Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

AP0100CS High-Dynamic Range (HDR) Image Signal Processor (ISP)

AP0100CS Datasheet, Rev. 6

For the latest product datasheet, please visit www.onsemi.com

Features

- Up to 1.2Mp (1280x960) ON Semiconductor sensor support
- 45 fps at 1.2Mp, 60 fps at 720p
- Optimized for operation with HDR sensors.
- Color and gamma correction
- Auto exposure, auto white balance, 50/60 Hz auto flicker detection and avoidance
- Adaptive Local Tone Mapping (ALTM)
- Programmable Spatial Transform Engine (STE).
- Pre-rendered Graphical Overlay
- Two-wire serial programming interface (CCIS)
- Interface to low-cost Flash or EEPROM through SPI bus (to configure and load patches, etc.)
- High-level host command interface
- Standalone operation supported
- Up to 5 GPIO
- Fail-safe IO
- Multi-Camera synchronization support
- Integrated video encoder for NTSC/PAL with overlay capability and 10-bit I-DAC

Applications

- IP cam and CCTV - HD
- Enables CCTV -HD w/ MP sensor

Table 1: Key Performance Parameters

Parameter	Value	
Primary camera interfaces	Parallel and HiSPi	
Primary camera input	RAW12 Linear/RAW12, RAW14 (HiSPi format only) Companded	
Output interface	Analog composite, up to 16-bit parallel digital output	
Output format	YUV422 8-bit,10-bit, and 10-, 12-bit tone-mapped Bayer	
Maximum resolution	1280x960 (1.2 Mp)	
NTSC output	720H x 487V	
PAL output	720H x 576V	
Input clock range	6-30 MHz	
Supply voltage	V_{DDIO_S}	1.8 or 2.8 V nominal
	V_{DDIO_H}	2.5 or 3.3 V nominal
	V_{DD_REG}	1.8 V nominal
	V_{DD}	1.2 V nominal
	V_{DD_PLL}	1.2 V nominal
	V_{DD_DAC}	1.2V nominal
	V_{DDIO_OTPM}	2.5 or 3.3 V nominal
	V_{DDA_DAC}	3.3 V nominal
	V_{DD_PHY}	2.8 V nominal
Operating temp.	–30°C to +70°C	
Power consumption	185 mW	

Notes: 1.

Ordering Information

Table 2: Available Part Numbers

Part Number	Product Description	Orderable Product Attribute Description
AP0100CS2L00SUGA0-DR1	1Mp Co-Processor, 100-ball VFBGA	Drypack
AP0100CS2L00SPGAD3-GEVK	AP0100CS Demo Kit	
AP0100CS2L00SPGAH-GEVB	AP0100CS Head Board	

See the ON Semiconductor Device Nomenclature document (TND310/D) for a full description of the naming convention used for image sensors. For reference documentation, including information on evaluation kits, please visit our web site at www.onsemi.com.

Table of Contents

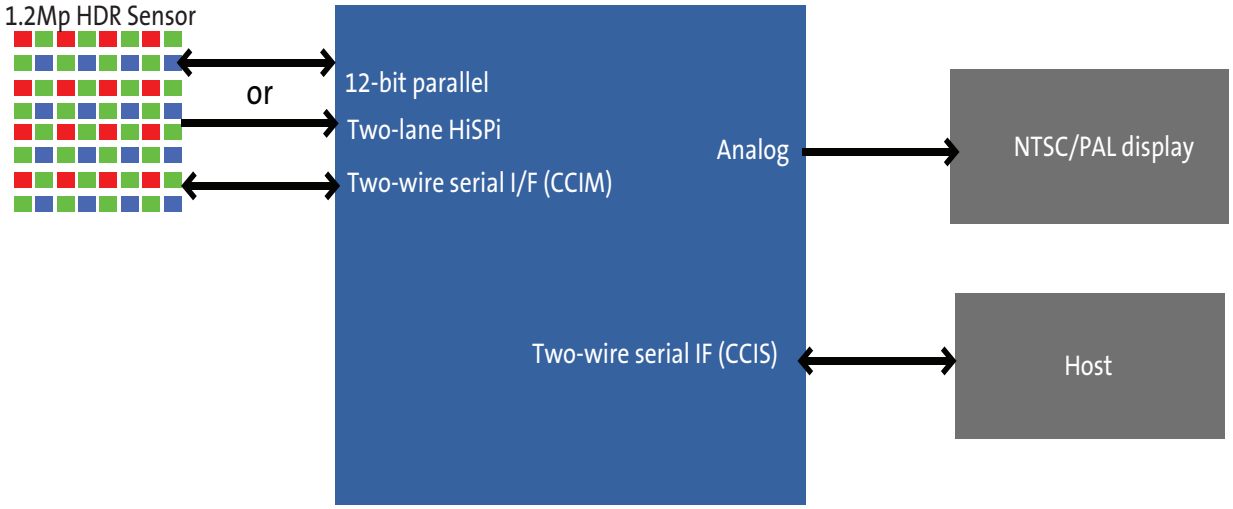
General Description

The ON Semiconductor AP0100CS is a high-performance, ultra-low power in-line, digital image processor optimized for use with HDR (High Dynamic Range) sensors. The AP0100CS provides full auto-functions support (AWB and AE) and ALTM (Adaptive Local Tone Mapping) to enhance HDR images and advanced noise reduction which enables excellent low-light performance.

Functional Overview

Figure 1 shows the typical configuration of the AP0100CS in a camera system. On the host side, a two-wire serial interface is used to control the operation of the AP0100CS, and image data is transferred using the analog or parallel interface between the AP0100CS and the host. The AP0100CS interface to the sensor also uses a parallel interface.

