

VGA Interface

Theoretical overview:

A standard VGA monitor consists of a grid of pixels that can be divided into rows and columns. A VGA monitor contains at least 480 rows, with 640 pixels per row, as shown in **Figure 1**. Each pixel can display various colours, depending on the state of the red, green, and blue signals.

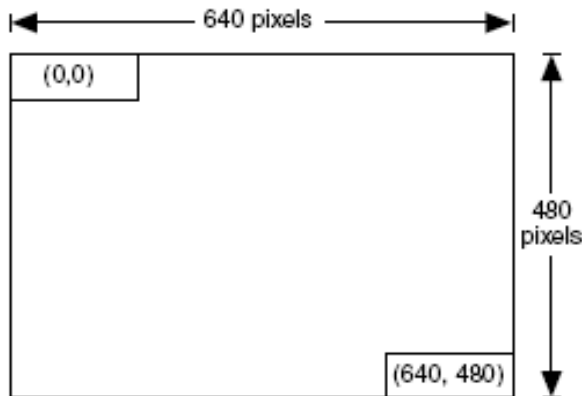


Figure 1.

Each VGA driver has an internal clock that determines when each pixel is updated. This clock operates at the VGA-specified frequency of 25.175 MHz. The monitor refreshes the screen in a prescribed manner that is partially controlled by the horizontal and vertical synchronization signals. The monitor starts each refresh cycle by updating the pixel in the top left-hand corner of the screen, which can be treated as the origin of an X-Y plane (see **Figure 1**).

After the first pixel is refreshed, the monitor refreshes the remaining pixels in the row. When the monitor receives a pulse on the horizontal synchronization, it refreshes the next row of pixels. This process is repeated until the monitor reaches the bottom of the screen. When the monitor reaches the bottom of the screen, the vertical synchronization pulses, causing the monitor to begin refreshing pixels at the top of the screen (i.e., at [0,0]). For the VGA monitor to work properly, it must receive data at specific times with specific pulses. Horizontal and vertical synchronization pulses must occur at specified times to synchronize the monitor while it is receiving colour data. Figures 2,3 shows the timing waveforms for the colour information with respect to the horizontal and vertical synchronization signals.

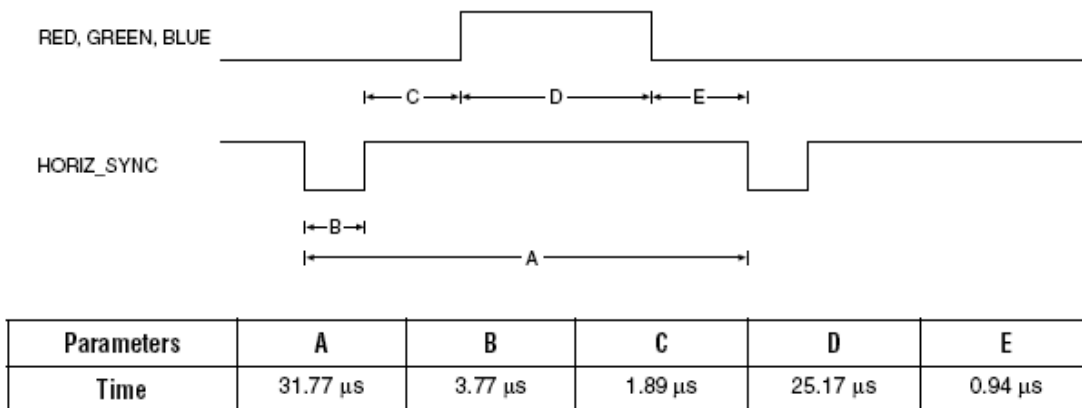


Figure 2.

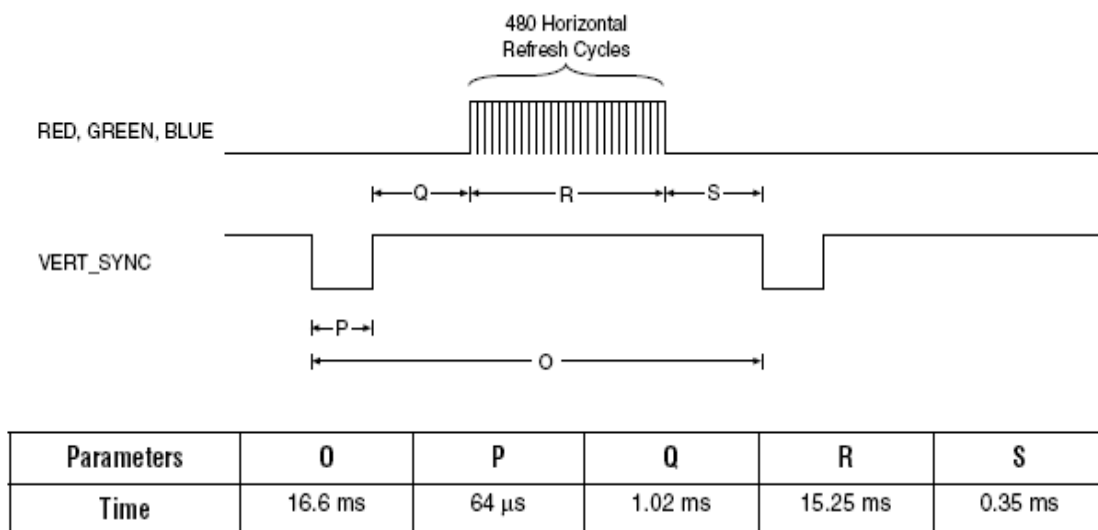


Figure 3.

