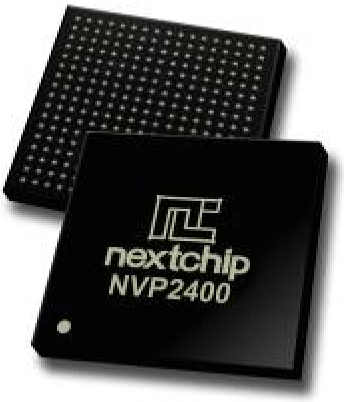
NVP2400

Data Sheet

Megapixel CMOS ISP for IP & HDcctv Camera



Information contained herein is subject to change without notice.

Make sure to check and use the latest version of the Data Sheet.

[주]넥스트칩
nextchip

www.nextchip.com

Revision History

VERSION	DATE	DESCRIPTION	NOTE
REV 0.0	2010-05-07	- Preliminary 1st version	
REV 1.0	2011-01-19	- Initial Draft	

Contact Information

Homepage: www.nextchip.com

E-mail : sales@nextchip.com

Phone : +82-2-3460-4700

Contents

Contents of Figures

Contents of Tables

1. INTRODUCTION

The NVP2400 is a cost-effective and high performance CMOS sensor ISP (Image Signal Processor) for IP network cameras, HD-SDI cameras and high-end analog CCTV applications. It includes fast 8 bits 8051 MCU to control ISP functions such as AE/AWB. It also contains SDRAM, kind of SiP for such functions as 3D-NR, Dzoom, FRC and OSD operations.

The NVP2400 can receive 12 bits parallel or sub-LVDS CMOS sensor input and provides BT.656, BT.1120, YC 16 bits and composite video as output. In particular, it can support high resolution analog CVBS output from the high sensitivity mega pixel CMOS sensor, which can cope with conventional high-end analog security camera application. It supports not only various IP camera interfaces but also HD-SDI interface standards.

The NVP2400's image signal processor including 3A (AE/AWB/AF) and enhancement functions ensures stable and high quality images, making it suitable for a security camera.

It also offers many useful security camera functions such as high resolution digital zoom, PIP, Electronic PTZ, FRC, flicker suppression, HLC/BLC, privacy zone masking, motion detection, IRIS control I/F and OSD.

1.1. FEATURES

- **Input Formats**

- 1.3M/2M pixel CMOS sensor bayer input
(Supports Up to 5M pixels @ 1920x1080p crop)

- 12bits 30fps-parallel/60fps-serial @ SXGA
(1280x1024)

- 12bits 30fps-parallel/serial @ 1920x1080p

- 4-pixel mixture (binning) mode (VGA)

- Long/Short combined (HDR) 10bits YUV input

- **Output Formats**

- BT.656 / BT.1120 / YC16 bits

- Supports HD-SDI I/F
(1920x1080 30p/1280x720 60p)

- Composite video
(Video Encoder 720H@27MHz, 960H@36MHz)

- **Fast 8 bits 8051 Embedded MCU**

- Dual DPTRs (Data Pointer Registers)
for fast block move

- 96Kbyte On-chip Code Memory

- 256byte On-chip Data SRAM (I-RAM)

- 6Kbyte On-chip Extended Data SRAM (X-RAM)

- Supports external SFR I/F for ISP block control

- **System Interface**

- 3 ports 16 bits timer including PWM

- 2 ports SPIs / UARTs

- 2 ports I2Cs (Master/Slave selectable)

- Sensor I/F (SPI or I2C)

- 2 ports external interrupt

- **Image Signal Processor**

- 3 Auto (AE/AWB/AF) control

  : Analog/digital (Sensor & ISP) gain control
  : Shutter speed control (High & Low shutter)
  : Color temperature (1800K ~ 10500K)
  : AF detection

- 2D/3D-NR

  : Temporal-IIR motion adaptive noise reduction by AGC level feedback
  : Ghost-tail suppression
  : 2D-edge preserving & flat area noise reduction

- Pre/Post-processor

  : DPC
  (Dead Pixel Correction, Live & Static)
  : BLC
  (Black Level Compensation, Digital Clamp)
  : CI (Color Interpolation, De-mosaicing)
  : LSC (Lens Shading Correction)
  : Y/C process (Saturation/HUE/Contrast/ Brightness/Y, C gamma/Edge Enhancement/ AGC Suppression/False Color Suppression
  : ACCE
  (Adaptive Color & Contrast Enhancement)
  : Programmable RGB gamma correction

- FRC (Frame Rate Control)

- Output Format Converter

- **Various Security Camera Functions**

- High resolution digital zoom (x64)

- PIP (Full Image / Zoom Image)

- Electronic Pan/Tilt/Zoom

- Day & Night function

- Vertical flip / horizontal mirror for 1.3M Pixel

- Progressive-to-Interlace conversion for composite video

- Flicker suppression

- HLC/BLC (High Light/Back Light Compensation)

- DEFOG (Visibility enhancement on the low contrast scene)

- Motion Detection (4 separable areas, Up to 1980-blocks, variable block size control)

- Privacy Zone Masking (8 flexible windows)

- 10 bits DAC for IRIS control interface (Video/DC)

- User definable OSD (SD/HD path)

- **Frame Buffer Memory**

- Internal SDRAM for 3D-NR/FRC/DZoom/OSD

- Up to 133MHz SDRAM operation

- Supports SDRAM power save & power down mode

- **Analog Video Encoder**

- 10 bits DAC with 27MHz/36MHz operation

- Generate 720H/960H CVBS for Video Standards

- NTSC (M/J), PAL (B/D/G/H/I/M/60)
Programmable gain control for sync, burst, luminance and chrominance

- **Miscellaneous**

- Programmable output sync generation

- Built-in Input/Output test pattern generation

- **Operating Voltage**

- 3.3V(I/O), 1.8V(SiP SDR)

- 1.2V(Internal)

- **Operating Temperature**

- TBD

- **Power Consumption**

- TBD

- **Package**

- 225-FBGA-1313-0.8p

- **Typical Application**

- Mega-pixel CMOS sensor ISP for IP camera and HD cctv security camera system

- High-resolution analog security camera system based on mega-pixel CMOS sensor.

1.2. BLOCK DIAGRAM

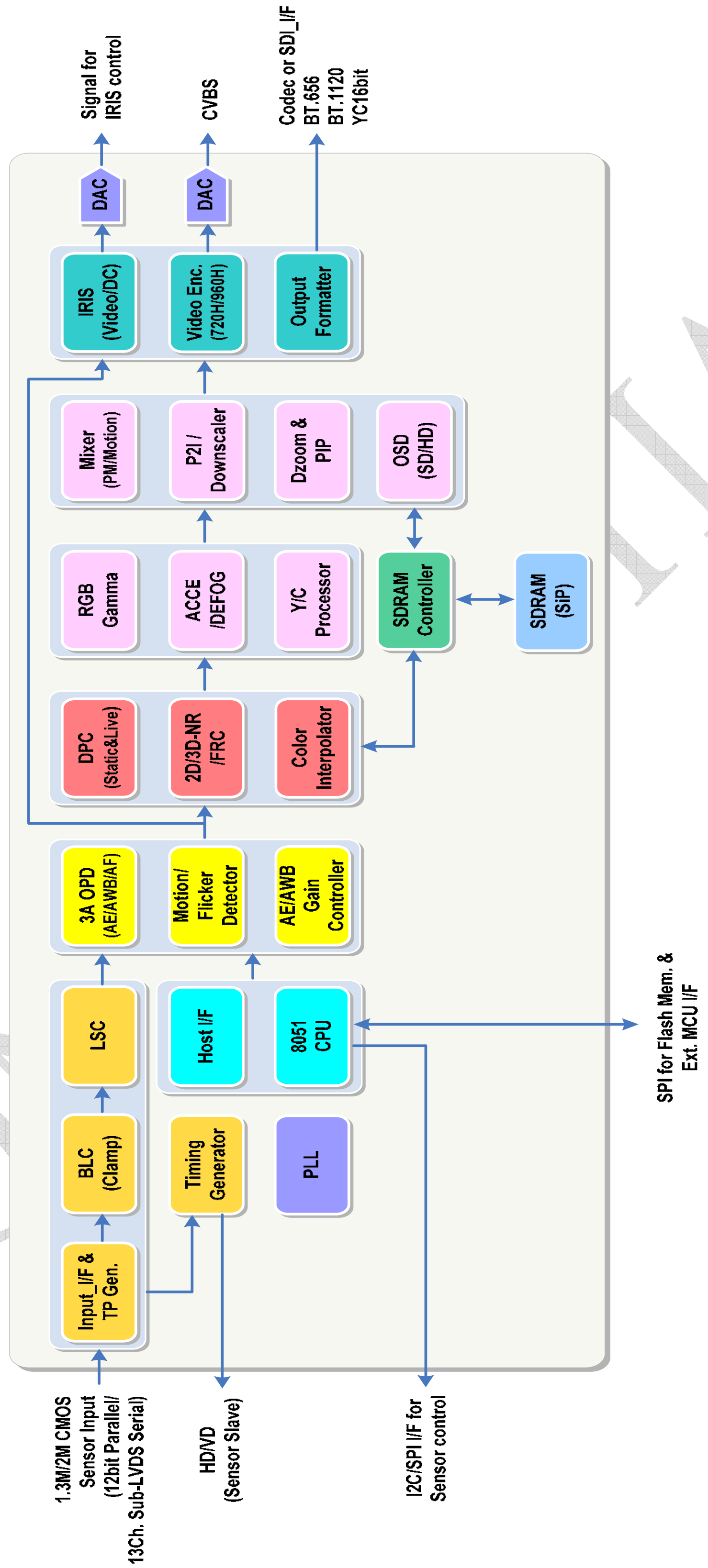
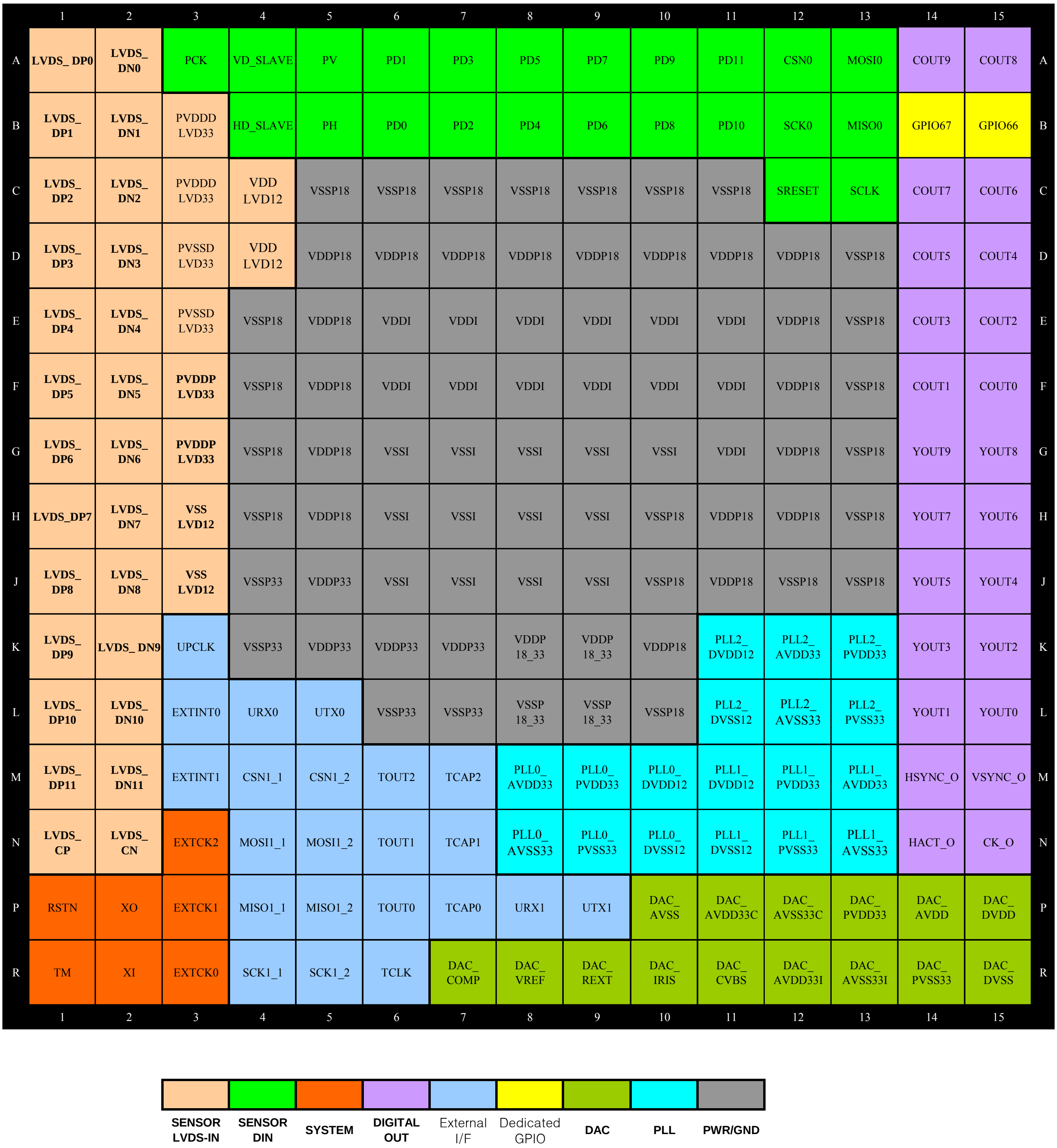


Fig. 1.1 NVP2400 Overall Block Diagram

2. BALL CONFIGURATION

2.1. BALL DIAGRAM



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
A	LVDS_ DP0	LVDS_ DN0	PCK	VD_SLAVE	PV	PD1	PD3	PD5	PD7	PD9	PD11	CSN0	MOSI0	COUT9	COUT8	A
B	LVDS_ DP1	LVDS_ DN1	PVDDD LVD33	HD_SLAVE	PH	PD0	PD2	PD4	PD6	PD8	PD10	SCK0	MISO0	GPIO67	GPIO66	B
C	LVDS_ DP2	LVDS_ DN2	PVDDD LVD33	VDD LVD12	VSSP18	VSSP18	VSSP18	VSSP18	VSSP18	VSSP18	VSSP18	SRESET	SCLK	COUT7	COUT6	C
D	LVDS_ DP3	LVDS_ DN3	PVSSD LVD33	VDD LVD12	VDDP18	VDDP18	VDDP18	VDDP18	VDDP18	VDDP18	VDDP18	VDDP18	VSSP18	COUT5	COUT4	D
E	LVDS_ DP4	LVDS_ DN4	PVSSD LVD33	VSSP18	VDDP18	VDDI	VDDI	VDDI	VDDI	VDDI	VDDI	VDDP18	VSSP18	COUT3	COUT2	E
F	LVDS_ DP5	LVDS_ DN5	PVDDP LVD33	VSSP18	VDDP18	VDDI	VDDI	VDDI	VDDI	VDDI	VDDI	VDDP18	VSSP18	COUT1	COUT0	F
G	LVDS_ DP6	LVDS_ DN6	PVDDP LVD33	VSSP18	VDDP18	VSSI	VSSI	VSSI	VSSI	VSSI	VDDI	VDDP18	VSSP18	YOUT9	YOUT8	G
H	LVDS_DP7	LVDS_ DN7	VSS LVD12	VSSP18	VDDP18	VSSI	VSSI	VSSI	VSSI	VSSP18	VDDP18	VDDP18	VSSP18	YOUT7	YOUT6	H
J	LVDS_ DP8	LVDS_ DN8	VSS LVD12	VSSP33	VDDP33	VSSI	VSSI	VSSI	VSSI	VSSP18	VDDP18	VSSP18	VSSP18	YOUT5	YOUT4	J
K	LVDS_ DP9	LVDS_ DN9	UPCLK	VSSP33	VDDP33	VDDP33	VDDP33	VDDP 18_33	VDDP 18_33	VDDP18	PLL2_ DVDD12	PLL2_ AVDD33	PLL2_ PVDD33	YOUT3	YOUT2	K
L	LVDS_ DP10	LVDS_ DN10	EXTINT0	URX0	UTX0	VSSP33	VSSP33	VSSP 18_33	VSSP 18_33	VSSP18	PLL2_ DVSS12	PLL2_ AVSS33	PLL2_ PVSS33	YOUT1	YOUT0	L
M	LVDS_ DP11	LVDS_ DN11	EXTINT1	CSN1_1	CSN1_2	TOUT2	TCAP2	PLL0_ AVDD33	PLL0_ PVDD33	PLL0_ DVDD12	PLL1_ DVDD12	PLL1_ PVDD33	PLL1_ AVDD33	HSYNC_O	VSYNC_O	M
N	LVDS_ CP	LVDS_ CN	EXTCK2	MOSI1_1	MOSI1_2	TOUT1	TCAP1	PLL0_ AVSS33	PLL0_ PVSS33	PLL0_ DVSS12	PLL1_ DVSS12	PLL1_ PVSS33	PLL1_ AVSS33	HACT_O	CK_O	N
P	RSTN	XO	EXTCK1	MISO1_1	MISO1_2	TOUT0	TCAP0	URX1	UTX1	DAC_ AVSS	DAC_ AVDD33C	DAC_ AVSS33C	DAC_ PVDD33	DAC_ AVDD	DAC_ DVDD	P
R	TM	XI	EXTCK0	SCK1_1	SCK1_2	TCLK	DAC_ COMP	DAC_ VREF	DAC_ REXT	DAC_ IRIS	DAC_ CVBS	DAC_ AVDD33I	DAC_ AVSS33I	DAC_ PVSS33	DAC_ DVSS	R
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	

SENSOR LVDS-IN	SENSOR DIN	SYSTEM	DIGITAL OUT	External I/F	Dedicated GPIO	DAC	PLL	PWR/GND

Fig. 2.1 225-FBGA-1313 Ball Map (Top View)

2.2. BALL DESCRIPTION

Table 2.1 NVP2400 Ball Description

NAME	IO	Description	BALL No.
System Service			
RSTN	I	Schmitt trigger input reset pad(Acrive Low)	P1
XI	I	Crystal oscillator input	R2
XO	O	Crystal oscillator output	P2
EXTCK0	B	External clock input 0	R3
EXTCK1	B	External clock input 1	P3
EXTCK2	B	External clock input 2	N3
TM	I	Test mode selection PAD(Normally 0)	R1
PLL			
PLL0_DVDD12	P	PLL0 VDD digital pad(1.2V)	M10
PLL0_DVSS12	P	PLL0 VSS digital pad	N10
PLL0_AVDD33	P	PLL0 VDD analog pad(3.3V)	M8
PLL0_AVSS33	P	PLL0 VSS analog pad	N8
PLL0_PVDD33	P	PLL0 PAD Power (3.3V)	M9
PLL0_PVSS33	P	PLL0 PAD Ground	N9
PLL1_DVDD12	P	PLL1 VDD digital pad(1.2V)	M11
PLL1_DVSS12	P	PLL1 VSS digital pad	N11
PLL1_AVDD33	P	PLL1 VDD analog pad(3.3V)	M13
PLL1_AVSS33	P	PLL1 VSS analog pad	N13
PLL1_PVDD33	P	PLL1 PAD Power (3.3V)	M12
PLL1_PVSS33	P	PLL1 PAD Ground	N12
PLL2_DVDD12	P	PLL2 VDD digital pad(1.2V)	K11
PLL2_DVSS12	P	PLL2 VSS digital pad	L11
PLL2_AVDD33	P	PLL2 VDD analog pad(3.3V)	K12
PLL2_AVSS33	P	PLL2 VSS analog pad	L12
PLL2_PVDD33	P	PLL2 PAD Power (3.3V)	K13
PLL2_PVSS33	P	PLL2 PAD Ground	L13
Serial (Sub-LVDS) Sensor IF			
LVDS_CP	I	Sub-LVDS input(clock +)	N1
LVDS_CN	I	Sub-LVDS input(clock -)	N2
LVDS_DP0	I	Sub-LVDS input(data0 +)	A1
LVDS_DN0	I	Sub-LVDS input(data0 -)	A2
LVDS_DP1	I	Sub-LVDS input(data1 +)	B1
LVDS_DN1	I	Sub-LVDS input(data1 -)	B2
LVDS_DP2	I	Sub-LVDS input(data2 +)	C1
LVDS_DN2	I	Sub-LVDS input(data2 -)	C2
LVDS_DP3	I	Sub-LVDS input(data3 +)	D1
LVDS_DN3	I	Sub-LVDS input(data3 -)	D2
LVDS_DP4	I	Sub-LVDS input(data4 +)	E1
LVDS_DN4	I	Sub-LVDS input(data4 -)	E2
LVDS_DP5	I	Sub-LVDS input(data5 +)	F1
LVDS_DN5	I	Sub-LVDS input(data5 -)	F2
LVDS_DP6	I	Sub-LVDS input(data6 +)	G1
LVDS_DN6	I	Sub-LVDS input(data6 -)	G2

LVDS_DP7	I	Sub-LVDS input(data7 +)	H1
LVDS_DN7	I	Sub-LVDS input(data7 -)	H2
LVDS_DP8	I	Sub-LVDS input(data8 +)	J1
LVDS_DN8	I	Sub-LVDS input(data8 -)	J2
LVDS_DP9	I	Sub-LVDS input(data9 +)	K1
LVDS_DN9	I	Sub-LVDS input(data9 -)	K2
LVDS_DP10	I	Sub-LVDS input(data10 +)	L1
LVDS_DN10	I	Sub-LVDS input(data10 -)	L2
LVDS_DP11	I	Sub-LVDS input(data11 +)	M1
LVDS_DN11	I	Sub-LVDS input(data11 -)	M2
PVDDDLVD33	P	LVDS analog power pad (3.3V)	B3, C3
PVSSDLVD33	P	LVDS analog ground pad	D3, E3
PVDDPLVD33	P	LVDS digital power pad for level shifter (3.3V)	F3, G3
VDDLVD12	P	LVDS digital power pad (1.2V)	C4, D4
VSSLVD12	P	LVDS digital ground pad (1.2V)	H3, J3
Parallel Sensor IF (1.8V)			
PD0	B	Parallel sensor input data 0	B6
PD1	B	Parallel sensor input data 1	A6
PD2	B	Parallel sensor input data 2	B7
PD3	B	Parallel sensor input data 3	A7
PD4	B	Parallel sensor input data 4	B8
PD5	B	Parallel sensor input data 5	A8
PD6	B	Parallel sensor input data 6	B9
PD7	B	Parallel sensor input data 7	A9
PD8	B	Parallel sensor input data 8	B10
PD9	B	Parallel sensor input data 9	A10
PD10	B	Parallel sensor input data 10	B11
PD11	B	Parallel sensor input data 11	A11
PH	B	Parallel sensor input Hsync	B5
PV	B	Parallel sensor input Vsync	A5
PCK	I	Parallel sensor input clock	A3
SCLK	B	Master clock output for Sensor Master Clock	C13
SRESET	B	Reset output for Sensor reset	C12
VD_SLAVE	B	Vsync out in Slave Mode	A4
HD_SLAVE	B	Hsync out in Slave Mode	B4
SCK0	B	SPI0 CLOCK or I2C0 SCL	B12
MISO0	B	SPI0 Master-In/Slave-Out or I2C0 SDA	B13
MOSI0	B	SPI0 Master-Out/Slave-In	A13
CSN0	B	SPI0 slave select	A12
External MCU Interface (3.3V)			
EXTINT0	B	External interrupt0 input	L3
EXTINT1	B	External interrupt1 input	M3
SCK1_1	B	SPI1_1 CLOCK	R4
MISO1_1	B	SPI1 first port for Master-In/Slave-Out	P4
MOSI1_1	B	SPI1 first port for Master-Out/Slave-In	N4
CSN1_1	B	SPI1 first port for slave select	M4
SCK1_2	B	SPI1 second port for CLOCK or I2C1 SCL	R5
MISO1_2	B	SPI1 second port for Master-In/Slave-Out or I2C1 SDA	P5
MOSI1_2	B	SPI1 second port for Master-Out/Slave-In	N5

CSN1_2	B	SPI1 second port for slave select	M5
UPCLK	B	UART External clock	K3
UTX0	B	UART0 Receive Data Input	L5
URX0	B	UART0 Transmit Data Output	L4
UTX1	B	UART1 Receive Data Input	P9
URX1	B	UART1 Transmit Data Output	P8
TCLK	B	Input clock for Timer 0/1/2	R6
TCAP0	B	Input capture for timer 0	P7
TOUT0	B	Output for timer 0	P6
TCAP1	B	Input capture for timer 1	N7
TOUT1	B	Output for timer 1	N6
TCAP2	B	Input capture for timer 2	M7
TOUT2	B	Output for timer 2	M6
DIGITAL OUTPUT (3.3V)			
CK_O	O	Output clock	N15
HACT_O	B	Output HACT for YC, RGB and BT.1120	N14
VSYNC_O	B	Output Vsync for YC, RGB and BT.1120	M15
HSYNC_O	B	Output Hsync for YC, RGB and BT.1120	M14
YOUT0	B	Output Y[0] data for YC / Output YC[0] for BT.656 Output Y[0] for BT.1120	L15
YOUT1	B	Output Y[1] data for YC / Output YC[1] for BT.656 Output Y[1] for BT.1120	L14
YOUT2	B	Output Y[2] data for YC / Output YC[2] for BT.656 Output Y[2] for BT.1120	K15
YOUT3	B	Output Y[3] data for YC / Output YC[3] for BT.656 Output Y[3] for BT.1120	K14
YOUT4	B	Output Y[4] data for YC / Output YC[4] for BT.656 Output Y[4] for BT.1120	J15
YOUT5	B	Output Y[5] data for YC / Output YC[5] for BT.656 Output Y[5] for BT.1120	J14
YOUT6	B	Output Y[6] data for YC / Output YC[6] for BT.656 Output Y[6] for BT.1120	H15
YOUT7	B	Output Y[7] data for YC / Output YC[7] for BT.656 Output Y[7] for BT.1120	H14
YOUT8	B	output Y[8] for BT.1120 / GPIO54	G15
YOUT9	B	output Y[9] for BT.1120 / GPIO55	G14
COUT0	B	Output C[0] data for YC / Output C[0] for BT.1120	F15
COUT1	B	Output C[1] data for YC / Output C[1] for BT.1120	F14
COUT2	B	Output C[2] data for YC / Output C[2] for BT.1120	E15
COUT3	B	Output C[3] data for YC / Output C[3] for BT.1120	E14
COUT4	B	Output C[4] data for YC / Output C[4] for BT.1120	D15
COUT5	B	Output C[5] data for YC / Output C[5] for BT.1120	D14
COUT6	B	Output C[6] data for YC / Output C[6] for BT.1120	C15
COUT7	B	Output C[7] data for YC / Output C[7] for BT.1120	C14
COUT8	B	Output C[8] for BT.1120 / GPIO port 64	A15
COUT9	B	Output C[9] for BT.1120 / GPIO port 65	A14
GPIO66	B	GPIO port 66	B15
GPIO67	B	GPIO port 67	B14
DAC			
DAC_PVDD33	P	DAC PAD Power (+3.3V)	P13
DAC_PVSS33	P	DAC PAD Ground	R14

DAC_AVDD33C	P	Analog power (+3.3V) for CVBS	P11
DAC_AVDD33I	P	Analog power (+3.3V) for IRIS	R12
DAC_AVSS33C	P	Analog ground for CVBS	P12
DAC_AVSS33I	P	Analog ground for IRIS	R13
DAC_AVDD	P	Analog power supply (+3.3)	P14
DAC_AVSS	P	Analog ground	P10
DAC_DVDD	P	Digital power supply (+1.2)	P15
DAC_DVSS	P	Digital ground	R15
DAC_CVBS	O	CVBS output	R11
DAC_IRIS	O	IRIS output	R10
DAC_REXT	B	DAC external resistor pin	R9
DAC_VREF	O	DAC Voltage Reference	R8
DAC_COMP	O	Compensation Pin	R7
POWER			
VDDI	P	Internal Digital Core Power (+1.2V)	E6, E7, E8, E9, E10, E11, F6, F7, F8, F9, F10, F11, G11
VSSI	P	Internal Digital Core Ground	G6, G7, G8, G9, G10, H6, H7, H8, H9, J6, J7, J8, J9
VDDP33	P	Digital Pad Power (+3.3V)	J5, K5, K6, K7
VSSP33	P	Digital Pad Ground	J4, K4, L6, L7
VDDP18	P	Digital Pad Power (+1.8V)	D5, D6, D7, D8, D9, D10, D11, D12, E5, E12, F5, F12, G5, G12, H5, H11, H12, J11, K10
VSSP18	P	Digital Pad Ground	C5, C6, C7, C8, C9, C10, C11, D13, E4, E13, F4, F13, G4, G13, H4, H13, J12, J13, H10, J10, L10
VDDP_18_33	P	Sensor Interface Digital Power(+1.8V or +3.3V)	K8, K9
VSSP_18_33	P	Sensor Interface Digital Ground	L8, L9

3. MCU FUNCTION DESCRIPTION

The NVP2400 includes fast 8 bits RISC 8051 Micro-Controller Unit (MCU). MCU is functionally compatible with the standard 805x micro-controller. MCU also supports dual DPTRs for fast block move and 96Kbytes on-chip code memory, 256byte on-chip data sram (I-RAM) and 6Kbyte on-chip X-RAM.

3.1. MEMORY MAP OF MICRO-CONTROLLER

There are three kinds of memory areas: 96KBytes Code memory, 256Bytes I-Ram and 6KBytes X-RAM. Fig.3.1 and Fig.3.2 show the memory configuration of the NVP2400 Micro-controller. I-RAM includes 4 banks of 8 working registers and is available as stack area. X-RAM is 6KBytes and can be used for the data memory. The total size of Code memory for the NVP2400 Micro-controller is 96KBytes. 96KBytes is made up of 3 Banks. Bank 0 is called ROOT_BANK AREA. ROOT_BANK AREA should have cstartup routine, interrupt vectors, interrupt service routines, segments holding constants, segments with initial values for initialized variables, assembler-written routines included in the runtime library, bank-switching routines and others. Bank 1 and 2 are called BANKED AREA. Normally, 8051 MCU can use only up to 64KBytes. But in BANKED code model, Bank 1 and Bank2 can be switched by BANK_SEL register to increase the code area up to 96KBytes.

BOOT ROM is used as code area only when NVP2400 is switched on. In other words, BOOT ROM is available for power-on sequence only.

3.2. BANKED VS. NON-BANKED FUNCTION CALLS

When the Banked Code Model is used, the software engineer should understand whether to allocate the C-function to ROOT_BANK AREA or BANKED AREA because function call from ROOT_BANK AREA is faster and takes up less code space than function call from BANKED AREA. In the Banked Code Model, functions declared __near_func should be used in order to allocate the functions to the ROOT_BANK AREA.

__near_func void SystemInitialization (void) ← allocated to ROOT_BANK AREA

__near_func void SystemInitialization (void) ← allocated to ROOT_BANK AREA

If a certain function is frequently called, it should be placed in ROOT_BANK AREA in order to reduce the overhead of function calls.