Figure 1: AP0100CS Connectivity



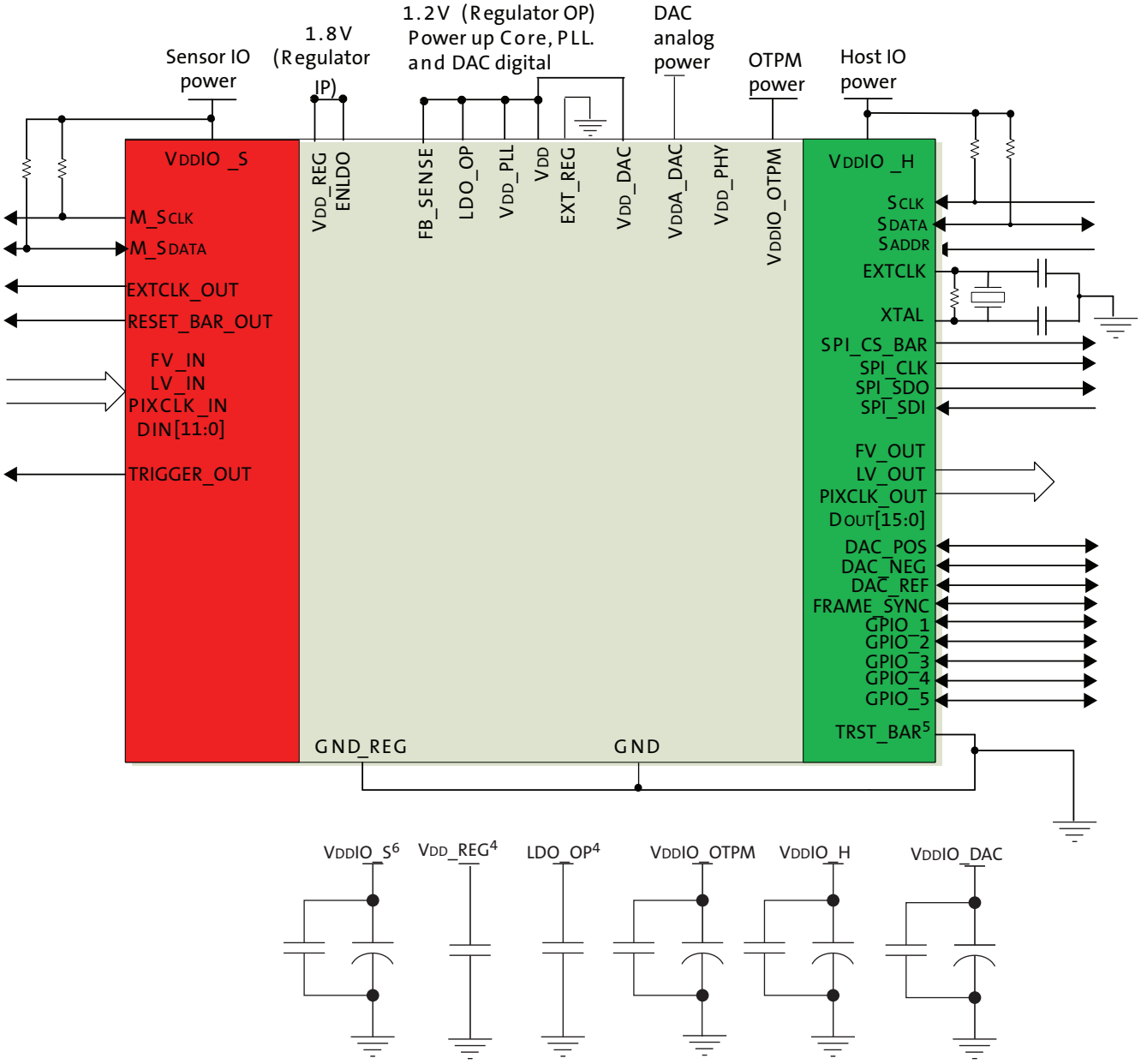
System Interfaces

Figure 2: “Typical Parallel Configuration,” on page 5 and Figure 3: “Typical HiSPi Configuration,” on page 6 show typical AP0100CS device connections.

All power supply rails must be decoupled from ground using capacitors as close as possible to the package.

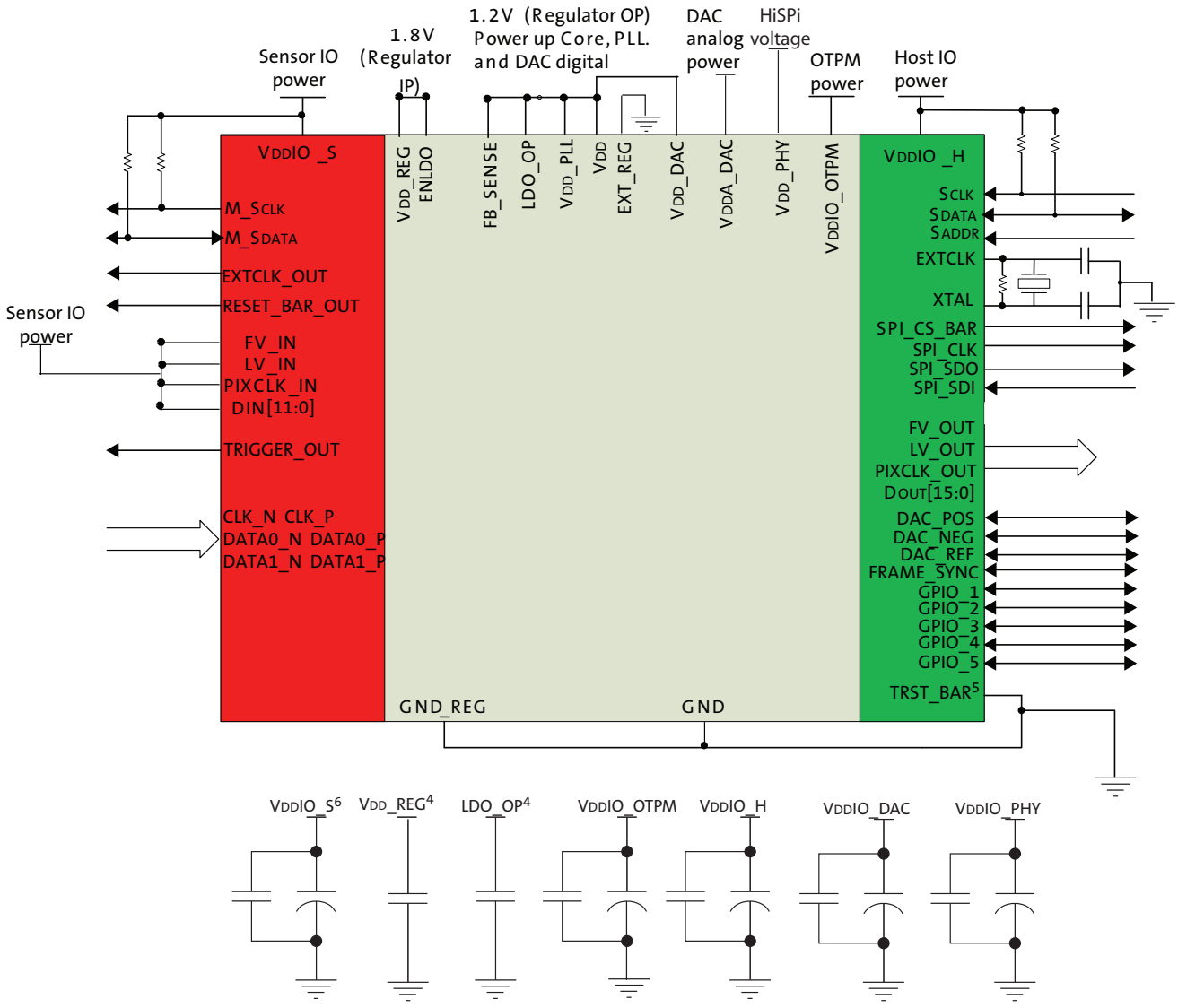
The AP0100CS signals to the sensor and host interfaces can be at different supply voltage levels to optimize power consumption and maximize flexibility. Table 1 on page 9 provides the signal descriptions for the AP0100CS.

Figure 2: Typical Parallel Configuration



Notes:
1. This typical configuration shows only one scenario out of multiple possible variations for this device.
2. ON Semiconductor recommends a 1.5kΩ resistor value for the two-wire serial interface $R_{PULL-UP}$; however, greater values may be used for slower transmission speed.
3. RESET_BAR has an internal pull-up resistor and can be left floating if not used.
4. The decoupling capacitors for the regulator input and output should have a value of 1.0uF. The capacitors should be ceramic and need to have X5R or X7R dielectric.
5. TRST_BAR connects to GND for normal operation.
6. ON Semiconductor recommends that 0.1µF and 1µF decoupling capacitors for each power supply are mounted as close as possible to the pin. Actual values and numbers may vary depending on lay-out and design consideration

Figure 3: Typical HiSPi Configuration



HiSPi and Parallel Connection

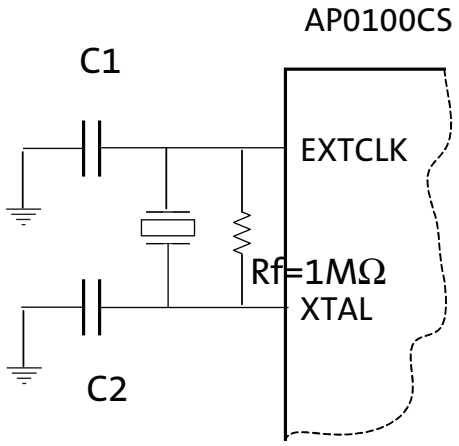
When using the HiSPi interface, the user should connect the parallel interface to VDDIO_S.

When using the parallel interface, the HiSPi interface and power supply (VDD_PHY) can be left floating.

