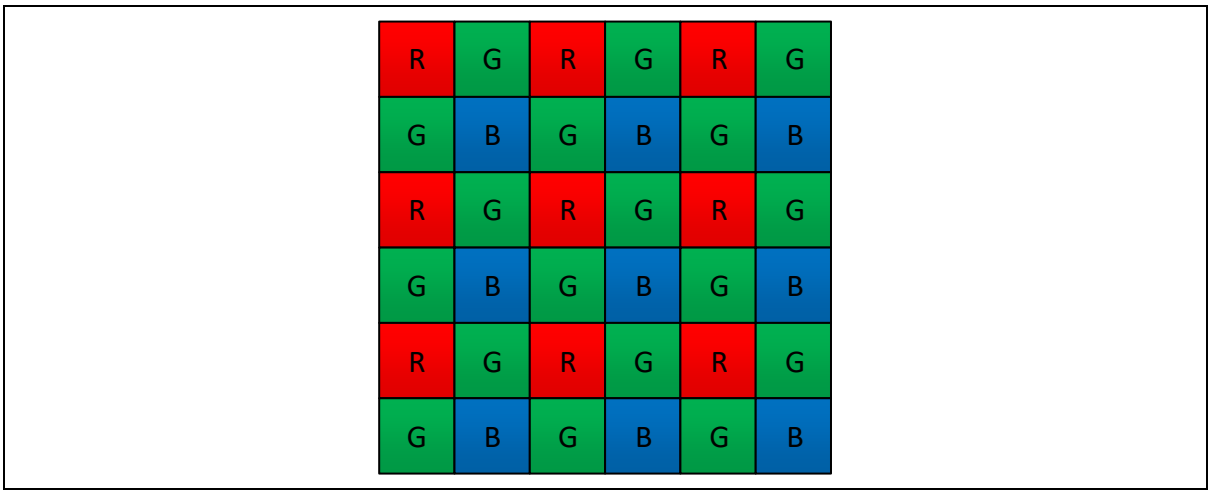
Table 6. Register Table - Frame structure

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
framewidth_h	A	06	[4:0]	0x08	RW	aev	Framewidth High Byte (must be larger than window width)

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
framewidth_l	A	07	[7:0]	0x97	RW	aev	Framewidth Low Byte (must be larger than window width)
fheight_a_h	A	08	[5:0]	0x04	RW	aev	Frameheight High Byte (must be larger than window height)
fheight_a_l	A	09	[7:0]	0x7C	RW	aev	Frameheight Low Byte (must be larger than window height)
windowx1_h	A	0C	[7:0]	0x00	RW	aev	Window horizontal start point High Byte
windowx1_l	A	0D	[7:0]	0x0C	RW	aev	Window horizontal start point Low Byte
windowy1_h	A	0E	[7:0]	0x00	RW	aev	Window vertical start point High Byte
windowy1_l	A	0F	[7:0]	0x00	RW	aev	Window vertical start point Low Byte
windowx2_h	A	10	[7:0]	0x07	RW	aev	Window horizontal end point High Byte
windowx2_l	A	11	[7:0]	0x9C	RW	aev	Window horizontal end point Low Byte
windowy2_h	A	12	[7:0]	0x04	RW	aev	Window vertical end point High Byte
windowy2_l	A	13	[7:0]	0x48	RW	aev	Window vertical end point Low Byte
rjump_top_h	B	1F	[5:0]	0x00	RW	aev	Row jump top value High Byte
rjump_top_l	B	20	[7:0]	0x0C	RW	aev	Row jump top value Low Byte
rjump_bot_h	B	21	[5:0]	0x00	RW	aev	Row jump bottom value High Byte
rjump_bot_l	B	22	[7:0]	0x0C	RW	aev	Row jump bottom value Low Byte

3.3. Pixel Data Format

Figure 9. Bayer color filter pattern data



The pixel array is covered by bayer color filters as shown in the Figure 9. Since each pixel can have only one type of filter on it, only one-color component can be produced by a pixel. PK9210K provides RGB bayer pattern data through a 10-bit channel which passes one-pixel data to the output bus at every pclk.

The PK9210K provides horizontal, vertical mirror which respectively reverse the sensor data readout order horizontally and vertically. The user can change “mirror” to change readout order. (refer to Table 7)

Figure 10 shows a normal image and a mirrored image.

Figure 10. Mirror

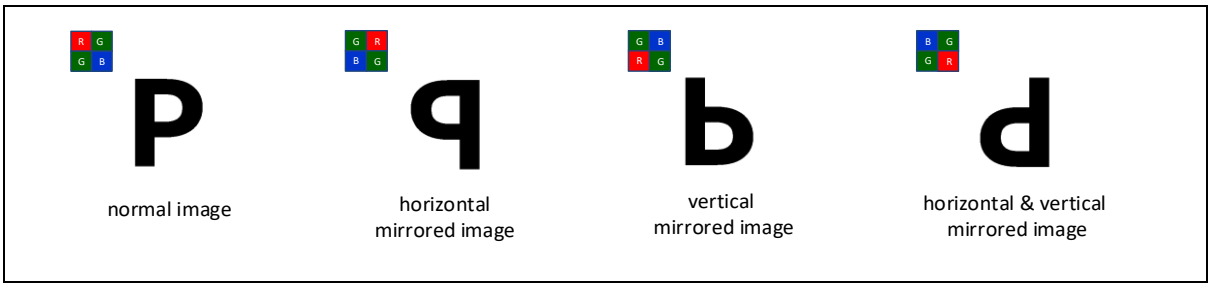


Table 7. Register Table - Mirror

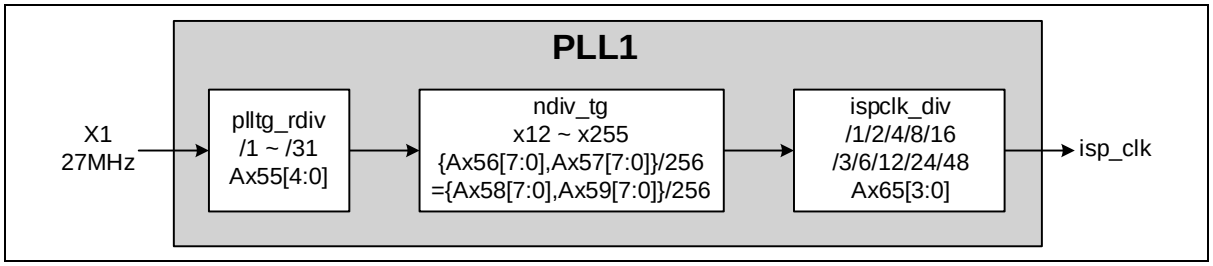
Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
mirror	A	05	[1:0]	0x00	RW	aev	Image Inversion mirror [1]: vertical inversion mirror [0]: horizontal inversion

3.4. Clock

The PK9210K implements two PLLs with both inputs connected to the X1 pin. One can support the ISP clock, while the other one can support MIPI clock.

The PLL1 generates clock for ISP. The VCO range is from 445MHz to 594MHz. A programmable clock is provided to generate different frequencies.

Figure 11. PLL1 Diagram



To overcome the EMI issue, spread spectrum clocking (SSC) is supported in this PLL1 to distribute the power on the ISP clock signal. Spread profile is a triangular shape. The spectrum can be up-spread, center-spread, or down-spread depending on the f_{SSC_UPPER} and f_{SSC_LOWER} frequency setting. $f_{SSC\_MOD\_RATE}$ is the modulation frequency, which is normally designed to be 30 kHz~33 kHz in many designs. f_{SSC_DEV} is modulation amplitude, which is smaller than 5000 ppm. SSC parameters calculation are given by following equations:

$$f_{SSC\_UPPER} = f_{REF} / plltg\_rdiv * ndiv\_tg\_upper * (1 / ispclk\_div)$$

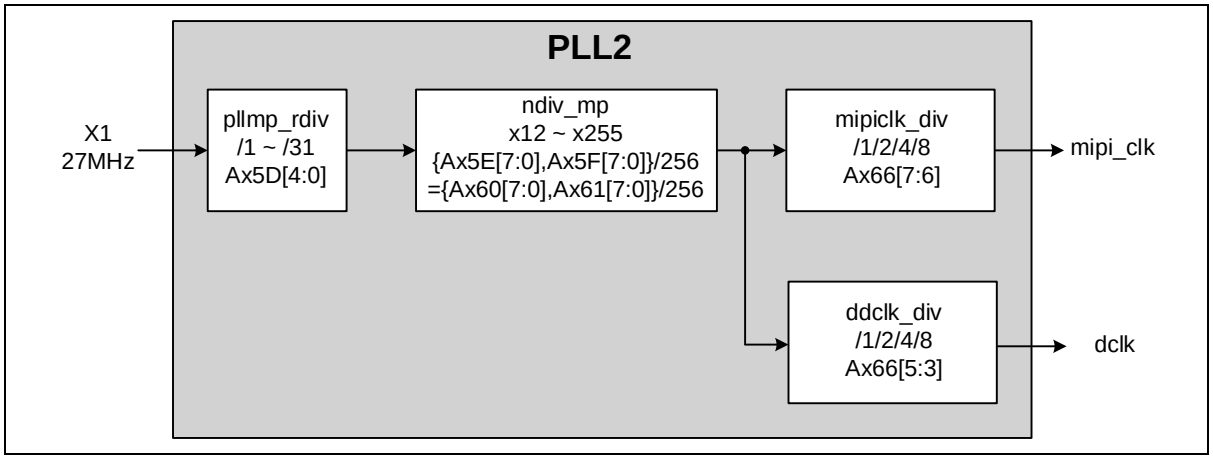
$$f_{SSC\_LOWER} = f_{REF} / plltg\_rdiv * ndiv\_tg\_lower * (1 / ispclk\_div)$$

$$plltg\_dsm\_ystep = 256*(ndiv\_tg\_upper – ndiv\_tg\_lower)/8$$

$$plltg\_dsm\_xstep = f_{REF}/ plltg\_rdiv * 1/(2*f_{SSC\_MOD\_RATE}) * (1/ plltg\_dsm\_ystep)$$

The PLL2 generates clock for MIPI. The VCO range is from 445MHz to 594MHz. A programmable clock is provided to generate different frequencies.

Figure 12. PLL2 Diagram



3.4.1. PLL

Table 8 Register Table – PLL

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
plltg_pd	A	4F	[5]	1'b1	RW		PLL1 power down mode 1'b0 : pll1 power on 1'b1 : pll1 power down
reg_pll_bypass	A	4F	[4]	1'b1	RW		PLL bypass 1'b0 : use pll mode 1'b1 : pll bypass mode
plltg_rdiv	A	55	[4:0]	0x01	RW		PLL1 Ref Freq. Divider 00000b : unused 00001b : Reference Clock/ 1 ... 11111b : Reference Clock/ 31
plltg_dsm_divh_h	A	56	[7:0]	0x16	RW		{divh_h,divh_l} = NDIV_TG * 256
plltg_dsm_divh_l	A	57	[7:0]	0x00	RW		
plltg_dsm_divl_h	A	58	[7:0]	0x16	RW		{divl_h, divl_l} = NDIV_TG * 256
plltg_dsm_divl_l	A	59	[7:0]	0x00	RW		
pllmp_pd	A	4F	[3]	1'b1	RW		PLL2 power down mode 1'b0 : pll1 power on 1'b1 : pll1 power down
pllmp_rdiv	A	5D	[4:0]	0x01	RW		PLL2 Ref Freq. Divider 00000b : unused 00001b : Reference Clock/ 1 ... 11111b : Reference Clock/ 31
pllmp_dsm_divh_h	A	5E	[7:0]	0x16	RW		{divh_h,divh_l} = NDIV_MP_UPPER * 256
pllmp_dsm_divh_l	A	5F	[7:0]	0x00	RW		
pllmp_dsm_divl_h	A	60	[7:0]	0x16	RW		{divl_h, divl_l} = NDIV_MP_LOWER * 256
pllmp_dsm_divl_l	A	61	[7:0]	0x00	RW		
plltg_dsm_xstep	A	5A	[7:0]	0x00	RW		SSC parameters XSTEP
plltg_dsm_ystep	A	5B	[7:0]	0x00	RW		SSC parameters YSTEP

3.4.2. PLL Clock Divider

Table 9. Register Table – PLL Clock divider

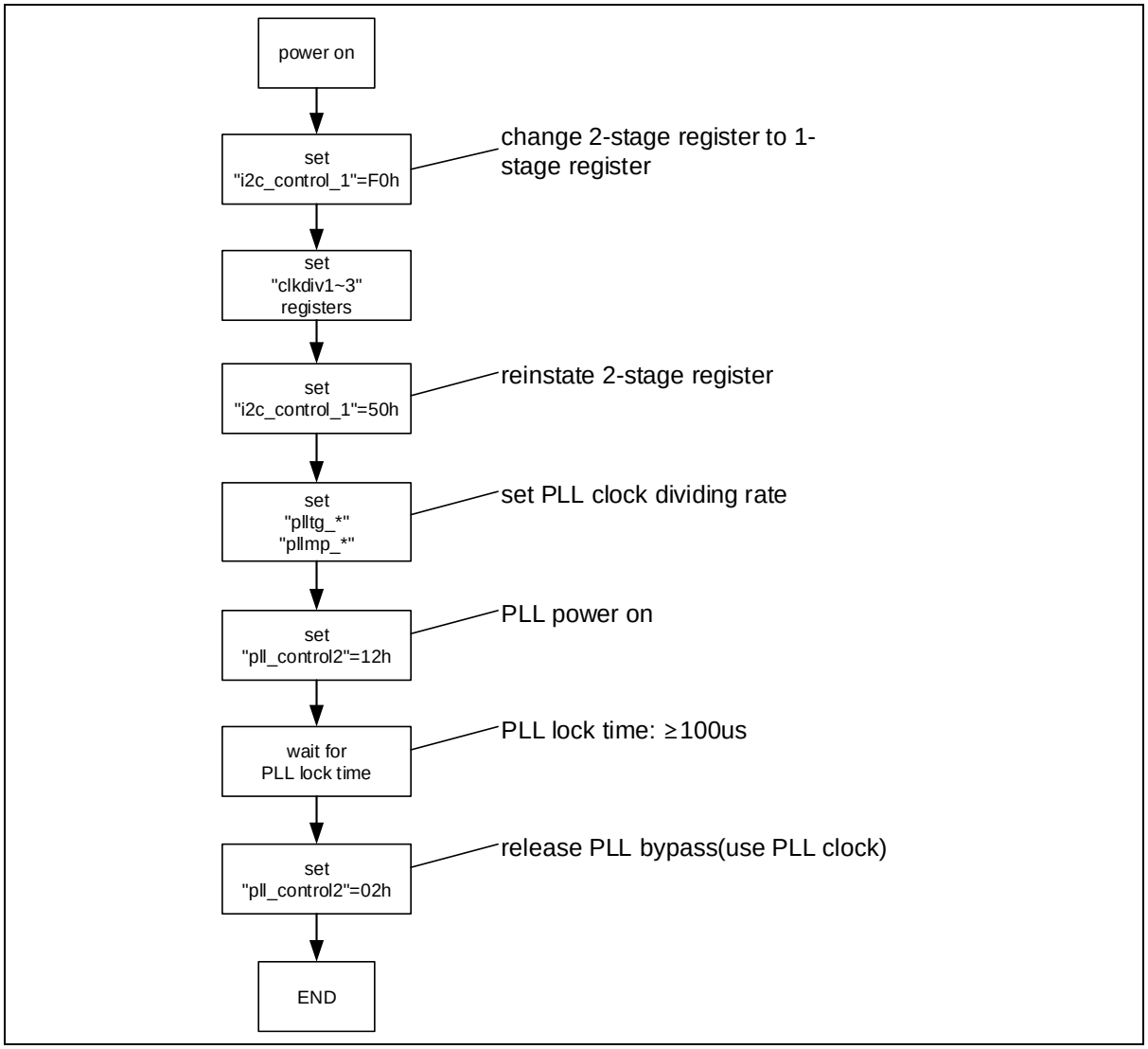
Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
ispclk_div	A	65	[3:0]	4'b0011	RW	aev	ISP clock divider 4'b0000 = 1/1 4'b0001 = 1/2 4'b0010 = 1/4 4'b0011 = 1/8 4'b0100 = 1/16 4'b1000 = 1/3 4'b0001 = 1/6 4'b0010 = 1/12 4'b0011 = 1/24 4'b0100 = 1/48
mipiclk_div	A	66	[7:6]	2'b00	RW	aev	MIPI clock divider mipi_clk =vco2/(2^mipiclk_div)
ddclk_div	A	66	[5:3]	3'b011	RW	aev	MIPI byte clock divider ddclk =vco2/[2^ddclk_div]

3.4.3. PLL and Clock Setting Sequence

When using PLL, set-up sequence, show Figure 13, is necessary.

I2C update timing register, “i2c_control_1”, is changed before setting clock dividers to immediately apply clock divider settings.

Figure 13 Clock setting sequence



4. Digital core

The digital core consists of Timing Generator (TG), Image Signal Processing (ISP), Formatter, and OTP controllers. This chapter describes TG, ISP, and Formatter. For OTP controllers, see the chapter OTP memory.

