S5K4ECGX

1/4" 5Mp CMOS Image Sensor SOC with an Embedded Image Signal Processor

Revision 0.01

August 2010

Application Note

SAMSUNG ELECTRONICS

SAMSUNG

Important Notice

The information in this publication has been carefully checked and is believed to be entirely accurate at the time of publication. Samsung assumes no responsibility, however, for possible errors or omissions, or for any consequences resulting from the use of the information contained herein.

Samsung reserves the right to make changes in its products or product specifications with the intent to improve function or design at any time and without notice and is not required to update this documentation to reflect such changes.

This publication does not convey to a purchaser of semiconductor devices described herein any license under the patent rights of Samsung or others.

Samsung makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Samsung assume any liability arising out of the application or use of any product or circuit and specifically disclaims any and all liability, including without limitation any consequential or incidental damages.

"Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals" must be validated for each customer application by the customer's technical experts.

Samsung products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, for other applications intended to support or sustain life, or for any other application in which the failure of the Samsung product could create a situation where personal injury or death may occur.

Should the Buyer purchase or use a Samsung product for any such unintended or unauthorized application, the Buyer shall indemnify and hold Samsung and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, expenses, and reasonable attorney fees arising out of, either directly or indirectly, any claim of personal injury or death that may be associated with such unintended or unauthorized use, even if such claim alleges that Samsung was negligent regarding the design or manufacture of said product.

S5K4ECGX, 1/4" 5Mp CMOS Image Sensor SOC with an Embedded Image Signal Processor
Application Note, Revision 0.01

Samsung Electronics Co., Ltd.
San #24 Nongseo-Dong, Giheung-Gu
Yongin-City, Gyeonggi-Do, Korea 446-711

Contact Us: *jm43.an@samsung.com*
TEL: (82)-(31)-209-8517
FAX: (82)-(31)-209-3262

Home Page: *http://www.samsungsemi.com*

Printed in the Republic of Korea

Revision History

Revision No.	Date	Description	Refer to	Author(s)
0.00	05, 24, 2010	- Initial draft	all	JM Ahn
0.01	08, 28, 2010	- Modify	all	JM Ahn

Table of Contents

List of Figures

List of Tables

List of Examples

1 SOC REQUIREMENT

1.1 OVERVIEW

The S5K4ECGX is a highly integrated 5Mp camera chip which includes a CMOS image sensor and an image signal processor with 8-bit ITU-R.656/601 parallel and MIPI CSI2 compliant serial interfaces. It is fabricated by SAMSUNG 0.09um CMOS image sensor process targeted for high-efficiency and low-power image sensors.

The CMOS image sensor consists of a 2608x1960 active pixel sensor (APS) array in 1/4 inch optical format. It has an on-chip 10-bit ADC array, and correlated double sampling (CDS) which reduce fixed pattern noise (FPN) significantly.

The image signal processor performs sophisticated image processing functions including color recovery and correction, false color suppression, lens shading correction, noise removal, edge enhancement, programmable gamma correction, image down scaling, auto defect correction, auto dark level compensation, auto flicker correction (50/60Hz), auto exposure (AE), auto white balance (AWB) and auto focus (AF). The auto functions are preformed by F/W on an embedded RISC processor. The host controller is able to access and control this devices via a general I2C bus.

1.2 FEATURES

1.2.1 IMAGE SENSOR

- Optical Format : 1/4 inch
- Unit Pixel Size : 1/4um
- Effective Resolution : 2608(H) x 1960(V)
- Active Resolution : 2592(H) x 1944(V)
- Color filter : RGB Bayer pattern
- Shutter type : Electronic rolling shutter
- Max. Capture frame rate : 15fps @full resolution
- Max. Video frame rate : 120fps @QVGA
- Max. Pixel clock frequency : 92MHz(JPEG)
- ADC accuracy : 10bit
- Progressive scan readout
- Window panning & cropping
- Vertical flip and horizontal mirror mode
- Continuous and single frame capture mode
- Frame rate control

1.2.2 IMAGE SIGNAL PROCESSOR

- Color interpolation and correction
- False color suppression
- Lens shading correction
- Noise removal
- Edge enhancement
- Scaler for preview and capture (3M and smaller at step 2)
- Programmable gamma correction
- Auto defect correction
- Auto dark level compensation
- Auto anti flicker correction(50/60Hz)
- Auto exposure (AE)
- Auto white balance (AWB)
- Auto focus (AF)
- Built-in test image generation
- Special effects
- 10bit parallel video interface, 8bit ITU-R.656/601
- MIPI CSI2 (dual lane)
- Output formats: YUV422, RGB565, RGB888, RAW8, RAW10, JPEG, Interleaved JPEG8 (SOSI, EOSI, SOEI, EOEI), Interleaved JPEG/Video (by data type).
- Integrated JPEG encoder and embedded SpeedTag for Fast JPEG management by Scalado SpeedView
- JPEG can be output in YUV422 format and support image rotation in JPEG.
- JPEG on-the-fly compression with embedded JPEG file size control.
- Standard frame spoof mode for JPEG8. Status and Pointers to interleaved video lines embedded in frame footer.

1.2.3 DEVICE

- Host control interface : I2C bus
- Internal PLL (24MHz to 100MHz input frequency)
- Operating temperature: -20℃ to +60℃
- Supply voltage : 2.8V for analog, 1.2V for digital core(with internal regulator), 1.8V ~ 2.8V for I/O
- 1.8V to 1.2V internal regulator

1.3 LOGICAL SYMBOL DIAGRAM

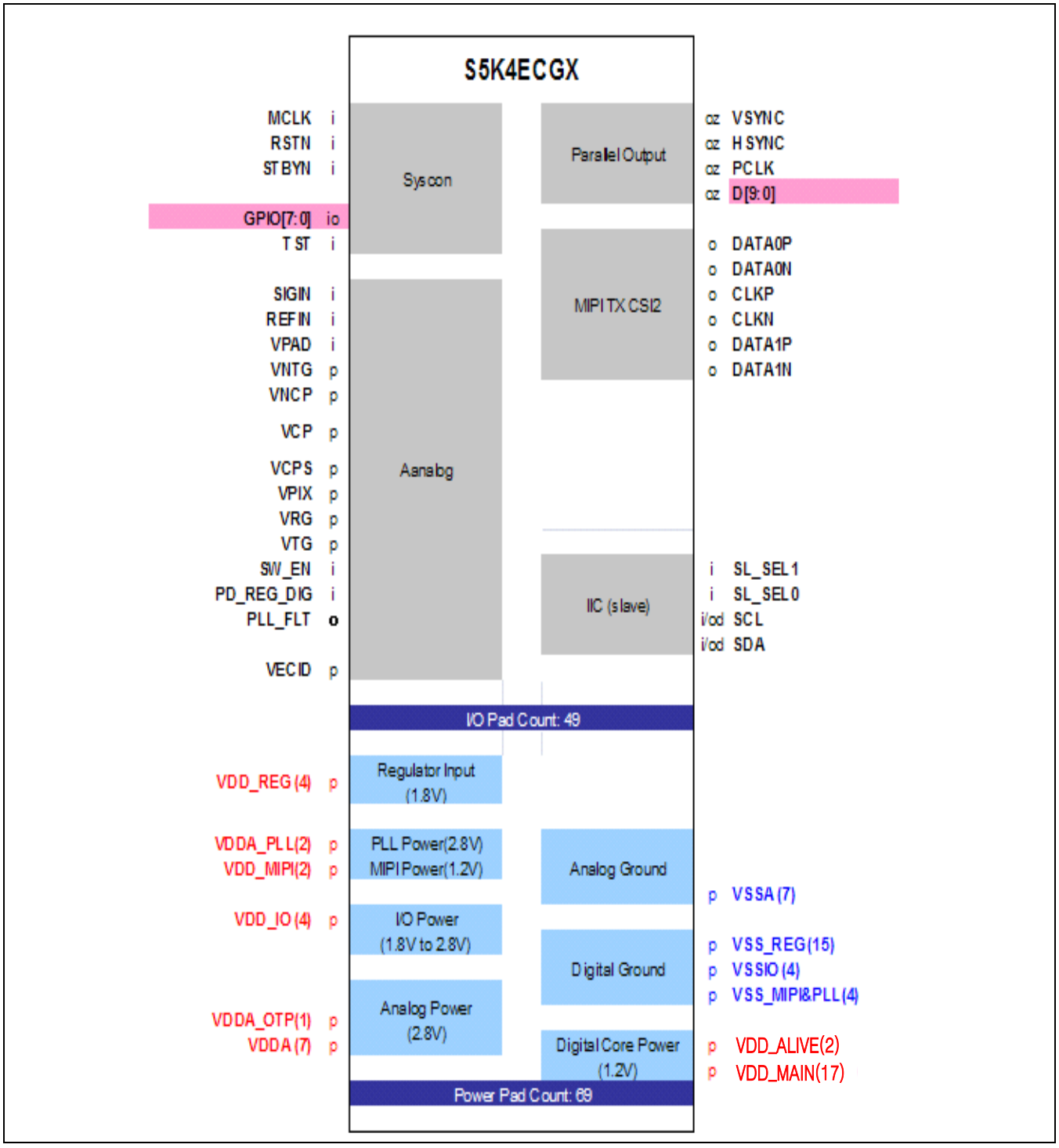

Figure 1-1 Logical Symbol Diagram

1.4 PAD CONFIGURATION

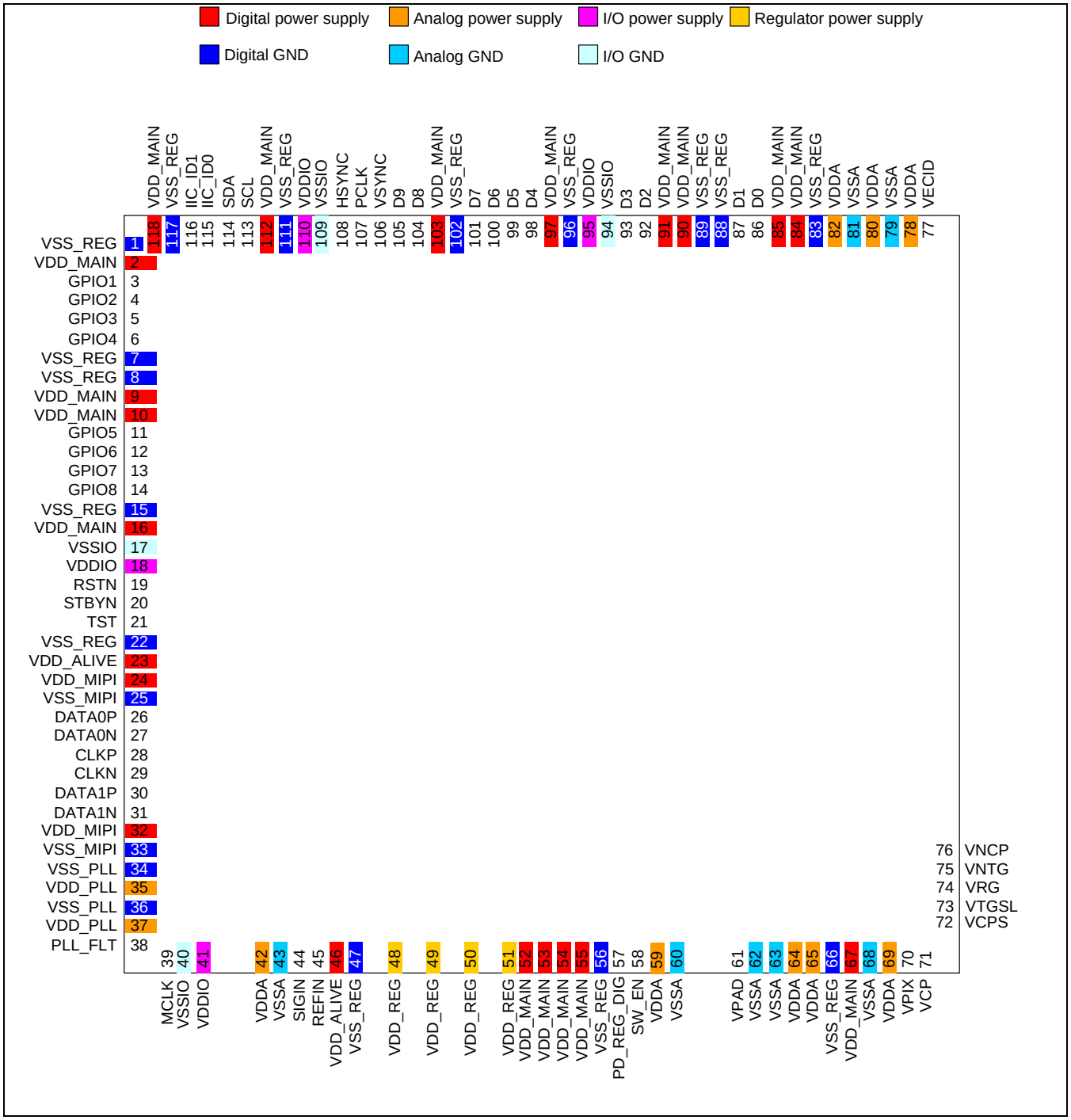

Figure 1-2 Pad Configuration

1.5 FUNCTIONAL BLOCK DIAGRAM

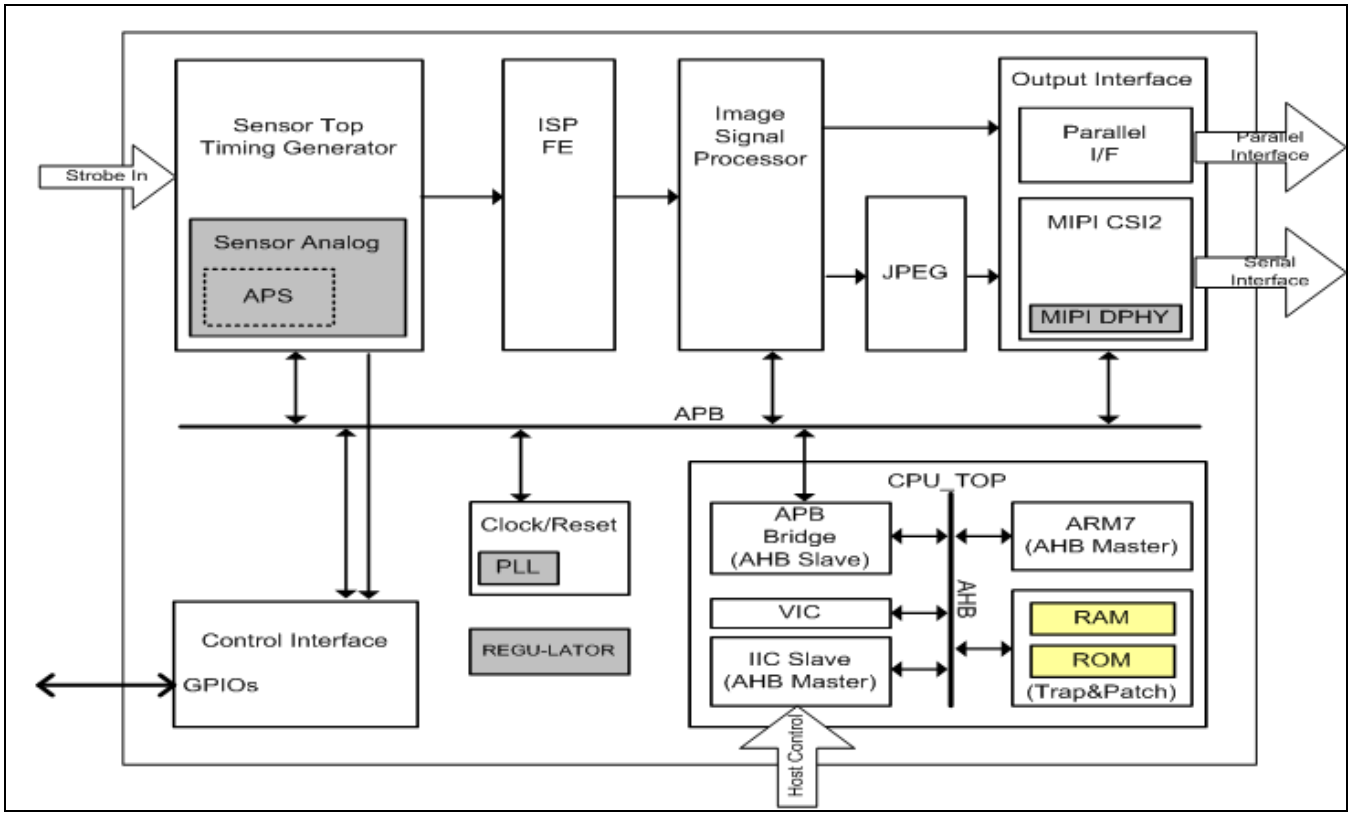

Figure 1-3 Functional Block Diagram

1.6 ISP BLOCK DIAGRAM

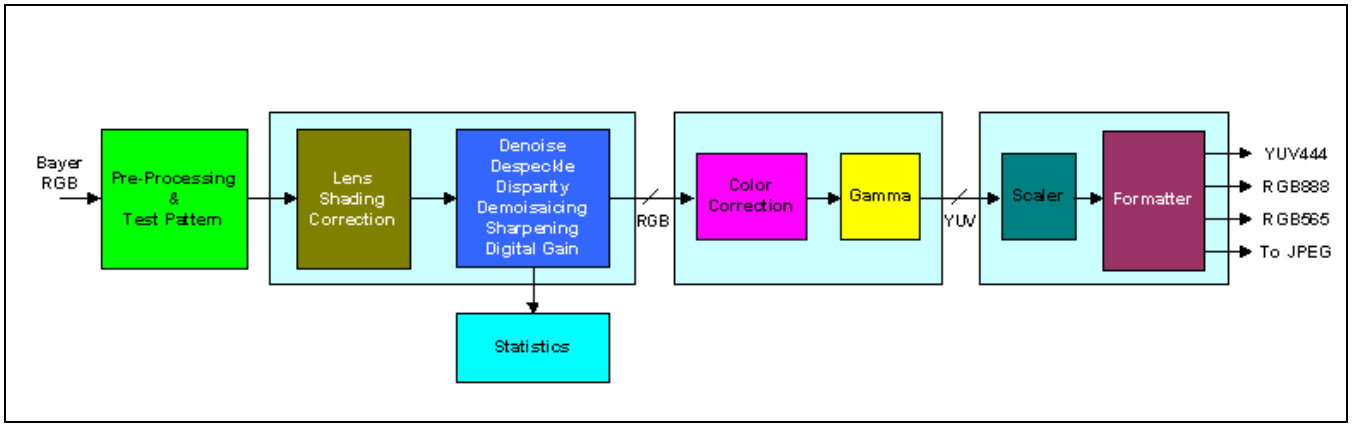

Figure 1-4 ISP Block Diagram

1.7 PIN DESCRIPTION

Pin No.	Name	I/O	A/D	Description
1	VSS_REG	GND	D	Digital ground
2	VDD_MAIN	Power	D	1.2V digital power supply
3	GPIO1	I/O	D	General purpose I/Os
4	GPIO2	I/O	D	
5	GPIO3	I/O	D	
6	GPIO4	I/O	D	
7	VSS_REG	GND	D	Digital ground
8	VSS_REG	GND	D	
9	VDD_MAIN	Power	D	1.2V digital power supply
10	VDD_MAIN	Power	D	
11	GPIO5	I/O	D	General purpose I/Os
12	GPIO6	I/O	D	
13	GPIO7	I/O	D	
14	GPIO8	I/O	D	
15	VSS_REG	GND	D	Digital ground
16	VDD_MAIN	Power	D	1.2V digital power supply
17	VSSIO	GND	D	I/O ground
18	VDDIO	Power	D	I/O power supply(1.8V ~ 2.8V)
19	RSTN	I	D	Master reset(Active low)
20	STBYN	I	D	Hardware standby mode (Active low)
21	TST	I	D	Test pin. NC
22	VSS_REG	GND	D	Digital ground
23	VDD_ALIVE	Power	D	1.2V digital power supply
24	VDD_MIPI	Power	D	1.2V digital power supply
25	VSS_MIPI	GND	D	Digital ground
26	DATA0P	O	A	CSI-2 Tx data lane 0 positive.
27	DATA0N	O	A	CSI-2 Tx data lane 0 negative.
28	CLKP	O	A	CSI-2 Tx clock positive
29	CLKN	O	A	CSI-2 Tx clock negative
30	DATA1P	O	A	CSI-2 Tx data lane 1 positive
31	DATA1N	O	A	CSI-2 Tx data lane 1 negative
32	VDD_MIPI	Power	D	1.2V digital power supply
33	VSS_MIPI	GND	D	Digital ground

34	VSS_PLL	GND	D	Digital ground
35	VDD_PLL	Power	A	2.8V analog power supply
36	VSS_PLL	GND	D	Digital ground
37	VDD_PLL	Power	A	2.8V analog power supply
38	PLL_FLT	O	A	Analog test. NC
39	MCLK	I	D	External clock.
40	VSSIO	GND	D	I/O ground
41	VDDIO	Power	D	I/O power supply(1.8V ~ 2.8V)
42	VDDA	Power	A	2.8V analog power supply
43	VSSA	GND	A	Analog ground
44	SIGIN	I	A	Analog test. NC
45	REFIN	I	A	Analog test. NC
46	VDD_ALIVE	Power	D	1.2V digital power supply
47	VSS_REG	GND	D	Digital ground
48	VDD_REG	Power	D	1.8V Regulator power supply
49	VDD_REG	Power	D	
50	VDD_REG	Power	D	
51	VDD_REG	Power	D	
52	VDD_MAIN	Power	D	1.2V digital power supply
53	VDD_MAIN	Power	D	
54	VDD_MAIN	Power	D	
55	VDD_MAIN	Power	D	
56	VSS_REG	GND	D	Digital ground
57	PD_REG_DIG	I	A	Regulator power down.(Active high)
58	SW_EN	I	A	Regulator switch enable.(Active high)
59	VDDA	Power	A	2.8V analog power supply
60	VSSA	GND	A	Analog ground
61	VPAD	I	A	Analog test. NC
62	VSSA	GND	A	Analog ground
63	VSSA	GND	A	
64	VDDA	Power	A	2.8V analog power supply
65	VDDA	Power	A	
66	VSS_REG	GND	D	Digital ground
67	VDD_MAIN	Power	D	1.2V digital power supply
68	VSSA	GND	A	Analog ground
69	VDDA	Power	A	2.8V analog power supply
70	VPIX	O	A	Connect to 2.8V analog power
71	VCP	O	A	Analog test. External cap 0.1uF

72	VCPS	O	A	Analog test. NC
73	VTGSL	O	A	Analog test. NC
74	VRG	O	A	Analog test. NC
75	VNTG	O	A	Analog test. NC
76	VNCP	O	A	Analog test. NC
77	VECID	I	A	Analog test (High voltage input for OTP memory write). NC
78	VDDA	Power	A	2.8V analog power supply
79	VSSA	GND	A	Analog ground
80	VDDA	Power	A	2.8V analog power supply
81	VSSA	GND	A	Analog ground
82	VDDA	Power	A	2.8V analog power supply
83	VSS_REG	GND	D	Analog ground
84	VDD_MAIN	Power	D	1.2V digital power supply
85	VDD_MAIN	Power	D	
86	D0	O	D	Pixel data output
87	D1	O	D	Pixel data output
88	VSS_REG	GND	D	Digital ground
89	VSS_REG	GND	D	
90	VDD_MAIN	Power	D	1.2V digital power supply
91	VDD_MAIN	Power	D	
92	D2	O	D	Pixel data output
93	D3	O	D	Pixel data output
94	VSSIO	GND	D	I/O ground
95	VDDIO	Power	D	I/O power supply(1.8V ~ 2.8V)
96	VSS_REG	GND	D	Digital ground
97	VDD_MAIN	Power	D	1.2V digital power supply
98	D4	O	D	Pixel data output
99	D5	O	D	Pixel data output
100	D6	O	D	Pixel data output
101	D7	O	D	Pixel data output
102	VSS_REG	GND	D	Digital ground
103	VDD_MAIN	Power	D	1.2V digital power supply
104	D8	O	D	Pixel data output
105	D9	O	D	Pixel data output (D9 ~ D0 : 10bit data, D9 ~ D2 : 8bit data)
106	VSYNC	O	D	Vertical sync for parallel interface
107	PCLK	O	D	Pixel clock output for parallel interface
108	HSYNC	O	D	Horizontal sync for parallel interface
109	VSSIO	GND	D	I/O ground

110	VDDIO	Power	D	I/O power supply(1.8V ~ 2.8V)
111	VSS_REG	GND	D	Digital ground
112	VDD_MAIN	Power	D	1.2V digital power supply
113	SCL	I/O	D	I2C slave clock
114	SDA	I/O	D	I2C slave data
115	IIC_ID0	I	D	I2C slave address selection.
116	IIC_ID1	I	D	[IIC_ID1,IIC_ID0] = 00 : 0x78, 01 : 0x7A, 10 : 0x5A(default), 11 : 0xAC
117	VSS_REG	GND	D	Digital ground
118	VDD_MAIN	Power	D	1.2V digital power supply

1.8 CONTROL INTERFACE DESCRIPTION

The I2C interface is a two-wire bi-directional serial bus. Both wires (Serial Clock Line -SCL and Serial Data Line - SDA) are connected to a positive supply via a pull-up resistor, and when the bus is free both lines are high. The output stage of the device must have an open-drain or open collector type IO cell so that a wired-AND function between all devices that are connected on the bus can be performed.

The two-wire serial interface defines several different transmission stages as follows:

- A start bit
- The slave device 7-bit address
- An (No) acknowledge bit coming from slave.
- An 8-bit or 16-bit message (address and/or data).
- A stop bit (or another 8bit or 16bit message in multiple Read/Write access)

The data on the SDA pin must be stable during the high period of the clock (SCL) as shown in the figure below. Only the master may change the data while SCL is high. A high-to-low transition marks a START condition, and a low-to-high a STOP condition.

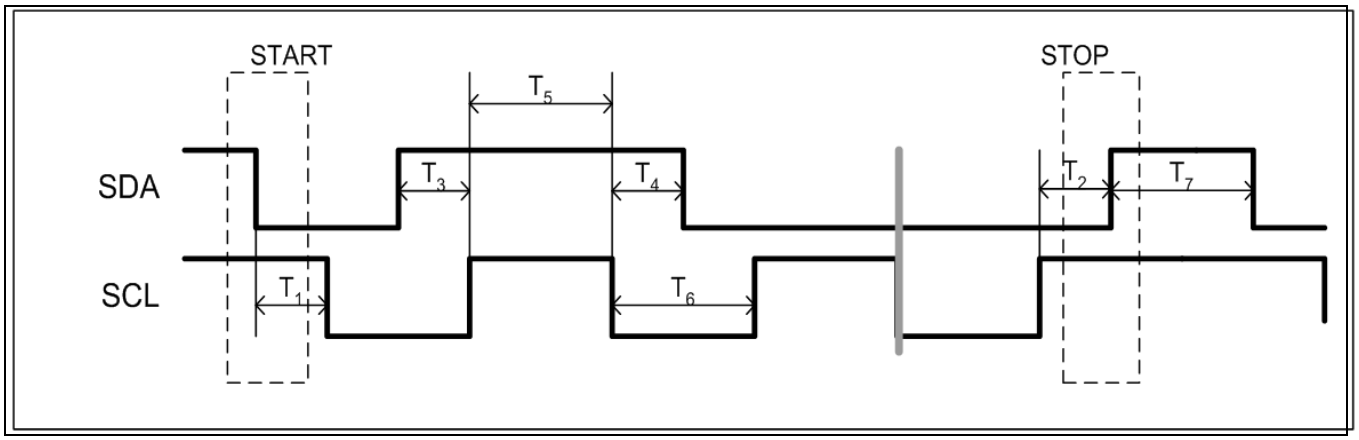

Figure 1-5 I2C General Timing Specification

Symbol	Description	Typical Mode		Fast Mode		Units
		Min.	Max	Min.	Max	
	SCL clock frequency	0	100	0	400	kHz
T1	Hold time for START condition	0.4	-	0.6	-	us
T2	Setup time for STOP condition	4.0	-	0.6	-	us
T3	Data setup time	250	-	160	-	ns
T4	Data hold time	0	3.45	0	0.9	us
T5	High period of the SCL clock	4.0	-	0.6	-	us
T6	Low period of the SCL clock	4.7	-	1.3	-	us

T7	Bus free time between STOP and START condition	4.7	-	1.3	-	us
	Rise time for both SDA and SCL signals		1000		300	ns
	Fall time for both SDA and SCL signals		300		300	ns
Cb	Capacitive load for each bus line		400		400	pF

The master device activates a START condition, and sends the first byte of data that contains the 7-bit address, and a direction bit (R/W#, 1 for read, 0 for write). The addressed device answers by pulling down the SDA line as an acknowledge procedure.

Detailed sequences of read and write data transfers are shown in the figures below.

- The colored boxes represent master-to-slave data transfer.
- The clear boxes represent slave-to-master data transfer.

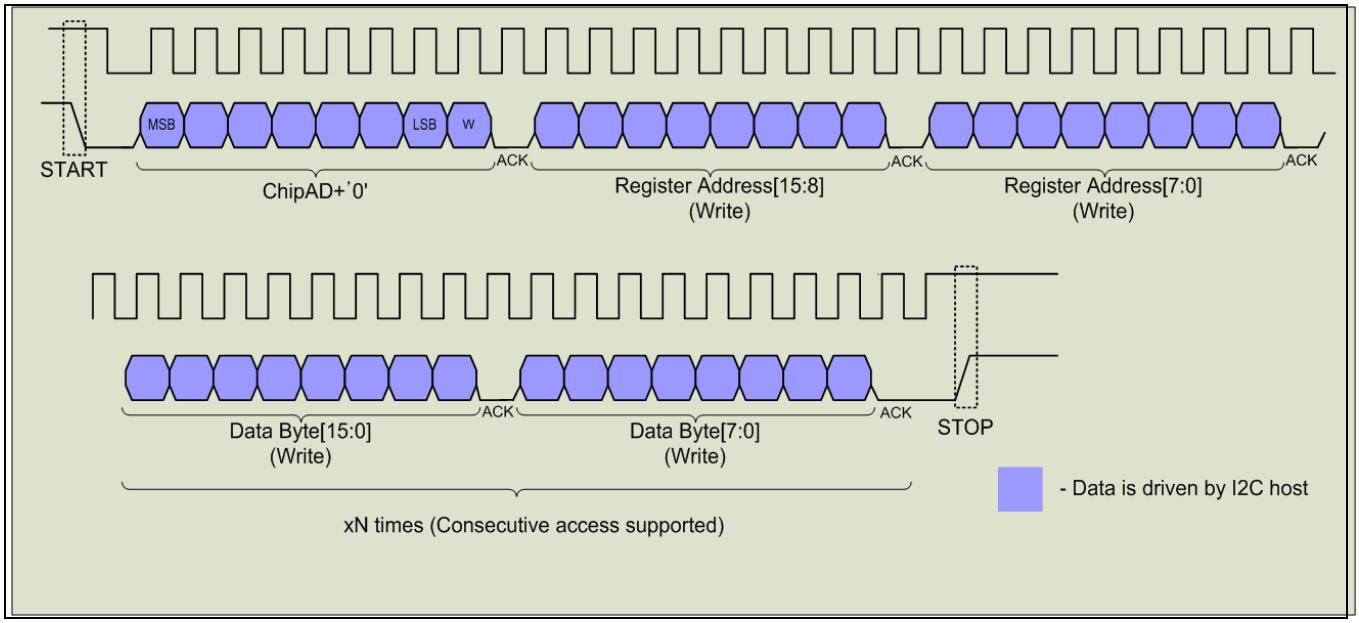

Figure 1-6 Example of I2C Write Timing (16 Address, 2 data bytes)

The device address can be changed by pin configuration of IIC_ID, which is described in Pad Description.

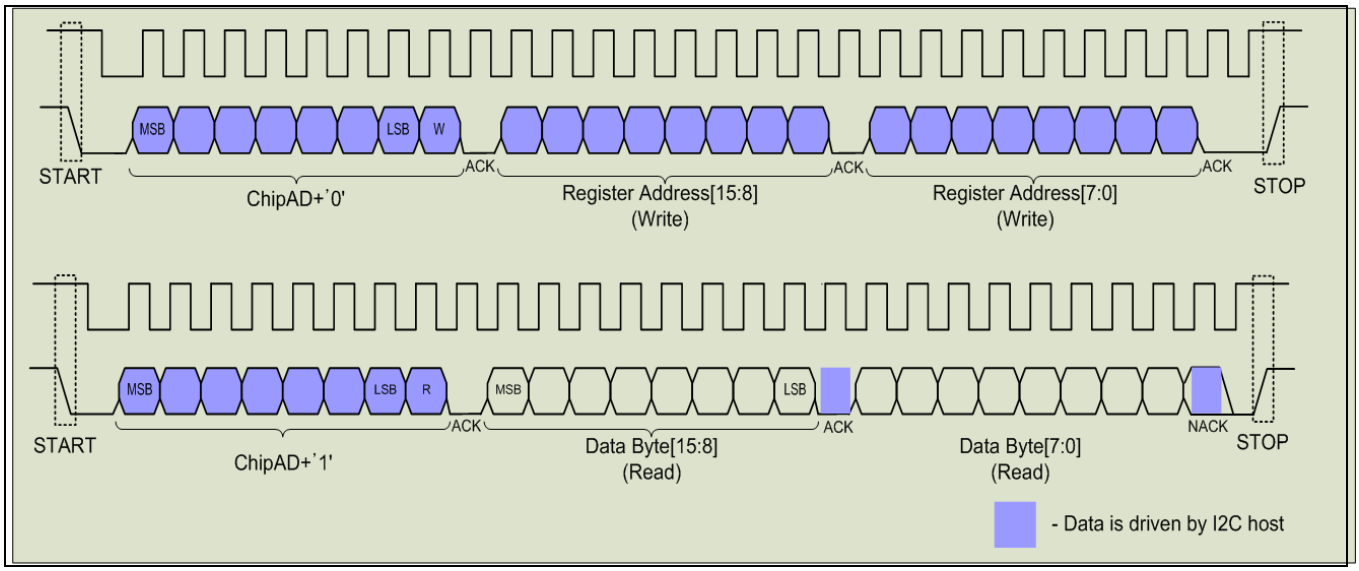

Figure 1-7 Example of I2C Single Read Timing (16 Address, 2 data bytes)

The device address can be changed by pin configuration of IIC_ID, which is described in Pad Description.