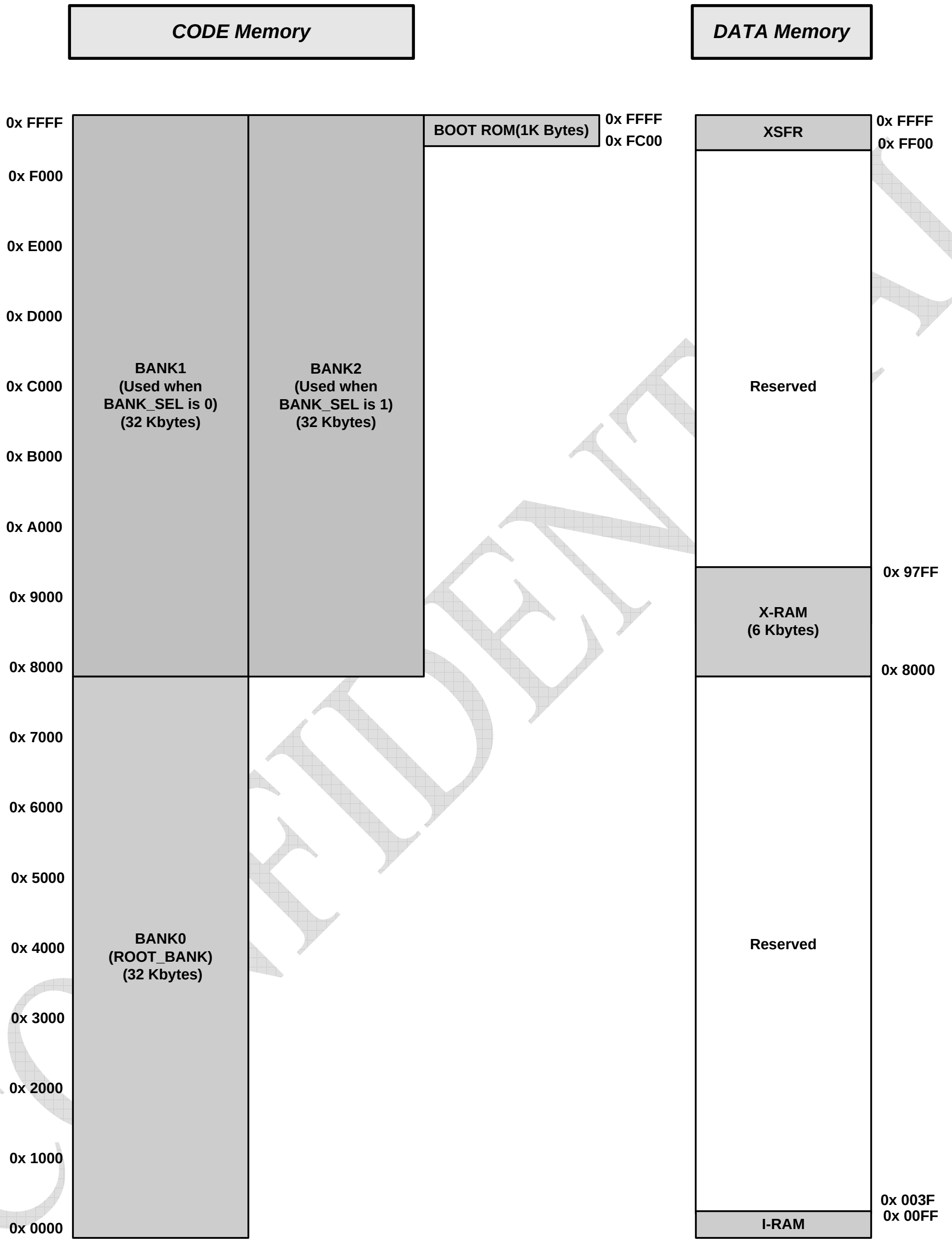
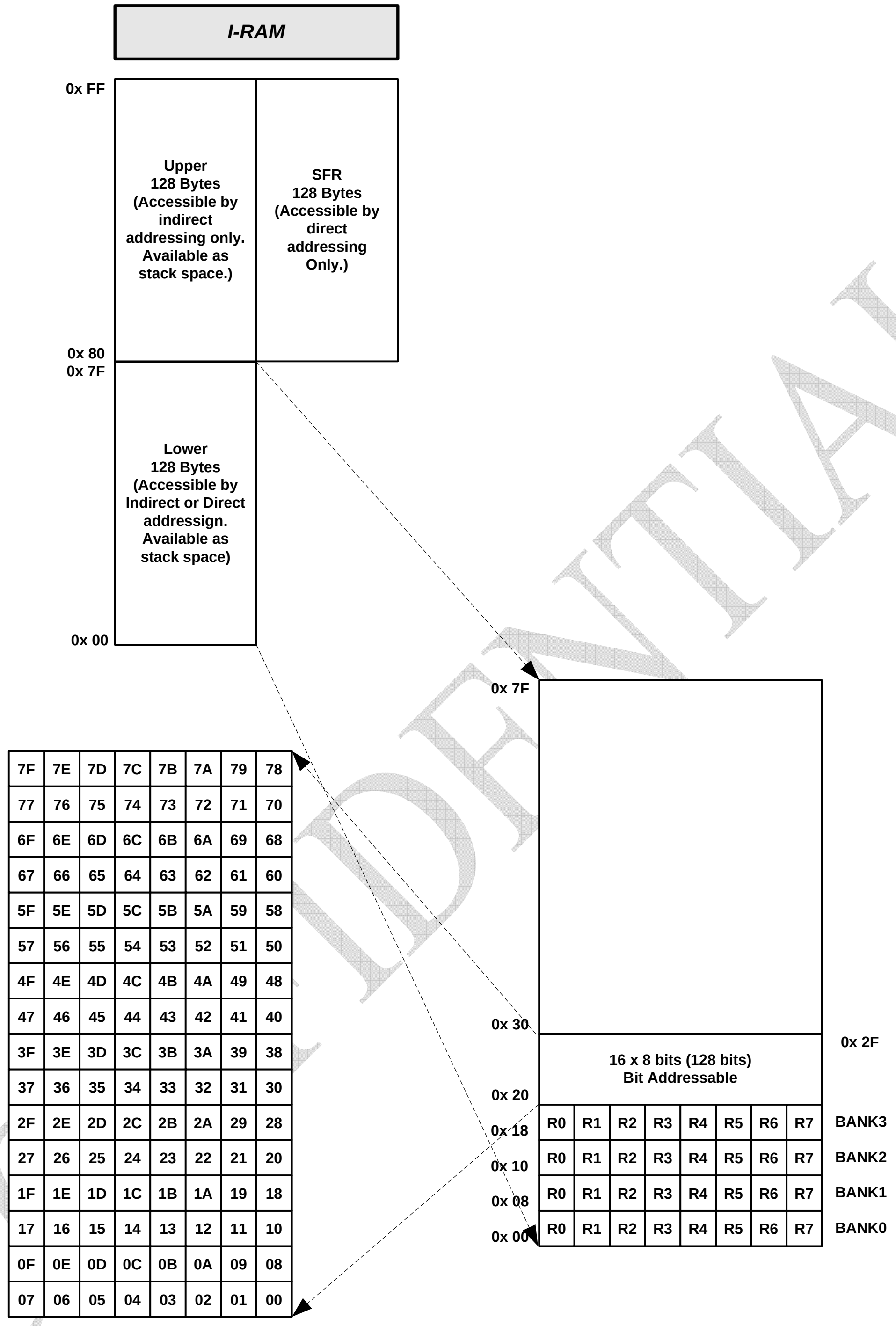


Fig. 3.1 NVP2400 Memory Map of Micro-Controller



7F	7E	7D	7C	7B	7A	79	78
77	76	75	74	73	72	71	70
6F	6E	6D	6C	6B	6A	69	68
67	66	65	64	63	62	61	60
5F	5E	5D	5C	5B	5A	59	58
57	56	55	54	53	52	51	50
4F	4E	4D	4C	4B	4A	49	48
47	46	45	44	43	42	41	40
3F	3E	3D	3C	3B	3A	39	38
37	36	35	34	33	32	31	30
2F	2E	2D	2C	2B	2A	29	28
27	26	25	24	23	22	21	20
1F	1E	1D	1C	1B	1A	19	18
17	16	15	14	13	12	11	10
0F	0E	0D	0C	0B	0A	09	08
07	06	05	04	03	02	01	00

R0	R1	R2	R3	R4	R5	R6	R7	BANK3
R0	R1	R2	R3	R4	R5	R6	R7	BANK2
R0	R1	R2	R3	R4	R5	R6	R7	BANK1
R0	R1	R2	R3	R4	R5	R6	R7	BANK0

Fig. 3.2 NVP2400 I-RAM Memory Map

3.3. SFR (SPECIAL FUNCTION REGISTER) MAP

The NVP2400 MCU employs dual data pointers to accelerate data/program memory block moves. It maintains the data pointers as DPTR0 at the SFR location of 0x82 and 0x83. It is not necessary to modify codes to use DPTR0. The NVP2400 MCU adds a second data pointer (DPTR1) at SFR locations of 0x84 and 0x85. The DPTR Select Bit in the bit 0 of DPSEL selects the active pointers. When DPSEL=0, instruction/data that use the DPTR will use DP0L and DP0H. When DPSEL=1, instruction/data that use the DPTR will use DP1L and DP1H. All DPTR related instruction/data use the currently selected data pointer. Toggle the DPSEL bit in order to switch the active pointer. Using dual data pointers ensures significantly enhanced efficiency when moving large blocks of data.

Table 3.1 NVP2400 MCU SFR MAP

MSB LSB	8	9	A	B	C	D	E	F
0	EINTMODE 0000 0000	GPMODE00 0010 0010	P2 xxxx xxxx		TCON 0000 0000	PSW 0000 0000	ACC 0000 0000	B 0000 0000
1	SP 0000 0111	GPMODE02 0010 0010	GPMODE32 0010 0010	PLL_MDIV0 0110 0011	T0MOD 0000 0000	GPMODE46 0010 0010	I2C1_CTRL 0000 0000	U0CON0 0000 0000
2	DP0L 0000 0000	GPMODE04 0010 0010	GPMODE34 0010 0010	PLL_NSDIV0 0011 1001	T0DATA0 1111 1111	GPMODE48 0010 0010	I2C1_DATA 0000 0000	U0CON1 0000 0000
3	DP0H 0000 0000	GPMODE06 0010 0010	GPMODE36 0010 0010	PLL_MDIV1 0010 0000	T0DATA1 1111 1111	GPMODE50 0010 0010	I2C1_BAUD 0000 0000	U1STAT 1110 00000
4	DP1L 0000 0000	GPMODE08 0010 0010	GPMODE38 0010 0010	PLL_NSDIV1 0011 0110	T0CNT0 0000 0000	GPMODE52 0010 0010	SPI1MOD 1000 0000	U0BAUD 0010 0000
5	DP1H 0000 0000	GPMODE10 0010 0010	GPMODE40 0010 0010	PLL_MDIV2 0100 0101	T0CNT1 0000 0000	GPMODE54 0010 0010	SPI1CK 0000 0000	U0INT 0000 0000
6	DPSEL 0000 0000	GPMODE12 0010 0010	GPMODE42 0010 0010	PLL_NSDIV2 0010 0111	T1MOD 0000 0000	GPMODE56 0010 0010	SPI1DATA0 xxxx xxxx	U0RXBUF xxxx xxxx
7		GPMODE14 0010 0010	GPMODE44 0010 0010	GIE 0000 0000	T1DATA0 1111 1111	GPMODE58 0010 0010	SPI1DATA1 xxxx xxxx	U0TXBUF xxxx xxxx
8		GPMODE16 0010 0010	IE0 0000 0000	IE1 0000 0000	T1DATA1 1111 1111	GPMODE60 0010 0010		
9	BANK_SEL 0000 0000	GPMODE18 0010 0010	IPENDSET0 0000 0000	IPENDSET1 0000 0000	T1CNT0 0000 0000	GPMODE62 0010 0010		U1CON0 0000 0000
A	XRAMB 0000 0000	GPMODE20 0010 0010	IPENDCLR0 xxxx xxxx	IPENDCLR1 xxxx xxxx	T1CNT1 0000 0000	GPMODE64 0010 0010		U1CON1 0000 0000
B		GPMODE22 0010 0010	IP0 0000 0000	IP1 0000 0000	T2MOD 0000 0000	GPMODE66 0010 0010		U1STAT 1110 00000
C	CHIP_ID 0001 0000	GPMODE24 0010 0010	IE2 0000 0000	I2C0_CTRL 0000 0000	T2DATA0 1111 1111	GPMODE68 0010 0010	SPI0MOD 1000 0000	U1BAUD 0010 0000
D	EF_CON 1000 0011	GPMODE26 0000 0010	IPENDSET2 0000 0000	I2C0_DATA 0000 0000	T2DATA1 1111 1111		SPI0CK 0000 0000	U1INT 0000 0000
E	ROM_EN 0000 0001	GPMODE28 0000 0000	IPENDCLR2 xxxx xxxx	I2C0_BAUD 0000 0000	T2CNT0 0000 0000		SPI0DATA0 xxxx xxxx	U1RXBUF xxxx xxxx
F	WDT_EN 0000_0000	GPMODE30 0010 0000	IP2 0000 0000	XSFR_BANK 0000 0000	T2CNT1 0000 0000	GPMODE74 0010 0010	SPI0DATA1 xxxx xxxx	U1TXBUF xxxx xxxx

3.4. EXTENDED SPECIAL FUNCTION REGISTER (XSFR)

The NVP2400 uses XSFR registers as ISP control registers. The software engineer should set BANK_SEL SFR before accessing XSFR registers because 256 XSFR is not enough for ISP control registers. Therefore NVP2400 uses BANKED register scheme. When bank is changed, XSFR REGISTER MAP is also changed.

3.5. SYSTEM FUNC. REGISTER & INTERRUPT STRUCTURE

The NVP2400 instructions are binary code compatible and perform the same functions as they do with the standard 8051 MCU. The effects of these instructions on bits, flags, and other status functions are identical to those of the standard 8051 MCU. However, the timing of the instructions is different, in terms of the number of clocks per instruction cycle and timing within an instruction cycle.

The NVP2400 interrupt controller has a total of 23 interrupt sources. Each Interrupt request can be generated by internal function blocks or external interrupt pins.

3.6. SPI

3.6.1. FEATURES

- Full-duplex, 3-wire or 4-wire Synchronous Data Transfer
- Master or Slave operation
- Baud rate clock selectable in Master Mode
- MSB First or LSB First Data Transfer
- Support 1 Byte / 2 Byte operation
- Support Burst slave operation for register read/write operation
-

3.6.2. BLOCK DIAGRAM

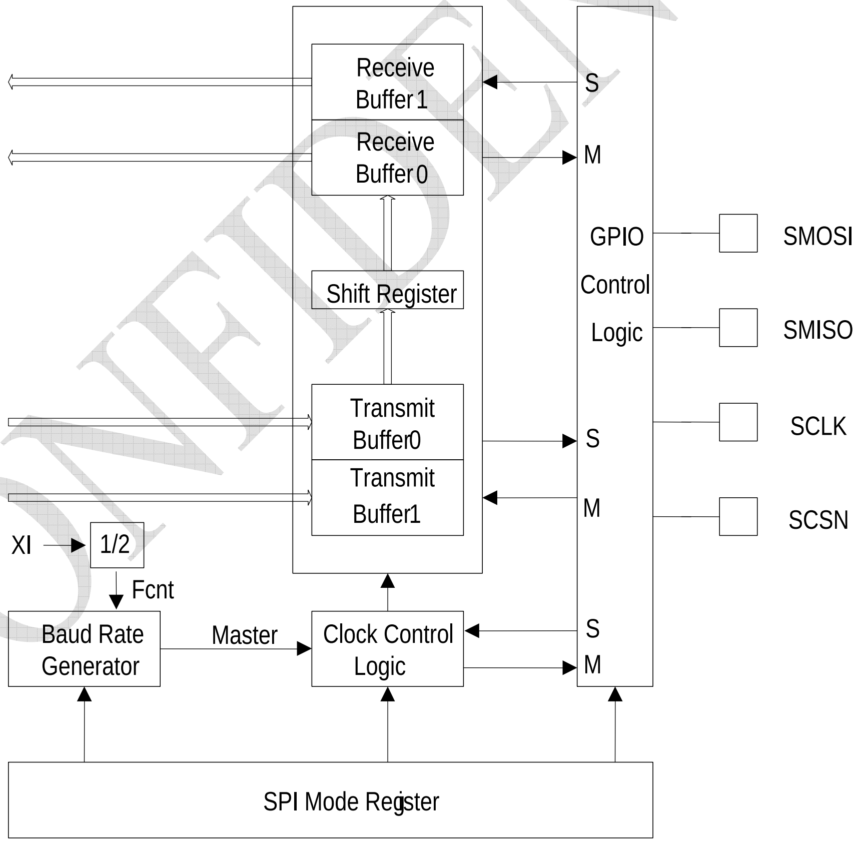


Fig. 3.3 SPI Block Diagram

3.7. UART

3.7.1. FEATURES

- Programmable baud rate
- Support external UART source clock
- Infra-Red (IR) transmit/receive
- Insert one or two stop bits per frame
- Selectable 5-bit, 6-bit, 7-bit or 8-bit data transfers
- Parity checking
- LSB first
-

3.7.2. BLOCK DIAGRAM

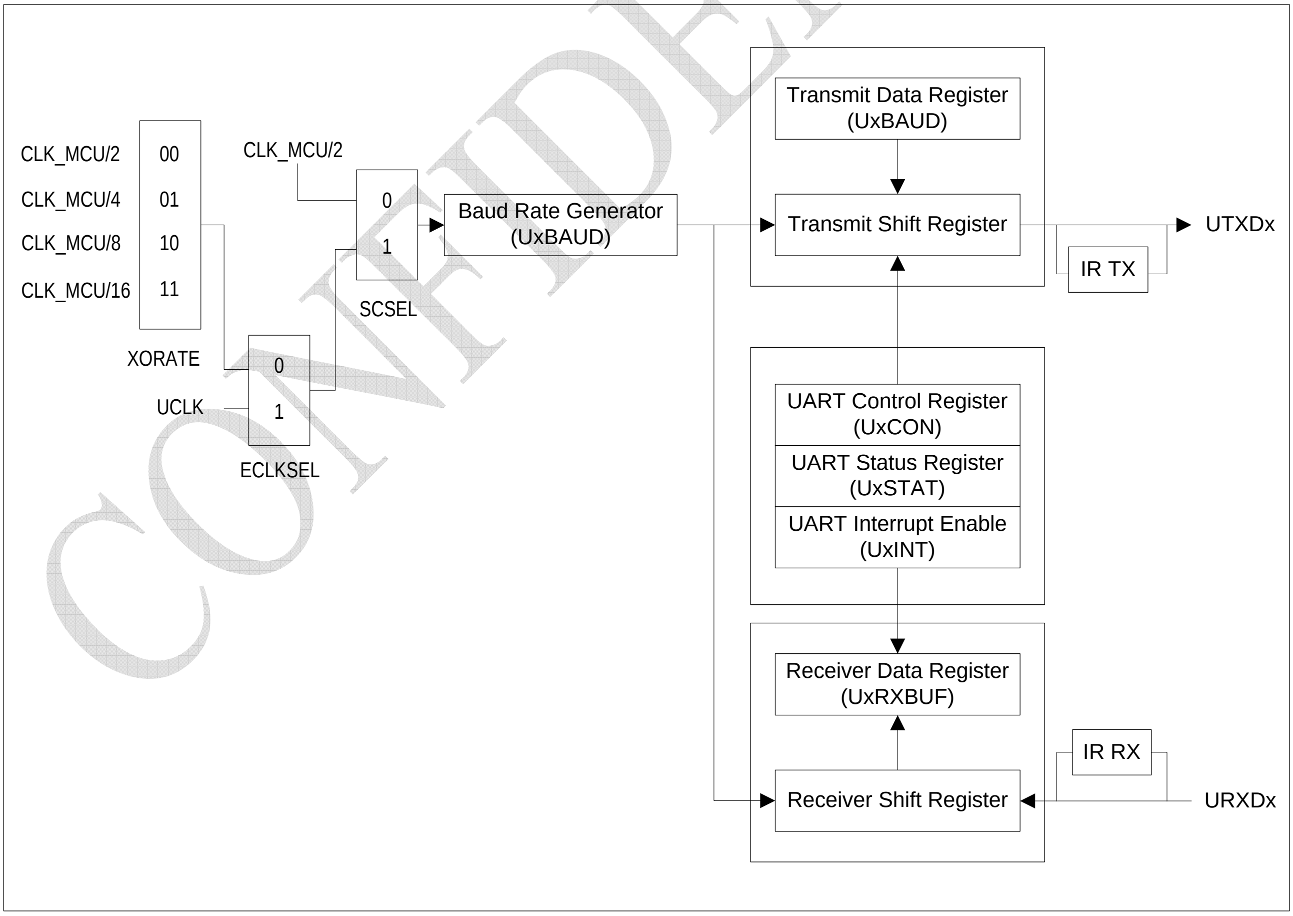

CLK_MCU/2
CLK_MCU/4
CLK_MCU/8
CLK_MCU/16
00
01
10
11
XORATE
UCLK
0
1
ECLKSEL
CLK_MCU/2
0
1
SCSEL
Baud Rate Generator (UxBAUD)
Transmit Data Register (UxBAUD)
Transmit Shift Register
IR TX
UTXDx
UART Control Register (UxCON)
UART Status Register (UxSTAT)
UART Interrupt Enable (UxINT)
Receiver Data Register (UxRXBUF)
Receiver Shift Register
IR RX
URXDx

Fig. 3.4 UART Block Diagram

3.8. TIMER

3.8.1. FEATURESS

- 3 Programmable Timers
- Interval Mode, Capture Mode or PWM Mode
- 3 PWM output and Timer output
- Watchdog Mode

1. Interval Mode

In Interval Mode, a timer continues to toggle as long as the timer reference data is same with the internal timer counter value. An interrupt request is generated whenever the level of the timer output signal is inverted (in other words, when the level toggles).

2. Capture Mode

In Capture Mode, the interrupt is generated whenever TnCAP ball is issued. Capture interrupt enable bit does not exist and TnOUT does not come out.

3. PWM Mode

Interrupt is generated by the timer matching interrupt enable and the overflow interrupt enable register. TnOUT goes high at the start of the counter but TnOUT turns low when a timer reference data is the same with an internal timer counter value. When the counter overflow occurs, the counter is reset to 0.

4. Watchdog Mode

In Watchdog Mode, Timer 2 supports the PWM mode. For the watchdog mode, interrupt enable and overflow interrupt enable register should be set to 1. When an overflow interrupt of watchdog timer is generated, reset is issued. In order to prevent this from happening, the timer counter value for Timer 2 should be initialized by firmware when a matching interrupt occurs.

3.8.2. BLOCK DIAGRAM

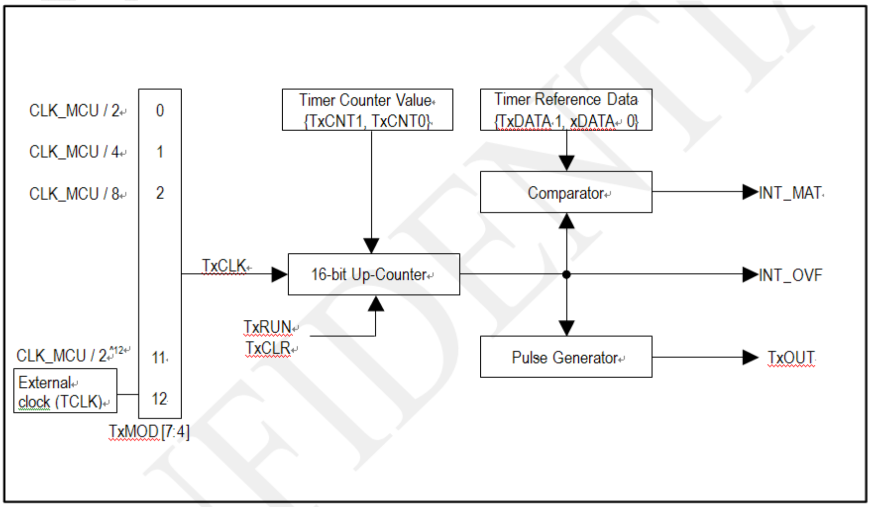


Fig. 3.5 Timers Block Diagram

3.9. I2C

3.9.1. FEATURES

- Support I2C Master / Slave mode
- Support Flexible Slave Device ID

3.9.2. BLOCK DIAGRAM

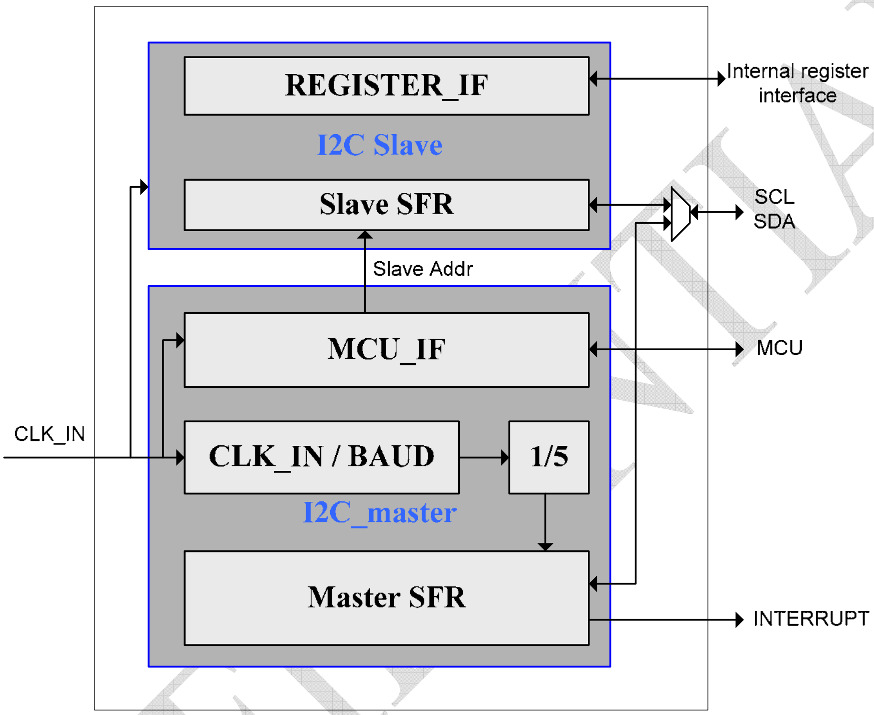


Fig. 3.6 I2C Block Diagram

3.9.3. I2C PROTOCOL

I2C block consists of two parts: Master and Slave. Master block is used to control I2C slave device such as sensor based on setting command from MCU and Slave block is used to set an internal register by receiving a command from the external I2C master such as external MCU or used as communication channel with the internal MCU.

3.10. EXTERNAL SFR INTERFACE

NVP2400 MCU uses an External SFR to set ISP register. Set an External SFR in the following order.

- XSFR_BANK Setting

- EXTERNAL address Setting

- **XSFR_BANK**

Table 3.2 XSFR_BANK Register

Bits	Reset Value	Type	Description
[7:0]	0	R/W	External SFR Bank selection register

4. ISP FUNCTION DESCRIPTION

4.1. INPUT INTERFACE & TP GENERATOR

Input Interface & TP(Test Pattern) Generator block converts serial or parallel data from sensor into the ones in appropriate data format and clock frequency that meet ISP and creates various internal test patterns.

4.1.1. BLOCK DIAGRAM

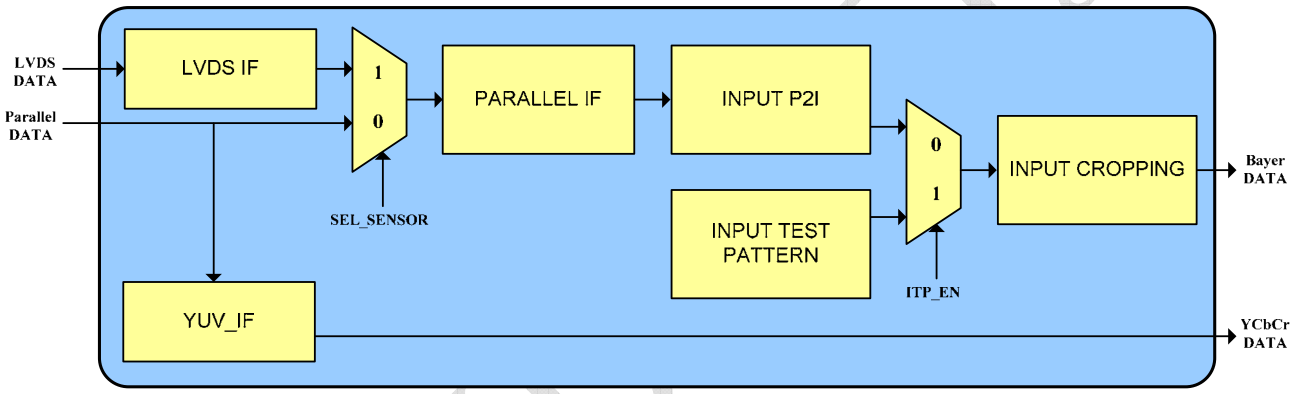


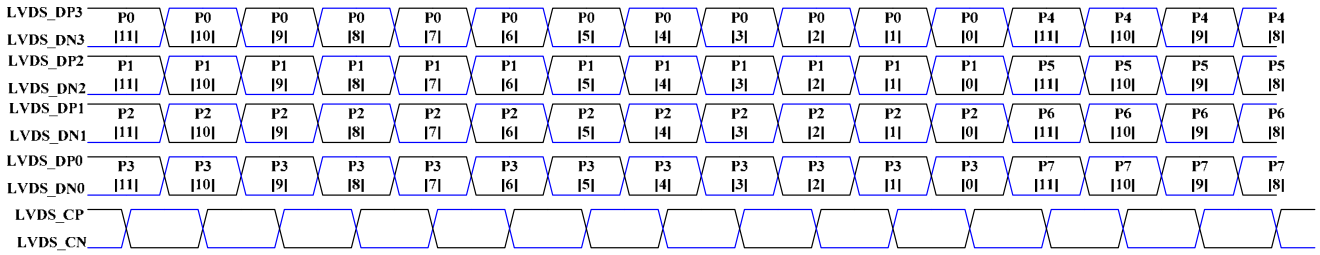
Fig. 4.1 Input Interface & TP Generator

4.1.2. SERIAL(LVDS) I/F

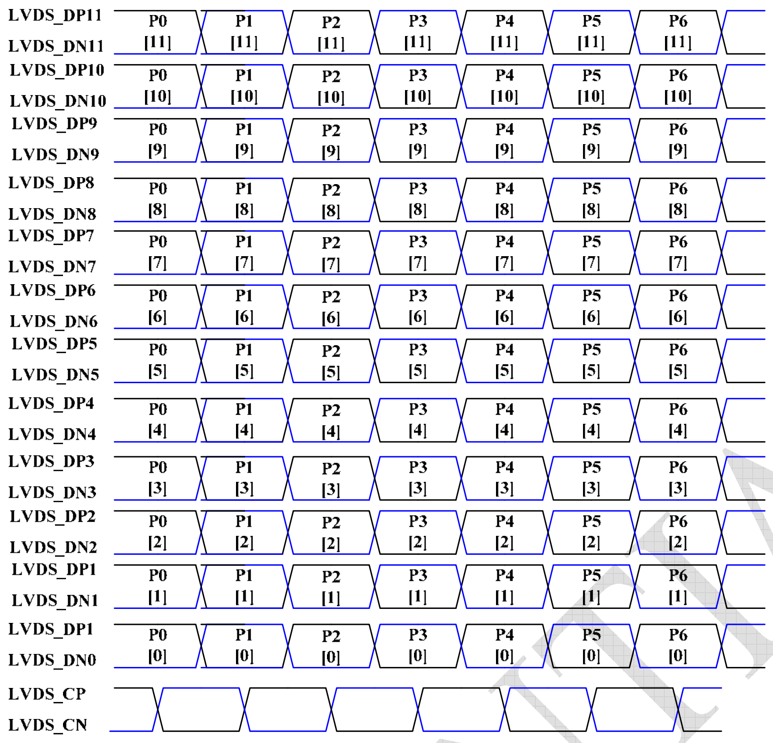
4.1.2.1. OVERVIEW

NVP2400 receives LVDS(Low Voltage Differential Signals) data output from sensor as basic serial input and provides H/V sync and bayer data by decoding LVDS. The LVDS I/F block of NVP2400 provides flexible LVDS I/F and supports up to 12 channels.

Fig. 4.2 shows examples of LVDS interface supported in NVP2400. As shown in Fig. 4.2 (a), when receiving 4 channel LVDS input, 12 bits pixel data can be formed by capturing bits from each channel at rising and falling edges of LVDS_CP/N. Another example is shown in Fig. 4.2 (b) which receives 12 channel LVDS input.


LVDS_DP3
LVDS_DN3
P0 [11] P0 [10] P0 [9] P0 [8] P0 [7] P0 [6] P0 [5] P0 [4] P0 [3] P0 [2] P0 [1] P0 [0] P4 [11] P4 [10] P4 [9] P4 [8]
LVDS_DP2
LVDS_DN2
P1 [11] P1 [10] P1 [9] P1 [8] P1 [7] P1 [6] P1 [5] P1 [4] P1 [3] P1 [2] P1 [1] P1 [0] P5 [11] P5 [10] P5 [9] P5 [8]
LVDS_DP1
LVDS_DN1
P2 [11] P2 [10] P2 [9] P2 [8] P2 [7] P2 [6] P2 [5] P2 [4] P2 [3] P2 [2] P2 [1] P2 [0] P6 [11] P6 [10] P6 [9] P6 [8]
LVDS_DP0
LVDS_DN0
P3 [11] P3 [10] P3 [9] P3 [8] P3 [7] P3 [6] P3 [5] P3 [4] P3 [3] P3 [2] P3 [1] P3 [0] P7 [11] P7 [10] P7 [9] P7 [8]
LVDS_CP
LVDS_CN

(a) The example of 4 channel LVDS I/F


LVDS_DP11
LVDS_DN11
P0 [11] P1 [11] P2 [11] P3 [11] P4 [11] P5 [11] P6 [11]
LVDS_DP10
LVDS_DN10
P0 [10] P1 [10] P2 [10] P3 [10] P4 [10] P5 [10] P6 [10]
LVDS_DP9
LVDS_DN9
P0 [9] P1 [9] P2 [9] P3 [9] P4 [9] P5 [9] P6 [9]
LVDS_DP8
LVDS_DN8
P0 [8] P1 [8] P2 [8] P3 [8] P4 [8] P5 [8] P6 [8]
LVDS_DP7
LVDS_DN7
P0 [7] P1 [7] P2 [7] P3 [7] P4 [7] P5 [7] P6 [7]
LVDS_DP6
LVDS_DN6
P0 [6] P1 [6] P2 [6] P3 [6] P4 [6] P5 [6] P6 [6]
LVDS_DP5
LVDS_DN5
P0 [5] P1 [5] P2 [5] P3 [5] P4 [5] P5 [5] P6 [5]
LVDS_DP4
LVDS_DN4
P0 [4] P1 [4] P2 [4] P3 [4] P4 [4] P5 [4] P6 [4]
LVDS_DP3
LVDS_DN3
P0 [3] P1 [3] P2 [3] P3 [3] P4 [3] P5 [3] P6 [3]
LVDS_DP2
LVDS_DN2
P0 [2] P1 [2] P2 [2] P3 [2] P4 [2] P5 [2] P6 [2]
LVDS_DP1
LVDS_DN1
P0 [1] P1 [1] P2 [1] P3 [1] P4 [1] P5 [1] P6 [1]
LVDS_DP1
LVDS_DN0
P0 [0] P1 [0] P2 [0] P3 [0] P4 [0] P5 [0] P6 [0]
LVDS_CP
LVDS_CN

(b) The example of 12 channel LVDS I/F

Fig. 4.2 Example of LVDS I/F

4.1.2.2. LVDS DECODING

For decoding input serial data at LVDS I/F, read direction (row or column), number of LVDS data ports, length of LVDS data clock, etc. can be determined by an internal register setting.

The LVDS output data of sensor includes H/V sync and data information and conversion into actual H/V sync is made by H/V sync code specified for each sensor. For example, the sync code for each sensor is listed in the following table.