4.1. Timing Generator

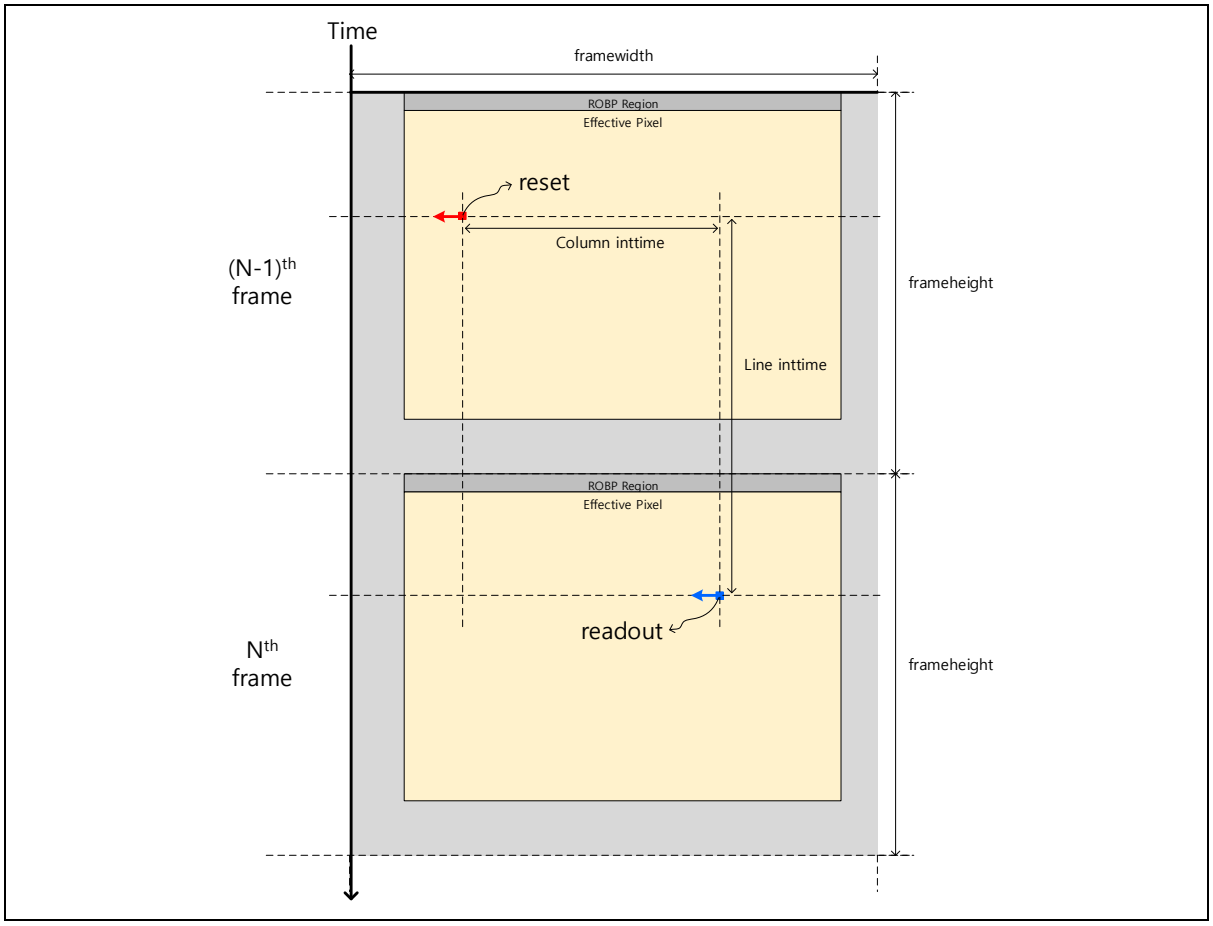
The TG generates signals that control analog circuits in the Analog core, and can control exposure time and global gain. The data received from the Analog core is processed with functions such as Black Level Calibration (BLC), Digital Gain, and White Balance Gain. And the data can be replaced with test pattern data. It has a Genlock function that synchronizes data timing with the master device.

4.1.1. Exposure Control

The PK9210K may adjust the brightness of an image by integration time, global gain, and digital gain.

4.1.1.1. Integration Time

Figure 14. Integration time structure in PK9210K



PK9210K employs rolling shutter for capturing image. Rolling shutter operate read process after reset process in row sequence. (refer to Figure 14)

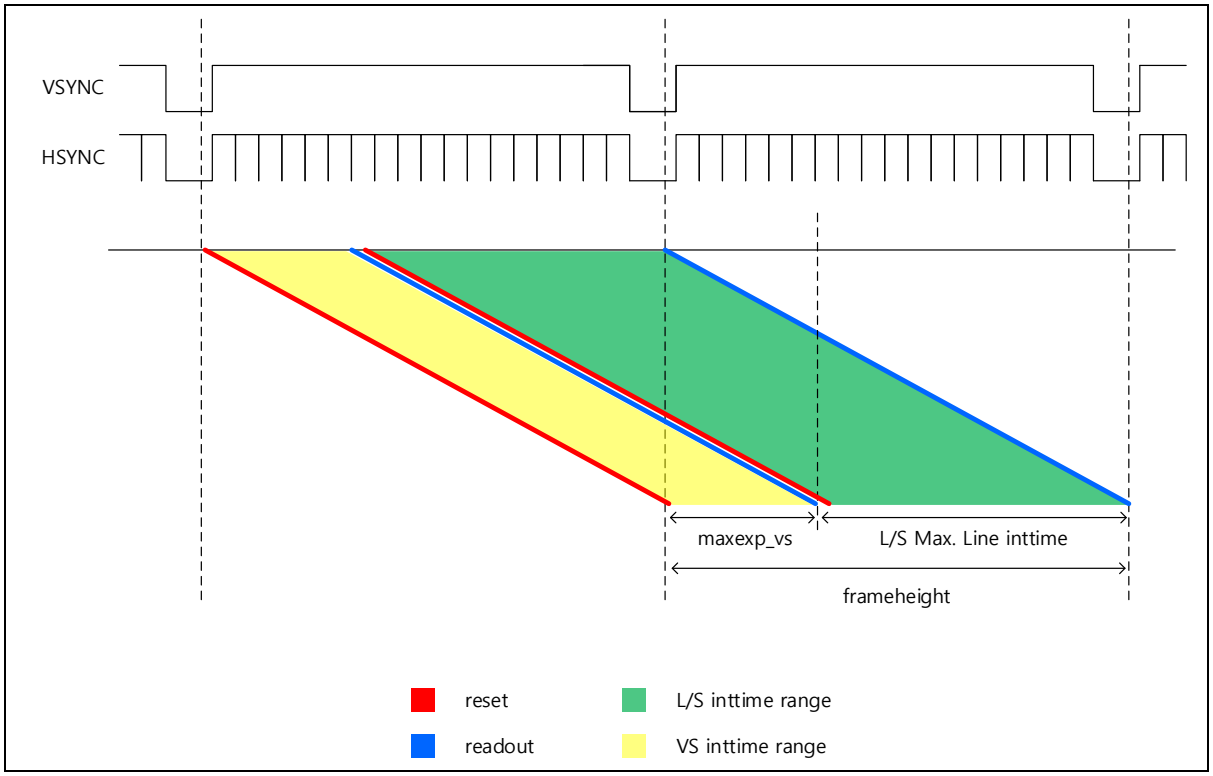
Reset process initializes ROBP region and Effective pixel in row sequence row by row. After reset process, Effective pixel are exposured to light. ROBP region aren't exposured to light. In sequence, readout process reads data of ROBP and Effective pixel. Readout process is identical at order and speed as reset process.

The difference in time between reset and readout process is known as integration time (inttime). The inttime controls time of effective pixel exposured to light. inttime can be adjusted in line unit level (line inttime) and column unit level (column inttime).

The PK9210K supports three modes of operation and achieves four different brightness images. The operation mode consists of L, S, VS.

Figure 15 shows the intime area operating according to L, S, and VS. Reset and readout operate in order based on VSYNC and HSYNC generated for each frame. It is a staggered exposure time, and L/S and VS have a difference in the inttime. The VS intime range is adjusted to the "maxexp_vs" value. The L/S intime range is an area excluding "maxexp_vs" from frameheight.

Figure 15. Fundamental concept of inttime



Depending on the operating mode, the inttime is adjusted to "inttime" and "inttime_vs". (refer to Table 10)

Table 10. Integration time range

Operation mode	Register name	Line inttime (hex)		Column inttime (hex)	
		max	min	max	Min
L / S	inttime	0455[a]	0000	FF	00
VS	inttime_vs	0008[b]	0000	FF	00

a. max line inttime = frameheight – “maxexp_vs” – 5
b. max line inttime = “maxexp_vs” – 2

Table 11 shows registers relevant to integration time.

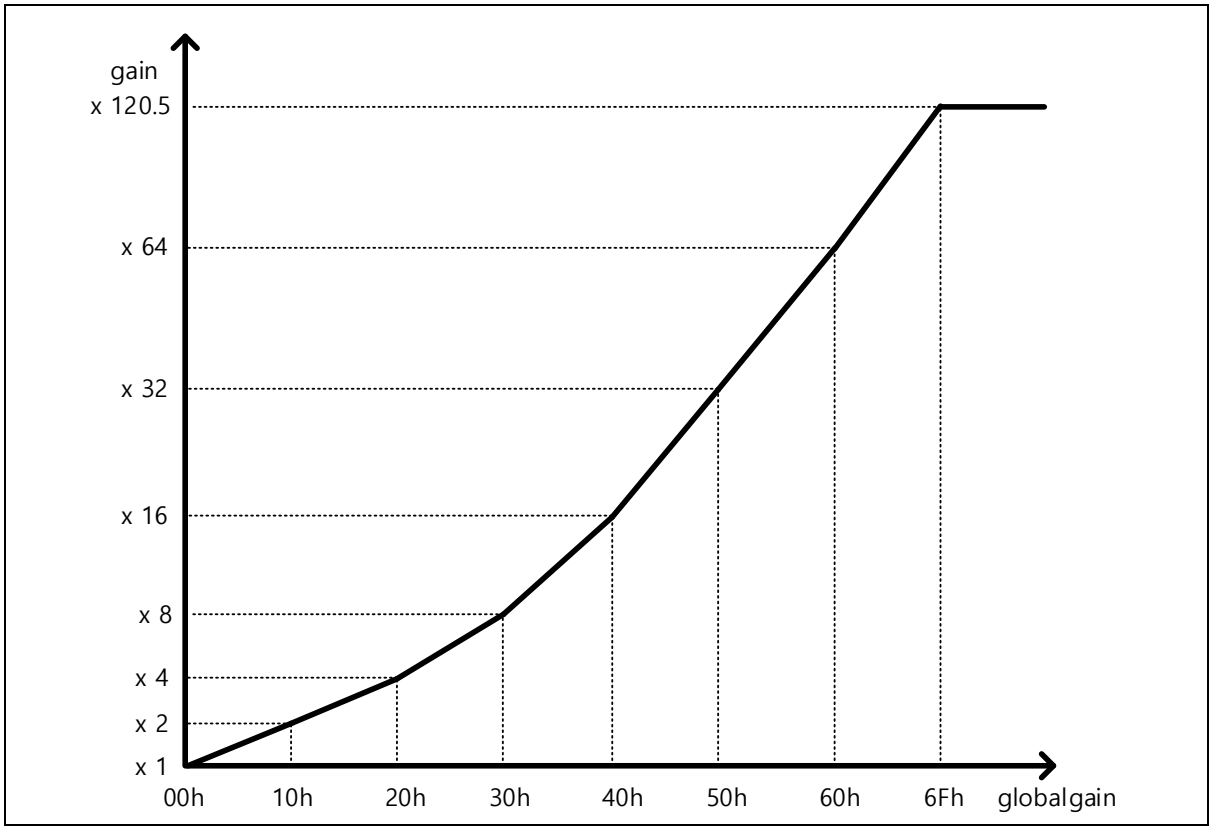
Table 11. Register Table - Integration time

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
maxexp_vs	B	60	[7:0]	0x0A	RW		Maximum exposure of very short image
inttime_h	B	6B	[7:0]	0x01	RW	wr_en	Long Line inttime High Byte
inttime_m	B	6C	[7:0]	0x40	RW	wr_en	Long Line inttime Low Byte
inttime_l	B	6D	[7:0]	0x00	RW	wr_en	Long Column inttime
inttime_vs_h	B	6E	[7:0]	0x00	RW	wr_en	Very short Line inttime High Byte
inttime_vs_m	B	6F	[7:0]	0x02	RW	wr_en	Very short Line inttime Low Byte
inttime_vs_l	B	70	[7:0]	0x00	RW	wr_en	Very shrot Column inttime

4.1.1.2. Globalgain

The globalgain is a register for controlling the real gain. It consist of 7bits and control the gain from 1x to 60.2x. MSB 3 bits double gain per code, and LSB 4bits control fine gain from 1x to 1.9x in 16 steps. Figure 16 shows the gain curve of global gain.

Figure 16. Globalgain's gain



Globalgain is divided into three registers in Table 12. "globalgain" controls long and short together, "globalgain_vs" controls very short.

Table 12. Register Table - Globalgain

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
globalgain	B	74	[7:0]	0x00	RW	wr_en	Analog gain 0
globalgain_vs	B	75	[7:0]	0x00	RW	wr_en	Analog gain 1

4.1.1.3. Digital Gain

Analog signal is converted to digital value through ADC operation, and the digital value can be amplified by “digitalgain”. “digitalgain” register's upper 4 bits are positive integer and lower 4 bits are fraction.

Table 13 shows registers relevant to ″digitalgain”.

Table 13. Register Table - Digital gain

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
digitalgain_l	B	78	[7:0]	0x10	RW	wr_en	Digital gain of long data
digitalgain_s	B	79	[7:0]	0x10	RW	wr_en	Digital gain of short data
digitalgain_vs	B	7A	[7:0]	0x10	RW	wr_en	Digital gain of very short data

4.1.1.4. Exposure factor update control

Depending on "wr_en_off", exposure related registers (integration time, “globalgain”, and “digitalgain”) are updated immediately or when "wr_en" is 1'b1. If "wr_en" sets to 1'b1, "wr_en" is changed to 1'b0 automatically.

Due to exposure related registers split across several registers, if they are changed over several frames, the brightness of the screen changes for each frame, which causes hunting. If exposure related registers are updated at once after writing all register, the brightness of the screen is not changed many times.

Table 14 shows registers relevant to exposure register update

Table 14. Register Table - Exposure register update

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
wr_en	B	9A	[0]	0x00	RW		Update exposure related register 1'b0 : no update 1'b1 : wr_en set
wr_en_off	B	9B	[0]	0x00	RW		Update exposure related register 1'b0 : update @ wr_en = 1'b1 1'b1 : Immediately update

4.1.2. Test Pattern (TP) Control

TP control generates test images from TG block. Test images type can be selected by setting “tp_control_0” registers. In case of test image types from 0x15 to 0x1A values for “tp_control_0”, “tp_control_1/2/3/4” registers are used as color values and the following rule shows how the color value is determined:

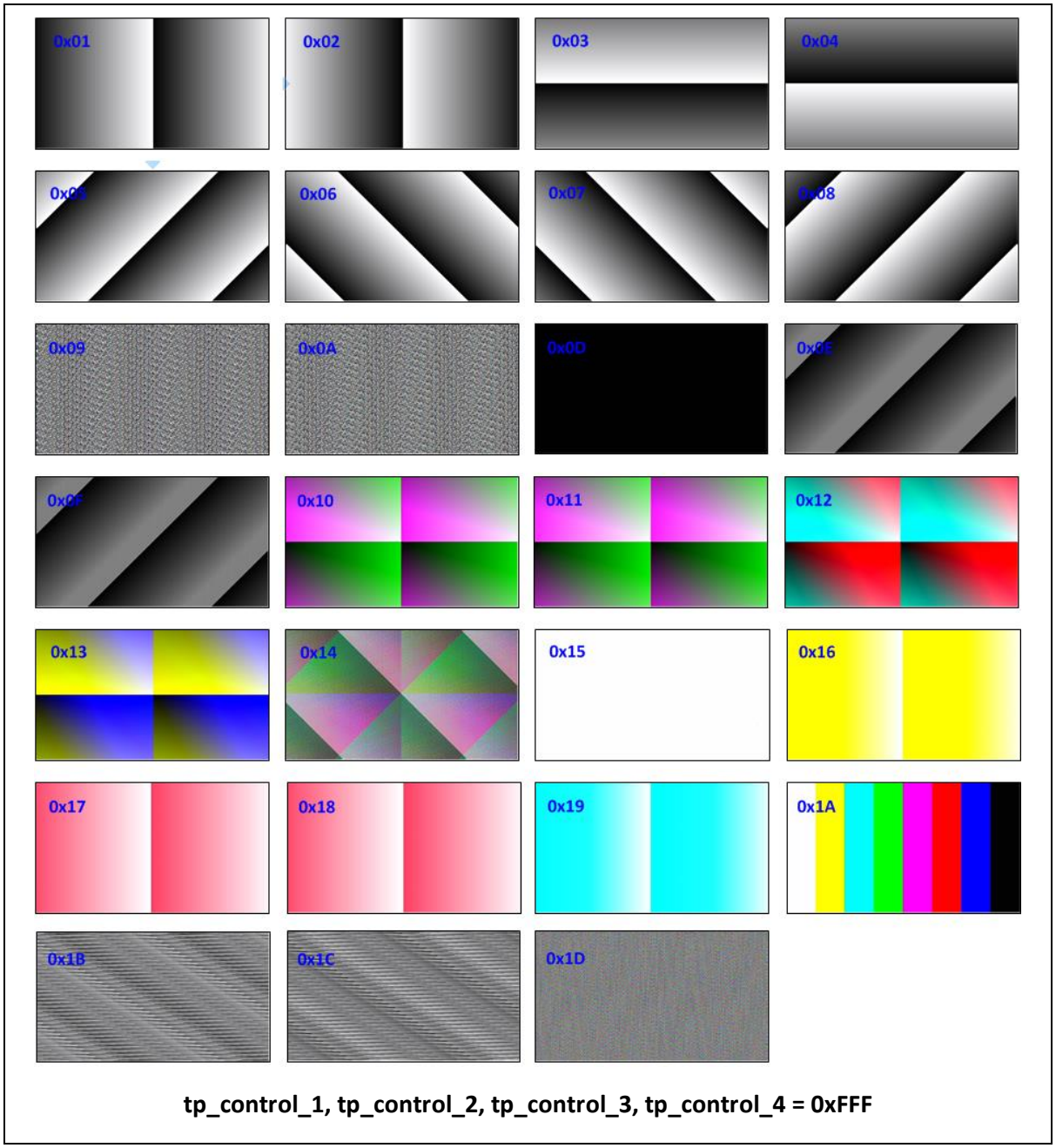
- R: {tp_control_1_h [1:0], tp_control_1_l}
- Gr: {tp_control_2_h [1:0], tp_control_2_l}
- Gb: {tp_control_3_h [1:0], tp_control_3_l}
- B: {tp_control_4_h [1:0], tp_control_4_l}

Table 15 shows registers relevant to Test Pattern control.

Table 15. Register Table - Test pattern control

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
tp_control_0	B	A2	[7:0]	0x00	RW		Test pattern selection
tp_control_1_h	B	A3	[1:0]	0x00	RW		R color for test pattern High Byte
tp_control_1_l	B	A4	[7:0]	0x00	RW		R color for test pattern Low Byte
tp_control_2_h	B	A5	[1:0]	0x00	RW		G1 color for test pattern High Byte
tp_control_2_l	B	A6	[7:0]	0x00	RW		G1 color for test pattern Low Byte
tp_control_3_h	B	A7	[1:0]	0x00	RW		G2 color for test pattern High Byte
tp_control_3_l	B	A8	[7:0]	0x00	RW		G2 color for test pattern Low Byte
tp_control_4_h	B	A9	[1:0]	0x00	RW		B color for test pattern High Byte
tp_control_4_l	B	AA	[7:0]	0x00	RW		B color for test pattern Low Byte

Figure 17. Test image



4.1.3. White Balance (WB) gain

The WB gain is functioning that process image in according to color of light source.

