

## Introduction

The IP core is designed to work with two AXI4-Stream sources among which one is background stream and another is foreground stream. The output is obtained such that the foreground stream is blended on the top of the background stream with partial or full transparency based on alpha level.

## Features

- Generates partial or full transparent color output.
- Configurable parameters such as resolution, alpha level,
- Supports a maximum of 1920x1080, resolution with 1 pixel per clock,
- AXI4-Lite control interface,
- AXI4-Stream interface for input and output video,
- Supports 24-bit RGB input and output.

## Applications

- Medical Imaging
- Photography
- Computer Vision

IP FACTS	
Core Specs	
Supported Device Family	Xilinx's 7 Series, Zynq-7000, *UltraScale/UltraScale+
Supported User Interfaces	AXI4-Lite, AXI4-Stream
Resources Utilization	Included in this document
Provided Sources	
Documentation	Product Guide
Design Files	Not Provided
Example Design	Yes
Test Bench	Not Provided
Simulation Models	Not Provided
Supported Software Drivers	Standalone
Tested Design Flows	
Design Entry Tools	Xilinx's Vivado Design Suite
Simulation	Xilinx's Vivado Simulator
Synthesis Tools	Xilinx's Vivado Synthesis
Support	
Provided by LogicTronix	

*Table 1. IP Facts*

*\* For >1080p resolution support, please contact us !*

## Overview

The Alpha Blending (ABLND) IP core is created for combining a video/image stream with a background stream to obtain partial or full transparency. Moreover, the secondary stream is blended on the top of the primary stream with an alpha setting. Note that, the *Background Stream* or *Master Stream* is indicated by *s\_axis* while the *Foreground Stream* is indicated by *s\_axis\_layer*. Thus, *s\_axis\_layer* data will be blended over the *s\_axis* data. Here users can provide the Alpha level from 0 to 255 to decrease or increase the transparency in the output image. This IP can be used on any input stream that gives RGB video/image output. The IP supports a maximum of 1920x1080 video resolution. The IP has AXI4-compliant input and output stream interfaces. These interfaces support a 24-bit RGB format.

The IP has an AXI4-Lite Interface which provides better control over the IP. This interface allows setting the video resolution and alpha value levels for the output.

## Port Description



Figure 1. Alpha Blending IP Top-Level View

The IP is created with industry-standard control and data interfaces. These interfaces allow communication with other IPs or system components.

The ABLND core ports are described by the following interfaces.

### 1. Clock, Reset and Interrupt signal Interface

These signals are summarized in the following table.

Signal Name	Width	Direction	Description
ap_clk	1	IN	Core clock for both AXI4-Stream as well as AXI4-Lite Interface
ap_rst_n	1	IN	Core ap_clk synchronous active low reset
Interrupt	1	OUT	Core Interrupt pin

*Table 2. Clock, Reset and Interrupt Signal Interface Description*

## 2. Video Interface

The IP has three data interfaces *s\_axis*, *s\_axis\_layer* and *m\_axis* that implement the AXI4-Stream interface protocol. These interfaces are used to get input as well as output stream data respectively.

### AXI4-Stream Signals

The following table gives a short description of the individual signal pins of the AXI4-Stream Interface.

Signal Name	Width	Direction	Description
<b>AXI4-Stream Input Signals</b>			
s_axis_tdata	24	IN	Input video Data
s_axis_tvalid	1	IN	Input valid
s_axis_tready	1	OUT	Input ready
s_axis_tuser	1	IN	Input video start of frame
s_axis_tlast	1	IN	Input video end of line
s_axis_tstrb	3	IN	Input video data strobe indicates whether the content of the associated byte of tdata is processed as a data byte or position byte
s_axis_tkeep	3	IN	Input video byte qualifier that indicates whether the content of the associated byte of tdata is processed as part of the data stream
s_axis_tid	1	IN	Input video data identifier
s_axis_tdest	1	IN	Input video data routing information
s_axis_layer_tdata	24	IN	Input video Data