Table 4.1 Sensor_TYPE_A @1280x1024, Progressive Mode Sync Code

Port	Name	Code(8 bytes)
Port1	Start of Frame	FFFFh 0000h 0000h 0002h
	Start of Line	FFFFh 0000h 0000h 0000h
	End of Line	FFFFh 0000h 0000h 0001h
	End of Frame	FFFFh 0000h 0000h 0003h
Port2	Start of Frame	FFFFh 0000h 0000h 0006h
	Start of Line	FFFFh 0000h 0000h 0004h
	End of Line	FFFFh 0000h 0000h 0005h
	End of Frame	FFFFh 0000h 0000h 0007h
Port3	Start of Frame	FFFFh 0000h 0000h 0012h
	Start of Line	FFFFh 0000h 0000h 0010h
	End of Line	FFFFh 0000h 0000h 0011h
	End of Frame	FFFFh 0000h 0000h 0013h
Port4	Start of Frame	FFFFh 0000h 0000h 0016h
	Start of Line	FFFFh 0000h 0000h 0014h
	End of Line	FFFFh 0000h 0000h 0015h
	End of Frame	FFFFh 0000h 0000h 0017h

Table 4.2 Sensor_TYPE_B @1920x1080, Progressive Mode Sync Code

Port	Name	Code(12bit x 4)
Port1	Start of Frame	FFFh 000h 000h 002h
	Start of Line	FFFh 000h 000h 000h
	End of Line	FFFh 000h 000h 001h
	End of Frame	FFFh 000h 000h 003h
Port2	Start of Frame	FFFh 000h 000h 006h
	Start of Line	FFFh 000h 000h 004h
	End of Line	FFFh 000h 000h 005h
	End of Frame	FFFh 000h 000h 007h
Port3	Start of Frame	FFFh 000h 000h 012h
	Start of Line	FFFh 000h 000h 010h
	End of Line	FFFh 000h 000h 011h
	End of Frame	FFFh 000h 000h 013h
Port4	Start of Frame	FFFFh 000h 000h 016h
	Start of Line	FFFFh 000h 000h 014h
	End of Line	FFFFh 000h 000h 015h
	End of Frame	FFFFh 000h 000h 017h

Table 4.3 Sensor_TYPE_C @1920x1080, Progressive Mode Sync Code

Sync code	1st code		2nd code		3rd code		4th code	
	10 bits	12 bits	10 bits	12 bits	10 bits	12 bits	10 bits	12 bits
SAV(valid lines)	3FFh	FFFh	000h	000h	000h	000h	200h	800h
EAV(valid lines)	3FFh	FFFh	000h	000h	000h	000h	274h	9D0h
SAV(invalid lines)	3FFh	FFFh	000h	000h	000h	000h	2Ach	AB0h
EAV(invalid lines)	3FFh	FFFh	000h	000h	000h	000h	2D8h	B60h

If the sync code cannot be made by one clock unit for Sensor_TYPE_A, the sync code can be made by using two clock units with register setting, and the bit data swap function is provided in order to change the output order for decoding LVDS data.

4.1.3. PARALLEL I/F

4.1.3.1. OVERVIEW

Parallel I/F block sets active area and optical black area by receiving Hsync(PH), Vsync(PV) and parallel data(PD0 ~ PD11) as input from sensor or receiving decoded data from LVDS I/F as shown in Fig. 4.3. It also changes the order of data bits.

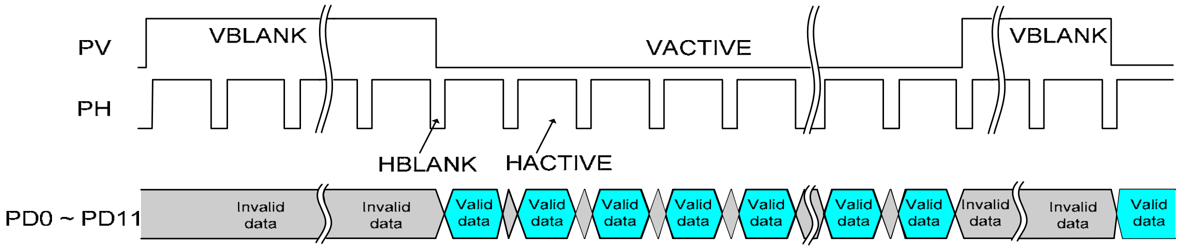


Fig. 4.3 Sensor Parallel Interface

4.1.3.2. PARALLEL I/F PROCESSING

Parallel I/F block uses Vsync and Hsync, which are received as direct input from sensor or decoded input from LVDS I/F in Fig. 4.1, in order to create hact, OB1, and OB2. Here, hact refers to the effective pixel area or active pixel area out of input sync while obact means the optical black area

Parallel I/F receives Hsync and Vsync and can set the vertical active area, horizontal active area, OB1 area and OB2 area respectively in the sensor format as shown in Fig. 4.4. OB1 and OB2 areas are used in the BLC(Black Level Compensation) block.

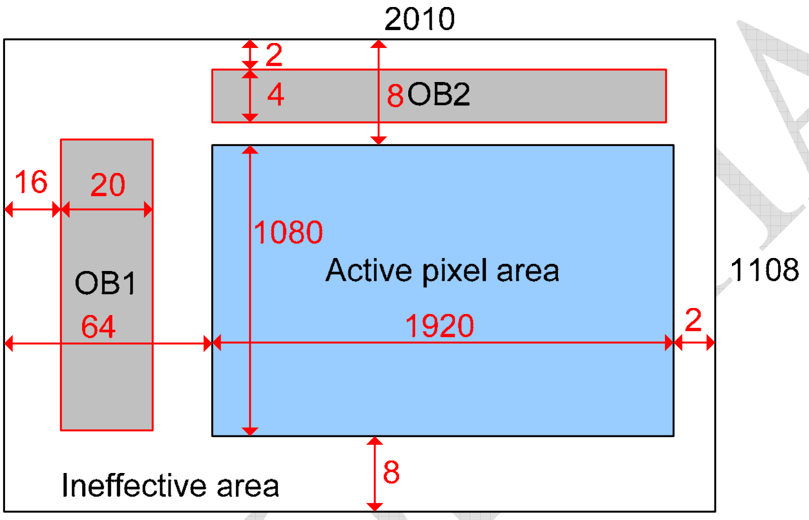


Fig. 4.4 Example of Active/OB1/OB2 Area

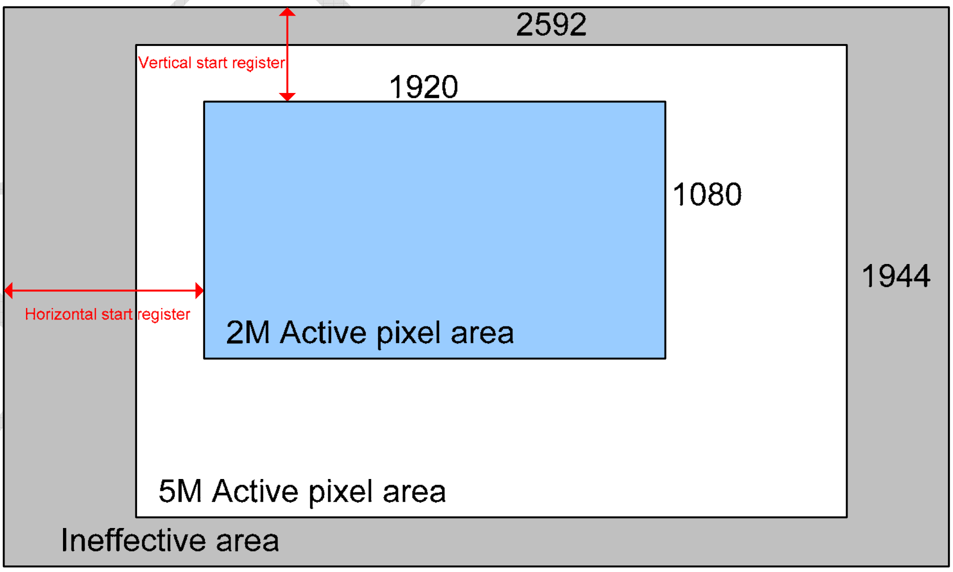


Fig. 4.5 E-PTZ of Parallel Interface

Fig 4.5 shows an example of E(Electronic)-PTZ concept, which is created by receiving images from the 5M Pixel Sensor. A certain area (2M Active pixel area) to be displayed within 5M Active pixel area can be set with the register setting. By changing the register setting, PTZ moving can be done.

4.1.4. INPUT P2I

4.1.4.1. OVERVIEW

INPUT P2I (Progressive to Interlace) block performs major two jobs.

First, it converts the video in the progressive scan format into the one in the interlaced scan format. This can be used to 1920x1080i output, which is for HD-SDI I/F. Second, it synchronizes the bayer data of sensor clock domain(CLK_SEN) to fit the ISP clock domain. This pixel clock domain synchronization allows the all the ISP processing after INPUT P2I processing stage to operate as the single ISP clock (CLK_ISP) domain.

4.1.4.2. DESCRIPTION

One of the functions of INPUT P2I block is to create the bayer pattern video data in the interlaced format using the progressive bayer pattern input video data. As shown in Fig. 4.6, the INPUT P2I block creates the RGB values for each field using the input bayer video data. It generates interlaced type video data based on the field information at the time of output.

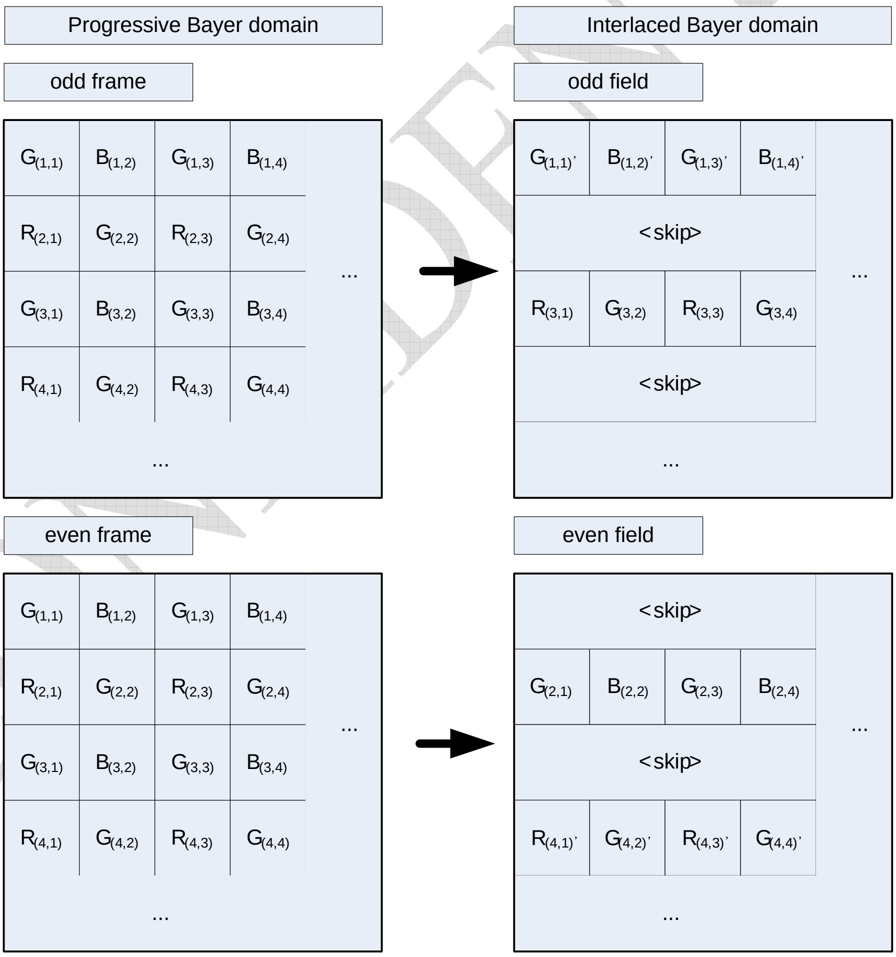

Progressive Bayer domain
Interlaced Bayer domain
odd frame
odd field
G(1,1) B(1,2) G(1,3) B(1,4)
R(2,1) G(2,2) R(2,3) G(2,4)
G(3,1) B(3,2) G(3,3) B(3,4)
R(4,1) G(4,2) R(4,3) G(4,4)
...
G(1,1)' B(1,2)' G(1,3)' B(1,4)'
<skip>
R(3,1) G(3,2) R(3,3) G(3,4)
<skip>
...
even frame
even field
G(1,1) B(1,2) G(1,3) B(1,4)
R(2,1) G(2,2) R(2,3) G(2,4)
G(3,1) B(3,2) G(3,3) B(3,4)
R(4,1) G(4,2) R(4,3) G(4,4)
...
<skip>
G(2,1) B(2,2) G(2,3) B(2,4)
<skip>
R(4,1)' G(4,2)' R(4,3)' G(4,4)'
...

Fig. 4.6 INPUT P2I Processing

4.1.5. INPUT TP GENERATOR

4.1.5.1. OVERVIEW

Input TP(Test Pattern) Generator creates 18 basic bayer patterns such as color bar, ramp, latch and others in order to detect the mal-operation of the sensor input and confirm normal operation of ISP functions except for those functions that are directly related with the sensor such as BLC(Black Level Compensation), LSC(Lens Shading Correction), etc. Test pattern generator can use the input control signal (Sync) as it is received from the sensor or it can create and use the control signal on the sync generation block inside the test pattern generator.

4.1.5.2. DESCRIPTION

For the sync generation block, interlace/progressive mode should be set in line with the signals for output. In addition, the sync does not need to be created but the data can be created for the input sync from the sensor. In case of the progressive mode, field signal is not required so it is always fixed low. The following diagram shows the controllable registers which are related to Hsync and Vsync control.

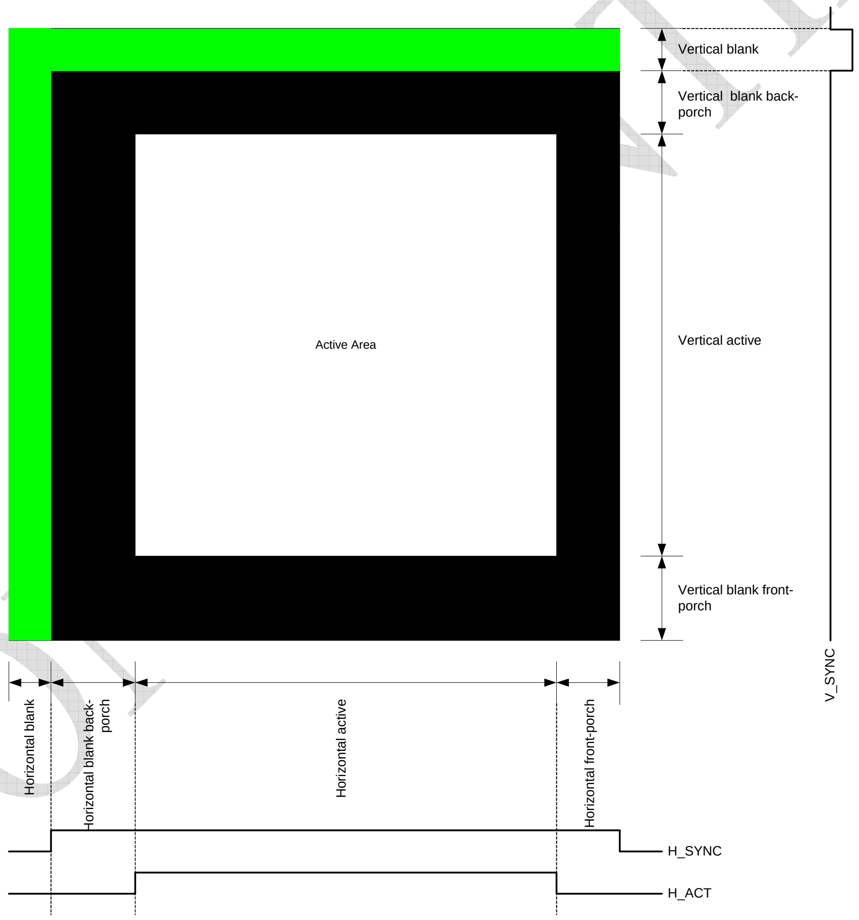


Fig. 4.7 Progressive Sync Generation

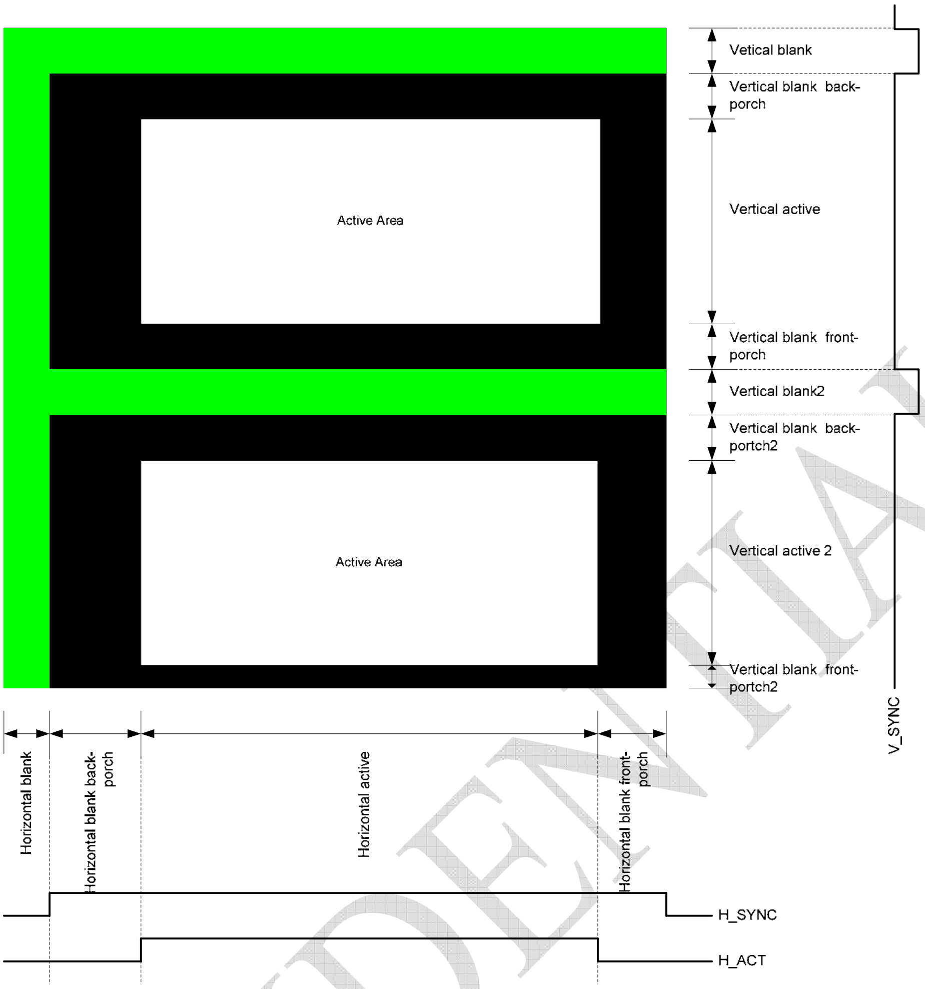
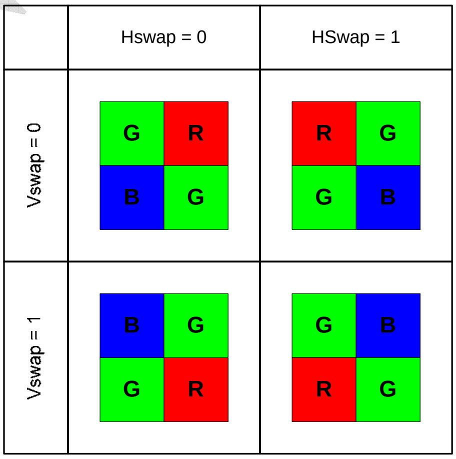

Active Area
Active Area
Vetical blank
Vertical blank back-porch
Vertical active
Vertical blank front-porch
Vertical blank2
Vertical blank back-portch2
Vertical active 2
Vertical blank front-portch2
V_SYNC
Horizontal blank
Horizontal blank back-porch
Horizontal active
Horizontal blank front-porch
H_SYNC
H_ACT

Fig. 4.8 Interlace Sync Generation

As shown in Fig. 4.9, the Input Test Pattern Generator provides output bayer's Horizontal(H)/Vertical(V) swap function. As seen in Fig. 4.10, there are basically 18 offered patterns and User Defined Color patterns can be output.



	Hswap = 0	HSwap = 1
Vswap = 0	G R B G	R G G B
Vswap = 1	B G G R	G B R G

Fig. 4.9 Swap Mode

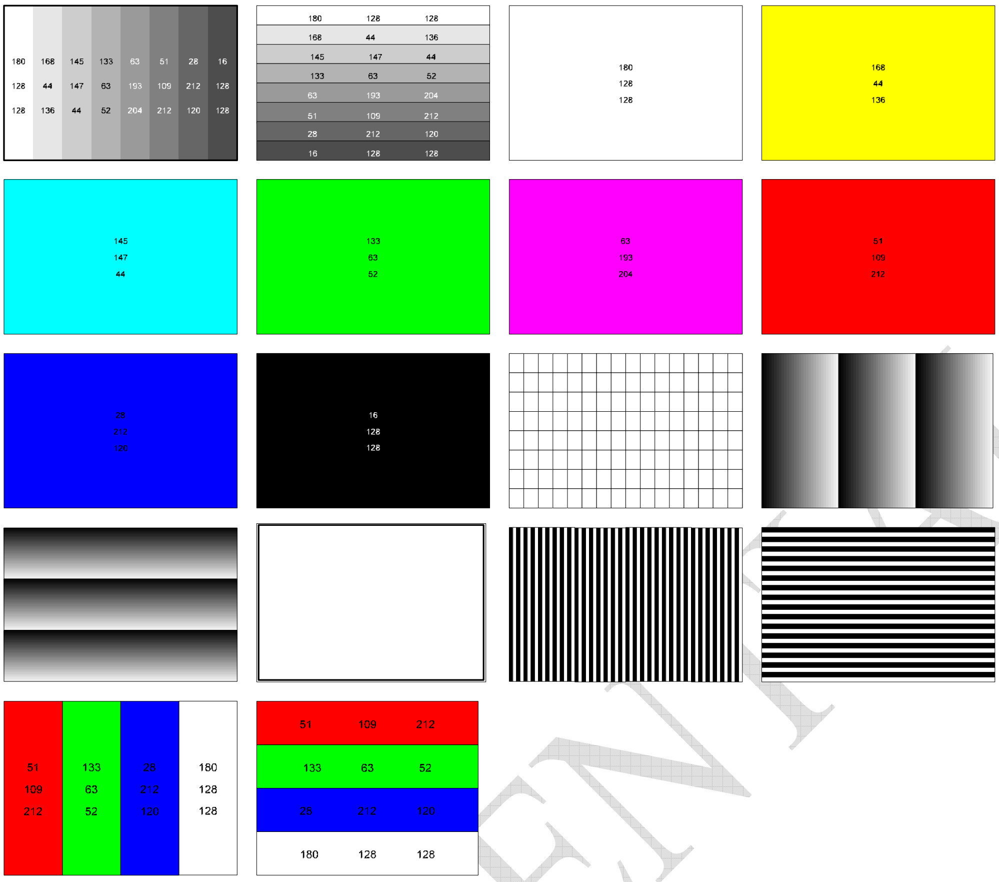


Fig. 4.10 Input Test Patterns

4.1.6. YUV I/F & INPUT CROPPING

4.1.6.1. OVERVIEW

INPUT CROPPING block supports cropping so that the image can be cropped in the desirable size using the original input from the sensor that has various resolutions (up to 1920 pixel in terms of the horizontal resolution).

Meanwhile, NVP2400 supports not only bayer format but also YUV format as an input. This is to support special part of the HDR(High Dynamic Range) CMOS sensor which provides YUV output. In such a case, bayer domain functions such as INPUT I/F, BLC, LSC, 3A OPD, Motion/Flicker Detector, DPC, 2D/3D-NR/FRC and Color Interpolation available in NVP2400 cannot be used.

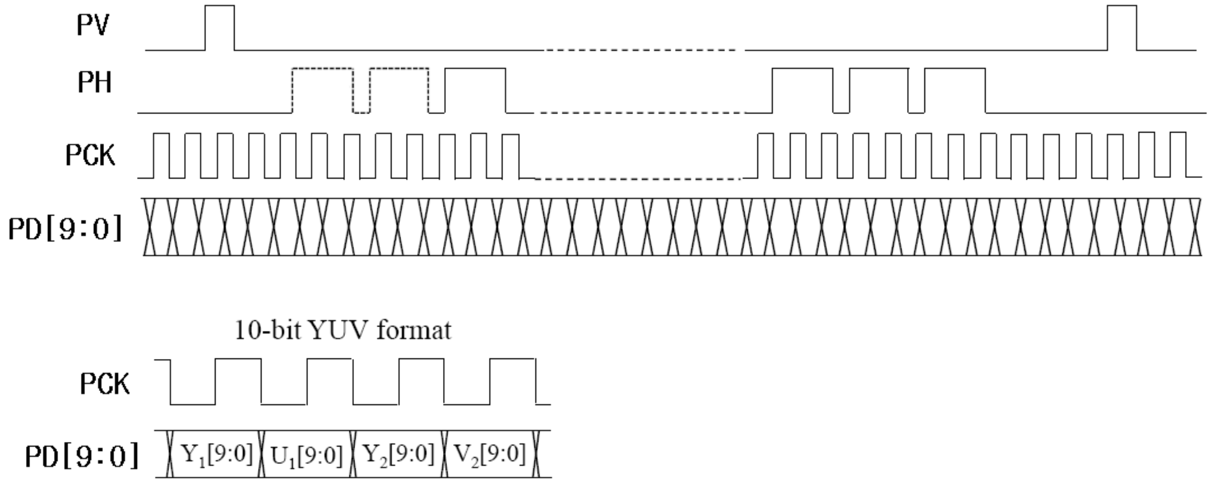


Fig. 4.11 NVP2400 YUV Input Format

4.1.6.2. INPUT CROPPING

INPUT CROPPING block supports copping of images by applying horizontal start, vertical start, horizontal active and vertical active register settings as shown in Fig 4.12. The areas outside the crop are represented as black data. Special register settings is needed in order to set the sync format to fit the area for crop data.

E-PTZ can be simply created using Input Cropping. As shown in Fig.4.13, PTZ moving can be easily realized by adjusting horizontal start and vertical start position of the image after cropping the image with 1280x720 using INPUT CROPPING function in order to generate 1280x720p output based on the assumption that image is received with the 2M pixel sensor.

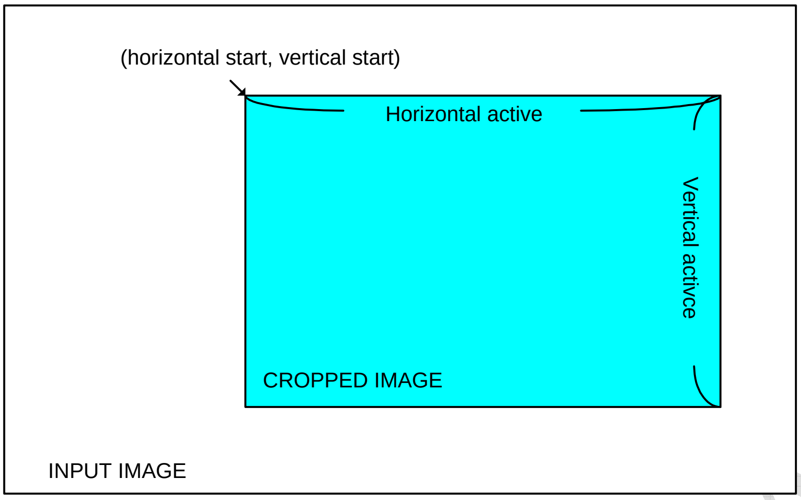


Fig. 4.12 NVP2400 Image Cropping

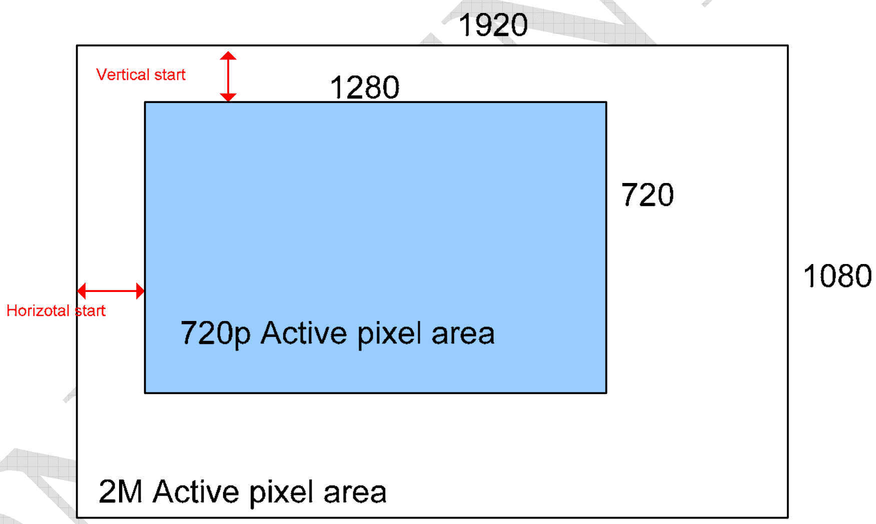


Fig. 4.13 Example of E-PTZ using Input Cropping

4.2. TIMING GENERATOR

4.2.1. OVERVIEW

Basically, NVP2400 is operated in the sensor master mode, which receives H/Vsync and data from the sensor. However, it can support sensor slave mode which sends H/Vsync to sensor depending on application and for those cases, NVP2400 has the TG (Timing Generator) block.

4.2.2. TG FOR SENSOR SLAVE MODE

Sensor slave mode is the mode where H/V sync generated by ISP is sent to sensor as input while H/V sync and bayer data, synchronized to sensor clock re-generated inside sensor, are sent to ISP. In other words, H/V sync generated from sensor and H/V sync from TG of ISP have the same period as they are locked each other. The advantage of operating sensor in slave mode is to exactly adjust the timing of sensor to the required frame rate of ISP.

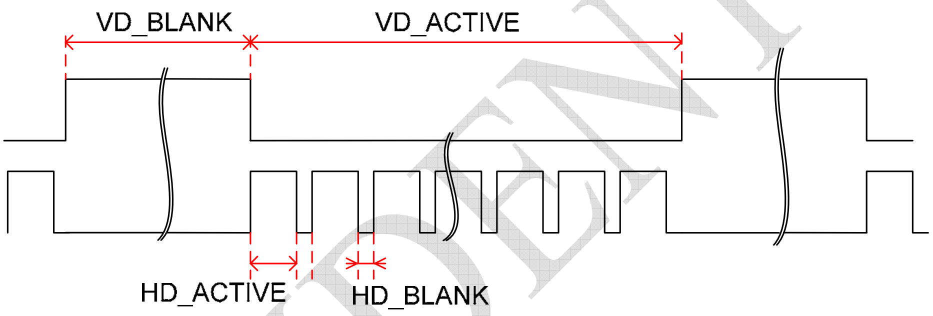


Fig. 4.14 TG Output Sync (disable FREERUN Mode)

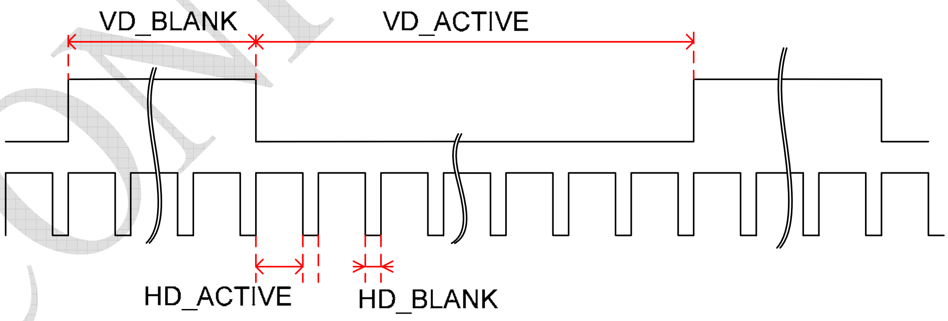


Fig. 4.15 TG Output Sync (enable FREERUN Mode)

TG responds to various slave mode format of the sensor so it supports Hsync's free-run mode. When the free-run mode is enabled, Hsync can be generated even within the VD_BLANK interval. Polarity of the TG output sync can be changed as well.

4.3. BLACK LEVEL COMPENSATION

4.3.1. OVERVIEW

CMOS sensor generates unnecessary dark current due to the process or other external environment such as temperature and others. This dark current generates signal elements in addition to the optical elements, allowing sensor’s black level to have a certain level not the pure black. Black Level Compensation (BLC) adjusts the error of the black level in the sensor generated by dark current.

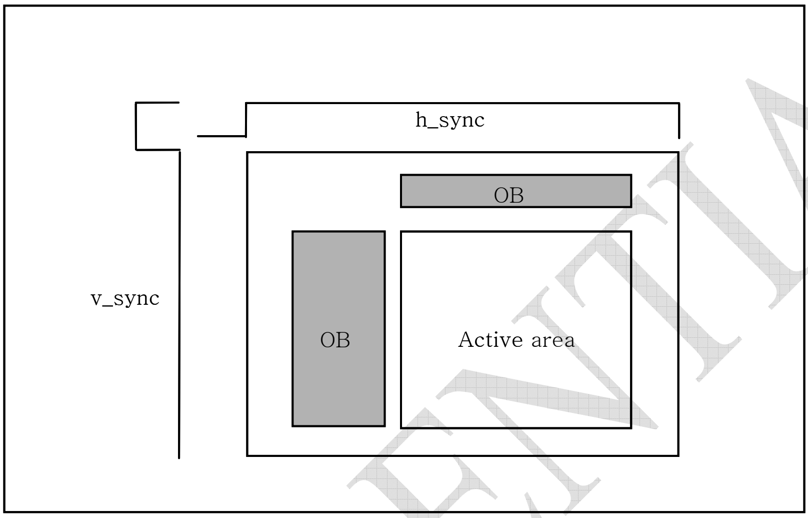


Fig. 4.16 Sensor Image Area

NVP2400 can refer to the black level of sensor’s vertical or horizontal Optical Black(OB) area in order to estimate the black level of the sensor. As shown in Fig.4.16, OB area is the area that receives no light so it is not an active area. Referring to the black level of the OB area, black level of the active area is estimated and is adjusted.