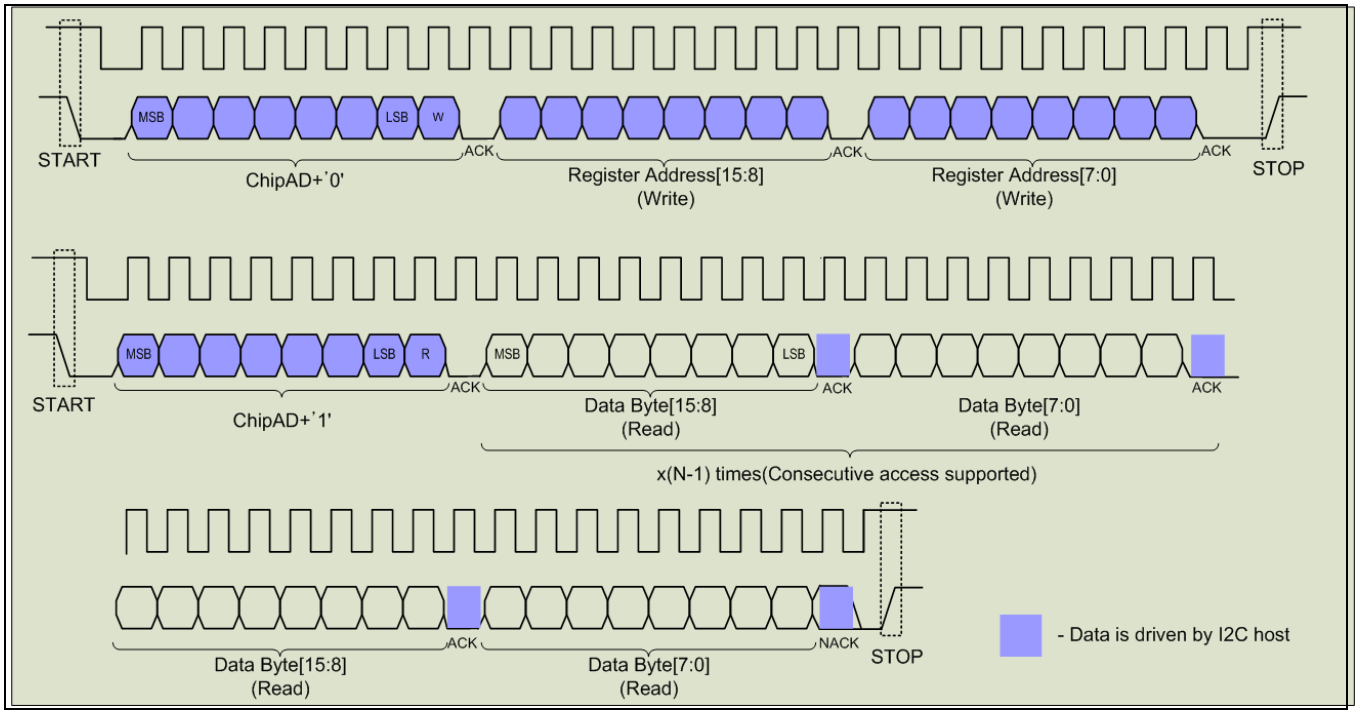

Figure 1-8 Example of I2C Multiple (N) Read Timing (16 Address, 2 data bytes)

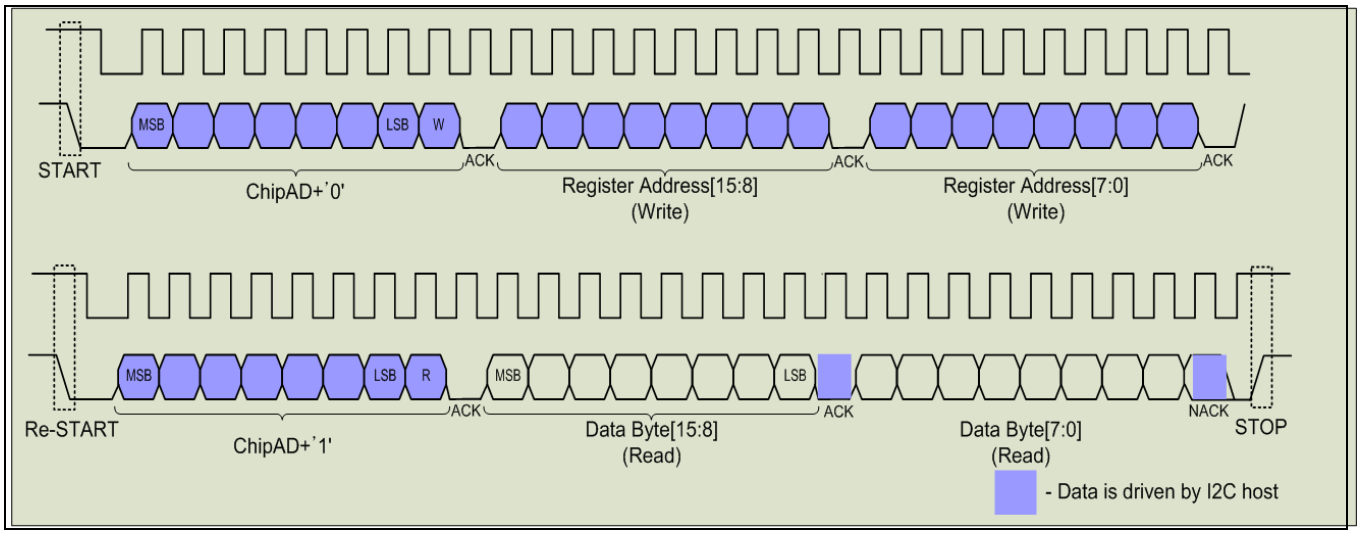

Figure 1-9 Example of I2C Single Read access with Repeated Start (16 Address, 2 data bytes)

Example of I2C Address

{IIC_ID_1, IIC_ID_0}	Device Address
00	0111_1000b/78h
01	0111_1010b/7Ah
10	0101_1010b/5Ah
11	1010_1100b/ACh

1.9 POWER UP/DOWN SEQUENCE

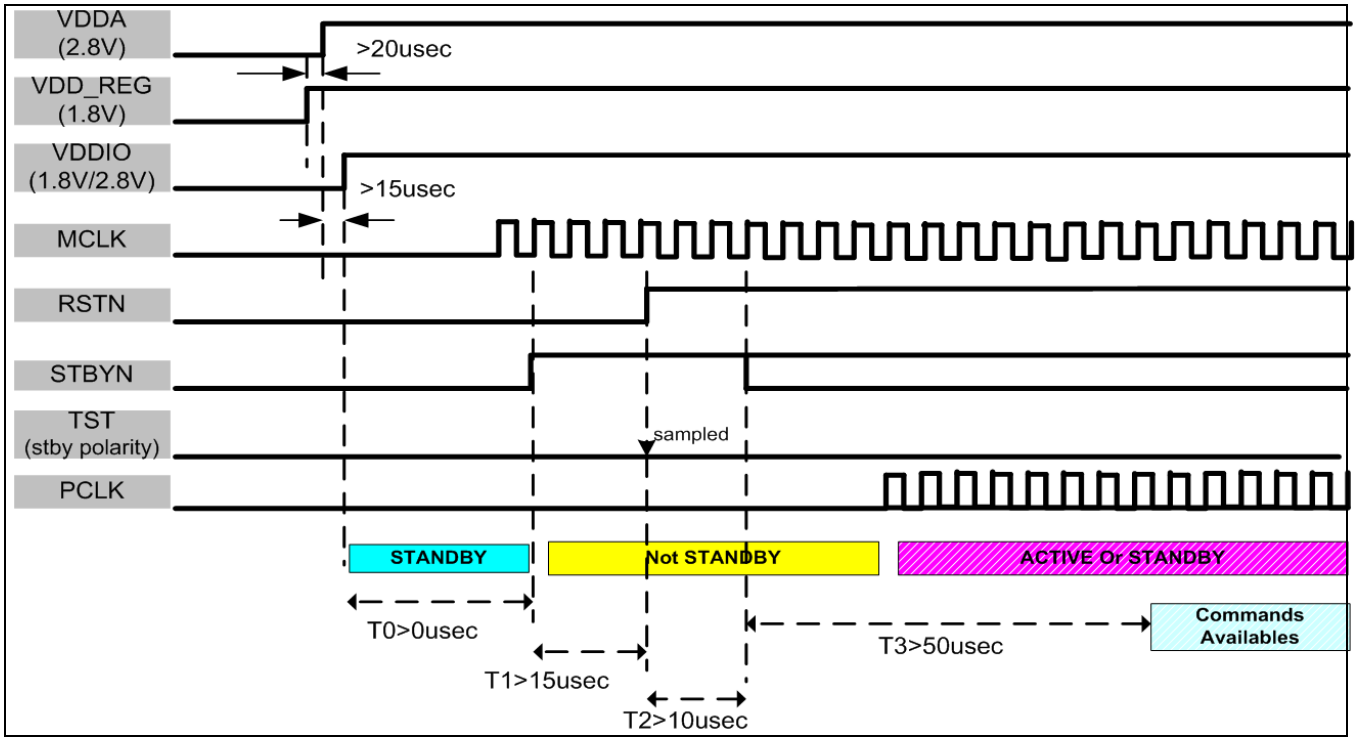

Figure 1-10 Power-Up Sequence

NOTE: If internal regulator is not used, VDD_REG apply to 1.2V.

- When STBYN=RSTN='0' then the chip in STANDBY state.
- STBYN pin de-asserted (at least 15usec before RSTN).
  - This time is required for Wake-Up process by HW (like main REG turn-on, exit from Fail-Safe and RESET signal propagation).
  - Then chip enter RESET state.
- RSTN goes to '1' (chip start init process).
- STBYN may stay '1' or change to '0' (at least 10usec after RSTN, for RESET signal propagation).
  - If stay at '1' - state is active, and STREAM ON is performed.
  - If change to '0' – chip enter to STANDBY mode by FW.
  - TST is STANDBY polarity control. When TST=0, STBYN is active low. When TST=1, STBYN is active high. TST pin state is sampled for STANDBY polarity when RSTN goes high.

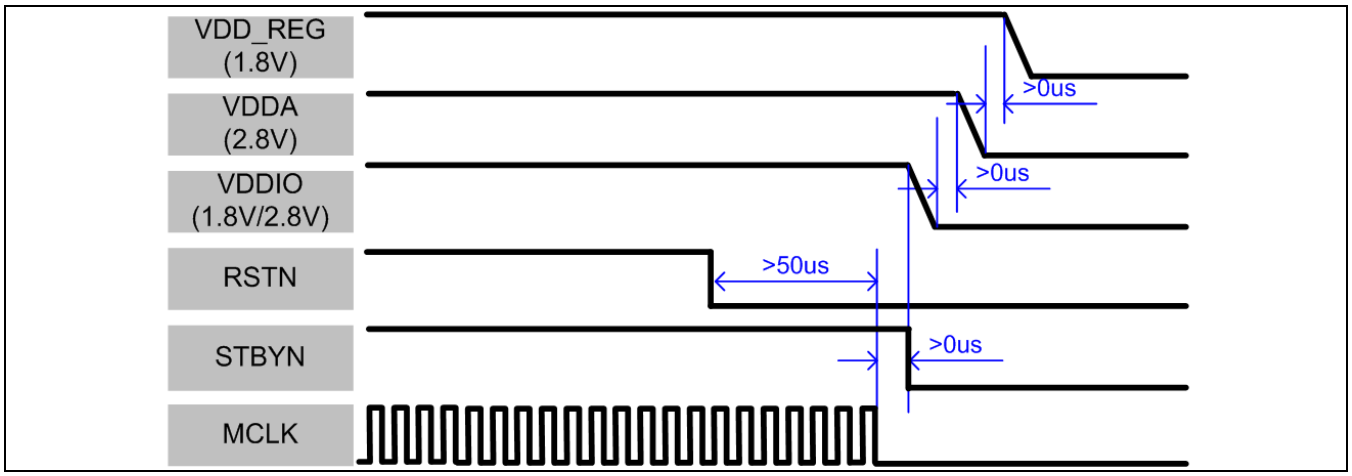

Figure 1-11 Power-down Sequence

NOTE: If internal regulator is not used, VDD_REG apply to 1.2V.

1.10 STANDBY SEQUENCE

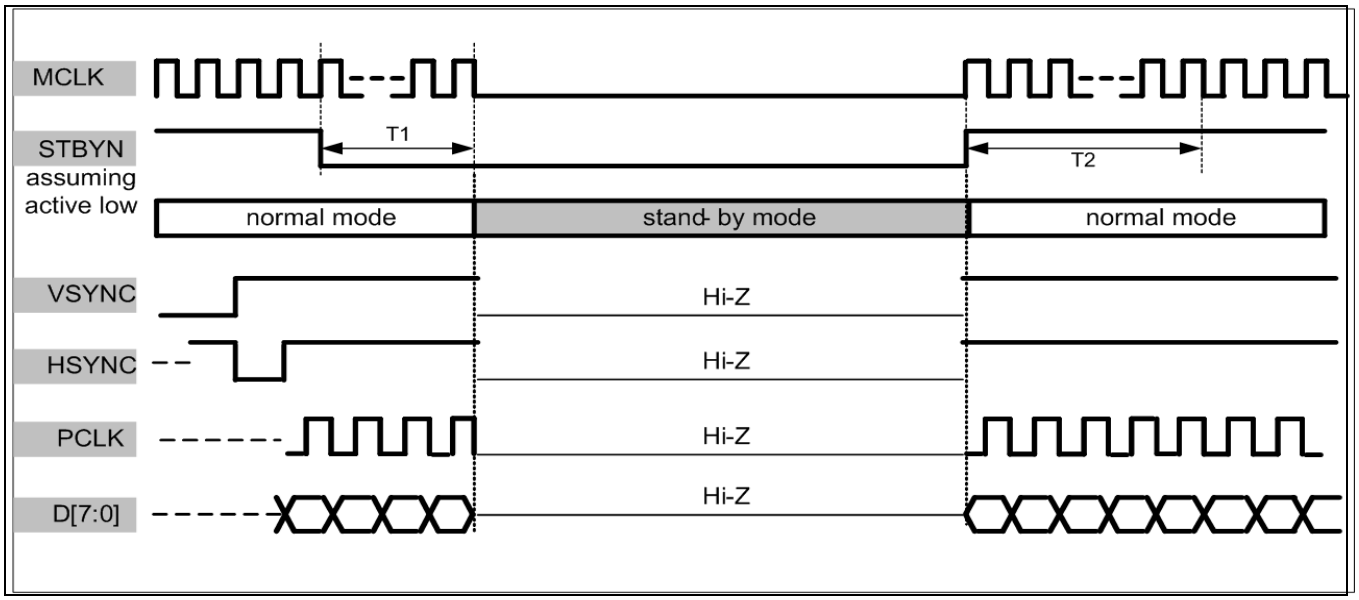

Figure 1-12 Standby/Wakeup waveform

1.10.1 ENTER STANDBY :

Assert STBYN pin ('0') → Stop MCLK (Optional)

1.10.2 EXIT STANDBY (WAKEUP)

Provide MCLK → De-assert STBYN pin ('1')

Basically, Two methods of standby are supported; One that has to use the STBYN pin (default) likes above and one alternative for a host that doesn't use the Asleep pin. In the default firmware we assume that the host uses the STBYN pin as part of the sleep sequence. If the host will not be using the STBYN pin it has to update a tuning parameter as follows: (S/W STBY D000107E 0001h)

This method of standby not using the STBYN pin is less efficient (takes a bit more power during sleep mode)

But, if the host doesn't have a available GPIO to connect to STBYN it can use this alternate sleep sequence.

1.10.3 HW STANDBY

If the internal regulator is not used, the HW stand-by does not support. But if the internal regulator is used, the power consumption is increased because the VDDD is increased.

1.10.4 BACKUP HW REGISTERS AFTER WAKEUP

Most of HW registers are reset after STBY mode except alive registers. To keep some registers which needs to be not changed after STBY mode , FW stores them with JPEG RTB memory right before system enters sleep mode. The stored registers will be restored to HW register as soon as wake-up.

Table 1-1 Register for Backup HW registers after wakeup

Mnemonic	Attr	Description
skl_usStbyBackupReg[X][Y]	R/W	First index relates to start address of HW registers to backup. Second index relates to size How many registers shoud be saved from start address

1.11 IMAGE SIGNAL PROCESSING OVERVIEW

1.11.1 AUTO EXPOSURE (AE)

The embedded AE control algorithm measures the luminance level of selected windows (ROI), and compares it with the AE target value, which is adaptive to the scene type and rendering options. The image brightness is adjusted by controlling analog and digital gains and image sensor integration time. The algorithm has various options to control convergence speed so as to handle illumination changes.

1.11.2 AUTO WHITE BALANCE (AWB)

The AWB algorithm adjusts image colors to best match human perceptions by controlling R, G, and B digital gain. The algorithm uses various statistics channels – warm, outdoor, low brightness, general, special color and so on. Each statistics channel applies a distinctive filter on normalized r and b plane and the filtered data are used for illuminant estimation. The algorithm has also a scene type detector which improves the accuracy of the illuminant estimation.

1.11.3 AUTO FOCUS (AF)

The auto-focus algorithm analyzes sharpness data gathered from the image, and determines the direction and distance for the lens to move in order to achieve a focused image. Movement commands are sent to a lens actuator driver. Special treatments are applied to saturated image areas. Both automatic and manual operation modes are supported. During a still image capture, the focus is adjusted further to an optimal location following a shutter button half-press. Once achieved, the optimal focus is locked and additional information regarding the confidence rate of the AF algorithm convergence is also passed to the host processor. In the manual mode, the lens position may be controlled by the user. The auto-focus algorithm could issue direct commands to an actuator driver IC via I2C, PWM or stepping-motor interfaces. Options for searching (Full/Peak-Detection, Global/Local, Coarse/Fine and so on) and device-dependant actuator characteristics compensation are also implemented.

1.11.4 AUTO FLICKER CORRECTION (AFC)

Flickers may occur when the sensor integration time is not an integral multiple of the half cycle of the electrical supply of the predominant illumination. For example, under a 50Hz or 60Hz fluorescent lamp, flicker could appear if the integration time is smaller than 1/100 or 1/120, respectively. The algorithm can detect the frequency of the illumination and adjust the integration time automatically and hence avoid the flicker from appearing.

1.11.5 LENS SHADING CORRECTION

Two complimentary methods of shading correction are used– one uses parabolic shading compensation, and the other one applies a grid model to remove residual effects. Shading correction is also adaptive to illuminations.

1.11.6 COLOR INTERPOLATION

RGB values at each pixel location of the Bayer plane are derived from a group of neighboring pixels. The algorithm uses multiple approaches such as text and natural modes. Distinctive decisions are made for each pixel in the image.

1.11.7 COLOR CORRECTION

A variety of color profiles are used for color representation improvement. The decision about the profile is taken based on scene brightness and illumination type. Color correction is done using non-linear transformation.

1.11.8 DEFECT PIXEL CORRECTION

This algorithm detects and corrects from isolated single bad pixels to clusters with 3 hot bad pixels on the raw image data on the fly. Maximum hot pixels could be configured from 1 to 3 and cold pixels from 1 to 2.

1.11.9 DENOISING

The denoising algorithm implements an "edge-preserving smoothing" algorithm. It averages pixels that are close in value to the central pixel. Neighboring pixels are equalized before averaging. The denoising algorithm features a slope estimation feature, enabling efficient noise reduction on edges and smooth shades. Denoising power can change radially to provide adequate effect as a function of lens shading compensation.

1.11.10 GAMMA CORRECTION

Gamma correction tables are used for the following color components:

- R, G and B – Three channels independently

Gamma settings can change dynamically as a function of illumination intensity, contrast ratio and noise index.

1.11.11 IMAGE DOWN-SCALING

The image from the sensor can be downscaled to an arbitrary even size smaller than or equal to 3M by same ratios in X and Y dimensions, with an accuracy within a margin of 3 to 5 pixels. More precise output sizes can be achieved by cropping. Among other resolutions, QXGA, UXGA, SXGA, SVGA, VGA, QVGA, QQVGA, CIF, and QCIF resolutions are supported. To increase frame rate, averaged sub-sampled scaling is also supported for output sizes of SVGA and below.

1.11.12 SPECIAL EFFECTS

The special effects may be used to create a Sepia (warm tone), Aqua (cool tone), Monochrome, Negative mono, Negative color, Sketch effect on the image.

1.11.13 OUTPUT FORMATTING

The ISP outputs 8-bit processed video data in the form of standard YUV ITU-R.656/601 or RGB data. Raw sensor data in Bayer format may also be outputted with 8-10 bit accuracy.

1.11.14 IMAGE RENDITION CONTROL OPTIONS

There are a number of image rendition related controls available to the host, such as Brightness, Contrast, Saturation, configurable sharpness, and Glamour and so on.

2 OPERATION

2.1 SET FILE DESCRIPTION

Generally, the set file is composed as following.

a. System initialization (ARM go / Trap & Patch)

b. Analog/APS setting

c. Operating configurations (PLL / preview & capture)

d. ISP tuning value

2.1.1 SET FILE TYPE

2.1.1.1 .TSET

```
0028    PAGE

002A    ADDRESS

0F12    DATA
```

For example, if the user wants to send "ARM go" command, user can write as below.

```
0028    D000            //HW register page

002A    0014            //"ARM go" command address

0F12    0001            //enable "ARM go"
```

2.1.1.2 .NSET

```
WRITE PAGE&ADDRESS     DATA
```

For example, if the user wants to send "ARM go" command, user can write as below.

```
WRITE D0000014 0001
```

2.1.2 MNEMONIC & NUMERICAL ADDRESS

There are two kinds of Address type, Mnemonic and Numerical. Mnemonic address means some registers have the "name" of theirs, so it is useful when user doesn't know its exact numerical address.

In General, HW registers have Numerical address only, but SW registers have Mnemonic including Numerical.

For example, if the user would like to change AE target, below two commands are the same.

```
WRITE   #TVAR_ae_BrAve 0030    //using mnemonic address, "#" is identifier

WRITE   700013C8       0030    //using numerical address
```

2.1.3 MANDATORY SETTINGS

```
1030    0000           // Clear host interrupt so main will wait

0014    0001           // ARM go

p100                   // 100mSec delay

0028    7000           // start add MSW

{T&P part, if apply, should be placed here}

1000    0001           // Set host interrupt so main start run
```

2.1.4 TRAP & PATCH (T&P)

T&P start address is defined in register <skl_ulFreeRamAddress>.

T&P size is defined in register <skl_usFreeRamSize>.

Caution: Do not modify these settings without FAE's confirmation.

2.2 PLL SETTING

Clock setup is performed in two phases. The first phase is performed during firmware initialization. The second phase is performed during each configuration enable.

2.2.1 INITIALIZATION PARAMETERS

Mnemonic	Description
REG_TC_IPRM_InClockLSBs	Input clock in KHz (lower 16 bit)
REG_TC_IPRM_InClockMSBs	Input clock in KHz (upper 16 bit)

The initialization parameters define the external input clock frequency in KHz. This value is used by the firmware to calculate the PLL multiplication/division for obtaining the system clock and output clock.

For example, for an input clock of 24MHz (24000KHz):

<REG_TC_IPRM_InClockLSBs>= 0X5DC0

<REG_TC_IPRM_InClockLSBs>= 0X0000

Mnemonic	Description
REG_TC_IPRM_UseNPviClocks	Number of PLL configurations to be computed (1-3)
REG_TC_IPRM_UseNMipiClocks	Number of MIPI configurations to be computed (1-3)

These parameters define the number of system/output clock sets to be calculated by the firmware. Up to three system/output clock sets can be configured.

For example, for two sets of PVI system/output clocks and one set of MIPI configuration:

<REG_TC_IPRM_UseNPviClocks>= 0X0002

<REG_TC_IPRM_UseNMipiClocks>= 0X0001

Mnemonic	Description
REG_TC_IPRM_bBlockInternalPllCalc	Use external PLL settings rather than internal FW calculation

If <REG_TC_IPRM_bBlockInternalPllCalc> =0X0001, then PLL configuration will not be performed by the FW. When using this option, refer to the document PLL setting guide. Currently, the firmware PLL calculation algorithm and the PLL HW write function were found to work as expected and hence it is not needed or recommended to use the bypass options.

There are also three sets of registers allowing three different clock configurations, where each set has the following registers (X=0,1,2):

Mnemonic	Description
REG_TC_IPRM_OpClk4KHz_X	First system clock frequency in KHz divided by 4
REG_TC_IPRM_MinOutRate4KHz_X	Minimal output rate of first clock in KHz divided by 4
REG_TC_IPRM_MaxOutRate4KHz_X	Maximal output rate of first clock in KHz divided by 4

During this phase, the firmware attempts to find a suitable setup for the internal dividers and multipliers of the clock generator unit that will satisfy the requested system clock and an output frequency that is between the minimal and maximal output frequencies. Completion of this phase ensures the system can output at the required maximal frame rate.

Giving a range that is too small may result in the FW returning an error because it cannot find a combination of division/multiplication to provide an output clock in that range.

2.2.2 PREVIEW CONFIGURATIONS

Each preview/capture configuration has three registers (shown are the registers for the first preview):

Mnemonic	Description
REG_0TC_PCFG_usMaxOut4KHzRate	Maximal output rate in KHz divided by 4
REG_0TC_PCFG_usMinOut4KHzRate	Minimal output rate in KHz divided by 4
REG_0TC_PCFG_uClockInd	System clock index (0-2)

During this phase, the output frequency is set such that it will be between the minimal and maximal rate, and can output the maximal frame rate required for the configuration (the maximal frame rate required for a configuration might be lower than maximal frame rate for the clock). In this phase, the firmware attempts to use a lower output rate for PVI output and a higher output rate for MIPI output, if possible, than the output rate specified in the first phase.

The minimum and maximum range of the output clock should be set as the minimum and maximum of the base-band.

The output clock range in the configurations MUST be equal to or bigger than the range given in the selected initialization. A smaller range may result in an error from the firmware if a frequency for the smaller range cannot be reached.

PLL must be used for configurations using binning modes.

2.2.3 PLL MANUAL SETTING

PLL setting of 4EC can be done by s/w as described in before chapter. That scheme only requires input clock, Mclk and output clock range from USER. But, this scheme has some limitation. If there are multiple solutions for the required clock, then s/w generally chooses the first searched one among them. In other words, there is no means to select special one among several solutions.

To cover this weakness, manual setting method for clock can be used.

The underlying idea of manual setting is that manually pre-calculated value for PLL and post dividers is used instead of the one which is calculated by s/w. To do this, PLL calculation by s/w should be prohibited and the registers where the results of PLL calculation are stored temporally are fulfilled the values which are manually calculated.

Mnemonic	Description
REG_TC_IPRM_bBlockInternalPllCalc	Use manual PLL settings rather than internal FW calculation
REGM_gSensorClocks[X1]_InputClk	input clock as same with MCLK
REGM_gSensorClocks[X]_PLL_PllHW_M	PLL multiplier M for system clock

1 ‘X’ means the order of clock configuration. There are 3 sets.

REGM_gSensorClocks[X]_PLL_PllHW_P	PLL pre-divider p for system clock
REGM_gSensorClocks[X]_PLL_PllHW_N	Post divider N for system clock
REGM_gSensorClocks[X]_PLL_PllHW_S	PLL S divider S for system clock
REGM_gSensorClocks[X]_PLL_PllHW_FinBits	PLL setting value depended on clock range
REGM_gSensorClocks[X]_PLL_PllHW_OIF_dphy_val	DPHY register value on MIPI mode
REGM_gSensorClocks[X]_usSysDiv	System clock divider = VCO / system clock
REGM_gSensorClocks[X]_PLL_usClkFreqDiv4	system clock
REGM_gSensorClocks[X]_PLL_PviPllHW_M	PLL multiplier M for output clock
REGM_gSensorClocks[X]_PLL_PviPllHW_P	PLL pre-divider p for output clock
REGM_gSensorClocks[X]_PLL_PviPllHW_N	Post divider N for output clock
REGM_gSensorClocks[X]_PLL_PviPllHW_S	PLL S divider S for output clock
REGM_gSensorClocks[X]_PLL_PviPllHW_FinBits	PLL setting value depended on clock range
REGM_gSensorClocks[X]_PLL_PviPllHW_OIF_dphy_val	DPHY register value on MIPI mode
REGM_gSensorClocks[X]_usOIFDenum	output clock divider = post divider N
REGM_gSensorClocks[X]_usPviFreqDiv4	output clock as same with PCLK

Suppose that Mclk, system clock and Pclk are 24MHz, 46MHz and 92MHz respectively. Then the values for PLL should be chosen. If P = 3, M = 92 and S = 3, then VCO output clock will be 24MHz x 92 / (3 x 2^3) = 92MHz. Because actually there is a 1/2 divider on the path for system clock, finally 92MHz / 2 = 46MHz can be obtained as system clock. Similarly, the setting values can be obtained for 92MHz as Pclk. At the same time, the values for PLL_PllHW_FinBits and PLL_PviPllHW_FinBits should be chosen based on system clock and Pclk respectively.

In addition, F/W of 4EC as old variables for internal use like usSysDiv and usOIFDenum. These can be calculated as following. usSysDiv = VCO clock of system PLL/ system clock = 736 MHz / 46 MHz = 16 and usOIFDenum = VCO clock of PCLK PLL / PCLK = 736 MHz / 92 MHz = 8. These values can be used in F/W so that these should be written in order for F/W to work properly.

Finally, Here is example set file.

Example 2-1 PLL Manual setting

```
//clk Settings
002A    #REG_TC_IPRM_bBlockInternalPllCalc
0F12    0001

002A #REGM_gSensorClocks[0]_InputClk // input clock          -> MCLK
0F12 5DC0

002A #REGM_gSensorClocks[0]_PLL_PllHW_M // PLL multiplier M for system clock
0F12 005c
002A #REGM_gSensorClocks[0]_PLL_PllHW_P // PLL pre-divider p for system clock
```

```
0F12 0003
002A #REGM_gSensorClocks[0]_PLL_PllHW_N // Post divider N for system clock
0F12 0001
002A #REGM_gSensorClocks[0]_PLL_PllHW_S // PLL S divider S for system clock
0F12 0003
002A #REGM_gSensorClocks[0]_PLL_PllHW_FinBits // PLL setting value depended on clock range
0F12 0007
002A #REGM_gSensorClocks[0]_PLL_PllHW_OIF_dphy_val // DPHY register value on MIPI mode
0F12 0000
002A #REGM_gSensorClocks[0]_usSysDiv // System clock divider = VCO / system clock
0F12 0010
002A #REGM_gSensorClocks[0]_PLL_usClkFreqDiv4 // system clock
0F12 2CEC

002A #REGM_gSensorClocks[0]_PLL_PviPllHW_M // PLL multiplier M for output clock
0F12 005c
002A #REGM_gSensorClocks[0]_PLL_PviPllHW_P // PLL pre-divider p for output clock
0F12 0003
002A #REGM_gSensorClocks[0]_PLL_PviPllHW_N // Post divider N for output clock
0F12 0001
002A #REGM_gSensorClocks[0]_PLL_PviPllHW_S // PLL S divider S for output clock
0F12 0003
002A #REGM_gSensorClocks[0]_PLL_PviPllHW_FinBits // PLL setting value depended on clock range
0F12 0007
002A #REGM_gSensorClocks[0]_PLL_PviPllHW_OIF_dphy_val // DPHY register value on MIPI mode
0F12 0000

002A #REGM_gSensorClocks[0]_usOIFDenum // output clock divider = post divider N

0F12 0001
002A #REGM_gSensorClocks[0]_usPviFreqDiv4 // output clock -> PCLK
0F12 59D8
```

2.3 OUTPUT FORMAT

The register below controls the format of Output image.

Table 2-1 Register setting for Output Format

Address	Attr.	Description	Default

REG_0TC_PCFG_Format
REG_0TC_CCFG_Format
	R/W	Output format:
		TC_FORMAT_RGB565 = 0,
		TC_FORMAT_RGB888 = 1,
		TC_FORMAT_FULL_YUV = 5, // YUV422 (0-255)
		TC_FORMAT_CROPPED_YUV = 6, // YUV422 (16-240)
		TC_FORMAT_BAYER = 7, // Bayer format, before ISP chain
		TC_FORMAT_MJPEG = 9, // JPEG
			05h

2.3.1 PROCESSED BAYER MODE

To change processed bayer mode, the user must set as below.

a. Set ”REG_xTC_P(C)CFG_Format” to “FORMAT_BAYER”

b. Set ”REG_xTC_P(C)CFG_PVIMask”[11] to 1b

c. Send configuration change command

The user can see proceessed bayer image with bayer decoding.

