## Introduction

This document describes the specifications of the Monochrome (old school black and white) Video Graphics Array (VGA) core for the On-Chip AXI Bus.

The VGA is a 32-bit master-slave module that attaches to the AXI.

## Features

- DMA capable 32-bit AXI v2.0 bus interface
- Monochrome (1 bit per pixel, 32-pixel per word)
- Supports 640 x 480 video resolution only (307200 b & w pixels)
- 60 Hz vertical synchronization
- Configurable start address
- Support for interruptions and polling

## Functional Description

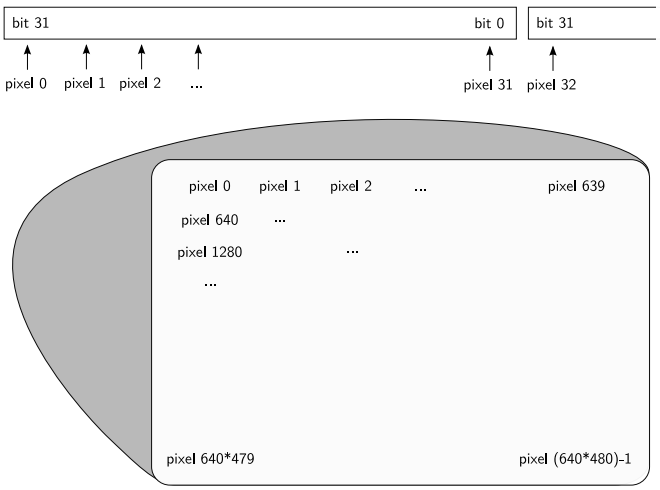


Figure 1: Video buffer data format

After reset, the VGA is initially idle. User sets the start address by writing to the CFG register. If the value written to this register is different from 0x0, then the VGA becomes enabled.

When enabled, the VGA continuously reads 32-bit words, starting from the start address (value of the CFG register), and incrementing until the end of the video buffer (9600 32-bit words long). It drives the color (RED, GREEN and BLUE) and synchronization (HSYNC, VSYNC) signals.

Each time the VGA reaches the end of the buffer, it drives the vertical synchronization signal VSYNC and sends an interruption using the IP2INTC_Irpt signal. It also sets the INT register to 0x1. The INT register can be cleared by writing 0x1 to it.

The VGA can be idled by writing 0x0 to the CFG register.

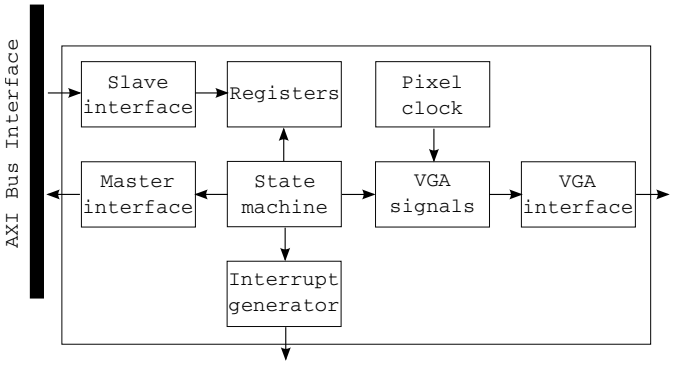


Figure 2: VGA Block diagram

## Programming Model

### Modes

The VGA provides the following modes:

- IDLE: when CFG register is 0x0. In this mode, a test pattern is displayed (vertical lines).
- ENABLED: otherwise

### Register offsets

| | Offset | Size | Type | Description |
|---|---|---|---|---|
| CFG | 0x00 | Word | R/W | Configuration reg. |
| STT | 0x04 | Word | R | Status register |
| INT | 0x08 | Word | R/W | Interrupt register |

### Registers descriptions

#### Configuration register

The configuration registers holds the start address of the video buffer. The value should be modified soon after the vertical synchronization to avoid tearing.

#### Status register

The status register is not implemented.

#### Interrupt register

The interrupt register is set to 0x1 after a vertical synchronization and can be cleared by writing 0x1 to it.