Crystal Usage

As an alternative to using an external oscillator, a crystal may be connected between EXTCLK and XTAL. Two small loading capacitors and a feedback resistor should be added, as shown in Figure 4.

Figure 4: Using a Crystal Instead of an External Oscillator



Rf represents the feedback resistor, an Rf value of 1MΩ is sufficient for AP0100CS. C1 and C2 are decided according to the crystal or resonator CL specification. In the steady state of oscillation, CL is defined as (C1 x C2)/(C1+C2). In fact, the I/O ports, the bond pad, package pin and PCB traces all contribute the parasitic capacitance to C1 and C2. Therefore, CL can be rewritten to be (C1* x C2*)/(C1*+C2*), where C1*=(C1+Cin, stray) and C2*=(C2+Cout, stray). The stray capacitance for the IO ports, bond pad and package pin are known which means the formulas can be rewritten as C1*=(C1+1.5pF+Cin, PCB) and C2*=(C2+1.3pF+Cout, PCB).

Table 3: Pin Descriptions

Name	Type	Description
EXTCLK	Input	Master input clock. This can either be a square-wave generated from an oscillator (in which case the XTAL input must be left unconnected) or direct connection to a crystal.
XTAL	Output	If EXTCLK is connected to one pin of a crystal, the other pin of the crystal is connected to XTAL pin; otherwise this signal must be left unconnected.
RESET_BAR	Input/PU	Master reset signal, active LOW. This signal has an internal pull up.
SCLK	Input	Two-wire serial interface clock (host interface).
SDATA	I/O	Two-wire serial interface data (host interface).
SADDR	Input	Selects device address for the two-wire slave serial interface. When connected to GND the device ID is 0x90. When wired to VDDIO_H, a device ID of 0xBA is selected.
FRAME_SYNC	Input	This signal is used to synchronize to external sources or multiple cameras together. This signal should be connected to GND if not used.
STANDBY	Input	Standby mode control, active HIGH.
EXT_REG	Input	Select external regulator if tied high

Table 3: Pin Descriptions (Continued)

Name	Type	Description
ENDLO	Input	Regulator enable (VDD_REG domain)
SPI_SCLK	Output	Clock output for interfacing to an external SPI flash or EEPROM memory.
SPI_SDI	Input/PU	Data in from SPI flash or EEPROM memory. When no SPI device is fitted, this signal is used to determine whether the AP0100CS should auto-configure: 0: Do not auto-configure; Two-wire interface will be used to configure the device (host-config mode) 1: Auto-configure. This signal has an internal pull-up resistor.
SPI_SDO	Output	Data out to SPI flash or EEPROM memory.
SPI_CS_BAR	Output	Chip select out to SPI flash or EEPROM memory.
EXT_CLK_OUT	Output	Clock to external sensor.
RESET_BAR_OUT	Output	Reset signal to external signal.
M_SCLK	Output	Two-wire serial interface clock (Master).
M_SDATA	I/O	Two-wire serial interface clock (Master).
FV_IN	Input	Sensor frame valid input.
LV_IN	Input	Sensor line valid input.
PIXCLK_IN	Input	Sensor pixel clock input.
DIN[11:0]	Input	Sensor pixel data input DIN[11:0]
CLK_N	Input	Differential HiSPi clock (sub-LVDS, negative).
CLK_P	Input	Differential HiSPi clock (sub-LVDS, positive).
DATA0_N	Input	Differential HiSPi data, lane 0 (sub-LVDS, negative).
DATA0_P	Input	Differential HiSPi data, lane 0 (sub-LVDS, positive).
DATA1_N	Input	Differential HiSPi data, lane 1 (sub-LVDS, negative).
DATA1_P	Input	Differential HiSPi data, lane 1 (sub-LVDS, positive).
TRIGGER_OUT	Output	Trigger signal for external sensor.
FV_OUT	Output	Host frame valid output (synchronous to PIXCLK_OUT)
LV_OUT	Output	Host line valid output (synchronous to PIXCLK_OUT)
PIXCLK_OUT	Output	Host pixel clock output.
DOUT[15:0]	Output	Host pixel data output (synchronous to PIXCLK_OUT) DOUT[15:0].
DAC_POS	Output	Positive video DAC output in differential mode. Video DAC output in single-ended mode. This interface is enabled by default using NTSC/PAL signaling. For applications where composite video output is not required, the video DAC can be placed in a power-down state under software control.
DAC_NEG	Output	Negative video DAC output in differential mode.
DAC_REF	Output	External reference resistor for Video DAC.
GPIO [5:1]	I/O	General purpose digital I/O.
TRST_BAR	Input	Must be tied to GND in normal operation.
VDDIO_S	Supply	Sensor I/O power supply.
VDDIO_H	Supply	Host I/O power supply.
VDD_PLL	Supply	PLL supply.
VDD	Supply	Core supply.
VDDIO_OTPM	Supply	OTPM power supply.
VDD_DAC	Supply	Video DAC digital power
VDDA_DAC	Supply	Video DAC analog power
VDD_PHY	Supply	PHY IO voltage for HiSPi

Table 3: Pin Descriptions (Continued)

Name	Type	Description
GND	Supply	Ground
VDD_REG	Supply	Input to on-chip 1.8V to 1.2V regulator.
LDO_OP	Output	Output from on chip 1.8V to 1.2V regulator.
FB_SENSE	Output	On-chip regulator sense signal.