2.4 FRAME RATE SETTING

2.4.1 FRAME RATE TYPE

The sensor supports three frame rate types:

- Dynamic frame rate – The AE algorithm can change the frame rate within the defined minimum and maximum frame time range.
- Fixed not accurate - The frame rate defined in the register is rounded to the nearest frame time allowed for flicker cancellation.
- Fixed accurate - The frame rate defined in the register is the output frame rate.

The frame rate is limited by the output interface capabilities.

Table 2-2 Register Map for Output Format

Control Register	Att.	Description	Default
REG_0TC_PCFG_usFrTimeType	R/W	Frame rate type: TC_FR_TIME_DYNAMIC = 0, TC_FR_TIME_FIXED_NOT_ACCURATE = 1, TC_FR_TIME_FIXED_ACCURATE = 2	0x0000
REG_0TC_PCFG_FrRateQualityType	R/W	Frame rate type: TC_FRVSQ_BEST_FRRATE = 1. Binning enabled TC_FRVSQ_BEST_QUALITY = 2. Binning disabled	0x0000
REG_0TC_PCFG_usMaxFrTimeMsecMult10	R/W	Required frame time for fixed FR/maximal frame time for dynamic FR [333 – 6500] (Units are in 0.1 ms) (for example, 333 for 33.3 ms)	0x1964h
REG_0TC_PCFG_usMinFrTimeMsecMult10	R/W	Minimal frame time for dynamic FR. Not valid for fixed FR [333 – 6500] (Units are in 0.1 ms)	0x0000

2.5 RESOLUTION & OUTPUT SIZE SETTING

2.5.1 INPUT IMAGE SIZE CONTROL (INCLUDING INPUT CROPPING)

The input image is the image collected from the CIS pixel array and transferred to ISP processing. If the sensor is set for best frame rate mode, the maximal values for width and height are half that used in normal mode. However, using following registers, resized Sensor image including cropping and sub-sampling can be processed in the ISP.

Table 2-3 Register Map for Input Image

Control Register	Attr	Description	Default
REG_TC_GP_PrevReqInputWidth	R/W	Input width	A20h
REG_TC_GP_PrevReqInputHeight	R/W	Input height	798h
REG_TC_GP_PrevInputWidthOfs	R/W	Input horizontal offset	0
REG_TC_GP_PrevInputHeightOfs	R/W	Input vertical offset	0

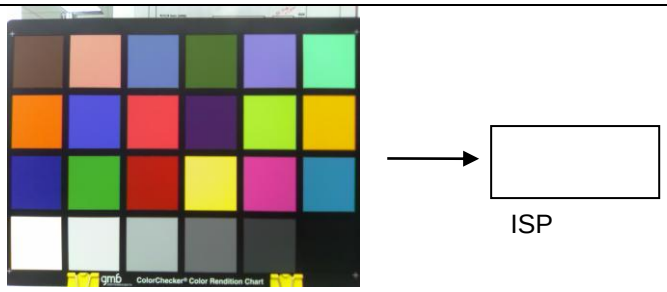

Figure 2-1 Full Image Input

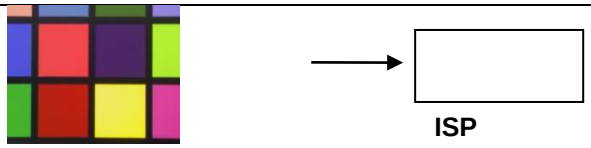

Figure 2-2 Cropped Input Image

2.5.2 IMAGE ZOOM

Image zoom has the same cropping effect as the input crop; however it is performed after ISP processing and thus does not affect ISP operation. The zoom width and height must be less than the input width and height respectively.

Table 2-4 Register Map for Image Zoom

Control Register	Attr	Description	Default
REG_TC_GP_PrevZoomReqInputWidth	R/W	Preview zoom width	A20
REG_TC_GP_PrevZoomReqInputHeight	R/W	Preview zoom height	798
REG_TC_GP_PrevZoomReqInputWidthOfs	R/W	Preview zoom horizontal offset	0
REG_TC_GP_PrevZoomReqInputHeightOfs	R/W	Preview zoom vertical offset	0
REG_TC_GP_CapZoomReqInputWidth	R/W	Capture zoom width	A20
REG_TC_GP_CapZoomReqInputHeight	R/W	Capture zoom height	798
REG_TC_GP_CapZoomReqInputWidthOfs	R/W	Capture zoom horizontal offset	0
REG_TC_GP_CapZoomReqInputHeightOfs	R/W	Capture zoom vertical offset	0

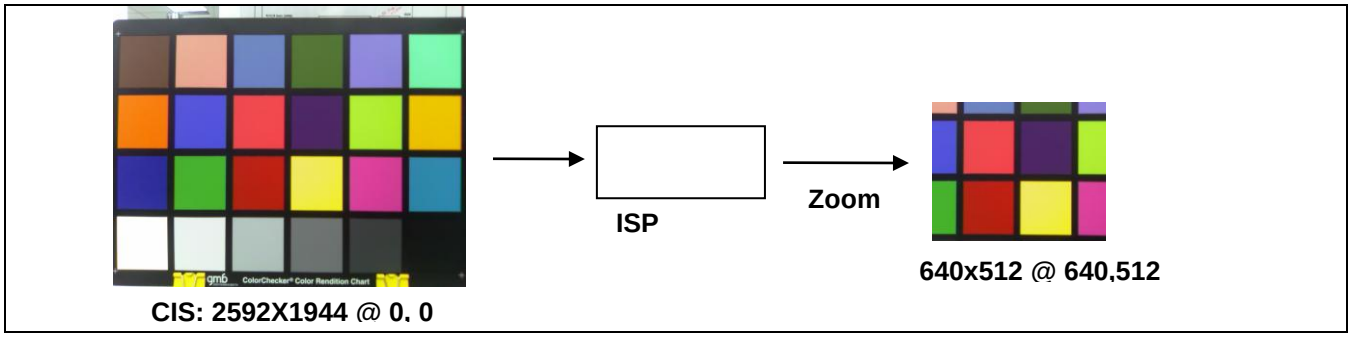

Figure 2-3 No Input Crop and Zoom 2x

2.5.3 CONTINUOUS PSEUDO ZOOM

Continuous pseudo zoom has the same cropping effect as the input crop; however it is performed after ISP processing and thus does not affect ISP operation. The host invokes the zoom command, the FW will output frames with changing zoom ratios until the target zoom is reached. The zoom width and height must be less than the input width and height respectively.

2.5.3.1 Continuous zoom

- STEPPING_UP(1) and STEPPING_DOWN(2) command are relevant for this mode.
- User has to set the target zoom ratio, zoom step and zoom speed before writes the continuous zoom command.

Table 2-5 Register Map for Zoom

Mnemonic	Default	Description
REG_TC_PZOOM_us88TargetZoom	0x0100	Target zoom ratio - 100h defines zoom x1 (no zoom, this is a minimal value), for zoom x4 should be set to 400h. 8:8 format is

		used.(upper two nibbles are used for integer values, lower two nibbles for fractional values)
REG_TC_PZOOM_us88ZoomStep	0x0019	One step ratio - 100h stands for 1.0 Zoom scale, for example, for step from Zoom x1 to Zoom x2. 8:8 format is used.
REG_TC_PZOOM_usZoomSpeed	0x000A	Continuous zoom speed - define zoom movement speed in N steps per second. e.g.) zoom speed x zoom step = zoom scale per second

Example 2-2 Continuous zoom mode

Zoom in

Ex) target: Zoom x4 , one step ratio : 0.25 , zoom speed : 10 steps per second

REG_TC_PZOOM_us88TargetZoom = 0x400 // Zoom x4

REG_TC_PZOOM_us88ZoomStep = 0x40 // one step ratio x0.25

REG_TC_PZOOM_usZoomSpeed = 0xA // 10 steps per second

REG_TC_PZOOM_ePzoomState = 1 // write STEPPING_UP command.

If zoom target is reached, zoom state is changed to IDLE(0) and then user can write zoom out command as following ways.

Zoom out

Ex) target : Zoom x1 , one step ratio : 0.25 , zoom speed : 10 steps per second

REG_TC_PZOOM_us88TargetZoom = 0x100 // Zoom x1

REG_TC_PZOOM_us88ZoomStep = 0x40 // one step ratio x0.25

REG_TC_PZOOM_usZoomSpeed = 0xA // 10 steps per second

REG_TC_PZOOM_ePzoomState = 2 // write STEPPING_DOWN command.

In 4EC EVT1, zoom in and zoom out command is same [REG_TC_PZOOM_ePzoomState = 1].

The zoom step ratio per frame should be bigger than 0.The zoom step ratio per frame is affected by zoom speed, one step ratio and frame rate.

Zoom step ratio per frame = REG_TC_PZOOM_us88ZoomStep x REG_TC_PZOOM_usZoomSpeed / FPS.

So [REG_TC_PZOOM_us88ZoomStep x REG_TC_PZOOM_usZoomSpeed] should be bigger than FPS

2.5.3.2 One step up and down

- ONE_STEP_UP(3) and ONE_STEP_DOWN(4) commands are relevant for this mode.
- This mode is used for increasing and decreasing by one step ratio which user sets.

- The one step ratio should be set before user writes one step command.
- The target zoom ratio is calculated by FW.

Example 2-3 One step up and down

One step up

Ex) Zoom x1 -> Zoom x1.5 -> Zoom x2

```
REG_TC_PZOOM_us88ZoomStep = 0x80  // one step ratio x0.5
REG_TC_PZOOM_ePzoomState = 3      // write ONE_STEP_UP command for Zoom x1.5
REG_TC_PZOOM_ePzoomState = 3      // write ONE_STEP_UP command again for Zoom x2
```

One step down

Ex) Zoom x2 -> Zoom x1.5 -> Zoom x1

```
REG_TC_PZOOM_us88ZoomStep = 0x80  // one step ratio x0.5
REG_TC_PZOOM_ePzoomState = 4      // write ONE_STEP_DOWN command for Zoom x1.5
REG_TC_PZOOM_ePzoomState = 4      // write ONE_STEP_DOWN command again for Zoom x1
```

2.5.3.3 Go Target immediately

- GO_TARGET(5) command is relevant for this mode.
- This mode is used to go target zoom ratio immediately.
- The target zoom ratio should be set before writes GO_TARGET command.

Example 2-4 Go Target immediately

GO_TARGET

Ex) Zoom x3

```
REG_TC_PZOOM_us88TargetZoom = 0x300    //Zoom x3
REG_TC_PZOOM_ePzoomState = 5           // write GO_TARGET command
```

2.5.3.4 User define mode

- FORCE_OUT_CROP(6) command is relevant for this mode.
- User has to set the cropping X/Y size and offset for zoom.

Example 2-5 User define

Ex) Zoom x2 (Sensor Input size : 1296x972) on Preview (VGA)

```
REG_TC_PZOOM_PrevZoomReqInputWidth = 0x510       //(1296)
REG_TC_PZOOM_PrevZoomReqInputHeight = 0x3CC      //(972)
REG_TC_PZOOM_PrevZoomReqInputWidthOfs = 0x288    // (648)
REG_TC_PZOOM_PrevZoomReqInputHeightOfs = 0x1E6   // (486)
```

REG_TC_PZOOM_ePzoomState = 6 // write FORCE_OUT_CROP command

2.5.3.5 Pause command

- IDLE(0) is pause command. It is used only for continuous zoom mode.
(Check if REGM_gPZoom_bPseudoZoomActive register is 1)
- If user writes IDLE(0) command , FW stops zooming and maintains the current zoom setting.
- If user wants to restart the continuous zoom from current zoom setting, user has to write the previous command.

Example 2-6 Pause command

Ex) Pause command on zooming in (STEPPING_UP) and restart.
REG_TC_PZOOM_ePzoomState = 0 // write IDLE(0) command
REG_TC_PZOOM_ePzoomState = 1 // write previous command to keep going

2.5.4 SCALE DOWN

The image output size is determined for each configuration individually. The image will be scaled down to the selected output size. The output width and height must be less than the zoom width and height respectively.

Control Register	Attr	Description	Default
REG_[0-4]TC_PCFG_usWidth	R/W	Preview output width	A20
REG_[0-4]TC_PCFG_usHeight	R/W	Preview output height	798
REG_[0-4]TC_CCFG_usWidth	R/W	Capture output width	A20
REG_[0-4]TC_CCFG_usHeight	R/W	Capture output height	798

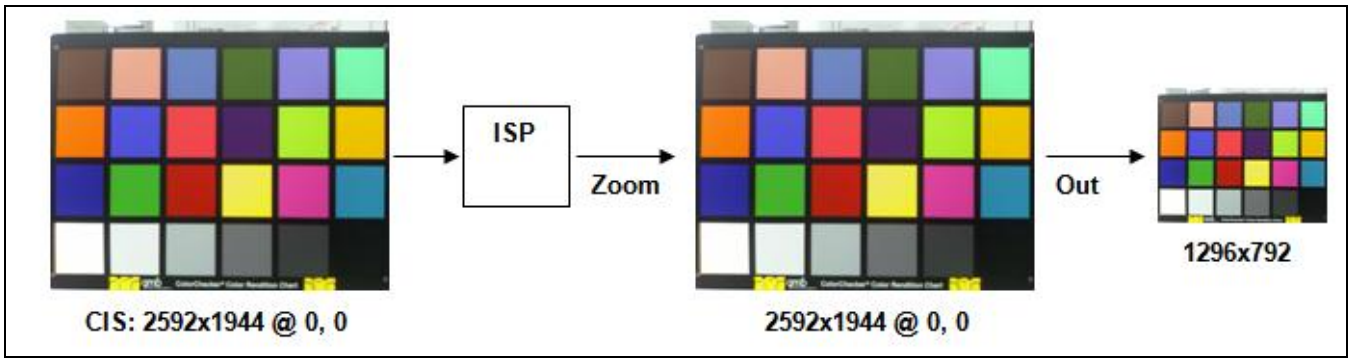

Figure 2-4 Crop, No Zoom and Scale Down 2x

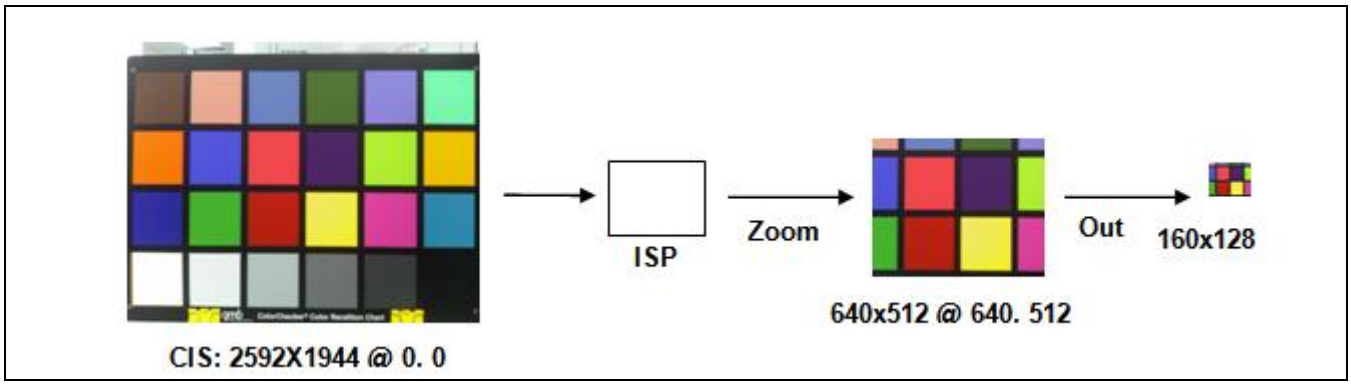

Figure 2-5 Crop, Zoom 2x and Scale Down 4x

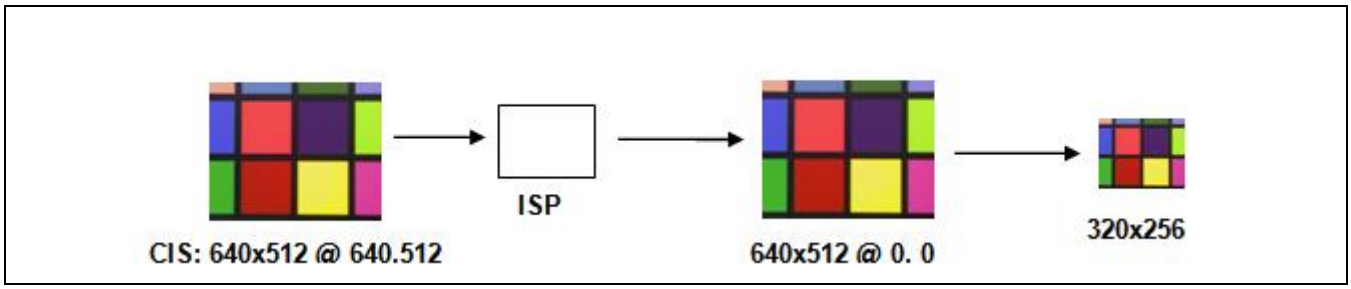

Figure 2-6 Crop, No Zoom and Scale Down 2x

3 ISP ALGORITHM DESCRIPTION

3.1 AUTO EXPOSURE (AE)

3.1.1 ABSTRACTS

Auto Exposure (AE) is an algorithm that controls image brightness automatically.

The target of the AE algorithm is to bring the image brightness to the AE Target Brightness value. The optimal AE Target Brightness value is usually determined by measuring a white patch at some condition.

The AE algorithm calculates the Ysum by dividing the image into several windows and giving each window a weight; Ysum is the weighted average of all luminance windows. The AE algorithm compares the Ysum with the AE Target Brightness value; both values are matched to be the same by adjusting the LEI (Linear Exposure Index). The LEI in turn is translated into exposure time, analogue gain and digital gain by the LEI Translation Algorithm.

3.1.2 Y SUMMATION

For calculating the Y summation, the image is divided into AE Windows and each has an AE Window Weight.

A total of 64 (8X8) AE weights are used. If the weight of a window is zero, the AE algorithm does not use the luminance data of that window.

Table 3-1 Register Map for AE Weight control

Attr.	Control Register	Description	Default
R/W	SARR_WeightTbl[B:n]	AE weight table. Weight value is [0-5]	See below

1	1	1	1	1	1	1	1
1	1	1	2	2	1	1	1
1	1	2	2	2	2	1	1
1	2	4	4	4	4	2	1
1	2	4	5	5	4	2	1
1	5	5	5	5	5	5	1
3	5	5	5	5	5	5	3
3	3	3	3	3	3	3	3

Figure 3-1 AE Weight Window (total 64) and default AE weight values

Y summation follows the formula below.

$$Ysum = \int (W_n \times Y_n) / \int (W_n)$$

(n=1~64, W_n: weight of the nth window, Y_n: Luminance average of nth window)

Figure 3-2 Y summation Formula

3.1.3 TARGET BRIGHTNESS

The target brightness level is controlled by the following register.

Table 3-2 Register Map for AE Target control

Attr.	Control Register	Description	Default
R/W	TVAR_ae_BrAve	AE Target Brightness value	3Dh

The AE Target Brightness can be fine tuned as a function of the current illumination type (color temperature) and current scene contrast. Illumination correction is controlled by the following registers:

Table 3-3 Register Map for AE Target control according to the illumination type

Attr.	Control Register	Description	Default
R/W	SARR_IllumType[0-6]	Illumination Type (Mon_AWB_RGainProj) [0]: Horizon [1]: Inca [2]: WW (U30) [3]: CW [4]: DL50 [5]: DL65 [6]: DL75 The values of Illumination Type must be increasing (IllumType[i]<IllumType[i+1]).	
R/W	SARR_IllumTypeF[0-6]	For each illumination type the respective correction factor (1x = 256).	100h

Contrast correction is controlled by the following registers.

Table 3-4 Register Map for AE Contrast control

Attr.	Control Register	Description	Default
R/W	ae_Contrast[0-15]	Contrast Value (Mon_AE_Contrast 0: low to 255: high) The Contrast Values must be increasing	

		(ae_Contrast[i]< ae_Contrast[i+1]).	
R/W	ae_ContrastF[0-15]	For each illumination type the respective correction factor (1x = 256).	100h

AE monitoring parameters:

Table 3-5 Register Map for AE monitoring

Attr.	Control Register	Description	Default
R	Mon_AE_Contrast	Current image contrast value.	
R	Mon_AE_CrntAvBr	Current image brightness.	
R	Mon_AE_ContrastFactor	Contrast correction factor (the result of ae_Contrast/ae_ContrastF table).	
R	Mon_AE_IllumFactor	Illumination Type factor (the result of SARR_IllumType/ SARR_IllumTypeF table).	
R	Mon_AE_YFactor	Overall correction factor: Mon_AE_ContrastFactor* Mon_AE_IllumFactor	
R	Mon_AE_GainAll	Gain needed to be applied to current brightness in order to bring it to corrected target brightness: TVAR_ae_BrAve * Mon_AE_YFactor / Mon_AE_CrntAvBr	
R	Mon_AE_FinalGain	Mon_AE_GainAll clipped to [ae_Glow: ae_GHigh].	

3.1.4 LEI BOUNDARY

To make AE stabilization, 4EC use boundary filter. Even if brightness changes in this threshold, AE does not change. lt_uInitPostToleranceCnt is number of AE action time in boundary. This register can help make certain AE brightness in large boundary.

In 4ECGX, lt_uLimitOption offer several options for AE boundary. First, Use default option that use lt_ulimitHigh and lt_uLimitLow. Second, AE boundary can be changed by Normalized Brightness. This option use lt_NormBr and lt_uLimit register. Third, AE boundary can be changed by Contrast in the scene. This option use lt_Contrast and lt_uLimit register.

Table 3-6 Register Map for LEI Boundary control

Mnemonic	Default	Description
lt_uLimitOption		AE boundary option. 0 : use default option. Use register lt_uLimitHigh and lt_uLimitLow 1 : Change boundary according to normbr, use register lt_NormBr and lt_uLimit 2 : Change boundary according to contrast, use register lt_Contrast and lt_uLimit
lt_uLimitHigh	0x0121	High threshold (factor) for LEI change (1x = 256).
lt_uLimitLow	0x00DF	Low threshold (factor) for LEI change (1x = 256)

lt_NormBr		Bondary tuning input register. Input data is norm br.
lt_Contrast		Bondary tuning input register. Input data is contrast value.
lt_uLimit		Boundary setting ouput value, value is according to lt_NormBr or lt_iLimit.
lt_uInitPostToleranceCnt		Number of times of AE action in boundary.

3.1.5 SMOOTH CONVERGENCE (LEI FILTERING)

The output of the AE algorithm is the LEI value. Before this value is passed on to the LEI Translation algorithm, some filtering is applied to the LEI value in order to smooth changes. The following registers control LEI smoothing.

Table 3-7 Register Map for LEI Smoothing control

Attr.	Control Register	Description	Default
R/W	lt_uSlowFilterCoef	Filter coefficient (1x = 256)	0h
R/W	lt_uSlowFilterUpLimit	Upper limit (factor) for LEI filtering (1x = 256).	133h
R/W	lt_uSlowFilterDownLimit	Low limit (factor) for LEI filtering (1x = 256).	CDh

The parameters are applied in the following manner:

- If the requested change in LEI value is too small than no change is applied to the LEI value: If CurrentLEI*lt_uLimitLow =< RequestedLEI =< CurrentLEI*lt_uLimitHigh then NewLEI = CurrentLEI.

- If the requested change in LEI value is too large then no smoothing is applied: If RequestedLEI >= CurrentLEI* lt_uSlowFilterUpLimit OR RequestedLEI <= CurrentLEI* lt_uSlowFilterDownLimit then NewLEI = RequestedLEI.

- Otherwise apply LEI filtering: NewLEI = RequestedLEI*(1-lt_uSlowFilterCoef) + CurrentLEI*lt_uSlowFilterCoef.

3.1.6 LEI TRANSLATION ALGORITHM

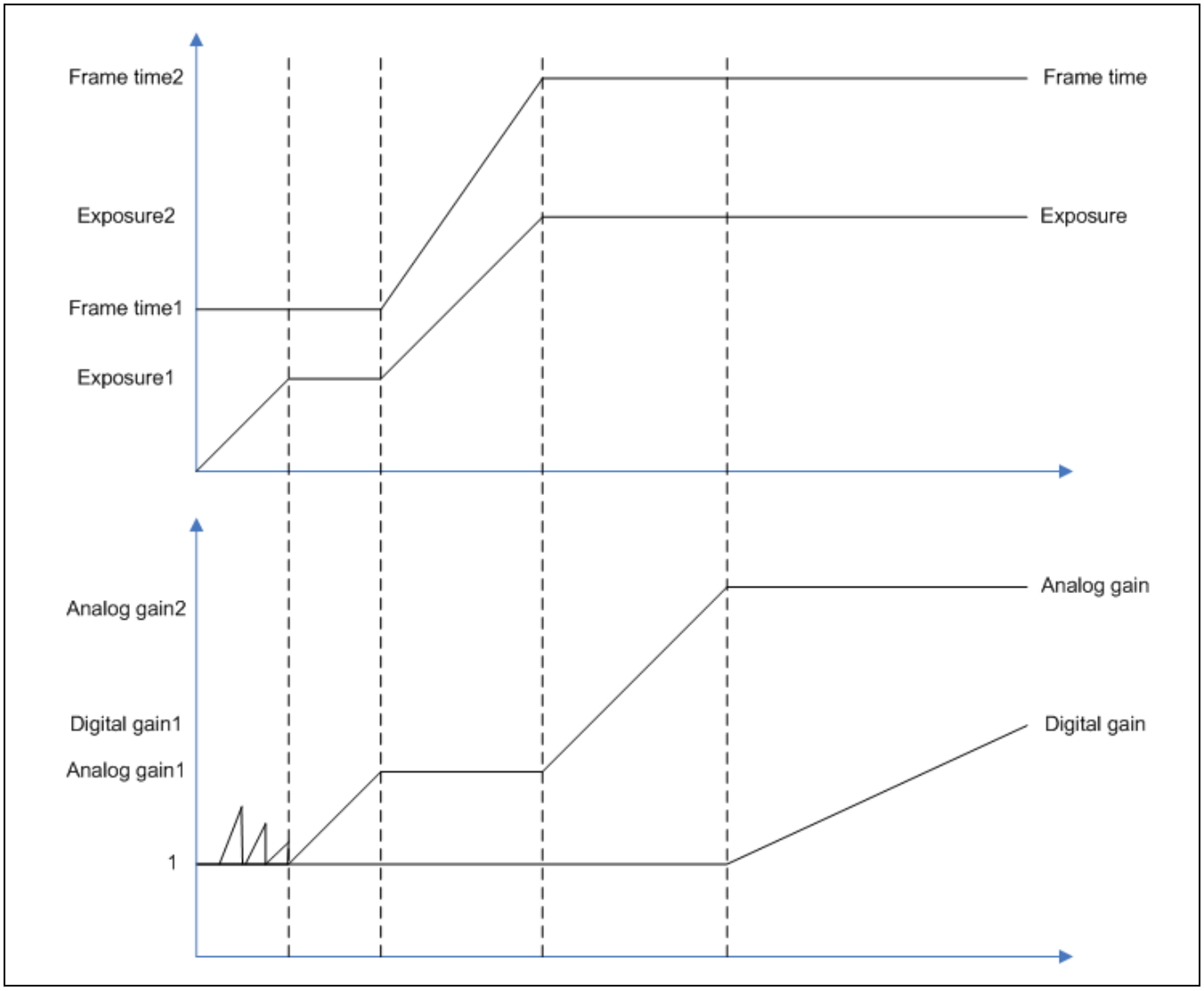

Figure 3-3 LEI Translation

When LEI grows, there are several zones:

1: Only exposure grows. The digital gain is 1. The analog gain is used only for the flicker compensation.

2: Only analog gain grows. The digital gain is 1, and the exposure is constant.

3: Only exposure grows. The digital gain is 1, and the analog gain is constant.

4: Only analog gain grows. The digital gain is 1, and the exposure is constant.

5: Only digital gain grows. The exposure and the analog gain are constant.

LEI Translation is controlled by the following registers.

Table 3-8 Register Map for LEI Translation control

Attr.	Control Register	Description	Default
R/W	lt_uMaxFlickerExp	Maximal value of exposure to which flicker correction is applied (i.e. if the exposure time requested by LEI translation is larger than this register, no flicker correction will be applied) (2us units).	186A0h
R/W	lt_uMaxExp1	Maximal allowed exposure while LEI is in zone 1 for preview mode (2us units).	7D00h
R/W	lt_uMaxExp2	Maximal allowed exposure for preview mode (2us units).	186A0h
R/W	lt_uCapMaxExp1	Maximal allowed exposure while LEI is in zone 1 for capture mode (2us units).	7D00h
R/W	lt_uCapMaxExp2	Maximal allowed exposure for capture mode (2us units).	186A0h
R/W	lt_uMaxAnGain1	Maximal allowed analog gain while LEI is in zone 2 (1x = 256).	200h
R/W	lt_uMaxAnGain2	Maximal allowed analog gain (1x = 256).	500h
R/W	lt_uMaxDigGain	Maximal allowed digital gain (1x = 256).	200h

LEI Translation can be monitored with the following registers.

Table 3-9 Register Map for LEI monitoring

Attr.	Control Register	Description	Default
R	Mon_LT_ulTargetLei	Target LEI requested by AE algorithm.	
R	Mon_LT_ulActualLei	Actual LEI achieved by FW.	
R	Mon_LT_usActualExp	Exposure time (2us units).	
R	Mon_LT_ActualGain_Digital88	Digital gain (1x = 256).	
R	Mon_LT_ActualGain_RequestedAnalog88	Analog gain requested by LEI Translation (1x = 256).	
R	Mon_LT_ActualGain_AccurateAnalog88	Actual analog gain available (i.e. requested gain after adjustment to HW limitations) (1x = 256).	

3.1.7 LT TABLE CONTROL

Customers have spec for exposure time in specific brightness (LUX). To pass their spec, need to make exposure table.

Control exposure time following to LEI. It makes easy to tuning, tuning people measure LEI in specific brightness and change exposure time table according to LEI.

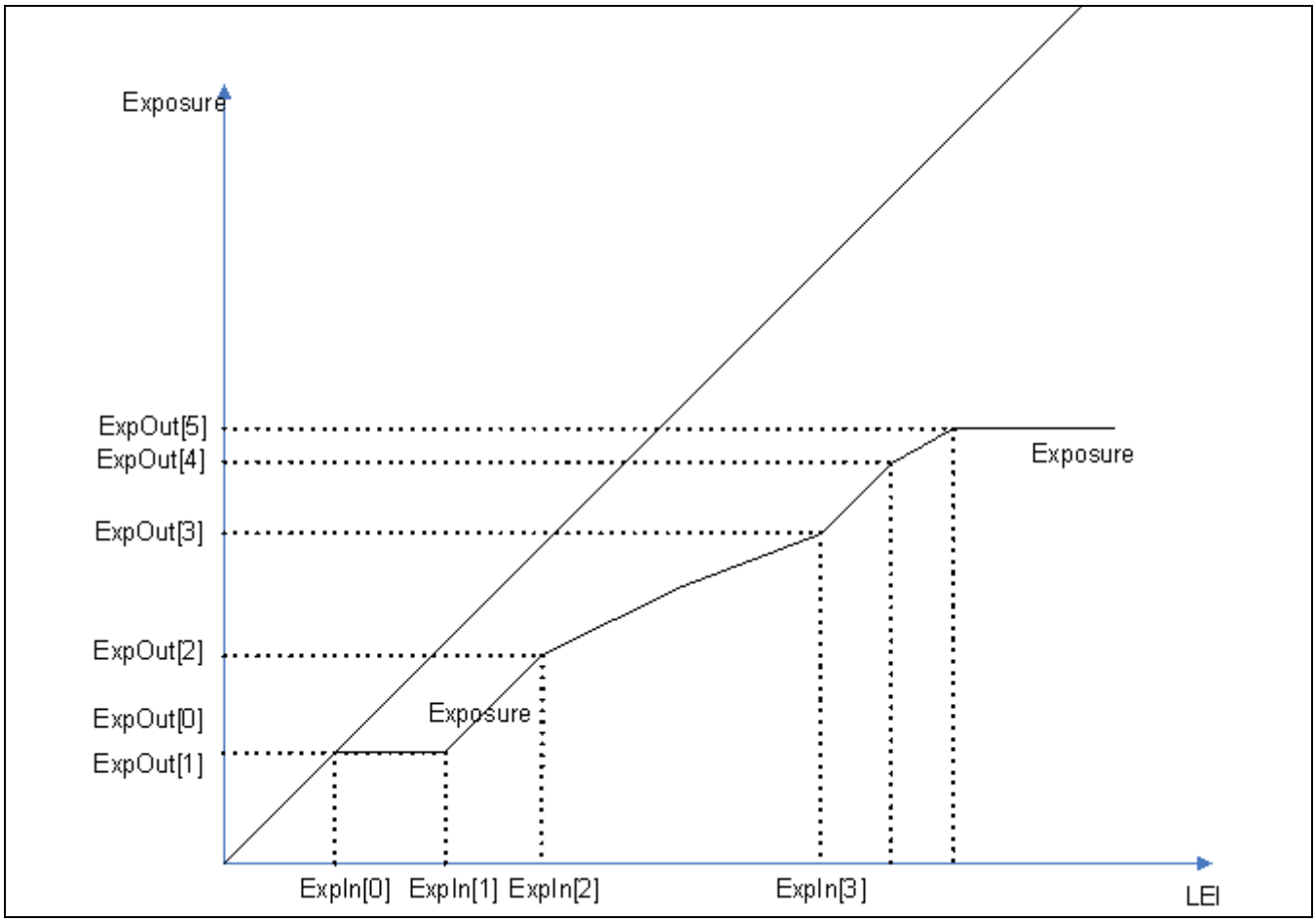

Figure 3-4 LT Table

The purpose of this system is for making requested exposure in specific LEI. Customer has exposure spec in specific LUX. To satisfy this spec, tuner can estimate LEI in specific LUX. Write LEI value to ExpIn and write Exposure value to ExpOut.

