*Technion*

*Electrical Engineering Department*

High Speed Digital System Lab

VESA Generator

Documentation

**Version**: 1.0

**Last update**: 26.02.2011

**Written By**: Beeri Schreiber and Alon Yavich

Table of content

[Table of Changes 3](#_Toc293164712)

[1 Scope 3](#_Toc293164713)

[2 Abbreviations 3](#_Toc293164714)

[3 General Description 3](#_Toc293164715)

[3.1 DVI Operation 3](#_Toc293164716)

[3.1.1 Electron Beam 4](#_Toc293164717)

[3.2 VESA Generator Operation 5](#_Toc293164718)

[3.2.1 Frame 5](#_Toc293164719)

[3.2.2 Defenition of Terms 6](#_Toc293164720)

[4 VESA Generator Pinout 6](#_Toc293164721)

[5 VESA Generator Generic Parameters 7](#_Toc293164722)

[6 Resources 8](#_Toc293164723)

[7 VESA Waves 9](#_Toc293164724)

[7.1 Start of Active Video 9](#_Toc293164725)

[7.1.1 Image Enables 9](#_Toc293164726)

[7.2 Between two frames 10](#_Toc293164727)

[7.3 Complete Frame (Image) 10](#_Toc293164728)

[7.3.1 ROI – Region of Interest 10](#_Toc293164729)

[7.4 Request for Data 11](#_Toc293164730)

[7.5 Prepare Data Provider Trigger 11](#_Toc293164731)

[8 Internal Implementation 11](#_Toc293164732)

[8.1 Horizontal and Vertical Counters 11](#_Toc293164733)

[8.2 Pipeline 12](#_Toc293164734)

[8.2.1 Code Example – Without Pipeline 13](#_Toc293164735)

[8.2.2 Code Example – With Pipeline 13](#_Toc293164736)

[9 Performed Tests for VESA Generator 14](#_Toc293164737)

[9.1 Output Files for Example 15](#_Toc293164738)

Table of Tables

[Table 1 – Table of Changes 3](#_Toc287446480)

[Table 2 – VESA Generator Pinout 7](#_Toc287446481)

[Table 3 – VESA Generator Generic Parameters 8](#_Toc287446482)

Tables of Figures

[Figure 1 – HSync, VSync and Electron Beam 4](#_Toc287014281)

[Figure 2 – Frame of Image 6](#_Toc287014282)

[Figure 3 – Definition of Terms 6](#_Toc287014283)

[Figure 4 – VESA Generator Pinout 6](#_Toc287014284)

[Figure 5 – VESA Start of Active Video 9](#_Toc287014285)

[Figure 6 – VESA VSync and Blank example 10](#_Toc287014286)

[Figure 7 – VESA Complete Frame 10](#_Toc287014287)

[Figure 8 – VESA Request for Data 11](#_Toc287014288)

[Figure 9 – VESA Prepare Data Provider Trigger 11](#_Toc287014289)

[Figure 10 – Counters 12](#_Toc287014290)

[Figure 11 – Pipeline 12](#_Toc287014291)

[Figure 12 – Code Example - Without Pipeline 13](#_Toc287014292)

[Figure 13 – Code Example - With Pipeline 13](#_Toc287014293)

[Figure 14 – Input Files 14](#_Toc287014294)

[Figure 15 – Output Files for Example 15](#_Toc287014295)

# Table of Changes

|  |  |  |
| --- | --- | --- |
| Version | Date of Change | Description |
| 1.0 | 26.02.2011 | Creation of documentation |

Table 1 – Table of Changes

# Scope

This document aims to describe the working method of the VESA Generator.

# Abbreviations

1. VESA – Video Electronics Standards Association
2. HSync – Horizontal Synchronization
3. VSync – Vertical Synchronization
4. VGA - Video Graphics Array
5. DVI - Digital Visual Interface
6. ROI – Region of Interest
7. TB – Test bench

# General Description

## DVI Operation

A DVI video signal contains 6 signals:

* Clock – Pixel Clock
* HSync – New Line signal
* VSync – New Frame (image) signal
* Blank – Blanking signal to the screen (ignore the RGB inputs)
* RED – Analog signal, used to control the color
* GREEN – Analog signal, used to control the color
* BLUE – Analog signal, used to control the color

Colors are produced by changing the analog levels of the three RGB signals.

The screen refresh process begins in the top left corner and paints 1 pixel at a time from left to right. At the end of the first row, the row increments and the column address is reset to the first column. Once the entire screen has been painted, the refresh process begins again.