s_axis_layer_tvalid	1	IN	Input valid
s_axis_layer_tready	1	OUT	Input ready
s_axis_layer_tuser	1	IN	Input video start of frame
s_axis_layer_tlast	1	IN	Input video end of line
s_axis_layer_tstrb	3	IN	Input video data strobe indicates whether the content of the associated byte of tdata is processed as a data byte or position byte
s_axis_layer_tkeep	3	IN	Input video byte qualifier that indicates whether the content of the associated byte of tdata is processed as part of the data stream
s_axis_layer_tid	1	IN	Input video data identifier
s_axis_layer_tdest	1	IN	Input video data routing information
<b>AXI4-Stream Output Signals</b>			
m_axis_tdata	24	OUT	Output video Data
m_axis_tvalid	1	OUT	Output valid
m_axis_tready	1	IN	Output ready
m_axis_tuser	1	OUT	Output video start of frame
m_axis_tlast	1	OUT	Output video end of line
m_axis_tstrb	3	OUT	Output video data strobe indicates whether the content of the associated byte of tdata is processed as a data byte or position byte
m_axis_tkeep	3	OUT	Output video byte qualifier that indicates whether the content of the associated byte of tdata is processed as part of the data stream
m_axis_tid	1	OUT	Output video data identifier
m_axis_tdest	1	OUT	Output video data routing information

*Table 3. AXI4-Stream Signal Names and Descriptions*

All streaming interfaces run at *ap\_clk*.

### 3. Control Interface

The IP consists of an AXI4-Lite interface as a control interface. This allows us to configure or control the IP dynamically. This interface will be connected to the GP interface of Zynq PS or Microblaze.

#### AXI4-Lite Interface Signals

The AXI4-Lite Interface signal names and their description are given in the following table.

Signal Name	Width	Direction	Description
s_axi_lite_awvalid	1	IN	AXI4-Lite Write Address Channel Write Address Valid
s_axi_lite_awread	1	OUT	AXI4-Lite Write Address Channel Write Address Ready. INDicates DMA ready to accept the wire address.
s_axi_lite_awaddr	6	IN	AXI4-Lite Write Address Bus
s_axi_lite_wvalid	1	IN	AXI4-Lite Write Data Channel Write Data Valid
s_axi_lite_wready	1	OUT	AXI4-Lite Write Data Channel write Data Ready. Indicates DMA is ready to accept the write data.
s_axi_lite_wdata	32	IN	AXI4-Lite Write Data bus
s_axi_lite_wstrb	4	IN	AXI4-Lite Write Data Strobe
s_axi_lite_bresp	2	OUT	AXI4-Lite Write Response Channel. Indicates results of the write transfer
s_axi_lite_bvalid	1	OUT	AXI4-Lite Write Response Channel Response Valid. Indicates response is valid
s_axi_lite_bready	1	IN	AXI4-Lite Write Response Channel Ready. This indicates the target is ready to receive a response.
s_axi_lite_arvalid	1	IN	AXI4-Lite Read Address Channel Read Address Valid
s_axi_lite_arready	1	OUT	AXI4-Lite Ready. Indicates DMA is ready to accept the read address.
s_axi_lite_araddr	6	IN	AXI4-Lite Read Address Bus

s_axi_lite_rvalid	1	OUT	AXI4-Lite Read Data Channel Read Data Valid
s_axi_lite_rready	1	IN	AXI4-Lite Read Data Channel Read Data Ready. INDicates target is ready to accept the read data.
s_axi_lite_rdata	32	OUT	AXI4-Lite Ready Data Bus.
s_axi_lite_rresp	2	OUT	AXI4-Lite Read Response Channel Response. INDicates results of the read transfer.

*Table 4. AXI4-Lite Interface Signal Names and Description*

This interface also runs at the ap\_clk clock.

## Register Space

The IP has built-in registers so that IP parameters can be set by accessing these registers and then setting the appropriate hex value. The register can be accessed by its offset address. These registers are programmed by AXI4-Lite Interface.

The register name, address and description are given below.