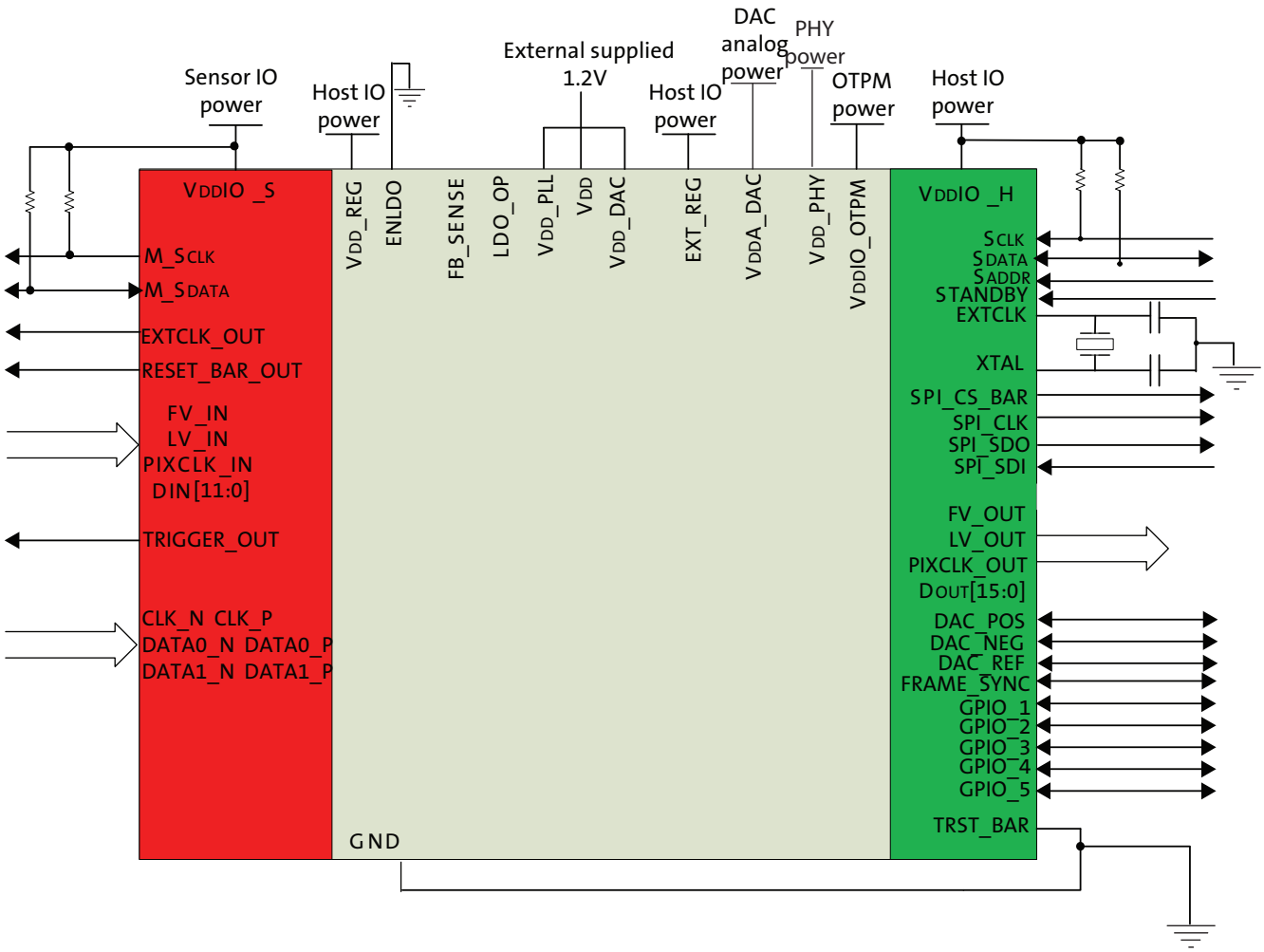
Table 4: Package Pinout

	1	2	3	4	5	6	7	8	9	10
A	DOUT[11]	DOUT[13]	PIXCLK_OUT	LV_OUT	GPIO_2	TRST_BAR	SPI_SDI	SADDR	SCLK	STANDBY
B	DOUT[12]	DOUT[10]	DOUT[14]	FV_OUT	GPIO_3	GPIO[5]	SPI_SCLK	SDATA	TRIGGER_OUT	RESET_BAR_OUT
C	DOUT[9]	DOUT[8]	DOUT[15]	GPIO[1]	GPIO_4	SPI_CS_BAR	SPI_SDO	VDDIO_H	M_SDATA	M_SCLK
D	DOUT[5]	DOUT[6]	DOUT[7]	VDDIO_H	VDDIO_HOST	VDD	FRAME_SYNC	VDD	FV_IN	MCLK_OUT
E	DOUT[2]	DOUT[3]	DOUT[4]	VDDIO_H	GND	GND	GND	LV_IN	PIXCLK_IN	DIN[11]
F	DOUT[0]	DOUT[1]	EXTCLK	VDDIO_H	GND	GND	GND	VDDIO_S	DIN[9]	DIN[10]
G	GND	VDD_PLL	XTAL	VDD	VDD	VDD	GND	DIN[6]	DIN[7]	DIN[8]
H	VDD_PLL	VDD_PLL	LDO_OUTPUT	VDDIO_OTPM	DAC_NEG	DAC_REF	GNDA_DAC	VDD_PHY	DIN[4]	DIN[5]
J	EXT_REG	RESET_BAR	VDD_REG	VDD_DAC	DAC_POS	DATA0_P	CLK_P	DATA1_N	DIN[0]	DIN[2]
K	GND	FB_SENSE	ENLDO	GND	VDDA_DAC	DATA0_N	CLK_N	DATA1_P	DIN[1]	DIN[3]

On-Chip Regulator

The AP0100CS has an on-chip regulator, the output from the regulator is 1.2 V and should only be used to power up the AP0100CS. It is possible to bypass the regulator and provide power to the relevant pins that need 1.2 V. Figure 5 shows how to configure the AP0100CS to bypass the internal regulator.

Figure 5: External Regulator



The following table summarizes the key signals when using/bypassing the regulator.

Table 5: Key Signals When Using the Regulator

Signal Name	Internal Regulator	External Regulator
VDD_REG	1.8 V	Connect to VDDIO_H
ENLDO	Connect to 1.8 V (VDD_REG)	GND
FB_SENSE	1.2 V (output)	Float
LDO_OP	1.2 V (output)	Float
EXT_REG	GND	Connect to VDDIO_H

Power-Up Sequence

Powering up the ISP requires voltages to be applied in a particular order, as seen in Figure 6. The timing requirements are shown in Table 6. The ISP includes a power-on reset feature that initiates a reset upon power up of the ISP.

Figure 6: Power-Up and Power-Down Sequence

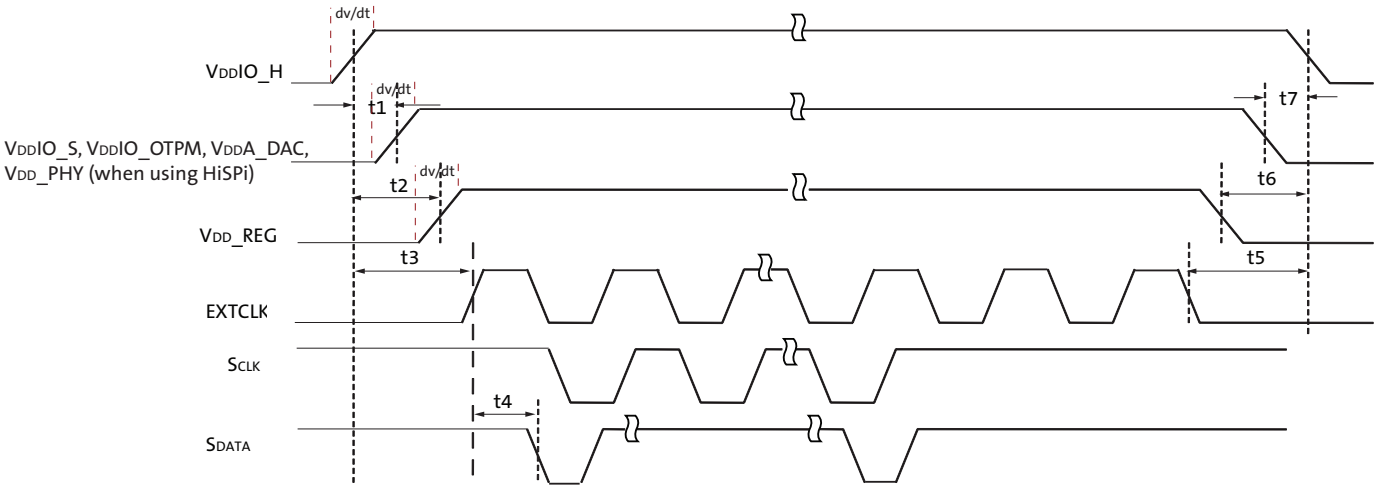


Table 6: Power-Up and Power-Down Signal Timing

Symbol	Parameter	Min	Typ	Max	Unit
t1	Delay from VDDIO_H to VDDIO_S, VDDIO_OTPM, VDDA_DAC, VDD_PHY (when using HiSPi)	0	–	50	ms
t2	Delay from VDDIO_H to VDD_REG	0	–	50	ms
t3	EXTCLK activation	t2 + 1	–	–	ms
t4	First serial command[1]	100	–	–	EXTCLK cycles
t5	EXTCLK cutoff	t6	–	–	ms
t6	Delay from VDD_REG to VDDIO_H	0	–	50	ms
t7	Delay from VDDIO_S, VDDIO_OTPM, VDDA_DAC, VDD_PHY (when using HiSPi) to VDDIO_H	0	–	50	ms
dv/dt	Power supply ramp time (slew rate)	–	–	0.1	V/µs

Note: 1. When using XTAL the settling time should be taken into account.

Reset

The AP0100CS has three types of reset available:

- A hard reset is issued by toggling the RESET_BAR signal
- A soft reset is issued by writing commands through the two-wire serial interface
- An internal power-on reset

Table 7 on page 13 shows the output states when the part is in various states.