The video signal must redraw the entire screen 60 times per second to provide motion in the image and to reduce flicker. This period is called the refresh rate.

In 800X600 pixel mode, with a 60 Hz refresh rate, this is approximately 25 ns per pixel, which requires a 40 MHz.

The **VSync** signal commands the monitor to start displaying a new frame, and the monitor starts in the upper left corner with pixel (0, 0).

The **HSync** signal commands the monitor to refresh another row of 800 pixels.

After 600 rows of pixels are refreshed with 600 HSync signals, a VSync signal resets the monitor to the upper left corner and the process continues.

During the time when pixel data is not being displayed (**Blanking**) and the beam is returning to the left column to start another horizontal scan, the RGB signals should all be set to black color (all zero).

### Electron Beam

The electron beam must be scanned over the viewing screen in a sequence of horizontal lines to generate an image. The RGB color information in the video signal is used to control the strength of the electron beam.

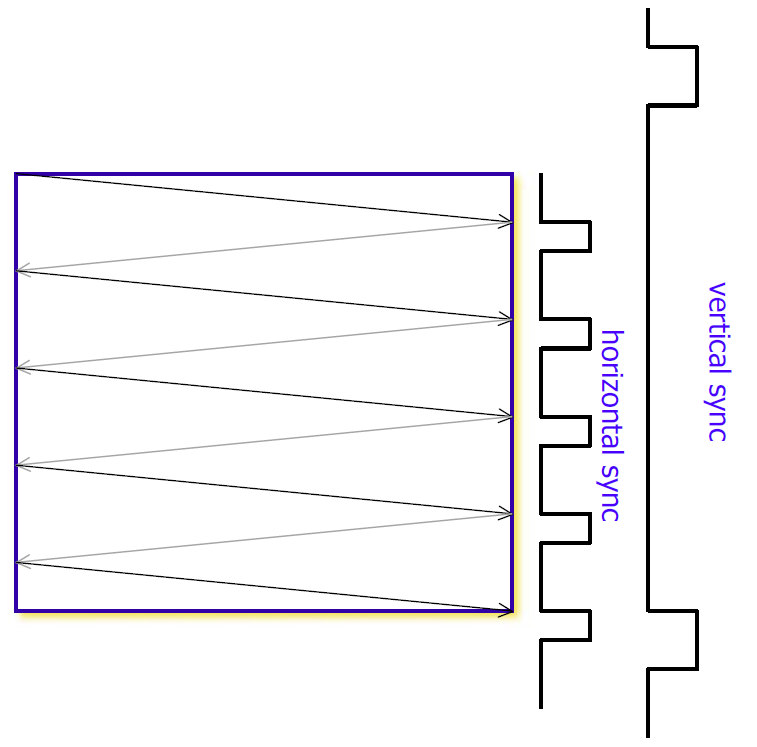


Figure 1 – HSync, VSync and Electron Beam

## VESA Generator Operation

VESA Generator is a VESA Non-Interlaced Controller, which transmit to the DVI screen the following data and signals:

* (R, G, B) Pixels
* Horizontal and Vertical Sync
* Blanking

**IMPORTANT:** The DVI should be supplied by pixel-clock. The VESA generator does notsupplies the pixel-clock. The user shall ensure that the DVI has the same clock input as the VESA generator.

The (R, G, B) Pixels should be supplied to the VESA generator by an outside component.

Back Porch, Front Proch, Left - Right - Upper - Lower borders, Sync Time and Active pixels / lines should be defined, according to the VESA standard, using the generic parameters.

The R, G and B port sizes may be changed individually, using a generic parameter.

The R, G and B values will be transmitted, one by one, in the each pixel-clock. See clock explanation in the DVI operation (40MHz in the example).

### Frame

A generic-color frame can wrap the transmitted image, using the left, right, lower and upper input registers.