4.3.2. DESCRIPTION

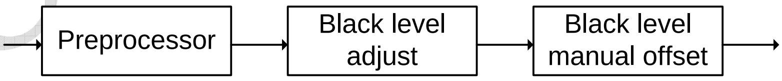


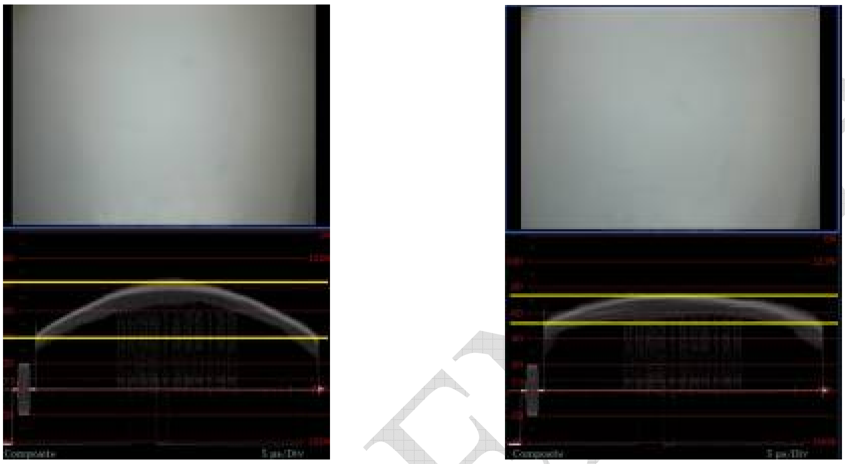
Fig. 4.17 BLC Block Diagram

Preprocessor block of BLC generates data where the black level can be easily detected by going through the filtering process in the OB area. In the black level adjust block, OB area data that went through the preprocessor block is referred in order to estimate the black level. Estimated black level value is reflected in the active area data to turn the black level of the active area data into the pure black. Black level manual offset block controls the black level of the active area using the manual offset with the separate register.

4.4. LENS SHADING CORRECTION

4.4.1. OVERVIEW

In general, the shorter the lens' focus, the bigger the difference in the incidence angle of the lights between the central and peripheral pixel. This results in the phenomenon where the image gets blurred on the periphery, which is called Lens Vignetting or Lens Shading. This Lens Shading Correction (LSC) corrects the phenomenon where the image gets darkened or blurred on the periphery. The image shown in Fig. 4.18 (a) is corrected into the one in Fig 4.18 (b).



(a) Vignetting image (b) Lens shading corrected image

Fig. 4.18 Lens Shading Correction

4.4.2. DESCRIPTION

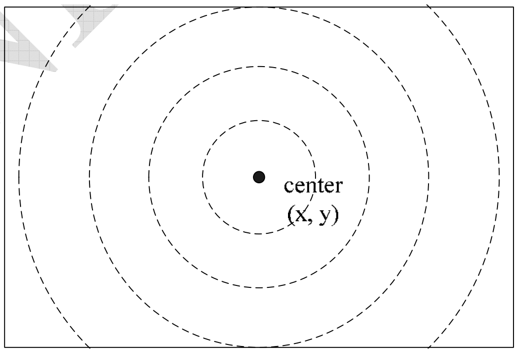


Fig. 4.19 Lens Shading Concept

Lens Shading Correction requires adjustment of the center position and brightness control of the periphery. Lens shading correction requires adjustment of the location of the center for the correction in order to respond to the location of the center of Vignetting, which changes depending on the lens. This adjustment of the center is done with the two dimensional pixel points, x and y center registers. The brightness of the periphery is adjusted by controlling the radiation intensity ratio of the modeled periphery and center. The gain of the radiation intensity ratio can be adjusted for bayer image’s Gr, R, B, Gb respectively. In addition, as the radiation intensity ratio can change depending on AE, the gain can be adjusted in accordance with AGC.

4.5. 3A OPD

4.5.1. AE/AWB OPD

4.5.1.1. OVERVIEW

AE/AWB OPD (OPtical Detector) offers the average values of the red, green, blue, luminance (Y) of the block set to process AE and AWB.

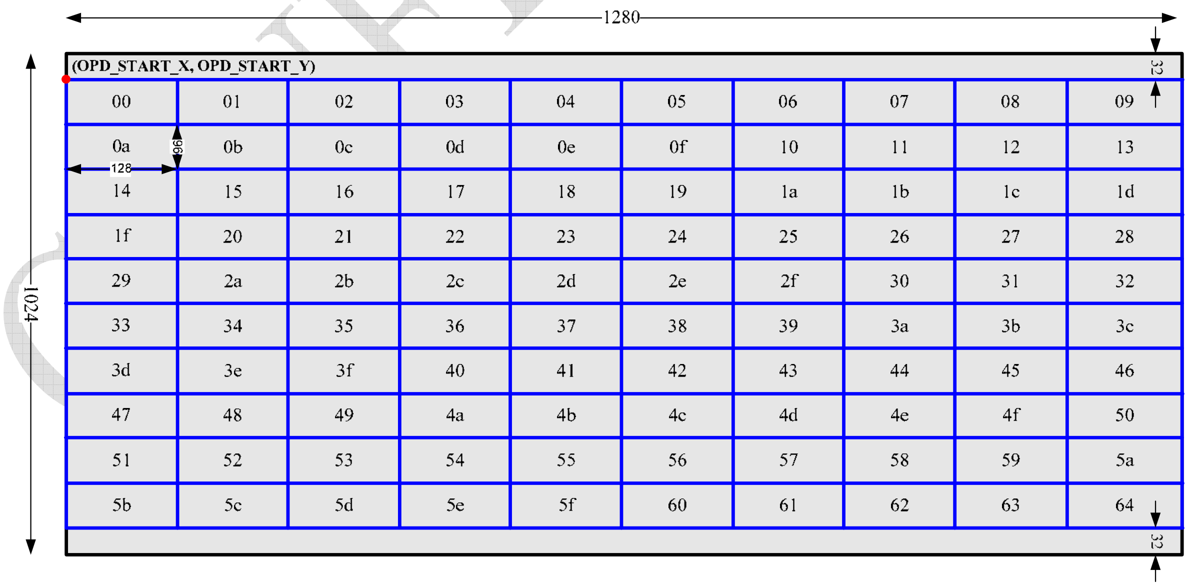
4.5.1.2. AE/AWB OPD PROCESSING

The input of AE/AWB OPD block is the bayer data from the CMOS sensor. This block offers information necessary to process AE and AWB by calculating the average values of R, G, B, Y of those OPD windows set by the register values and Y value of the entire OPD windows.

The maximum number of AE/AWB OPD window is 128 and the starting point of the AE/AWB OPD window can be adjusted on the image using the register setting.

In the case of the average value of the Luminance(Y), weight value can be given to the average calculated by using a certain register setting and its range is ×0 ~ ×15.93625. Each window can be divided into two areas: one where weight of Y mean value is given and the other where no such weight is given. This allows using different weight values of Y for different areas such as the central and periphery areas.

Fig. 4.20 shows the location of 100 R, G, B, Y windows by setting the size of each window as 128×96 when the size of the input image is 1280×1024. The blue line in Fig. 4.20 represents the OPD grid, which helps the user to find the area when setting AE/AWB OPD window.



00	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
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	41	42	43	44	45	46
47	48	49	4a	4b	4c	4d	4e	4f	50
51	52	53	54	55	56	57	58	59	5a
5b	5c	5d	5e	5f	60	61	62	63	64

Fig. 4.20 OPD Process

4.5.1.3. AE/AWB OPD GRID

OPD GRID shows the areas that the user designated in the format of a grid by referring to the starting point and the size of the window set on the AE/AWB OPD. Grid colors support black and white only. Fig.4.22 (b) shows the OPD GRID setting example for BLC(Back Light Compensation) and the designated BLC area inside the grid can be expressed in semi-transparency.

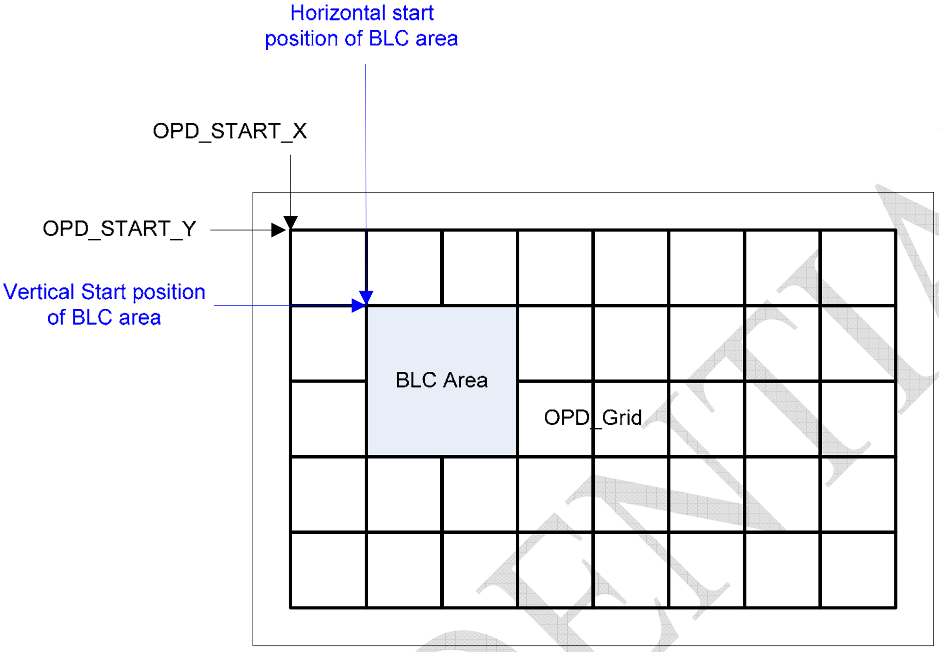

Horizontal start position of BLC area
OPD_START_X
OPD_START_Y
Vertical Start position of BLC area
BLC Area
OPD_Grid

Fig. 4.21 OPD Grid Register Description

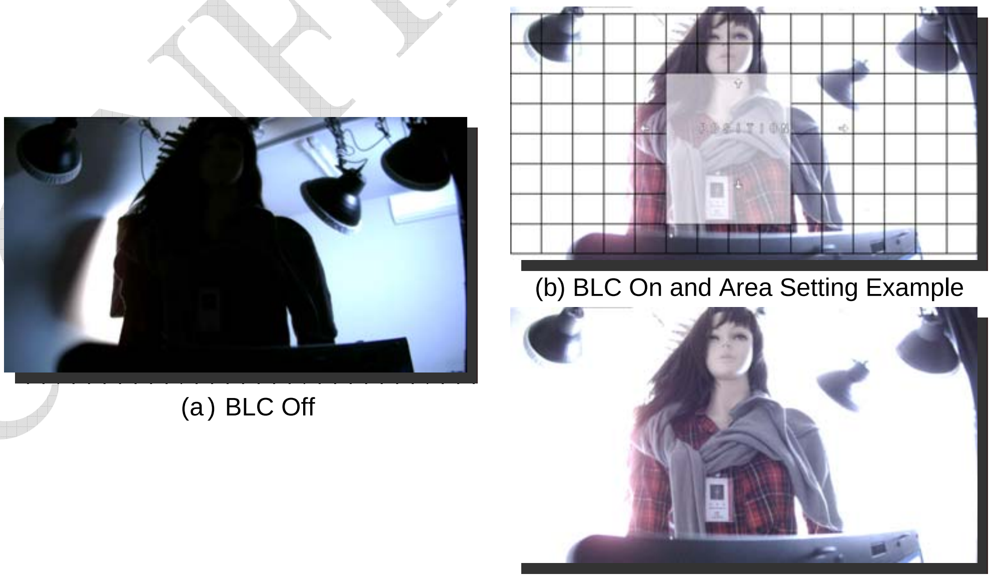


Fig. 4.22 BLC On/Off via OPD Grid Setting

4.5.2. AF OPD

4.5.2.1. OVERVIEW

NVP2400 provides edge information of an image for an external Auto Focus (AF) module. In total, five areas can be defined. For an area, information on the sum of the edge value in accordance with the high pass filter, and the pixel number of edge in the area is provided.

4.5.2.2. AUTO FOCUS PROCESSING

As shown in Fig. 4.23, AF block extracts information on the edge by applying 3x3 filtering to the brightness element (Y) of the original image or what passed 1X3 Low pass filter or 1x5 Low pass filter depending on the register setting value chosen. On the AF block, an image can be divided into 5 windows and each window can be separately saved. As shown in Fig. 4.23, coefficient values can be controlled for each window when 3X3 filter is applied.

Y_AF_COEFF11	Y_AF_CPEFF12	Y_AF_COEFF13
Y_AF_COEFF21	Y_AF_COEFF22	Y_AF_COEFF23
Y_AF_COEFF31	Y_AF_COEFF32	Y_AF_COEFF33

Fig. 4.23 3×3 Filtering for Edge Detection

Depending on the setting, information of the edges of those pixels that belong to the chosen brightness (Y) level can be summed up. For example, Fig.4.24 shows the sum of the edges that are equivalent to Y in the range of I_LO_TH ≤ Y ≤ I_HI_TH. It is configured in a way that allows a separate register setting of those pixel numbers that belong to I_HI_TH ≤ Y and I_LO_TH ≥ Y to be saved and they also can be read through an external MCU.

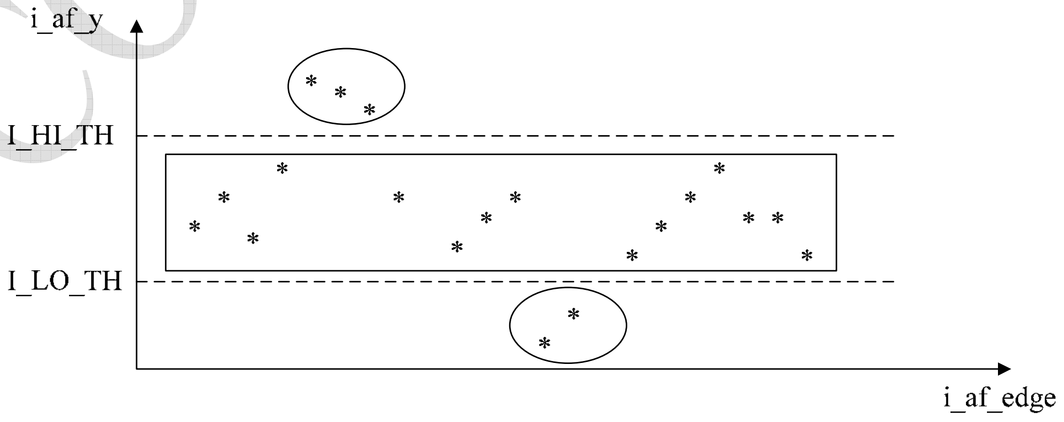


Fig. 4.24 Edge Counting

4.6. AE/AWB GAIN CONTROLLER

4.6.1. AWB GAIN CONTROLLER

NVP2400's internal MCU analyzes RGB components of the images received from AE/AWB OPD and delivers RGB coefficient for AWB control to the ISP part. AWB GAIN CONTROLLER block receives this to adjust the GAIN values of each element of R/G/B in order to maintain white balance.

4.6.2. AE CONTROLLER

AE CONTROLLER block controls the input data by making adjustment of the digital gain and offset. For the AE control, analog gain and digital gain control of the sensor and digital gain control of the NVP2400 are also supported.

AE CONTROLLER makes a linear control of the input data as follows:

$$\text{OUTPUT DATA} = \beta * (\text{INPUT DATA} + \alpha)$$

Herein α functions as the 'offset' and β 'gain'. In this block, gain is used in order to do digital AE (Auto Exposure) and offset for the manual global BLC (Black Level Compensation).

4.7. MOTION & FLICKER DETECTOR

4.7.1. MOTION DETECTION

4.7.1.1. OVERVIEW

MD (Motion Detection) consists of the pre-processor that extracts motion information and the mixer that displays the motion information on the screen. MD pre-processor divides the screen into predefined blocks and identifies whether each block has motion information. NVP2400's MOTION block also includes HLC (High Light Compensation) block. HLC masks the areas with a certain level of brightness in order to prevent the exposure time from growing short by the bright area. In such a case, the area which was not visible before is made visible as the exposure time increases. On the contrary, the bright area is not visible as it gets too saturated with light.

4.7.1.2. BLOCK DIAGRAM

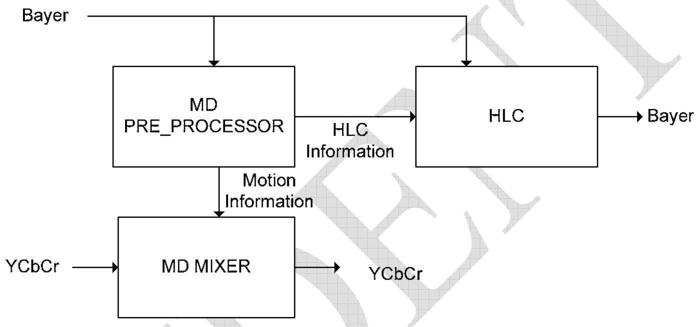


Fig. 4.25 NVP2400 MOTION Block Diagram

4.7.1.3. MD PRE-PROCESSOR

MD Pre-Processor extracts motion information of each block. The motion information extracted from each block is stored in 1980 bits so that MCU can refer to the motion information of each block. The size, number and sensitivity of each motion block can be chosen with register. However, the total number of blocks cannot exceed 1980.

The motion detection can have its sensitivity adjusted for a single level. However, it can be controlled in a way that the sensitivity is linked to the AGC value depending on the register setting as shown in Fig. 4.27.

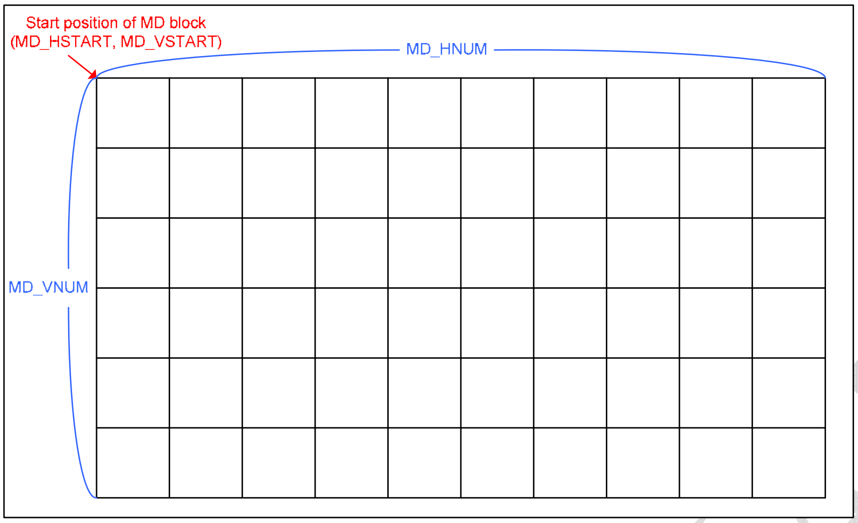


Fig. 4.26 MD Block

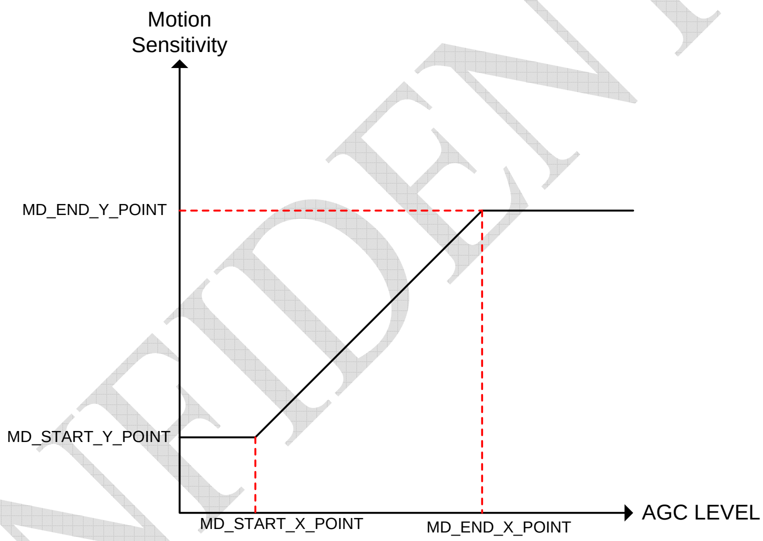


Fig. 4.27 Motion Sensitivity by AGC

4.7.1.4. MD MIXER

MD MIXER displays the motion information received from MD pre-processor on four windows of the screen. Each window can be controlled in its position and size in MD block which is determined by MD pre-processor. Each window can be set for its specific display styles such as blink, line style, color and transparency level. Motion information for each window can be referred through MD_MOTION_n(n=0, 1, 2, 3) in MCU and MD_MOTION_n(n=0, 1, 2, 3) can be outputted to EXTCK1, EXCK2, EXTINT0 and EXTINT1 BALL according to GPIO configuration

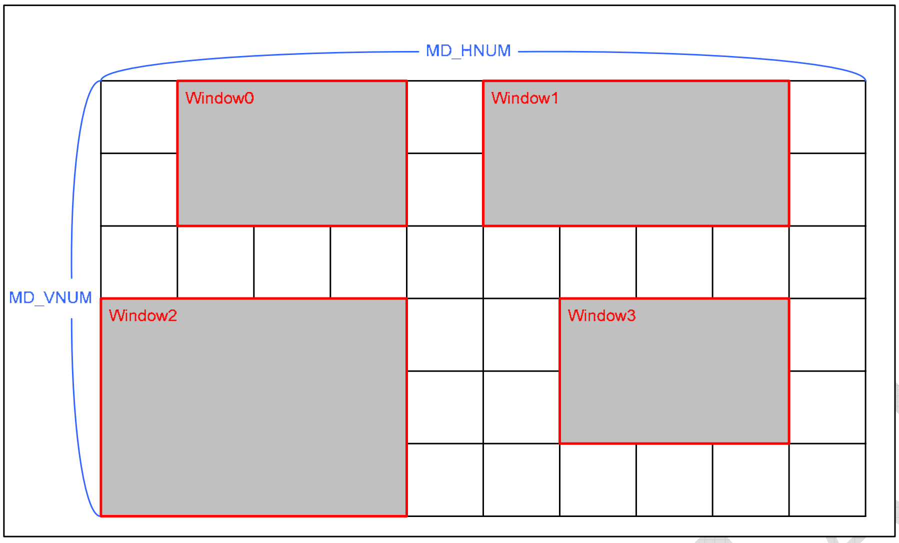


Fig. 4.28 MD Window

4.7.1.5. HIGH LIGHT COMPENSATION

NVP2400's HLC (High Light Compensation) consists of the Block-based HLC and Pixel-based HLC. The Block-based HLC operates based on the HLC information received from the MD pre-processor for each block. As shown in Fig.4.26, MD block is made in a way that MD and HLC share the same zones. Using the HLC information received from MD pre-processor, the screen is divided into four windows and they are displayed on the screen. In addition, MD block sends HLCed image to the OPD. In each window, the location and size can be controlled within the MD block defined by the MD pre-processor. For each window, display style to display on the screen can be set.

In the Block-based HLC, threshold can be controlled using HLC_LTH and HLC_HTH. Here, the Block-based HLC masks the zone which is brighter than HLC_HTH (masking zone) and bypasses the zone which is darker than HLC_LTH (bypass zone). The zones between the HLC_HTL ~ HLC_HTH are maintained in the previous state (keeping zone) (masking or bypass).

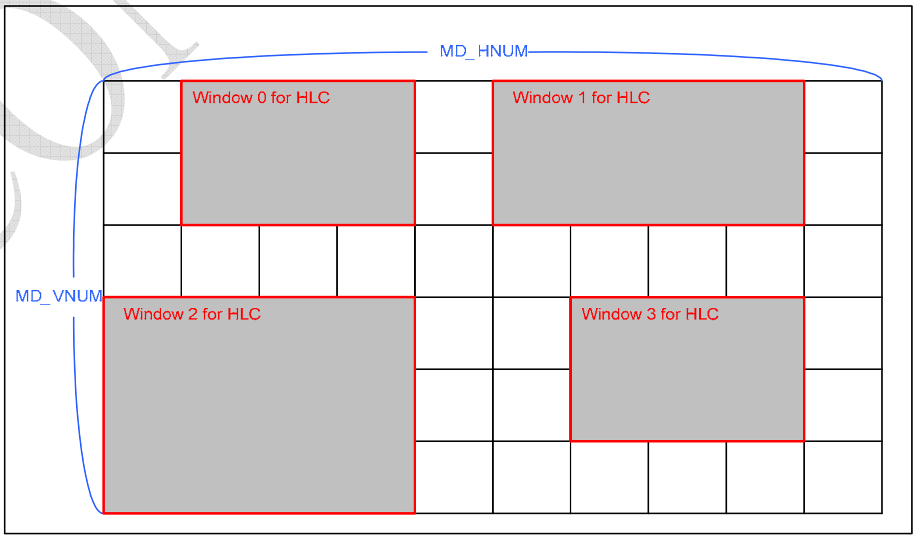


Fig. 4.29 HLC Window

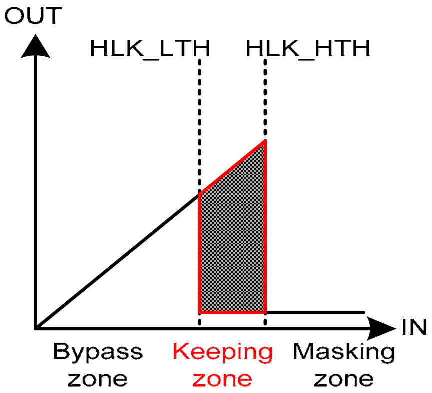
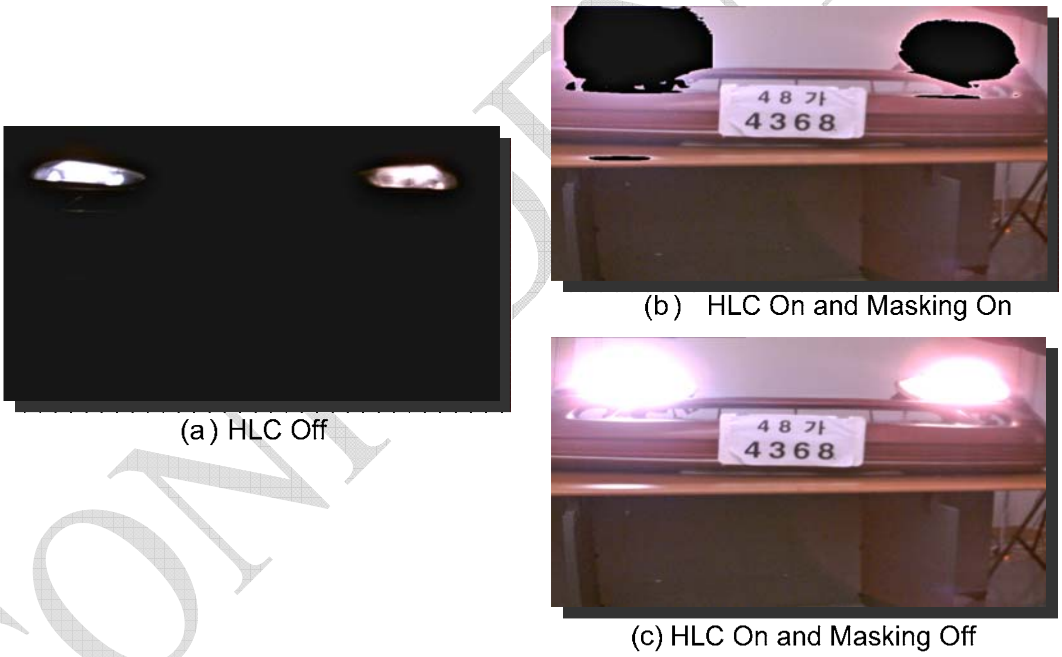


Fig. 4.30 Block-based HLC Threshold

As it was the case for Block-based HLC, four windows can be set for the Pixel-based HLC. The difference lies in the fact that masking is done in the unit of pixel as shown in Fig.4.31.



(a) HLC Off

(b) HLC On and Masking On

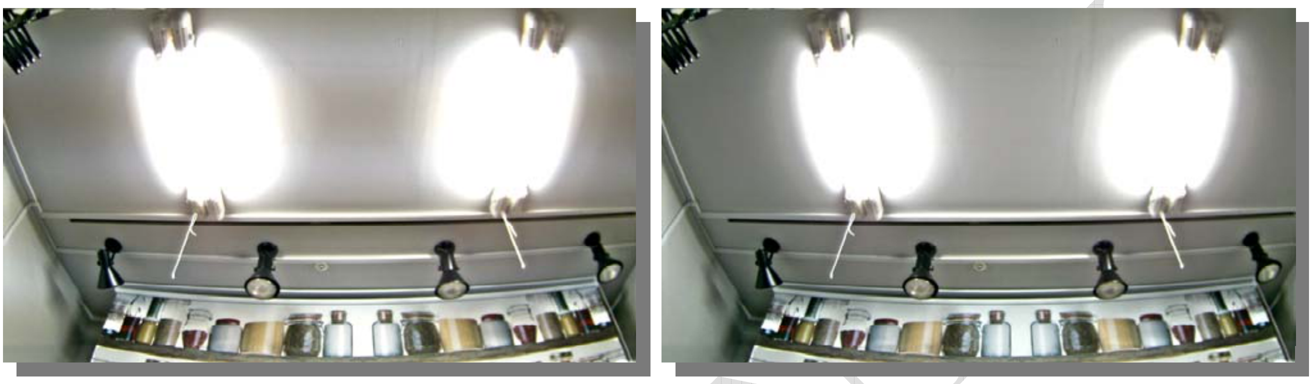
(c) HLC On and Masking Off

Fig. 4.31 Pixel-based HLC On/Off

4.7.2. FLICKER DETECTION

4.7.2.1. OVERVIEW

Flicker detection determines whether the frequency of power supply is 50 Hz or 60 Hz by obtaining the sum of Y of 128 windows in order to extract statistical information. Depending on mode, flicker detection can generate the sum of brightness level or the weighted sum of brightness level of each line.



(a) Flicker Image

(b) Flicker Suppressed Image

Fig. 4.32 Flicker Suppress On/Off

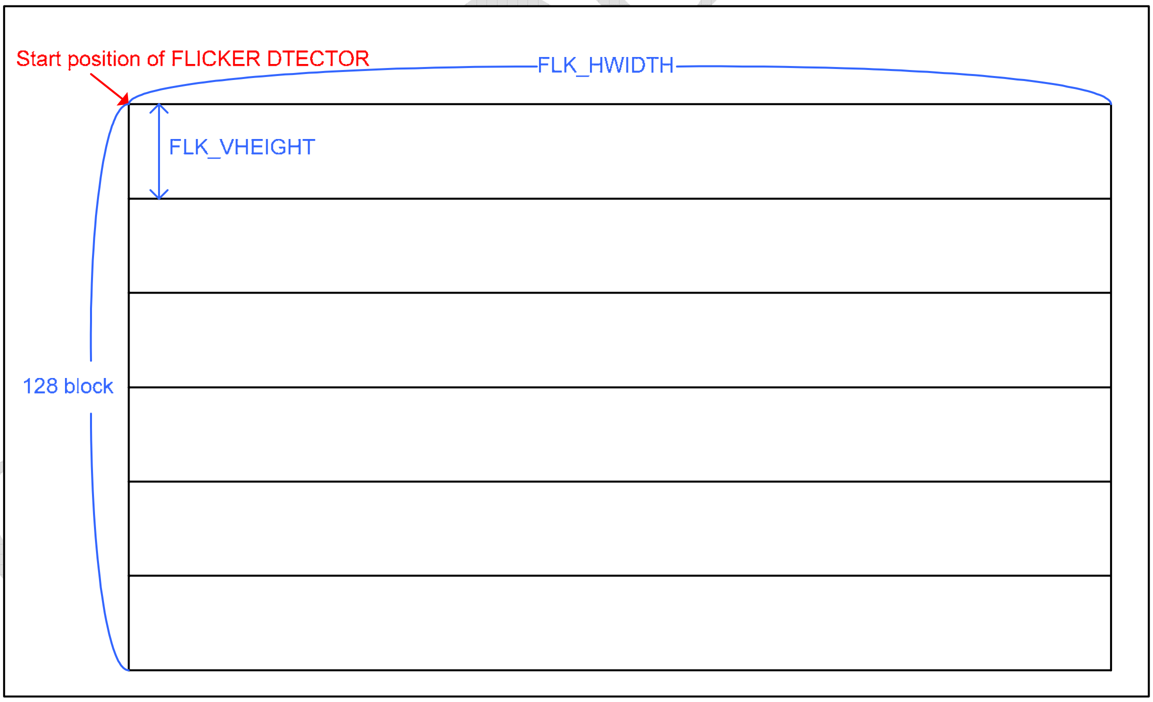
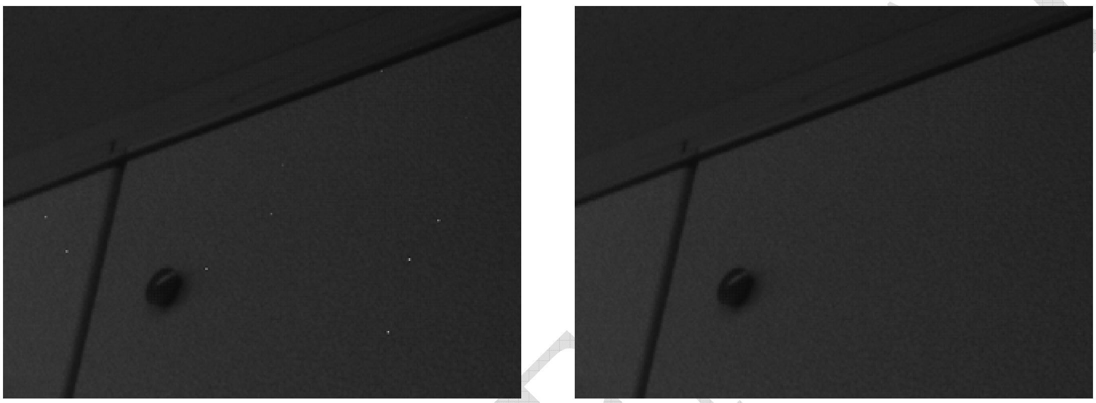


Fig. 4.33 Flicker Detector Window

4.8. DEAD PIXEL CORRECTION

4.8.1. OVERVIEW

Sensors could have defects for something wrong in the process of their storage or manufacturing process. Such defects are called dead pixels, which consist of two types: static dead pixel and dynamic one. The first can be found from the beginning while the second is found over time after using sensors for a certain period of time. Dynamic dead pixels may not be visible on the screen of ordinary illumination. However, they can be made visible by amplifying the analog / digital gain. Dead pixel correction (DPC) corrects such defects as shown in Fig.4.34 (b) after finding them in Fig.4.34 (a).