<b>BASEADDR Offset (Hex)</b>	<b>Register Name</b>	<b>Type</b>	<b>Description</b>
0x00	Control Signals	R/W	Bit 0: ap_start Bit 1: ap_done Bit 2: ap_idle Bit 3: ap_ready Bit 7: auto_restart
0x04	Global Interrupt Enable Register	R/W	Bit 0: Global Interrupt Enable
0x08	IP Interrupt Enable Register	R/W	Bit 0: Channel 0 (ap_done) Bit 1: Channel 1 (ap_ready)
0x0C	IP Interrupt Status Register	R/W	Bit 0: Channel 0 (ap_done) Bit 1: Channel 1 (ap_ready)
0x10	Alpha	R/W	Alpha value of <i>s_axis_layer</i> input stream
0x18	Height	R/W	Number of Active Lines Per Frame
0x20	Width	R/W	Number of active pixels per scanline

*Table 5. IP Register Names, Offset Addresses and Descriptions*

## Video Data

The core generates RGB video data with 1 pixel per clock and 8-bits per component through the *m\_axis* port.



Figure 2. Single Pixel Per Clock, 8-bits per component RGB Video Data Format

## Data Flow

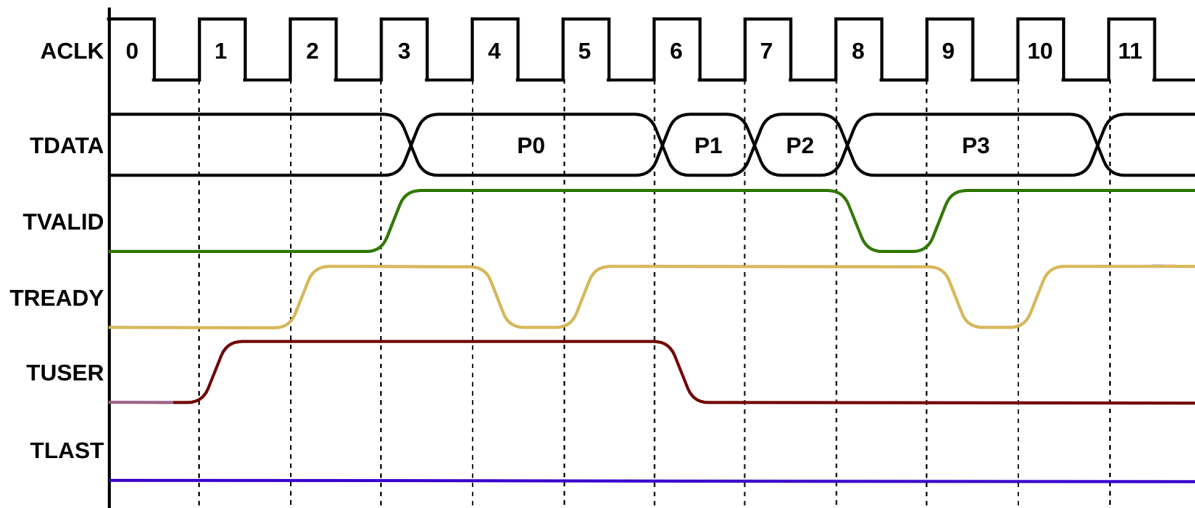


Figure 3. AXI4-Stream Data flow mechanism

The flow of data is based on AXI4-Stream protocol, which mainly occurs by the four signals in the current IP design. They are; *m\_axis\_tvalid*, *m\_axis\_tready*, *m\_axis\_tuser* (SOF) and *m\_axis\_tlast* (EOL). These signals are called AXI4-stream handshaking signals. The data is carried out by *m\_axis\_tdata*.

For a complete transfer of the frame, the SOF signal goes high, which indicates the beginning of the frame. During this moment, *m\_axis\_tready* and *m\_axis\_tvalid* must be at HIGH state to begin the flow of valid data from this IP (master) to receiving IP (slave). EOL is asserted HIGH when the transfer of pixels per scanline is completed.

## Designing with the Alpha Blending IP core

The design with the core has been summarized by the following picture.