Table 7: Output States

Name	Hardware States		Firmware States				Notes
	Reset State	Default State	Hard Standby	Soft Standby	Streaming	Idle	
EXTCLK	(clock running or stopped)	(clock running)	(clock running or stopped)	(clock running)	(clock running)	(clock running)	Input
XTAL	n/a	n/a	n/a	n/a	n/a	n/a	Input
RESET_BAR	(asserted)	(negated)	(negated)	(negated)	(negated)	(negated)	Input
SCLK	n/a	n/a	(clock running or stopped)	(clock running or stopped)	(clock running or stopped)	(clock running or stopped)	Input. Must always be driven to a valid logic level
SDATA	High-impedance	High-impedance	High-impedance	High-impedance			Input/Output. A valid logic level should be established by pull-up
SADDR	n/a	n/a	n/a	n/a	n/a	n/a	Input. Must always be driven to a valid logic level
FRAME_SYNC	n/a	n/a	n/a	n/a	n/a	n/a	Input. Must always be driven to a valid logic level
STANDBY	n/a	(negated)	(asserted)	(negated)	(negated)	(negated)	Input. Must always be driven to a valid logic level
EXT_REG	n/a	n/a	n/a	n/a	n/a	n/a	Input. Must always be driven to a valid logic level
ENLDO	n/a	n/a	n/a	n/a	n/a	n/a	Input. Must be tied to VDD_REG or GND
SPI_SCLK	High-impedance	driven, logic 0	driven, logic 0	driven, logic 0			Output
SPI_SDI	Internal pull-up enabled	Internal pull-up enabled	Internal pull-up enabled	internal pull-up enabled			Input. Internal pull-up permanently enabled.
SPI_SDO	High-impedance	driven, logic 0	driven, logic 0	driven, logic 0			Output
SPI_CS_BAR	High-impedance	driven, logic 1	driven, logic 1	driven, logic 1			Output
EXT_CLK_OUT	driven, logic 0	driven, logic 0	driven, logic 0	driven, logic 0			Output
RESET_BAR_OUT	driven, logic 0	driven, logic 0	driven, logic 1	driven, logic 1			Output. Firmware will release sensor reset
M_SCLK	High-impedance	High-impedance	High-impedance	High-impedance			Input/Output. A valid logic level should be established by pull-up
M_SDATA	High-impedance	High-impedance	High-impedance	High-impedance			Input/Output. A valid logic level should be established by pull-up
FV_IN ,LV_IN, PIXCLK_IN, DIN[11:0]	n/a	n/a	n/a	n/a	Dependent on interface used	n/a	Input. Must always be driven to a valid logic level

Table 7: Output States

Name	Hardware States		Firmware States				Notes
	Reset State	Default State	Hard Standby	Soft Standby	Streaming	Idle	
CLK_N	Disabled	Disabled	Dependent on interface used	Dependent on interface used	Dependent on interface used	Dependent on interface used	Input. Will be disabled and can be left floating
CLK_P							
DATA0_N							
DATA0_P							
DATA1_N							
DATA1_P							
FV_OUT, LV_OUT, PIXCLK_OUT, DOUT[15:0]	High-impedance	Varied	Driven if used	Driven if used	Driven if used	Driven if used	Output. Default state dependent on configuration
DAC_POS	Varied	Varied	Driven if used	Driven if used	Driven if used	Driven if used	Output. Default state dependent on configuration. Tie to ground if VDAC not used
DAC_NEG							
DAC_REF	n/a	n/a	n/a	n/a	n/a	n/a	Input. Requires reference resistor. Tie to ground if VDAC not used
GPIO[5:2]	High-impedance	Input, then high-impedance	Driven if used	Driven if used	Driven if used	Driven if used	Input/Output. After reset, these pins are sampled as inputs as part of auto-configuration.
GPIO1	High-impedance	High-impedance	High-impedance	High-impedance	High-impedance	High-impedance	
TRIGGER_OUT	High-impedance	High-impedance	Driven if used	Driven if used	Driven if used	Driven if used	
TRST_BAR	n/a	n/a	(negated)	(negated)	(negated)	(negated)	Input. Must always be driven to a valid logic level.

Hard Reset

The AP0100CS enters the reset state when the external RESET_BAR signal is asserted LOW, as shown in Figure 7. All the output signals will be in High-Z state.

Figure 7: Hard Reset Operation

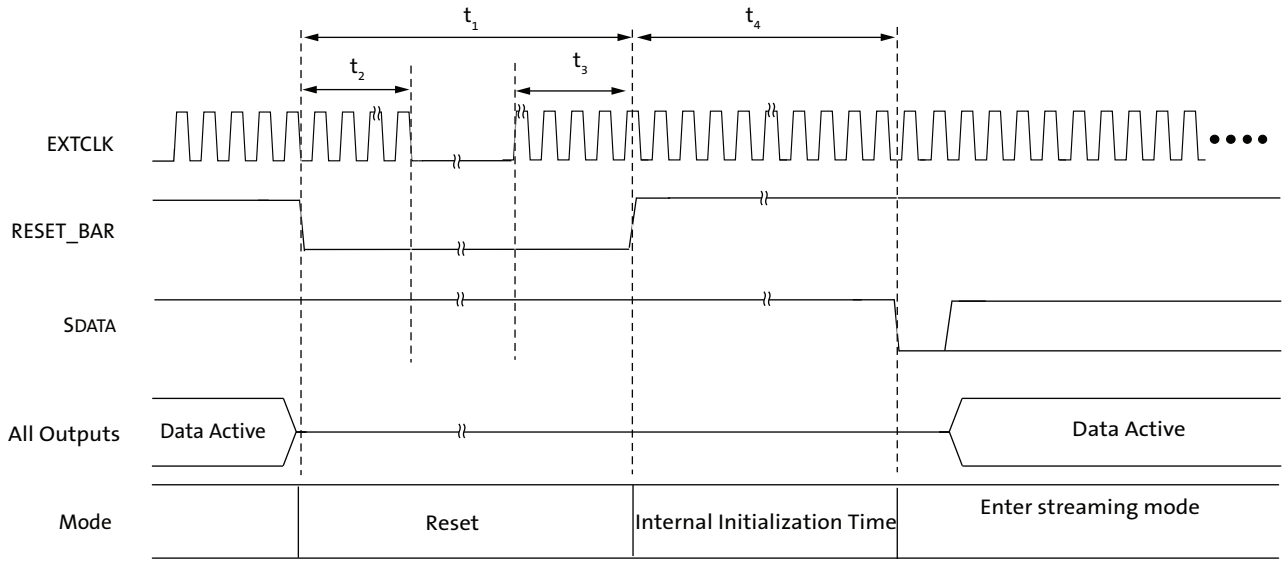


Table 8: Hard Reset

Symbol	Definition	Min	Typ	Max	Unit
t_1	RESET_BAR pulse width	50	–	–	EXTCLK cycles
t_2	Active EXTCLK required after RESET_BAR asserted	10	–	–	
t_3	Active EXTCLK required before RESET_BAR de-asserted	10	–	–	
t_4	First two-wire serial interface communication after RESET is HIGH	100	–	–	

Soft Reset

A soft reset sequence to the AP0100 CS can be activated by writing to a register through the two-wire serial interface.

Hard Standby Mode

The AP0100CS can enter hard standby mode by using external STANDBY signal, as shown in Figure 8.