Table 3-10 Register Map for LT Table control

Mnemonic	Default	Description
lt_ExpGain_uSubsamplingmode	0	LT algorithm Mode. 0 : Using lt_uMaxExp1, lt_uMaxExp2 register. 1 : Using ExpCurve 1. 2: Using ExpCurve 2.
lt_ExpGain_uNonSubsampling	0	LT algorithm Mode. 0 : Using lt_uCapMaxExp1, lt_uCapMaxExp2 register. 1 : Using ExpCurve 1. 2 : Using ExpCurve 2.

lt_ExpGain_ExpCurveGainMaxStr[x]_uMaxAnGain		Max analog gain in Exp curve x.
lt_ExpGain_ExpCurveGainMaxStr[x]_uMaxDigGain		Max digital gain in Exp curve x.
lt_ExpGain_ExpCurveGainMaxStr[x]_ulExpIn[0]		Input LEI data for EXP curve.
lt_ExpGain_ExpCurveGainMaxStr[x]_ulExpOut[0]		Output Exposure data for EXP curve.

3.1.8 MANUAL AE CONTROL

To manually set the LEI value exposure time and total gain value, the AE algorithm must first be turned off.

<REG_TC_DBG_AutoAlgEnBits[b:1]>=0

Table 3-11 Register Map for Manual AE control

Attr.	Control Register	Description	Default
R/W	REG_TC_DBG_AutoAlgEnBits	Auto-algorithms enable/disable. [b:1] : AE algorithm 1: enable 0: disable.	77Fh
R/W	REG_SF_USER_LeiLow	LEI value. LSB (0.01ms units).	0h
R/W	REG_SF_USER_LeiHigh	LEI value. MSB (0.01ms units).	0h
R/W	REG_SF_USER_LeiChanged	Apply LEI value.	0h

To manually set exposure time and gain values, the AE and LEI Translation algorithms must first be turned off.

<REG_TC_DBG_AutoAlgEnBits[b:2:1]>=0

Table 3-12 Register Map for Manual AE and LEI Translation control

Attr.	Control Register	Description	Default
R/W	REG_TC_DBG_AutoAlgEnBits	Auto-algorithms enable/disable. [b:1] : AE algorithm 1: enable 0: disable. [b:2] : LEI Translation algorithm 1: enable 0: disable.	77Fh
R/W	REG_SF_USER_Exposure	Exposure time. LSB (0.01ms units).	0h
R/W	REG_SF_USER_ExposureHigh	Exposure time. MSB (0.01ms units).	0h
R/W	REG_SF_USER_ExposureChanged	Apply exposure time.	0h
R/W	REG_SF_USER_TotalGain	Total gain (1x = 256). This value will be translated by FW into analog gain and digital gain by usual LEI Translation rules.	0h
R/W	REG_SF_USER_TotalGainChanged	Apply total gain.	0h
R/W	REG_SF_USER_aGain	Analog gain (1x = 256).	0h
R/W	REG_SF_USER_aGainChanged	Apply analog gain.	0h
R/W	REG_SF_USER_dGain	Digital gain (1x = 256).	0h
R/W	REG_SF_USER_dGainChanged	Apply digital gain.	0h

3.1.9 AE BOOST

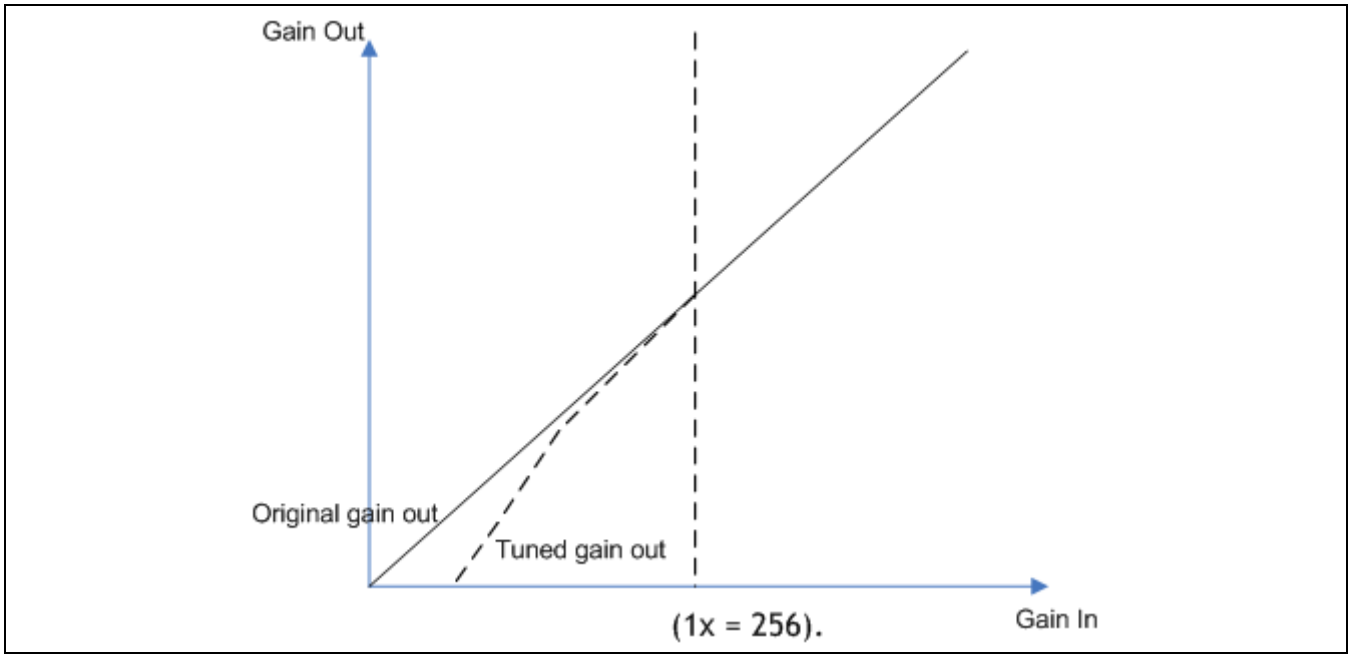

Figure 3-5 AE Boost

Table 3-13 Register Map for AE Boost control

Mnemonic	Default	Description
ae_GainIn	-	X value of AE gain transformation (1x = 256).
ae_GainOut	-	Y value of AE gain transformation (1x = 256).

3.1.10 ISO

ISO Auto mode is normal mode using LT table. ISO classic mode is gain off set mode. ISO sport limits exposure time.

Table 3-14 ISO Modes

	Divide LEI by ISO value. (According to LT table, exposure value is not divided by ISO value.)	Divide Exposure value by .
In ISO mode, 4EC use digital gain for ISO gain.	ISO_MODE_DIGITAL_LEI_OFFSET	ISO_MODE_DIGITAL_GAIN_OFFSET
In ISO mode, 4EC use total gain for ISO gain.	ISO_MODE_TOTAL_GAIN_OFFSET	ISO_MODE_TOTAL_LEI_OFFSET

ISO sport mode limits max exposure.
(Max exposure value set by table that matched with ISO value)

Table 3-15 Max exposure values according to ISO value

ISO value	100	125	160	200	250	320	400	500	640	800	1000	1250	1600
Max exposure value	0	1	2	3	4	5	6	7	8	9	10	11	12
	Max exposure value “seti_usIsoMaxExpTbl” is set by above array number.												

Table 3-16 Register Map for ISO control

Mnemonic	Default	Description
REG_SF_USER_IsoType	-	ISO type state. 0: ISO_MODE_AUTO. 1: ISO_MODE_ANALOG_GAIN_FIX 2: ISO_MODE_DIGITAL_GAIN_OFFSET. 3: ISO_MODE_TOTAL_DIGITAL_GAIN_OFFSET. 4: ISO_MODE_TOTAL_EXPOSURE_OFFSET. 5: ISO_MODE_TOTAL_LEI_GAIN_OFFSET
REG_SF_USER_IsoVal	-	Set ISO value.
REG_SF_USER_IsoChanged	-	Synchronize FW with total ISO change
seti_usIsoMaxExpTbl		Max exposure table for ISO sport mode.

3.2 AUTO WHITE BALANCE (AWB)

3.2.1 ABSTRACTS

The AWB algorithm adjusts image colors to best match human perceptions by controlling R, G, and B digital gain.

The algorithm uses various statistics channels – warm, outdoor, low brightness, general, special color and so on.

Each statistics channel has its own filter on normalized r/b plane and those filtered data used to estimate illuminant. The algorithm has also scene type detector which scene detection result can drive selection or mixing valid statistics channels on various illuminant.

3.2.2 BASIC DESCRIPTION ABOUT AWB

4EC AWB also uses basic AWB algorithm that was used in previous SEC product like 4BA, 4CA… and It also has some unique feature that can enhance illuminant estimation performance.

Basically, AWB use both algorithm, GWA (Gray World Assumption) and Scene type detection method that can choose gray filter according to scene type. And as previous SEC products do, white point refinement is processed using luminance and chromaticity level threshold.

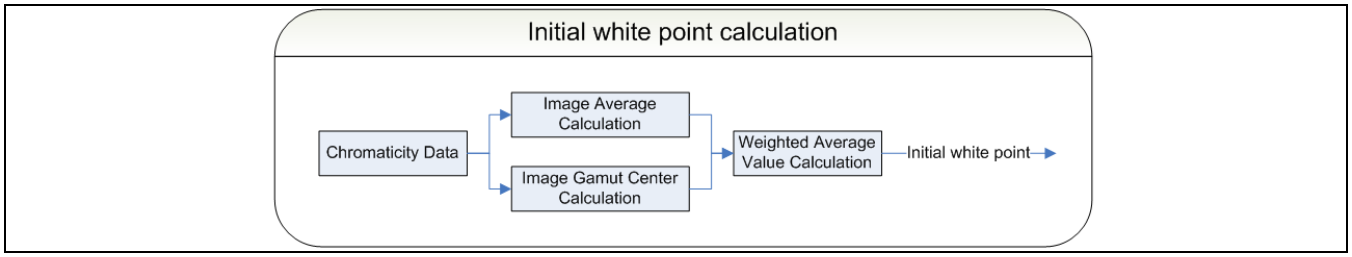

Figure 3-6 Exposure Initial white point calculation

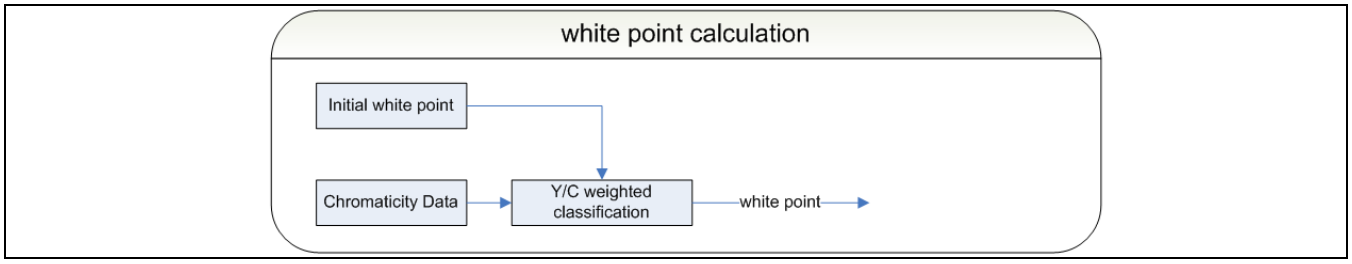

Figure 3-7 White point calculation

In 4EC, main difference with previous product's AWB is like below:

- Scene detection was redefined: Typical scene type was defined that includes general, lowtemp and cloudy scene type.

- Some color component can be used for illuminant estimation (memory color based AWB): AWB use not only gray information but also some color information in the images.

3.2.3 SETTING WHITE LOCUS

3.2.3.1 Calculate Real White Point

At the first, S/W Gray region's boundary should be maximized. And we can obtain correct information.

In D65, CWF, Incand A, Horizon Light source environment, take a picture of Macbeth white balance chart and read "Mon_AWB_IndoorSt_AvE_Point" value. This procedure should be proceeded a few module.

Each light source that value should be averaged and make those real white point values.

3.2.3.2 Calculate Global Variables

Calculate Global Variables which used for Locus projection and Moving Equation's coefficient by use Excel's trend line function. And Using below formula set Global variables to registers.

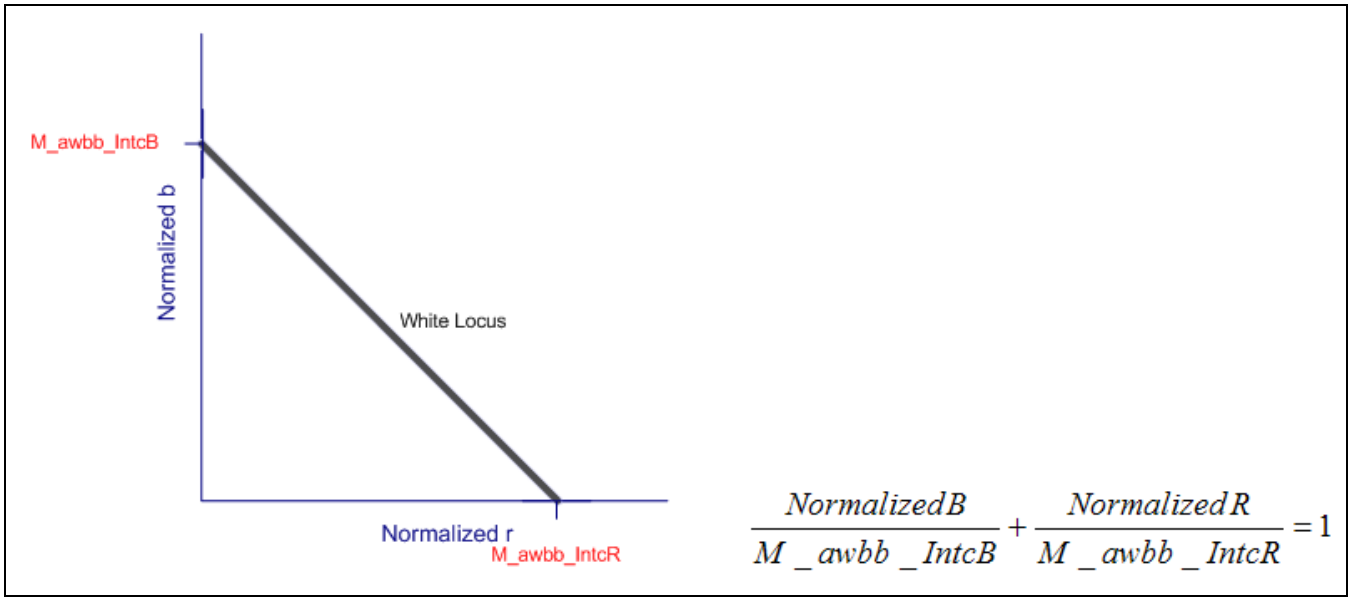

Figure 3-8 White Locus formula and graph

Table 3-17 Awbb registers for White locus setting

Mnemonic	Default	Description
awbb_IntcR	0x013D	White Locus Setting value
awbb_IntcB	0x012B	White Locus Setting value

Table 3-18 Mon_AWB registers for White locus setting

Mnemonic	Attr	Description
Mon_AWB_IndoorSt_AvE_Point_r	R	Indoor region average point
Mon_AWB_IndoorSt_AvE_Point_b	R	Indoor region average point
Mon_AWB_IntcR_trans	R	White locus line monitor parameter

Mon_AWB_IntcB_trans	R	White locus line monitor parameter
Mon_AWB_CoffL	R	White locus line monitor parameter
Mon_AWB_CoffH	R	White locus line monitor parameter
Mon_AWB_CoffJ	R	White locus line monitor parameter
Mon_AWB_CoffK	R	White locus line monitor parameter

3.2.4 ADJUSTMENT OF INDOOR GRAY REGION BOUNDARY

Control Indoor Gray region's boundary. It's important that gray boundary should be established to minimum at the level which doesn't have opposite effect for normalcy AWB performance.

Table 3-19 Indoor Gray Region control registers

Mnemonic	Default	Description
awbb_IndoorGrZones_m_BGrid[0]_m_left	0x02EE	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[0]_m_right	0x0348	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[1]_m_left	0x02C6	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[1]_m_right	0x032A	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[2]_m_left	0x029E	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[2]_m_right	0x030C	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[3]_m_left	0x0280	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[3]_m_right	0x02EE	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[4]_m_left	0x0262	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[4]_m_right	0x02D0	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[5]_m_left	0x0244	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[5]_m_right	0x02B2	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[6]_m_left	0x0226	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[6]_m_right	0x0294	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[7]_m_left	0x0203	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[7]_m_right	0x0276	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[8]_m_left	0x01E0	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[8]_m_right	0x0258	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[9]_m_left	0x01BD	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[9]_m_right	0x0235	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[10]_m_left	0x01AE	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[10]_m_right	0x0212	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[11]_m_left	0x0190	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[11]_m_right	0x01EF	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[12]_m_left	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[12]_m_right	0x0000	IndoorGrZones

awbb_IndoorGrZones_m_BGrid[13]_m_left	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[13]_m_right	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[14]_m_left	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[14]_m_right	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[15]_m_left	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[15]_m_right	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[16]_m_left	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[16]_m_right	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[17]_m_left	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[17]_m_right	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[18]_m_left	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[18]_m_right	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[19]_m_left	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_BGrid[19]_m_right	0x0000	IndoorGrZones
awbb_IndoorGrZones_m_GridStep	0x0005	IndoorGrZones
awbb_IndoorGrZones_ZInfo_m_GridSz	0x000C	IndoorGrZones
awbb_IndoorGrZones_m_Boffs	0x017C	IndoorGrZones

3.2.5 ADJUSTMENT OF OUTDOOR GRAY REGION BOUNDARY

Because outdoor gray region is used only case of outdoor, It includes just D65 ~ D50 white point. (It is recommended that outdoor gray region is established as subset of General Region)

Table 3-20 Outdoor Gray Region control registers

Mnemonic	Default	Description
awbb_OutdoorGrZones_m_BGrid[0]_m_left	0x0230	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[0]_m_right	0x029E	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[1]_m_left	0x021C	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[1]_m_right	0x028A	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[2]_m_left	0x0208	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[2]_m_right	0x0276	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[3]_m_left	0x01F4	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[3]_m_right	0x0262	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[4]_m_left	0x01E0	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[4]_m_right	0x024E	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[5]_m_left	0x01CC	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[5]_m_right	0x023A	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[6]_m_left	0x01B8	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[6]_m_right	0x0226	OutdoorGrZones

awbb_OutdoorGrZones_m_BGrid[7]_m_left	0x01B8	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[7]_m_right	0x0212	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[8]_m_left	0x01B8	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[8]_m_right	0x01FE	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[9]_m_left	0x01AE	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[9]_m_right	0x01EA	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[10]_m_left	0x0000	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[10]_m_right	0x0000	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[11]_m_left	0x0000	OutdoorGrZones
awbb_OutdoorGrZones_m_BGrid[11]_m_right	0x0000	OutdoorGrZones
awbb_OutdoorGrZones_m_GridStep	0x0004	OutdoorGrZones
awbb_OutdoorGrZones_ZInfo_m_GridSz	0x000A	OutdoorGrZones
awbb_OutdoorGrZones_m_Boffs	0x0244	OutdoorGrZones

3.2.6 ADJUSTMENT OF LOW BRIGHTNESS GRAY REGION BOUNDARY

Generally, Low Brightness region was roughly and widely set.

Table 3-21 Low brightness Gray Region control registers

Mnemonic	Default	Description
awbb_LowBrGrZones_m_BGrid[0]_m_left	0x034D	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[0]_m_right	0x045B	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[1]_m_left	0x02EA	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[1]_m_right	0x0445	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[2]_m_left	0x02A4	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[2]_m_right	0x0428	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[3]_m_left	0x0254	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[3]_m_right	0x03FB	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[4]_m_left	0x0207	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[4]_m_right	0x03C7	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[5]_m_left	0x01CF	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[5]_m_right	0x0365	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[6]_m_left	0x019E	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[6]_m_right	0x031A	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[7]_m_left	0x0170	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[7]_m_right	0x02DB	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[8]_m_left	0x0149	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[8]_m_right	0x029E	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[9]_m_left	0x0122	LowBrGrZones

awbb_LowBrGrZones_m_BGrid[9]_m_right	0x025D	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[10]_m_left	0x00FE	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[10]_m_right	0x022E	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[11]_m_left	0x00E4	LowBrGrZones
awbb_LowBrGrZones_m_BGrid[11]_m_right	0x01F1	LowBrGrZones
awbb_LowBrGrZones_m_GridStep	0x0006	LowBrGrZones
awbb_LowBrGrZones_ZInfo_m_GridSz	0x000C	LowBrGrZones
awbb_LowBrGrZones_m_Boffs	0x00D6	LowBrGrZones

3.2.7 REGION TUNING METHOD USING DIRECT REGISTER CONTROL

Gray Zones are defined in frame of reference Normalized r (= 2^11 * R / (R + G + B) (axis X)) and Normalized B (= 2^11 * B / (R + G + B) (axis Y)) by structures.

Here are registers which can control Gray Zone.

Table 3-22 Zone control registers

Parameters	Description
m_BGrid[N]_m_left	define the polygon vertexes (N: order of vertex)
m_BGrid[N]_m_right	define the polygon vertexes (N: order of vertex)
m_GridStep	define the grid for Normalized B axis (deltaB = 2^m_GridStep)
m_GridSz	define number of vertexes for left and right borders
m_Boffs	the minimal values of Normalized B

The polygon defined by two arrays of vertexes. The vertexes defined from bottom to top as left and right arrays.

The coordinates of left vertex number k (0 <= k < m_GridSz) are :

(m_BGrid [k].left, m_Boffs + k * (1<< m_GridStep))

The coordinates of right vertex number k are :

(m_BGrid [k].right, m_Boffs + k * (1<< m_GridStep))

cf) scale that was represented in simian tuning tool can be different in setting registers.

3.2.8 ADJUSTMENT OF LOW TEMPERATURE GRAY REGION BOUNDARY

Low temperature condition's gray boundary is a circle. It is recommended that the circle contain incandescent A and Horizon reference light source white point.

Table 3-23 Low temperature circle region control registers

Mnemonic	Default	Description
awbb_CrclLowT_R_c	0x034B	CrclLowT
awbb_CrclLowT_B_c	0x019E	CrclLowT
awbb_CrclLowT_Rad_c	0xEF10	CrclLowT

3.2.9 DEDUCTION WHICH IS CONCERNED ABOUT SENSOR FEATURES

At each scene, AWB produce color gamut size information through below registers.

Gamut size was defined like below Figure.

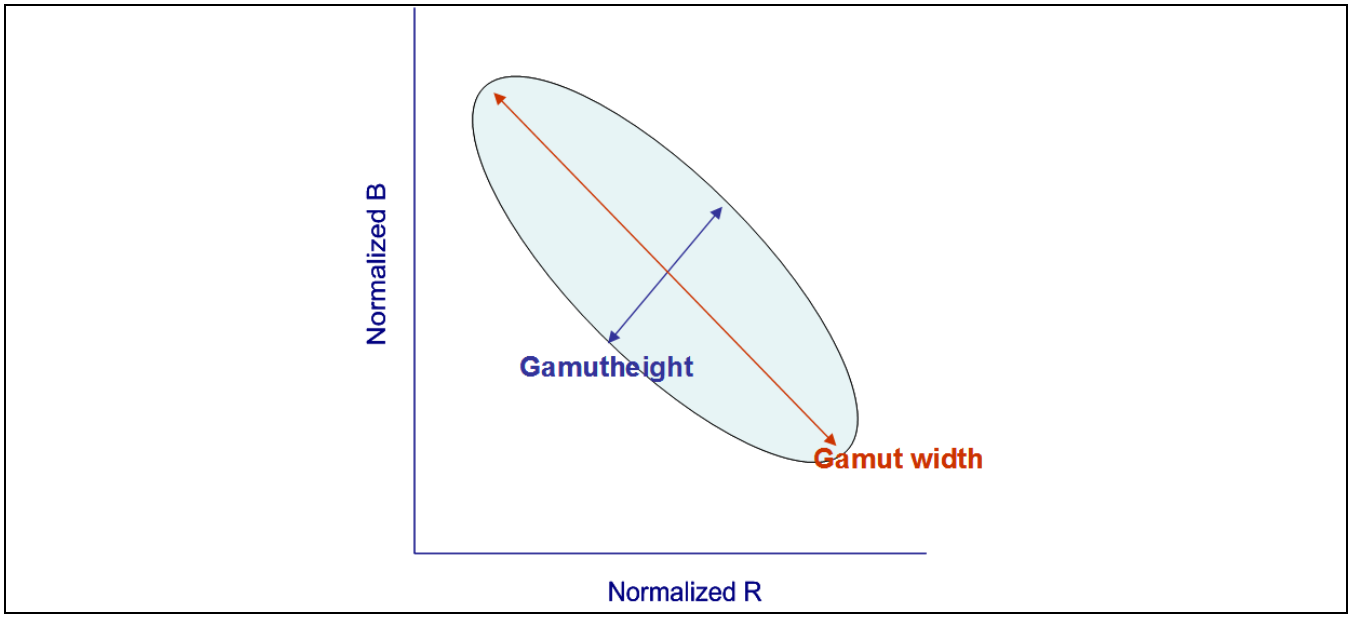

Figure 3-9 Gamut Size Definition

Table 3-24 Color Gamut size information registers

Mnemonic	Attr	Description
Mon_AWB_uGamutWidth	R	-
Mon_AWB_uGamutHeight	R	-

3.2.9.1 Macbeth Color Checker feature

Find Macbeth Color Checker's Minimum Gamut Height and Width under each light source(D65, CWF, Incand A).

3.2.9.2 Macbeth White balance chart feature

Find Macbeth White Balance Chart's Minimum Gamut Height and Width under each light source.

3.2.10 GAMUT THRESHOLDS SETTING

A value which is subtracted 5% margin from Macbeth color checker's Minimum gamut width, height is set on threshold1.

A value which is added 5% margin from Macbeth white balance chart's Maximum gamut width, height is set on threshold2.

Table 3-25 Color Gamut size threshold registers

Mnemonic	Default	Description
awbb_GamutWidthThr1	0x05F0	Gamut threshold1
awbb_GamutHeightThr1	0x01F4	Gamut threshold1
awbb_GamutWidthThr2	0x006C	Gamut threshold2
awbb_GamutHeightThr2	0x0038	Gamut threshold2

3.2.11 SCENE DETECTION TUNING.

In order to acquire easy to tune AWB scene detection, previous AWB scene detection was modified like below.

- Indoor white mode added.
- Indoor high-temp mode was deleted.
- Typical mode was defined.

3.2.11.1 Scene Type Redefine - 4EC scene detection hierarchy

Mixed scene type is consisted of three scene types- sunny, low brightness and typical scene. And typical scene includes Indoor, Lowtemp and cloudy scene type.

The summation of sunny, low brightness and typical scene weight should be 100h. And the components summation of typical scene also should be 100h.

As can see below figure, indoor scene type has two components: white scene and General scene. And cloudy scene type has also two components: large gamut and General cloudy scene.

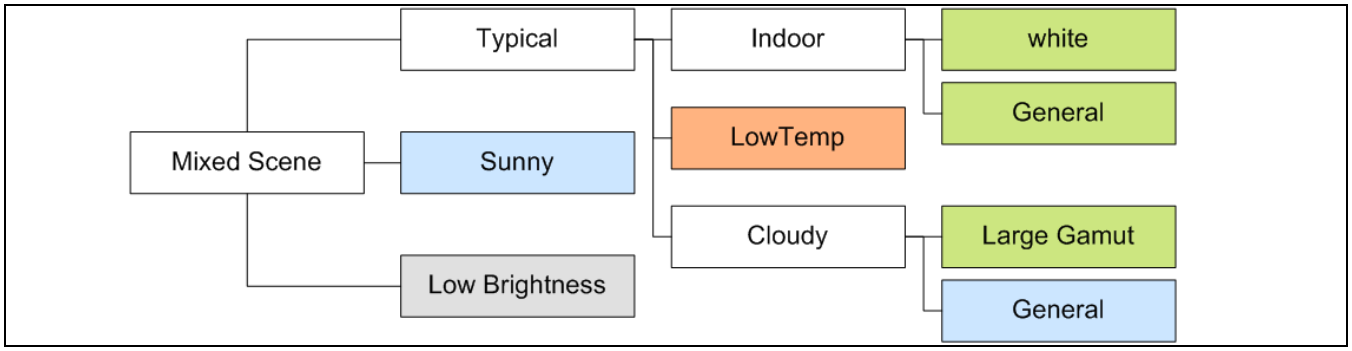

Figure 3-10 4EC scene detection hierarchy

- White scene
  - White mode takes indoor zone's average value for its white point.
- Typical-Indoor-General scene
  - It's same as previous indoor scene. It uses indoorzone.
- Typical-LowTemp scene
  - It's same as previous Low temp scene. It uses Lowtempzone.
- Typical-Cloudy-LargeGamut scene

  - It's same as previous temporal Cloudy-LargeGamut type. It was classified to scene. It uses indoorzone.
- Typical-Cloudy-General scene
  - It is same as previous cloudy scene. It uses outdoorzone.
- Sunny scene
  - It uses outdoorzone.
- Low Brightness scene
  - It uses LowBrzone.

3.2.11.2 Scene Detection Map

In order to calculate the weights of mixed scenes, 4EC AWB use below scene detection map.

As using scene detection map, AWB can be calibrated more easily.

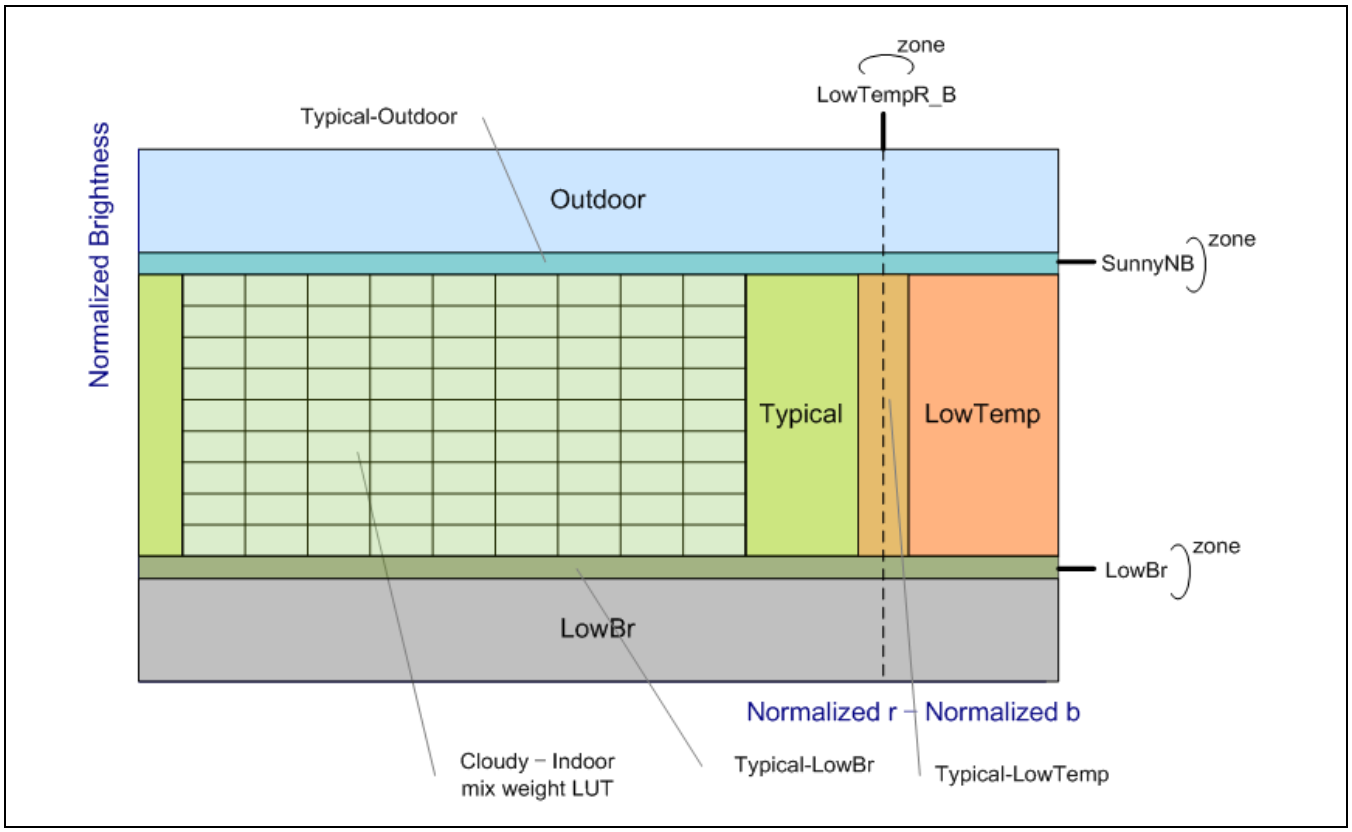

Figure 3-11 4EC Scene detection map

Table 3-26 4EC AWB scene detection parameters

Mnemonic	Attr	Description
Mon_AWB_RminusB	R	R-B monitoring value
Mon_AAIO_PrevFrmData_NormBr	R	Normalize Brightness monitoring value
awbb_LowBr	R/W	Normalize brightness threshold for LowBr and Typical

awbb_LowBr_NBzone	R/W	NB threshold smoothing zone
awbb_SCDetectionMap_SEC_SceneDetectionMap	R/W	Outdoor(cloudy)-typical scene weight LUT component
awbb_SCDetectionMap_SEC_StartR_B	R/W	Outdoor-typical scene weight LUT R-B index parameter
awbb_SCDetectionMap_SEC_StepR_B	R/W	Outdoor-typical scene weight LUT R-B index parameter
awbb_SCDetectionMap_SEC_SunnyNB	R/W	Outdoor-typical scene weight LUT NB index parameter
awbb_SCDetectionMap_SEC_StepNB	R/W	Outdoor-typical scene weight LUT NB index parameter
awbb_SCDetectionMap_SEC_LowTempR_B	R/W	R-B threshold for LowTemp and Typical Scene
awbb_SCDetectionMap_SEC_SunnyNBZone	R/W	NB threshold smoothing zone
awbb_SCDetectionMap_SEC_LowTempR_BZone	R/W	R-B threshold smoothing zone

Using Upper two Scene detection monitoring values (R-B and Normalize Brightness) and Scene detection map diagram, we can check 4EC AWB scene detection.