(a) Dead pixel

(b) Dead pixel correction

Fig. 4.34 Dead Pixel Correction

4.8.2. DESCRIPTION

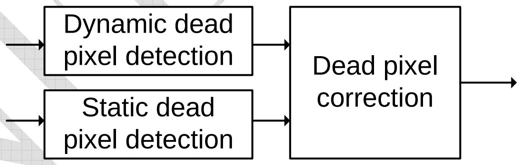


Fig. 4.35 DPC Block Diagram

Dynamic dead pixel detection block finds dead pixels on the screen with dedicated filtering. Dead pixels are represented in either white or black and both are detected by filtering. Various different types of filters are used and the detection rate is adjusted for each filter by using the threshold to judge the flat zone and the threshold to compare the pixels in the surroundings. For threshold adjustment, however, the higher the detection rate, the higher the false detection rate. So they are in trade off relations.

Static dead pixel detection block finds dead pixels using the dead pixel coordinates stored in the flash memory. Dead pixel coordinates stored in the flash memory are retrieved through static dead pixel scanning processing. Static dead pixel scan processing obtains the dead pixel coordinates through threshold adjustment in the situation where the dead pixels are made visible by adjusting the analog/digital gain and shutter with the lens closed.

Dead pixel correction block finds defects based on the information obtained from the dynamic and static dead pixel detection block and then corrects such defects.

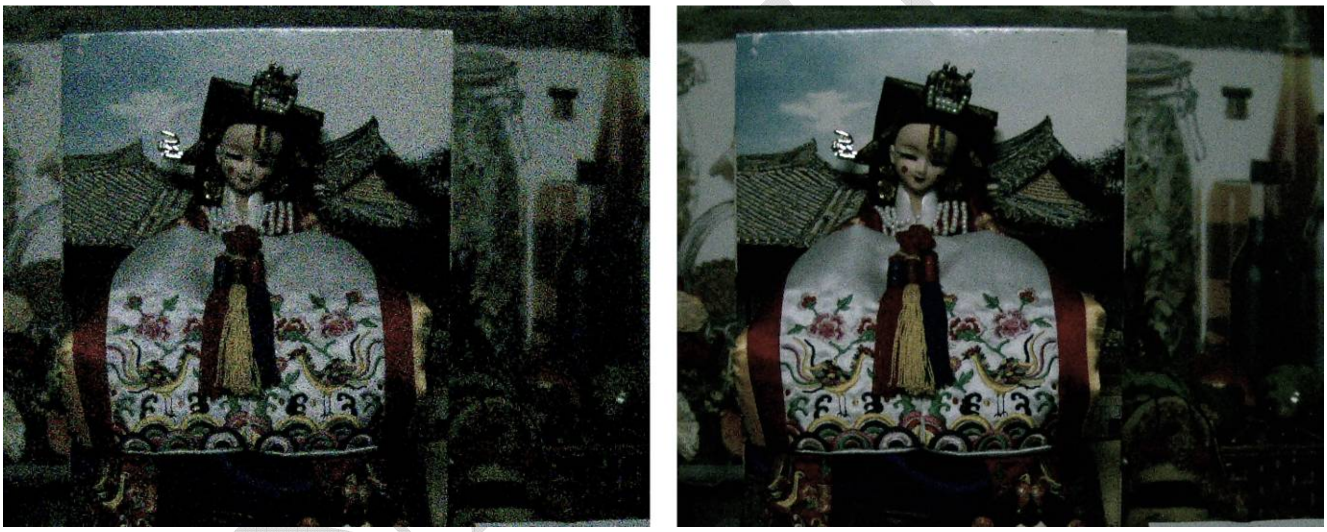
4.9. 3D-NR

4.9.1. OVERVIEW

Noise Reduction (NR) is one of the important ISP functions, which is used in order to obtain a high quality output image and enhance compression efficiency. NVP2400 uses motion adaptive 3D-NR in order to effectively remove noise.

4.9.2. DESCRIPTION

NVP2400 performs 3D noise reduction by estimating the noise level of current and previous input video. It effectively suppresses artifacts such as ghost-tail, etc., by using the motion adaptive method which reduces noise to a lesser extent in motion area where inter-pixel difference between current and previous video is greater than noise level measured and reduces noise to a greater extent in area where the inter-pixel difference is relatively small compared to the noise level measured.



(a) 3D-NR Off (b) 3D-NR On

Fig. 4.36 3D-NR On/Off

Since the critical value for the motion detection should change depending on the noise level, AGC values are used in order to estimate the noise level. For the weight of the critical values for motion detection, the AGC levels 0~255 are classified into the 7 intervals and in each interval parameters are set in a linear approach.

NVP2400's 3D-NR offers control register sets for the two (R0, R1) so that different register sets can be used depending on the state of the images. For example, if strong NR setting values are set for the R0 register set and weak NR setting values for the R1 register set, 3D-NR's performance can improve and the effect like ghost tail can reduce by applying the R1 register set if there is motion detected in the selected area and R0 register set if there is no motion. In order to detect motion, MD block's motion window is linked with the 3D-NR block.

4.10. 2D-NR

2D-NR offers the 2D-edge preserving & flat area noise reduction which preserves the edge zone by identifying the 5x5 Bayer pattern data’s edge zone and eliminates the noise on the flat zone.

2D-NR can adjust the sensitivity of NR in accordance with the AGC values. As it is the case with 3D-NR, 2D-NR has AGC levels 0~255 classified into 7 intervals and parameters are set in a linear approach for each interval. Those 7 intervals are 0~63, 64~127, 128~159, 160~191, 192~223, 224~239, and 240~255.

Fig. 4.37 shows the output threshold values in accordance with the AGC values.

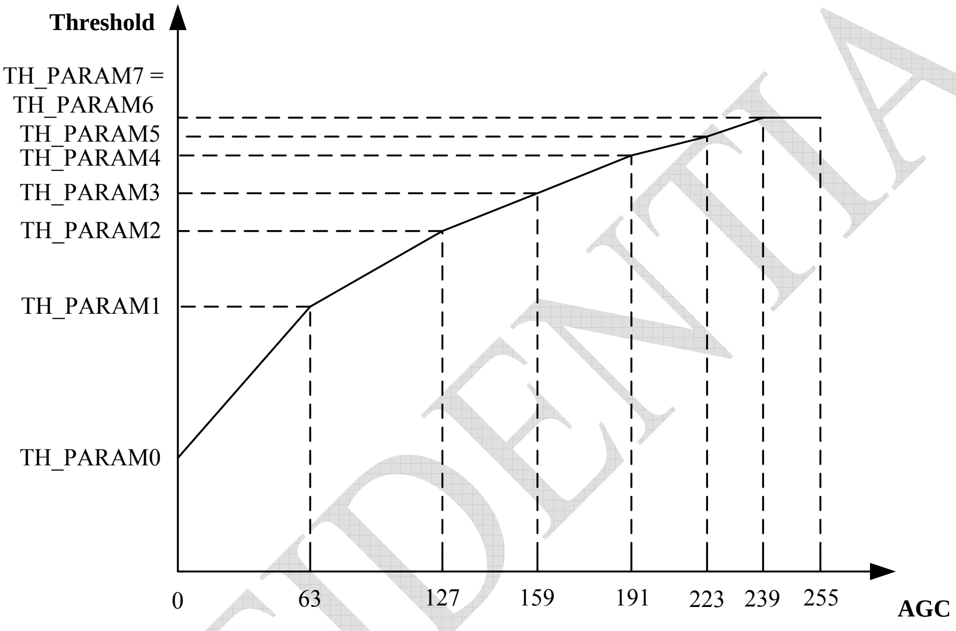
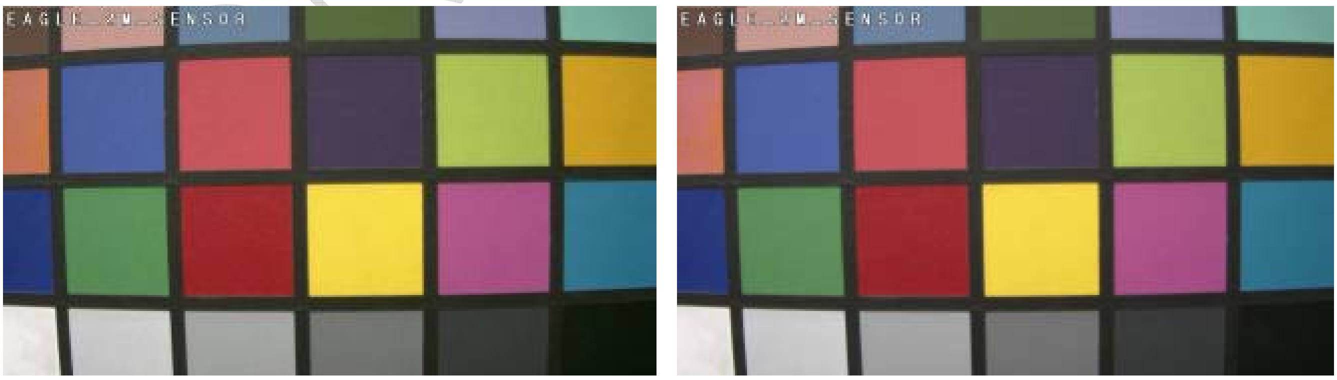


Fig. 4.37 2D-NR Threshold According to AGC

Fig. 4.38 shows 2D-NR’s on/off images.



(a) 2D-NR Off

(b) 2D-NR On

Fig. 4.38 2D-NR On/Off

4.11. FRAME RATE CONTROL

NVP2400 can generate output data using the internal timing generator and internal clock regardless of the input sensor's video sync and clock. Therefore NVP2400 can generate video data on a regular interval regardless of the input sensor and sensor mode. Fig.4.39 shows the relations between the input and output of the NVP2400 in the frame unit.

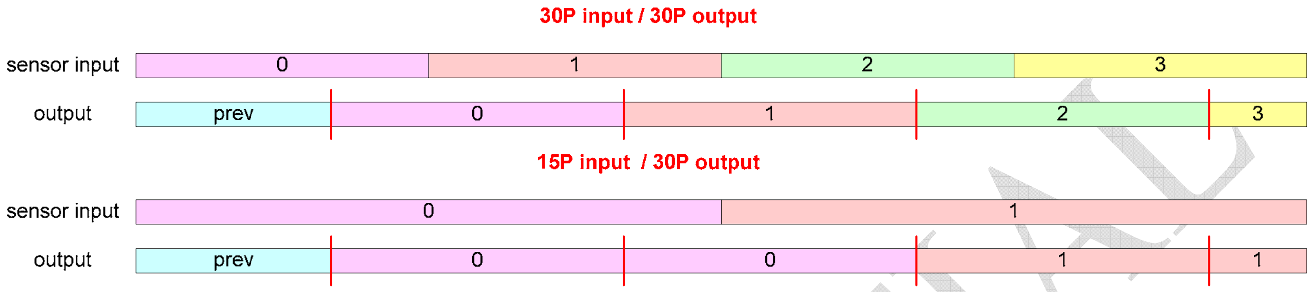


Fig. 4.39 Frame Rate Control Example of NVP2400

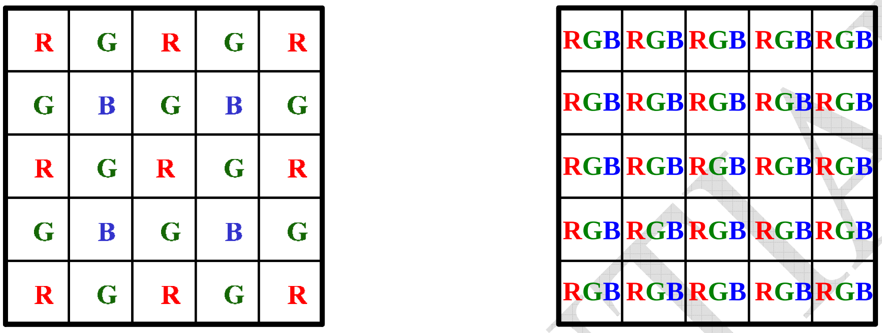
NVP2400 operates FRC (Frame Rate Control) operation using the FRC timing generator and frame memory read channel for FRC. The FRC timing generator generates new video sync depending on what the user set and FRC read channel reads the video data from the frame memory by the video sync. signals (vertical sync, horizontal sync) which were generated by the FRC timing generator. In addition, FRC read channel can supports the vertical flip and horizontal mirror operation on the image. Table 4.4 shows the example of the values set on the NVP2400.

Table 4.4 Examples of FRC Timing Generator Setting Value

Image Size	Total Value		Active Value		Sync. Value		Others
	Horizontal	Vertical	Horizontal	Vertical	Horizontal	Vertical	(BF, FF)
1280 x 720	1650	750	1280	720	370	30	0
1920 x 1080	2200	1125	1920	1080	280	45	0

4.12. COLOR INTERPOLATION

An optic filter named Color Filter Array (CFA) is required in order to express color images using the Single CMOS image sensor. CFA has a filter array where each pixel has predetermined color values to pass through and the most widely used CFA is Bayer's pattern shown in Fig. 4.40 (a). The images coming from the sensor after passing through the CFA has a single color data per pixel. This data are called Bayer Pattern Data. The rest two colors need to be restored in order to express the data into a color image and this process is called CI (Color Interpolation) or Demosaic.



R	G	R	G	R
G	B	G	B	G
R	G	R	G	R
G	B	G	B	G
R	G	R	G	R

(a) Bayer Pattern Data

RGB	RGB	RGB	RGB	RGB
RGB	RGB	RGB	RGB	RGB
RGB	RGB	RGB	RGB	RGB
RGB	RGB	RGB	RGB	RGB
RGB	RGB	RGB	RGB	RGB

(b) Demosaiced Data

Fig. 4.40 Color Interpolation

4.13. RGB GAMMA

4.13.1. RGB2RGB CSC

This image converted into RGB from Bayer's with Color Interpolation goes through the RGB-to-RGB Color Space Conversion (CSC) process again in order to express the colors that the user wants. Such RGB-to-RGB conversion is done using the following formula.

$$R' = GAIN0*R + GAIN1*G + GAIN2*B + R\_OFFSET$$

$$G' = GAIN3*R + GAIN4*G + GAIN5*B + G\_OFFSET$$

$$B' = GAIN6*R + GAIN7*G + GAIN8*B + B\_OFFSET$$

The gain of each channel is to be set in accordance with the target output.

4.13.2. RGB GAMMA CORRECTION

4.13.2.1. OVERVIEW

With NVP2400, Gamma correction refers to the process of making sure that linear features are expressed using the Inverse Gamma Transform in an attempt to compensate the nonlinear features of the RGB color elements, which result from the spectral feature of the sensor. NVP2400 can adjust the gamma curve for each channel of R, G, B using the piecewise linear gamma correction as shown in Fig. 4.41. In addition, using Y-axis' gamma point register, each channel's gamma curve can be controlled.

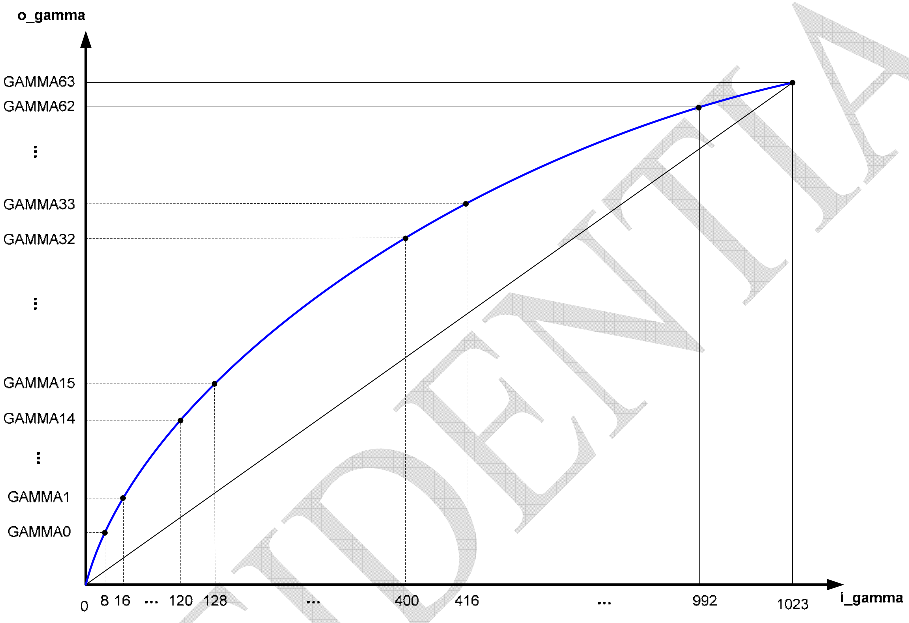


Fig. 4.41 Gamma Correction

4.13.3. RGB2YCBCR CSC

Within ISP, data on the Bayer domain are converted through Color Interpolation into the data on the RGB domain. In addition, the RGB data are converted into YC data in order to support processing on the YC domain. RGB-to-YCbCr CSC block is used to convert the RGB data into the YCbCr data and the coefficient values required for the conversion can be set as necessary.

RGB-to-YCbCr conversion is done with the following formula.

Y = GAIN0*R + GAIN1*G + GAIN2*B + Y_OFFSET

Cb = GAIN3*R + GAIN4*G + GAIN5*B + Cb_OFFSET

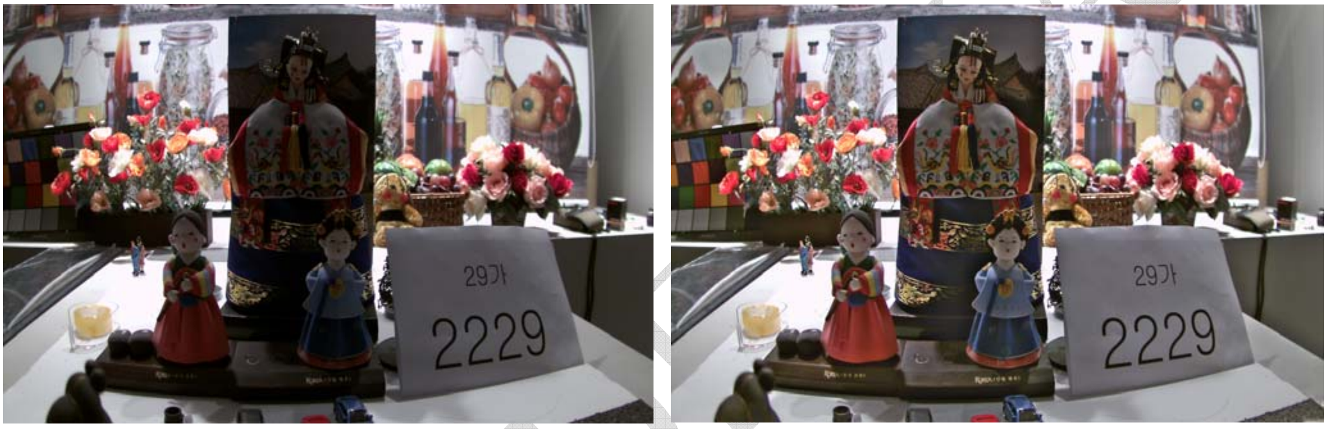
Cr = GAIN6*R + GAIN7*G + GAIN8*B + Cr_OFFSET

The GAIN of each channel is to be set in line with the target output.

4.14. ACCE

4.14.1. OVERVIEW

ACCE (Adaptive Contrast and Color Enhancement) block performs an image enhancement processing to enhance visibility of an image by changing the brightness values to the level that people can recognize the change. This function is offered based on the basic feature of the human eye sights, which are more sensitive to the changes in the areas with high brightness level than those with low brightness values. This processing technique is equivalent to the dynamic range reduction technique, which converts the HDR (High Dynamic Range) into the restricted LDR (Low Dynamic Range). This dynamic range reduction technique is one of the WDR (Wide Dynamic Range) technologies. Through this, the contrast ratio of the image can be effectively expressed and the edge information can be improved in order to enhance the overall visibility of the image.

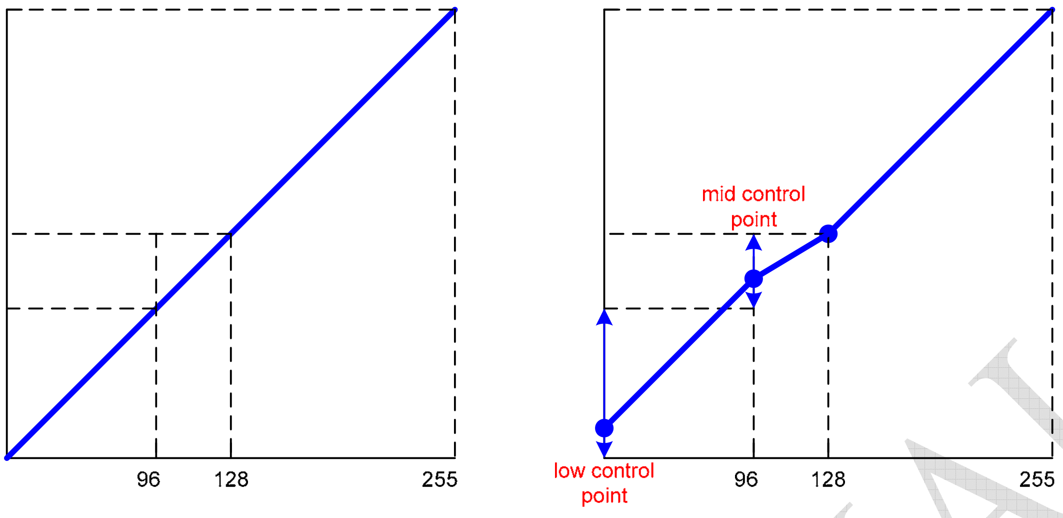


(a) ACCE Off (b) ACCE On

Fig. 4.42 ACCE On/Off

4.14.2. DESCRIPTION

One of ACCE functions is to the detail of the edge, which results in a better edge contrast of the overall image. The level of enhancement of the details of the edge can be controlled with the register. Enhancing the detail of the edge may lead to an amplified noise in the zone of low luminance. This is why ACCE has a built-in function to reduce noise through the ALL-NR (Adaptive Low Luminance Noise Reduction) and YLPF processing. ACCE controls overall luminance values by applying the tone mapping function in addition to enhancing its edge details.

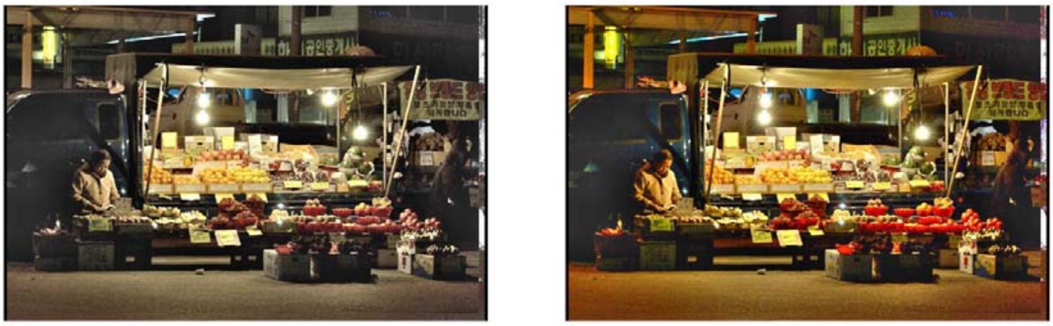


(a) Default tone mapping function (b) ACCE-On tone mapping function

Fig. 4.43 ACCE Tone Mapping Function

In Fig. 4.43 (a), default tone mapping function with ACCE Off has no change between input and output as the input brightness values (X axis) and output brightness values (Y axis) in the range of 8bit 0~255 have the same values. However, with the ACCE On, a tone mapping function is used to adjust the brightness values of the zones with a low luminance by adjusting 2 control knee points in the vertical direction. Raising the brightness value of the zones with the low luminance through the low/mid control point can enhance visibility. This function is offered based on the basic feature of the human eye sights, which are more sensitive to the changes in the areas with high brightness level than those with low brightness values.

ACCE performs color enhance processing for not only luminance but also color channels. The changes in the input and output colors can be controlled with a register.



(a) ACCE Color Processing (Before) (b) ACCE Color Processing (After)

Fig. 4.44 ACCE Color Processing

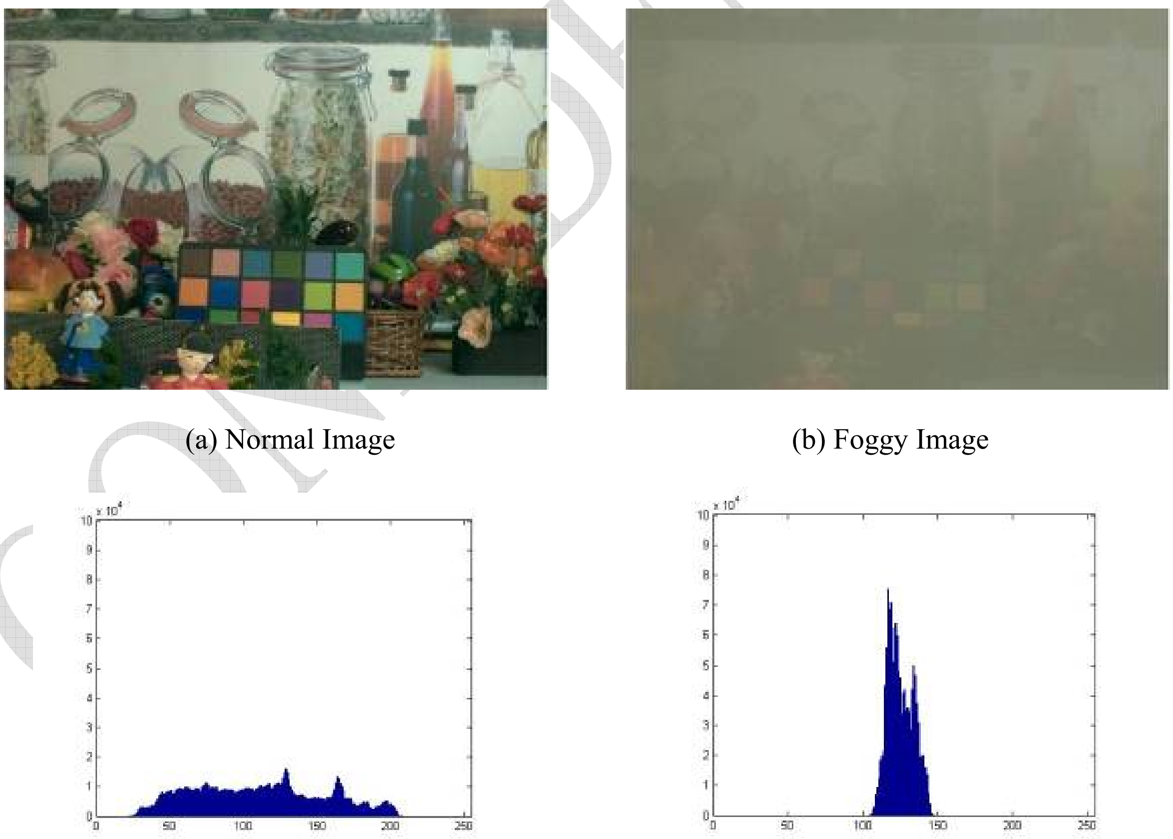
4.15. DEFOG

4.15.1. OVERVIEW

Images in extraordinary environment such as fog or rain or in a very strong luminous intensity have lower dynamic range than ordinary images. DVP2400 has a contrast-based defog function, which is used to overcome such shortcoming.

4.15.2. DESCRIPTION

In the environment of ordinary luminance, a very high proportion of the light reflected from the object is received by the sensor light receptor. However, in an extraordinary environment such as in a fog, a number of small particles are received by the sensor light receptor along with other light sources in the surroundings. This phenomenon reduces dynamic range of an image. Comparison of the image of Fig. 4.45 (a) obtained in a general environment and another image shown in Fig. 4.45 (b), which is obtained in a simulated fog situation, shows the clear difference between the two images in terms of dynamic range. To make it easier to compare the two, the luminance elements were expressed in a histogram as shown in Fig. 4.45 (c) and Fig. 4.45 (d). While the image obtained in an ordinary environment has a wide distribution of the luminance elements ranging from 25 to 200, the one in a fog has a narrow distribution from 100 to 150.



(a) Normal Image

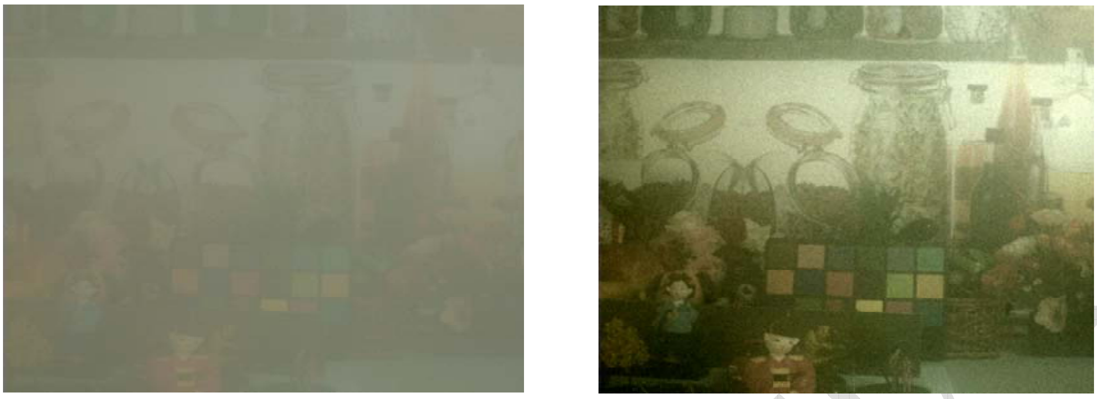
(b) Foggy Image

(c) Histogram of Normal Image

(d) Histogram of Foggy Image

Fig. 4.45 Histogram Distribution of Normal and Foggy Image

DEFOG block enhances the dynamic range of an image as shown in Fig. 4.46 by analyzing the histogram for each zone for the foggy image and improving it into the one of normal histogram.



(a) DEFOG Off (b) DEFOG On

Fig. 4.46 DEFOG On/Off

4.16. YC PROCESSOR

4.16.1. OVERVIEW

NVP2400's YC Processor consists of Y Processor and C Processor. Y Processor performs the Contrast, Brightness, Y gamma, and Edge enhancement functions while C Processor performs Saturation, C gamma, Hue control, and AGC/Highlight/Edge suppression functions.

4.16.2. CONTRAST & BRIGHTNESS

Fig.4.47 shows calculations of contrast and brightness. GAIN controls the contrast of Y values and the scope of application is ×0 ~ ×1.992. OFFSET controls the brightness and has the scope of application of -128 ~ +127.

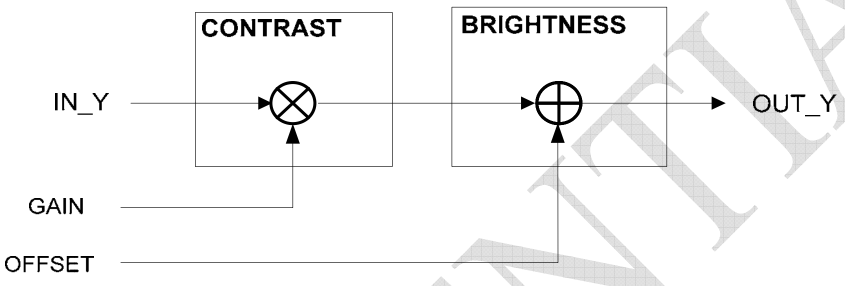


Fig. 4.47 Contrast and Brightness Control

4.16.3. Y GAMMA CORRECTION

As it was the case for RGB GAMMA Correction, Gamma Table is applied to Y. Y GAMMA also consists of 64 steps like RGB GAMMA.

4.16.4. EDGE ENHANCEMENT

Edge enhancement enhances the sharpness around the edge and reducing the transition time of the edge. Fig. 4.48 explains the edge enhancement. Through the 2D mask operation, edge intensity, edge coring and clipping and other functions are supported.

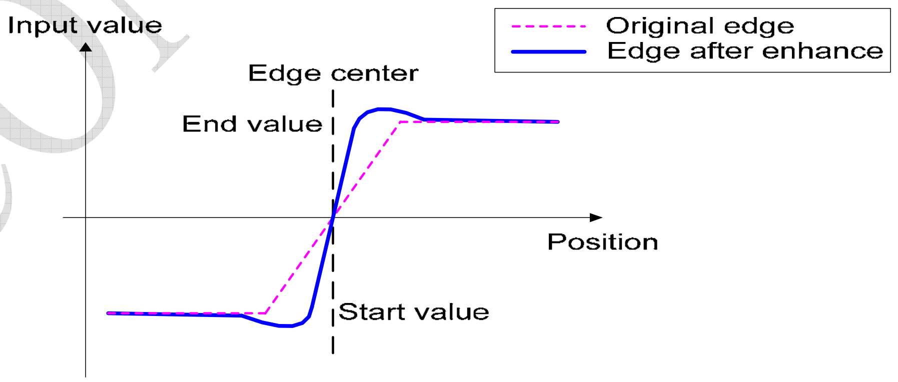


Fig. 4.48 Edge Enhancement

Besides the 2-dimensional edge enhancement function, NVP2400 also provides another horizontal edge enhance block, which is called MAIN HPF. This MAIN HPF supports cropping and gaining of the horizontal edge.

4.16.5. SATURATION & HUE

Saturation control can be made by multiplying gain values to all the C (cb/cr) pixel values. The range of adjustment is ×1 ~ ×1.992.

For the sake of detailed adjustment, hue control can be done in six zones depending on the color distribution as shown in Fig. 4.49.