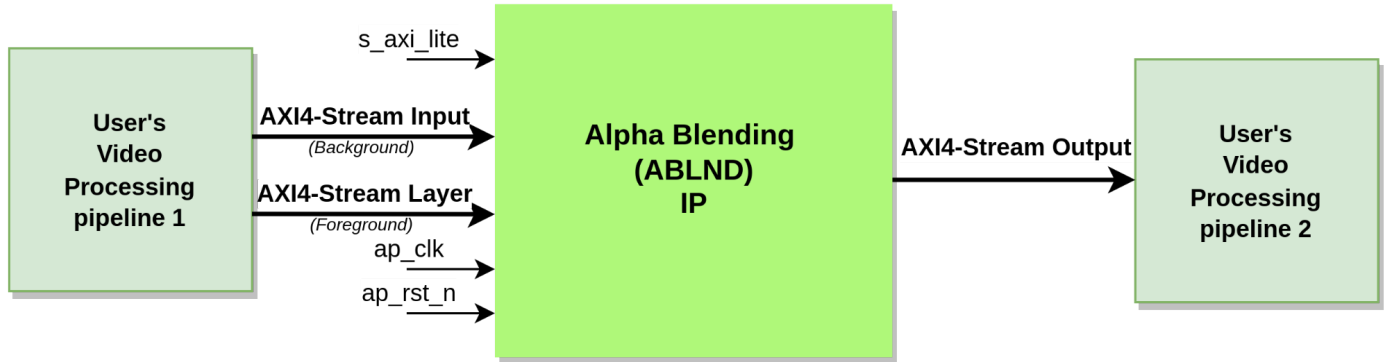


Figure 4. Designing with the Alpha Blending IP core

The IP needs a single clock for both the AXI4-Lite interface and streaming Interface. And the corresponding active low synchronous reset is also required to connect. The IP has an AXI4-Lite interface, which is connected with the master interface of the host processor. This interface is used to configure the IP. Therefore, a software application is required, which will initialize, configure and run the IP with a specific value. The IP does pixel by pixel processing. User has to set the alpha value from 0 to 255 for the transparency of the foreground image with the background.

---

## Performance

### Maximum Frequencies

The maximum frequencies that the IP core can be operated on are different due to board types, tool versions and the way of design with the core.

### Throughput

The ABLND IP core has input and output AXI4-Stream interface. The data throttling is bidirectional between input and output interfaces. The flow of data takes place as long as the source produces valid data and the destination is ready to receive data.

Technically, if *s\_axis\_tvalid* is not asserted, the ABLND core cannot produce valid data. On the other hand, if *m\_axis\_tready* is not asserted, the core cannot receive valid data from the source.

If source is producing valid data, that is, *s\_axis\_tvalid* is asserted and destination is ready to receive the valid data, that is, *m\_axis\_tready* is asserted, the ABLND IP core generates the valid data, which is indicated by asserting *m\_axis\_tvalid*. At this moment, the core delivers the 24-bit valid data with one pixel per clock as per *ap\_clk*. The core must be operated at least 148.5MHz clock for the 1080p60 resolution.

## Resource Utilization

The FPGA resources consumed by Alpha Blending IP core is summarized as follows;

<b>Board</b>	Xilinx's ZYNQ ZC706 Evaluation Board		
<b>Device</b>	xc7z045ffg900-2		
<b>Vivado Version</b>	2019.1		
Resource Utilization			
Site Type	Available	Utilization	Utilization %
<b>Slice</b>	54650	178	0.33
<b>LUT</b>	218600	341	0.16
<b>LUTRAM</b>	70400	0	0.00
<b>FF</b>	437200	598	0.14
<b>DSP</b>	900	6	0.67
<b>BRAM 36K</b>	545	0	0.00
<b>BRAM 18K</b>	1090	0	0.00
<b>MMCME2_ADV</b>	8	0	0.00
<b>PLLE2_ADV</b>	8	0	0.00

*Table 6. Resource Utilization by Alpha Blending core*

## Example Design

This section provides information about the example design with the core. This example design only consists of a synthesizable design.