Entering Standby Mode

1. Assert STANDBY signal HIGH.

Exiting Standby Mode

1. De-assert STANDBY signal LOW.

Figure 8: Hard Standby Operation

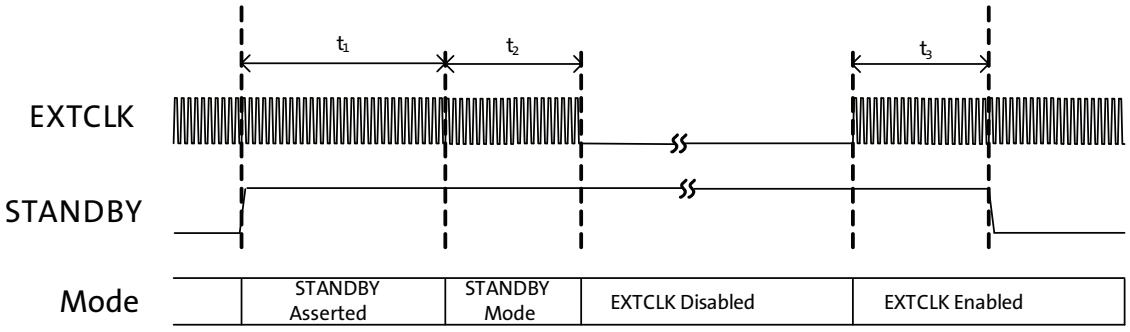


Table 9: Hard Standby Signal Timing

Symbol	Parameter	Min	Typ	Max	Unit
t_1	Standby entry complete	–	–	2 Frames	Lines
t_2	Active EXTCLK required after going into STANDBY mode	10	–	–	EXTCLKs
t_3	Active EXTCLK required before STANDBY de-asserted	10	–	–	EXTCLKs

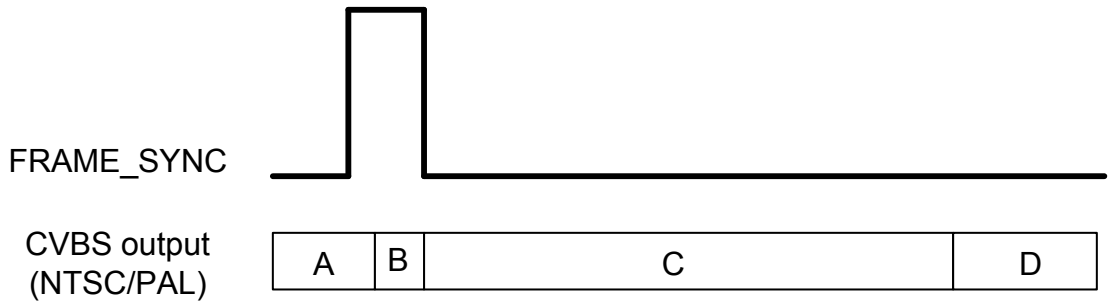
Multi-Camera Synchronization Support

The AP0100CS supports multi-camera synchronization through the FRAME_SYNC pin.

The behavior will be different depending if the user is using interlaced or progressive mode.

When using the interlaced modes, on the rising edge of FRAME_SYNC this will cause the output to stop the current frame (A) and during B the image output will be indeterminate. On the falling edge of FRAME_SYNC this will cause the re-synchronization to begin, this will continue for a period (C), during C black fields will be output. The re-synchronized interlaced signal will be available at D. During C if the user toggles the FRAME_SYNC input the AP0100CS will ignore it, the user cannot re-synchronize again until at D.

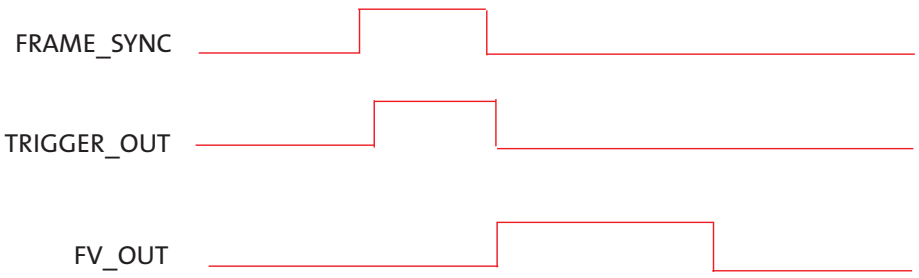
Figure 9: Frame Sync Behavior with Interlaced Mode



When using progressive mode, the host (or controlling entity) 'broadcasts' a sync-pulse to all cameras within the system that triggers capture. The AP0100AT will propagate the signal to the TRIGGER_OUT pin, and subsequently to the attached sensor's TRIGGER pin.

The AP0100CS supports two different trigger modes when using progressive output. The first mode supported is 'single-shot'; this is when the trigger pulse will cause one frame to be output from the image sensor and AP0100CS (see Figure 10).

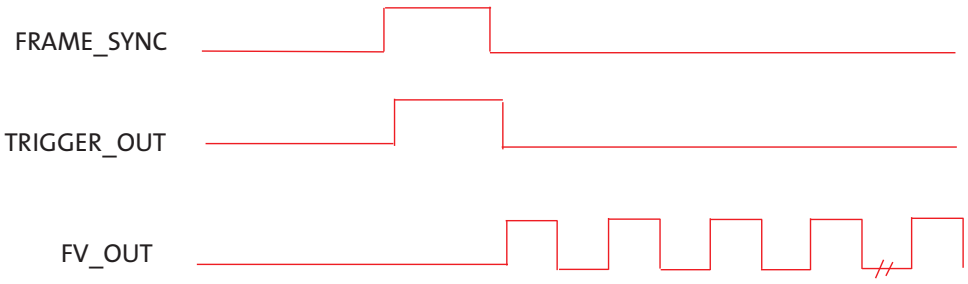
Figure 10: Single-Shot Mode



Note: This diagram is not to scale.

The second mode supported is called 'continuous', this is when a trigger pulse will cause the part to continuously output frames, see Figure 11. This mode would be especially useful for applications which have multiple sensors and need to have their video streams synchronized (for example, surround view or panoramic view applications).

Figure 11: Continuous Mode



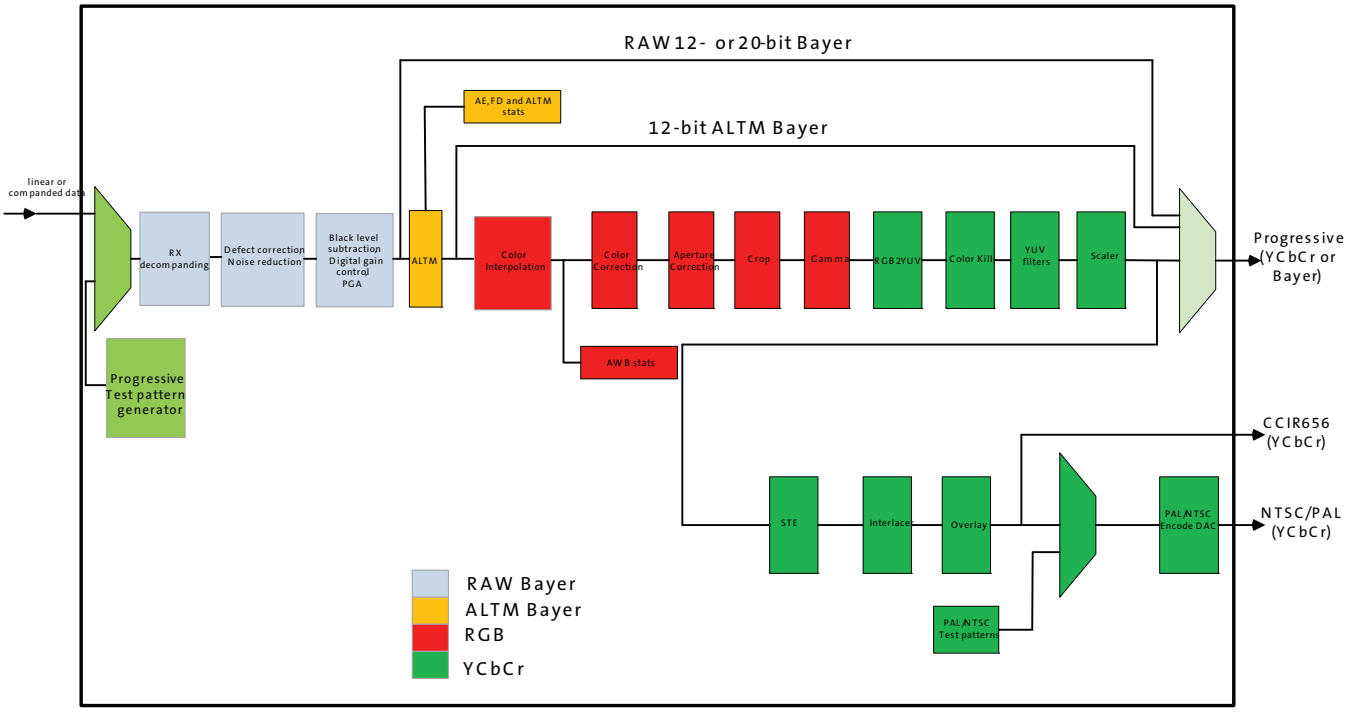
Note: This diagram is not to scale.

