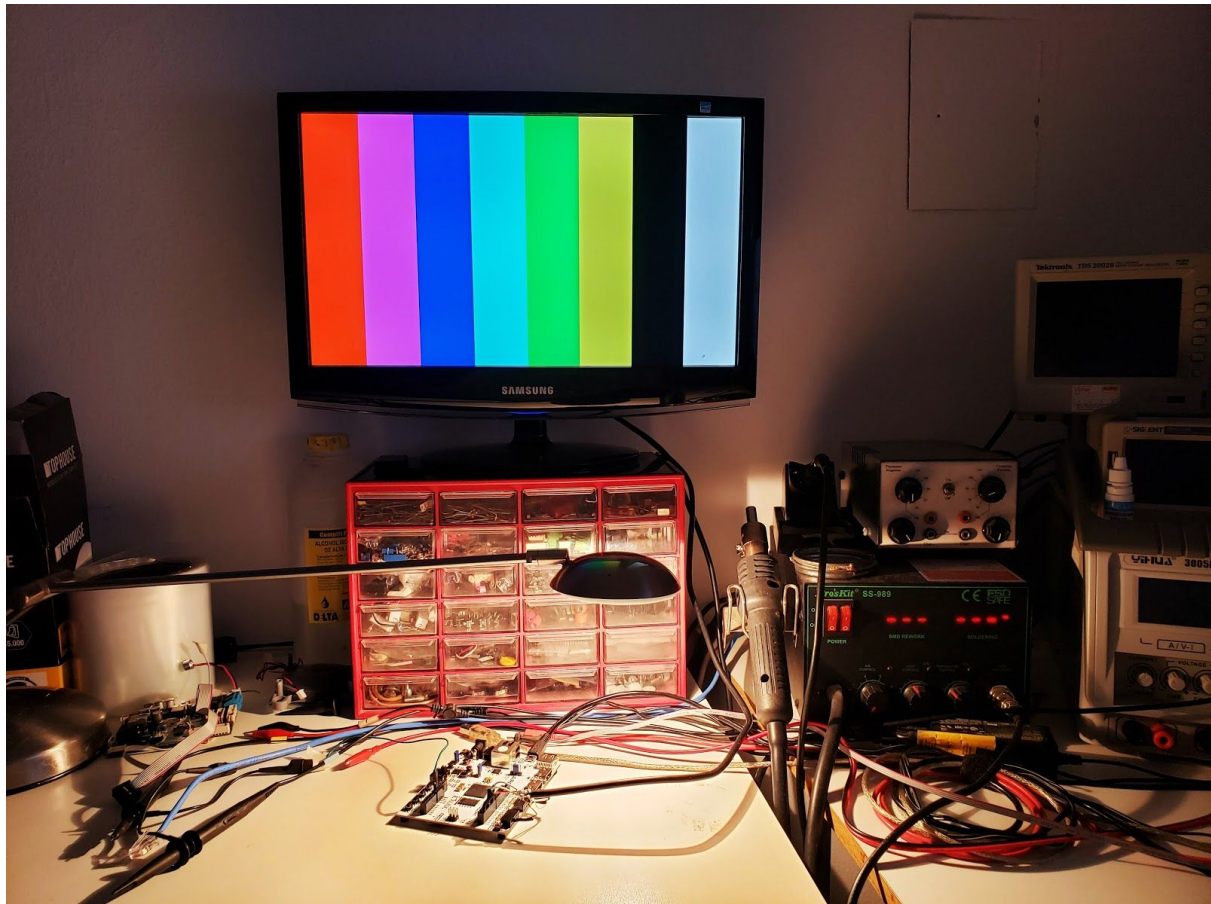


Controlador VGA

Metodología de trabajo que se utilizó para construir un controlador VGA en una FPGA de XILINX con lenguaje VHDL.

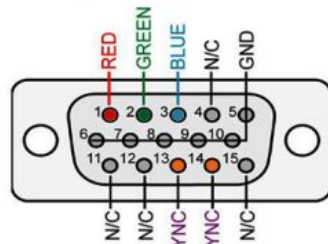
Este trabajo genera un patrón de video fijo y se realizó en el contexto de la materia de circuitos lógicos programables, carrera de especialización en sistemas embebidos de la UBA, año 2020.