The frequency of operation and the number of pixels that the monitor must update determines the time required to updating each pixel, and the time required to update the whole screen. The following equations roughly calculate the time required for the monitor to perform all of its functions.

$$T_{\text{pixel}} = 1/f_{\text{CLK}} = 1000/25.175 \approx 39.722 \text{ ns}$$

$$T_{\text{ROW}} = A+B+C+D+E = (T_{\text{pixel}} \times 640 \text{ pixels}) + \text{row} + \text{guard bands} \approx 31.77 \mu\text{s}$$

$$T_{\text{Screen}} = O=P+Q+R+S = (T_{\text{ROW}} \times 480 \text{ rows}) + \text{guard bands} \approx 16.6 \text{ ms}$$

Where: T_{pixel} = Time required to update a pixel; $f_{\text{CLK}} = 25.175 \text{ MHz}$

T_{ROW} = Time required to update one row

T_{Screen} = Time required to update the screen

B, C, E = Guard bands

P, Q, S = Guard bands

The monitor writes to the screen by sending red, green, blue, horizontal synchronization and vertical synchronization signals when the screen is at the expected location. Once the timing of the horizontal and vertical synchronization signals is accurate, the monitor only needs to keep track of the current location, so it can send the correct colour data to the pixel.

Interface:

The VGA interface allows the CYCLONE II device of DE2 board to control an external video monitor. This interface is composed of a simple diode-resistor network, DAC, and a 15-pin D-sub connector (labeled VGA), where the monitor can plug into the boards. Three VGA signals are **red**, **green**, and **blue**, while the other two signals are **horizontal** and **vertical** synchronization. Manipulating these signals allows write images to the monitor (see Altera DE2 user manual for detailed information).

Practical part:

The simply way to implement the VGA driver called "**the virtual screen**" (fig. 4). The size of virtual screen calculates by the following formula:

$$\text{Number of rows} = T_{\text{Screen}}/T_{\text{row}} = 16.6 \text{ ms} / 31.77 \mu\text{s} = 524.$$

$$\text{Pixels in row} = T_{\text{row}}/T_{\text{pixel}} = T_{\text{row}}/T_{\text{clk}} = 31.77 \mu\text{s} / 39.722 \text{ ns} = 800.$$

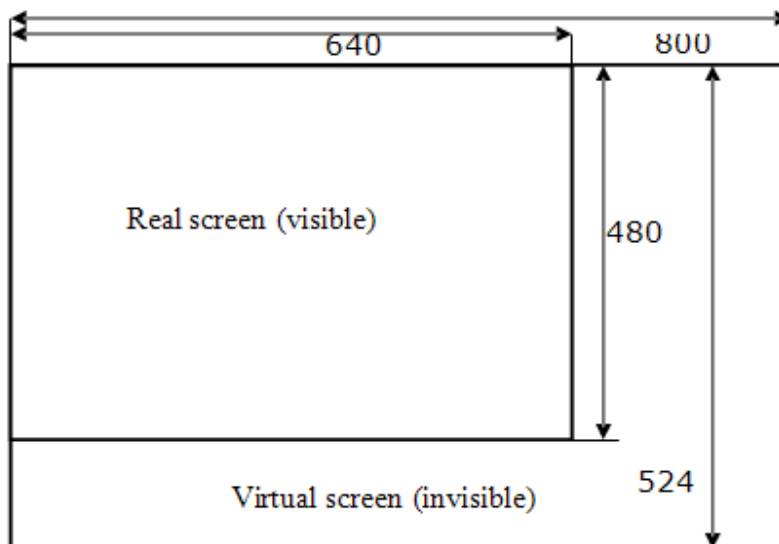


Figure 4

The position of synchronization signals on the virtual screen: horizontal synchronization signal comes low between 658 and 754 pixel; vertical synchronization signal comes low between 493 and 495 row. The colour of invisible area of virtual screen must be black. The current pixel colour provides by RGB method according to **table 1**.

Read	Green	Blue	Colour
'0'	'0'	'0'	BLACK
'0'	'0'	'1'	BLUE
'0'	'1'	'0'	GREEN
'0'	'1'	'1'	TURQUOISE
'1'	'0'	'0'	RED
'1'	'0'	'1'	PURPLE
'1'	'1'	'0'	YELLOW
'1'	'1'	'1'	WHITE

Table 1: RGB definition

The task:

Write the VHDL code of VGA driver that presents on VGA monitor white rectangle (128x128 pixels) on the black background. This rectangle should move from left side of the screen to right side and vice versa (in the centre of vertical alignment). Each movement (left to right and vice versa) must be completed in 1 sec.

Experiment requirements :

- ***Clock – 50 MHz Asynchronous reset from DE2 dip-switch***
- ***All outputs of design must be registered***
- ***Build the block diagram of your VGA driver***
- ***Write VHDL RTL code and TB for critical parts of design***
- ***Perform synthesis and timing analysis***