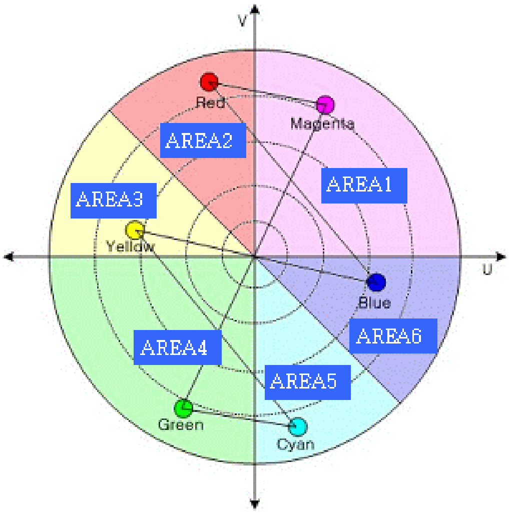
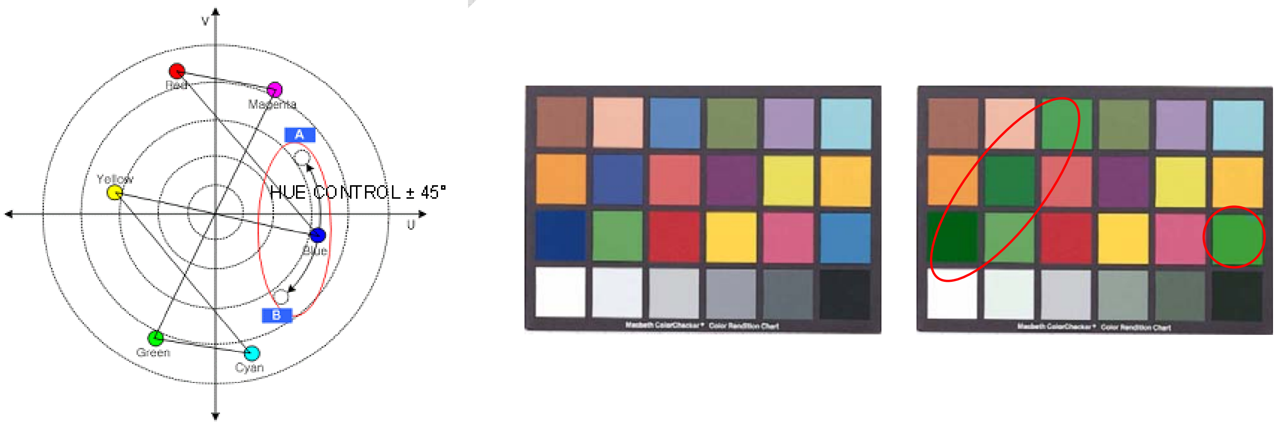


Fig. 4.49 Color Area

Hue control is a combination of Cb and Cr as shown in Eq. (1).

$$\begin{aligned} Cb' &= Cb \cos\theta + Cr \sin\theta \\ Cr' &= Cr \cos\theta - Cb \sin\theta \end{aligned} \qquad (1)$$

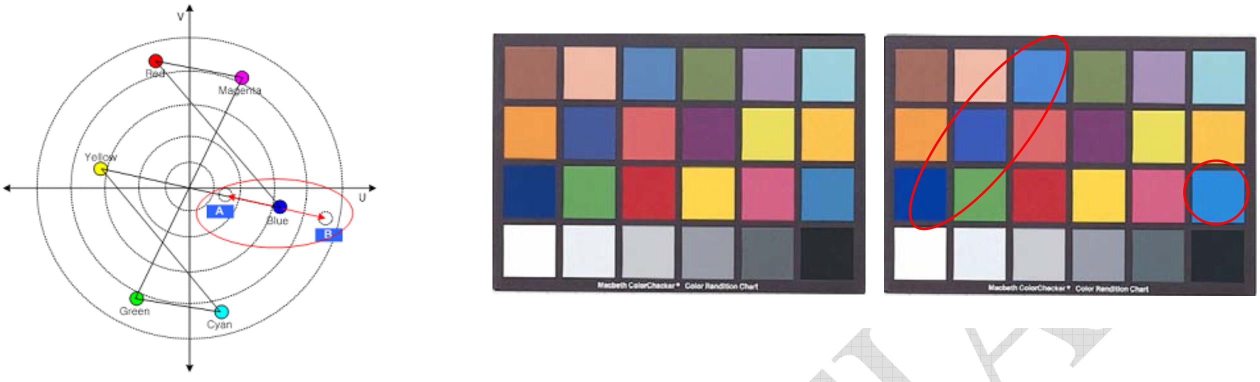
For the 6 color areas, ±45° Hue control can be done. Fig. 4.50 shows an example of adjusting blue, which is AREA6 by controlling a register. As seen in the figure, the angle can be adjusted in the direction of A and B.


V
Red
Magenta
A
Yellow
HUE CONTROL ± 45°
U
Blue
B
Green
Cyan
Macbeth ColorChecker Color Rendition Chart
Macbeth ColorChecker Color Rendition Chart

(a) HUE Control (b) Original Image (c) Result Image

Fig. 4.50 Hue Control

In addition, not only the angle but also hue gain control can be made for the 6 color areas. Fig. 4.51 shows adjustment of the blue, which is AREA6. Using the register, it can be adjusted in the directions of A and B on the vector scope. As shown in Fig. 4.51, raising the hue gain results in a dark blue while lowering the gain results in bright blue.


V
Red
Magenta
Yellow
U
A
Blue
B
Green
Cyan
Macbeth ColorChecker Color Rendition Chart
Macbeth ColorChecker Color Rendition Chart

(a) HUE GAIN control (b) Original image (c) Result image

Fig. 4.51 Hue Gain Control

4.16.6. CB/CR GAMMA CORRECTION

As it was the case with RGB Gamma and Y Gamma correction, 64-step Gamma table is applied to Cb and Cr.

4.16.7. FALSE COLOR SUPPRESSION

NVP2400 has three type of false color suppression functions including AGC color suppression, highlight color suppression and edge color suppression.

In a general low luminance, AGC is raised in order to brighten up the screen. This amplifies noise as well. AGC Color suppression is used in order to improve the quality of image by suppressing the color difference noise amplified by AGC. The level of color suppression due to AGC can be adjusted with the linear function. Linear function reduces the output value by multiplying the gain value to the input value and the range of gain values is ×0 ~ ×1. Here the range of START_X and END_X is 0 ≤ START_X < END_X ≤ 255 while START_Y and END_Y have the range of 0 ≤ START_Y < END_Y ≤ 128. Various functions can be created by adjusting START_X, END_X, START_Y, and END_Y register.

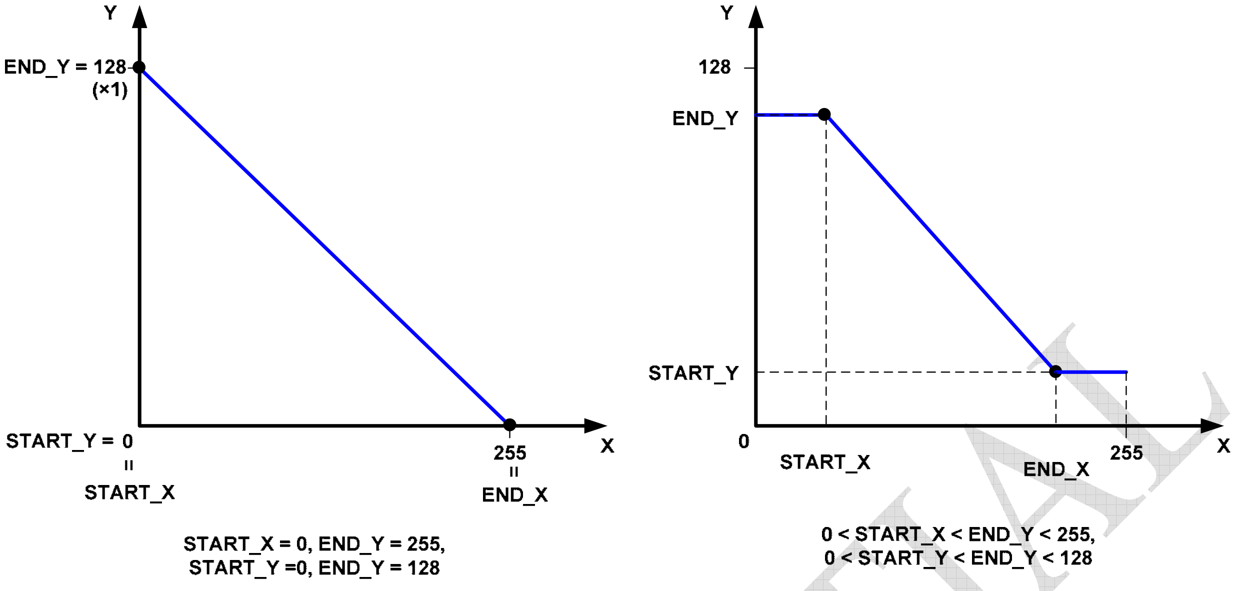
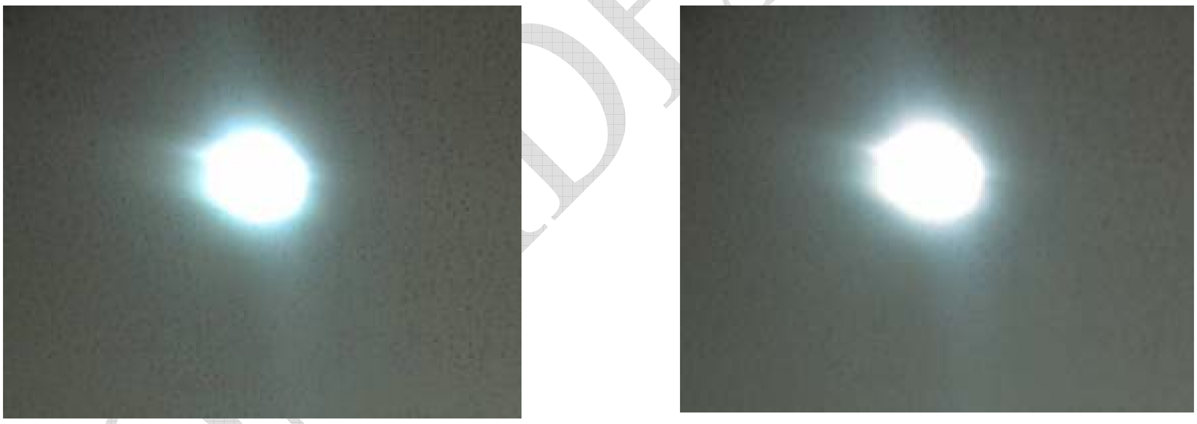


Fig. 4.52 Linear Function

In the case of CMOS sensor, false color is generated around the very bright light source like the light bulb in Fig. 4.53 (a) and this is called blooming. Suppressing the colors which are brighter or darker than a predetermined value is called highlight color suppression.



(a) Original Image (b) Result Image

Fig. 4.53 High Light Color Suppression

False color can also be created on the edge. This is why edge color suppression is performed based on the edge information received from Y processor block. Based on the information, suppression level can be adjusted using the linear function as shown in Fig.4.52.

4.17. PRIVACY MASKING (PM)

Privacy masking allows masking a certain spot on the screen. NVP2400 offers a total of 8 Privacy Masking Windows. Priority is allocated to each window so when multiple number of windows is overlapped, window (Window 7 is given the highest priority) at a higher level is given priority. For each window, color, boundary (yes/no), image inversion (yes/no), black & white (yes/no) and transparent level can be adjusted.

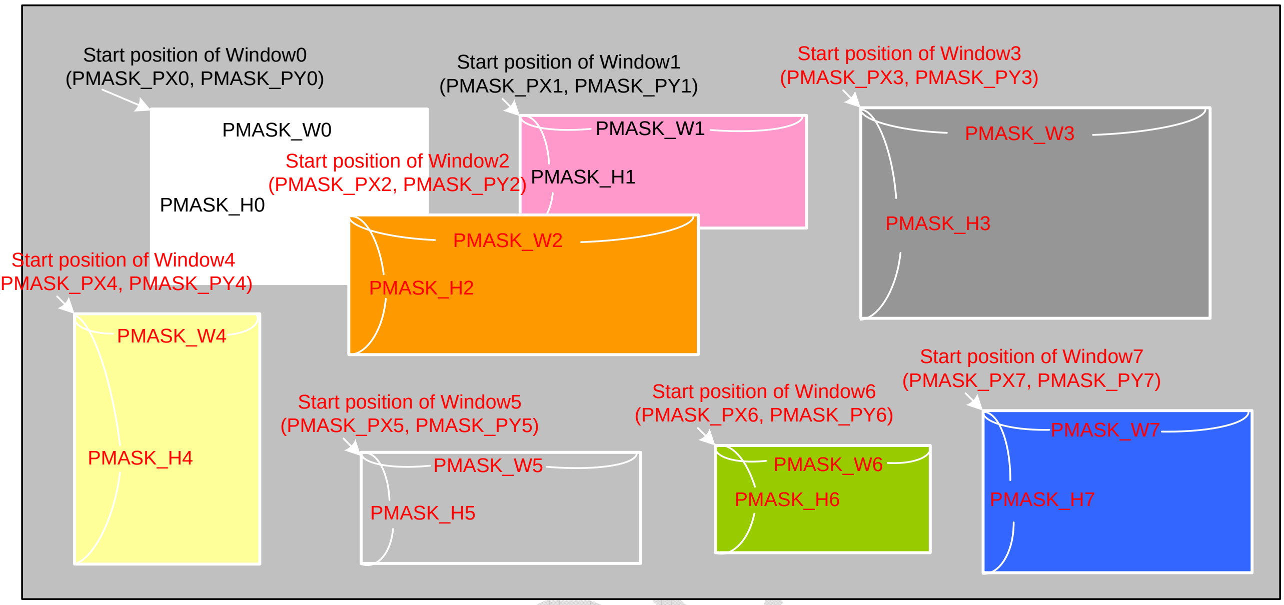


Fig. 4.54 Privacy Masking Window

4.18. D-ZOOM & PIP

4.18.1. OVERVIEW

NVP2400 offers Digital Zoom and PIP functions. ZOOM supports 4 ~ 64 times of zooming while PIP is supported from 1/9 to 1/16 times.

4.18.2. DESCRIPTION

Fig.4.55 is the diagrams of D-ZOOM & PIP concepts. As seen in the diagram, the image in the ZOOM area on the left is zoomed up (D-ZOOM) as shown in the diagram to the right. The entire image of the left is downsized to the PIP image on the right.

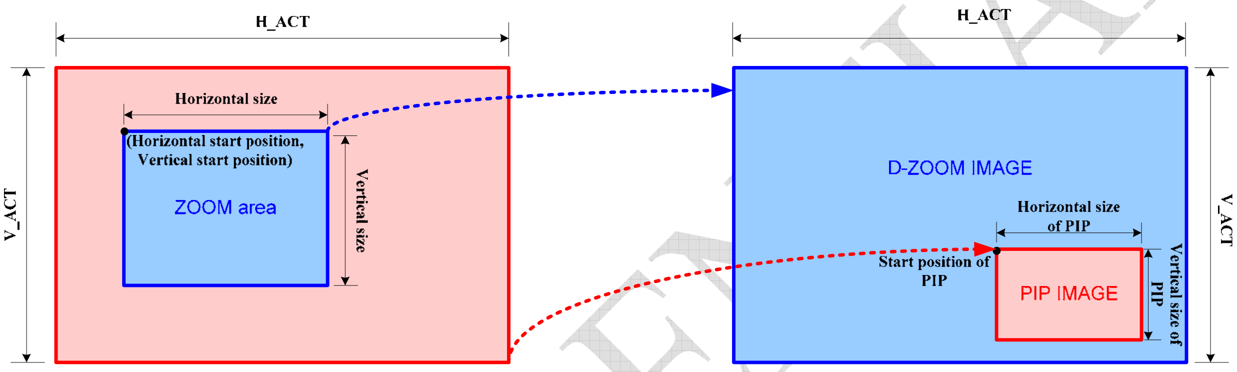


Fig. 4.55 Conceptual Diagram of D-ZOOM & PIP

ZOOM area can be determined by cropping an input image. In Fig. 4.55, the size of an entire image is H_ACT × V_ACT. It is zoomed by as much as the horizontal size starting from the horizontal start position in the horizontal direction and by as much as vertical size from the vertical start position in the vertical direction. Fig. 4.56 shows an example of zooming of an actual image.

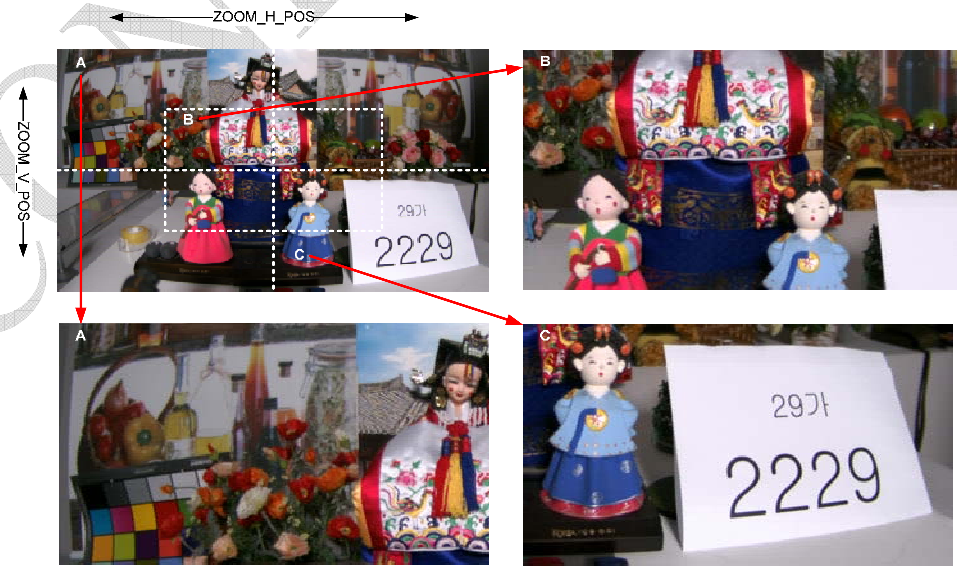


Fig. 4.56 D-Zoomed Area

With NVP2400, D-ZOOM area can be expressed within a PIP image. Fig 4.57 shows an example of indicating a D-ZOOM area within a PIP image.

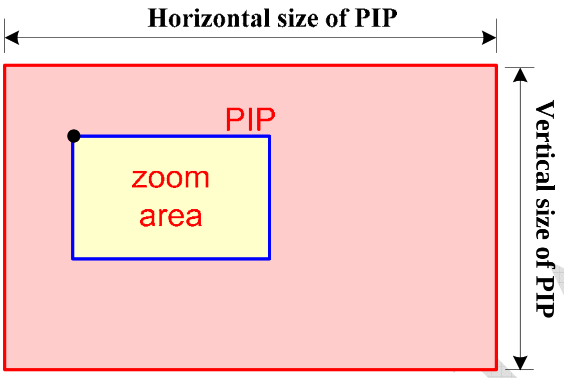


Fig. 4.57 D-ZOOM Area Indication in PIP Image

Fig 4.58 shows overlaid images of 4 time D-ZOOM and 1/9 PIP of the actual image size of 1920×1080.

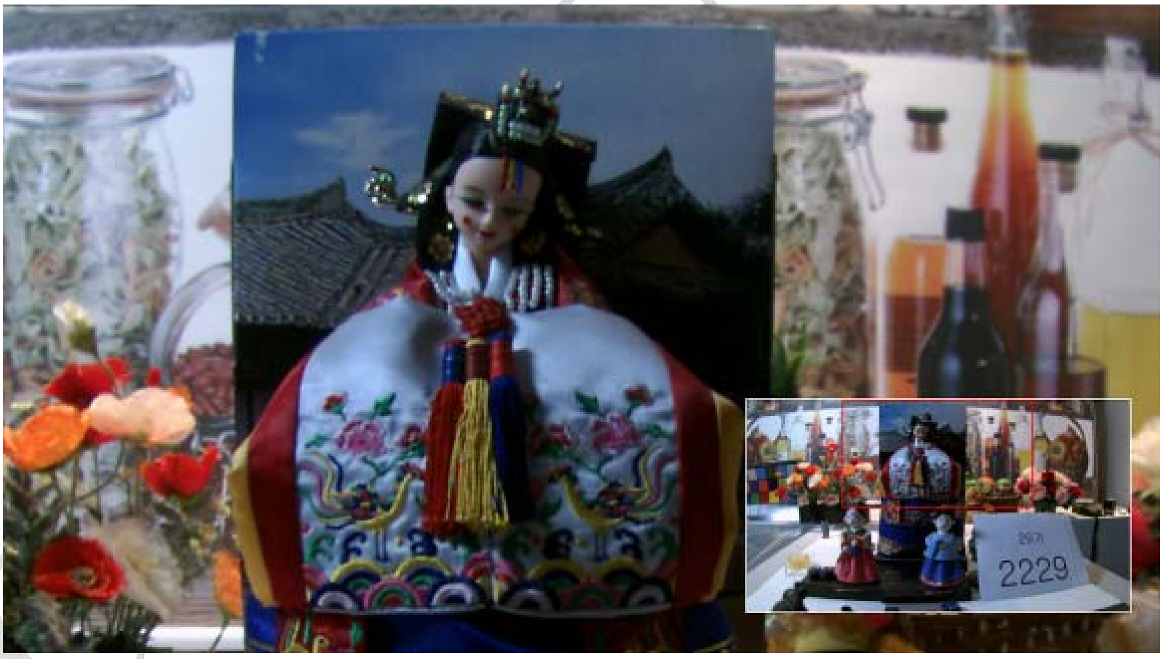


Fig. 4.58 D-ZOOM(×4) & PIP(1/9)

4.19. OSD

4.19.1. OVERVIEW

OSD (On Screen Display) is a function to display on the screen the user configured menu by using font data. It is broken down into the SD OSD for OVBS output and HD OSD which supports HD.

HD OSD supports RAM fonts only and each font comes in 48x48 pixels. SD OSD supports RAM and ROM fonts and each font has the 32x32 pixel. RAM font supports 448 fonts while SD OSD's ROM font supports 64 including figures, english characters and symbols. For each font, its color, background, blinking and transparency level can be controlled. The user can use 8 colors for the font and 3 for the background color. As for the outline, font color is used for the background and brightness (Y) level can be adjusted.

4.19.2. DESCRIPTION

HD OSD's fonts consist of 48 x 48 pixels, while SD OSD fonts 32x32 pixels. In the case of SD OSD supporting ROM font (Fig. 4.60), index 0 ~ 63 (64 units) are ROM font's index while index 64 ~ 512 (448 units) is RAM font's index.

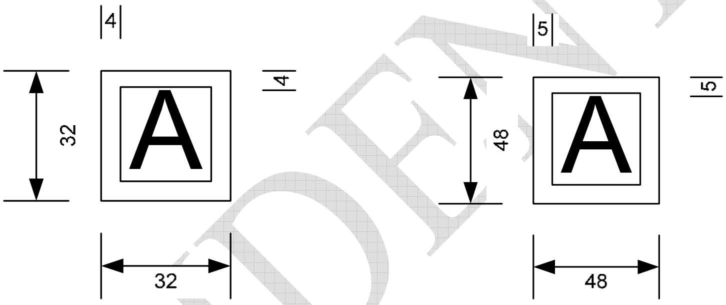
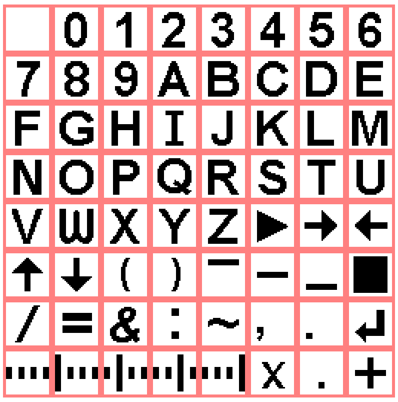


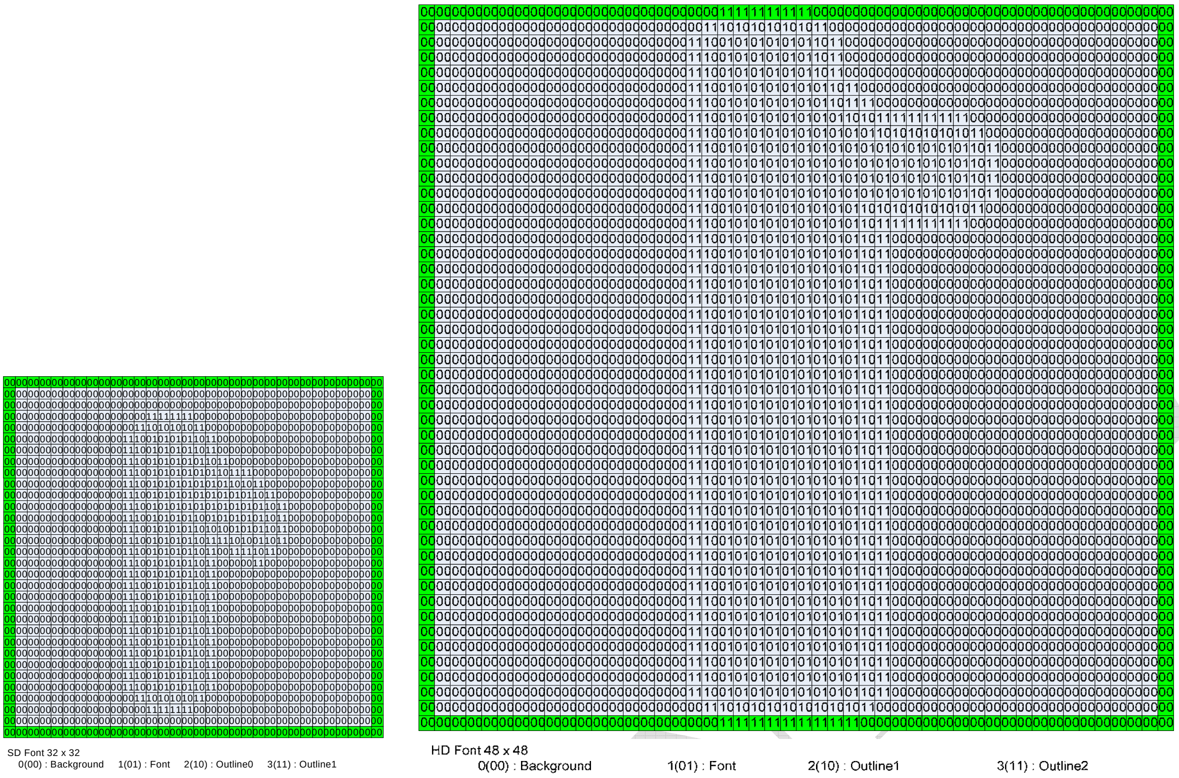
Fig. 4.59 SD OSD & HD OSD Font Size

Font data are expressed in 2 bits. As shown in the diagram below, 0 represents the background while 1 is the font and 2 & 3 are the outline of the font. The outline closer to the background is 3 whereas the one closer to the font is 2. As shown in Fig.4.61, font data should have its right and left sides converted.



	0	1	2	3	4	5	6
7	8	9	A	B	C	D	E
F	G	H	I	J	K	L	M
N	O	P	Q	R	S	T	U
V	W	X	Y	Z	▶	→	←
↑	↓	(	)	‾	–	_	■
/	=	&	:	~	,	.	↵
····	|···	·|··	··|·	···|	x	.	+

Fig. 4.60 SD OSD ROM Font Data


SD Font 32 x 32
0(00) : Background 1(01) : Font 2(10) : Outline0 3(11) : Outline1
HD Font 48 x 48
0(00) : Background 1(01) : Font 2(10) : Outline1 3(11) : Outline2

(a) SD OSD font

(b) HD OSD font

Fig. 4.61 Composition of OSD Font Data

4.19.2.1. CONSTRUCTION OF THE DISPLAY RAM

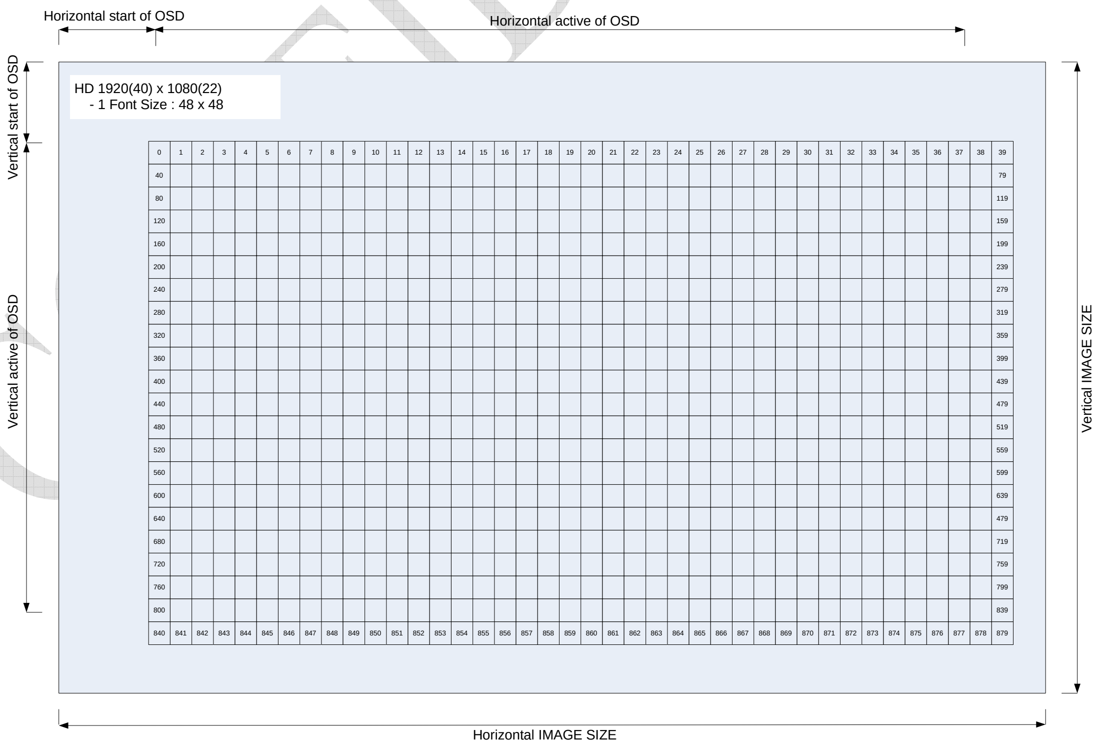


Fig. 4.62 Example of HD OSD Font Grid

Display RAM is the memory (RAM) that can determine the type of fonts, its colors and attributes on the screen. In case of the HD OSD, up to 880 fonts can be displayed on a single screen while SD OSD can have 540 fonts displayed on a screen. (The size of the actual use depends on the setting of the horizontal and vertical grid.) Each font can be expressed on the screen by selecting the font index and attribute of 16 bits for it.

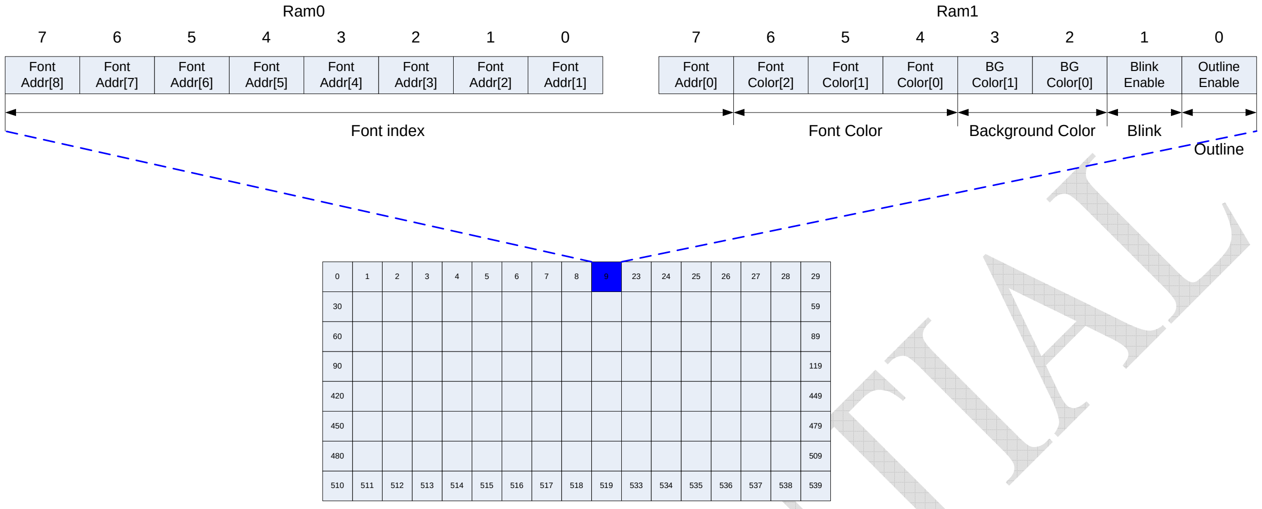

Ram0
7 6 5 4 3 2 1 0
Font Addr[8] | Font Addr[7] | Font Addr[6] | Font Addr[5] | Font Addr[4] | Font Addr[3] | Font Addr[2] | Font Addr[1]
Ram1
7 6 5 4 3 2 1 0
Font Addr[0] | Font Color[2] | Font Color[1] | Font Color[0] | BG Color[1] | BG Color[0] | Blink Enable | Outline Enable
Font index
Font Color
Background Color
Blink
Outline
0 1 2 3 4 5 6 7 8 23 24 25 26 27 28 29
30 59
60 89
90 119
420 449
450 479
480 509
510 511 512 513 514 515 516 517 518 519 533 534 535 536 537 538 539

Fig. 4.63 Composition of OSD Display RAM

OSD's display RAM can be accessed with the read/write XSFR only in the indirect address access approach. (However, a different approach is used when reading on the SD OSD.)

In addition to the function to control attribute for each font, OSD has other functions to control the global attribute of all the fonts such as color control (mono), color inversion (inversion), transparency control (transparency), reference table of the font (LUT), function to control outline Y values, blink time control function, etc.)

4.20. CVBS P2I & DOWN SCALER

4.20.1. BLOCK DIAGRAM

NVP2400 performs down scaling and progressive to interlace conversion for CVBS output. Herein, HPF filter and cropping functions can be independently applied to the CVBS output only.