- R gain[8:0] = {wb_rgain_h[0], wb_rgain_l[7:0]}
- G gain[8:0] = {wb_ggain_h[0], wb_ggain_l[7:0]}
- B gain[8:0] = {wb_bgain_h[0], wb_bgain_l[7:0]}

The user can change registers to control WB gain. The sequence of bayer pattern is R[0], Gr[1], Gb[2], and B[3]. The WB gain of each channel consists of R, G, and B channel with 9-bit (3i.6f). In the WB gain of R channel, LSB 1-bit of "wb_rgain_h" and MSB 2-bit of "wb_rgain_l" are integer, and LSB 6-bit of "wb_rgain_l" are fraction. If WB gain of R channel is x1, "wb_rgain" is 0x0040. The operation of G and B channel are same such as R channel.

Table 16 shows registers relevant to WB gain control.

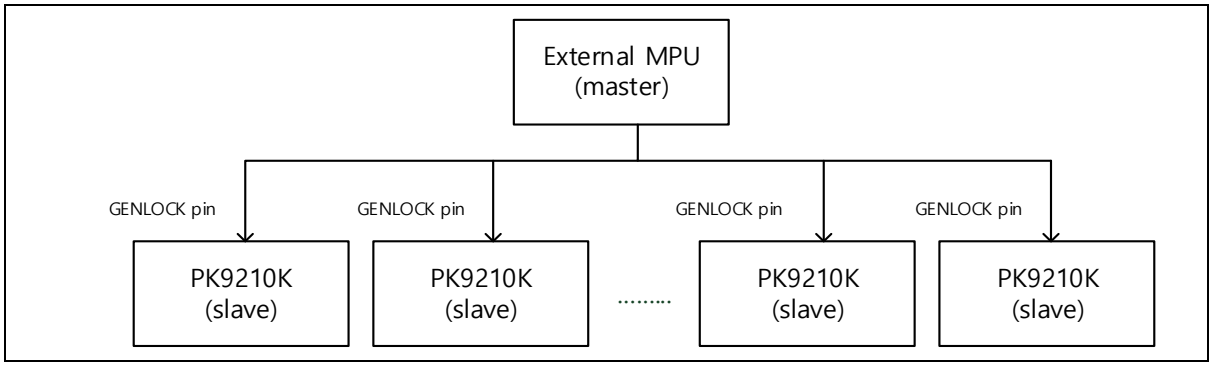
Table 16. Register Table – WB gain control

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
wb_en	B	06	[7]	1'b0	RW		White balance enable control 1'b0 : disable 1'b1 : enable
wb_rgain_l_h	B	81	[0]	0x00	RW		White balance "R" gain for long High Byte
wb_rgain_l_l	B	82	[7:0]	0x40	RW		White balance "R" gain for long Low Byte
wb_ggain_l_h	B	83	[0]	0x00	RW		White balance "G" gain for long High Byte
wb_ggain_l_l	B	84	[7:0]	0x40	RW		White balance "G" gain for long Low Byte
wb_bgain_l_h	B	85	[0]	0x00	RW		White balance "B" gain for long High Byte
wb_bgain_l_l	B	86	[7:0]	0x40	RW		White balance "B" gain for long Low Byte
wb_rgain_s_h	B	87	[0]	0x00	RW		White balance "R" gain for short High Byte
wb_rgain_s_l	B	88	[7:0]	0x40	RW		White balance "R" gain for short Low Byte
wb_ggain_s_h	B	89	[0]	0x00	RW		White balance "G" gain for short High Byte
wb_ggain_s_l	B	8A	[7:0]	0x40	RW		White balance "G" gain for short Low Byte
wb_bgain_s_h	B	8B	[0]	0x00	RW		White balance "B" gain for short High Byte
wb_bgain_s_l	B	8C	[7:0]	0x40	RW		White balance "B" gain for short Low Byte
wb_rgain_vs_h	B	8D	[0]	0x00	RW		White balance "R" gain for very short 1 High Byte
wb_rgain_vs_l	B	8E	[7:0]	0x40	RW		White balance "R" gain for very short 1 Low Byte
wb_ggain_vs_h	B	8F	[0]	0x00	RW		White balance "G" gain for very short 1 High Byte
wb_ggain_vs_l	B	90	[7:0]	0x40	RW		White balance "G" gain for very short 1 Low Byte
wb_bgain_vs_h	B	91	[0]	0x00	RW		White balance "B" gain for very short 1 High Byte
wb_bgain_vs_l	B	92	[7:0]	0x40	RW		White balance "B" gain for very short 1 Low Byte

4.1.4. Genlock

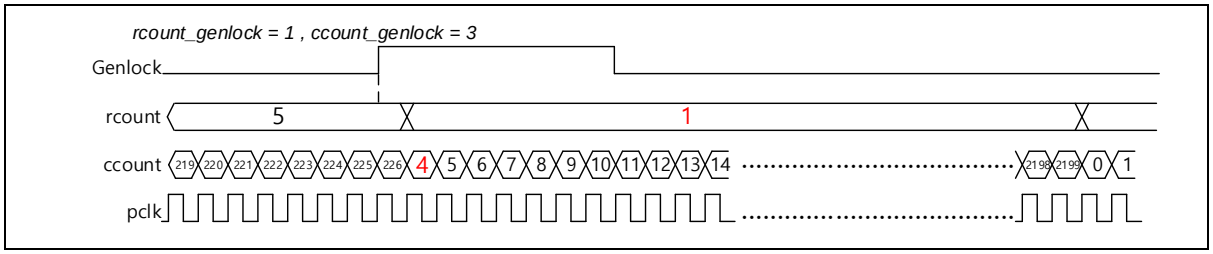
Generator locking (Genlock) function synchronizes internal synchronous timing of master device and slave device.

Figure 18. Genlock Sync configuration with MPU



When PK9210K receives genlock signal from the master device through GENLOCK PAD, internal row counter and column counter are initialized as “rcount_genlock” and “ccount_genlock” value at the rising edge of genlock signal. Pulse width of genlock signal requires at least 4 pclk period to enable reliable genlock operation.

Figure 19. Genlock timing



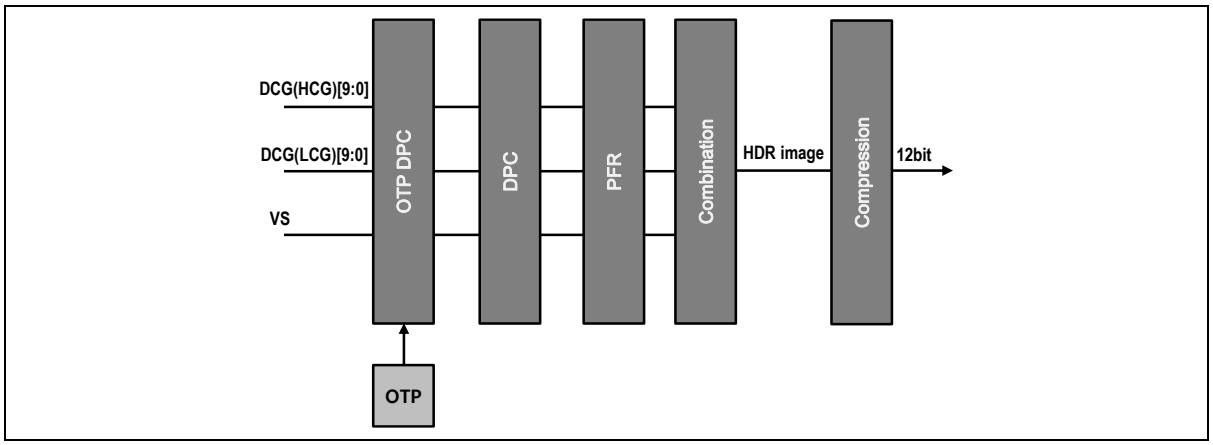
Note If the synchronization timing difference between master and slave is large, sudden shift in brightness may occur in slave device's image.

Table 17. Genlock control

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
genlock_pad_en	A	20	[0]	1'b0	RW		GENLOCK pad enable 1'b0 : disable 1'b1 : enable
genlock_en	B	0A	[3]	1'b0	RW		GENLOCK enable 1'b0 : disable 1'b1 : enable
rcount_genlock_h	B	57	[5:0]	0x00	RW		Genlock row count High Byte
rcount_genlock_l	B	58	[7:0]	0x01	RW		Genlock row count Low Byte
ccount_genlock_h	B	59	[4:0]	0x00	RW		Genlock column count High Byte
ccount_genlock_l	B	5A	[7:0]	0x01	RW		Genlock column count Low Byte

4.2. Image Signal Processing

Figure 20 ISP data path



ISP contains OTP DPC, DPC (Defect Pixel Correction), HDR combine, and Compress functions. First, remove the defect pixels and reduce the noisy pixels by DPC (including OTP DPC). After that, the PFR (Purple Fringing Reduction) function relieves the purple fringing phenomenon caused by chromatic aberration. According to the setting for 3 images, The HDR Combine makes a HDR image from 1-exposure (DCG(HCG+LCG)) or 2-exposure (DCG(HCG+LCG) + VS) images . In order to reduce the amount of data transmission, HDR combine data is compressed into 12 bits data.

4.2.1. OTP DPC

OTP memory has space to store the coordinates of 64 pixel defects. In general, when there is only a single defect, it is well corrected even without OTP DPC, but when there is a couple defect or defects caused by single defect + noise, it is difficult to correct only with normal DPC. The defect of the neighboring pixels is transmitted to the DPC block so that the correction works well if the center pixel is defective. Table 18 shows registers for OTP DPC control.

Table 18. Register Table – OTP DPC

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
otp_dpc_en_l	I	08	[7]	1'b0	RW		OTP DPC enable control 1'b0 : disable 1'b1 : enable
otp_dpc_en_s	I	08	[6]	1'b0	RW		
otp_dpc_en_vs	I	08	[5]	1'b0	RW		

4.2.2. DPC

Including OTP DPC information, it corrects dynamic defects caused by noise. Calculate the center-pixel(Pc) and peri-pixel deviation(N1v~N8v), compare it with the DPC threshold, and interpolate the center-pixel. WDPC is enabled when “wdpc_en” is set to 1’b1, and BDPC is enabled when “bdpc_en” is set to 1’b1. The higher the “dark_dpc_p” and “dark_dpc_n” are set, the greater the DPC effect. The smaller “dpc_offset” is set, the greater the DPC effect. “dpc_offset2” is used when the value of “dark_dpc_p/n” is 64d or less, and the larger the setting, the smaller the DPC effect. “dpc_offset2” must be set to 0x20 or higher.

Figure 21. DPC control

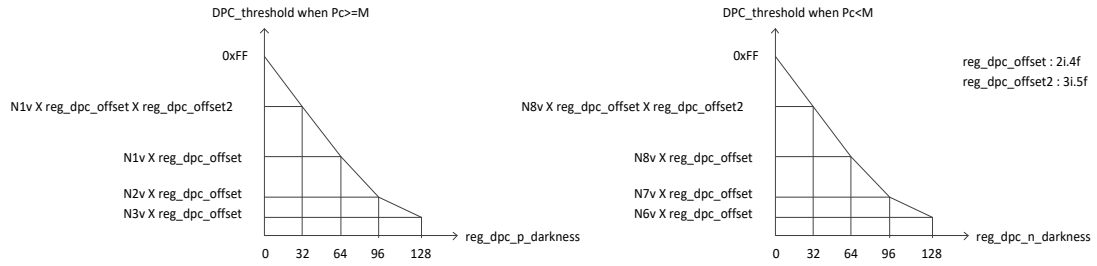


Table 19. Register Table – DPC

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
wdpc_en_l	I	04	[7]	1'b1	RW		DPC enable control 1'b0 : disable 1'b1 : enable
bdpc_en_l	I	04	[6]	1'b1	RO		
wdpc_en_s	I	04	[5]	1'b1	RO		
bdpc_en_s	I	04	[4]	1'b1	RO		
wdpc_en_vs	I	04	[3]	1'b1	RO		
bdpc_en_vs	I	04	[2]	1'b1	RO		
dpc_offset_l	I	0B	[5:0]	0x10	RW		DPC offset1 control
dpc_offset2_l	I	0C	[7:0]	0x20	RW		DPC offset2 control
dpc_offset_s	I	1B	[5:0]	0x10	RW		DPC offset1 control
dpc_offset2_s	I	1C	[7:0]	0x20	RW		DPC offset2 control
dpc_offset_vs	I	2B	[5:0]	0x10	RW		DPC offset1 control
dpc_offset2_vs	I	2C	[7:0]	0x20	RW		DPC offset2 control
dpc_dark_p_l	I	0D	[7:0]	0x40	RO		DPC strength 0x40 : 1-pixel correction 0x60 : 2-pixel correction 0x7F : 3-pixel correction
dpc_dark_n_l	I	0E	[6:0]	0x40	RO		
dpc_dark_p_s	I	1D	[7:0]	0x40	RO		
dpc_dark_n_s	I	1E	[6:0]	0x40	RO		
dpc_dark_p_vs	I	2D	[7:0]	0x40	RO		
dpc_dark_n_vs	I	2E	[6:0]	0x40	RO		

4.2.3. PFR

Due to the chromatic aberration caused by the difference in R, G, and B wavelength lengths, purple fringing occurs where R and B colors are seen around a white object (especially a light source). PFR calculates the gradient of purple fringing characteristics, and the PFR intensity can be adjusted through “fading” and “slope” control. As the “fading” value decreases and the “slope” value increases, the PFR effect increases. PFR enable can be used for each of R and B

channels. Figure 22 shows PFR intensity control according to PFR “fading” settings. Figure 23 shows the PFR intensity control according to the PFR “slope” setting. Table 20 shows the PFR control register.

Figure 22. PFR fading control

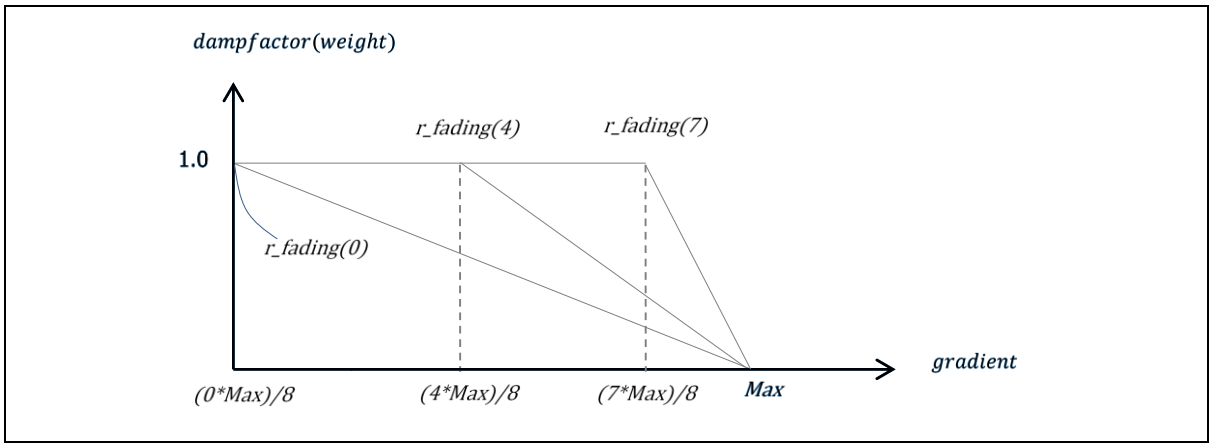


Figure 23. PFR slope control

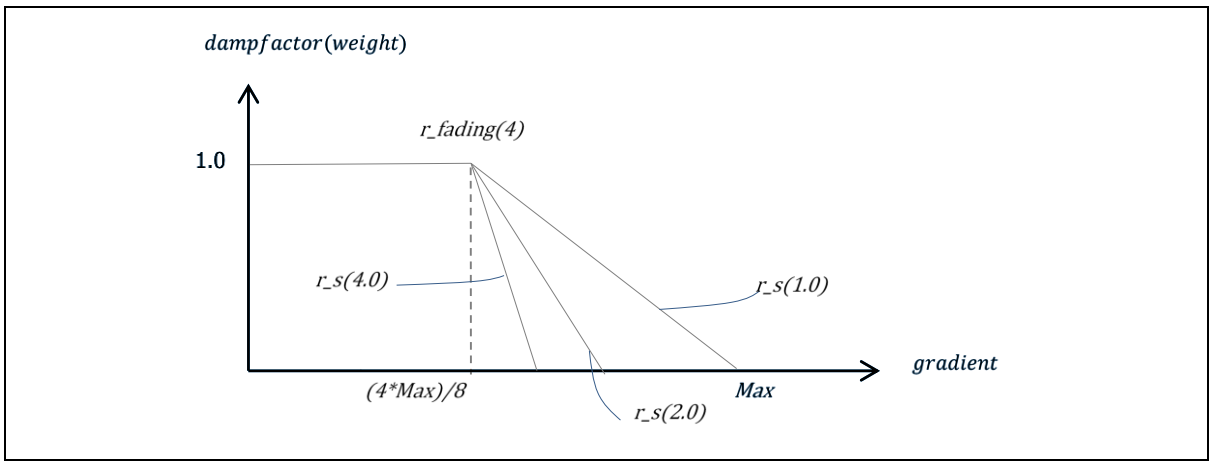


Table 20. Register Table – PFR

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
ren_l	J	31	[1]	1'b0	RW		PFR enable control 1'b0 : disable 1'b1 : enable
ben_l	J	31	[0]	1'b0	RW		
fading_l	J	32	[7:5]	3'h0	RW		PFR fading control
slope_l	J	32	[4:0]	5'h00	RW		PFR slope control
ren_s	J	33	[1]	1'b0	RW		PFR enable control 1'b0 : disable 1'b1 : enable
ben_s	J	33	[0]	1'b0	RW		
fading_s	J	34	[7:5]	3'h0	RW		PFR fading control
slope_s	J	34	[4:0]	5'h00	RW		PFR slope control
ren_vs	J	35	[1]	1'b0	RW		PFR enable control

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
ben_vs	J	35	[0]	1'b0	RW		1'b0 : disable 1'b1 : enable
fading_vs	J	36	[7:5]	3'h0	RW		PFR fading control
slope_vs	J	36	[4:0]	5'h00	RW		PFR slope control