Figure 2 – Frame of Image

### Defenition of Terms

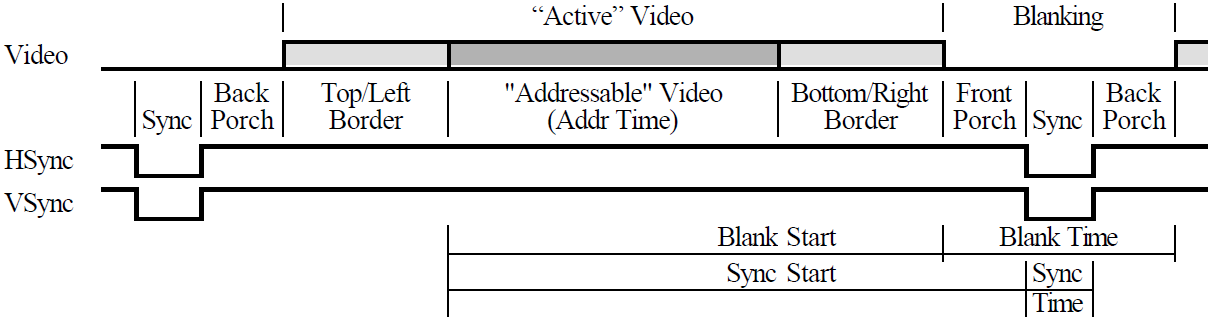


Figure 3 – Definition of Terms

# VESA Generator Pinout

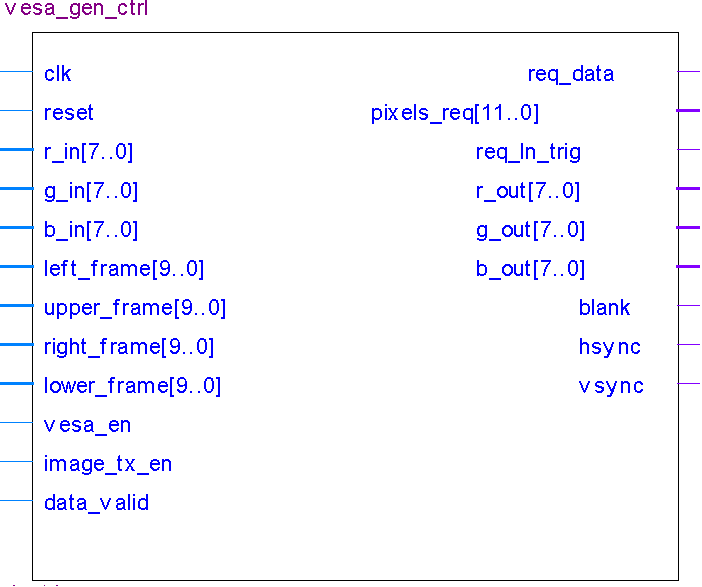


Figure 4 – VESA Generator Pinout

| Pin Name | Direction | Description |
| --- | --- | --- |
| Clk | In | Clock |
| Reset | In | Reset. Reset polarity will be set according to the generic parameter '*reset\_polarity\_g*' |
| R\_in [red\_width\_g – 1..0] | In | Input RED Pixel |
| G\_in [green\_width\_g – 1..0] | In | Input GREEN Pixel |
| B\_in [blue\_width\_g – 1..0] | In | Input BLUE Pixel |
| Left\_frame [*log*(hor\_active\_pixels\_g)-1..0] | In | Left frame size (Number of pixels) – Wraps the ROI. |
| Upper\_frame [*log*(ver\_active\_lines\_g)-1..0] | In | Upper frame size (number of lines) – Wraps the ROI. |
| Right\_frame [*log*(hor\_active\_pixels\_g)-1..0] | In | Right frame size (Number of pixels) – Wraps the ROI. |
| Lower\_frame [*log*(ver\_active\_lines\_g)-1..0] | In | Lower frame size (number of lines) – Wraps the ROI. |
| Vesa\_en | In | Enable VESA to transmit frame to the DVI |
| Image\_tx\_en | In | Image transmitter (data provider to the VESA generator) is enabled |
| Data\_valid | In | Input data to the generator is valid. In case *data\_valid* is expected, but is not supplied – BLACK pixel will replace the required pixel. |
| Req\_data | Out | Request for data from the data provider |
| Pixel\_req[*log*(hor\_active\_pixels\_g\*req\_lines\_g) -1..0] | Out | Request for PIXELS\*LINES pixels from synthetic data provider |
| Req\_ln\_trig | Out | Trigger to the image transmitter to load the FIFO with new data |
| R\_out [red\_width\_g – 1..0] | Out | Output RED pixel |
| G\_out[green\_width\_g – 1..0] | Out | Output GREEN pixel |
| B\_out[blue\_width\_g – 1..0] | Out | Output BLUE pixel |
| Blank | Out | Blanking signal |
| Hsync | Out | Horizontal Synchronization signal |
| Vsync | Out | Vertical synchronization |