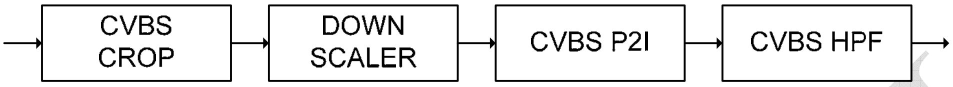


Fig. 4.64. CVBS P2I & Down Scaler

4.20.2. CVBS HPF

NVP2400 has an edge enhance block, which is used to control the edge of the CVBS output. This Horizontal Edge enhancer called CVBS HPF has an edge crop function, positive and negative edge gain control function and it can control the edge enhancement level.

4.20.3. CVBS CROP

NVP2400 has a cropping block for CVBS. CVBS CROP functions by setting the start position along with the horizontal size and vertical size. For example, CVBS cropping enables to output only some part of the image out of the full-HD size image, which realizes the E-PTZ function for CVBS output without damaging resolution. Moving the start position for crop, Pen/Tilt operation can be easily done.

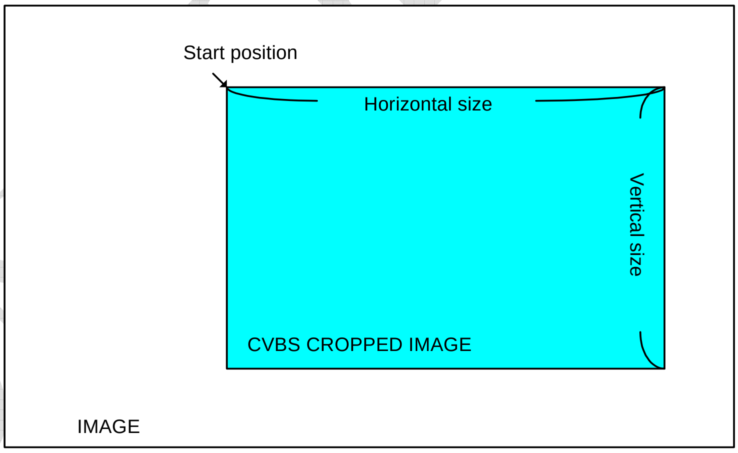


Fig. 4.65. CVBS CROP

4.20.4. CVBS DOWN SCALER

For CVBS output, the images received from the CVBS CROP block should be downscaled in accordance with the CVBS output image size (720*480, 960*480, 720*576 or 960*576). CVBS DOWN SCALER performs this function of downsizing the input image for the sake of output.

4.20.5. CVBS P2I

NVP2400 has CVBS P2I block in order to support the CVBS video. The CVBS P2I block downscales video data received from the sensor or outputs cropped image in the CVBS video format. CVBS P2I block separates the image entered in the progressive approach into the odd lines and even lines. Using the odd lines or even lines of the input image, it generates images in the interlaced approach.

4.21. CVBS VIDEO ENCODER (VE)

4.21.1. OVERVIEW

NVP2400 has an independent video encoder that generates standard NTSC and PAL video signals. Video encoder receives digital video signals and generates analog video signals. The video encoder consists of Pre-processing, Y/C low-pass filter, Modulation part and Output mux. The video encoder works on 27 or 36 MHz.

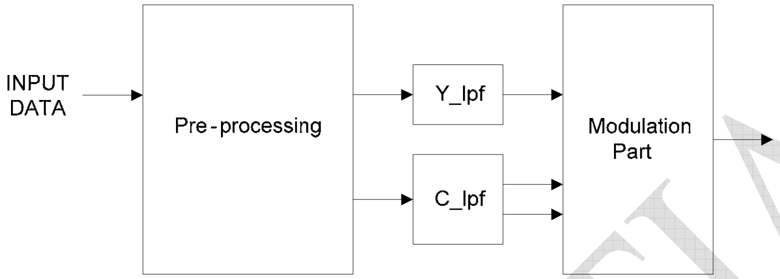


Fig. 4.66 Timing Diagram of P2I Read Channel

4.21.2. DESCRIPTION

4.21.2.1. PRE-PROCESSING

Pre-processing block has three functions. First, it creates timing signals (Timing Generator). Second, it converts Y, Cb, Cr into Y, U, V (Color Space Conversion). Third, it controls the scale (Gain Control). The timing generator generates the sync signals out of the Horizontal and Vertical Blank as well as various timing signals. As for the color space conversion, the following formulas are used to convert Y, Cb, Cr into Y, U, and V.

Standards	Y	U	V
NTSC	0.591 (Y601 – 64)	0.504 (Cb – 512)	0.711 (Cr – 512)
PAL	0.625 (Y601 – 64)	0.533 (Cb – 512)	0.752 (Cr – 512)

As for the gain control, Sync, Burst, Luma and Chroma can be adjusted in the scale from 0 to about 2. The increment in scale is 1/127. Fig. 4.67 shows adjustments for each control.

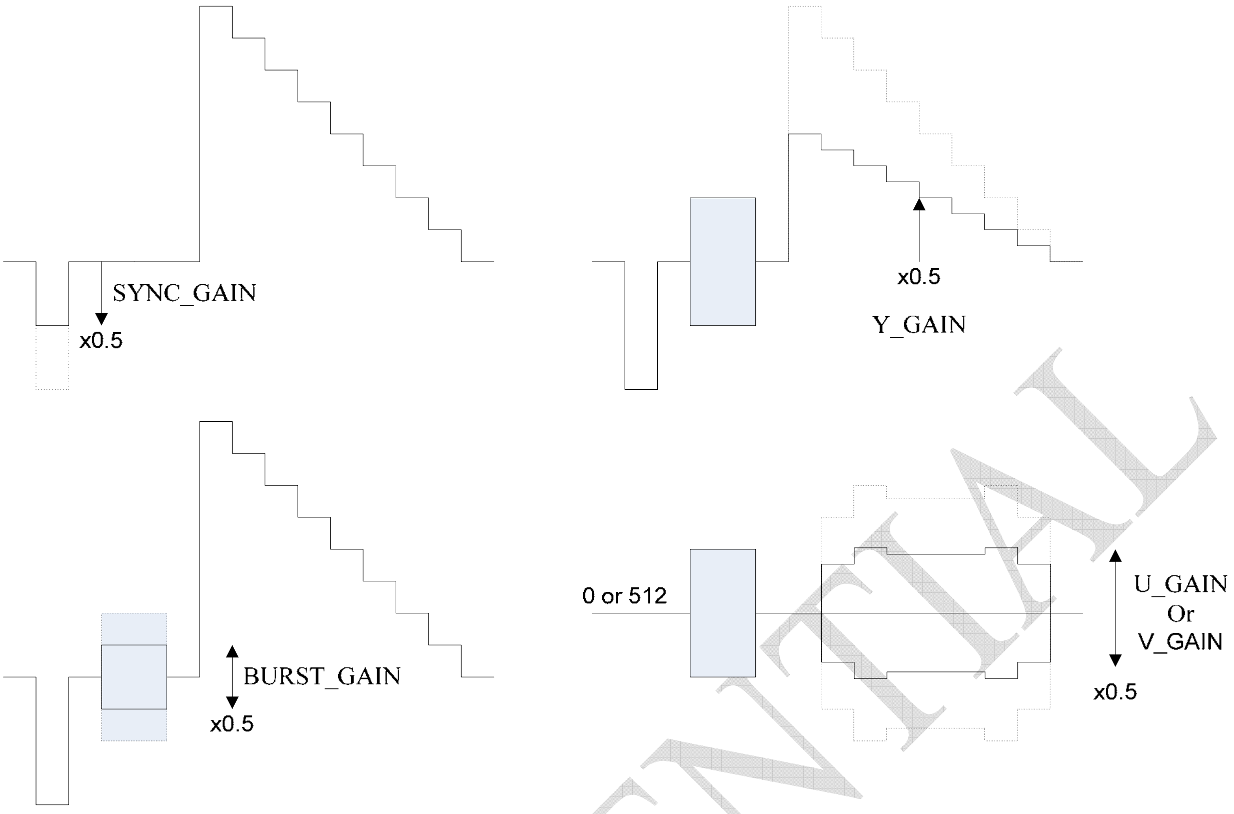


Fig. 4.67 Gain Control for Burst, Sync, Luma, Chroma

4.21.2.2. Y/C LOW-PASS FILTER

Low-pass filtering at 6 MHz is done is order to remove high frequency components that happen when Y is over-sampled. To eliminate aliases that may occur after modulation, low-pass filters of 0.6 MHz, 1 MHz, and 1.3 MHz are used in accordance with the frequency bandwidths of U and V

4.21.2.3. MODULATION PART

The modulation block generates chroma signals through AM modulation. AM modulation is applied to NTSC and PAL. Fig. 4.68 shows chroma formed by way of AM modulation for NTSC and PAL.

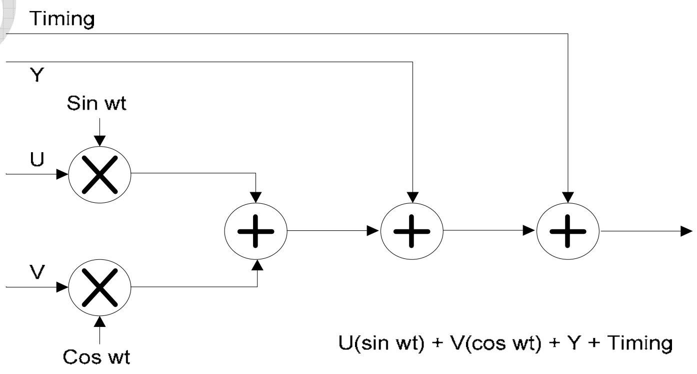


Fig. 4.68 AM Modulation for NTSC and PAL

4.22. IRIS

IRIS circuit controls the lens in a way that it can maintain a certain level of light. It operates the motor which receives the open / close signal for the lens. NVP2400 supports DC IRIS and Video IRIS. Video IRIS outputs the signal of the current brightness (intensity of radiation) and DC IRIS outputs open / close pulse as shown in Fig4.70.

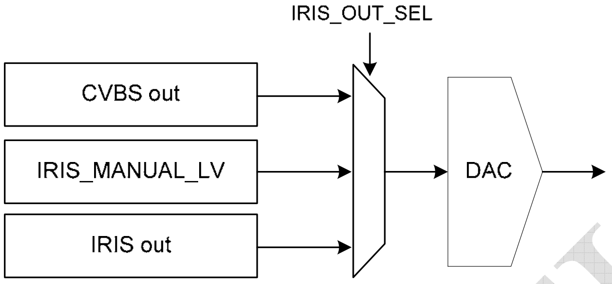


Fig. 4.69 IRIS Diagram

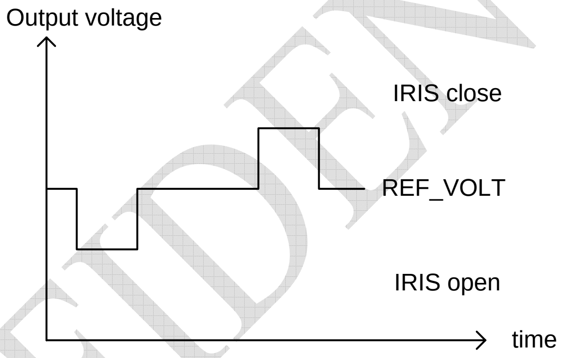


Fig. 4.70 DC IRIS Output

4.23. OUTPUT FORMATTER

4.23.1. OVERVIEW

Output formatter generates the final output of YC 16Bit of HD OSD as it is or converts it into other format such as BT.656, BT.1120. As the standard output format, YC 16Bit comes out along with the control sync and data whereas BT.656 or BT.1120 comes out in the format of the embedded syncs including control signals. However, optionally BT.656 and BT.1120 control signals can be generated for the seperated syncs output format. It also supports the function to change the polarity of the signal and to crop the output test pattern and output data.

4.23.2. BLOCK DIAGRAM

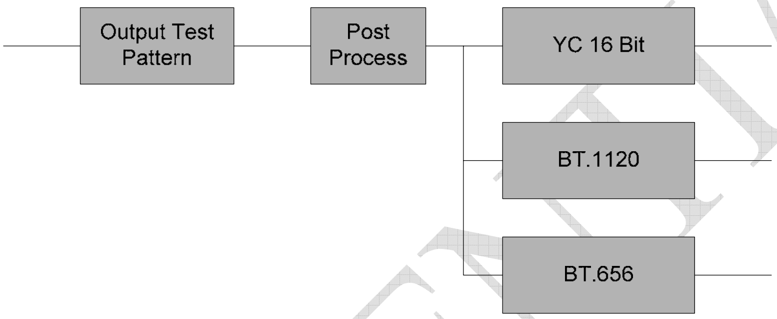


Fig. 4.71 Output Formatter Block Diagram

The final output is determined out of YC 16Bit, BT.656, BT.1120 based on the GPIO setting.

4.23.3. DESCRIPTION

Output formatter consists of output test pattern, the post processor that supports cropping function and format converter.

Output test pattern generator contains 18 basic YC patterns including color bar, ramp, or latch in order to determine whether the final output of the internal function is accurate or to test other chips that receive final output. In the output test pattern, control signals of the final output can be used as they are or control signals can be created at the sync generation block of the test pattern. Sync generation block should be set at one of the interlace / progressive mode depending on the signals for output.

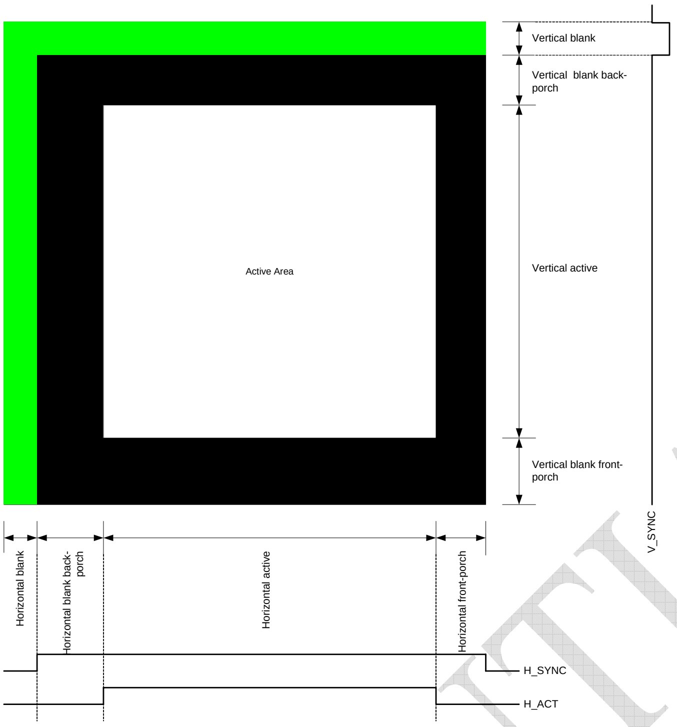


Fig. 4.72 Progressive Sync-generation

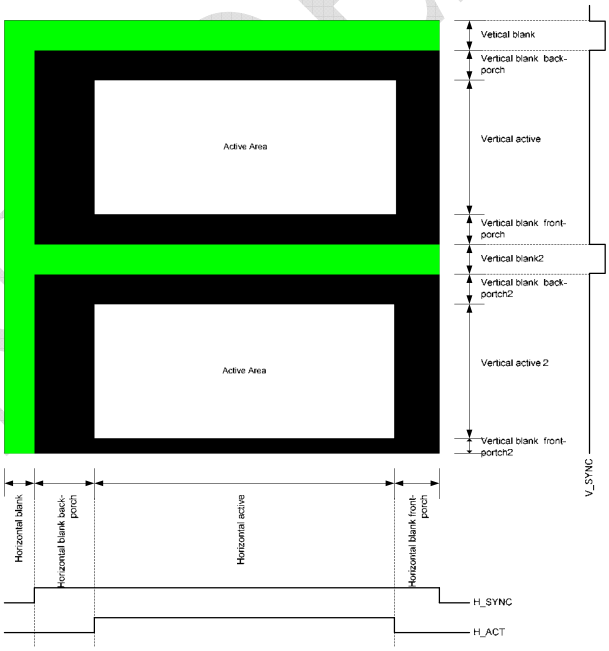

Vetical blank
Vertical blank back-porch
Vertical active
Vertical blank front-porch
Vertical blank2
Vertical blank back-portch2
Vertical active 2
Vertical blank front-portch2
V_SYNC
Active Area
Active Area
Horizontal blank
Horizontal blank back-porch
Horizontal active
Horizontal blank front-porch
H_SYNC
H_ACT

Fig. 4.73 Interlace Sync-generation

Basically, 18 test patterns are provided as shown in Fig. 4.74 and user defined colors can be generated.

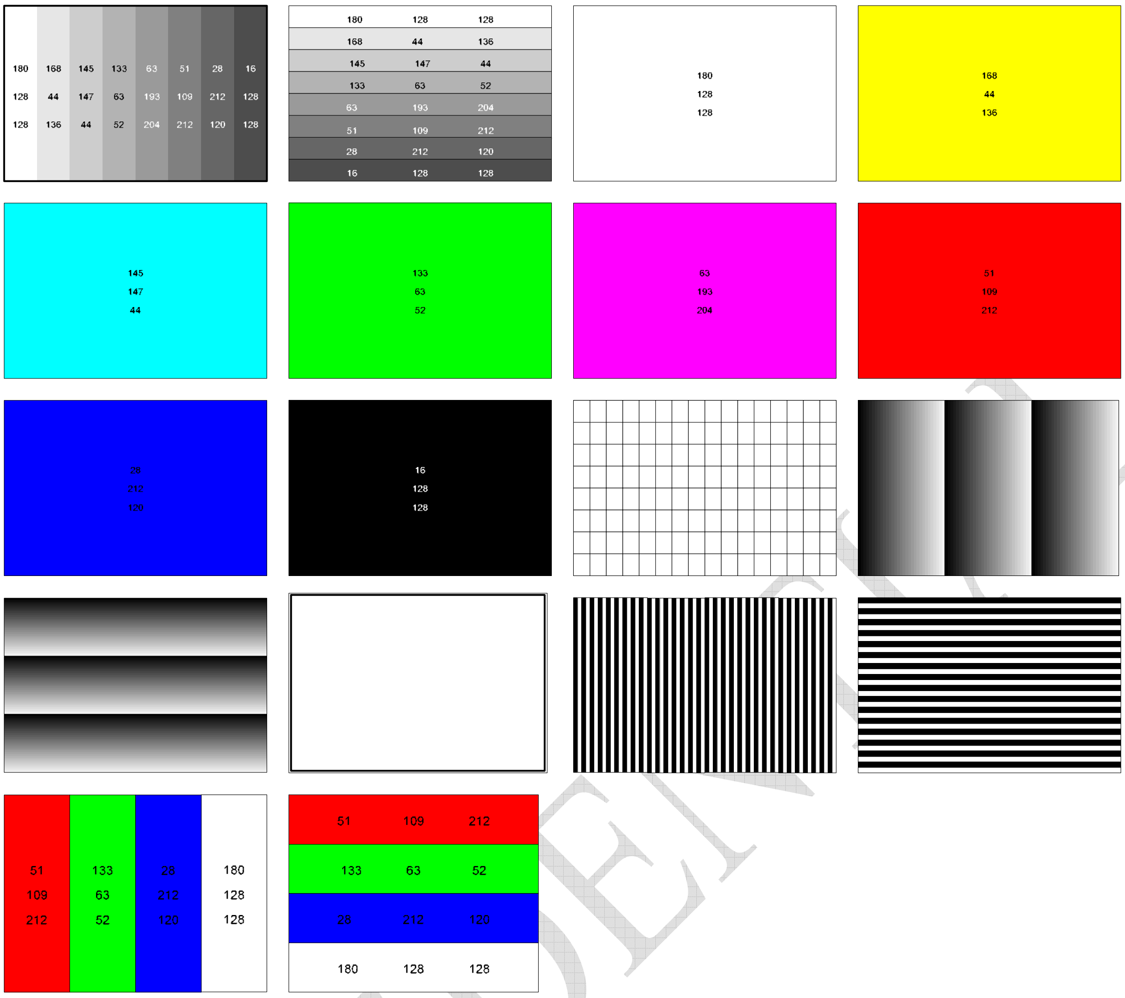


Fig. 4.74 Output Test Patterns

Post processor supports the YC swap function that can converts the sequence of the YC values into C,Y or Y,C. It also supports C swap function to change the sequence of Cb and Cr. It also has the function to restrict the output Y to 16 ~ 235 and C to 16 ~ 240 along with the cropping function such as INPUT CROP or CVBS CROP. Background-color converts the output data into the BG_Y, BG_CB, BG_CR as determined by the user. The final output is YC 16 bits or BT.656 or BT.1120.

4.23.3.1. Y/C 16 BIT OUTPUT

Sync polarity of YC 16bit output mode can be modified.

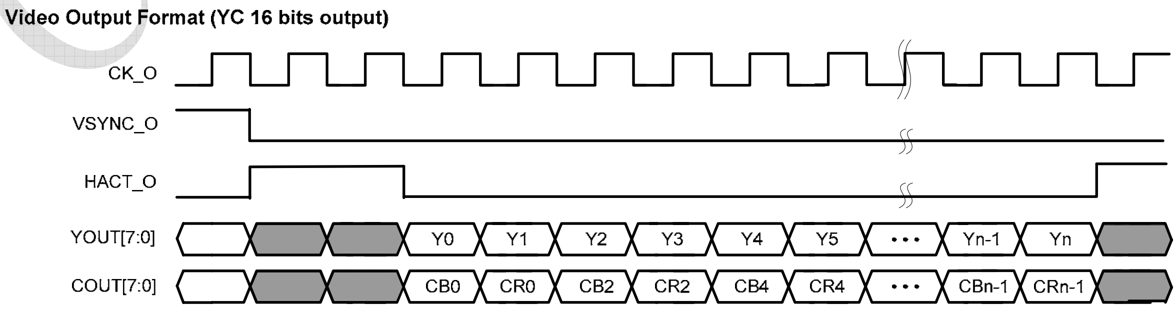


Fig. 4.75 Y/C 16bits Output Timing Diagram

4.23.3.2. BT.656

BT.656 output supports BT.656 as seen in Fig. 4.76. However, if necessary, the horizontal and vertical signals (H/Vsync) can be separately sent out through separate balls.

Video Output Format (BT.656)

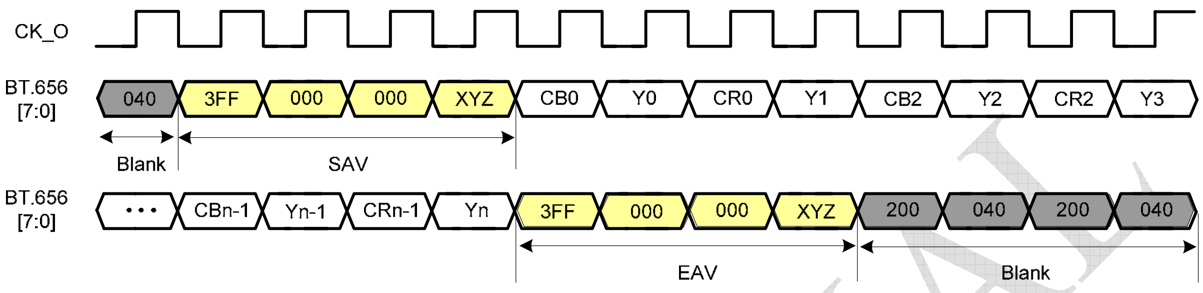


Fig. 4.76 BT.656 Timing Diagram

4.23.3.3. BT.1120

BT.1120 output supports Standard BT.1120 output as shown in Fig. 4.77. However, it can send out H/Vsync output through separate balls.

Video Output Format (BT.1120 Parellel Interface)

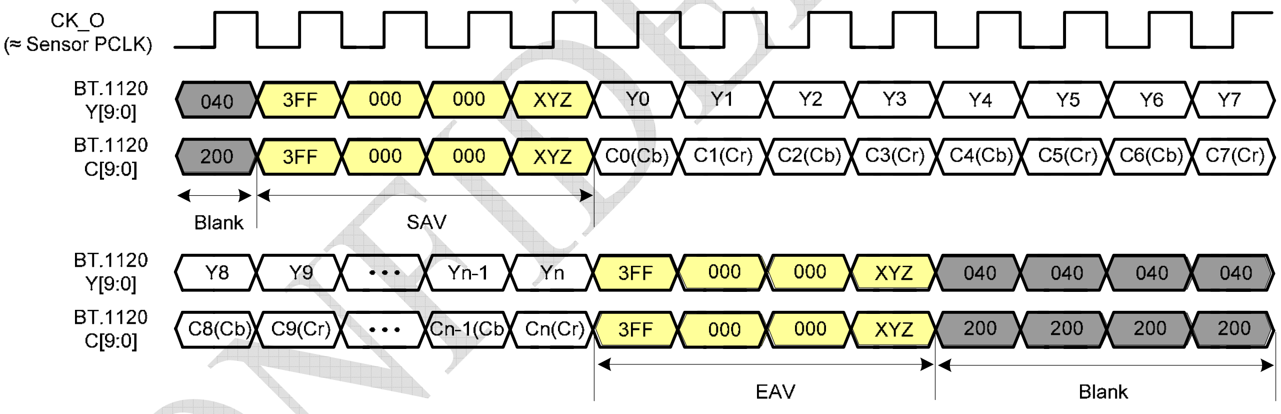


Fig. 4.77 BT.1120 Timing Diagram

5. CLOCK SYSTEM

5.1. CLOCK SYSTEM

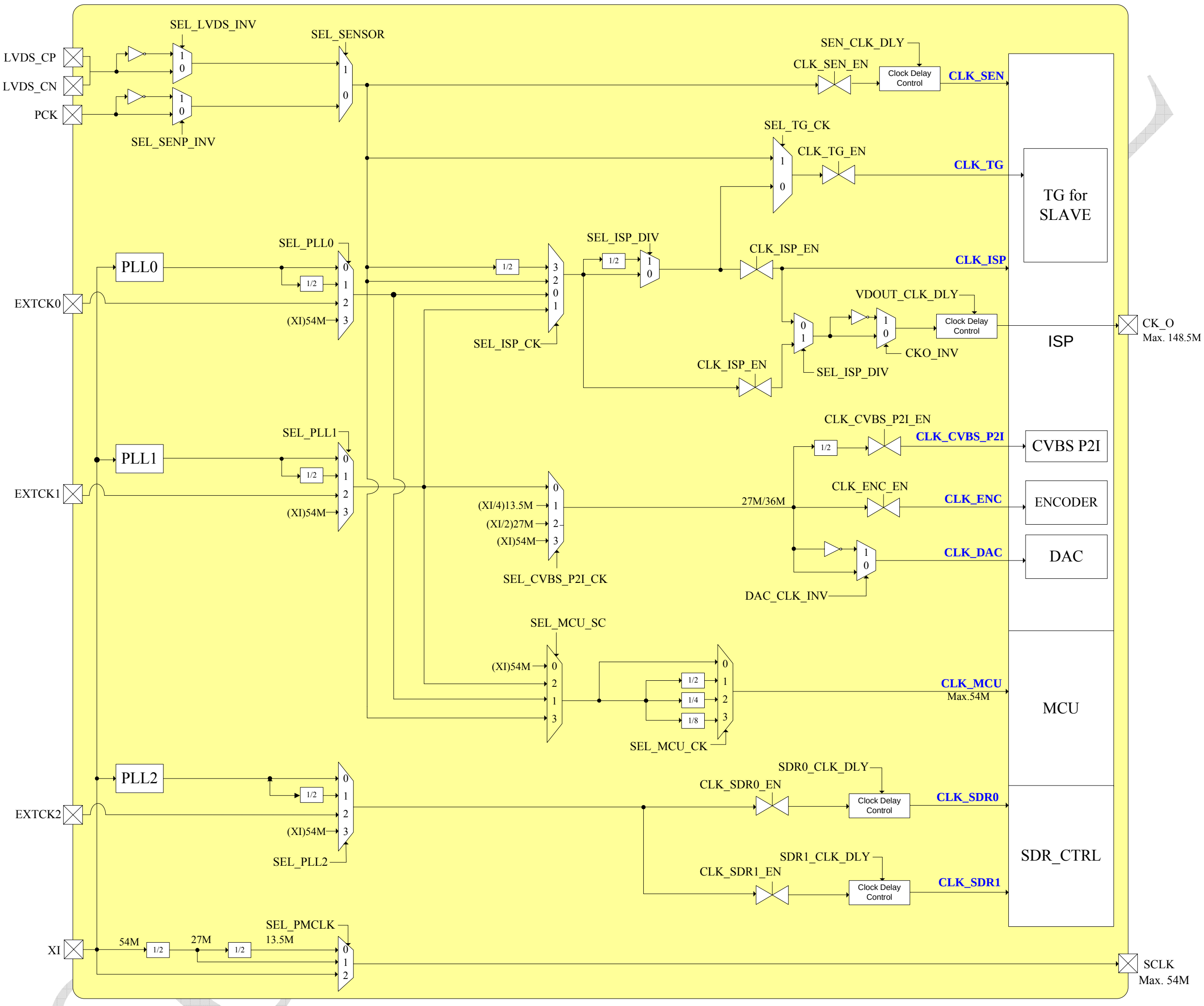


Fig. 5.1 Clock System (When XI is 54Mhz)

NVP2400 has a total of three PLL. Among these, PLL0 is mainly used as the ISP main clock (CLK_ISP) while PLL1 is used for CVBS output clock (CLK_CVBS_P2I, CLK_ENC, CLK_DAC) and PLL2 as the clock exclusively for the frame memory. PLL0 or PLL2 can be used for different purposes depending on the situation for which the clock system needs to be set properly. Table 5.1 below shows a representative clock mux setting value in accordance with NVP2400's target application. There could be many other ways for setting. SEL_PMCLK, which is not shown among the clock muxs on Table 5.1 is a register where the clock to supply for the sensor is to be selected. It can be sent to the SCLK BALL.

Table 5.1 Clock Mux Table (When Oscillator is 54Mhz)

TARGET APPLICATION					CLOCK MUX										PLL Frequency		
SENSOR MODE (MASTER or SLAVE)	SENSOR IF (LVDS or PARALLEL)	MCU FREQ. (Mhz) (Max. 54Mhz)	OUTPUT TYPE	CVBS TYPE	SEL_SENSOR	SEL_PLL0	SEL_ISP_CK	SEL_ISP_DIV	SEL_TG_CK	SEL_PLL1	SEL_CVBS_P2I_CK	SEL_PLL2	SEL_MCU_SC	SEL_MCU_CK	PLL0 (Mhz)	PLL1 (Mhz)	PLL2 (Mhz)
M	L	54	BT.1120	720H	1	0	0	0	X	X	2	0	0	0	74.25	X	133
M	L	54	BT.1120	960H	1	0	0	0	X	0	0	0	0	0	74.25	36	133
M	L	54	BT.656	720H	1	0	0	1	X	X	2	0	0	0	148.5	X	133
M	L	54	BT.656	960H	1	0	0	1	X	0	0	0	0	0	148.5	36	133
M	L	54	YC 16bits	720H	1	0	*A	0	X	X	2	0	0	0	*A	X	133
M	L	54	YC 16bits	960H	1	0	*A	0	X	0	0	0	0	0	*A	36	133
M	L	*C	BT.1120	720H	1	0	0	0	X	X	2	0	2	0	74.25	*C	133
M	L	*C	BT.1120	960H	*B												
M	L	*C	BT.656	720H	1	0	0	1	X	X	2	0	2	0	148.5	*C	133
M	L	*C	BT.656	960H	*B												
M	L	*C	YC 16bits	720H	1	0	*A	0	X	X	2	0	2	0	*A	*C	133
M	L	*C	YC 16bits	960H	1	0	2, 3	0	X	0	0	0	1	0	*C	36	133
M	P	54	BT.1120	720H	0	0	0	0	X	X	2	0	0	0	74.25	X	133
M	P	54	BT.1120	960H	0	0	0	0	X	0	0	0	0	0	74.25	36	133
M	P	54	BT.656	720H	0	0	0	1	X	X	2	0	0	0	148.5	X	133
M	P	54	BT.656	960H	0	0	0	1	X	0	0	0	0	0	148.5	36	133
M	P	54	YC 16bits	720H	0	0	*A	0	X	X	2	0	0	0	*A	X	133
M	P	54	YC 16bits	960H	0	0	*A	0	X	0	0	0	0	0	*A	36	133
M	P	*C	BT.1120	720H	0	0	0	0	X	X	2	0	2	0	74.25	*C	133
M	P	*C	BT.1120	960H	*B												
M	P	*C	BT.656	720H	0	0	0	1	X	X	2	0	2	0	148.5	*C	133
M	P	*C	BT.656	960H	*B												
M	P	*C	YC 16bits	720H	0	0	*A	0	X	X	2	0	2	0	*A	*C	133
M	P	*C	YC 16bits	960H	0	0	2	0	X	0	0	0	1	0	*C	36	133
S	L	54	BT.1120	720H	1	0	0	0	0, 1	X	2	0	0	0	74.25	X	133
S	L	54	BT.1120	960H	1	0	0	0	0, 1	0	0	0	0	0	74.25	36	133
S	L	54	BT.656	720H	1	0	0	1	0, 1	X	2	0	0	0	148.5	X	133
S	L	54	BT.656	960H	1	0	0	1	0, 1	0	0	0	0	0	148.5	36	133
S	L	54	YC 16bits	720H	1	0	*A	0	0, 1	X	2	0	0	0	*A	X	133
S	L	54	YC 16bits	960H	1	0	*A	0	0, 1	0	0	0	0	0	*A	36	133
S	L	*C	BT.1120	720H	1	0	0	0	0, 1	X	2	0	2	0	74.25	*C	133
S	L	*C	BT.1120	960H	*B												
S	L	*C	BT.656	720H	1	0	0	1	0, 1	X	2	0	2	0	148.5	*C	133
S	L	*C	BT.656	960H	*B												
S	L	*C	YC 16bits	720H	1	0	*A	0	0, 1	X	2	0	2	0	*A	*C	133
S	L	*C	YC 16bits	960H	1	0	2, 3	0	0, 1	0	0	0	1	0	*C	36	133
S	P	54	BT.1120	720H	0	0	0	0	0, 1	X	2	0	0	0	74.25	X	133
S	P	54	BT.1120	960H	0	0	0	0	0, 1	0	0	0	0	0	74.25	36	133
S	P	54	BT.656	720H	0	0	0	1	0, 1	X	2	0	0	0	148.5	X	133

S	P	54	BT.656	960H	0	0	0	1	0, 1	0	0	0	0	0	148.5	36	133
S	P	54	YC 16bits	720H	0	0	*A	0	0, 1	X	2	0	0	0	*A	X	133
S	P	54	YC 16bits	960H	0	0	*A	0	0, 1	0	0	0	0	0	*A	36	133
S	P	*C	BT.1120	720H	0	0	0	0	0, 1	X	2	0	2	0	74.25	*C	133
S	P	*C	BT.1120	960H	*B												
S	P	*C	BT.656	720H	0	0	0	1	0, 1	X	2	0	2	0	*A	*C	133
S	P	*C	BT.656	960H	*B												
S	P	*C	YC 16bits	720H	0	0	*A	0	0, 1	X	2	0	2	0	*A	*C	133
S	P	*C	YC 16bits	960H	0	0	2	0	0, 1	0	0	0	1	0	*C	36	133

NOTE :

*A : In case of YC 16bits output, CK_O can be any frequency depending on the data rate or back-end codec chip.