4.2.4. HDR Combination

Combination takes 2-exposure (DCG(HCG + LCG), VS(Very-Short) images and combines them into an HDR image. Based on DCG (HCG), images are synthesized by applying the respective combination ratios to DCG (LCG) and VS. “ls_ratio” is applied to DCG (LCG) as a ratio determined by the HCG/LCG ratio. “ls_ratio” * “svs_ratio” is applied to VS. “svs_ratio” is determined by VS exposure time. Figure 24 shows the combination concept of 3-image. Table 21 shows the combination control registers

Figure 24. 2-exposure HDR combination

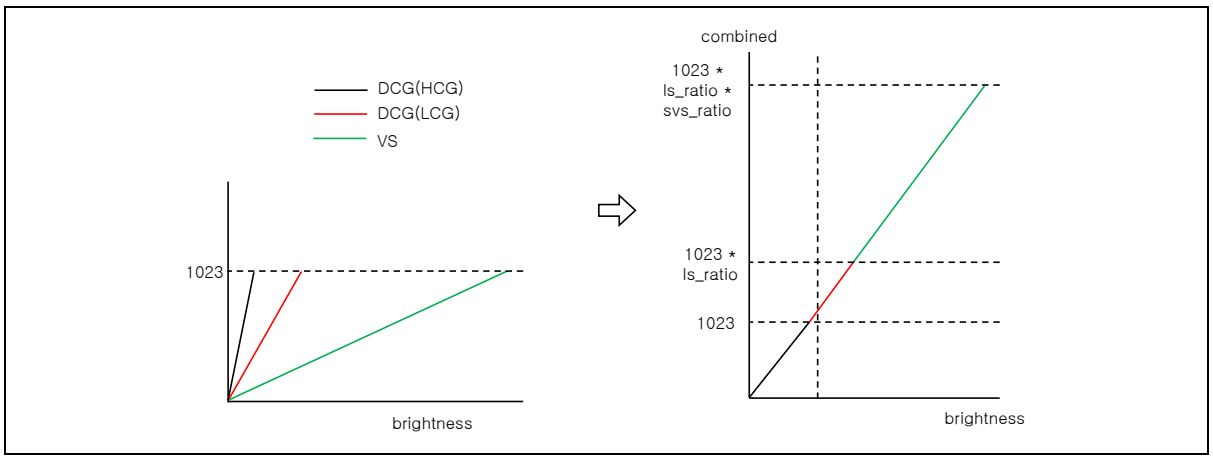


Table 21. Register Table – Combination

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
ls_ratio_h	I	52	[7:0]	8’h08	RW		Long-Short ratio : 0x0200 = x1
ls_ratio_l	I	53	[7:0]	8’h00	RW		
svs_ratio_h	I	54	[7:0]	8’h80	RW		Short-VS ratio : 0x0200 = x1
svs_ratio_l	I	55	[7:0]	8’h00	RW		

4.2.5. Compression

Compression makes HDR 18 bits data into 12 bits data. Compression can set 4 knee points in PWL method.

Figure 29. Compression Knee points

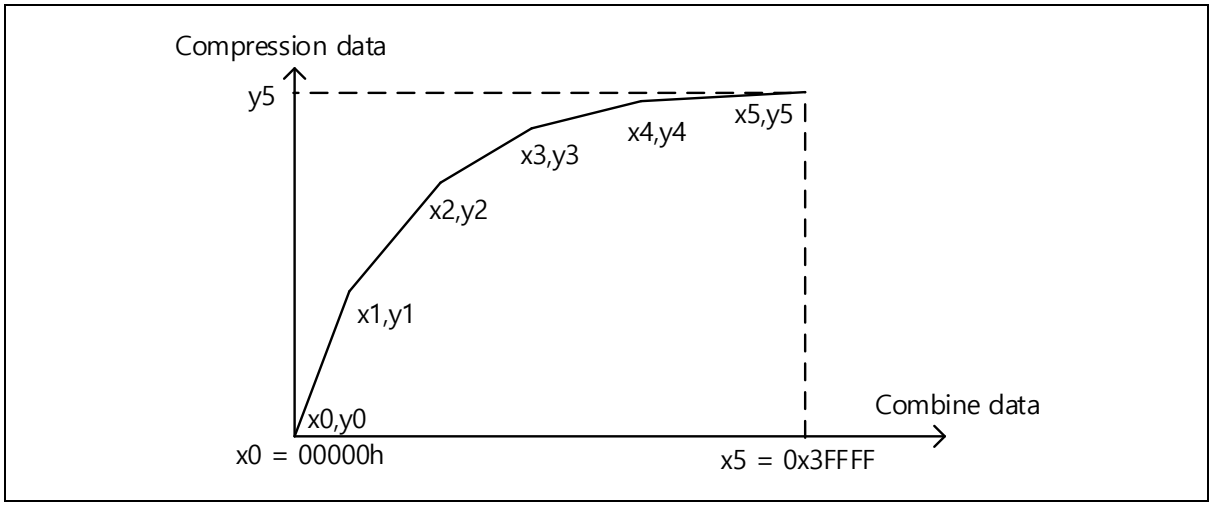


Table 22 Register Table – Compression

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
cmp_x1_h	I	E2	[1:0]	2'h0	RW		Knee point x1
cmp_x1_m	I	E3	[7:0]	8'h04	RW		
cmp_x1_l	I	E4	[7:0]	8'h00	RW		
cmp_x2_h	I	E5	[1:0]	2'h0	RW		Knee point x2
cmp_x2_m	I	E6	[7:0]	8'h10	RW		
cmp_x2_l	I	E7	[7:0]	8'h00	RW		
cmp_x3_h	I	E8	[1:0]	2'h1	RW		Knee point x3
cmp_x3_m	I	E9	[7:0]	8'h00	RW		
cmp_x3_l	I	EA	[7:0]	8'h00	RW		
cmp_x4_h	I	EB	[1:0]	2'h3	RW		Knee point x4
cmp_x4_m	I	EC	[7:0]	8'hFF	RW		
cmp_x4_l	I	ED	[7:0]	8'hFF	RW		
cmp_y0_h	I	EE	[3:0]	4'h0	RW		Knee point y0
cmp_y0_l	I	EF	[7:0]	8'h00	RW		
cmp_y1_h	I	F0	[3:0]	4'h4	RW		Knee point y1
cmp_y1_l	I	F1	[7:0]	8'h00	RW		
cmp_y2_h	I	F2	[3:0]	4'h8	RW		Knee point y2
cmp_y2_l	I	F3	[7:0]	8'h00	RW		
cmp_y3_h	I	F4	[3:0]	4'hC	RW		Knee point y3
cmp_y3_l	I	F5	[7:0]	8'h00	RW		
cmp_y4_h	I	F6	[3:0]	4'hF	RW		Knee point y4

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
cmp_y4_l	I	F7	[7:0]	8'hFF	RW		
cmp_y5_h	I	F8	[3:0]	4'hF	RW		Knee point y5
cmp_y5_l	I	F9	[7:0]	8'hFF	RW		

4.3. Formatter

Formatter receives 12-bit compressed data from the ISP and configures the data structure differently depending on the MIPI and DVP interfaces.

4.3.1. Output Data format

The ISP block on the PK9210K produces an HDR output that combines multiple images. The format to which the image is combined varies according to exposure.

Table 23 shows output data format for the PK9210K to support.

Table 23. Output Data format

Format		Image combined
12-bit data compressed	2-exposure	L+S+VS
	1-exposure	L+S

4.3.2. DVP Output Data Structure

DVP output data structure include Start of Active Video (SAV), active data, blank data and End of Active Video (EAV). (refer to Figure 25, Table 24)

The starting data of the blank data outputs "sync_CCIR_FF", "sync_CCIR_00", "sync_CCIR_00", and "sync_blankSAV" in order. The blank data outputs "sync_CCIR_80" and "sync_CCIR_10" repeatedly. The end data of the blank data outputs "sync_CCIR_FF", "sync_CCIR_00", "sync_CCIR_00", and "sync_blankEAV" in order.

The starting data of the active data outputs "sync_CCIR_FF", "sync_CCIR_00", "sync_CCIR_00", and "sync_activeSAV" in order. The active data outputs 12-bit compressed data. The end data of the active data outputs "sync_CCIR_FF", "sync_CCIR_00", "sync_CCIR_00", and "sync_activeEAV" in order.

Figure 25. DVP Output Data Structure

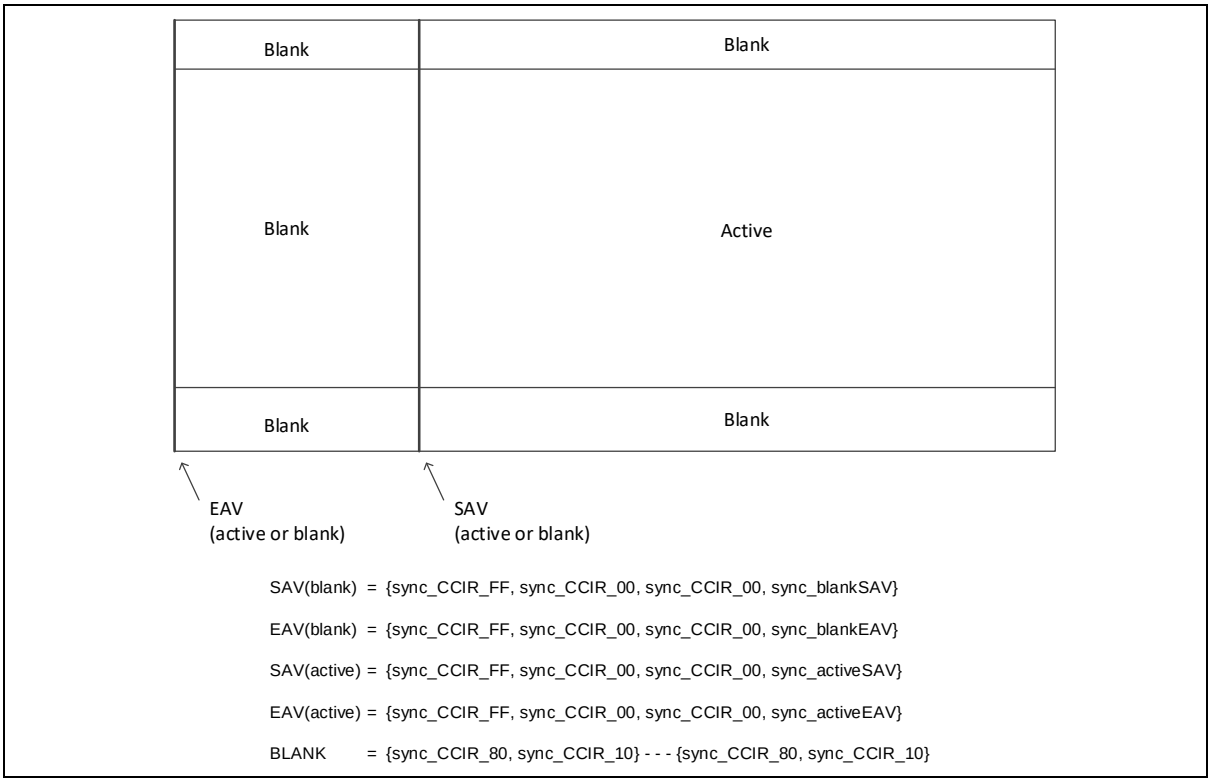


Table 24 Register Table - DVP data control

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
sync_blankEAV_h	A	6D	[3:0]	0x0B	RW		Blanking EAV control High Byte
sync_blankEAV_l	A	6E	[7:0]	0x60	RW		Blanking EAV control Low Byte
sync_blankSAV_h	A	6F	[3:0]	0x0A	RW		Blanking SAV control High Byte
sync_blankSAV_l	A	70	[7:0]	0xB0	RW		Blanking SAV control Low Byte
sync_activeEAV_h	A	71	[3:0]	0x09	RW		Active EAV control High Byte
sync_activeEAV_l	A	72	[7:0]	0xD0	RW		Active EAV control Low Byte
sync_activeSAV_h	A	73	[3:0]	0x08	RW		Active SAV control High Byte

sync_activeSAV_l	A	74	[7:0]	0x00	RW	Active SAV control Low Byte
sync_CCIR_FF_h	A	75	[3:0]	0x0F	RW	Format header control 0 (FF) High Byte
sync_CCIR_FF_l	A	76	[7:0]	0xFF	RW	Format header control 0 (FF) Low Byte
sync_CCIR_00_h	A	77	[3:0]	0x00	RW	Format header control 1 (00) High Byte
sync_CCIR_00_l	A	78	[7:0]	0x00	RW	Format header control 1 (00) Low Byte
sync_CCIR_80_h	A	79	[3:0]	0x08	RW	Blank data control 0 (80) High Byte
sync_CCIR_80_l	A	7A	[7:0]	0x00	RW	Blank data control 0 (80) Low Byte
sync_CCIR_10_h	A	7B	[3:0]	0x01	RW	Blank data control 1 (10) High Byte
sync_CCIR_10_l	A	7C	[7:0]	0x00	RW	Blank data control 1 (10) Low Byte

If active data is in the same order and value as SAV and EAV, the beginning and end of the data are unknown. Use "data_clamp" to exclude such cases. When “data_clamp” is enabled, active data is clamped by “data_min” and “data_max” as shown in Table 25. “data_min” determines minimum value of active data, and “data_max” determines maximum value of active data. (refer to Table 26)

Table 25. Clamped data

output bit	data_min	data_max
Data 12bit	0010h	0FE0h

Table 26. Register Table - data clamp

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
data_clamp	A	B2	[1]	1'b0	RW	aev	Effective data clamping enable 1'b0 : disable 1'b1 : enable
data_min_h	A	B4	[3:0]	0x00	RW		Minimum active data High Byte
data_min_l	A	B5	[7:0]	0x00	RW		Minimum active data Low Byte
data_max_h	A	B6	[3:0]	0x0F	RW		Maximum active data High Byte
data_max_l	A	B7	[7:0]	0xFF	RW		Maximum active data Low Byte

4.3.2.1. DVP PCLK

The polarity of the PCLK pulse is controlled by "pclk_polarity" as shown in Figure 26.

Figure 26. PCLK polarity

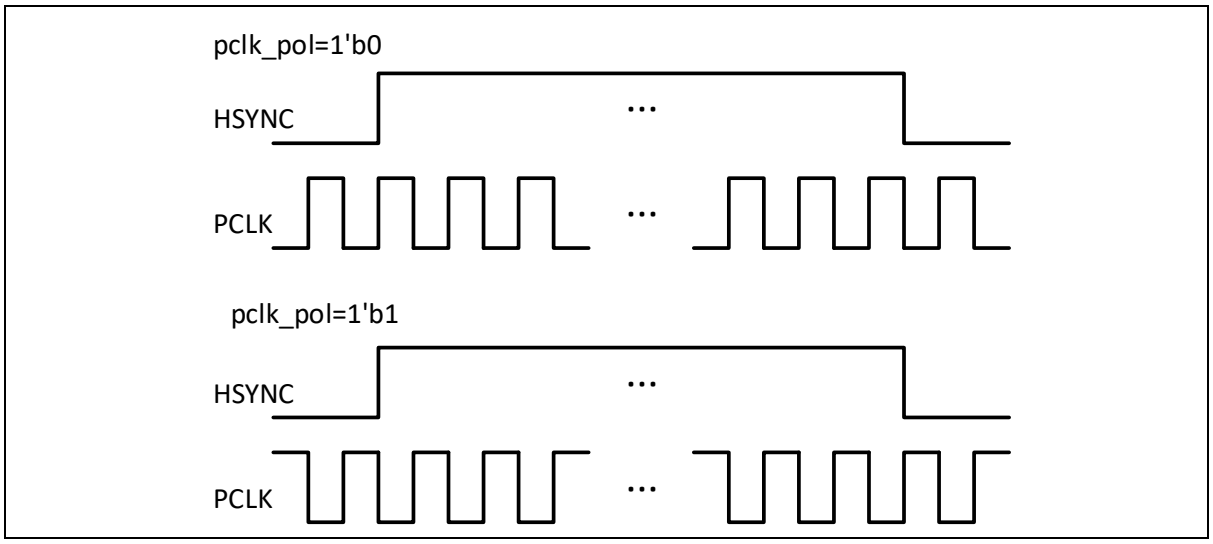


Table 27. Register Table - PCLK control

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
pclk_polarity	A	20	[2]	1'b0	RW		Change PCLK phase

4.3.2.2. DVP Vsync and Hsync

The starting and ending points of the VSYNC pulse are controlled by "vsyncstartrow0", "vsyncstoprow0", and "vsynccolumn0". HSYNC pulse operates in synchronization with active data, so it does not adjust the start and end points separately.

Figure 27 shows operation of “vsync_drop” and “hsync_drop”. The outputs of VSYNC and HSYNC pulse are controlled by "vsync_drop" and "hsync_drop", respectively. When "sync_hsync_all_lines" is enabled, an HSYNC pulse is generated during the VSYNC blank area. Figure 28 shows operation of “sync_hsync_all_lines”.