VGA - Video Graphics Array (VGA)

Se denomina así a una matriz de gráficos de vídeo con las siguientes características

- Una pantalla estándar analógica de computadora.
- La resolución 640×480 píxeles.
- El conector de 15 contactos D subminiatura.
- La tarjeta gráfica que comercializó IBM por primera vez en 1988.



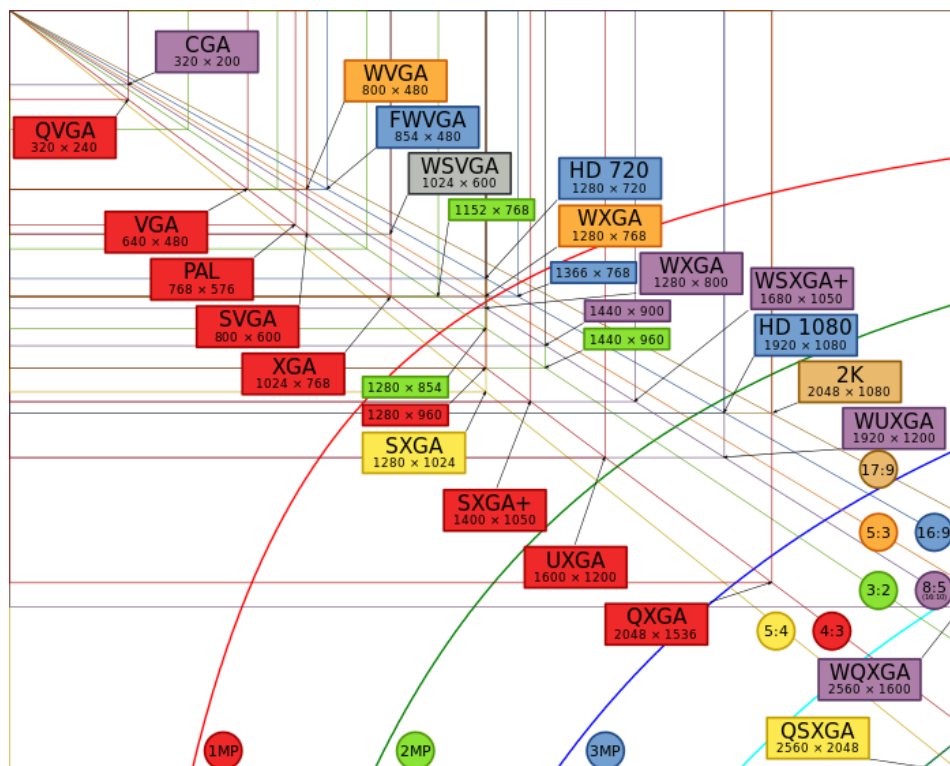
DETAIL FUNGSI DARI MASING-MASING PIN

- | | | |
|-----------------|---------------|------------------------|
| 1 - Red Video | 6 - Red GND | 11 - Monitor ID |
| 2 - Green Video | 7 - Green GND | 12 - DDC SDA |
| 3 - Blue Video | 8 - Blue GND | 13 - H _{sync} |
| 4 - Reserved | 9 - +5 V DC | 14 - V _{sync} |
| 5 - GND | 10 - Sync GND | 15 - DDC SCL |

Pin	Name	Dir	Description
1	RED	OUT	Red Video (75 ohm, 0.7 V p-p)
2	GREEN	OUT	Green Video (75 ohm, 0.7 V p-p)
3	BLUE	OUT	Blue Video (75 ohm, 0.7 V p-p)
4	ID2	IN	Monitor ID Bit 2
5	GND	-----	Ground
6	RGND	-----	Red Ground
7	GGND	-----	Green Ground
8	BGND	-----	Blue Ground
9	KEY	-	Key (No pin)
10	SGND	-----	Sync Ground
11	ID0	IN	Monitor ID Bit 0 GND=Color, NC=Mono
12	ID1 or SDA	IN	Monitor ID Bit 1 NC=Color, GND=Mono Some systems only uses ID0 for monitor ID
13	HSYNC or CSYNC	OUT	Horizontal Sync (or Composite Sync)
14	VSYSN	OUT	Vertical Sync
15	ID3 or SCL	IN	Monitor ID Bit 3