If the point(R-B,NB) exists on outdoor region of R-B/NB plain, the result of AWB scene detection must be outdoor 100%(256). And if the point exists on typical-outdoor region, the result will be mixed scene type of typical and outdoor. The mixing weight is calculated by distance with each threshold (threshold-zone and threshold+zone).

Cloudy-indoor mix weight LUT

Typical scene consists of three sub-scenetypes-cloudy, indoor and lowtemp. Each scene type has different data and method for white point estimation so isolation of scene type is important.

Actually, Lowtemp scene type can be classified easily using R-B threshold that was described table before.

And 4EC has cloudy-indoor mix weight LUT that can easily control cloudy-indoor mix weight.

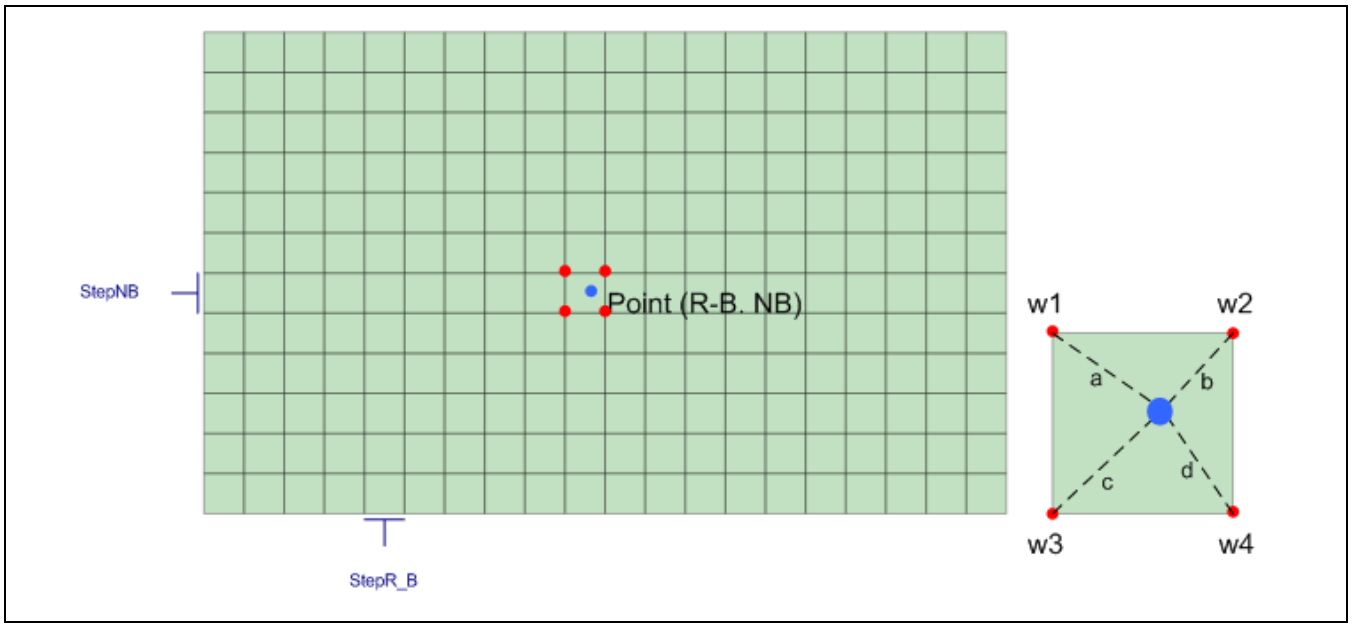

Figure 3-12 Cloudy-indoor mix weight LUT

The result weight of upper point is:

$$cloudy\_weight = (w1 \times d + w2 \times c + w3 \times b + w4 \times a)/(a + b + c + d)$$
$$indoor\_weight = 256(100\%) - cloudy\_weight$$

3.2.12 COMPENSATION METHOD FOR MODULE-TO-MODULE VARIATION

In order to compensate AWB module-to-module variation, we use below linear transform, each coefficient and constant can be set on E-fuse ROM (on test stage).

$$Normalize\_R_{trans} = coefficient\_r \times Normalize\_R_{original} + const\_r$$
$$Normalize\_B_{trans} = coefficient\_b \times Normalize\_B_{original} + const\_b$$

Figure 3-13 Module variation compensation formulas

Table 3-27 module variation compensate parameters (monitor)

Mnemonic	Attr	Description
Mon_AWB_ModuleVar_CoefR	R	coefficient_r
Mon_AWB_ModuleVar_ConstR	R	const_r
Mon_AWB_ModuleVar_CoefB	R	coefficient_b
Mon_AWB_ModuleVar_ConstB	R	const_b

3.2.13 AWB CONVERGENCE CONTROL

Table 3-28 AWB Convergence control related registers

Mnemonic	Attr	Description
Mon_AWB_PostFilterPt	R	Final point of AWB
Mon_AWB_PrevFinalPt	R	Final point of previous awb frame
Mon_AWB_FinalPt	R	Target point of present awb frame
awbb_WpFilterMinThr	R/W	Minimum white point difference threshold
awbb_WpFilterMaxThr	R/W	Maximum white point difference threshold
awbb_WpFilterCoef	R/W	White point convergence speed coefficient (Max:100h)
awbb_WpFilterSize	R/W	White point averaging window size parameter

White point difference is defined like:

$$diff = | newPt.r - currPt.r | + | newPt.b - currPt.b |$$

If diff > Max threshold, white point queue update speed should be boosted (three times weighted).

If diff < Min threshold, white point queue is not be updated.

Else, white point queue updated with no weight.

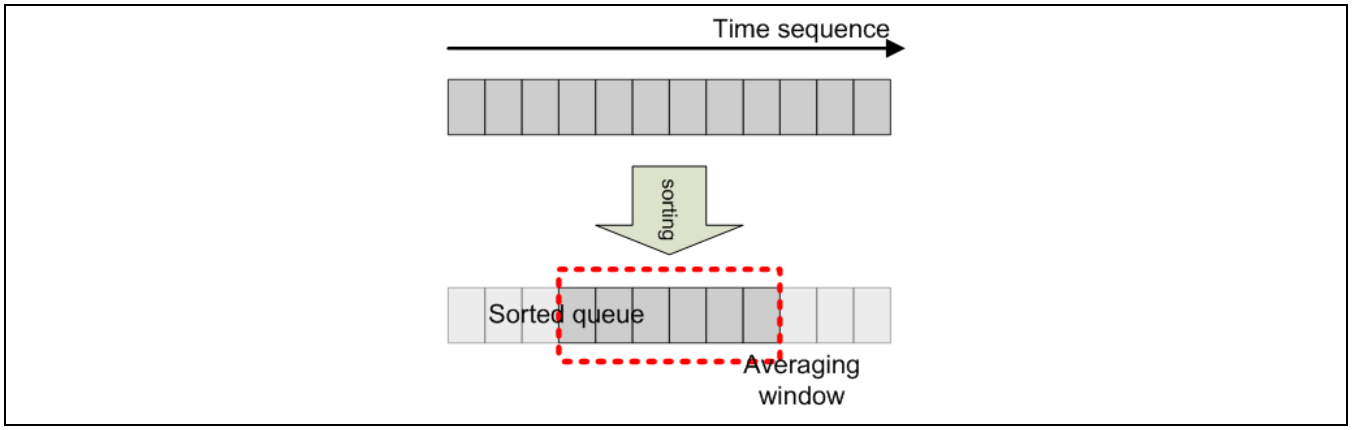

Figure 3-14 AWB convergence queue structure

And using inside components of WP Queue averaging window, Averaged white point (AvPt) can be calculated.

Target white point is calculated as:

$$WP = (WpFilterCoef \times AvPt + (100h - WpFilterCoef) \times PvPt) / 100h$$

(PvPt is previous white point.)

3.3 AUTO FOCUS (AF)

3.3.1 ABSTRACTS

Auto focus algorithm is required when autofocus lens module is used. The objective of auto focus algorithm is to control the autofocus lens movement in order to obtain a sharp image of the scene. The provided autofocus algorithm is highly versatile and configurable. Two configurable AF windows are used to analyze the image sharpness. In each window, several sharpness measures are collected and analyzed in order to guarantee robust AF operation in various scenes and lighting conditions. AF algorithm uses scene type classification in order to select the best sharpness measure and AF window depending on current scene content and illumination level.

The search for best focus lens position can be performed in the following searching modes:

Single shot mode –typical mode used for snapshots. In this mode the search is performed and the lens is moved to the position with the best scene focus to make a capture.

- Continuous (video) mode – typical mode used for recording video. In this mode the auto focus algorithm continuously analyzes the images for changes in the scene or in image sharpness. The lens is refocused automatically when such changes are detected. The search for best lens position is performed by smoothly starting from the current lens position to minimize effects of abrupt lens position change.

- Manual mode –the lens is controlled by the host.

Both speed and accuracy of AF performance can be achieved by combining coarse and fine search modes. Coarse mode is used for fast search for best lens position across the entire AF range. The speed is further improved by stopping the coarse search early if the best sharpness position has already been found. Fine mode is used to improve the accuracy by searching in the neighborhood of the coarse mode best position. To further boost AF speed, the AF algorithm supports special High-Speed AF mode in Single shot scenario. During High-Speed AF mode the system frame rate is increased temporarily by cropping input from the sensor to the size of AF window (preview is blocked during this mode). To improve the AF speed at macro distances AF algorithm provides Macro search option when the search is limited to macro range.

Auto focus algorithm is able to control a wide variety of AF lens actuators via AF lens driver framework. In total 10 output pins (AF ports) are available for actuator connection: 6 GPIO ports, up to 6 PWM ports and I2C. Most common actuators with voice-coil motor (VCM) or piezo actuators are already supported by corresponding AF drivers in current FW. AF drivers for VCM actuators include special control functions to guarantee smooth and noise-free lens movement and reduce lens vibrations. Support for new actuators can be easily added via hook functions uploaded into system RAM (Trap&Patch).

Auto focus algorithm provides 4 different options to deal with lens actuator hysteresis if the actuator has it: backlash compensation and 3 options for dual fly-back lens control.

Auto focus algorithm includes support for external sensor which provides the algorithm with feedback on subject distance.

3.3.2 AF BASIC TUNING PROCEDURE

Single AF tuning procedure is shown the figure below.

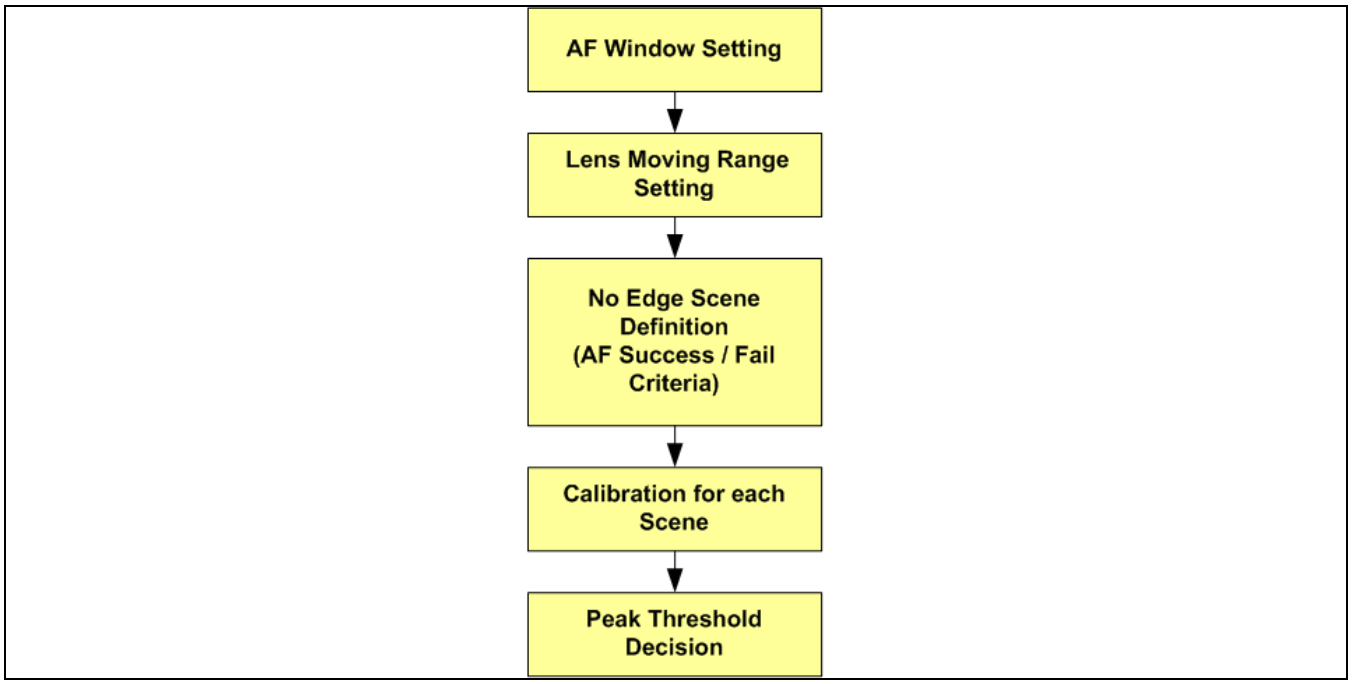

Figure 3-15 Single AF tuning procedure

3.3.3 AF INTERFACE SETTING

Mnemonic	Attr	Description	Deault
REG_TC_IPRM_CM_Init_AfModeType	RW	0:NONE 2:VCM_PWM 3:VCM_I2C	0003
REG_TC_IPRM_CM_Init_PwmConfig1	RW		0000
REG_TC_IPRM_CM_Init_PwmConfig2	RW		0000
REG_TC_IPRM_CM_Init_GpioConfig1	RW		0031
REG_TC_IPRM_CM_Init_GpioConfig2	RW		0000
REG_TC_IPRM_CM_Init_Mi2cBits	RW		200C
REG_TC_IPRM_CM_Init_Mi2cRateKhz	RW	MI2C Speed Ex)400KHz : 400 X 2 = 800	0320

3.3.4 AF WINDOW SETTING

First Window is outer window (large window) and Second Window is inner window (small window). Each window can be placed to any position independently.

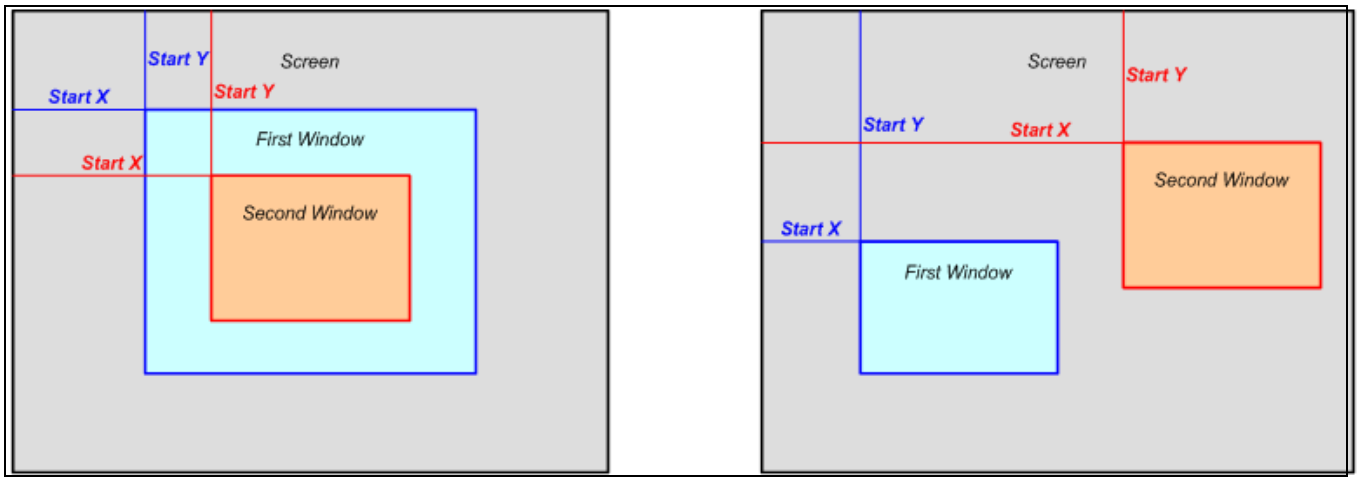

Figure 3-16 AF Window Setting

Table 3-29 Register Map for AF Window control

Mnemonic	Attr	Description	Deault
REG_TC_AF_FstWinStartX		First window's start X position ratio. (0x100 means the 1/4 size of total image H size.)	256
REG_TC_AF_FstWinStartY		First window's start Y position ratio	206
REG_TC_AF_FstWinSizeX		First window's width ratio	512
REG_TC_AF_FstWinSizeY		First window's height ratio	356
REG_TC_AF_ScndWinStartX		Second window's start X position ratio	384
REG_TC_AF_ScndWinStartY		Second window's start Y position ratio	256
REG_TC_AF_ScndWinSizeX		Second window's width ratio	256
REG_TC_AF_ScndWinSizeY		Second window's height ratio	256
REG_TC_AF_WinSizesUpdated		Synchronize FW with updated statistics window set-tings	0000

3.3.5 LENS MOVING RANGE SETTING

Lens position can be set by the value between 0x00 and 0xFF. But the real lens moving range is not matched with it. Firstly, start/end lens position should be found.

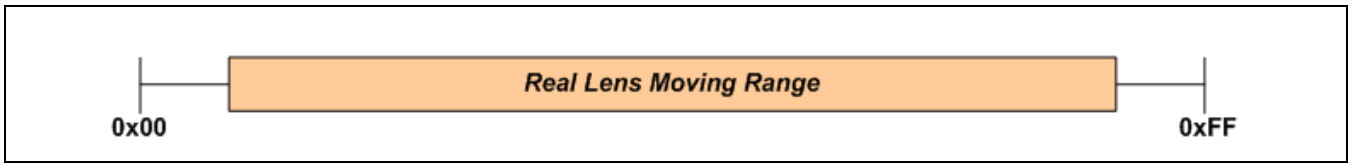

Figure 3-17 Lens Moving Range

“Real Lens Moving Range” can be found by checking the change of view in each lens position.

And the number of lens moving step may be different according to user because it is directly related to AF speed.

1. Lens Step Num↑ : Speed↓ Accuracy↑

2. Lens Step Num↓ : Speed↑ Accuracy↓

If start/end lens position and the step number is decided, set the value to following registers.

Mnemonic	Attr	Description	Default
af_pos_usTableLastInd	RW	Index of last valid position in the lens position table. Used to define the size of the table.	-
af_pos_usTable[0] ~ [31]	RW	Lens position table for Coarse AF search	-

3.3.6 AF COMMAND

The HOST controls AF solution using AF commands. This section describes which commands are available for the host and how they should be used. Parameters related to AF Command are located in section named [REG_TC_AF]. (REG_TC_AF_AfCmd, REG_TC_AF_AfCmdParam)

Mnemonic	Attr	Description	Default
REG_TC_AF_AfCmd	RW	[0]:Stands for [NO COMMAND] [1]:Aborts currently executing command (except INIT command) [2]:AF Sleep command: affects AFD subsystem – actuator IC power down [3]: Initialize AF subsystem (AF driver, AF algorithm) [4]:Manual AF command [5]: Single AF command [6]: Continuous AF command [7]:Reserved AF command (can be inserted using Trap-and-Patch)	-

3.4 AUTO FLICKER CORRECTION (AFC)

3.4.1 ABSTRACTS

The change in quantity of light causes flicker, at a frequency of 50 Hz or 60 Hz. The flicker will appear unless the camera shutter is operated at the rate of integral multiple. The frame rate for a camera controls the rate of flicker band. The shutter speeds cause image luminance.

Following Figure show us a frequency of AC 60Hz light source.

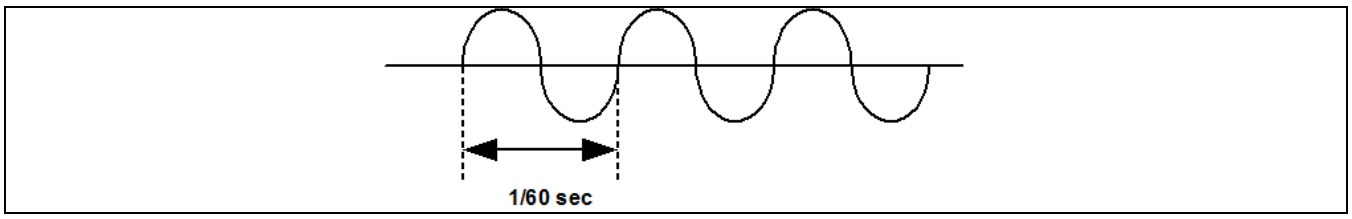

Figure 3-18 Sources of Electricity (60Hz)

Following Figure show us a frequency of power density of AC 60Hz light source. Because of AC source's characteristic, brightness will be changed according to time.

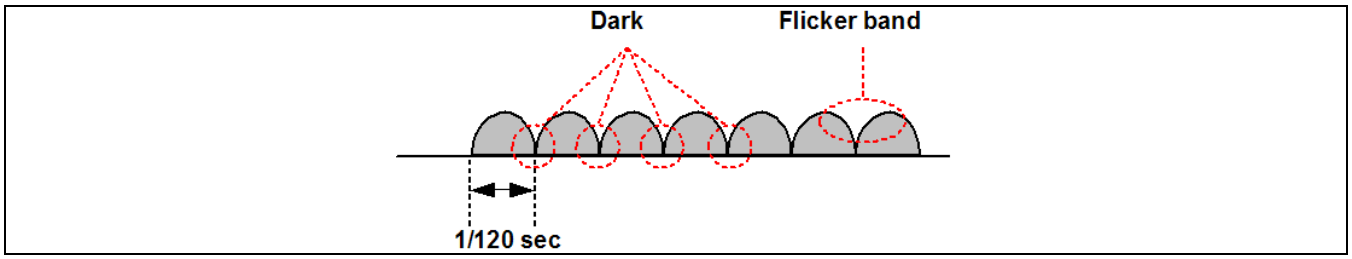

Figure 3-19 Voltage Fluctuation on Lamps (60Hz)

20 Hz and 15 Hz as shown in Figure 3-24 and Figure 3-25 are integral multiples of power frequency. Therefore, the bands are not flowing because the charged energy is the same at every second.

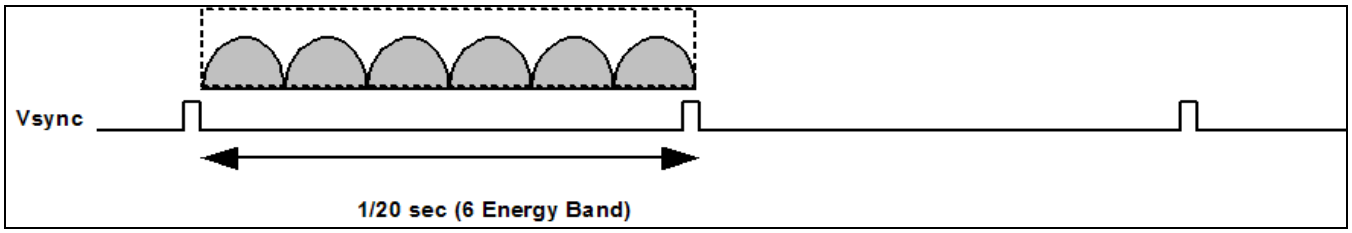

Figure 3-20 Frame rate 20Hz (No Flicker Fluctuation)

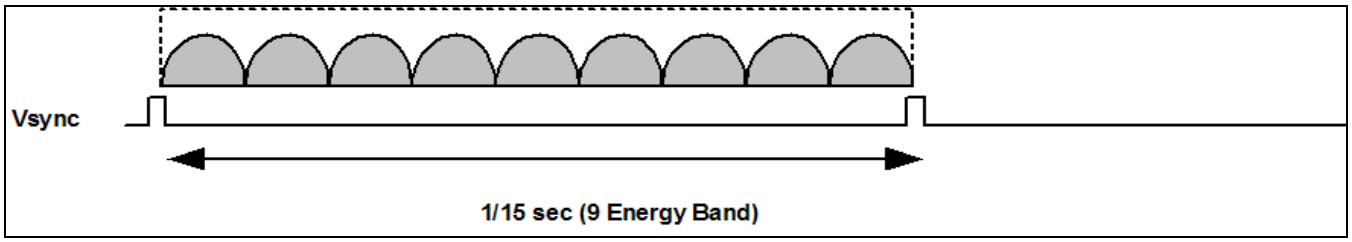

Figure 3-21 Frame rate 15Hz (No Flicker Fluctuation)

If frame rates are not an integral multiple of power frequency, the flicker bands are flowing as shown in Figure 3-26.

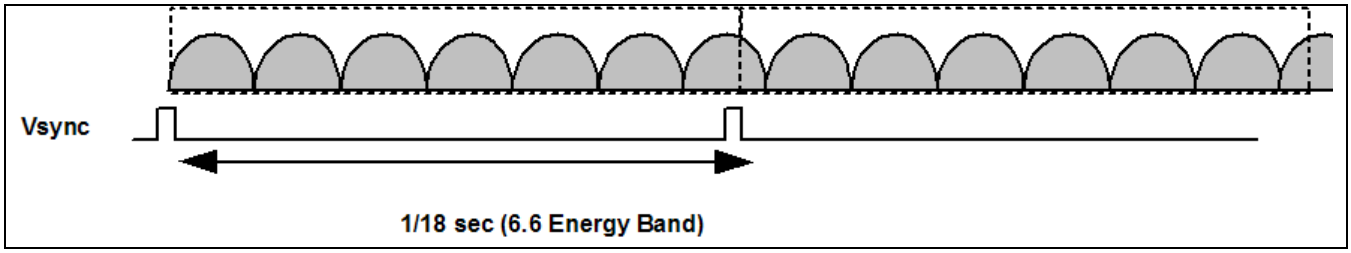

Figure 3-22 Frame Rate 18 Hz (Flicker Fluctuation)

The darkness of flicker band depends on the intensity of luminance at the rate of 20Hz, where flicker band is fixed. Table 3-4 show us the frame rates which have a fixed flicker band. The frame rates which are colored with yellow have a fixed flicker band in both 50Hz and 60Hz.

Table 3-30 Flicker-Fluctuation-less Frame Rate Table

Number of Energy Band	Frame rate of 60Hz Lamp	Frame rate of 50Hz Lamp	Number of Energy Band	Frame rate of 60Hz Lamp	Frame rate of 50Hz Lamp
1	120.00	100.00	31	3.87	3.23
2	60.00	50.00	32	3.75	3.13
3	40.00	33.33	33	3.64	3.03
4	30.00	25.00	34	3.53	2.94
5	24.00	20.00	35	3.43	2.86
6	20.00	16.67	36	3.33	2.78
7	17.14	14.29	37	3.24	2.70
8	15.00	12.50	38	3.16	2.63
9	13.33	11.11	39	3.08	2.56
10	12.00	10.00	40	3.00	2.50
11	10.91	9.09	41	2.93	2.44
12	10.00	8.33	42	2.86	2.38
13	9.23	7.69	43	2.79	2.33
14	8.57	7.14	44	2.73	2.27
15	8.00	6.67	45	2.67	2.22
16	7.50	6.25	46	2.61	2.17
17	7.06	5.88	47	2.55	2.13
18	6.67	5.56	48	2.50	2.08
19	6.32	5.26	49	2.45	2.04
20	6.00	5.00	50	2.40	2.00
21	5.71	4.76	51	2.35	1.96
22	5.45	4.55	52	2.31	1.92

23	5.22	4.35	53	2.26	1.89
24	5.00	4.17	54	2.22	1.85
25	4.80	4.00	55	2.18	1.82
26	4.62	3.85	56	2.14	1.79
27	4.44	3.70	57	2.11	1.75
28	4.29	3.57	58	2.07	1.72
29	4.14	3.45	59	2.03	1.69
30	4.00	3.33	60	2.00	1.67

If the shutter exposure time is longer than 1/120 sec, then the difference of luminance causes weakening of the flicker band as shown in the Figure 3-17.

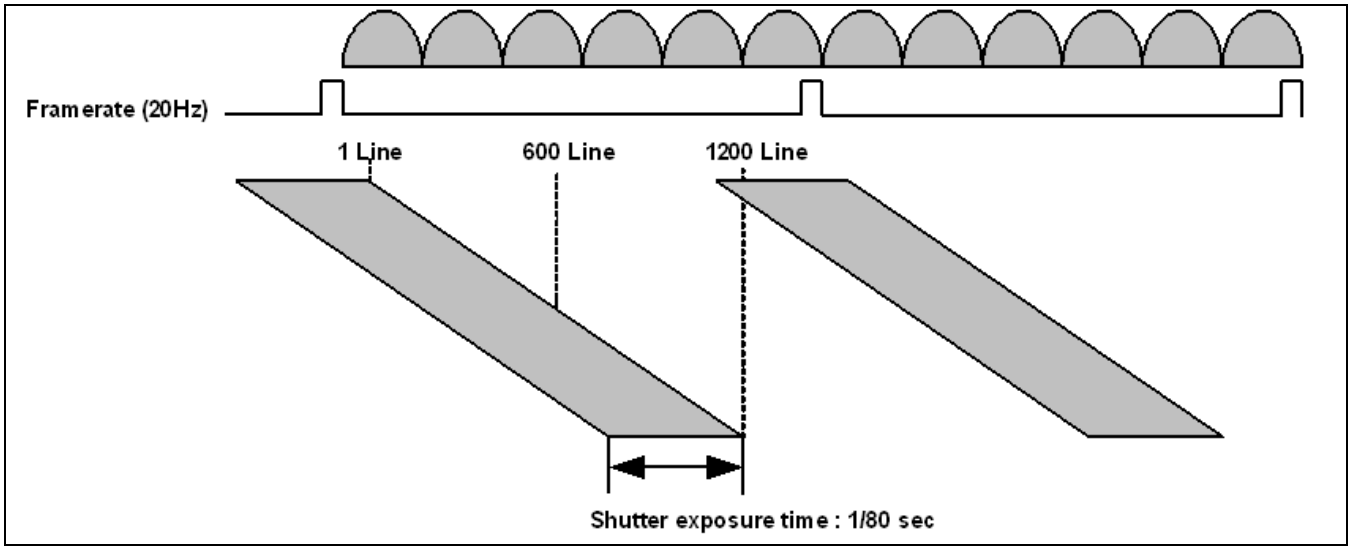

Figure 3-23 Exposure Time of "Weak Flicker Band"

If the shutter exposure time takes multiple of 1/120 sec, then flicker does not occur, because the luminance which is got by CIS is the same for all lines (see Figure 3-28).