Figure 27. Sync drop

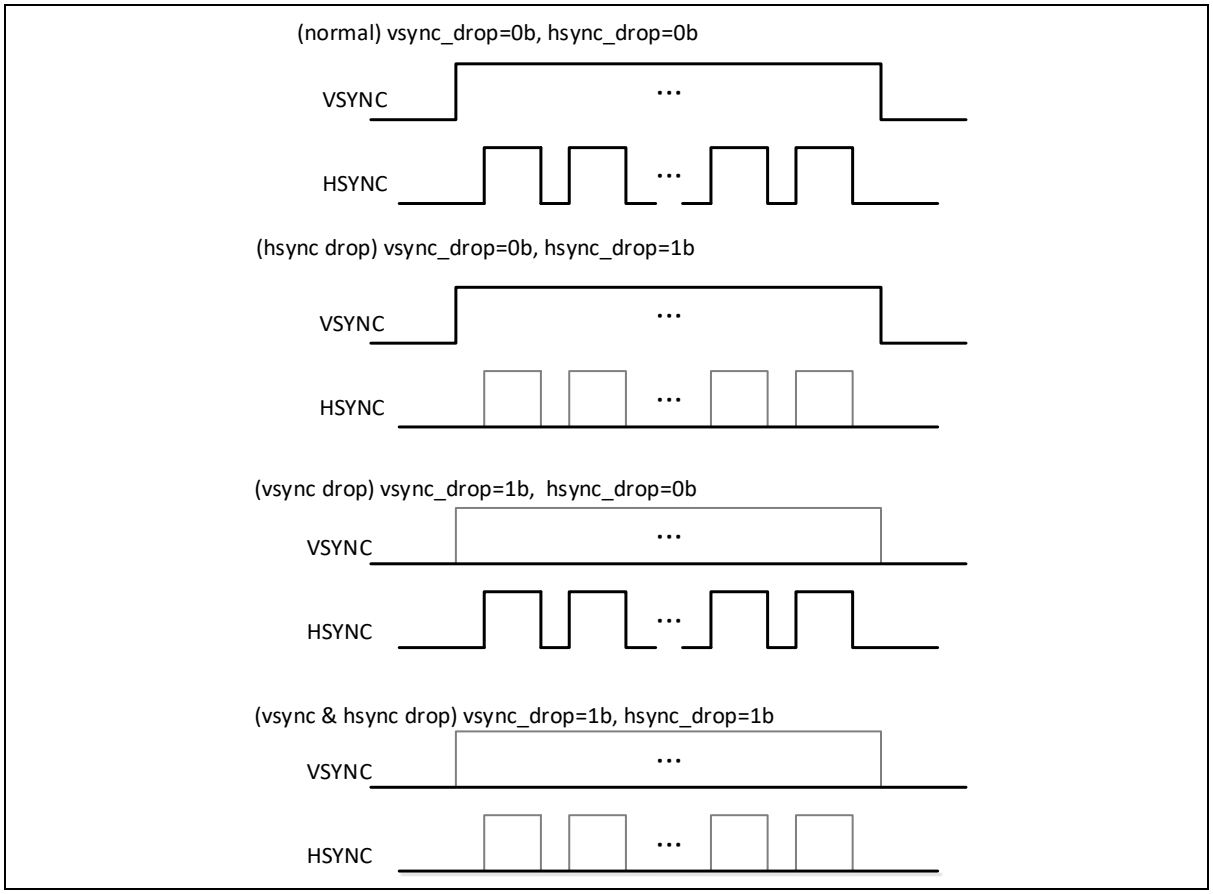
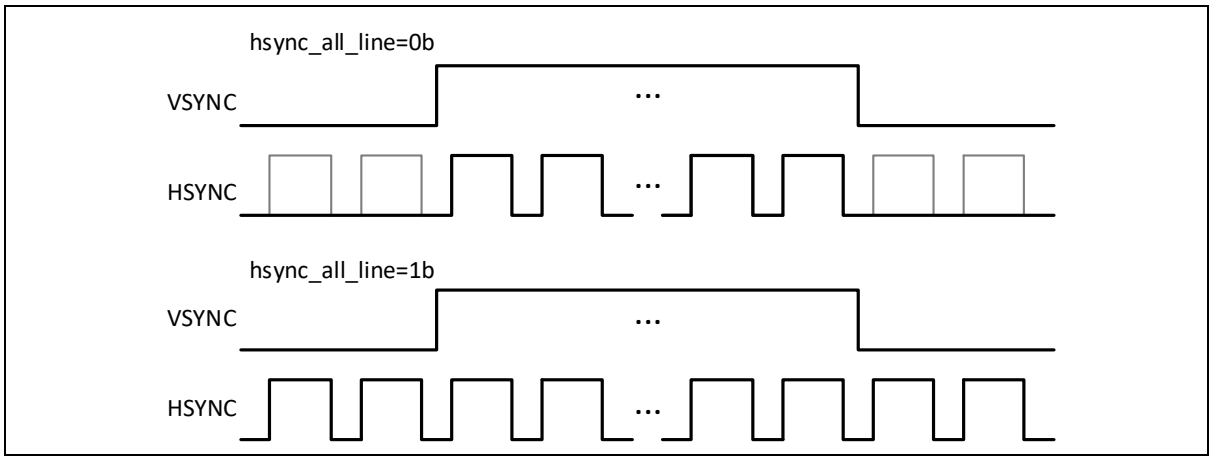


Figure 28. hsync_all_line



The polarity of the VSYNC and HSYNC pulses is controlled by "sync_vsync_polarity" and "sync_hsync_polarity" as shown in Figure 29.

Figure 29. Sync polarity

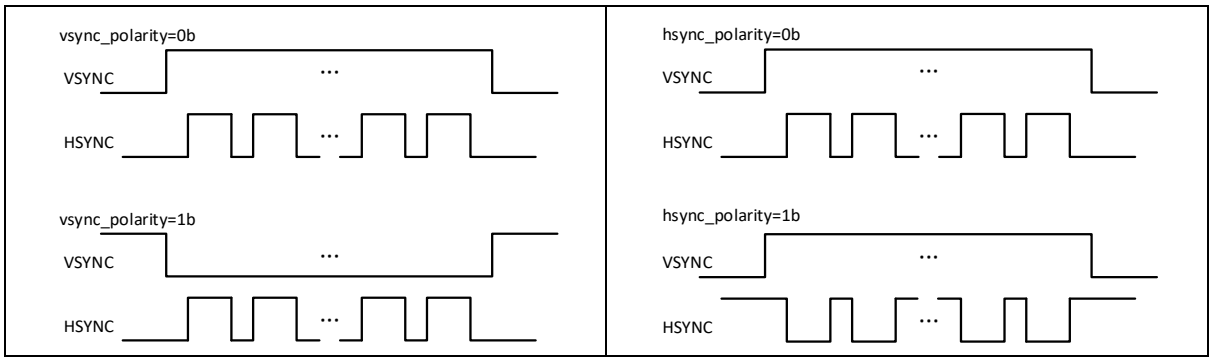


Table 28. Register Table - sync control

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
vsyncstartrow0_h	A	7E	[5:0]	0x00	RW	aev	Parallel interface - Vertical sync start control High Byte MIPI interface @ Virtual channel 0 - Frame start control High Byte
vsyncstartrow0_l	A	7F	[7:0]	0x15	RW	aev	Parallel interface - Vertical sync start control Low Byte MIPI interface @ Virtual channel 0 - Frame start control Low Byte
vsyncstoprow0_h	A	80	[5:0]	0x04	RW	aev	Parallel interface - Vertical sync end control High Byte MIPI interface @ Virtual channel 0 - Frame end control High Byte
vsyncstoprow0_l	A	81	[7:0]	0x5D	RW	aev	Parallel interface - Vertical sync end control Low Byte MIPI interface @ Virtual channel 0 - Frame end control Low Byte
vsynccolumn0_h	A	82	[4:0]	0x00	RW	aev	Internal vsync 0 start point High Byte @ column counter
vsynccolumn0_l	A	83	[7:0]	0x02	RW	aev	Internal vsync 0 start point Low Byte @ column counter
vsync_drop	A	B1	[6]	1'b0	RW	aev	Vsync drop control 1'b0 : No drop 1'b1 : vsync drop
hsync_drop	A	B1	[5]	1'b0	RW	aev	hsync drop control 1'b0 : No drop 1'b1 : hsync drop
vsync_polarity	A	B2	[6]	1'b0	RW	aev	Vsync polarity change 1'b0 : disable 1'b1 : enable
hsync_all_line	A	B2	[5]	1'b0	RW	aev	Hsync output all lines enable(black and active) 1'b0 : No hsync during vertical blank 1'b1 : hsync during vertical blank

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
hsync_polarlity	A	B2	[4]	1'b0	RW	aev	Hsync polarity change 1'b0 : disable 1'b1 : enable

4.3.3. MIPI Output Data Structure

MIPI output data structure include Frame Start (FS), Packet Header (PH), active data, Packet Footer (PF), and Frame End (FE). (refer to Figure 30)

Figure 30. MIPI Output Data Structure

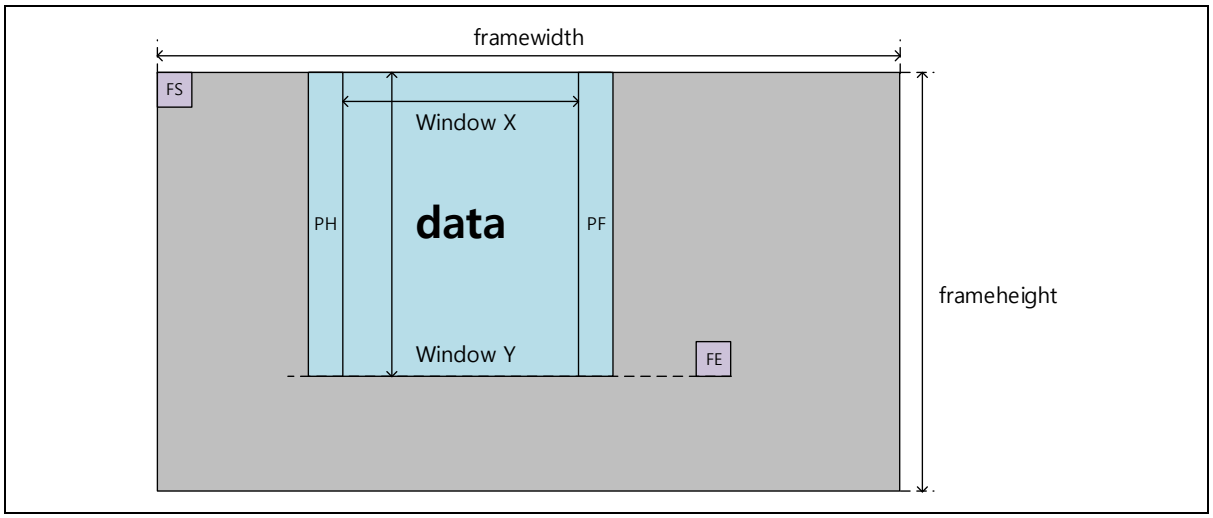


Table 29. Data ID and Data Type

Data ID	Data Type (Image Format)
00 hex	Frame start
01 hex	Frame end
2C hex	Raw bayer 12-bit

Table 30 shows the registers for packet data output selection

Table 30. Register Table - MIPI packet control

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
vsyncstartrow0_h	A	7E	[5:0]	0x00	RW	aev	Parallel interface - Vertical sync start control High Byte MIPI interface @ Virtual channel 0 - Frame start control High Byte
vsyncstartrow0_l	A	7F	[7:0]	0x15	RW	aev	Parallel interface - Vertical sync start control Low Byte MIPI interface @ Virtual channel 0 - Frame start control Low Byte
vsyncstoprow0_h	A	80	[5:0]	0x04	RW	aev	Parallel interface - Vertical sync end control High Byte MIPI interface @ Virtual channel 0 - Frame end control High Byte
vsyncstoprow0_l	A	81	[7:0]	0x5D	RW	aev	Parallel interface - Vertical sync end control Low Byte MIPI interface @ Virtual channel 0 - Frame end control Low Byte

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
vsynccolumn0_h	A	82	[4:0]	0x00	RW	aev	Internal vsync 0 start point High Byte @ column counter
vsynccolumn0_l	A	83	[7:0]	0x02	RW	aev	Internal vsync 0 start point Low Byte @ column counter
mipi_data_id0	G	42	[7:0]	0x2B	RW		MIPI data 0 identifier

5. Image Output Interface

The PK9210K supports HDR output through an output interface with MIPI and DVP.

5.1. MIPI Interface

The MIPI interface consists of a total of 6 pins, from D0 to D5, and shares the pins with the DVP interface.

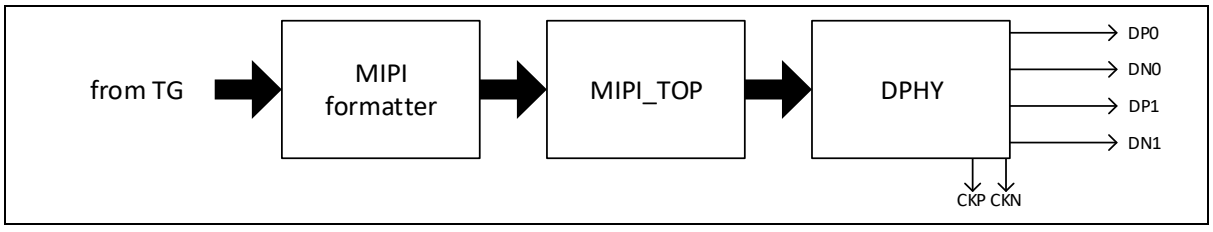
Table 31 Register Table - Interface control

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
d5_pad_en	A	21	[1]	1'b0	RW		D5 pad control 1'b0 : disable 1'b1 : enable
d4_pad_en	A	21	[0]	1'b0	RW		D4 pad control 1'b0 : disable 1'b1 : enable
d3_pad_en	A	22	[7]	1'b0	RW		D3 pad control 1'b0 : disable 1'b1 : enable
d2_pad_en	A	22	[6]	1'b0	RW		D2 pad control 1'b0 : disable 1'b1 : enable
d1_pad_en	A	22	[5]	1'b0	RW		D1 pad control 1'b0 : disable 1'b1 : enable
d0_pad_en	A	22	[4]	1'b0	RW		D0 pad control 1'b0 : disable 1'b1 : enable

5.1.1. Reference for Design

MIPI design in the PK9210K is based on “MIPI Alliance Standard for Camera Serial Interface 2 (CSI-2), Version 1.00” and “MIPI Alliance Standard for D-PHY, Version 0.65” specification documents. Output of MIPI consists of one clock lane and two data lanes.

Figure 31. MIPI block diagram



5.1.2. MIPI Clock Relations

Figure 32 and Table 32 shows Clock and MIPI operation. The mipi_clk operates at a frequency four times faster than dclk. In addition, it is used for generating MIPI data and clock lane signal. The dclk is clock rate determined by tg_clk, raw bit, MIPI lane.

Figure 32. MIPI clock relations diagram

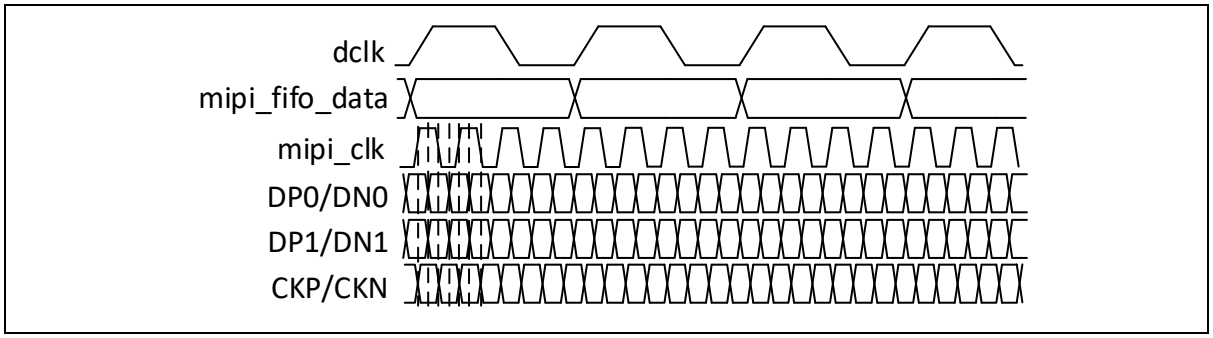


Table 32. MIPI clock with lane and frame rate

Data type	MIPI lane	fps	pclk	dclk	mipi_clk	PLL1	PLL2	unit
raw 12-bit	2-lane	30	74.25	55.6875	222.75	594	445.5	[MHz]

Figure 33 shows raw 12-bit format.

Figure 33. MIPI raw12 frame format

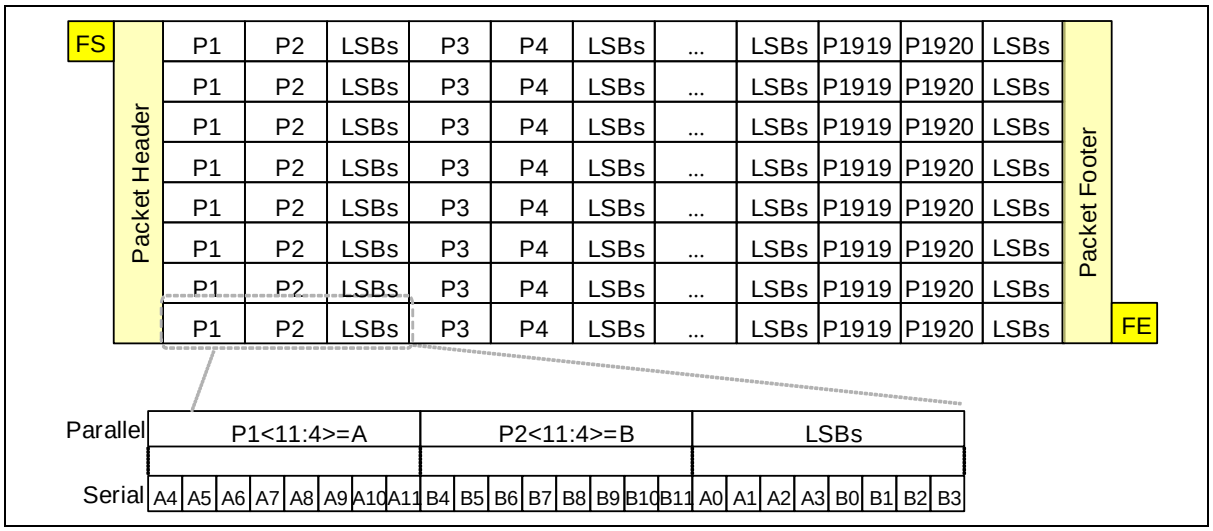


Table 33 shows registers relevant to MIPI mode setting.

Table 33. Register Table - MIPI mode setting

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
mipi_en	G	04	[6]	1'b0	RW	aev	MIPI enable 1'b0 : disable 1'b1 : enable
clk_hs_mode	G	04	[4]	1'b1	RW	aev	MIPI clock lane hs mode 1'b0 : LP & HS mode 1'b1 : only HS mode

5.2. DVP Interface

The DVP interface consists of 12 data pins from D0 to D11 and VSYNC, HSYNC, and PCLK pins.