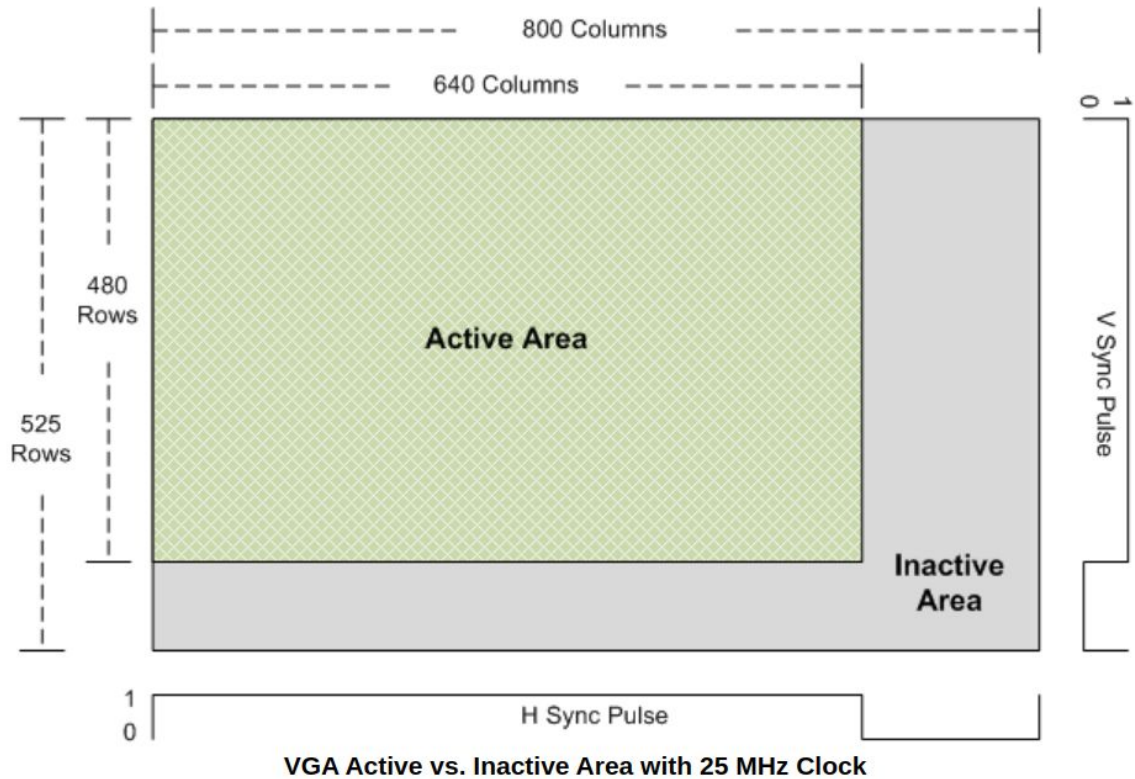
VGA PINOUT

A partir del VGA, se amplía por otras organizaciones y fabricantes, siguiendo el mismo principio. De esta manera el trabajo sirve para generar otras resoluciones de video como por ejemplo 800x600 (SVGA).



Señales de sincronismo

En la siguiente imagen se puede ver la relación que hay entre las señales de sincronismo vertical y horizontal. La zona activa es la información que se visualiza en pantalla y las zonas grises corresponde a la información utilizada para sincronizar la imagen.



El sincronismo horizontal está relacionado con la posición en pantalla de las señales de video. Cuando se terminan de mostrar todos los píxeles correspondientes a una fila aparecen 3 tiempos importantes. Primero el Front porch, luego un estado bajo denominado pulso de sincronismo y un último tiempo en alto denominado back porch. En este momento se muestra la información correspondiente a la siguiente fila.