*B : Normally, it cannot be used but some application can be used in special conditions.

*C : MCU can use PLL output clock if PLL is not used for any clock.

Table 5.2 shows the register involved in the operation of the Clock System.

Table 5.2 Clock System Register

Address	Register Name	Bit	R/W	Description
0x0010	SEL_PLL_XIN	7:6	R/W	Select PLL input clock 0 : XI 1 : XI/2 2 : XI/4 3 : reserved
	SEL_PLL2	5:4	R/W	Output mux of PLL2 0 : PLL2 output clock 1 : PLL2 output clock / 2 2 : EXT2 3 : XI
	SEL_PLL1	3:2	R/W	Output mux of PLL1 0 : PLL1 output clock 1 : PLL1 output clock / 2 2 : EXT1 3 : XI
	SEL_PLL0	1:0	R/W	Output mux of PLL0 0 : PLL0 output clock 1 : PLL0 output clock / 2 2 : EXT0 3 : XI
0x0011	SEL_ISP_DIV	5	R/W	1 : divide ISP CLK
	SEL_SENSOR	4	R/W	Select sensor type 0 : Parallel sensor 1 : serial sensor
	SEL_SENP_INV	3	R/W	Parallel sensor clock insertion 0 : Bypass 1 : Inversion

Address	Register Name	Bit	R/W	Description
	CKO_INV	1	R/W	output clock Inversion 0 : Bypass 1 : Inversion
	SEL_LVDS_INV	0	R/W	Serial sensor clock insertion 0 : Bypass 1 : Inversion
0x0012	SEL_PMCLK	5:4	R/W	Select Sensor input clock 0 : XI 1 : XI/2 2 : XI/4 3 : PLL0 output / 2
	SEL_TGCLK	3	R/W	0 : CLK_SEN 1 : CLK_ISP
0x0013	SEL_CVBS_P2I_CK	7:6	R/W	Select CVBS_P2I clock 0: Output mux of PLL1 1: XI / 4 2: XI / 2 3 : XI
	SEL_MCU_CK	5:4	R/W	Select MCU clock 0: MCU source clock 1: MCU source clock / 2 2: MCU source clock / 4 3: MCU source clock / 8
	SEL_MCU_SC	3:2	R/W	Select MCU source clock 0: XI 1: CLK_SEN (sensor clock) 2: Output mux of PLL1 3: Output mux of PLL0
	SEL_ISP_CK	1:0	R/W	Select ISP Data-path main clock 0:Output mux of PLL0 1: Output mux of PLL1 2: CLK_SEN(sensor clock)
0x0014	CLK_SEN_EN	7	R/W	0:Disable, 1:Enable
	CLK_ISP_EN	6	R/W	0:Disable, 1:Enable
	CLK_TG_EN	4	R/W	0:Disable, 1:Enable
	CLK_CVBS_P2I_EN	3	R/W	0:Disable, 1:Enable
	CLK_ENC_EN	2	R/W	0:Disable, 1:Enable
	CLK_SDR0_EN	1	R/W	0:Disable, 1:Enable
	CLK_SDR1_EN	0	R/W	0:Disable, 1:Enable
0x0015	SEN_CLK_DLY	5:0	R/W	CLK_SEN clock delay
0x0016	SDR0_CLK_DLY	5:0	R/W	SDR0_CLK clock delay
0x0017	SDR1_CLK_DLY	5:0	R/W	SDR1_CLK clock delay
0x0018	VDOUT_CLK_DLY	4:0	R/W	CLK_OUT clock delay
0x0030	DAC_CLK_INV	4	R/W	1: INVERT DAC_CLK(CLK_ENC)

5.2. PLL

The NVP2400 has 3 PLL blocks. Each PLL can generate stable high-speed clock from a slower clock signal. The output frequency is adjustable and can be up to 1000MHz. This PLL integrates a Phase Frequency Detector (PFD), a Low Pass Filter (LPF), a Voltage Controlled Oscillator (VCO) and other associated circuit. All fundamental building blocks as well as fully programmable dividers are integrated in the core.

5.2.1. THE EQUATION OF PLL

The PLL output frequency, PLL_OUT, is determined by the ratio set between the value set in the input divider and the feedback divider. PLL output frequency PLL_OUT is calculated using the following equations:

$$\text{PLL\_OUT} = (\text{XIN} * \text{M}) / (\text{N} * 2^{\text{S}})$$

[Attention]
[1] 1MHz ≤ XIN / N ≤ 25Mhz
[2] 200MHz ≤ VCO(= XIN*M / N) ≤ 1000MHz
[3] 25MHz ≤ PLL_OUT ≤ 1000MHz
[4] M ≥ 2 and N ≥ 2

The table below describes PLL control registers.

Table 5.3 PLL Control Special Function Registers

Name	Add.	R/W	Description	Reset
PLL_MDIV0	0xB1	R/W	PLL0 M(Feedback diver) value	0x63
PLL_NSDIV0	0xB2	R/W	PLL0 N(Input divider) value for PLL_NSDIV0[5:4]	0x3
PLL_NSDIV0	0xB2	R/W	PLL0 S(output divider) value for PLL_NSDIV0[3:0]	0x9
PLL_MDIV1	0xB3	R/W	PLL1 M(Feedback diver) value	0x20
PLL_NSDIV1	0xB4	R/W	PLL1 N(Input divider) value for PLL_NSDIV1[5:4]	0x3
PLL_NSDIV1	0xB4	R/W	PLL1 S(output divider) value for PLL_NSDIV1[3:0]	0x6
PLL_MDIV2	0xB5	R/W	PLL2 M(Feedback diver) value	0x45
PLL_NSDIV2	0xB6	R/W	PLL2 N(Input divider) value for PLL_NSDIV2[5:4]	0x2
PLL_NSDIV2	0xB6	R/W	PLL2 S(output divider) value for PLL_NSDIV2[3:0]	0x7

5.2.2. RECOMMENDED VALUES OF PLL

Recommended PLL setting values are described in Table 5.4 with the oscillator input clock set at 54MHz and also described in Table 5.5 with the oscillator input clock set at 27MHz.

Table 5.4 Recommended Values of PLL (When Oscillator Clock is 54Mhz)

CLK_OUT	M	N	S
21.375	19	6	3
22.500	20	6	3
23.625	21	6	3
24.750	22	6	3
25.875	23	6	3
27.000	24	6	3
28.125	25	6	3

CLK_OUT	M	N	S
100.500	134	9	3
101.250	135	9	3
102.000	136	9	3
102.750	137	9	3
103.500	138	9	3
104.250	139	9	3
105.000	140	9	3

29.250	26	6	3
30.375	27	6	3
31.500	28	6	3
32.625	29	6	3
33.750	30	6	3
34.875	31	6	3
36.000	32	6	3
37.125	33	6	3
36.643	38	7	3
71.719	85	8	3
72.563	86	8	3
73.406	87	8	3
74.250	99	9	3
75.000	100	9	3
75.750	101	9	3
76.500	102	9	3
77.250	103	9	3
78.000	104	9	3
78.750	105	9	3
79.500	106	9	3
80.250	107	9	3
81.000	108	9	3
81.750	109	9	3
82.500	110	9	3

105.750	141	9	3
106.500	142	9	3
107.250	143	9	3
108.000	144	9	3
125.357	65	7	2
127.286	66	7	2
129.214	67	7	2
131.143	68	7	2
133.071	69	7	2
133.3125	79	8	2
135	80	8	2
136.6875	81	8	2
116.4375	69	8	2
118.125	70	8	2
119.8125	71	8	2
135	80	8	2
136.6875	81	8	2
138.375	82	8	2
140.0625	83	8	2
141.75	84	8	2
143.4375	85	8	2
145.125	86	8	2
146.8125	87	8	2
148.5	88	8	2

Table 5.5 Recommended Values of PLL (When Oscillator Clock is 27Mhz)

CLK_OUT	M	N	S
21.9375	39	6	3
22.5	40	6	3
23.0625	41	6	3
23.625	42	6	3
24.1875	43	6	3
24.75	44	6	3
25.3125	45	6	3
25.875	46	6	3
26.4375	47	6	3
27	48	6	3
27.5625	49	6	3
28.125	50	6	3
28.6875	51	6	3
29.25	52	6	3
29.8125	53	6	3
30.375	54	6	3
33.75	60	6	3
34.3125	61	6	3
34.875	62	6	3
35.4375	63	6	3

CLK_OUT	M	N	S
100.285714	104	7	2
101.25	105	7	2
102.214286	106	7	2
103.178571	107	7	2
104.142857	108	7	2
105.107143	109	7	2
106.071429	110	7	2
107.035714	111	7	2
108	112	7	2
125.357143	130	7	2
126.321429	131	7	2
127.285714	132	7	2
128.25	133	7	2
129.214286	134	7	2
130.178571	135	7	2
131.142857	136	7	2
132.107143	137	7	2
133.071429	138	7	2
138.857143	144	7	2
140.90625	167	8	2

36	64	6	3
36.5625	65	6	3
37.125	66	6	3
73.125	195	9	3
73.5	196	9	3
73.875	197	9	3
74.25	198	9	3
74.625	199	9	3
75	200	9	3

141.75	168	8	2
142.59375	169	8	2
143.4375	170	8	2
144.28125	171	8	2
145.125	172	8	2
145.96875	173	8	2
146.8125	174	8	2
147.65625	175	8	2
148.5	176	8	2

6. S/W RESET

In addition to the hardware reset by RSTN(P1) ball, NVP2400 is configured in a way that software can be reset for each block with the register setting by an internal MCU as shown in Table 6.1.

Table 6.1 Software Reset Register Map

Address	Register Name	Bit	R/W	Description	Reset
0x00DE	SWRESET_SEN	7	R/W	Software reset in the INPUT INTERFACE block. 0: normal operation, 1: reset	
	SWRESET_TG	6	R/W	Software reset in sensor slave tg block 0: normal operation, 1: reset	
	SWRESET_CVBS_P2I	5	R/W	Software reset from SD-OSD to VE 0: normal operation, 1: reset	
	SWRESET_ENC	4	R/W	Software reset in VE block 0: normal operation, 1: reset	
	SWRESET_SDR0	3	R/W	Software reset in SDRAM0 clock(CLK_SDR0) domain 0: normal operation, 1: reset	
	SWRESET_SDR1	2	R/W	Software reset in SDRAM1 clock(CLK_SDR1) domain 0: normal operation, 1: reset	
	SWRESETISP_PRE_PRO	1	R/W	Software reset from LSC to DPC and glue logic. 0: normal operation, 1: reset	
	SWRESET_ISP_NR	0	R/W	Software reset in 3D/2DNR block. 0: normal operation, 1: reset	
0x00DF	SWRESET_POST_PRO	3	R/W	Software reset in the blocks from CI to MD Mixer and glue logic 0:normal operation, 1: reset	
	SWRESET_ZOOM_PIP	2	R/W	Software reset in ZOOM/PIP block 0:normal operation, 1: reset	
	SWRESET_HDOUTPUT	1	R/W	Software reset in HD OSD and Output formatter block. 0: normal operation, 1: reset	
	SWRESET_SDOUTPUT	0	R/W	Software reset in Down scaler for CVBS and glue logic. 0: normal operation, 1: reset	

7. GPIO (GENERAL PURPOSE I/O)

GPIO can be used to interface with external devices and peripherals. These can act as input to read digital signals from other devices or output to send signals to other devices. NVP2400 has 76 GPIO ports and they are shared with other pins for peripherals (such as I2C, UART, SPI and so on) and image input/output port.

7.1. THE STRUCTURE OF GPIO REGISTERS

The table below shows the example of GPIO registers. GPIO00 Register contains the control registers of GPIO0 and GPIO1. Each GPIOn (n = 0,1,2,3, … , 75) is broken down into 2 parts (GPOn and GPMODEn). GPMODEn is PAD IO mode selection register. GPOn is the output value when the user selects GPO mode (GPMODEn = 3'b011).

Table 7.1 Examples of GPIO Registers

Register name : GPIO00							
7	6	5	4	3	2	1	0
GPO1	GPMODE1 Selection Register			GPO0	GPMODE0 Selection Register		

7.2. GPIO SFR

Table 7.2 GPIO SFR Registers

Addr.	Bit	Register Name	BALL NAME		GPIO MODE								
					Main Function		Second Function	Third Function		GPI Function		GPO Function	
					000		001	100		010		011	
System Service													
0x90	3:0	GPMODE00	EXTCK0	B	EXTCK0	I		MOTION_ALL	O	GPI0	I	GPO0	O
0x90	7:4	GPMODE00	EXTCK1	B	EXTCK1	I		MOTION0	O	GPI1	I	GPO1	O
0x91	3:0	GPMODE02	EXTCK2	B	EXTCK2	I		MOTION1	O	GPI2	I	GPO2	O
Parallel Sensor IF													
0x91	7:4	GPMODE02	PD0	B	PD[0]	I				GPI3	I	GPO3	O
0x92	3:0	GPMODE04	PD1	B	PD[1]	I				GPI4	I	GPO4	O
0x92	7:4	GPMODE04	PD2	B	PD[2]	I				GPI5	I	GPO5	O
0x93	3:0	GPMODE06	PD3	B	PD[3]	I				GPI6	I	GPO6	O
0x93	7:4	GPMODE06	PD4	B	PD[4]	I				GPI7	I	GPO7	O
0x94	3:0	GPMODE08	PD5	B	PD[5]	I				GPI8	I	GPO8	O
0x94	7:4	GPMODE08	PD6	B	PD[6]	I				GPI9	I	GPO9	O
0x95	3:0	GPMODE10	PD7	B	PD[7]	I				GPI10	I	GPO10	O
0x95	7:4	GPMODE10	PD8	B	PD[8]	I				GPI11	I	GPO11	O
0x96	3:0	GPMODE12	PD9	B	PD[9]	I				GPI12	I	GPO12	O
0x96	7:4	GPMODE12	PD10	B	PD[10]	I				GPI13	I	GPO13	O

0x97	3:0	GPMODE14	PD11	B	PD[11]	I					GPI14	I	GPO14	O
0x97	7:4	GPMODE14	PH	B	PH	I	HD_SLAVE	O			GPI15	I	GPO15	O
0x98	3:0	GPMODE16	PV	B	PV	I	VD_SLAVE	O			GPI16	I	GPO16	O
External Interrupt IF														
0x98	7:4	GPMODE16	EXTINT0	B	EXTINT0	I			MOTION2	O	GPI17	I	GPO17	O
0x99	3:0	GPMODE18	EXTINT1	B	EXTINT1	I			MOTION3	O	GPI18	I	GPO18	O
External Interface					SPI IF		I2C IF							
0x99	7:4	GPMODE18	SCK0	B	SCK0	B	SCL0	B			GPI19	I	GPO19	O
0x9A	3:0	GPMODE20	MISO0	B	MISO0	B	SDA0	B			GPI20	I	GPO20	O
0x9A	7:4	GPMODE20	MOSI0	B	MOSI0	B					GPI21	I	GPO21	O
0x9B	3:0	GPMODE22	CSN0	B	CSN0	B					GPI22	I	GPO22	O
0x9B	7:4	GPMODE22	SCK1_1	B	SCK1_1	B					GPI23	I	GPO23	O
0x9C	3:0	GPMODE24	MISO1_1	B	MISO1_1	B					GPI24	I	GPO24	O
0x9C	7:4	GPMODE24	MOSI1_1	B	MOSI1_1	B					GPI25	I	GPO25	O
0x9D	3:0	GPMODE26	CSN1_1	B	CSN1_1	B					GPI26	I	GPO26	O
0x9D	7:4	GPMODE26	SCK1_2	B	SCK1_2	B	SCL1	B			GPI27	I	GPO27	O
0x9E	3:0	GPMODE28	MISO1_2	B	MISO1_2	B	SDA1	B			GPI28	I	GPO28	O
0x9E	7:4	GPMODE28	MOSI1_2	B	MOSI1_2	B					GPI29	I	GPO29	O
0x9F	3:0	GPMODE30	CSN1_2	B	CSN1_2	B					GPI30	I	GPO30	O
UART IF														
0x9F	7:4	GPMODE30	UPCLK	B	UPCLK	I					GPI31	I	GPO31	O
0xA1	3:0	GPMODE32	UTX0	B	UTX0	O					GPI32	I	GPO32	O
0xA1	7:4	GPMODE32	URX0	B	URX0	I					GPI33	I	GPO33	O
0xA2	3:0	GPMODE34	UTX1	B	UTX1	O					GPI34	I	GPO34	O
0xA2	7:4	GPMODE34	URX1	B	URX1	I					GPI35	I	GPO35	O
TIMER IF														
0xA3	3:0	GPMODE36	TCLK	B	TCLK	I					GPI36	I	GPO36	O
0xA3	7:4	GPMODE36	TCAP0	B	TCAP0	I					GPI37	I	GPO37	O
0xA4	3:0	GPMODE38	TOUT0	B	TOUT0	O					GPI38	I	GPO38	O
0xA4	7:4	GPMODE38	TCAP1	B	TCAP1	I					GPI39	I	GPO39	O
0xA5	3:0	GPMODE40	TOUT1	B	TOUT1	O					GPI40	I	GPO40	O
0xA5	7:4	GPMODE40	TCAP2	B	TCAP2	I					GPI41	I	GPO41	O
0xA6	3:0	GPMODE42	TOUT2	B	TOUT2	O					GPI42	I	GPO42	O
DIGITAL OUTPUT					Y/C OUT		BT.1120 OUT		BT.656 OUT					
0xA6	7:4	GPMODE42	HACT_O	B	HACT_O	O	HACT_O	O	HACT_O	O	GPI43	I	GPO43	O
0xA7	3:0	GPMODE44	VSYNC_O	B	VSYNC_O	O	VSYNC_O	O	VSYNC_O	O	GPI44	I	GPO44	O
0xA7	7:4	GPMODE44	HSYNC_O	B	HSYNC_O	O					GPI45	I	GPO45	O
0xD1	3:0	GPMODE46	YOUT0	B	YOUT[0]	O	YOUT[0]	O	DATA[0]	O	GPI46	I	GPO46	O
0xD1	7:4	GPMODE46	YOUT1	B	YOUT[1]	O	YOUT[1]	O	DATA[1]	O	GPI47	I	GPO47	O
0xD2	3:0	GPMODE48	YOUT2	B	YOUT[2]	O	YOUT[2]	O	DATA[2]	O	GPI48	I	GPO48	O
0xD2	7:4	GPMODE48	YOUT3	B	YOUT[3]	O	YOUT[3]	O	DATA[3]	O	GPI49	I	GPO49	O
0xD3	3:0	GPMODE50	YOUT4	B	YOUT[4]	O	YOUT[4]	O	DATA[4]	O	GPI50	I	GPO50	O
0xD3	7:4	GPMODE50	YOUT5	B	YOUT[5]	O	YOUT[5]	O	DATA[5]	O	GPI51	I	GPO51	O
0xD4	3:0	GPMODE52	YOUT6	B	YOUT[6]	O	YOUT[6]	O	DATA[6]	O	GPI52	I	GPO52	O
0xD4	7:4	GPMODE52	YOUT7	B	YOUT[7]	O	YOUT[7]	O	DATA[7]	O	GPI53	I	GPO53	O
0xD5	3:0	GPMODE54	YOUT8	B			YOUT[8]	O			GPI54	I	GPO54	O
0xD5	7:4	GPMODE54	YOUT9	B			YOUT[9]	O			GPI55	I	GPO55	O

0xD6	3:0	GPMODE56	COUT0	B	COUT[0]	O	COUT[0]	O			GPI56	I	GPO56	O
0xD6	7:4	GPMODE56	COUT1	B	COUT[1]	O	COUT[1]	O			GPI57	I	GPO57	O
0xD7	3:0	GPMODE58	COUT2	B	COUT[2]	O	COUT[2]	O			GPI58	I	GPO58	O
0xD7	7:4	GPMODE58	COUT3	B	COUT[3]	O	COUT[3]	O			GPI59	I	GPO59	O
0xD8	3:0	GPMODE60	COUT4	B	COUT[4]	O	COUT[4]	O			GPI60	I	GPO60	O
0xD8	7:4	GPMODE60	COUT5	B	COUT[5]	O	COUT[5]	O			GPI61	I	GPO61	O
0xD9	3:0	GPMODE62	COUT6	B	COUT[6]	O	COUT[6]	O			GPI62	I	GPO62	O
0xD9	7:4	GPMODE62	COUT7	B	COUT[7]	O	COUT[7]	O			GPI63	I	GPO63	O
0xDA	3:0	GPMODE64	COUT8	B			COUT[8]	O			GPI64	I	GPO64	O
0xDA	7:4	GPMODE64	COUT9	B			COUT[9]	O			GPI65	I	GPO65	O
0xDB	3:0	GPMODE66	GPIO66	B							GPI66	I	GPO66	O
0xDB	7:4	GPMODE66	GPIO67	B	FIELD	O	FIELD	O	FIELD	O	GPI67	I	GPO67	O
Sensor IF														
0xDC	3:0	GPMODE68	SCLK	B	SCLK	O					GPI68	I	GPO68	O
0xDC	7:4	GPMODE68	SRESET	B							GPI69	I	GPO69	O
0xDF	3:0	GPMODE74	VD_SLAVE	B	VD_SLAVE	O					GPI74	I	GPO74	O
0xDF	7:4	GPMODE74	HD_SLAVE	B	HD_SLAVE	O					GPI75	I	GPO75	O

8. ELECTRICAL CHARACTERISTICS

8.1. DC CHARACTERISTICS

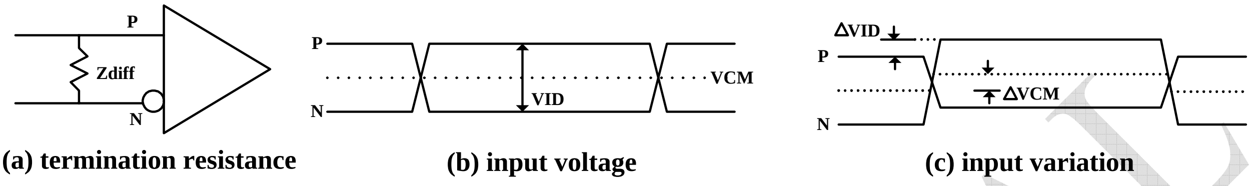


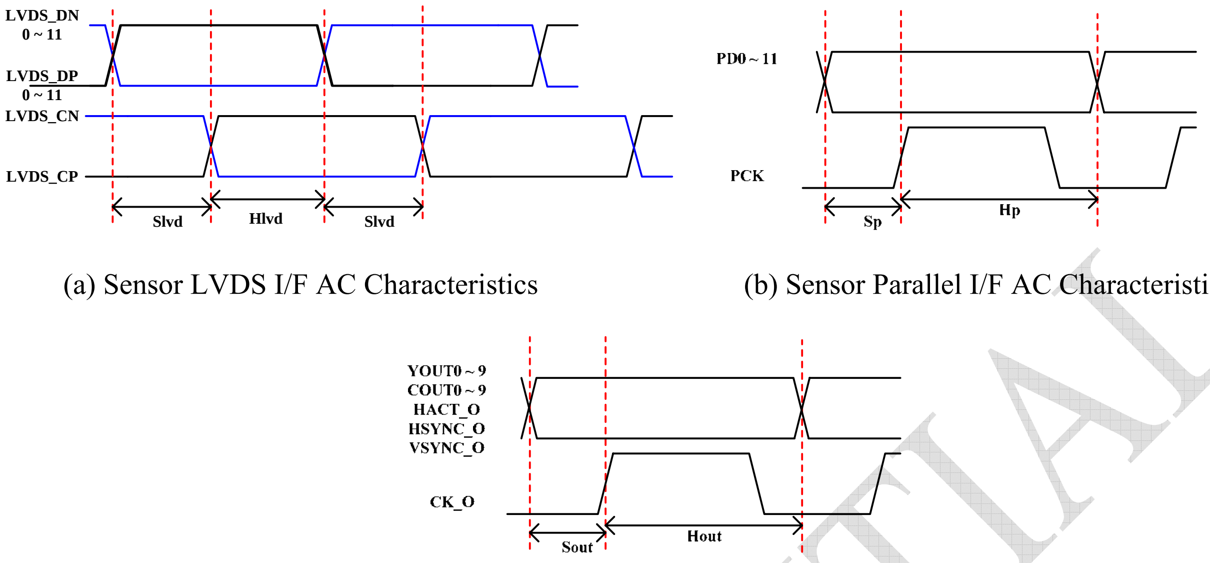
Fig. 8.1 LVDS I/F DC Characteristics

Table 8.1 NVP2400 DC Characteristics

Parameter	Symbol	Min.	Typ.	Max.	Unit	Remarks
LVDS Input common voltage	VCM	0.8	0.9	1	V	
LVDS Amplitude range of input differential voltage	VID	100	150		mV	
LVDS Terminal resistance value	Zdiff	98	100	102	Ohm	
LVDS VCM variation	ΔVCM	-25		25	mV	
LVDS VID variation	ΔVID	-25		25	mV	
LVDS post-driver analog power	PVDDDLVD33	3	3.3	3.6	V	
LVDS level shifter power	PVDDPLVD33	3	3.3	3.6	V	
LVDS pre-driver power	VDDLVD12	1.08	1.2	1.32	V	
DAC Operating voltage range	DAC_AVDD33C DAC_AVDD33I DAC_AVDD	2.97	3.3	3.63	V	
DAC Operating voltage range	DAC_PVDD33		3.3		V	
DAC Operating voltage range	DAC_DVDD	1.08	1.2	1.32	V	
DAC Max per channel output current			34.1		mA	
DAC Max output voltage			1.278		V	
DAC Integral non-linearity	INL		+/-1	+/-1.5	LSB	
DAC Differential non-linearity error	DNL		+/-0.5	+/-1	LSB	
PLL Operating voltage range	PLL0_AVDD33 PLL1_AVDD33 PLL2_AVDD33	2.97	3.3	3.63	V	Dedicated PLL Analog Power Supply
PLL PAD Power	PLL0_PVDD33 PLL1_PVDD33 PLL2_PVDD33	2.97	3.3	3.63	V	Dedicated PLL Pad Power Supply
PLL Operating voltage range	PLL0_DVDD12 PLL1_DVDD12 PLL2_DVDD12	1.08	1.2	1.32	V	Dedicated PLL Digital Power Supply

Digital Internal Power	VDDI	1.08	1.2	1.32	V	Internal Power
Digital Pad Power	VDDP33	2.97	3.3	3.63	V	3.3V I/O Power
1.8V Digital Power	VDDP18	1.62	1.8	1.98	V	
1.8V Core Power	VDDA_S18	1.62	1.8	1.98	V	
Sensor Interface Power	VDDP_18_33	1.62 / 2.97	1.8 / 3.3	1.98 / 3.63	V	Selectable Power (1.8V or 3.3V)
Input High Voltage for VDDP_18_33	VIH18_33	0.65* VDDP18_33		VDDP18_33 + 0.3	V	When VDDP18_33 is 1.8V
		2.0		VDDP18_33 + 0.3	V	When VDDP18_33 is 3.3V
Input Log Voltage for VDDP_18_33	VIL18_33	-0.3		0.35* VDDP18_33	V	When VDDP18_33 is 1.8V
		-0.3		0.8	V	When VDDP18_33 is 3.3V
Output High Voltage for VDDP_18_33	VOH18_33	VDDP18_33 - 0.45			V	When VDDP18_33 is 1.8V
		2.4			V	When VDDP18_33 is 3.3V
Output Low Voltage for VDDP_18_33	VOL18_33			0.45	V	When VDDP18_33 is 1.8V
				0.4	V	When VDDP18_33 is 3.3V
Input High Voltage for VDDP33	VIH33	1.7		5.5	V	
Input Log Voltage for VDDP33	VIL33	-0.3		0.7	V	
Output High Voltage for VDDP33	VOH33	2.4			V	
Output Low Voltage for VDDP33	VOL33			0.4	V	

8.2. AC CHARACTERISTICS



(a) Sensor LVDS I/F AC Characteristics

(b) Sensor Parallel I/F AC Characteristics

(c) Output Data AC Characteristics

Fig. 8.2 AC Characteristics

Table 8.2 NVP2400 AC Characteristics

Parameter	Symbol	Min	Typ	Max	Unit	Remarks
LVDS Input Clock Frequency	LVDS_CP/N			216	MHz	
LVDS Input Setup/Hold Time	Slvd / Hlvd		*1		ns	
Input Crystal oscillator Frequency	XI			54	MHz	
Parallel Sensor Input Clock Frequency	PCK			108	MHz	
Parallel IF Input Setup/Hold Time	Sp / Hp		*2		ns	
Output Clock Frequency	CK_O			148.5	MHz	
Duty cycle of Output Clock		45	50	55	%	Duty cycle of CK_O
Output Data Skew of YC output				2.61	ns	
Output Data Skew of BT.1120 output				2.16	ns	
Output Data Skew of BT.656 output				1.32	ns	
Output Setup/Hold Time	Sout/Hout		*3		ns	

Note

*1 : LVDS Input Setup/Hold Time can be controlled by SEL_LVDS_INV and SEN_CLK_DLY register.

*2 : Parallel I/F Input Setup/Hold Time can be controlled by SEL_SENP_INV and SEN_CLK_DLY register.

*3 : Output Data Setup/Hold Time can be controlled by VDOUT_CLK_DLY and CKO_INV register.

9. PACKAGE INFORMATION

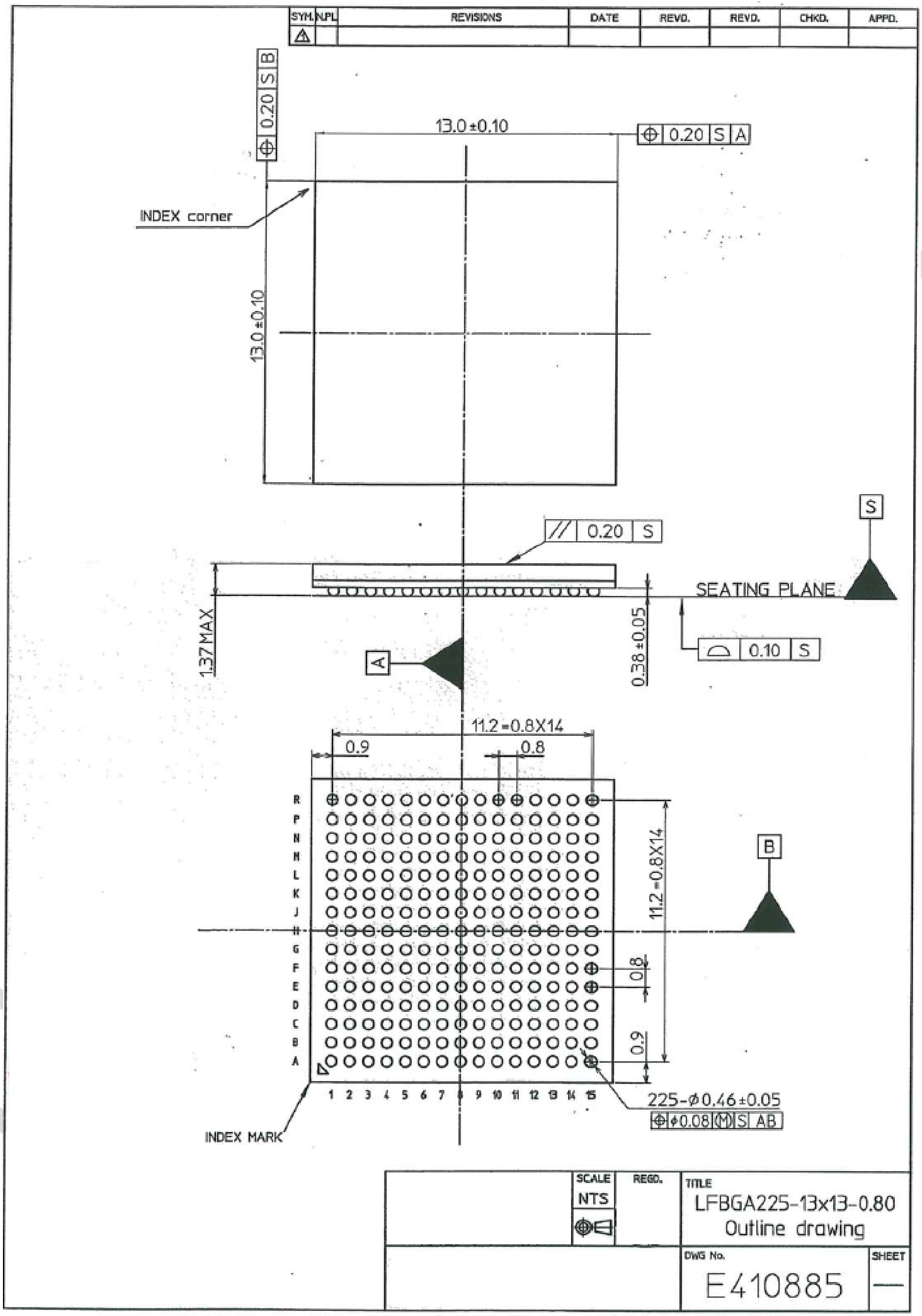


Fig. 9.1 225-FBGA-1313(0.8p) Package