Each pin can control the output individually, and the drivability can be adjusted. If data pin order mapping reverse in DVP interface, the user can change “data_swap” to swap data pin order from MSB to LSB or from LSB to MSB. (refer to Table 34)

Table 34. Register Table - Interface control

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
pclk_pad_en	A	20	[3]	1'b0	RW		PCLK pad enable 1'b0 : disable 1'b1 : enable
vsync_pad_en	A	20	[7]	1'b0	RW		Vsync pad enable 1'b0 : disable 1'b1 : enable
hsync_pad_en	A	20	[4]	1'b0	RW		Hsync pad enable 1'b0 : disable 1'b1 : enable
d11_pad_en	A	21	[7]	1'b0	RW		D11 pad control 1'b0 : disable 1'b1 : enable
d10_pad_en	A	21	[6]	1'b0	RW		D10 pad control 1'b0 : disable 1'b1 : enable
d9_pad_en	A	21	[5]	1'b0	RW		D9 pad control 1'b0 : disable 1'b1 : enable
d8_pad_en	A	21	[4]	1'b0	RW		D8 pad control 1'b0 : disable 1'b1 : enable
d7_pad_en	A	21	[3]	1'b0	RW		D7 pad control 1'b0 : disable 1'b1 : enable
d6_pad_en	A	21	[2]	1'b0	RW		D6 pad control 1'b0 : disable 1'b1 : enable
d5_pad_en	A	21	[1]	1'b0	RW		D5 pad control 1'b0 : disable 1'b1 : enable
d4_pad_en	A	21	[0]	1'b0	RW		D4 pad control 1'b0 : disable 1'b1 : enable
d3_pad_en	A	22	[7]	1'b0	RW		D3 pad control 1'b0 : disable 1'b1 : enable
d2_pad_en	A	22	[6]	1'b0	RW		D2 pad control 1'b0 : disable 1'b1 : enable

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
d1_pad_en	A	22	[5]	1'b0	RW		D1 pad control 1'b0 : disable 1'b1 : enable
d0_pad_en	A	22	[4]	1'b0	RW		D0 pad control 1'b0 : disable 1'b1 : enable
pad_drv[1:0]	A	1F	[7:6]	2'b00	RW		Data pad drivability control
pclk_drv[1:0]	A	1F	[5:4]	2'b00	RW		PCLK pad drivability control
hsync_drv	A	20	[6:5]	2'b00	RW		Hsync pad drivability control
dpad_swap	A	20	[1]	1'b0	RW		Data pad swap option 1'b0 : [MSB:LSB] 1'b1 : [LSB:MSB]

6. I2C Interface

6.1.1. I2C Communication

I2C communication is a serial interface which utilizes SCL/SDA lines to transfer 8-bit data per transaction. PK9210K includes only I2C slave function and requires external master to access the internal registers. Each transaction requires 8-bit data and 1-bit acknowledge bit. slave address can be extended by CADDR0 PAD, CADDR1 PAD. (refer to Table 35)

Table 35. Slave address summary

	Case 1	Case 2	Case 3	Case 4
CADDR1 PAD	**Low**	**Low**	**High**	**High**
CADDR0 PAD	**Low**	**High**	**Low**	**High**
Slave address (hex)	74	75	76	77
Write address (hex)	E8	EA	EC	EE
Read address (hex)	E9	EB	ED	EF

There are four types of operations supported in PK9210K's I2C operation: single write, multiple write, single read, multiple read.

Figure 34. Single write mode operation

Figure 34 shows single write operation of I2C communication. In single write operation, after the start state, 7-bit slave address and write bit are transmitted from master device to PK9210K. If correct slave address is detected, PK9210K reply with acknowledge bit as confirmation of valid address. Then master device transmits register address to PK9210K and reply with acknowledge bit from PK9210K. Lastly, 8-bit data is sent to PK9210K and waits for acknowledge bit again. Once acknowledge bit is received, master device announces the stop state to terminate I2C communication.

Figure 35. Multiple Write Mode operation

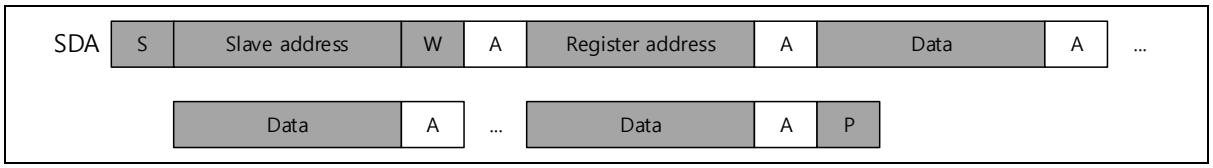


Figure 35 shows multiple write operation of I2C communication. Multiple write operation works the same until stop state procedure except transferring data. Before announcing the stop state, master device transmits more data. If PK9210K detects multiple write operation, any data stream following the first 8-bit is stored in subsequent register addresses of the first register address.

Figure 36. Single read mode operation

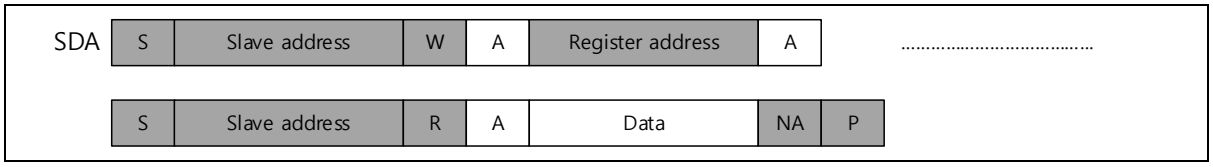
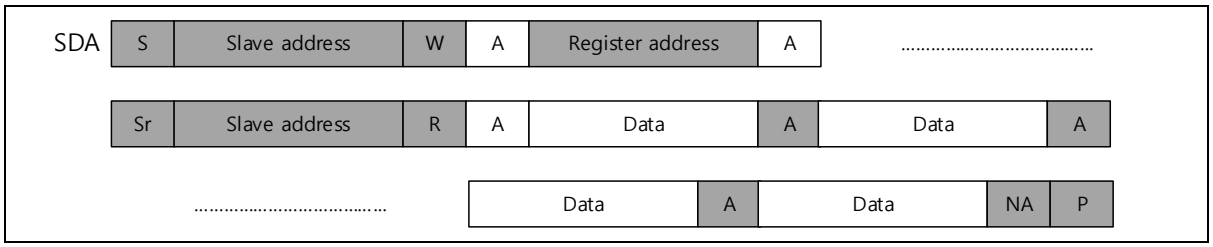


Figure 36 shows read operation of I2C communication. Read operation consists of two sub-procedure: address write, and data read. The procedure is performed exactly same as register address write procedure from single write operation. Afterward, master device announces repeated start and transmit slave address with read bit.

When PK9210K detects read operation, PK9210K sends acknowledge bit to master device, then reads register corresponding to register address. PK9210K transmits read data to master device and waits for master device to respond. If master device responds with no acknowledge bit followed by stop state, read operation is terminated.

Figure 37. Multiple read mode operation



On the other hand, if master device responds with acknowledge, PK9210K reads the subsequent register and transmits again. If master device replies with acknowledge bit after each data transaction, PK9210K will continuously read the subsequent register and transmit until no acknowledge bit followed by stop state is presented. (refer to Figure 37)

If only one 8-bit data is read, the procedure is single read operation. Whereas, reading more than 8-bit data is multiple read operation.

6.1.2. Register Update Timing

Registers has three different types of update timing: aev and autov update, regular update.

Registers with aev and autov update type update new values from I2C write operation at the last line of the frame. Whereas, registers with regular update type apply new values immediately after I2C write operation. However, by changing update control register value, register updates for aev and autov type can either be disabled or be updated immediately.

Table 36 shows registers relevant to register update control.

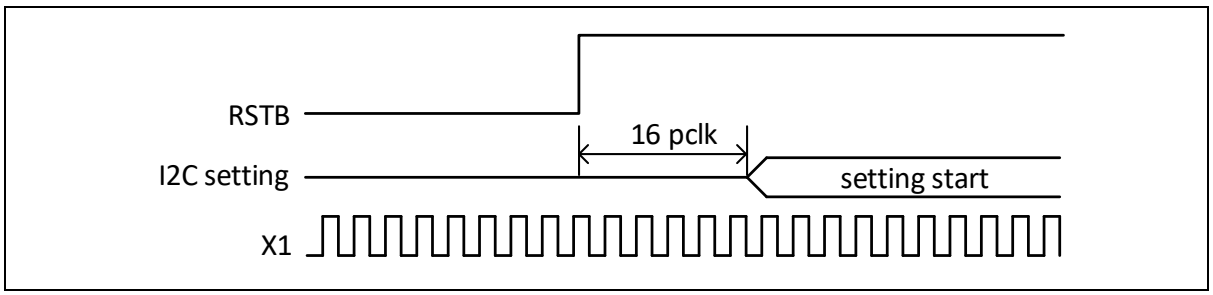
Table 36. I2C update timing control

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
updatecontrol[3:0]	A	15	[7:4]	4'b0101	RW		Control I2C register with autov, aev update type LSB 2-bit updatecontrol[1:0] controls aev update and MSB 2-bit updatecontrol[3:2] controls autov update 2'b00 : no update 2'b01 : aev update 2'b1x : immediate aev update

6.1.3. Initialization Timing for I2C interface

Register control through I2C communication is possible at the point where 16 pclk has passed after RSTB becomes high. (refer to Figure 38).

Figure 38. Available timing for I2C communication after system reset.



7. OTP memory

The PK9210K contains a total of 2K-bit OTP memory. A large portion of 2K-bit are used by PixelPlus for pixel sensitivity deviation correction and x,y coordinates of the OTP DPC. The user can use 128-bit OTP memory. The user can write or read data values with I2C to OTP memory. The user may store necessary information (identification information, etc.) in the OTP memory.

The OTP memory can write or read 64-bit data values at a time. The addresses of the OTP memory that you want to read or write to are set to "otp_addr_start" and "otp_addr_stop"(refer to Table 37). The 64-bit write value is set to "otp_wdata0~otp_wdata7", and the 64-bit read value is stored in "otp_rdata0~otp_rdata7".

Table 37. OTP address

OTP bit	OTP address
[0]	0x00
[1]	0x01
[2]	0x02
-	-
[125]	0x7D
[126]	0x7E
[127]	0x7F

The user must be very careful when writing data values to OTP memory. Because OTP memory can only be used once. If the bit is changed to 1'b1, it can never be returned to 1'b0. So the user should follow the process below.

7.1. The process of reading the OTP memory

The process of reading the OTP memory is as follows.

(1) Set the values of "otp_addr_start" and "otp_addr_stop".

(2) Set the value of "otp_mode_read" to 1'b1.

(3) Check the values from "otp_rdata0" to "otp_rdata7".

7.2. The process of writing to the OTP memory

The process of writing the OTP memory is as follows.

(1) Check that all read values are 1'b0.

(2) Set the values of "otp_addr_start" and "otp_addr_stop".

(3) Set values from "otp_wdata0" to "otp_wdata7".

(4) Set the value of "otp_mode_write" to 1'b1.

(5) Compare the write value and the read value values.

Table 38. Register Table - OTP

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
otp_mode_read	H	04	[5]	1'b0	RW		OTP memory read mode 1'b0 : disable 1'b1 : enable
otp_mode_write	H	04	[4]	1'b0	RW		OTP memory write mode 1'b0 : disable 1'b1 : enable
otp_addr_start_h	H	06	[2:0]	0x00	RW		OTP start address High Byte
otp_addr_start_l	H	07	[7:0]	0x00	RW		OTP start address Low Byte
otp_addr_stop_h	H	08	[2:0]	0x00	RW		OTP stop address High Byte
otp_addr_stop_l	H	09	[7:0]	0x3F	RW		OTP stop address Low Byte
otp_wdata0	H	0A	[7:0]	0x00	RW		OTP write data 0
otp_wdata1	H	0B	[7:0]	0x00	RW		OTP write data 1
otp_wdata2	H	0C	[7:0]	0x00	RW		OTP write data 2
otp_wdata3	H	0D	[7:0]	0x00	RW		OTP write data 3
otp_wdata4	H	0E	[7:0]	0x00	RW		OTP write data 4
otp_wdata5	H	0F	[7:0]	0x00	RW		OTP write data 5
otp_wdata6	H	10	[7:0]	0x00	RW		OTP write data 6
otp_wdata7	H	11	[7:0]	0x00	RW		OTP write data 7
otp_rdata0	H	9E	[7:0]		RO		OTP read data 0
otp_rdata1	H	9F	[7:0]		RO		OTP read data 1
otp_rdata2	H	A0	[7:0]		RO		OTP read data 2
otp_rdata3	H	A1	[7:0]		RO		OTP read data 3
otp_rdata4	H	A2	[7:0]		RO		OTP read data 4
otp_rdata5	H	A3	[7:0]		RO		OTP read data 5
otp_rdata6	H	A4	[7:0]		RO		OTP read data 6
otp_rdata7	H	A5	[7:0]		RO		OTP read data 7

8. Electrical Characteristics

8.1. Absolute Maximum Ratings

AVDD supply voltage:	-0.3 [V] to 4.0 [V]
HVDD supply voltage:	-0.3 [V] to 4.0 [V]
DVDD supply voltage:	-0.3 [V] to 1.8 [V]
DVDDM supply voltage:	-0.3 [V] to 1.8 [V]
DC VTG at any input pin:	-0.3 [V] to HVDD+0.3 [V]
DC VTG at any output pin:	-0.3 [V] to HVDD+0.3 [V]
Storage temperature:	-40°C to +125°C

8.2. DC Characteristics

Table 39. DC characteristics

Symbol	Descriptions	Min	Typ	Max	Unit
AVDD	Analog VDD voltage relative to AGND level	2.7	2.8	2.9	[V]
HVDD	IO VDD voltage relative to DGND level	1.62	1.8~2.8	2.9	[V]
DVDD	Digital VDD voltage relative to DGND level	1.08	1.2	1.32	[V]
IDDD @ 30fps	HVDD=1.8/2.8 [V] @ DVP	8	8/16	15	[mA]
	AVDD=2.8 [V] @ DVP	25	27	29	
	DVDD=1.2 [V] @ DVP	81	95	118	
	HVDD=1.8/2.8 [V] @ MIPI	1	1/1	1	
	AVDD=2.8 [V] @ MIPI	25	27	29	
	DVDD=1.2 [V] @ MIPI	90	103	125	
IDDS	HVDD=1.8/2.8 [V]	21	29/69	84	[uA]
	AVDD=2.8 [V]	0.2	0.2	0.2	
	DVDD=1.2 [V]	5	1600	3200	
V_{IL1}	Input voltage low level	-	-	HVDD*0.3	[V]
V_{IH1}	Input voltage high level	HVDD*0.7	-	-	[V]
V_{IL2}	Input voltage low level for rClk, rData.	-	-	HVDD*0.3	[V]
V_{IH2}	Input voltage high level for rClk, rData .	HVDD*0.7	-	-	[V]
C_{IN}	Input pin capacitance	-	-	10	[pF]
V_{OL1}	Output voltage low	-	-	HVDD*0.2	[V]
V_{OH1}	Output voltage high	HVDD*0.8	-	-	[V]
V_{OL2}	Output voltage low level for rClk, rData.	-	-	HVDD*0.2	[V]
V_{OH2}	Output voltage high level for rData.	HVDD*0.8	-	-	[V]
I_{IN}	Input leakage current	-10	-	10	[uA]

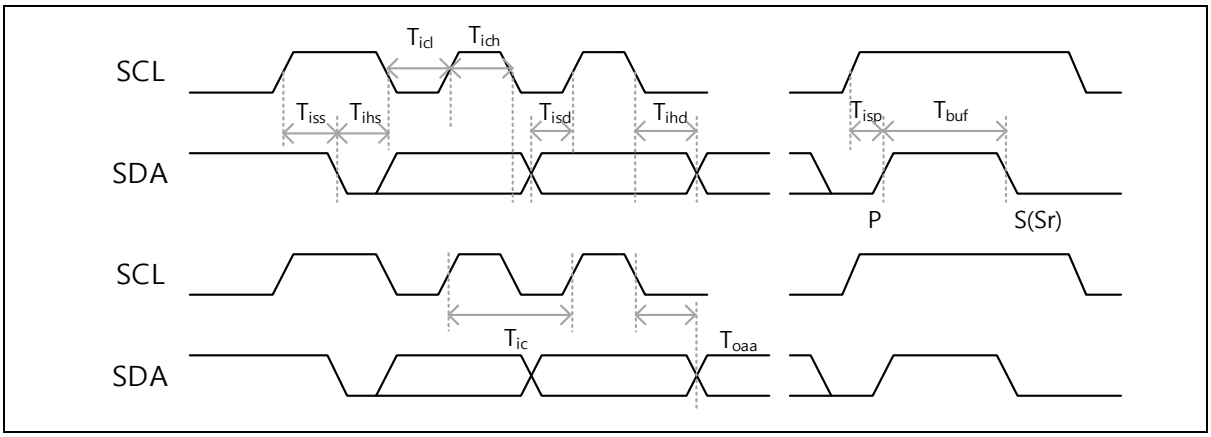
I_{OT}	Output leakage current	-10	-	10	[uA]

8.3. AC Characteristics

Table 40. 2-wire serial interface characteristics

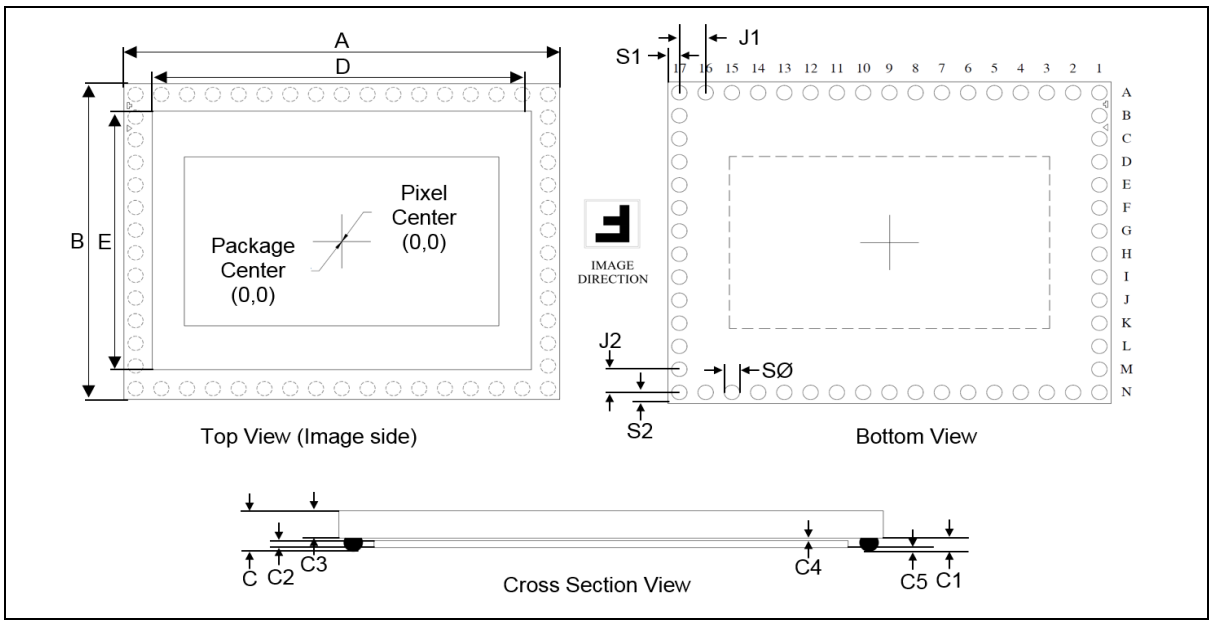
Symbol	Descriptions	Min	Typ	Max	Unit
f_{SCLK}	2-wire serial interface Clock frequency	-	-	400	kHz
T_{ic}	2-wire serial interface Clock period	2.5	-	-	us
T_{icl}	2-wire serial interface Clock low level width	1.66	-	-	us
T_{ich}	2-wire serial interface Clock high level width	0.83	-	-	us
T_{iss}	Setup time for start condition	0.83	-	-	us
T_{ihs}	Hold time for start condition	0.83	-	-	us
T_{isd}	Setup time for input data	266	-	-	ns
T_{ihd}	Hold time for input data	-	-	-	ns
T_{isp}	Setup time for stop condition	0.83	-	-	us
T_{buf}	Bus free time between a stop and a new start condition	1.66	-	-	us
T_{oaa}	Delay from SCL falling edge to output data transition	-	-	354	ns
T_r	10% to 90% rising time for SCL/SDA (load : 10pF)	-	-	46	ns
T_f	90% to 10% falling time for SCL/SDA (load : 10pF)	-	-	37	ns
R_p	SCL, SDA pull-up resistor	-	2	-	kΩ