Cuando se completan todas las filas, se produce de manera similar al horizontal, el pulso de sincronismo vertical. Éste marca la frecuencia de muestreo de la pantalla.

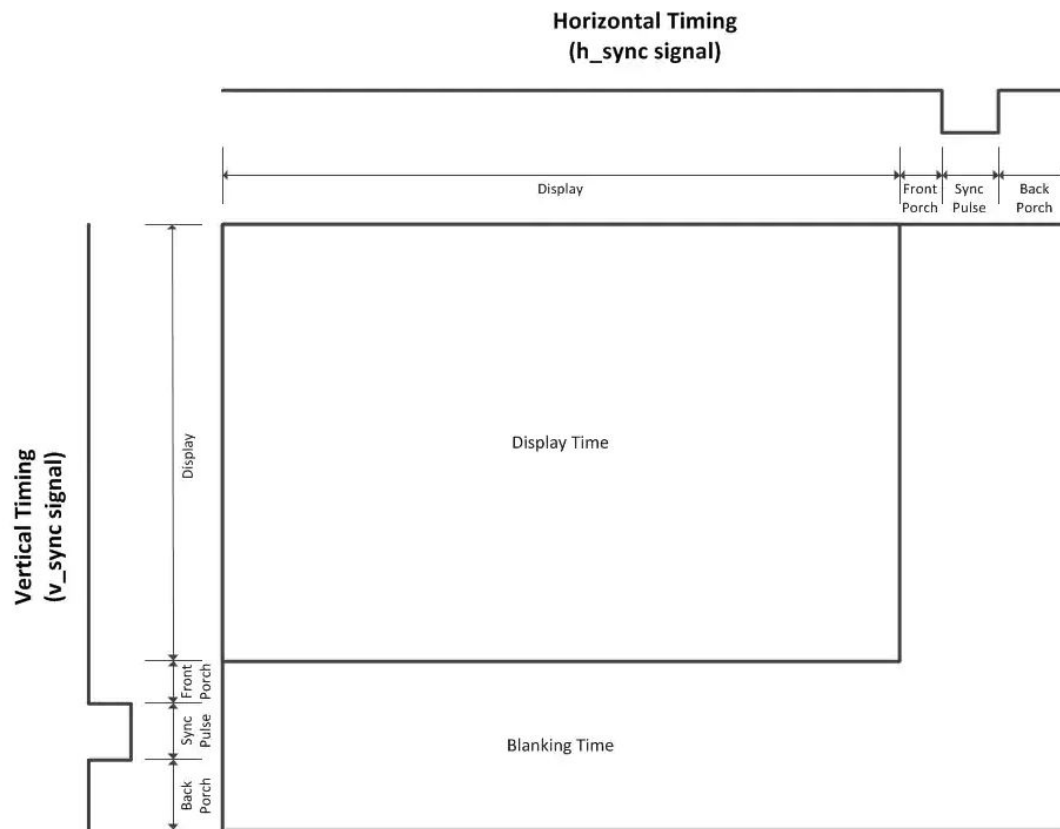


Figure 3. Signal Timing Diagram

VGA timing

En la siguiente imagen extraída de la página <http://tinyvga.com/vga-timing> se pueden ver los tiempos que deben cumplirse para las diferentes resoluciones de video.

En particular se utilizó la de la siguiente imagen.

VGA Signal 640 x 480 @ 60 Hz Industry standard timing

General timing

Screen refresh rate	60 Hz
Vertical refresh	31.46875 kHz
Pixel freq.	25.175 MHz

Horizontal timing (line)

Polarity of horizontal sync pulse is negative.

Scanline part	Pixels	Time [μs]
Visible area	640	25.422045680238
Front porch	16	0.63555114200596
Sync pulse	96	3.8133068520357
Back porch	48	1.9066534260179
Whole line	800	31.777557100298

Vertical timing (frame)

Polarity of vertical sync pulse is negative.