### Synthesizable Design

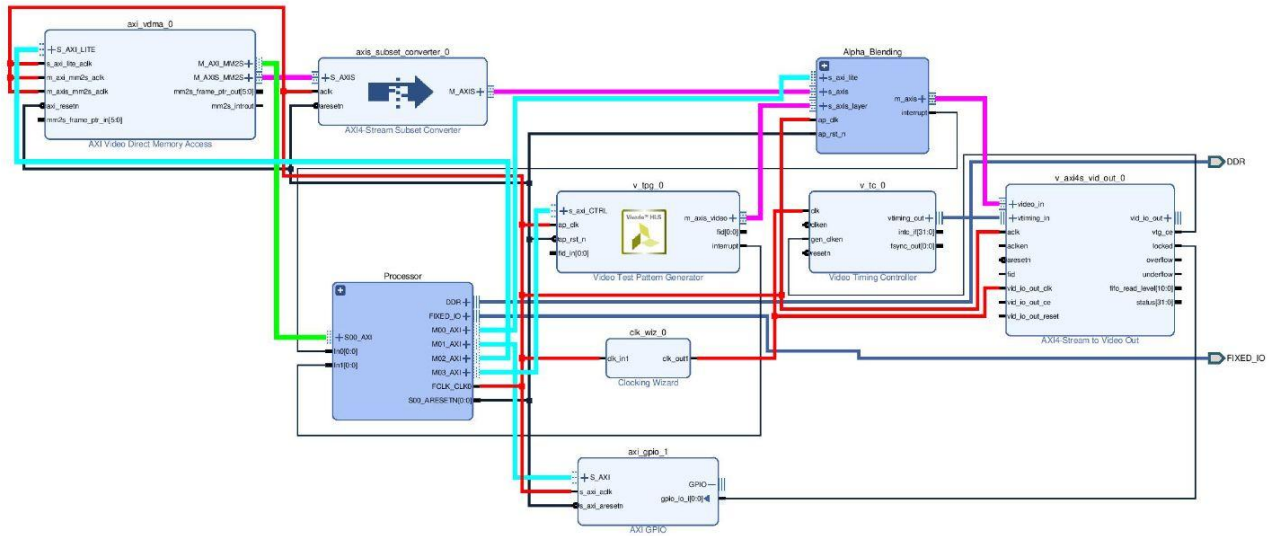


Figure 5. Synthesizable Alpha Blending IP core Example Design

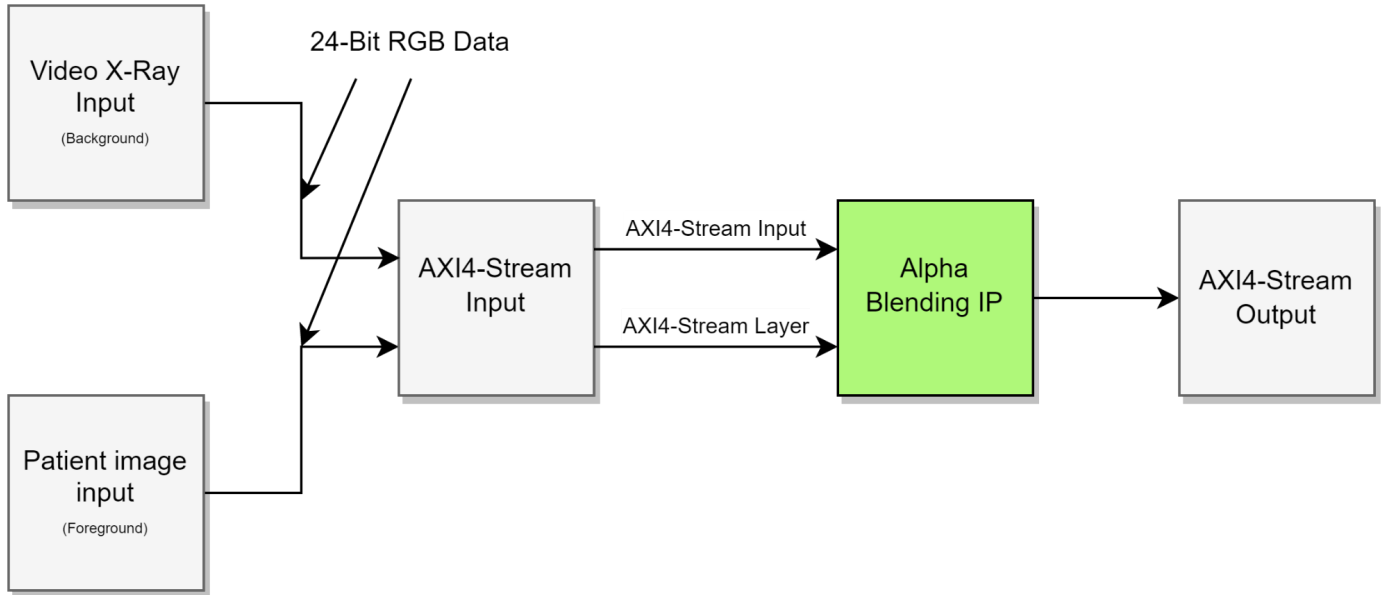
The synthesizable example design is shown in the above picture. Please visit the Use Case section to get information about the use of the IP core in practical situations.