Figure 39. Timing diagram of SCL and SDA



9. Physical specifications (unit : mm)

Figure 40. NeoPAC I Package specifications



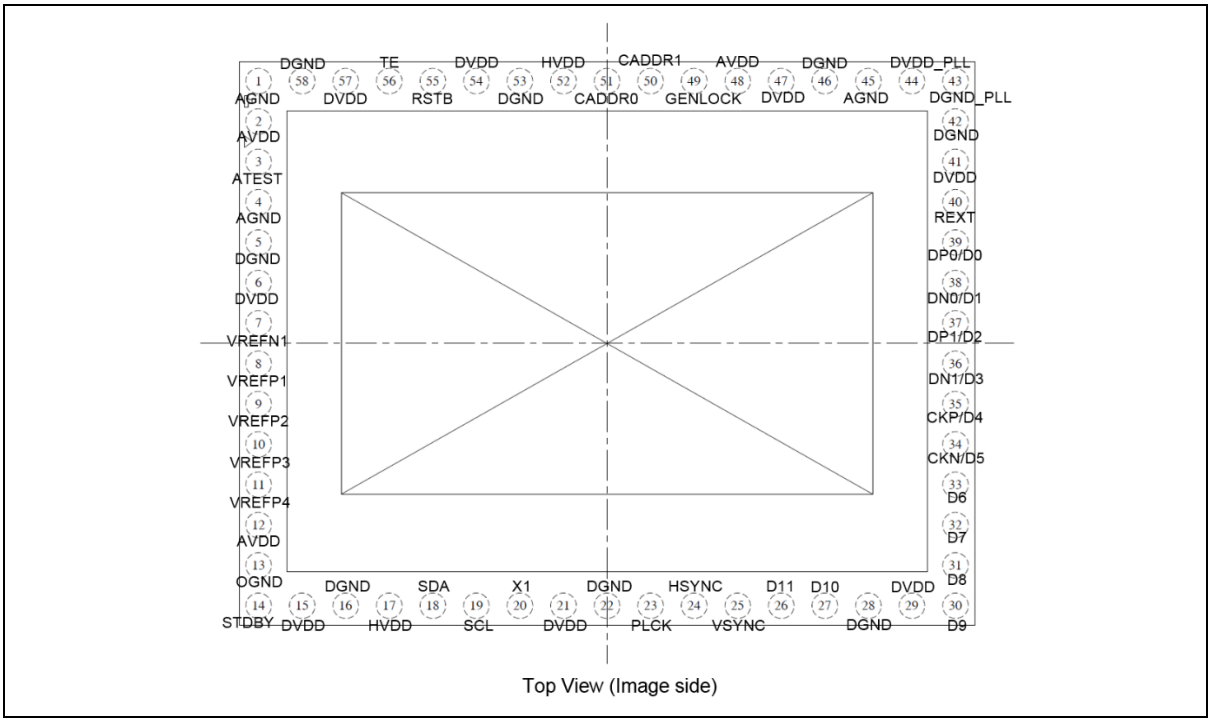
Under fill must be applied to PK9210K package products for automotive/semi-automotive. If under fill is not applied, the user of automotive/semi-automotive may have quality issues.

Table 41. Package dimensions

Parameter	Symbol	Min	Typ	Max	Unit
Package Body Dimension X	A	7.560	7.600	7.640	mm
Package Body Dimension Y	B	5.820	5.860	5.900	mm
Package Height	C	0.555	0.600	0.645	mm
Ball Height	C1	0.185	0.200	0.215	mm
Sensor Thickness	C2	0.090	0.100	0.110	mm
Glass Thickness	C3	0.390	0.400	0.410	mm
Air gap	C4	0.030	0.040	0.050	mm
Sensor bottom to Ball bottom distance	C5	0.040	0.060	0.080	mm
Sensor chip Dimension X (w/o Scribe lane)	D		6.620		mm
Sensor chip Dimension Y (w/o Scribe lane)	E		4.800		mm
Ball Pitch X axis	J1		0.450		mm
Ball Pitch Y axis	J2		0.420		mm
Total Ball Count	N		58		ea
Ball Count X axis	N1		17		ea
Ball Count Y axis	N2		14		ea
Edge to Ball Center Distance along X	S1		0.200		mm
Edge to Ball Center Distance along Y	S2		0.200		mm

Ball Diameter	SØ	0.250	0.265	0.280	mm

Figure 41. NeoPAC I Package Ball Coordinate Image



Top View (Image side)

Table 42 Ball Coordinate table

Ball pad#	Ball coordinate	Ball pad#	Ball coordinate
1	(-3.6000, 2.7300)	30	(3.6000, -2.7300)
2	(-3.6000, 2.3100)	31	(3.6000, -2.3100)
3	(-3.6000, 1.8900)	32	(3.6000, -1.8900)
4	(-3.6000, 1.4700)	33	(3.6000, -1.4700)
5	(-3.6000, 1.0500)	34	(3.6000, -1.0500)
6	(-3.6000, 0.6300)	35	(3.6000, -0.6300)
7	(-3.6000, 0.2100)	36	(3.6000, -0.2100)
8	(-3.6000, -0.2100)	37	(3.6000, 0.2100)
9	(-3.6000, -0.6300)	38	(3.6000, 0.6300)
10	(-3.6000, -1.0500)	39	(3.6000, 1.0500)
11	(-3.6000, -1.4700)	40	(3.6000, 1.4700)
12	(-3.6000, -1.8900)	41	(3.6000, 1.8900)
13	(-3.6000, -2.3100)	42	(3.6000, 2.3100)
14	(-3.6000, -2.7300)	43	(3.6000, 2.7300)
15	(-3.1500, -2.7300)	44	(3.1500, 2.7300)
16	(-2.7000, -2.7300)	45	(2.7000, 2.7300)
17	(-2.2500, -2.7300)	46	(2.2500, 2.7300)
18	(-1.8000, -2.7300)	47	(1.8000, 2.7300)
19	(-1.3500, -2.7300)	48	(1.3500, 2.7300)

20	(-0.9000, -2.7300)	49	(0.9000, 2.7300)
21	(-0.4500, -2.7300)	50	(0.4500, 2.7300)
22	(0.0000, -2.7300)	51	(0.0000, 2.7300)
23	(0.4500, -2.7300)	52	(-0.4500, 2.7300)
24	(0.9000, -2.7300)	53	(-0.9000, 2.7300)
25	(1.3500, -2.7300)	54	(-1.3500, 2.7300)
26	(1.8000, -2.7300)	55	(-1.8000, 2.7300)
27	(2.2500, -2.7300)	56	(-2.2500, 2.7300)
28	(2.7000, -2.7300)	57	(-2.7000, 2.7300)
29	(3.1500, -2.7300)	58	(-3.1500, 2.7300)

Table 43 Pad Assignment Table

Ball pad#	Die pad#	Pad name	Ball pad#	Die pad#	Pad name
1	69,70	AGND (GND4)	30	31	D9
2,12	2,71,72,74	AVDD	31	32	D8
3	73	ATEST	32	33	D7
4	75	AGND (GND4)	33	34	D6
5	76	DGND (GND3)	34	35	CKN/D5
6	77	DVDD (DVDD3)	35	36	CKP/D4
7	78	VREFN1	36	37	DN1/D3
8	79	VREFP1	37	38	DP1/D2
9	80	VREFP2	38	39	DN0/D1
10	81	VREFP3	39	40	DP0/D0
11	82	VREFP4	40	41	REXT
13	3	OGND (GND4)	41	42	DVDD (DVDD1)
14	4	STDBY	42	43, 44	DGND, HGND (GND1)
15	5,6	DVDD (DVDD1)	43	46	DGND_PLL (GND2)
16	7,8,9	DGND, HGND (GND1)	44	47	DVDD_PLL (DVDD2)
17,52	10,21,30,45,57	HVDD	45	49	AGND (GND4)
18	11	SDA	46	50,51	DGND (GND3)
19	12	SCL	47	52,53	DVDD (DVDD3)
20	13	X1	48	48	AVDD
21	14,15,16	DVDD (DVDD1)	49	54	GENLOCK
22	17,18,19,20	DGND, HGND (GND1)	50	55	CADDR1
23	22	PCLK	51	56	CADDR0
24	23	HSYNC	53	58,59,60	DGND, HGND (GND3)
25	24	VSYNC	54	61,62	DVDD (DVDD3)

26	25	D11	55	63	RSTB
27	26	D10	56	64	TE
28	28,29	DGND, HGND (GND1)	57	65,66	DVDD (DVDD3)
29	27	DVDD (DVDD1)	58	67,68	DGND (GND3)

10. Register Map

Table 44. Register Table - Group A

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
DeviceID_H	A	00	[7:0]	0x92	RO		Device ID High Byte
DeviceID_L	A	01	[7:0]	0x10	RO		Device ID Low Byte
bank	A	03	[7:0]	0x00	RW		Register group selector
mirror	A	05	[1:0]	0x00	RW	aev	Image Inversion mirror[1] : vertical inversion mirror[0] : horizontal inversion
framewidth_h	A	06	[4:0]	0x08	RW	aev	Framewidth High Byte (must be larger than window width)
framewidth_l	A	07	[7:0]	0x97	RW	aev	Framewidth Low Byte (must be larger than window width)
fheight_a_h	A	08	[5:0]	0x04	RW	aev	Frameheight High Byte (must be larger than window height)
fheight_a_l	A	09	[7:0]	0x7C	RW	aev	Frameheight Low Byte (must be larger than window height)
windowx1_h	A	0C	[7:0]	0x00	RW	aev	Window horizontal start point High Byte
windowx1_l	A	0D	[7:0]	0x0C	RW	aev	Window horizontal start point Low Byte
windowy1_h	A	0E	[7:0]	0x00	RW	aev	Window vertical start point High Byte
windowy1_l	A	0F	[7:0]	0x00	RW	aev	Window vertical start point Low Byte
windowx2_h	A	10	[7:0]	0x07	RW	aev	Window horizontal end point High Byte
windowx2_l	A	11	[7:0]	0x9C	RW	aev	Window horizontal end point Low Byte
windowy2_h	A	12	[7:0]	0x04	RW	aev	Window vertical end point High Byte
windowy2_l	A	13	[7:0]	0x48	RW	aev	Window vertical end point Low Byte
i2c_control_1	A	15	[7:0]	0x50	RW		I2c control register 1
softreset	A	1B	[0]	0x00	RW		Soft reset 1'b0 : disable 1'b1 : enable (after succesful reset value reverts to 0)
pad_control2	A	1F	[7:0]	0x00	RW		Pad control 2
pad_control3	A	20	[7:0]	0x00	RW		Pad control 3
pad_control4	A	21	[7:0]	0x00	RW		Pad control 4
pad_control5	A	22	[7:0]	0x00	RW		Pad control 5
pll_control2	A	4F	[7:0]	0x7A	RW		PLL control 2
plltg_rdiv	A	55	[4:0]	0x01	RW		PLL1 Ref Freq. Divider 00000b : unused 00001b : Reference Clock/ 1 … 11111b : Reference Clock/ 31
plltg_dsm_divh_h	A	56	[7:0]	0x16	RW		{divh_h,divh_l} = NDIV_TG * 256

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
plltg_dsm_divh_l	A	57	[7:0]	0x00	RW		
plltg_dsm_divl_h	A	58	[7:0]	0x16	RW		{divl_h, divl_l} = NDIV_TG * 256
plltg_dsm_divl_l	A	59	[7:0]	0x00	RW		
plltg_dsm_xstep	A	5A	[7:0]	0x00	RW		SSC parameters XSTEP
plltg_dsm_ystep	A	5B	[7:0]	0x00	RW		SSC parameters YSTEP
pllmp_rdiv	A	5D	[4:0]	0x01	RW		PLL2 Ref Freq. Divider 00000b : unused 00001b : Reference Clock/ 1 … 11111b : Reference Clock/ 31
pllmp_dsm_divh_h	A	5E	[7:0]	0x16	RW		{divh_h,divh_l} = NDIV_MP_UPPER * 256
pllmp_dsm_divh_l	A	5F	[7:0]	0x00	RW		
pllmp_dsm_divl_h	A	60	[7:0]	0x16	RW		{divl_h, divl_l} = NDIV_MP_LOWER * 256
pllmp_dsm_divl_l	A	61	[7:0]	0x00	RW		
clkdiv1	A	65	[7:0]	0xA3	RW	aev	Clock divider 1
clkdiv2	A	66	[7:0]	0x18	RW	aev	Clock divider 2
sync_blankEAV_h	A	6D	[3:0]	0x0B	RW		Blanking EAV control High Byte
sync_blankEAV_l	A	6E	[7:0]	0x60	RW		Blanking EAV control Low Byte
sync_blankSAV_h	A	6F	[3:0]	0x0A	RW		Blanking SAV control High Byte
sync_blankSAV_l	A	70	[7:0]	0xB0	RW		Blanking SAV control Low Byte
sync_activeEAV_h	A	71	[3:0]	0x09	RW		Active EAV control High Byte
sync_activeEAV_l	A	72	[7:0]	0xD0	RW		Active EAV control Low Byte
sync_activeSAV_h	A	73	[3:0]	0x08	RW		Active SAV control High Byte
sync_activeSAV_l	A	74	[7:0]	0x00	RW		Active SAV control Low Byte
sync_CCIR_FF_h	A	75	[3:0]	0x0F	RW		Format header control 0 (FF) High Byte
sync_CCIR_FF_l	A	76	[7:0]	0xFF	RW		Format header control 0 (FF) Low Byte
sync_CCIR_00_h	A	77	[3:0]	0x00	RW		Format header control 1 (00) High Byte
sync_CCIR_00_l	A	78	[7:0]	0x00	RW		Format header control 1 (00) Low Byte
sync_CCIR_80_h	A	79	[3:0]	0x08	RW		Blank data control 0 (80) High Byte
sync_CCIR_80_l	A	7A	[7:0]	0x00	RW		Blank data control 0 (80) Low Byte
sync_CCIR_10_h	A	7B	[3:0]	0x01	RW		Blank data control 1 (10) High Byte
sync_CCIR_10_l	A	7C	[7:0]	0x00	RW		Blank data control 1 (10) Low Byte
vsyncstartrow0_h	A	7E	[5:0]	0x00	RW	aev	Parallel interface - Vertical sync start control High Byte MIPI interface @ Virtual channel 0 - Frame start control High Byte
vsyncstartrow0_l	A	7F	[7:0]	0x15	RW	aev	Parallel interface - Vertical sync start control Low Byte

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
							MIPI interface @ Virtual channel 0 - Frame start control Low Byte
vsyncstoprow0_h	A	80	[5:0]	0x04	RW	aev	Parallel interface - Vertical sync end control High Byte MIPI interface @ Virtual channel 0 - Frame end control High Byte
vsyncstoprow0_l	A	81	[7:0]	0x5D	RW	aev	Parallel interface - Vertical sync end control Low Byte MIPI interface @ Virtual channel 0 - Frame end control Low Byte
vsynccolumn0_h	A	82	[4:0]	0x00	RW	aev	Internal vsync 0 start point High Byte @ column counter
vsynccolumn0_l	A	83	[7:0]	0x02	RW	aev	Internal vsync 0 start point Low Byte @ column counter
sync_control_0	A	B1	[7:0]	0x80	RW	aev	Sync_control 0
sync_control_1	A	B2	[7:0]	0x00	RW	aev	Sync_control 1
data_min_h	A	B4	[3:0]	0x00	RW		Minimum active data High Byte
data_min_l	A	B5	[7:0]	0x00	RW		Minimum active data Low Byte
data_max_h	A	B6	[3:0]	0x0F	RW		Maximum active data High Byte
data_max_l	A	B7	[7:0]	0xFF	RW		Maximum active data Low Byte

Table 45. Register Table - Group B

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
bayer_control_03	B	06	[7]	0x40	RW		bayer control 03
bayer_control_07	B	0A	[7:0]	0x00	RW		bayer control 07
rjump_top_h	B	1F	[5:0]	0x00	RW	aev	Row jump top value High Byte
rjump_top_l	B	20	[7:0]	0x0C	RW	aev	Row jump top value Low Byte
rjump_bot_h	B	21	[5:0]	0x00	RW	aev	Row jump bottom value High Byte
rjump_bot_l	B	22	[7:0]	0x0C	RW	aev	Row jump bottom value Low Byte
rcount_genlock_h	B	57	[5:0]	0x00	RW		Genlock row count High Byte
rcount_genlock_l	B	58	[7:0]	0x01	RW		Genlock row count Low Byte
ccount_genlock_h	B	59	[4:0]	0x00	RW		Genlock column count High Byte
ccount_genlock_l	B	5A	[7:0]	0x01	RW		Genlock column count Low Byte
maxexp_vs	B	60	[7:0]	0x0A	RW		Maximum exposure of very short image
inttime_h	B	6B	[7:0]	0x01	RW	wr_en	Long Line inttime High Byte
inttime_m	B	6C	[7:0]	0x40	RW	wr_en	Long Line inttime Low Byte
inttime_l	B	6D	[7:0]	0x00	RW	wr_en	Long Column inttime
inttime_vs_h	B	6E	[7:0]	0x00	RW	wr_en	Very short Line inttime High Byte