When two or more cameras have a signal applied to the FRAME_SYNC input at the same time, the respective FV_OUT signals would be synchronized within 5 PIXCLK_OUT cycles. This assumes that all cameras have the same configuration settings and that the exposure time is the same.

Image Flow Processor

Image and color processing in the AP0100CS is implemented as an image flow processor (IFP) coded in hardware logic. During normal operation, the embedded microcontroller will automatically adjust the operating parameters. For normal operation of the AP0100CS, streams of raw image data from the attached image sensor are fed into the color pipeline. The user also has the option to select a number of test patterns to be input instead of sensor data. The IFP is broken down into different sections, as outlined in Figure 12.

Figure 12: AP0100CS IFP



Test Patterns

The AP0100CS has a number of test patterns that are available when using the progressive, NTSC and PAL modes. The test patterns can be selected by programming variables. To enter test pattern mode, set R0xC88F to 0x02 and issue a Change-Config request; to exit this mode, set R0xC88F to 0x00, and issue a Change-Config request.

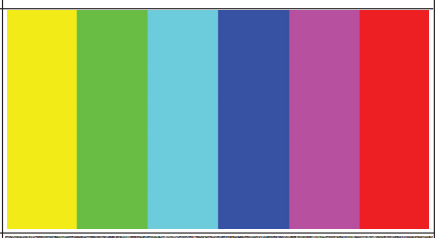
NTSC and PAL test patterns can only be selected when the device is configured for interlaced operation.

Progressive Test Patterns

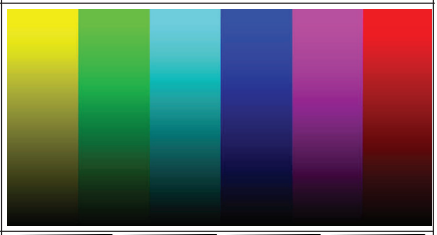
Figure 13: Progressive Test Patterns

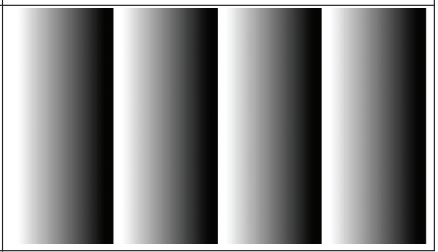
Test Pattern	Example

FLAT FIELD
REG= 0xC88C, 0x02 // CAM_MODE_SELECT
REG= 0xC88F, 0x01 // CAM_MODE_TEST_PATTERN_SELECT
REG= 0xC890, 0x000FFFFF // CAM_MODE_TEST_PATTERN_RED
REG= 0xC894, 0x000FFFFF // CAM_MODE_TEST_PATTERN_GREEN
REG= 0xC898, 0x000FFFFF // CAM_MODE_TEST_PATTERN_BLUE
Load = Change-Config
Changing the values in R0xC890-R0x898 will change the color of the test pattern (will require a Refresh operation).	

100% Color Bar
REG= 0xC88C, 0x02 // CAM_MODE_SELECT
REG= 0xC88F, 0x02 // CAM_MODE_TEST_PATTERN_SELECT
Load = Change-Config
	

Pseudo-Random
REG= 0xC88C, 0x02 // CAM_MODE_SELECT
REG= 0xC88F, 0x05 // CAM_MODE_TEST_PATTERN_SELECT
Load = Change-Config
	

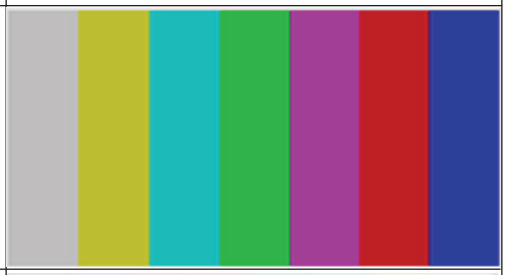
Fade-to-Gray
REG= 0xC88C, 0x02 // CAM_MODE_SELECT
REG= 0xC88F, 0x08 // CAM_MODE_TEST_PATTERN_SELECT
Load = Change-Config
	

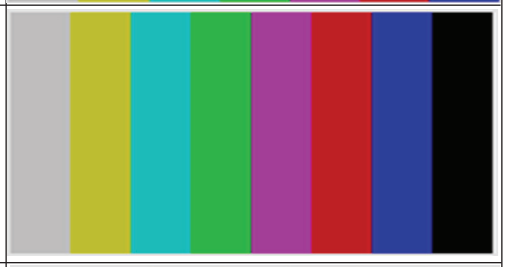
Linear Ramp
REG= 0xC88C, 0x02 // CAM_MODE_SELECT
REG= 0xC88F, 0x09 // CAM_MODE_TEST_PATTERN_SELECT
Load = Change-Config
	

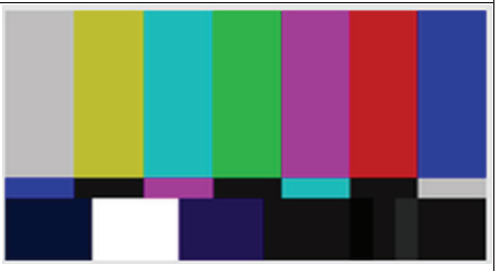
NTSC Test Patterns

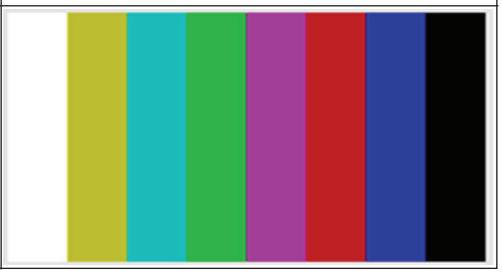
Figure 14: NTSC Test Patterns

Test Pattern	Example

EIA Full Field 7 Color Bars
REG= 0xC88C, 0x02 // CAM_MODE_SELECT
REG= 0xC88F, 0x14 // CAM_MODE_TEST_PATTTERN_SELECT
Load = Change-Config
	

EIA Full Field 8 Color Bars
REG= 0xC88C, 0x02 // CAM_MODE_SELECT
REG= 0xC88F, 0x15 // CAM_MODE_TEST_PATTERN_SELECT
Load = Change-Config
	

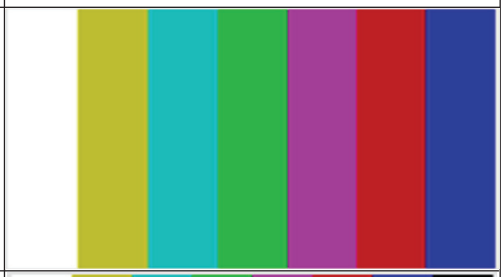
SMPTE EG 1-1990
REG= 0xC88C, 0x02 // CAM_MODE_SELECT
REG= 0xC88F, 0x16 // CAM_MODE_TEST_PATTERN_SELECT
Load = Change-Config
	