**Note:** This example design is created for the **Xilinx ZC706 Evaluation Board**.

In this design, Zynq processor is used as a host processor. The ABLND IP core is interfaced with the Zynq processor by the AXI4-Lite interface. In this current example design, the background image is loaded into the DDR memory, which is read back and streamed by the AXI VDMA core. The Xilinx Test Pattern Generator (XTPG) is used to get the foreground image stream. The data are in AXI4-stream format so that they can be fed to the Alpha Blending IP core. Thus, the IP core generates a 24-bit blended output stream based on the alpha level selected.

The VTC and AXI4-Stream to Video Out are used for outputting the stream in 1080p resolution. A locked signal from AXI4-stream to Video Out IP is monitored by an AXI GPIO IP to indicate output is obtained successfully or not.

## Use Case



*Figure 6. Typical application of Alpha Blending IP core*

The above picture shows a general use-case of the ABLND IP core. In real-life, the alpha blending comes under the post-image blending technique. In case of surgeries, the doctor and surgeon need precision data to perform surgery, so there is a need of blending technique to combine the patient image with the Video X-Ray or CT scan image to determine the precise point for the surgery. For such a scenario, several post-image processing algorithms might be required to blend the images perfectly. When blending of the images is required, then an alpha blending algorithm comes into play and this is what is achieved by ABLND IP. This IP core is a wise choice for FPGA based image processing.

To use and apply alpha blending, ABLND IP core requires AXI4-complaint input image, which means that the image data needs to be in AXI4-Stream. And oppositely, to get a blended image displayed, the AXI4-Stream is needed to convert into 24-bit Native Video format.

---

## References

1. Vivado Design Suite: AXI Reference Guide ([UG1037](#))
2. AXI Reference Guide ([UG761](#))
3. Vivado Design Suite User Guide: Designing with IP ([UG896](#))
4. Vivado Design Suite User Guide: Getting Started ([UG910](#))
5. Vivado Design Suite User Guide: Programming and Debugging ([UG908](#))
6. Vivado Design Suite User Guide: Implementation ([UG904](#))

## Revision History

The following table shows the revision history of this product guide - PGL036.

Date	Version	Detail
December 27, 2021	1.0	Initial Release

*Table 7. IP core Revision History*

## About LogicTronix

LogicTronix provides Turnkey Solutions, Design Services, and Intellectual Property (IP) to customers on FPGA Design, Computer/Machine Vision, Machine Learning Acceleration on FPGA [Edge or Cloud] for various applications including ADAS, Surveillance, Computer Vision, FinTech, etc.

**LogicTronix also offers solutions on "Real-Time Traffic Video Analytics Solution (TVAS) - including Automatic vehicle Number-Plate Recognition (ANPR) Solution", "Enhancing Financial Trading Algorithms with AI/ML" and "High-Frequency Trading (HFT) based Infrastructure".**

## For IP Licensing, Sales and Support: **Contact**

LogicTronix Technologies Pvt. Ltd.

### **Xilinx Certified Partner**

Email: [ip-sales@logictronix.com](mailto:ip-sales@logictronix.com)

Web: [www.LogicTronix.com](http://www.LogicTronix.com)