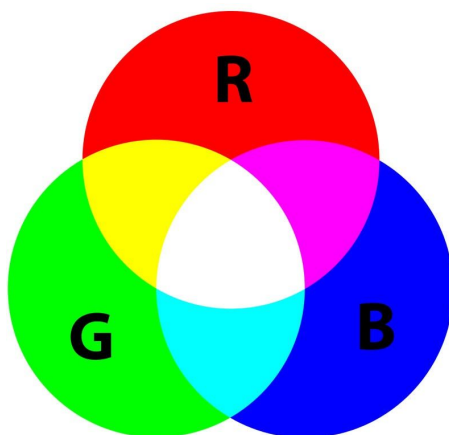
Frame part	Lines	Time [ms]
Visible area	480	15.253227408143
Front porch	10	0.31777557100298
Sync pulse	2	0.063555114200596
Back porch	33	1.0486593843098
Whole frame	525	16.683217477656

Para la implementación en circuito digital se utilizó la información del siguiente cuadro. Se puede ver que las magnitudes correspondientes a la señal horizontal están expresados en pulsos de reloj o también llamado “píxel clock” y, las correspondientes a la vertical en “líneas”.

Pixel Clock (MHz)	Horizontal (pixel clocks)				Vertical (rows)				Polarity	
	Display	Front Porch	Sync Pulse	Back Porch	Display	Front Porch	Sync Pulse	Back Porch	h_sync	v_sync
25.175	640	16	96	48	480	10	2	33	n	n
44.9	1024	8	176	56	768	0	8	41	p	p
65	1024	24	136	160	768	3	6	29	n	n
234	1920	128	208	344	1440	1	3	56	n	p
297	1920	144	224	352	1440	1	3	56	n	p

Señales de video

Si bien se generan 3 bits para cada señal de video, en las pruebas se utilizó un 1 bits para cada color. En la siguiente imagen se pueden ver los posibles colores que se pueden formar con esta información.



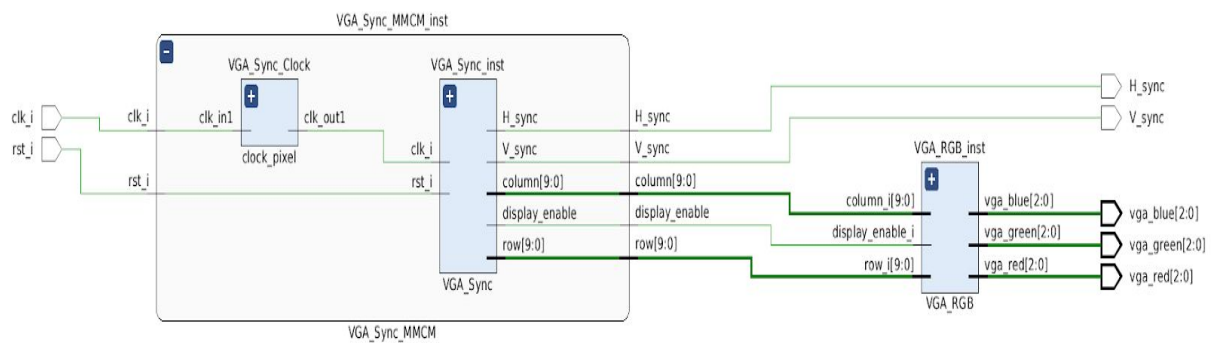
Descripción del hardware

Se pueden encontrar las siguientes entidades. VGA_Sync es la que genera los pulsos de sincronismo en base a la señal de reloj. La entidad VGA_Sync_MMCM agrega a la anterior un MMCM de la herramienta vivado, para poder escalar el reloj de la placa (125 MHz) a la de pixel que necesitamos (25.175 MHz). La entidad VGA_RGB es la que genera las señales de video.

Es importante destacar que en la zona fuera de pantalla, estas señales deben ser nulas para no producir ruido o sincronizaciones incorrectas.

Por último, una entidad VGA_background que corresponde a nuestro circuito completo, es decir al “top level”.