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
inttime_vs_m	B	6F	[7:0]	0x02	RW	wr_en	Very short Line inttime Low Byte
inttime_vs_l	B	70	[7:0]	0x00	RW	wr_en	Very shrot Column inttime
globalgain	B	74	[7:0]	0x00	RW	wr_en	Analog gain 0
globalgain_vs	B	75	[7:0]	0x00	RW	wr_en	Analog gain 1
digitalgain_l	B	78	[7:0]	0x10	RW	wr_en	Digital gain of long data
digitalgain_s	B	79	[7:0]	0x10	RW	wr_en	Digital gain of short data
digitalgain_vs	B	7A	[7:0]	0x10	RW	wr_en	Digital gain of very short data
wb_rgain_l_h	B	81	[0]	0x00	RW		White balance "R" gain for long High Byte
wb_rgain_l_l	B	82	[7:0]	0x40	RW		White balance "R" gain for long Low Byte
wb_ggain_l_h	B	83	[0]	0x00	RW		White balance "G" gain for long High Byte
wb_ggain_l_l	B	84	[7:0]	0x40	RW		White balance "G" gain for long Low Byte
wb_bgain_l_h	B	85	[0]	0x00	RW		White balance "B" gain for long High Byte
wb_bgain_l_l	B	86	[7:0]	0x40	RW		White balance "B" gain for long Low Byte
wb_rgain_s_h	B	87	[0]	0x00	RW		White balance "R" gain for short High Byte
wb_rgain_s_l	B	88	[7:0]	0x40	RW		White balance "R" gain for short Low Byte
wb_ggain_s_h	B	89	[0]	0x00	RW		White balance "G" gain for short High Byte
wb_ggain_s_l	B	8A	[7:0]	0x40	RW		White balance "G" gain for short Low Byte
wb_bgain_s_h	B	8B	[0]	0x00	RW		White balance "B" gain for short High Byte
wb_bgain_s_l	B	8C	[7:0]	0x40	RW		White balance "B" gain for short Low Byte
wb_rgain_vs_h	B	8D	[0]	0x00	RW		White balance "R" gain for very short 1 High Byte
wb_rgain_vs_l	B	8E	[7:0]	0x40	RW		White balance "R" gain for very short 1 Low Byte
wb_ggain_vs_h	B	8F	[0]	0x00	RW		White balance "G" gain for very short 1 High Byte
wb_ggain_vs_l	B	90	[7:0]	0x40	RW		White balance "G" gain for very short 1 Low Byte
wb_bgain_vs_h	B	91	[0]	0x00	RW		White balance "B" gain for very short 1 High Byte
wb_bgain_vs_l	B	92	[7:0]	0x40	RW		White balance "B" gain for very short 1 Low Byte
wr_en	B	9A	[0]	0x00	RW		Update exposure related register 1'b0 : no update 1'b1 : wr_en set
wr_en_off	B	9B	[0]	0x00	RW		Update exposure related register 1'b0 : update @ wr_en = 1'b1 1'b1 : Immediately update

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
tp_control_0	B	A2	[7:0]	0x00	RW		Test pattern selection
tp_control_1_h	B	A3	[1:0]	0x00	RW		R color for test pattern High Byte
tp_control_1_l	B	A4	[7:0]	0x00	RW		R color for test pattern Low Byte
tp_control_2_h	B	A5	[1:0]	0x00	RW		G1 color for test pattern High Byte
tp_control_2_l	B	A6	[7:0]	0x00	RW		G1 color for test pattern Low Byte
tp_control_3_h	B	A7	[1:0]	0x00	RW		G2 color for test pattern High Byte
tp_control_3_l	B	A8	[7:0]	0x00	RW		G2 color for test pattern Low Byte
tp_control_4_h	B	A9	[1:0]	0x00	RW		B color for test pattern High Byte
tp_control_4_l	B	AA	[7:0]	0x00	RW		B color for test pattern Low Byte

Table 46. Register Table - Group G

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
mipi_control_0	G	04	[7:0]	0x30	RW	aev	MIPI control 0
mipi_data_id0	G	42	[7:0]	0x2B	RW		MIPI data 0 identifier

Table 47. Register Table - Group H

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
otp_ctrl0	H	04	[7:0]	0x00	RW		OTP control 0
otp_addr_start_h	H	06	[2:0]	0x00	RW		OTP start address High Byte
otp_addr_start_l	H	07	[7:0]	0x00	RW		OTP start address Low Byte
otp_addr_stop_h	H	08	[2:0]	0x00	RW		OTP stop address High Byte
otp_addr_stop_l	H	09	[7:0]	0x3F	RW		OTP stop address Low Byte
otp_wdata0	H	0A	[7:0]	0x00	RW		OTP write data 0
otp_wdata1	H	0B	[7:0]	0x00	RW		OTP write data 1
otp_wdata2	H	0C	[7:0]	0x00	RW		OTP write data 2
otp_wdata3	H	0D	[7:0]	0x00	RW		OTP write data 3
otp_wdata4	H	0E	[7:0]	0x00	RW		OTP write data 4
otp_wdata5	H	0F	[7:0]	0x00	RW		OTP write data 5
otp_wdata6	H	10	[7:0]	0x00	RW		OTP write data 6
otp_wdata7	H	11	[7:0]	0x00	RW		OTP write data 7
otp_rdata0	H	9E	[7:0]		RO		OTP read data 0
otp_rdata1	H	9F	[7:0]		RO		OTP read data 1
otp_rdata2	H	A0	[7:0]		RO		OTP read data 2

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
otp_rdata3	H	A1	[7:0]		RO		OTP read data 3
otp_rdata4	H	A2	[7:0]		RO		OTP read data 4
otp_rdata5	H	A3	[7:0]		RO		OTP read data 5
otp_rdata6	H	A4	[7:0]		RO		OTP read data 6
otp_rdata7	H	A5	[7:0]		RO		OTP read data 7

Table 48. Register Table - Group I

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
isp_func_0	I	04	[7:0]	0xFF	RW		ISP control 0
isp_func_4	I	08	[7:9]	0x00	RW		ISP control 4
dpc_offset_l	I	0B	[5:0]	0x10	RW		DPC offset1 control
dpc_offset2_l	I	0C	[7:0]	0x20	RW		DPC offset2 control
dpc_offset_s	I	1B	[5:0]	0x10	RW		DPC offset1 control
dpc_offset2_s	I	1C	[7:0]	0x20	RW		DPC offset2 control
dpc_offset_vs	I	2B	[5:0]	0x10	RW		DPC offset1 control
dpc_offset2_vs	I	2C	[7:0]	0x20	RW		DPC offset2 control
dpc_dark_p_l	I	0D	[7:0]	0x40	RO		DPC strength 0x40 : 1-pixel correction 0x60 : 2-pixel correction 0x7F : 3-pixel correction
dpc_dark_n_l	I	0E	[6:0]	0x40	RO		
dpc_dark_p_s	I	1D	[7:0]	0x40	RO		
dpc_dark_n_s	I	1E	[6:0]	0x40	RO		
dpc_dark_p_vs	I	2D	[7:0]	0x40	RO		
dpc_dark_n_vs	I	2E	[6:0]	0x40	RO		
ls_ratio_h	I	52	[7:0]	0x08	RW		Long-Short ratio : 0x0200 = x1
ls_ratio_l	I	53	[7:0]	0x00	RW		
svs_ratio_h	I	54	[7:0]	0x80	RW		Short-VS ratio : 0x0200 = x1
svs_ratio_l	I	55	[7:0]	0x00	RW		
cmp_x1_h	I	E2	[1:0]	0x00	RW		Knee point x1
cmp_x1_m	I	E3	[7:0]	0x04	RW		
cmp_x1_l	I	E4	[7:0]	0x00	RW		
cmp_x2_h	I	E5	[1:0]	0x00	RW		Knee point x2
cmp_x2_m	I	E6	[7:0]	0x10	RW		
cmp_x2_l	I	E7	[7:0]	0x00	RW		
cmp_x3_h	I	E8	[1:0]	0x01	RW		Knee point x3

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
cmp_x3_m	I	E9	[7:0]	0x00	RW		
cmp_x3_l	I	EA	[7:0]	0x00	RW		
cmp_x4_h	I	EB	[1:0]	0x03	RW		Knee point x4
cmp_x4_m	I	EC	[7:0]	0xFF	RW		
cmp_x4_l	I	ED	[7:0]	0xFF	RW		
cmp_y0_h	I	EE	[3:0]	0x00	RW		Knee point y0
cmp_y0_l	I	EF	[7:0]	0x00	RW		
cmp_y1_h	I	F0	[3:0]	0x04	RW		Knee point y1
cmp_y1_l	I	F1	[7:0]	0x00	RW		
cmp_y2_h	I	F2	[3:0]	0x08	RW		Knee point y2
cmp_y2_l	I	F3	[7:0]	0x00	RW		
cmp_y3_h	I	F4	[3:0]	0x0C	RW		Knee point y3
cmp_y3_l	I	F5	[7:0]	0x00	RW		
cmp_y4_h	I	F6	[3:0]	0x0F	RW		Knee point y4
cmp_y4_l	I	F7	[7:0]	0xFF	RW		
cmp_y5_h	I	F8	[3:0]	0x0F	RW		Knee point y5
cmp_y5_l	I	F9	[7:0]	0xFF	RW		

Table 49. Register Table - Group J

Register name	Address		Bits	Init. Val.	Type	Update	Description
	Bank	Hex					
pfr_ctrl0	J	31	[7:0]	0x00	RW		PFR control 0
pfr_ctrl1	J	32	[7:0]	0x00	RW		PFR control 1
pfr_ctrl2	J	33	[7:0]	0x00	RW		PFR control 2
pfr_ctrl3	J	34	[7:0]	0x00	RW		PFR control 3
pfr_ctrl4	J	35	[7:0]	0x00	RW		PFR control 4
pfr_ctrl5	J	36	[7:0]	0x00	RW		PFR control 5

Table 50. Register Table - Control register map

Register name	Address		Bits	Init. Val.	Type	Up date	Description
	Bank	Hex					
i2c_control_1	A	15	[7:4]	4'b0101	RW		**updatecontrol** Control I2C register with autov, aev update type LSB 2-bit updatecontrol[1:0] controls aev update and MSB 2-bit updatecontrol[3:2] controls autov

						update 2'b00 : no update 2'b01 : aev update 2'b1x : immediate aev update
pad_control2	A	1F	[7:6]	2'b00	RW	**pad_drv** Data pad drivability control
	A	1F	[5:4]	2'b00	RW	**pclk_drv** PCLK pad drivability control
pad_control3	A	20	[7]	1'b0	RW	**vsync_pad_en** Vsync pad enable 1'b0 : disable 1'b1 : enable
	A	20	[6:5]	2'b00	RW	**hsync_drv** Hsync Pad drivability control
	A	20	[4]	1'b0	RW	**hsync_pad_en** Hsync pad enable 1'b0 : disable 1'b1 : enable
	A	20	[3]	1'b0	RW	**pclk_pad_en** PCLK pad enable 1'b0 : disable 1'b1 : enable
	A	20	[2]	1'b0	RW	**pclk_polarity** Change PCLK phase
	A	20	[1]	1'b0	RW	**dpad_swap** Data pad swap option 1'b0 : [MSB:LSB] 1'b1 : [LSB:MSB]
	A	20	[0]	1'b0	RW	**genlock_pad_en** GENLOCK pad enable 1'b0 : disable 1'b1 : enable
pad_control4	A	21	[7]	1'b0	RW	**d11_pad_en** D11 pad control 1'b0 : disable 1'b1 : enable
	A	21	[6]	1'b0	RW	**d10_pad_en** D10 pad control 1'b0 : disable 1'b1 : enable
	A	21	[5]	1'b0	RW	**d9_pad_en** D9 pad control 1'b0 : disable 1'b1 : enable
	A	21	[4]	1'b0	RW	**d8_pad_en** D8 pad control 1'b0 : disable 1'b1 : enable
	A	21	[3]	1'b0	RW	**d7_pad_en** D7 pad control 1'b0 : disable 1'b1 : enable
	A	21	[2]	1'b0	RW	**d6_pad_en** D6 pad control

							1'b0 : disable 1'b1 : enable
	A	21	[1]	1'b0	RW		**d5_pad_en** D5 pad control 1'b0 : disable 1'b1 : enable
	A	21	[0]	1'b0	RW		**d4_pad_en** D4 pad control 1'b0 : disable 1'b1 : enable
pad_control5	A	22	[7]	1'b0	RW		**d3_pad_en** D3 pad control 1'b0 : disable 1'b1 : enable
	A	22	[6]	1'b0	RW		**d2_pad_en** D2 pad control 1'b0 : disable 1'b1 : enable
	A	22	[5]	1'b0	RW		**d1_pad_en** D1 pad control 1'b0 : disable 1'b1 : enable
	A	22	[4]	1'b0	RW		**d0_pad_en** D0 pad control 1'b0 : disable 1'b1 : enable
pll_control2	A	4F	[3]	1'b1	RW		**pllmp_pd** PLL2 power down mode 1'b0 : pll1 power on 1'b1 : pll1 power down
	A	4F	[5]	1'b1	RW		**plltg_pd** PLL1 power down mode 1'b0 : pll1 power on 1'b1 : pll1 power down
	A	4F	[4]	1'b1	RW		**reg_pll_bypass** PLL bypass 1'b0 : use pll mode 1'b1 : pll bypass mode
clkdiv1	A	65	[3:0]	4'b0011	RW	aev	**ispclk_div** ISP clock divider 4'b0000 = 1/1 4'b0001 = 1/2 4'b0010 = 1/4 4'b0011 = 1/8 4'b0100 = 1/16 4'b1000 = 1/3 4'b0001 = 1/6 4'b0010 = 1/12 4'b0011 = 1/24 4'b0100 = 1/48
clkdiv2	A	66	[7:6]	2'b00	RW	aev	**mipiclk_div** MIPI clock divider mipi_clk =vco2/(2^mipiclk_div)
	A	66	[5:3]	3'b011	RW	aev	**ddclk_div** MIPI byte clock divider ddclk =vco2/[2^ddclk_div]

sync_control_0	A	B1	[6]	1'b0	RW	aev	**vsync_drop** Vsync drop control 1'b0 : No drop 1'b1 : vsync drop
	A	B1	[5]	1'b0	RW	aev	**hsync_drop** hsync drop control 1'b0 : No drop 1'b1 : hsync drop
sync_control_1	A	B2	[6]	1'b0	RW	aev	**vsyncpolarity** Vsync polarity change 1'b0 : disable 1'b1 : enable
	A	B2	[5]	1'b0	RW	aev	**hsync_all_line** Hsync output all lines enable(black and active) 1'b0 : No hsync during vertical blank 1'b1 : hsync during vertical blank
	A	B2	[4]	1'b0	RW	aev	**hsync_polarity** Hsync polarity change 1'b0 : disable 1'b1 : enable
	A	B2	[1]	1'b0	RW	aev	**data_clamp** Effective data clamping enable 1'b0 : disable 1'b1 : enable
bayer_control_03	B	06	[7]	1'b0	RW		**wb_en** White balance enable control 1'b0 : disable 1'b1 : enable
bayer_control_07	B	0A	[3]	1'b0	RW		**genlock_en** GENLOCK enable 1'b0 : disable 1'b1 : enable
mipi_control_0	G	04	[6]	1'b0	RW	aev	**mipi_en** MIPI enable 1'b0 : disable 1'b1 : enable
	G	04	[4]	1'b1	RW	aev	**clk_hs_mode** MIPI clock lane hs mode 1'b0 : LP & HS mode 1b'1 : only HS mode
otp_ctrl0	H	04	[5]	1'b0	RW		**otp_mode_read** OTP memory read mode 1'b0 : disable 1'b1 : enable
	H	04	[4]	1'b0	RW		**otp_mode_write** OTP memory write mode 1'b0 : disable 1'b1 : enable
isp_func_0	I	4	[7]	1'b1	RW		**wdpc_en_l** DPC enable control 1'b0 : disable 1'b1 : enable
	I	4	[6]	1'b1	RO		**bdpc_en_l** DPC enable control 1'b0 : disable 1'b1 : enable

	I	4	[5]	1'b1	RO	**wdpc_en_s** DPC enable control 1'b0 : disable 1'b1 : enable
	I	4	[4]	1'b1	RO	**bdpc_en_s** DPC enable control 1'b0 : disable 1'b1 : enable
	I	4	[3]	1'b1	RO	**wdpc_en_vs** DPC enable control 1'b0 : disable 1'b1 : enable
	I	4	[2]	1'b1	RO	**bdpc_en_vs** DPC enable control 1'b0 : disable 1'b1 : enable
isp_func_4	I	8	[7]	1'b0	RW	**otp_dpc_en_l** OTP DPC enable control 1'b0 : disable 1'b1 : enable
	I	8	[6]	1'b0	RW	**otp_dpc_en_s** OTP DPC enable control 1'b0 : disable 1'b1 : enable
	I	8	[5]	1'b0	RW	**otp_dpc_en_vs** OTP DPC enable control 1'b0 : disable 1'b1 : enable
pfr_ctrl0	J	31	[1]	1'b0	RW	**ren_l** PFR enable control 1'b0 : disable 1'b1 : enable
	J	31	[0]	1'b0	RW	**ben_l** PFR enable control 1'b0 : disable 1'b1 : enable
pfr_ctrl1	J	32	[7:5]	3'h0	RW	**fading_l** PFR fading control
	J	32	[4:0]	5'h00	RW	**slope_l** PFR slope control
pfr_ctrl2	J	33	[1]	1'b0	RW	**ren_s** PFR enable control 1'b0 : disable 1'b1 : enable
	J	33	[0]	1'b0	RW	**ben_s** PFR enable control 1'b0 : disable 1'b1 : enable
pfr_ctrl3	J	34	[7:5]	3'h0	RW	**fading_s** PFR fading control
	J	34	[4:0]	5'h00	RW	**slope_s** PFR slope control
	J	35	[1]	1'b0	RW	**ren_vs**

pfr_ctrl4						PFR enable control 1'b0 : disable 1'b1 : enable
	J	35	[0]	1'b0	RW	**ben_vs** PFR enable control 1'b0 : disable 1'b1 : enable
pfr_ctrl5	J	36	[7:5]	3'h0	RW	**fading_vs** PFR fading control
	J	36	[4:0]	5'h00	RW	**slope_vs** PFR slope control

Revision history

Version	Date	Description
V0.1	2023.02.02	Initial Release
V0.2	2023.04.12	Updated DC characteristic, Power consumption
V0.3	2023.06.13	Updated optical specification, DC characteristic, Power consumption