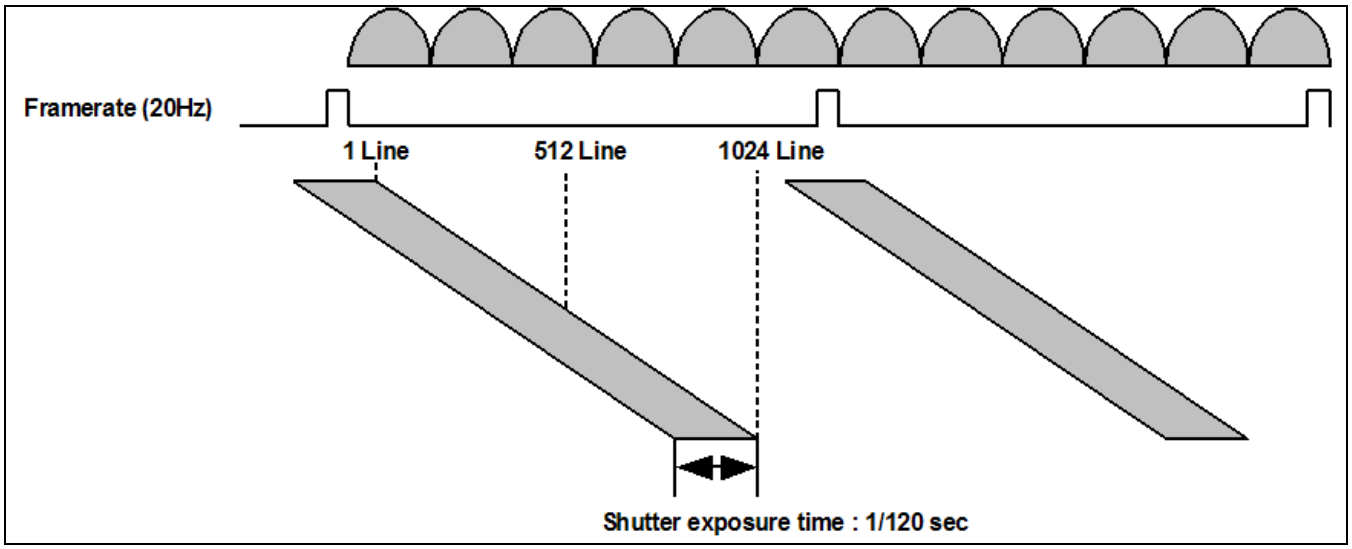

Figure 3-24 Exposure Time of "No Flicker Band"

If the exposure time of shutter is shorter than 1/120 sec, then there is a great difference of exposure in each lines so, the flicker bands occur heavily (see Figure 3-29).

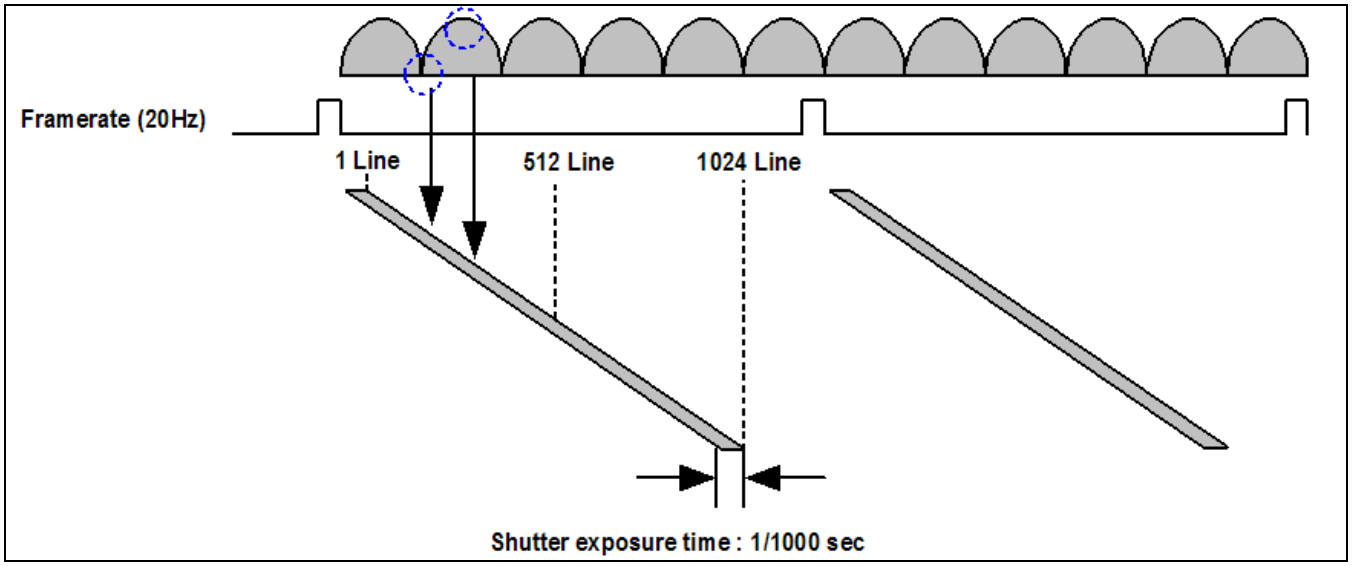

Figure 3-25 Exposure Time of "Strong Flicker Band"

3.4.2 AUTO FLICKER DETECTOR SETTING

3.4.2.1 Function Description

The user can use a flicker detector to set flicker canceller automatically. If the user does not want to use a flicker detector, the user can turn off a flicker detector also. And the user can control a flicker canceller manually.

3.4.2.2 Register Description

These registers are related to setting of Auto flicker detector as below.

Table 3-31 Register Map for Auto Flicker Detector

Mnemonic	Default	Description
AFC_Default60Hz	0x0001	When auto detection mode is enabled, this register will decide the beginning flicker canceling frequency. 1 : Auto detection mode will be started with 60Hz flicker frequency 0 : Auto detection mode will be started with 50Hz flicker frequency
REG_TC_DBG_AutoAlgEnBits	0x077F	[5] : Auto flicker detection enable bit 1b : enable In this mode, the ISP detects the flicker rate automatically, 50Hz or 60Hz, and changes the flicker quantization accordingly 0b : disable In this mode, the ISP does not detect the flicker rate automatically.

3.4.2.3 Usage

There are control examples of an auto flicker detector as below.

Table 3-32 Usage for Controlling of Auto Flicker Detector

Control	Example
Auto flicker detector off	#REG_TC_DBG_AutoAlgEnBits [5] = 0b - If the user turn off auto flicker detector, the user should set manual flicker canceling mode (please see next chapter).
Auto flicker detector on Beginning with 50Hz	#AFC_Default60Hz = 0 #REG_TC_DBG_AutoAlgEnBits [5] = 1b
Auto flicker detector on Beginning with 60Hz	#AFC_Default60Hz = 1 #REG_TC_DBG_AutoAlgEnBits [5] = 1b

3.4.3 FLICKER CANCELLER SETTING

3.4.3.1 Function Description

Auto flicker detector just can find a flicker in image. Flicker canceller gets the information from an auto flicker detector, and then it will control AE to avoid a flicker in image.

NOTE: The AFC requires either dynamic frame rate or fixed frame rate to be selected in register < REG_xTC_PCFG_usFrTimeType/REG_xTC_CCFG_usFrTimeType>. If “FR_TIME_FIXED_ACCURATE” frame type is selected in register < REG_xTC_PCFG_usFrTimeType/REG_xTC_CCFG_usFrTimeType>, this setting will prevent the AFC from changing the exposure time to the correct quantization time, and in effect, AFC is disabled.

3.4.3.2 Register Description

These registers are related to setting of a flicker canceller as below.

Table 3-33 Register Map for Flicker Canceller

Mnemonic	Default	Description
REG_SF_USER_FlickerQuant	0x0000	This register decides which flicker canceling frequency will be used. 00b : do not cancel flicker 01b : set flicker canceller to 50Hz 02b : set flicker canceller to 60Hz If the user wants to apply this register setting, the user should write ‘1’ to “REG_SF_USER_FlickerQuantChanged” register.
REG_SF_USER_FlickerQuantChanged	0x0000	This register applies value of “REG_SF_USER_FlickerQuant” to CIS. 01b : apply value of “REG_SF_USER_FlickerQuant” to CIS This register value is cleared automatically. Do not set this register when the user is using auto flicker detector.
Mon_AFC_usFlExpQuant	0x0000	Information register which has current flicker frequency. 0FA0h : 50Hz flicker canceller 0D04h : 60Hz flicker canceller 0001h : Flicker canceller off

3.4.3.3 Usage

There are control examples of a flicker canceller as below.

Table 3-34 Usage for Controlling of Flicker Canceller

Control	Example
50Hz Flicker canceling mode by manually	#REG_TC_DBG_AutoAlgEnBits [5] = 0b #REG_SF_USER_FlickerQuant = 1 #REG_SF_USER_FlickerQuantChanged = 1 If the user wants to use standby mode, the user must add below setting. #AFC_Default60Hz = 0
60Hz Flicker canceling mode by manual	#REG_TC_DBG_AutoAlgEnBits [5] = 0b #REG_SF_USER_FlickerQuant = 2 #REG_SF_USER_FlickerQuantChanged = 1 If the user wants to use standby mode, the user must add below setting. #AFC_Default60Hz = 1
Flicker canceling off mode by manual	#REG_SF_USER_FlickerQuant = 0 #REG_SF_USER_FlickerQuantChanged = 1 If the user wants to use standby mode, the user must rewrite below command after waking up. #REG_SF_USER_FlickerQuant = 0 #REG_SF_USER_FlickerQuantChanged = 1

3.4.4 RELATIONSHIP BETWEEN AFC AND AE

3.4.4.1 Function Description

After flicker is detected, Auto flicker canceling is worked by exposure control and frame rate. Exposure control is for Auto flicker canceling, exposure control makes same power in all the line in screen. And frame rate control makes flicker band stop in screen.

3.4.4.2 Exposure control for flicker

Exposure control makes exposure time has multiple value of flicker quant in order to make same energy in all the line in screen. This method makes truncate exposure time, so S/W uses more gain for making same brightness to compensate truncated exposure time.

In alternating current, (+) (-) power is not ideally same so S/W can use double flicker quant value. Even though S/W setting is double flicker quant mode, S/W can use flicker quant 1.

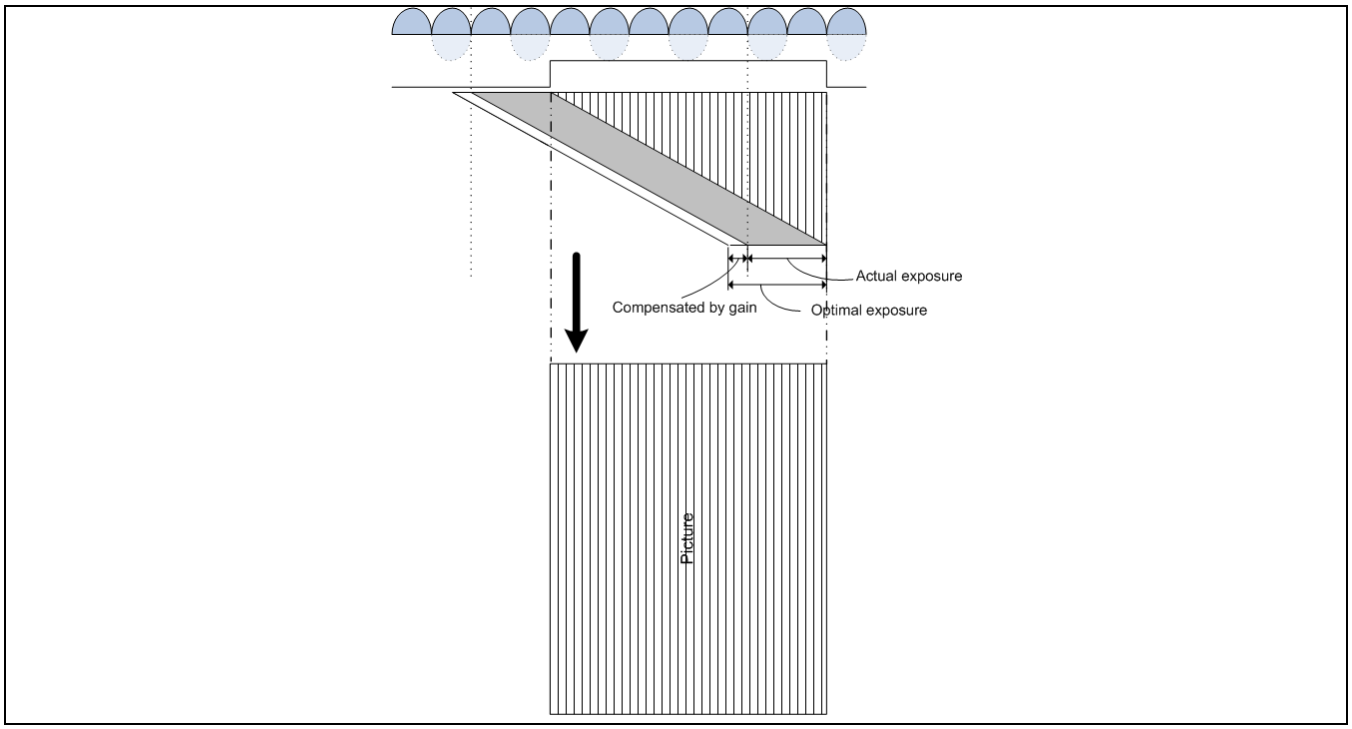

Figure 3-26 Exposure Time Control in Flicker

Above picture show that all line of scene has same multiple flicker quant.

3.4.5 FRAME TIME CONTROL FOR FLICKER.

For making flicker band stop, frame time control is used. S/W makes frame time be multiple flicker quant value. This system stop flicker band in screen.

Frame time also can be chosen double flicker quant value and single quant value.

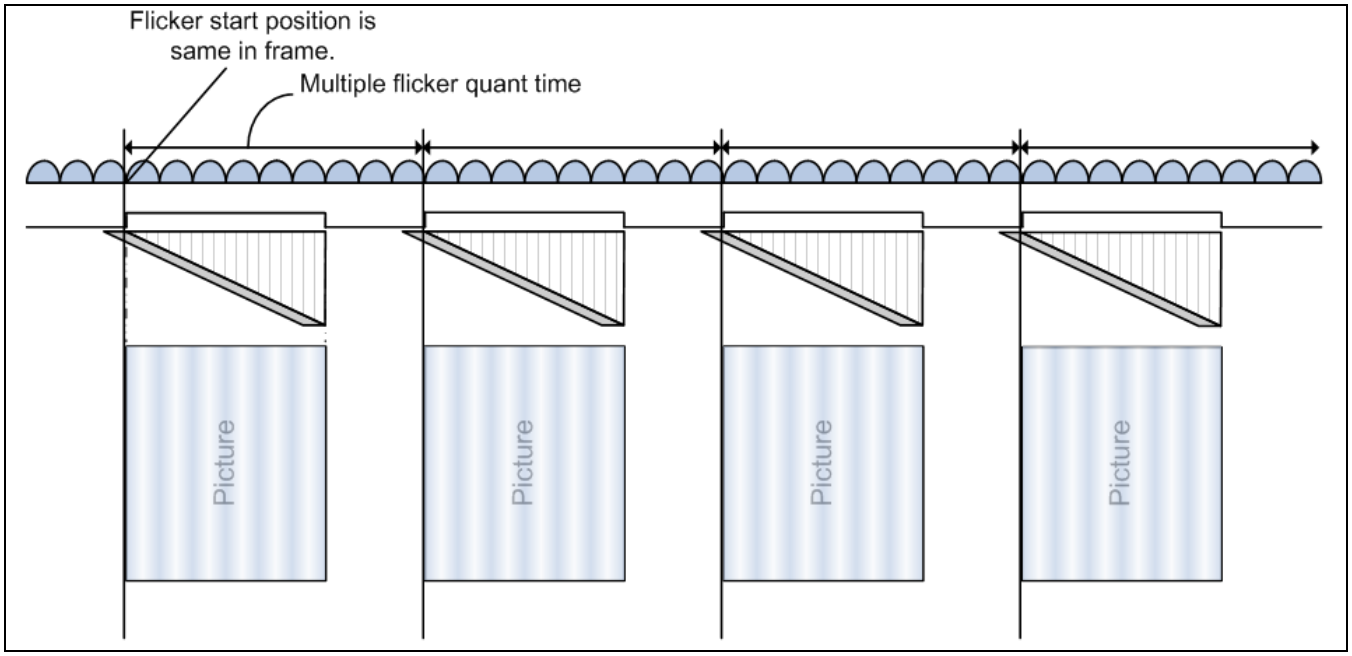

Figure 3-27 Frame Time Control in Flicker

3.4.5.1 Manual flicker quant

S/W controls exposure flicker quant in order to restrict exposure time to meet customer spec.

In flicker table mode, flicker number can be only number in flicker table register. S/W selects smaller flicker number that is in flicker table than optimized flicker number. S/W has 50Hz and 60Hz table so user can make setting for 50Hz and 60Hz.

3.4.5.2 Register

Table 3-35 Table for Controlling of Flicker

Mnemonic	Default	Description
AFC_bUseDoubleQuant	0x0000	This register decides S/W use single quant flicker time or double quant flicker time. 00b : Use single quant flicker time. 01b : Use double quant flicker time.
AFC_ManualQuant	0x0000	This register decides to use manual flicker table on/off. 00b : Not use flicker table. 01b : Use flicker table.
AFC_AeManual50FlickerTable	0x0000	Flicker table value for 50hz. This register uses 8bit data, even though flicker table is 16bit. Because S/W use little endian, number order is changed. Flicker number has to increase.
AFC_AeManual60FlickerTable	0x0000	Flicker table value for 60hz. This register uses 8bit data, even though flicker table is 16bit. Because S/W use little endian, number order is changed. Flicker number has to increase.
AFC_usFrQuantNum	0x0001	This register use to select Frame time quant. 01b : Use single flicker quant for frame time. 02b : Use double flicker quant for frame time.

3.5 LENS SHADING CORRECTION

3.5.1 LENS SHADING CORRECTION

The Anti-shading Profile is a 13x11 grid of weights, distributed uniformly over the image, that are used for shading compensation. There are four sets of weights for each channel of the Bayer image: R, Gr, Gb, B. A default Anti-shading Profile is stored in ROM. A custom Anti-shading Profile can be set in RAM and then the <TVAR_ash_pGAS> register is set to point to the location of the custom Anti-shading Profile, followed by writing 1 to <REG_TC_DBG_ReInitCmd> in order to load the new profile into HW. The default location of Anti-shading profile in ram is in the <Tune_wbt_GAS> register.

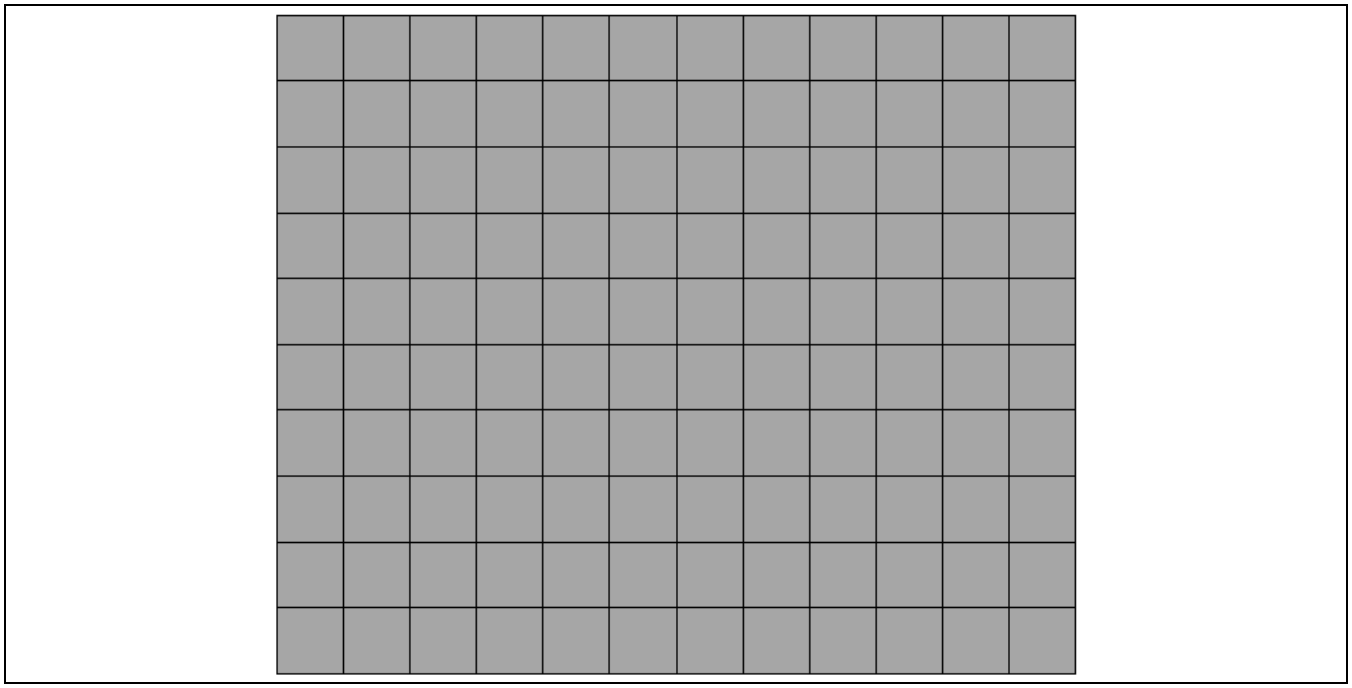

Figure 3-28 Anti-shading Grid

Shading compensation is further tuned by illumination type (color temperature). Each color channel profile is multiplied by a coefficient called an alpha coefficient that depends on the current illumination type. Thus:

$$Pout(x,y) = Pin(x,y) * (1 + GainR(x,y) * \alpha R)$$ for R pixels

$$Pout(x,y) = Pin(x,y) * (1 + GainGr(x,y) * \alpha Gr)$$ for Gr pixels

$$Pout(x,y) = Pin(x,y) * (1 + GainGb(x,y) * \alpha Gb)$$ for Gb pixels

$$Pout(x,y) = Pin(x,y) * (1 + GainB(x,y) * \alpha B)$$ for B pixels

There are seven sets of alpha-coefficients that are calibrated for indoor illuminations DL75, DL65, DL50, CWF, WW, Inca and Horizon illuminations. An additional set is calibrated for outdoor illumination. Normalized brightness is used to distinguish between indoor and outdoor illuminations. AWB gains are used to determine the illumination type (Mon_AWB_RGainProj). If the current illumination type is between two calibrated illumination types the alpha gains are interpolated from the calibrated illuminations.

3.5.2 TUNING PARAMETERS

Table 3-36 Register Map for Anti-shading Control

Control Register	Attr	Description
TVAR_ash_pGAS_high	R/W	Pointer to Anti-shading Profile gains.
TVAR_ash_pGAS_low	R/W	Pointer to Anti-shading Profile gains.
REG_TC_DBG_ReInitCmd	R/W	Signals to FW to load the Anti-shading Profile.
TVAR_ash_AwbAshCord[X]	R/W	Illumination Type (Mon_AWB_RGainProj). X: - [0]: Horizon - [1]: Incandescent - [2]: Warm White - [3]: Cool White - [4]: DL50 - [5]: DL65 - [6]: DL75
TVAR_ash_GASAlpha[X][Y]	R/W	Alpha factors for indoor illumination. Y: - [0]: R - [1]: Gr - [2]: Gb - [3]: B
wbt_bUseOutdoorASH	R/W	0 : Not use outdoor alpha. 1 : Use outdoor alpha.
TVAR_ash_GASOutdoorAlpha[Y]	R/W	Alpha factors for outdoor illumination.

Table 3-37 Register for Alpha monitoring

Control Register	Attr	Description
Mon_AAIO_NextAcqCtxt_GASAlpha[Y]	R/W	Current alpha factors. Y: - [0]: R - [1]: Gr - [2]: Gb - [3]: B

3.6 COLOR CORRECTION

3.6.1 COLOR CORRECTION

Color correction is applied to each pixel with a color correction matrix (CCM). For each pixel the CCM is selected from among six possible CCMs. The calculation is performed as follows:

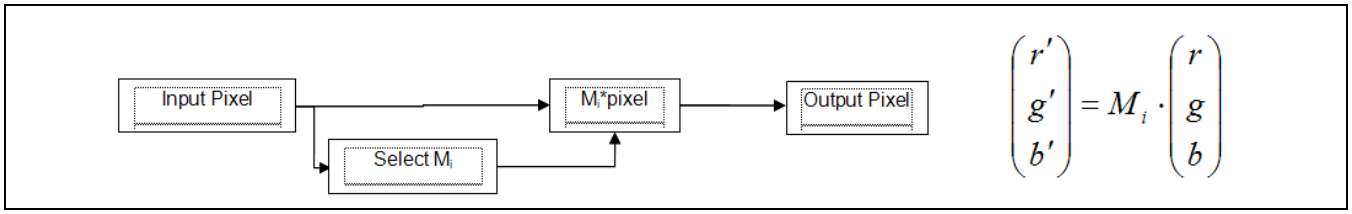

Figure 3-29 Color Correction Matrix

The six CCMs are constructed from six base vectors: R, Y, G, C, B and M.

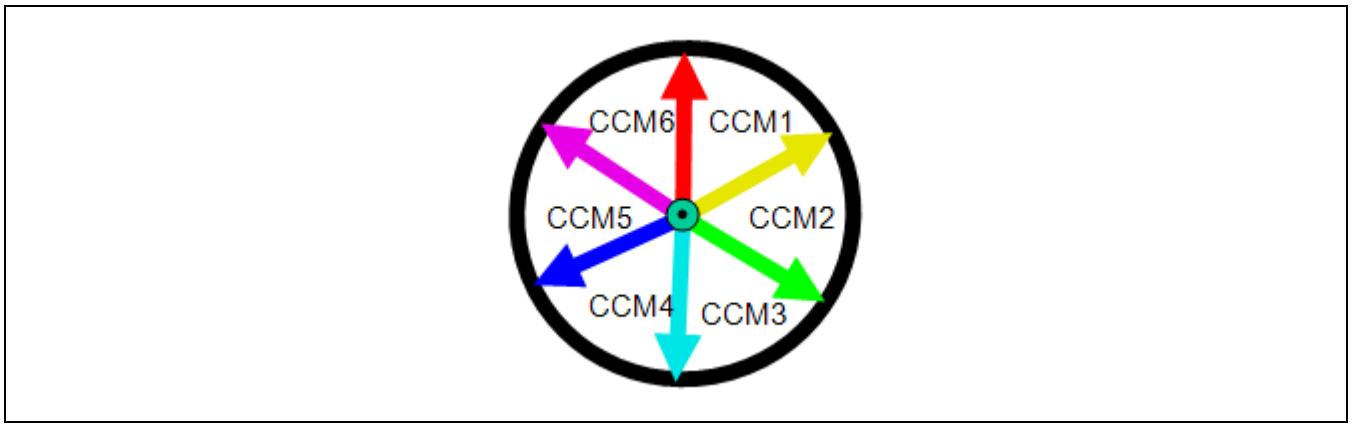

Figure 3-30 Colors of CCM

The CCM to be used is selected according to the relative position of RGB colors in the input pixel (W is the vector for white color):

Table 3-38 CCM Selection Rule

CCM	Selection Rule (from RGB of input pixel)
M0 = [R; Y–R; W–Y]	r >= g and g >= b (red <-> yellow)
M1 = [Y–G; G; W–Y]	g > r and r >= b (yellow <-> green)
M2 = [W-C; G; C–G]	g >= b and b > r (green <-> cyan)
M3 = [W–C; C–B; B]	b > g and g >= r (cyan <-> blue)
M4 = [M-B; W–M; B]	b >= r and r > g (blue <-> magenta)
M5 = [R; W–M; M–R]	r > b and b > g (magenta <-> red)

Single CCM can be emulated by selecting R, G and B vectors (observing that R+G+B=W) and deriving from them Y, C and M: Y=R+G; C=G+B; M=B+R.

Color correction is further fine tuned by providing six sets of the base vectors for different types of indoor illumina-

tion (color temperature) and another set for outdoor illumination. The sets for indoor illuminations are calibrated for DL65, DL50, CWF, WW, Inca and Horizon illuminations. AWB gains are used to determine the illumination type (Mon_AWB_RGainProj). Normalized brightness is used to distinguish between indoor and outdoor illuminations. If the current illumination type is between two calibrated illumination types the CCMs are interpolated from the calibrated illuminations.

3.6.2 TUNING PARAMETERS

The CCMs data is stored in ROM. Custom CCMs can be set in RAM and then have <TVAR_wbt_pBaseCcms> and <TVAR_wbt_pBaseCcmsAddr> registers to point to the location of the CCMs. The default location of indoor CCMs in RAM is <Tune_wbt_BaseCcms> and for outdoor CCM is <Tune_wbt_OutdoorCcm>.

Table 3-39 Register Map for CCM control

Mnemonic	Attr	Description
TVAR_wbt_pBaseCcms	R/W	Pointer to base indoor CCMs.
TVAR_wbt_pBaseCcmsAddr[N]	R/W	Array of pointers to indoors CCMs. The values of the array is based are inferred from TVAR_wbt_pBaseCcms: [0] : TVAR_wbt_pBaseCcms + 0. [1] : TVAR_wbt_pBaseCcms + 36. [2] : TVAR_wbt_pBaseCcms + 72. [3] : TVAR_wbt_pBaseCcms + 108. [4] : TVAR_wbt_pBaseCcms + 144. [5] : TVAR_wbt_pBaseCcms + 180.
TVAR_wbt_pOutdoorCcm	R/W	Pointer to outdoor CCM.
SARR_AwbCcmCord[X]		Illumination Type (Mon_AWB_RGainProj). X: - [0]: Horizon - [1]: Incandescent - [2]: Warm White - [3]: Cool White - [4]: DL50 - [5]: DL65 The values of Illumination Type must be increasing (SARR_AwbCcmCord[i] < SARR_AwbCcmCord[i+1]).
Tune_wbt_BaseCcms[X][Y]	R/W	Default location for indoor CCMs. Y: - [0]: R vector - [3]: Y vector - [6]: G vector - [9]: C vector - [12]: B vector

		- [15]: M vector (1x = 256)
Tune_wbt_OutdoorCcm[Y]	R/W	Default location for outdoor CCM.
SARR_CcmsArray[X]	R	Currently used CCMs (not the base vectors but actual CCMs). [0] : M0 [9] : M1 [18] : M2 [27] : M3 [36] : M4 [45] : M5

3.7 GAMMA CORRECTION

3.7.1 GAMMA

Gamma correction is done with a 16-point lookup table for each of the color channels R, G, B. The input values of the table are fixed at 0, 4, 8, 16, 32, 64, 96, 128, 192, 256, 384, 512, 640, 768, 896, and 1023. The outputs are user set according to the following table:

Table 3-40 Register Map for RGB Gamma (Indoor/Outdoor)

Mnemonic	Attr	Description
SARR_usGammaLutRGBIndoor[0][X]	R/W	Indoor Gamma LUT for R. X: 0-15.
SARR_usGammaLutRGBIndoor[1][X]	R/W	Indoor Gamma LUT for G. X: 0-15.
SARR_usGammaLutRGBIndoor[2][X]	R/W	Indoor Gamma LUT for B. X: 0-15.
SARR_usGammaLutRGBOutdoor[0][X]	R/W	Outdoor Gamma LUT for R. X: 0-15.
SARR_usGammaLutRGBOutdoor [1][X]	R/W	Outdoor Gamma LUT for G. X: 0-15.
SARR_usGammaLutRGBOutdoor [2][X]	R/W	Outdoor Gamma LUT for B. X: 0-15.

3.7.2 GAMMA RELATED AFIT PARAMETERS

Gamma curve can be controlled by following AFIT parameters.