Table 2 – VESA Generator Pinout

# VESA Generator Generic Parameters

| Generic Parameter | Default Value | Description |
| --- | --- | --- |
| Reset\_polartiy\_g | '0' | Reset active in this polarity |
| hsync\_polarity\_g | '1' | HSync polarity |
| vsync\_polarity\_g | '1' | VSync polarity |
| blank\_polarity\_g | '0' | Blank polarity |
| red\_default\_color\_g | 0 | Default Red pixel for Frame |
| green\_default\_color\_g | 0 | Default Green pixel for Frame |
| blue\_default\_color\_g | 0 | Default Blue pixel for Frame |
| red\_width\_g | 8 | Default std\_logic\_vector size of Red Pixels |
| green\_width\_g | 8 | Default std\_logic\_vector size of Green Pixels |
| blue\_width\_g | 8 | Default std\_logic\_vector size of Blue Pixels |
| req\_delay\_g | 1 | Number of clocks between the "*req\_data*" request to the "*data\_valid*" answer |
| req\_lines\_g | 3 | Number of lines to request from image transmitter, to hold in its FIFO |
| hor\_active\_pixels\_g | 800 | Active pixels per line |
| ver\_active\_lines\_g | 600 | Active lines per frame |
| hor\_left\_border\_g | 0 | Horizontal Left Border (Pixels) |
| hor\_right\_border\_g | 0 | Horizontal Right Border (Pixels) |
| hor\_back\_porch\_g | 88 | Horizontal Back Porch (Pixels) |
| hor\_front\_porch\_g | 40 | Horizontal Front Porch (Pixels) |
| hor\_sync\_time\_g | 128 | Horizontal Sync Time (Pixels) |
| ver\_top\_border\_g | 0 | Vertical Top Border (Lines) |
| ver\_buttom\_border\_g | 0 | Vertical Bottom Border (Lines) |
| ver\_back\_porch\_g | 23 | Vertical Back Porch (Lines) |
| ver\_front\_porch\_g | 1 | Vertical Front Porch (Lines) |
| ver\_sync\_time\_g | 4 | Vertical Sync Time (Lines) |

Table 3 – VESA Generator Generic Parameters

# Resources

Required resources, when synthesizing, using Synplify, for Altera's Cyclon II FPGA:

* Combinational logic cells: 161
* DFF – 68

**Maximum frequency**: 171MHz

# VESA Waves

The waves are divided into 6 sections:

1. **I/O RGB** – Input and Output RGB values
2. **Sync & Blank** – HSync, VSync and Blank signals
3. **Counters** – Internal Counters (Cannot be seen, when the component is treated as a black box)
4. **Data Handshake** – Handshake with the data provider
5. **Frame Size** – Left, Right, Upper and Lower frame's size
6. **Image Enables** – Enables of the VESA and the data provider

## Start of Active Video

Figure 5 – VESA Start of Active Video

In the above wave, Active Video comes together with the de-activation of the Blank signal (in this case, blank is active-low).

R\_out, G\_out and B\_out values are '0', until the active-video area is reached. It is possible to see in the wave, that during blanking, R\_in, G\_in and B\_in are not zero (23, 3, 3), but the R\_out, G\_out and B\_out values are '0'.

### Image Enables

There are two image enables flags:

1. **Vesa\_en** – When disabled – the transmitted RGB values will be 0, and the VSync, HSync and Blank signals will not be triggered.
2. **Image\_tx\_en** – When disabled, the VESA generator will keep transmitting data, HSync, VSync and blanking signals to the DVI. In this case, the RGB values that will be transmitted will be the default RGB values, as defined in the generic parameters (*red/green/blue\_default\_color\_g*).

These flags will be samples during the VSync activation. The data provider **must** supply these two signals during the VSync activation. Changing of these two flags when VSync is negated will be ignored.

## Between two frames

Figure 6 – VESA VSync and Blank example

In the wave above, it is possible to see the end of the first frame (Lower Frame, where the out RGB pixels are zero), and the start of the new frame (Upper Frame, where the out RGB pixels are zero).

## Complete Frame (Image)

Figure 7 – VESA Complete Frame

In the wave above, one frame (image) is being transmitted, wrapped by two VSync signals.

### ROI – Region of Interest

The ROI is the part of the image, which will be transmitted to the VESA generator, and will be shown on the screen, wrapped by upper, lower, left and right frame. In the wave above, only the left and right frames are being show. It can be seen, that the active-video area is wider that the ROI. The rest of the area is the frame.