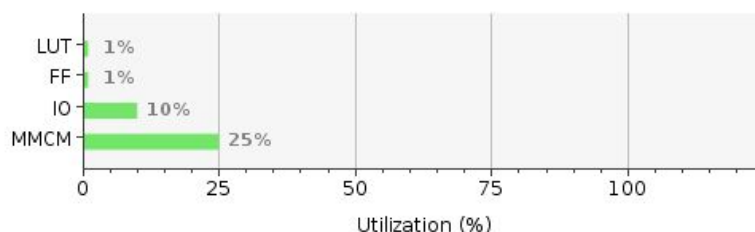
A continuación se puede observar el esquemático del RTL (register transfer level)



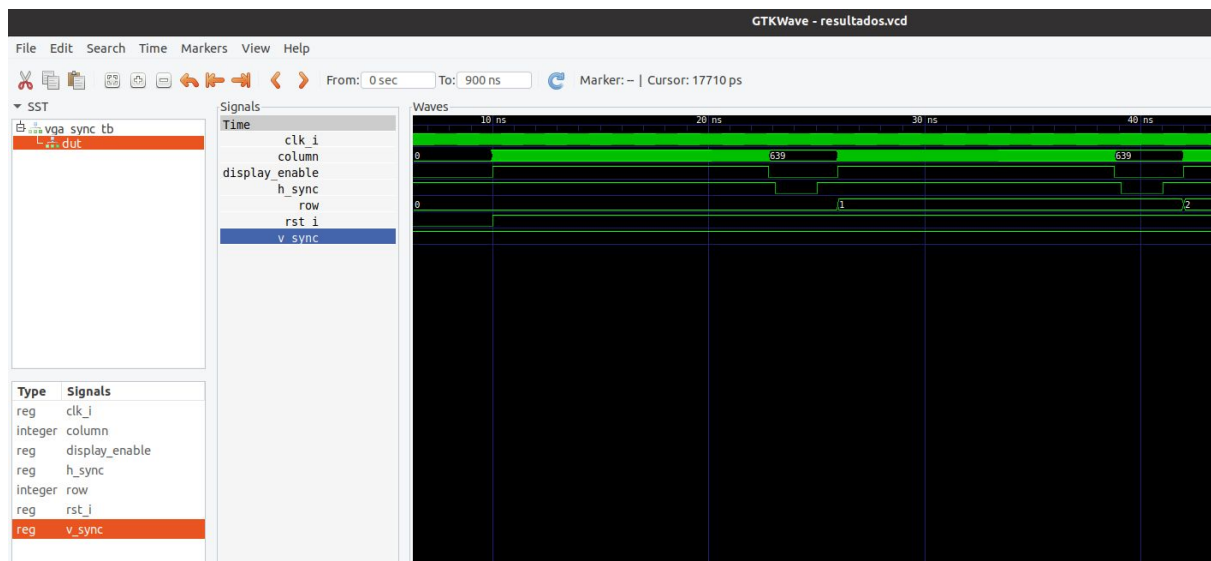
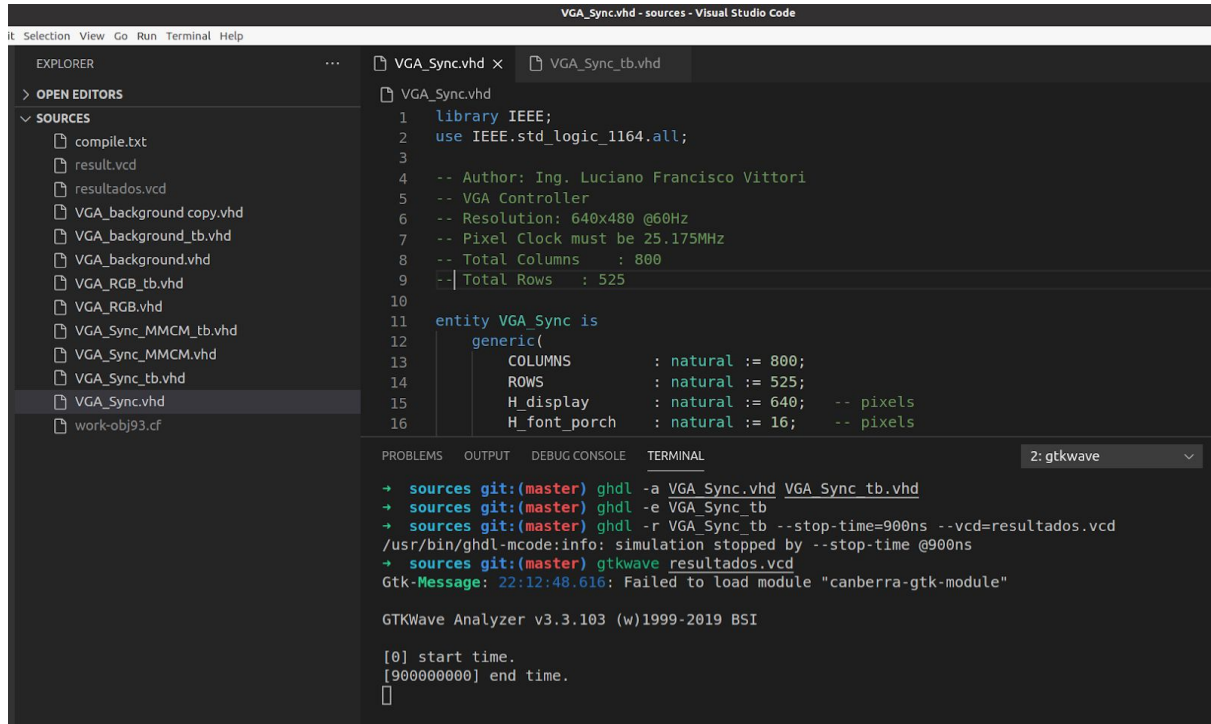
Reporte de utilización de la herramienta Vivado - Placa ARTY Z-20