Table 3-41 Gamma related AFIT Parameters

Mnemonic	Attr	Description
AFIT8_RGBGamma2_iLinearity	R/W	Strength of the RGB gamma compensation. (0~128) = 0 – Linearity is 100%. Final gamma curve is linear. = 64 – Linearity is 50% = 128 – Linearity is 0 %. This parameter does not impact the gamma curve.
AFIT8_RGBGamma2_iDarkReduce	R/W	Strength of dark power reduction in RGB Gamma(0~255)

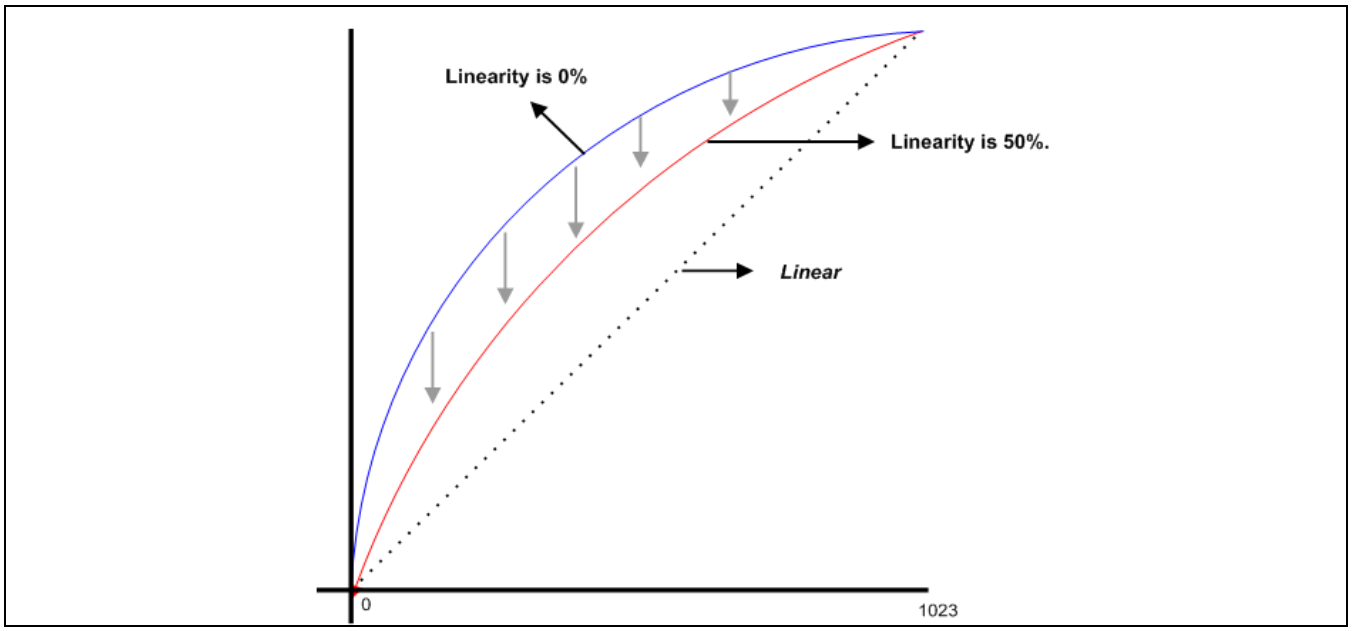

Figure 3-31 Gamma Linearity

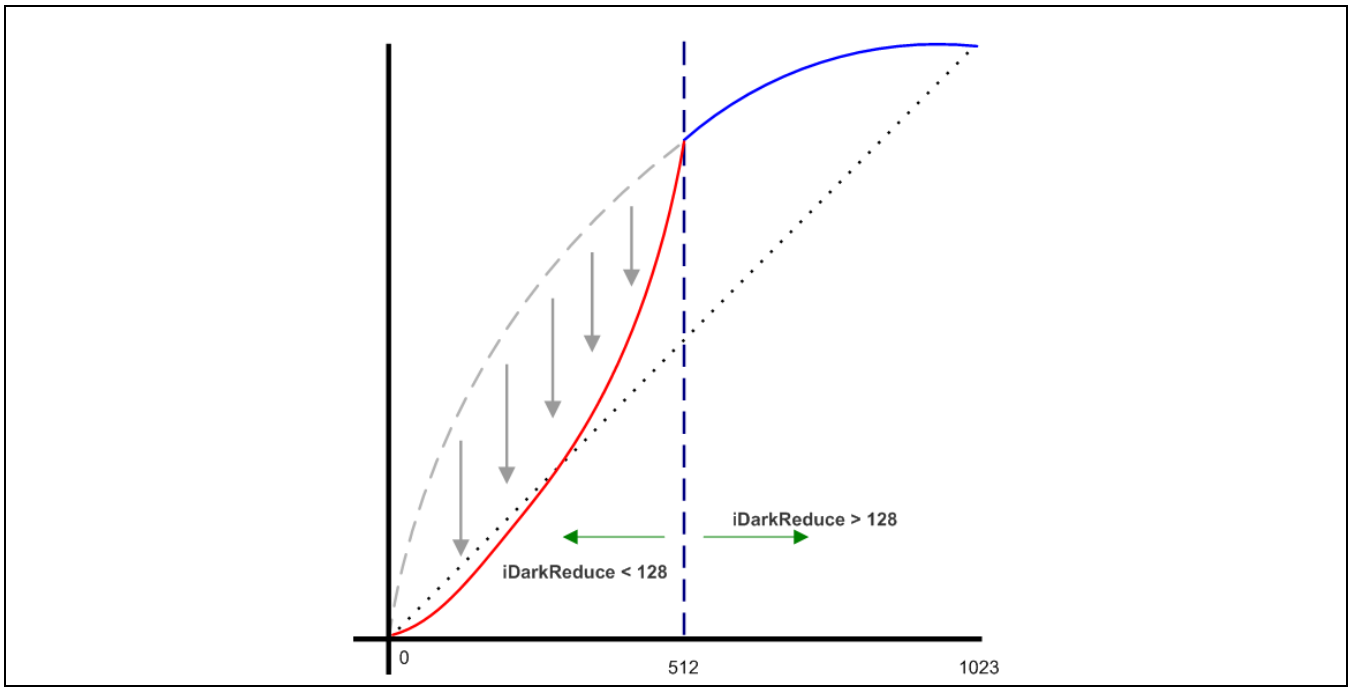

Figure 3-32 Dark Reduce

3.8 AUTOMATIC FINE IMAGE TUNING (AFIT)

Caution: Due to their low level nature, AFIT parameters are usually tuned by the manufacture. The following describes the basic principles of operation.

3.8.1 NOISE INDEX

The Automatic Fine Image Tuning (AFIT) algorithm adjusts various parameters of the Denoising, Desparity, Despeckle, Sharpening and other blocks as a function of the current Noise Index.

Table 3-42 Register for Noise Index monitoring

Control Register	Attr	Description
Mon_AFIT_uNoiseInd	R	Current Noise Index.

The Noise Index is calculated as follows.

Table 3-43 Parameters for Noise Index calculation

Parameter	Description
dg	Digital Gain: Mon_AAIO_NextAcqCtxt_DGain. (1x = 256)
dg_pwr	Digital Gain Power: afit_DigGainPower. (1x = 256)
rg	Red Gain: Mon_AAIO_NextAcqCtxt_RGBGains[0] (1x = 1024)
gg	Green Gain: Mon_AAIO_NextAcqCtxt_RGBGains[1] (1x = 1024)
bg	Blue Gain: Mon_AAIO_NextAcqCtxt_RGBGains[2] (1x = 1024)
ag	Analog Gain: Mon_AAIO_NextAcqCtxt_Again (1x = 256)

3.8.2 DYNAMIC AFIT PARAMETERS

There are 5 sets of dynamic AFIT parameters. Each set has a different balance between noise reduction and sharpening. Thus an AFIT set with high noise reduction and low sharpening should be used in low illumination (low NI) conditions and an AFIT set with low noise reduction and high sharpening should be used in high illumination (high NI) conditions.

The current AFIT set is calculated from the 5 predefined AFIT based on the current NI and indoor / outdoor condition as follows:

1. The current NI is used to lookup into <afit_uNoiseIndInDoor>:

a. If NI < afit_uNoiseIndInDoor[0] then IL0 = IR0 = 0.

b. If NI > afit_uNoiseIndInDoor[4] then IL0 = IR0 = 4.

Otherwise: afit_uNoiseIndInDoor[IL0] < NI <= afit_uNoiseIndInDoor[IR0] (IR0 = IL0 + 1).

IW = (NI - afit_uNoiseIndInDoor[IL0]) / (afit_uNoiseIndInDoor[IR0]- afit_uNoiseIndInDoor[IL0])

Similarly OL0, OR0 and OW are found based on <afit_uNoiseIndOutDoor>.

2. A direct lookup into <afit_InDoorList> and <afit_OutDoorList>:

a. IL = afit_InDoorList[IL0]; IR = afit_InDoorList[IR0];

b. OL = afit_OutDoorList[OL0]; OR = afit_OutDoorList[OR0];

In the default case the mapping is one to one i.e. IL=IL0, IR=IR0, OL=OL0 and OR=OR0.

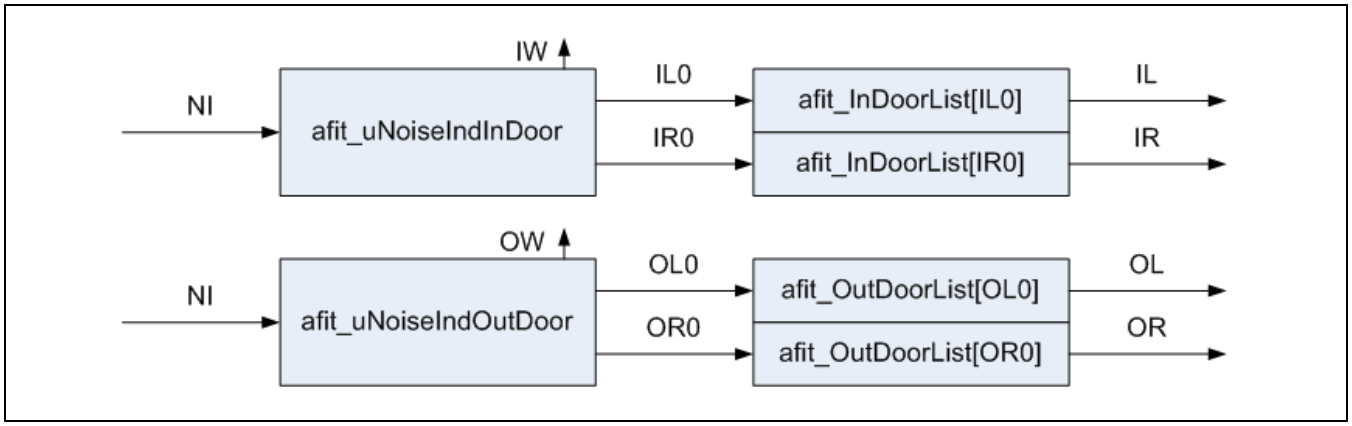

Figure 3-33 AFIT parameter calculation sequence

3. An indoor AFIT set, IndoorBaseVals, is calculated from a linear interpolation between the AFIT sets AfitBaseVals[IL] and AfitBaseVals[IR]: IndoorBaseVals = AfitBaseVals[IL]*IW + AfitBaseVals[IR]*(1-WI). Similarly, an outdoor AFIT set, OutdoorBaseVals is calculated from AfitBaseVals[OL] and AfitBaseVals[OR].

4. The final AFIT set, AfitVals is calculated: AfitVals = IndoorBaseVals*(1-OutDoorW) + OutdoorBaseVals* OutDoorW. Where OutDoorW = <Mon_AAIO_PrevFrmData_OutDoorW>/256 is the confidence level of outdoor conditions.

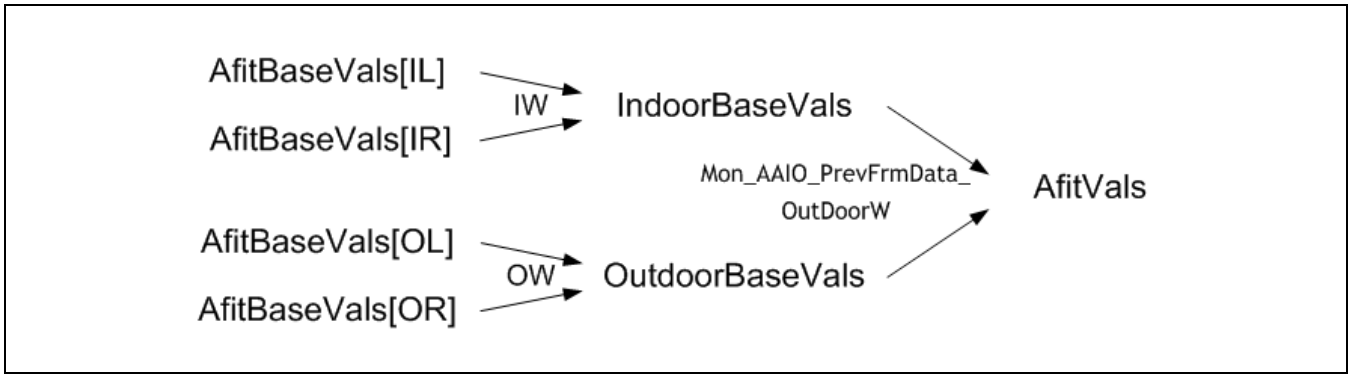

Figure 3-34 Final AFIT value calculation

Table 3-44 Register Map for Dynamic AFIT control

Control Register	Attr	Description
afit_uNoiseIndInDoor[X]	R/W	List of NI for indoor conditions. X=0-4.
afit_InDoorList[X]	R/W	Map AFIT set index to a new value for indoor conditions. Default mapping is trivial. X=0-4.
TVAR_afit_pBaseVals	R/W	Pointer to AFIT values sets.
AfitBaseVals[X][Y]	R/W	Default location for AFIT values sets. X=0-4 is the AFIT set index. Y is the index of parameter:

3.8.3 CONSTANT AFIT PARAMETERS

There are some AFIT parameters which are fixed for all conditions.

Table 3-45 Register Map for Constant AFIT control

Control Register	Attr	Description
afit_pConstBaseVals	R/W	Pointer to constant AFIT values.
ConstAfitBaseVals [X]	R/W	Default location for AFIT values sets. X:

3.9 JPEG CONTROL

The 4ECGX CIS has an on the fly JPEG encoder. If the user wants to use a JPEG output format, the user can set various JPEG modes as following chapters.

3.9.1 GENERAL JPEG

The user can use a JPEG output and control a JPEG quality by setting to the following register

Table 3-46 Register Map for general JPEG control

Mnemonic	Default	Description
REG_xTC_PCFG_Format	0x0005	9 : FORMAT_JPEG
REG_xTC_CCFG_Format	0x0009	9 : FORMAT_JPEG
REG_TC_BRC_usPrevQuality	0x0055	The JPEG quality of the preview mode when the user turn off BRC Recommended values : - Best Quality : 85d - Good Quality : 50d - Poor Quality : 20d **NOTE:** Maximum/Minimum quality value can be limited
REG_TC_BRC_usCaptureQuality	0x0055	The JPEG quality of the capture mode when the user turn off BRC Recommended values : - Best Quality : 85d - Good Quality : 50d - Poor Quality : 20d **NOTE:** Maximum/Minimum quality value can be limited

3.9.2 SPOOF MODE

In this mode, the user can fix the size of a JPEG output by setting some registers. The Spoof mode has three parts of packets, the first part has a JPEG data packets (If the user uses the video interleave mode, additionally this part includes video packets), the second part has dummy packets, the third part has information. If the size of JPEG is smaller than the Spoof size, rest of output packets will be filled with dummy packets and information packets. If the size of JPEG is bigger than the Spoof size, the information packet will be notified an overflow to the user.

In the Spoof mode, frame rate can be lower than normal JPEG mode. Because CIS can not know a JPEG size before encoding, rest of packet will be outputted after encoding whole image data, and if the rest of data is too many, CIS can not output all data in one frame time.

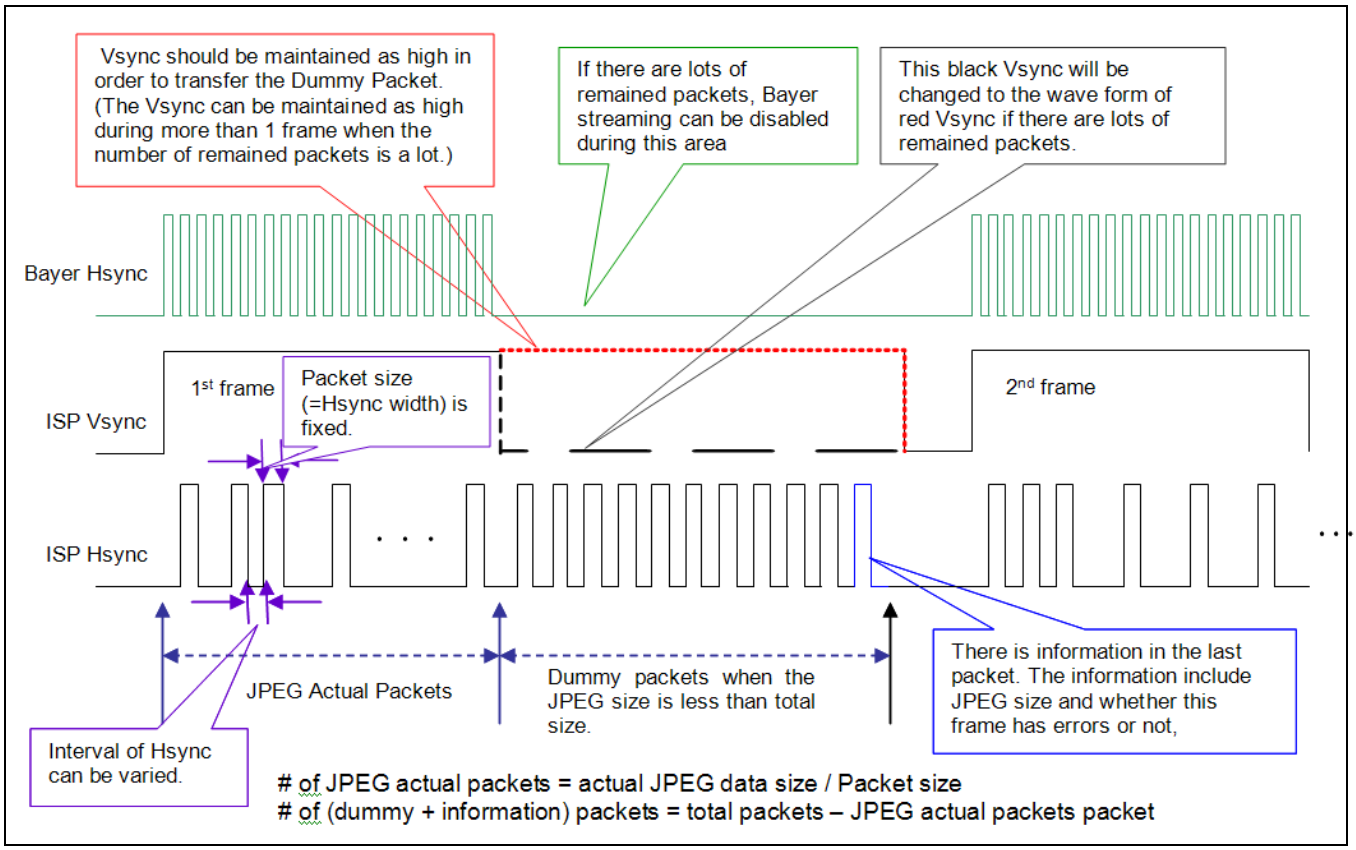

Figure 3-35 Example sync of Spoof JPEG

Vsync length can be varied if there are lots of remained packets.
(Remained packet = Total packets – actual packet)

But, even though Vsync length is varied the numbers of packets which are outputted in one frame are same in every frame. And last packet always has information.

The format of the JPEG Spoof packet which is used all options is like the figure below.

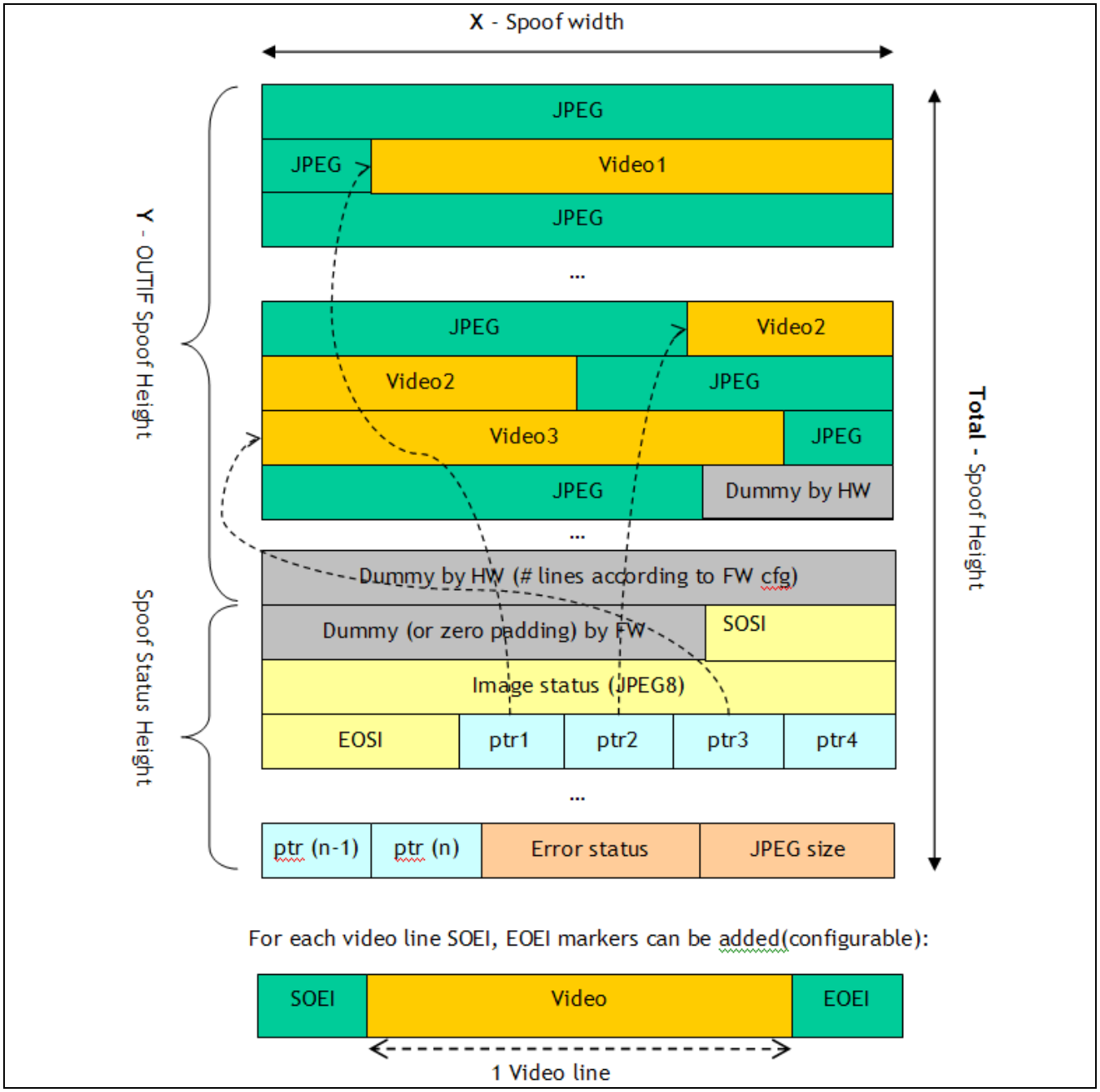

Figure 3-36 Example of Spoof JPEG packet with full options

X - Spoof width mean the packet size(= packet width = Hsync width), the user can set this size by setting the register.

Total – Spoof height mean the numbers of packets, the user can set this size by setting the register.

3.9.2.1 Contents of the Spoof Information Packet

Information packets can have three information packets, the first is a JPEG8 status packet (The user can select this by option), the second is a video pointer (The user can select this by option), the third is a Spoof status packet (mandatory).

Table 3-47 Contents of Spoof information packet

Name/ Description		Size(byte)	Units
Dummy (or zero padding)		size de-pendant	
SOSI (Start Of Status Information) – JPEG8 marker 0xFFBC		2	
JPEG8 status information (Padded Size)	JPEG8 status information size1 in bytes(This information include the size1/2)	4	
	Status Information Version	4	
	CHIP ID(4bytes), EVT number (4bytes) e.g. A0A4AEAA, A0A0A0A0	8	
	Image resolution – Width (4 bytes), Height (4 bytes)	8	
	Thumbnail resolution - Width (4 bytes), Height (4 bytes)	8	
	Exposure time in uSec -	8	uSec
	Frame time in mSec (1/FPS)	4	mSec
	Analog gain	4	8:8
	Digital gain	4	8:8
	White balance gains – R (4 bytes), G (4 bytes), B (4 bytes)	12	8:8
	User brightness setting	4	
	User contrast setting	4	
	AF current lens position (when applicable)	4	
	JPEG8 status information size2 in bytes(This information include the size1/2)	4	
EOSI (End Of Status Information) – JPEG8 marker 0xFFBD		2	
Video line pointers (when applicable) – 3 bytes each pointer		3*(Thumb line size)	
Error status: 0 – Frame O.K Else – Frame Error, Frame should be ignored		2	
JPEG size – Number of JPEG bytes in frame		3	

All fields are in format of Big Indian (MSB first)

If the user does not use “OIF_B_JPEG8_SI_PADDING” option, the size of JPEG status information should be half and ‘0x0A’ will not be padded.

3.9.2.2 Register Description

If the user wants to use Spoof mode, the user must write a value which is bigger than zero to the “usJpegTotal-Packets” register before changing a configuration.

Table 3-48 Register Map for JPEG Spoof mode

Mnemonic	Default	Description
REG_xTC_PCFG_usJpegPacketSize	0x01E0	The size of JPEG packet of capture mode - If the user wants to use a none Spoof mode, this value must be set to multiple of 12 - If the user uses none Spoof interleave mode, packet size will be fixed by header size
REG_xTC_PCFG_usJpegTotalPackets	0x0000	The number of JPEG packets of preview mode 0 : Spoof mode is disabled Else : Spoof mode is enabled
REG_xTC_CCFG_usJpegPacketSize	0x01E0	The size of JPEG packet of capture mode - If the user wants to use a none Spoof mode, this value must be set to multiple of 12 - If the user uses none Spoof interleave mode, packet size will be fixed by header size
REG_xTC_CCFG_usJpegTotalPackets	0x0000	The number of JPEG packets of capture mode 0 : Spoof mode is disabled else : Spoof mode is enabled
REG_xTC_PCFG_OIFMask	0x0000	[4] : OIF_B_SPOOF_ENABLE - The user must set a JPEG total packet size to bigger than 0 to use this option. This option just informs the tool that CIS is working in Spoof mode, so although the user writes a ‘1’ to this bit, the CIS will be no affected.
REG_xTC_CCFG_OIFMask	0x0000	[4] : OIF_B_SPOOF_ENABLE - The user must set a JPEG total packet size to bigger than 0 to use this option. This option just informs the tool that CIS is working in Spoof mode, so although the user writes a ‘1’ to this bit, the CIS will be no affected.

3.9.3 JPEG8

The user can use a type of JPEG8 packet by register setting. 4ECGX can support JPEG8 status and JPEG8 video marker (This marker will be applied when the user uses a interleave mode). In case of PVI mode, JPEG8 status can be supported only when the user uses a Spoof mode.

JPEG8 has following markers.

Table 3-49 The marker which is related to JPEG8

Maker Value	Maker Name
0xFFBE	SOEI, Start of Embedded image marker
0xFFBF	EOEI, End of Embedded Image marker
0xFFBC	SOSI, Start of Status Inform marker
0xFFBD	EOSI, End of Status Inform marker

JPEG8 status information packet has a format as Table 3-58

Each byte[7:4] of JPEG8 status information packet should be padded with ‘0x0A’ (except the JPEG8 marker) to avoid outputting a ‘0xFF’ in a JPEG packer. So the size of JPEG8 status information packet will be doubled of the original size. The user can pad the ‘0x0A’ to JPEG8 status information packet by setting an “OIF_B_JPEG8_SI_PADDING”

Table 3-50 Example of the JPEG8 padding

Original information	Padded information
0x04 0xEA	0xA0 0xA4 0xAE 0xAA

3.9.3.1 Register Description

The user can use JPEG8 mode by enabling following register before changing a configuration.

Table 3-51 Register Map for JPEG8

Mnemonic	Default	Description
REG_xTC_PCFG_OIFMask	0x0000	[6] : OIF_B_ENABLE_JPEG8 [12] : OIF_B_JPEG8_SI_PADDING - Padding 0x0A at JPEG8 status info[7:4], using with the “OIF_B_ENABLE_JPEG8”
REG_xTC_CCFG_OIFMask	0x0000	[6] : OIF_B_ENABLE_JPEG8 [12] : OIF_B_JPEG8_SI_PADDING - Padding 0x0A at JPEG8 status info[7:4], using with the “OIF_B_ENABLE_JPEG8”

3.9.4 INTERLEAVE MODE

The user can use a JPEG interleave mode only in the capture mode. The user can select a video size of interleaving up to QVGA and select some kind of options which are related to classifying between video and JPEG packet.

3.9.4.1 Interleave in the Spoof mode

There are two methods to classify a video packet in the Spoof mode. One is using the JPEG8 embedded video marker, the other is using the video pointer array (Of course the user can use both). The format of packet is like the Figure 3-44. If the user does not use options for classifying there is no way to classify a video packet in JPEG packets.

3.9.4.2 Interleave in normal mode

There are two methods to classify a video packet in the no Spoof mode. One is using a DT(Data Type) marker, the other is no using a DT marker.

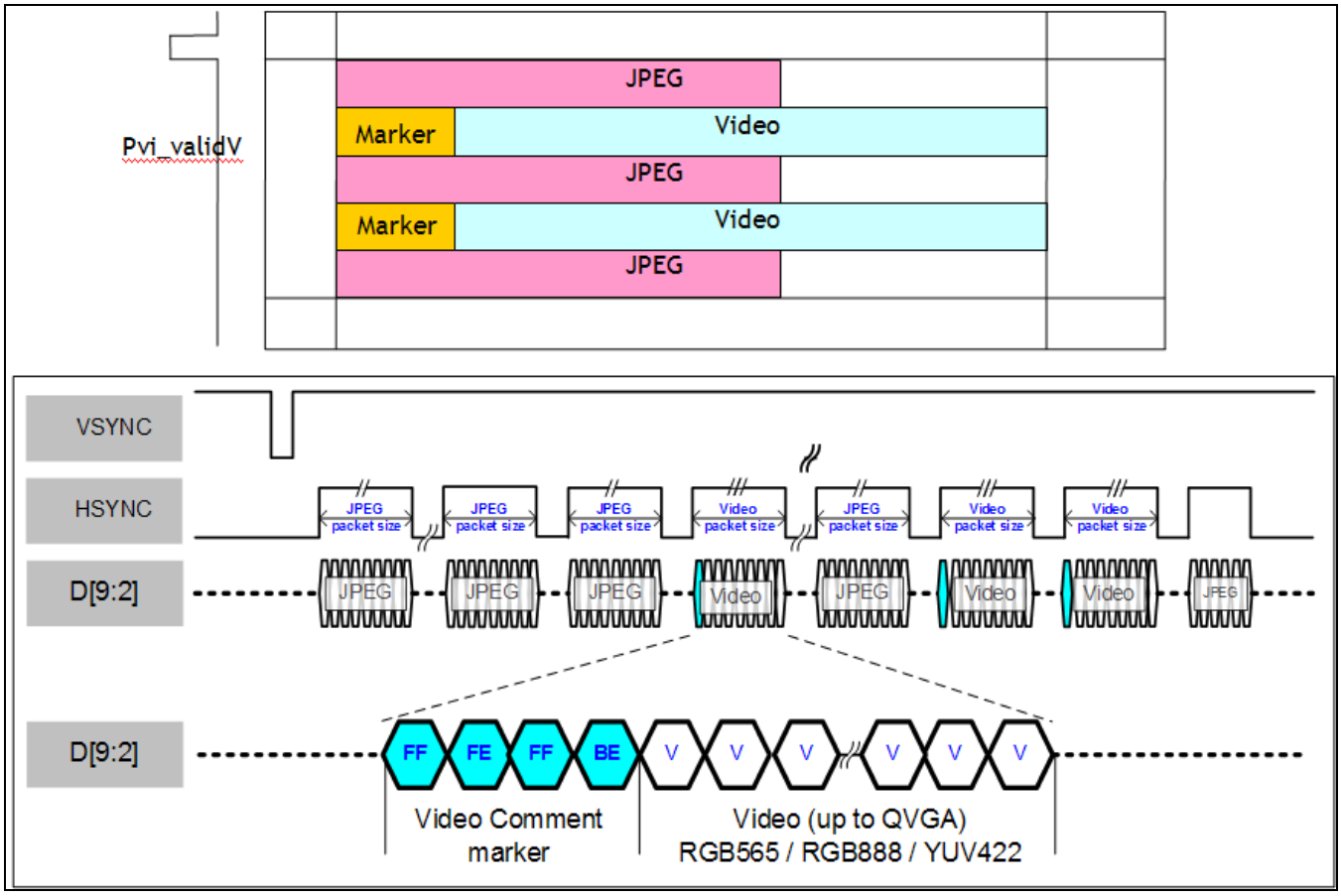

Figure 3-37 Example of normal interleave packet without DT marker

If the user dose not use a DT marker, JPEG/Video interleave packet will be outputted as Figure 3-44

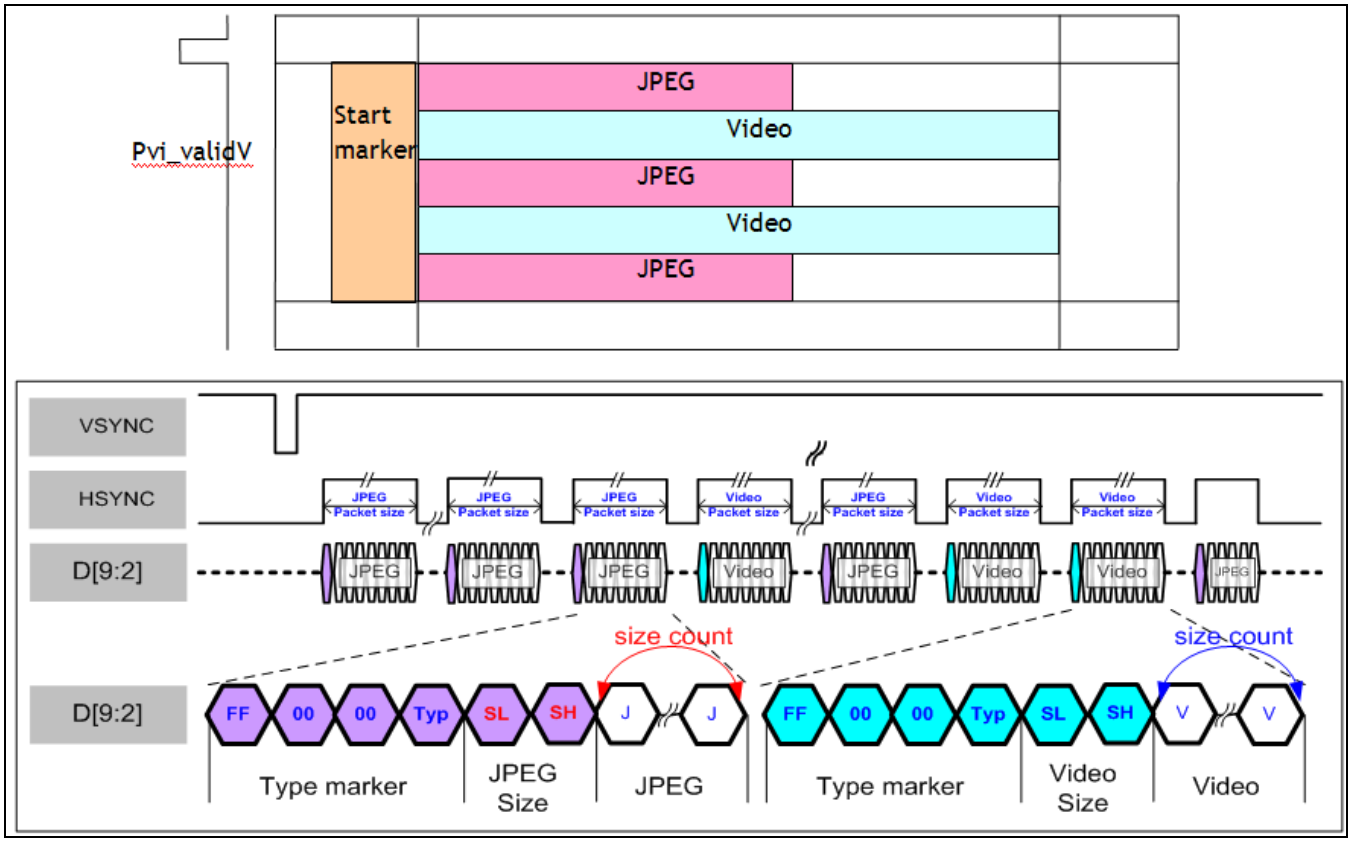

Figure 3-38 Example of normal interleave packet with DT marker

If the user uses a DT marker, JPEG/Video interleave packet will be outputted as Figure 3-45

The user can know the data type of a packet by reading the type byte. The type of DT marker is as Table 3-63

Table 3-52 Description of the Data Type Marker for Interleave DT Marker

Value	Data Type
0x12	Embedded Data
0x1E	YUV422
0x22	RGB565
0x24	RGB888
0x30~0x33	JPEG

3.9.4.3 Register Description

The user can use JPEG interleave mode by enabling following register before changing a configuration.