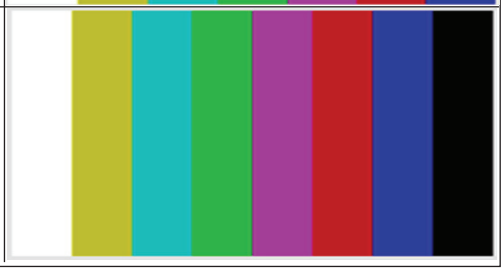
EIA Full Field 8 Color Bars 100 IRE
REG= 0xC88C, 0x02 // CAM_MODE_SELECT
REG= 0xC88F, 0x17 // CAM_MODE_TEST_PATTERN_SELECT
Load = Change-Config
	

PAL Test Patterns

Figure 15: PAL Test Patterns

Test Pattern	Example

EBU Full Field 7 Color Bars
REG= 0xC88C, 0x02 // CAM_MODE_SELECT
REG= 0xC88F, 0x1E // CAM_MODE_TEST_PATTERN_SELECT
Load = Change-Config
	

EBU Full Field 8 Color Bars
REG= 0xC88C, 0x02 // CAM_MODE_SELECT
REG= 0xC88F, 0x1F // CAM_MODE_TEST_PATTERN_SELECT
Load = Change-Config
	

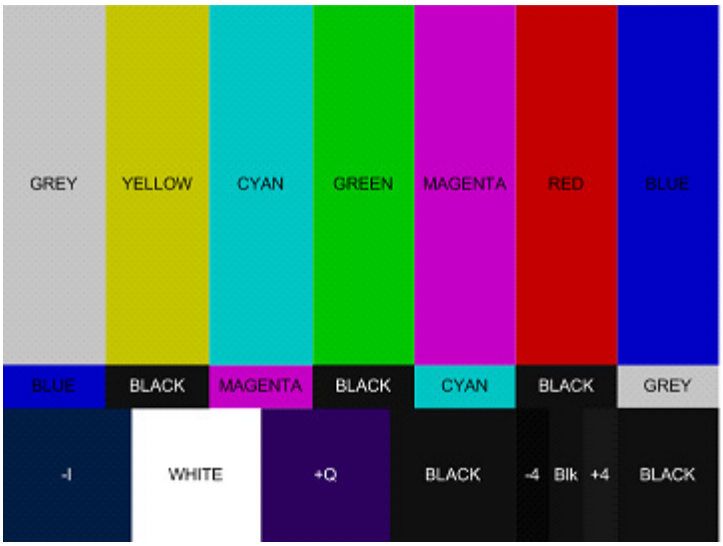
Each NTSC/PAL test pattern consists of seven or eight color bars (white, yellow, cyan, green, magenta, red, blue and optionally black). The Y, Cb and Cr values for each bar are detailed in Table 10.

For the NTSC SMPTE test pattern it is also required to generate -I, +Q, -4 black and +4 black.

Table 10: NTSC/PAL Test Pattern Values

	Nominal Range	White 100%	White 75%	Yellow	Cyan	Green	Magenta	Red	Blue	Black	-I	-Q	-4 black	+4 black
Y	16 to 235	235	180	162	131	112	84	65	35	16	16	16	7	25
Cb	16 to 240	128	128	44	156	72	184	100	212	128	156	171	128	128
Cr	16 to 240	128	128	142	44	58	198	212	114	128	97	148	128	128

Figure 16: Test Pattern



Defect Correction

Image stream processing commences with the defect correction function immediately after data decompanding.

To obtain defect free images, the pixels marked defective during sensor readout and the pixels determined defective by the defect correction algorithms are replaced with values derived from the non-defective neighboring pixels. This image processing technique is called defect correction.

AdaCD (Adaptive Color Difference)

Automotive applications require good performance in extremely low light, even at high temperature conditions. In these stringent conditions the image sensor is prone to higher noise levels, and so efficient noise reduction techniques are required to circumvent this sensor limitation and deliver a high quality image to the user.

The AdaCD Noise Reduction Filter is able to adapt its noise filtering process to local image structure and noise level, removing most objectionable color noise while preserving edge details.

Black Level Subtraction and Digital Gain

After noise reduction, the pixel data goes through black level subtraction and multiplication of all pixel values by a programmable digital gain. Independent color channel digital gain can be adjusted with registers. Black level subtraction (to compensate for sensor data pedestal) is a single value applied to all color channels. If the black level subtraction produces a negative result for a particular pixel, the value of this pixel is set to 0.

Positional Gain Adjustments (PGA)

Lenses tend to produce images whose brightness is significantly attenuated near the edges. There are also other factors causing fixed pattern signal gradients in images captured by image sensors. The cumulative result of all these factors is known as image shading. The AP0100CS has an embedded shading correction module that can be programmed to counter the shading effects on each individual R, Gb, Gr, and B color signal.

The Correction Function

The correction functions can then be applied to each pixel value to equalize the response across the image as follows:

$$P_{corrected}(row, col) = P_{sensor}(row, col) \times f(row, col) \quad \text{(EQ 1)}$$

where P are the pixel values and f is the color dependent correction functions for each color channel.

Adaptive Local Tone Mapping (ALTM)

Real world scenes often have very high dynamic range (HDR) that far exceeds the electrical dynamic range of the imager. Dynamic range is defined as the luminance ratio between the brightest and the darkest object in a scene. In recent years many technologies have been developed to capture the full dynamic range of real world scenes. For example, the multiple exposure method is widely adopted for capturing high dynamic range images, which combines a series of low dynamic range images of the same scene taken under different exposure times into a single HDR image.

Even though the new digital imaging technology enables the capture of the full dynamic range, low dynamic range display devices are the limiting factor. Today's typical LCD monitor has contrast ratio around 1,000:1; however, it is not typical for an HDR image (the contrast ratio for an HDR image is around 250,000:1). Therefore, in order to reproduce HDR images on a low dynamic range display device, the captured high dynamic range must be compressed to the available range of the display device. This is commonly called tone mapping.

Tone mapping methods can be classified into global tone mapping and local tone mapping. Global tone mapping methods apply the same mapping function to all pixels. While global tone mapping methods provide computationally simple and easy to use solutions, they often cause loss of contrast and detail. A local tone mapping is thus necessary in addition to global tone mapping for the reproduction of visually more appealing images that also reveal scene details that are important for automotive safety and surveillance applications. Local tone mapping methods use a spatially variable mapping function determined by the neighborhood of a pixel, which allows it to increase the local contrast and the visibility of some details of the image. Local methods usually yield more pleasing results because they exploit the fact that human vision is more sensitive to local contrast.

ON Semiconductor's ALTM solution significantly improves the performance over global tone mapping. ALTM is directly applied to the Bayer domain to compress the dynamic range from 20-bit to 12-bit. This allows the regular color pipeline to be used for HDR image rendering.

Color Interpolation

In the raw data stream fed by the external sensor to the IFP, each pixel is represented by a 20- or 12-bit integer number, which can be considered proportional to the pixel's response to a one-color light stimulus, red, green, or blue, depending on the pixel's position under the color filter array. Initial data processing steps, up to and including ALTM, preserve the one-color-per-pixel nature of the data stream, but after ALTM it must be converted to a three-colors-per-pixel stream appropriate for standard color processing. The conversion is done by an edge-sensitive color interpolation module. The module pads the incomplete color information available for each pixel with information extracted from an appropriate set of neighboring pixels. The algorithm used to select this set and extract the information seeks the best compromise between preserving edges and filtering out high frequency noise in flat field areas. The edge threshold can be set through register settings.