In the wave above – the transmitted image's size is 640X480, but the active-video's size is 800X600. In this example – the left and right frames' sizes are () 80 pixels.

## Request for Data

Figure 8 – VESA Request for Data

VESA Generator requests for data from the data provider. *Data\_valid* signal is expected req*\_delay\_g* (generic parameter) clocks after the *req\_data* activation.

In the example above – *req\_delay\_g* = 1 (one clock delay).

## Prepare Data Provider Trigger

Figure 9 – VESA Prepare Data Provider Trigger

Once in a *[req\_lines\_g* X *hor\_active\_pixels\_g*] (= 3 X 800 = 2400 in this example) pixels, *req\_ln\_trig* signal will be activated for one clock cycle, to command the FIFO to prepare itself with *pixel\_req* pixels. It means that from the previous *req\_ln\_trig* activation to the current *req\_ln\_trig* activation, 2400 pixels (in this example) have been transmitted. This trigger aims to avoid a situation, where the VESA generator requests for data, but the data is not ready yet by the data provider. In this case – default color (according to the *red/green/blue\_default\_color\_g* generic parameters) pixel will replace the required pixel.

# Internal Implementation

There are two subjects regarding the internal VESA generator implementation, which will be explained here, in the documentation:

1. Horizontal and Vertical counters
2. Pipeline

## Horizontal and Vertical Counters

There are two internal signals, named '*hcnt'* and *'vcnt'*, which are the horizontal (*hcnt*) and vertical (*vcnt*) counters.

* **Hcnt** – Counts from 0 to (horizontal back porch + left frame + active pixels + left border + sync time + horizontal front porch). The *HSync* and *Blank* signals are activated and deactivated according to this signal.
* **Vcnt** – Counts from 0 to (vertical back porch + upper frame + active lines + bottom border + sync time + vertical front porch). The *VSync* and *Blank* signals are activated and deactivated according to this signal.

Figure 10 – Counters

## Pipeline

Two processes are using pipelines: The *req\_data\_proc* and the *blanking\_signal\_proc*. Instead of using a long 'if' condition, which will cause the VESA generator to work in a lower frequency, we use another process, which calculates each condition by itself, one clock before the calculation suppose to happen, and then the main combinational logic takes less time 🡺 higher frequency.

Figure 11 – Pipeline

### Code Example – Without Pipeline

Figure 12 – Code Example - Without Pipeline

### Code Example – With Pipeline

Figure 13 – Code Example - With Pipeline

# Performed Tests for VESA Generator

Two pictures where transmitted to the VESA generator. The first one is 800X600, and the second one is 640X480.

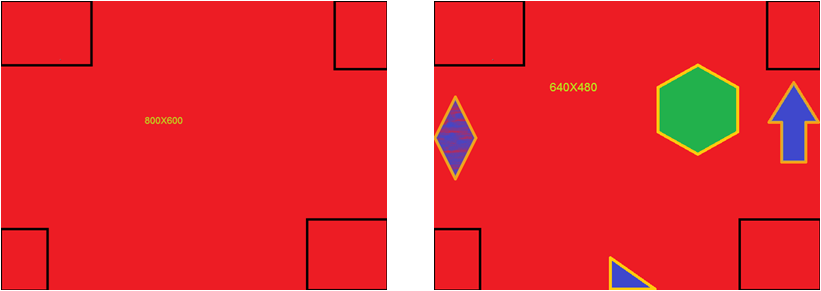


Figure 14 – Input Files

The following tests were performed on the VESA Generator:

1. 800X600 Image transmission, with normal parameters, according to VESA standard
2. 800X600 Image transmission, with different parameters than the VESA standard
3. 640X480 Image Transmission – 3 frames where transmitted
4. 640X480 Image Transmission, where the frame registers diminishes the input image
5. 640X480 Image Transmission, with different frame size
6. 640X480 Image Transmission, with different frame position
7. 640X480 Image Transmission, where the *vesa\_en* is negated before the frame's VSync, and activated during the frame transmission
8. 640X480 Image Transmission, where the *image\_tx\_en* is negated before the frame's VSync, and activated during the frame transmission
9. 640X480 Image Transmission, where the *vesa\_en* and *image\_tx\_*en are activated before the frame's VSync, and negated during the frame transmission.
10. Different *pixel\_req\_g* (1, 2 and 3 where tested)
11. Different *req\_lines\_g* (1 and 3 where tested)

## Output Files for Example

Figure 15 – Output Files for Example