Table 3-53 Register Map for JPEG interleave mode

Mnemonic	Default	Description
REG_TC_THUMB_Thumb_bActive	0x0000	0 : JPEG interleave mode is off 1 : JPEG interleave mode is on - This option can be used in JPEG capture mode
REG_TC_THUMB_Thumb_uWidth	0x0000	The width of the video image during JPEG interleave mode
REG_TC_THUMB_Thumb_uHeight	0x0000	The height of the video image during JPEG interleave mode
REG_TC_THUMB_Thumb_Format	0x0001	The format of video during JPEG interleave mode 0 : FORMAT_RGB565 1 : FORMAT_RGB888 5 : FORMAT_FULL_YUV (YUV422 0-255)
REG_xTC_PCFG_OIFMask	0x0000	[5] : OIF_B_SPOOF_USE_VIDEO_PTR [6] : OIF_B_ENABLE_JPEG8 [12] : OIF_B_JPEG8_SI_PADDING - Padding 0x0A at JPEG8 status info[7:4], using with the “OIF_B_ENABLE_JPEG8”
REG_xTC_CCFG_OIFMask	0x0000	[5] : OIF_B_SPOOF_USE_VIDEO_PTR [6] : OIF_B_ENABLE_JPEG8 [12] : OIF_B_JPEG8_SI_PADDING - Padding 0x0A at JPEG8 status info[7:4], using with the “OIF_B_ENABLE_JPEG8”
REG_xTC_PCFG_PVIMask	0x0002	[10] PVI_B_JPEG_DT_MARKER_BITS_BIT : When bit is set JPEG DT marker is enabled
REG_xTC_CCFG_PVIMask	0x0002	[10] PVI_B_JPEG_DT_MARKER_BITS_BIT : When bit is set JPEG DT marker is enabled

3.9.5 BRC (BIT RATE CONTROL FOR JPEG)

Basically the BRC function can control a JPEG quality to fit a target size of JPEG through iteration. The probability of fitting a target JPEG size will be increased if the BRC function can consume many frames.

The BRC function has three options. The user can set the options to following register "REG_TC_BRC_BRC_type".

The acceptance criteria are ±10% of target size.

3.9.5.1 Inspection Mode

If the user uses the "INSPECTION_ON", JPEG will be working in preview mode and be estimating a JPEG size then this estimated quality2 will be applied to JPEG of capture mode.

3.9.5.2 Auto Size Setting of BRC in the Spoof Mode

If the user uses the "AUTO_SPOOF_BRC", the target JPEG size will be calculated automatically by the Spoof size. In this mode, a target size of JPEG which is set by the user will be ignored and then a target size of JPEG is set by the value of "jpeg_BppAutoBrcSpoofPercentage" register and the Spoof size.

3.9.5.3 Register Description

The user can use BRC function by enabling following register before changing a configuration.

Table 3-54 Register Map for JPEG BRC

Mnemonic	Default	Description
REG_xTC_PCFG_uBpp88	0x0300	The target JPEG file size of preview mode The format of this register is 8.8 fixed point The user can calculate JPEG file size as below : (BPP88 * Image Size) / 2048 If the inspection of BRC is on, BRC will use Bpp88 of capture mode
REG_xTC_CCFG_uBpp88	0x0300	The target JPEG file size of capture mode The format of this register is 8.8 fixed point The user can calculate target JPEG file size as below : ("This register value" * Image Size) / 2048
REG_TC_BRC_BRC_type	0x0000	[0] : BRC_ON [1] : INSPECTION_ON - This option must be used with [0] = 1b. - If the user uses a preview with YUV and a capture with JPEG, CIS will predict JPEG size in YUV preview mode. But, JPEG HW will be working in YUV mode too. And the result of prediction can be wrong case by case. - The inspection will be active only when preview dimensions

2 Note: This estimation can not be correct in all case, so JPEG overflow can occur.

		are 16*8 multiply [2] : AUTO_SPOOF_BRC - This option must be used with [0] = 1b. - If the user uses this option, CIS will set a target JPEG size with Spoof size.(Please refer "jpeg_BppAutoBrcSpoofPercentage") [3] : LIMIT_MAX_Q_BY_USER - When the user use the BRC, maximum quality value is limited by the user set quality value("REG_TC_BRC_usPrevQuality", "REG_TC_BRC_usCaptureQuality")
jpeg_BrcMinQuality	0x0010	Minimum quality factor if the user will be using BRC
jpeg_BppAutoBrcSpoofPercentage	0x00DA	The percentage of JPEG size in the Spoof mode (Format of this register is 8.8 fixed point) If the user use the "AUTO_SPOOF_BRC" mode, Target size of JPEG will be decided as below : Target JPEG size = Spoof total size * (this register value / 256) – Interleave video size
skl_bBlockCaptureOnBadBRC	0x0001	0 : Turning off this option 1 : Skipping the first frame of capture to avoid size overflow when the user use a BRC in JPEG mode

3.9.6 MODIFYING THE Q TABLE

If the user wants to use a Q table which is made by the user, the user can use that.

Before using a JPEG, the user must write user's Q table to an empty memory and write the address of the user's Q table to the register in the Table 3-43. If the user wants to use untouched user's Q table, the user must turn off the BRC and write 50d to "REG_TC_BRC_usPrevQuality"/"REG_TC_BRC_usCaptureQuality" register. Otherwise the CIS will modify a Q table for CIS's functions using the user's Q table.

3.9.6.1 Inputting order of Q Table

The value of Q table will be applied to JPEG file by raster scan order. So if the user writes the Q table as below order, Q table will be applied to JPEG file as below example.

Table 3-55 Example of Q table raster order

Q table order of the user writing	Raster scan order	Q table order of the JPEG file header
01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F 20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F 30 31 32 33 34 35 36 37 38 39 3A 3B 3C 3D 3E 3F 40		01 02 09 11 0A 03 04 0B 12 19 21 1A 13 0C 05 06 0D 14 1B 22 29 31 2A 23 1C 15 0E 07 08 0F 16 1D 24 2B 32 39 3A 33 2C 25 1E 17 10 18 1F 26 2D 34 3B 3C 35 2E 27 20 28 2F 36 3D 3E 37 30 38 3F 40

3.9.6.2 Register Description

The user can use BRC function by modifying following register before using a JPEG

Table 3-56 Register Map for the user Q table

Mnemonic	Default	Description
jpeg_pLumQT	0x012366	The address of JPEG Q table for luma
jpeg_pChromQT	0x0123A6	The address of JPEG Q table for chroma

3.10 SETTING FOR MIPI

Output clock unit is byte clock. But in MIPI case, data is outputted in bit unit.

e.g.> OutRate4KHz = 2CECh, PVI = 46 MHz, MIPI = 368 MHz

“REG_xTC_PCFG_uClockInd” will be made following order.

- Total index = from 0 to (“REG_TC_IPRM_UseNPviClocks” + “REG_TC_IPRM_UseNMipiClocks” – 1)
- PVI clock will be placed at index 0.
- MIPI clock will be placed after last PVI clock.

e.g> if “REG_TC_IPRM_UseNPviClocks” is 2 and “REG_TC_IPRM_UseNMipiClocks” is 1. Clock index 0~1 is for PVI clock and index 2 is for MIPI clock

3.10.1 REGISTER DESCRIPTION

The user can set a MIPI by modifying following register

Table 3-57 Register for MIPI Setting

Mnemonic	Attr	Description
REG_TC_IPRM_MinOutRate4KHz_x	R/W	Minimal output rate of first clock in KHz divided by 4
REG_TC_IPRM_MaxOutRate4KHz_x	R/W	Maximal output rate of first clock in KHz divided by 4
REG_xTC_PCFG_PVIMask REG_xTC_CCFG_PVIMask	R/W	[6] PVI_B_VCLK_OUT_ACTIVE_NO_DATA : Additional modes (no clock on v-sync, clock overlap h-sync etc.) are set separately : When the user wants to use MIPI, this bit control “DPHY continuous clock”3 - 0b : DPHY disable continuous clock. - 1b : DPHY enable continuous clock.
REG_xTC_PCFG_uClockInd	R/W	System clock index (0~2)
REG_0TC_PCFG_usMaxOut4KHzRate	R/W	Upper limit of output clock(PCLK) in KHz unit
REG_0TC_PCFG_usMinOut4KHzRate	R/W	Lower limit of output clock(PCLK) in KHz unit There is a minimum required output clock which is calculated by FW automatically based on image size, output format and so on. If this required clock can be placed in the range which is limited by above upper limit and lower limit, then trial for Configuration change will be succeeded. If not, then Configuration change will not happen
REG_TC_IPRM_UseNMipiClocks	R/W	Number of PLL configurations to be computed for MIPI (0-2) Total sum of UseNPviClocks and UseNMipiClocks can not be greater than 3

3 NOTE : This option is changed in EVT1(0b is enable in EVT0)

		UseNPviClocks and UseNMipiClocks are also indexes which determine what is used among following clock sets.
REG_TC_IPRM_NumberOfMipiLanes	R/W	Number of MIPI lanes (0: PVI, 1: 1 lane MIPI, 2: 2 lane MIPI)

3.11 SETTING FOR ITU-656

Registers which are related to ITU-656 are separated into HW and SW.

3.11.1 REGISTER DESCRIPTION

If the user wants to use ITU-656 type, the user can set ITU-656 type through below registers

Table 3-58 SW Register for ITU-656

Mnemonic	Attr	Description
REG_xTC_PCFG_OIFMask REG_xTC_CCFG_OIFMask	R/W	[3] : OIF_B_USE_ITU656_MARKERS - Please refer the "REG_TC_OIF_ITU656bits" register
REG_TC_OIF_ITU656bits	R/W	If the user uses a "OIF_B_USE_ITU656_MARKERS" option, this value will be written at 0xD000B052(It is a HW register, so please see a HW register map)
REG_TC_OIF_ITU656Blank	R/W	If the user uses a "OIF_B_USE_ITU656_MARKERS" option, this value will be written at 0xD000B056(It is a HW register, so please see a HW register map)

Table 3-59 HW Register for ITU-656

Mnemonic	Addr	Attr	Description
outif_pvi_656_config	0xB052	R/W	[4] outregs_pvi_blank_line - Insert PVI Blank line [3] outregs_pvi_656_field - 0b : Use PVI 656 field 0 only - 1b : Toggle 0/1 in PVI 656 field [2] outregs_pvi_656_qual - 0b : Active data only in valid H - 1b : Active data and start marker(except end marker) in valid H [1] outregs_pvi_656_sync - 0b : Tie H/V sync to ground - 1b : H/V sync will be outputted [0] outregs_pvi_656_marker - 0b : PVI 656 marker off - 1b : PVI 656 marker on
outif_pvi_blank_line_w	0xB056	R/W	[15:0] PVI blank line length

3.12 TIMINGS FOR MODE CHANGE

3.12.1 INTERVAL BETWEEN PREVIEW AND CAPTURE MODE

Besides changing between preview and capture, all of configuration changes sequence is almost same.

e.g> Size changing, output format changing and etc.

3.12.1.1 Interval in worst case

Interval time is dependent on current setting.

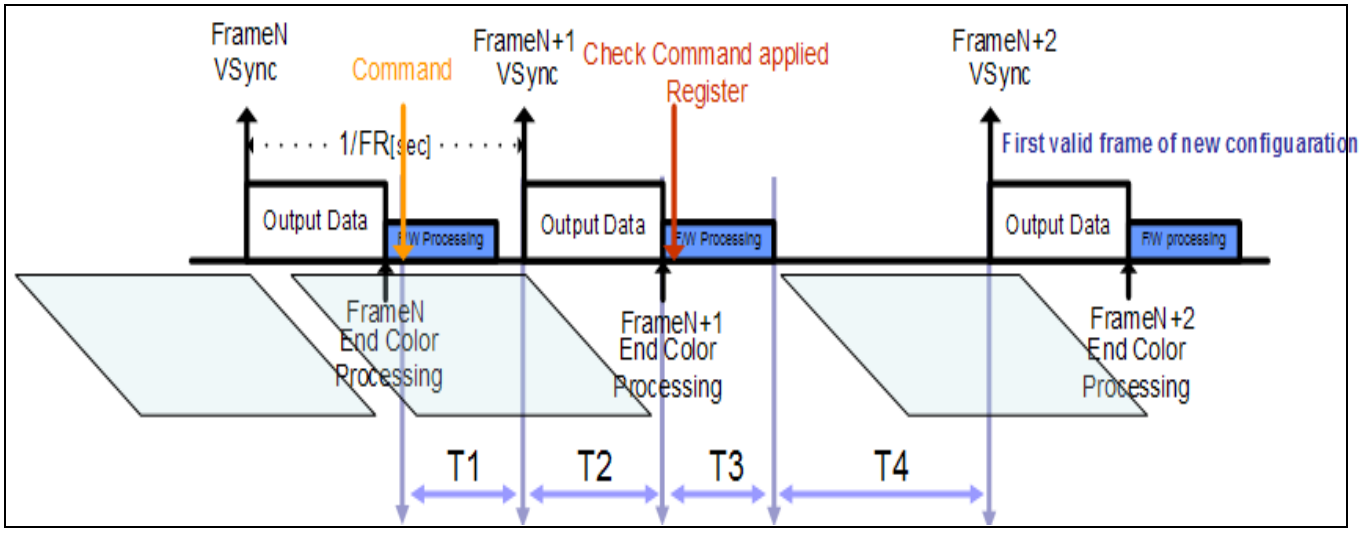

Figure 3-39 Interval in worst case for mode change

T1 : Vblank time (Almost same with Vblank)

T2 : Frame size of preview without Vblank

T3 : FW needed time for process

“setot_uOnlineClocksDiv40”, this time is ARM cycle(=System clock / 2)

Default value of 4ECGX is 110,000 ARM cycles(=220,000 System clock is needed)

T4 : Exposure time(= Capture frame time in worst case)

3.12.1.2 Interval in best case

Interval time is dependent on current setting

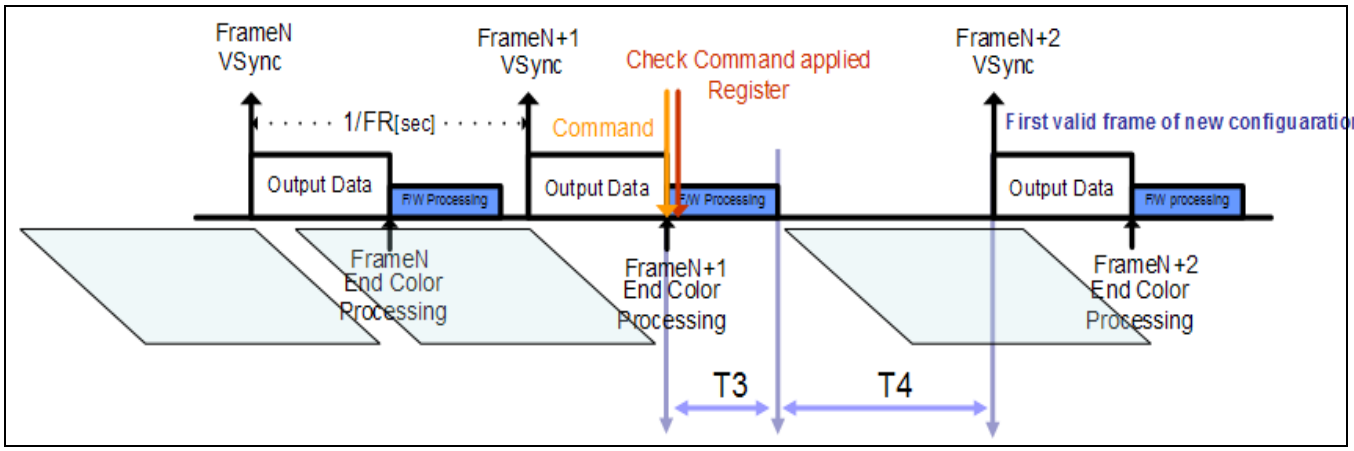

Figure 3-40 Interval in best case for mode change

T1 : None

T2 : None

T3 : FW needed time for process

“setot_uOnlineClocksDiv40”, this time is ARM cycle(=System clock / 2)

Default value of 4ECGX is 110,000 ARM cycles(=220,000 System clock is needed)

T4 : Exposure time(= Capture frame time in worst case)

3.13 FLASH CONTROL

4EC support LED Flash.

3.13.1 LED FLASH

Usually mobile phone use LED flash for making brightness because consumption of power is small.

4EC uses GPIO to control LED flash. LED GPIO is synchronized with configuration so it can be turned on in capture frame unless LED flash is not forced mode.

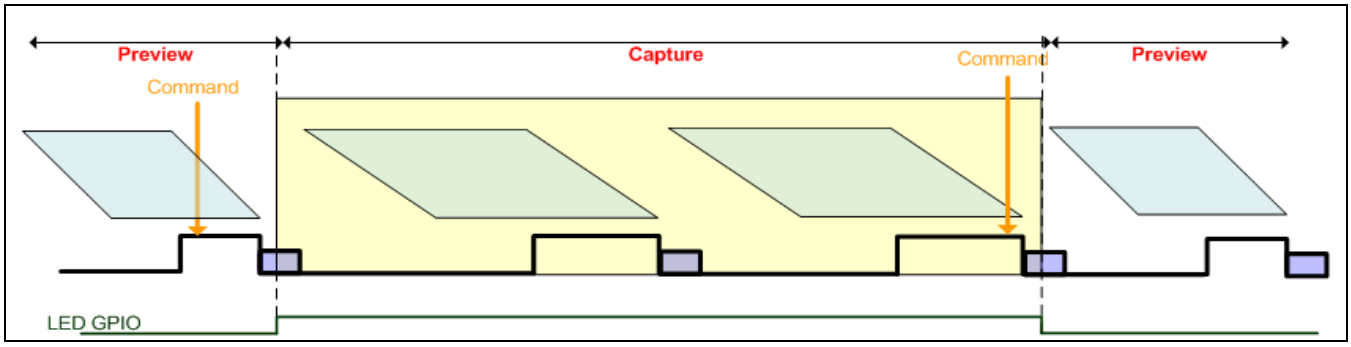

Figure 3-41 LED control

Table 3-60 Register Map for LED control

Mnemonic	Attr	Description
REG_TC_FLS_Mode	R/W	LED flash mode. 0 : LED flash disable. 1 : LED flash On.(LED GPIO is on if once this value is set. It is not synchronized with exposure.) 2 : LED flash is burst on in every capture. (It doesn't care threshold) 3 : LED flash on/off is controlled automatically by threshold.
REG_TC_FLS_Threshold	R/W	LED flash threshold.
REG_TC_FLS_Polarity	R/W	LED flash GPIO polarity.

3.13.2 LED GPIO

LED and xenon flash GPIO setting can be set "REG_TC_IPRM_LedGpio" register. "REG_TC_IPRM_LedGpio" register is registered in initialize code in F/W so register have to be placed before first configuration change in setting file.

Table 3-61 Register Map for LED GPIO control

Mnemonic	Attr	Description
REG_TC_IPRM_LedGpio	R/W	LED and xenon flash GPIO number. LED GPIO can be GIPO 1~8.

3.14 CHIP VERSION INFORMATION

The user can know version information of the 4ECGX by reading following registers.

Table 3-62 Register for Version information

Mnemonic	Default	Description
REG_FWverControlStr_usFWsenID	0x4EC0	Version information
REG_FWverControlStr_usHWversion	0x0000	Revision information

4 MOBILE CAMERA PHONE APPLICATION S/W GUIDE

4.1 PHONE SOFTWARE GUIDE

4.1.1 SUB-SAMPLING

Table 4-1 Register for Sub-sampling control

Attr	Control Register	Description
R/W	sw.REG_xTC_PCFG_FrRateQualityType	Frame rate quality: TC_FRVSQ_DYNAMIC = 0, TC_FRVSQ_BEST_FRRATE = 1, TC_FRVSQ_BEST_QUALITY = 2

4.1.2 HORIZONTAL MIRROR AND VERTICAL FLIP

4.1.2.1 Horizontal Mirror

Table 4-2 Register for Horizontal Mirror control

Attr	Control Register	Description
R/W	REG_xTC_PCFG_uPrevMirror	Preview mirror mode (X/Y) - Bit mask
R/W	REG_xTC_PCFG_uCaptureMirror	Capture mirror mode (X/Y) - Bit mask

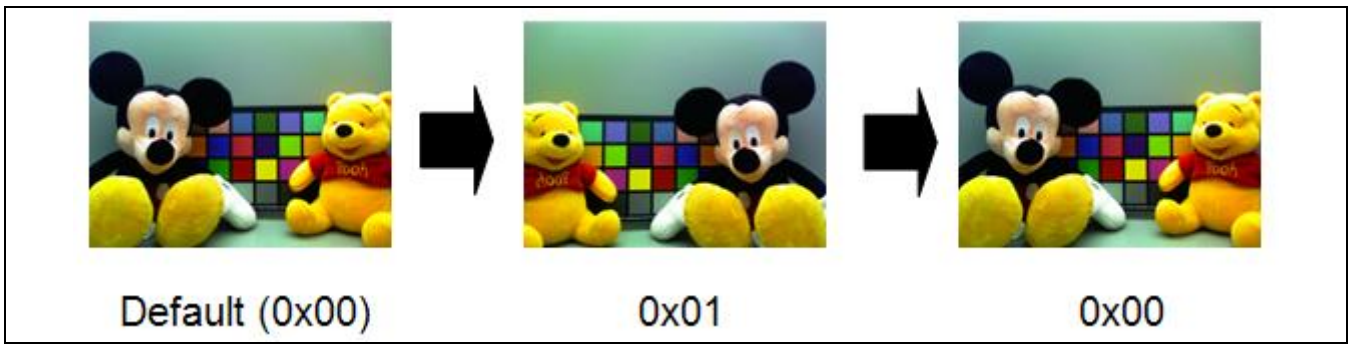

Figure 4-1 Horizontal Mirror

4.1.2.2 Vertical Flip

Table 4-3 Register for Vertical Flip control

Attr	Control Register	Description
R/W	REG_xTC_PCFG_uPrevMirror	Preview mirror mode (X/Y) - Bit mask
R/W	REG_xTC_PCFG_uCaptureMirror	Capture mirror mode (X/Y) - Bit mask

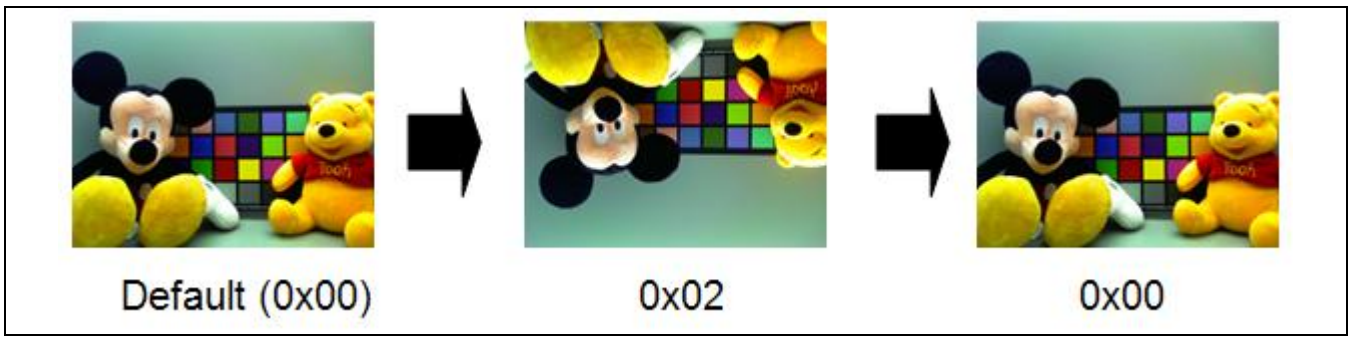

Figure 4-2 Vertical Flip

4.1.2.3 Horizontal Mirror & Vertical Flip

Table 4-4 Register for H-Mirror & V-Flip

Attr	Control Register	Description
R/W	REG_xTC_PCFG_uPrevMirror	Preview mirror mode (X/Y) - Bit mask
R/W	REG_xTC_PCFG_uCaptureMirror	Capture mirror mode (X/Y) - Bit mask

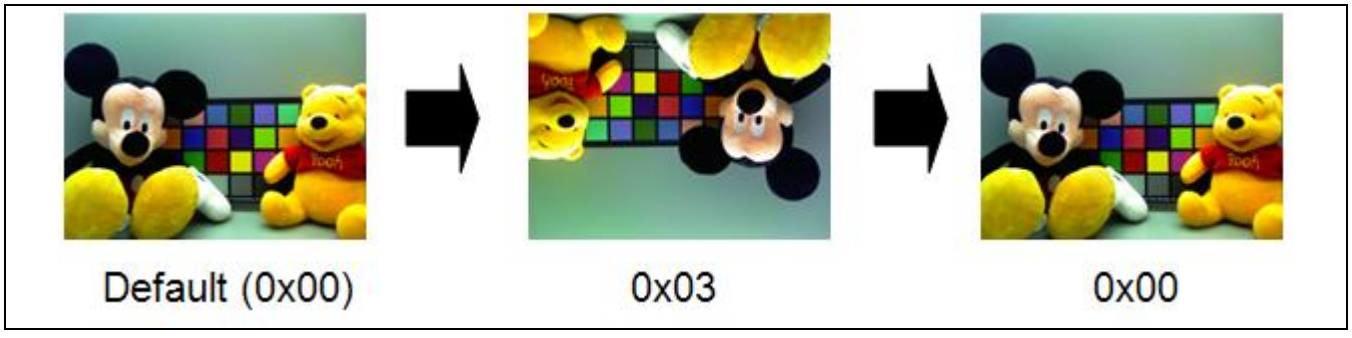

Figure 4-3 H-Mirror & V-Flip

4.2 SPECIAL EFFECTS

Table 4-5 Register for Special Effects control

Control Register	Attr	Description	Default
REG_TC_GP_SpecialEffects	R/W	Special Effect 0: No special effect 1: Monochrome 2: Negative monochrome 3: Negative Color 4: Sepia 5: Aqua 6: Reddish 7: Bluish 8: Greenish 9: Sketch	0
seti_usReddishStrength	0x20	Reddish power (0 ~ 255)	
seti_usGreenishStrength	0x20	Greenish power (0 ~ 255)	
seti_usBluishStrength	0x20	Bluish power (0 ~ 255)	
seti_sSepiaBrownStrength	0x19	Sepia power (0~255)	

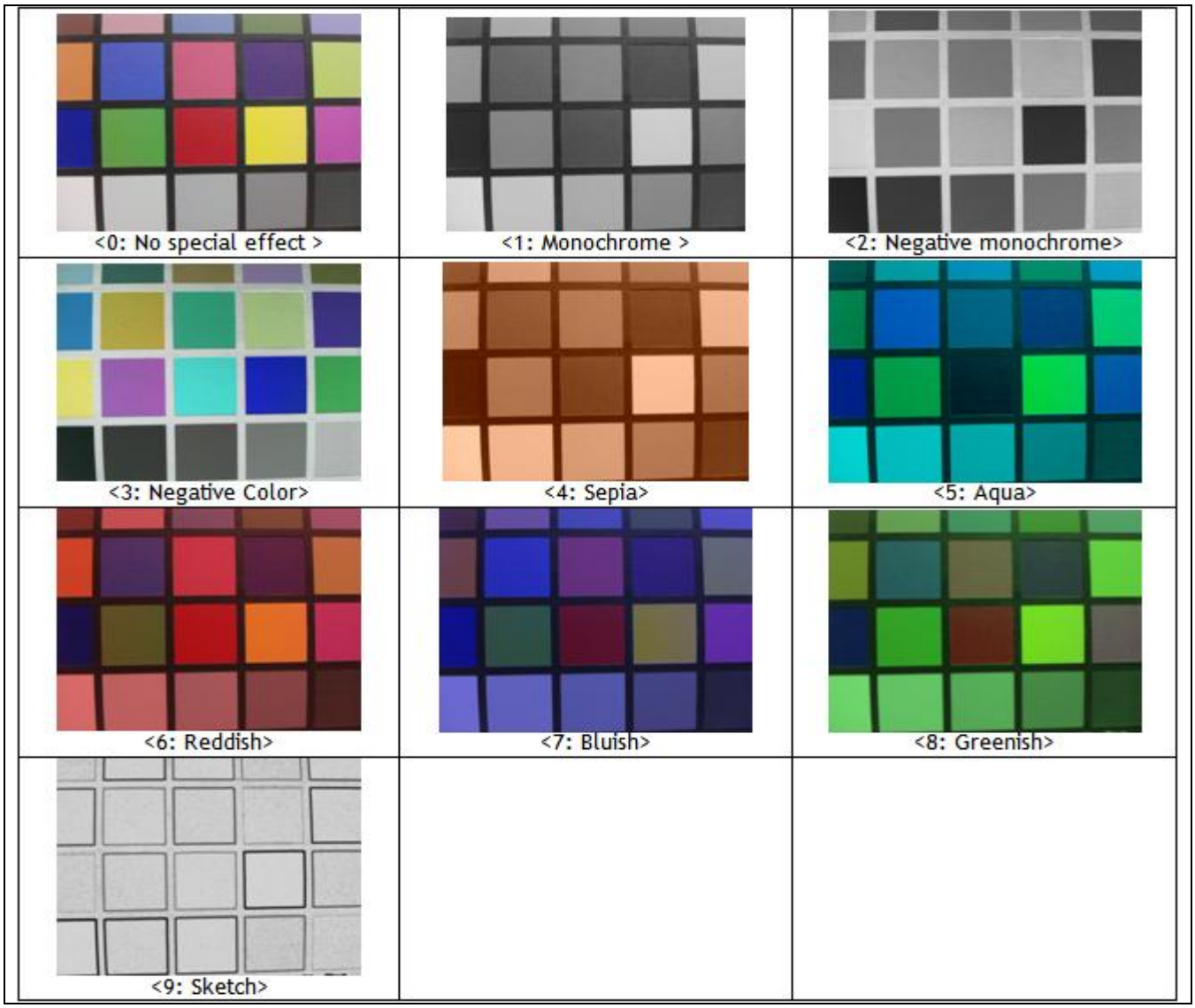

Figure 4-4 Special Effects

4.3 BRIGHTNESS

Table 4-6 Register for Brightness control

Attr	Control Register	Description	Default
R/W	REG_TC_UserBrightness	Control brightness value FF81-7Fh	0

4.4 CONTRAST

Table 4-7 Register for Contrast control

Attr	Control Register	Description	Default
R/W	REG_TC_UserContrast	Control contrast value FF81-7Fh	0

4.5 SATURATION

Table 4-8 Register for Saturation control

Attr	Control Register	Description	Default
R/W	REG_TC_UserSaturation	Control saturation value FF81-7Fh	0

4.6 SHARPNESS

Table 4-9 Register for Sharpness control

Attr	Control Register	Description	Default
R/W	REG_TC_UserSharpBlur	Control sharpness value FF81-7Fh	0