Summary

Resource	Utilization	Available	Utilization %
LUT	49	53200	0.09
FF	37	106400	0.03
IO	13	125	10.40
MMCM	1	4	25.00



Si bien la síntesis se realizó en la herramienta VIVADO, la mayor parte se pudo desarrollar en el editor de texto VISUAL CODE STUDIO, analizando y compilando con GHDL y GTKWAVE para la visualización de los testbench.



The screenshot shows the GTKWave IDE with the source code for `VGA_Sync_tb.vhd` in the editor. The code defines testbench signals and a DUT (Device Under Test) for the VGA_Sync module. The terminal window shows the command-line execution of the simulation using `ghdl` and `gtkwave`.

```

40 signal rst_tb      : std_logic := '0';
41 signal H_sync_tb   : std_logic;
42 signal V_sync_tb   : std_logic;
43 signal column_tb   : natural range 0 to COLUMNS_tb;
44 signal row_tb      : natural range 0 to ROWS_tb;
45
46 begin
47     -- Parte descriptiva:
48     --clk_tb  <= not clk_tb after 19.86 ns; -- 25.175 MHz
49     clk_tb  <= not clk_tb after 10 ps;    -- 500 MHz @Only for simulation
50     rst_tb  <= '1' after 10 ns;
51
52     DUT: VGA_Sync
53         generic map(
54             COLUMNS => COLUMNS_tb,
55             ROWS      => ROWS_tb
56         )
57 
```

Terminal Output:

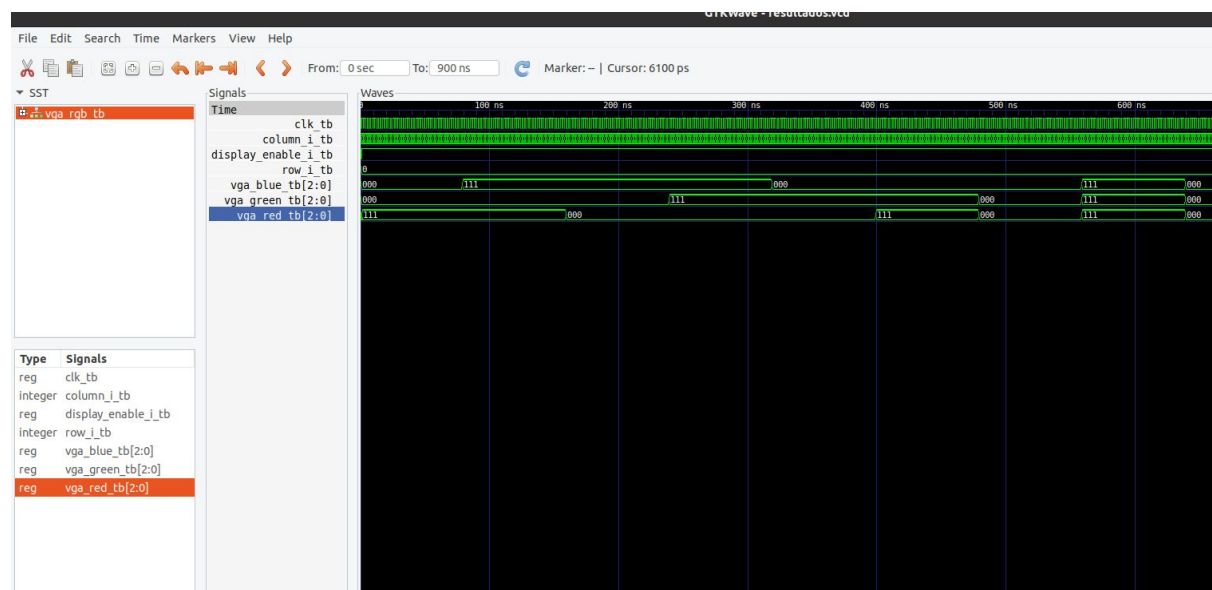
```

+ sources git:(master) ghdl -a VGA_RGB.vhd VGA_RGB_tb.vhd
+ sources git:(master) ghdl -e VGA_RGB_tb
+ sources git:(master) ghdl -r VGA_RGB_tb --stop-time=900ns --vcd=resultados.vcd
/usr/bin/ghdl-mcode:info: simulation stopped by --stop-time @900ns
+ sources git:(master) gtkwave resultados.vcd
Gtk-Message: 22:17:08.834: Failed to load module "canberra-gtk-module"

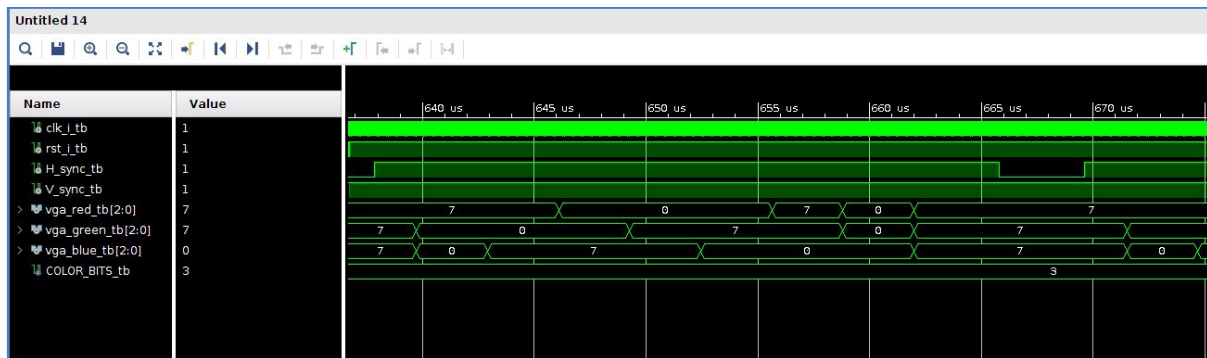
GTKWave Analyzer v3.3.103 (w)1999-2019 BSI

[0] start time.
[900000000] end time.
[]

```



A continuación se muestra una simulación en vivo del bloque final.



Opciones para utilizar varios bits de video

A continuación se muestran dos ejemplos de cómo proseguir para una señal de video más detallada.

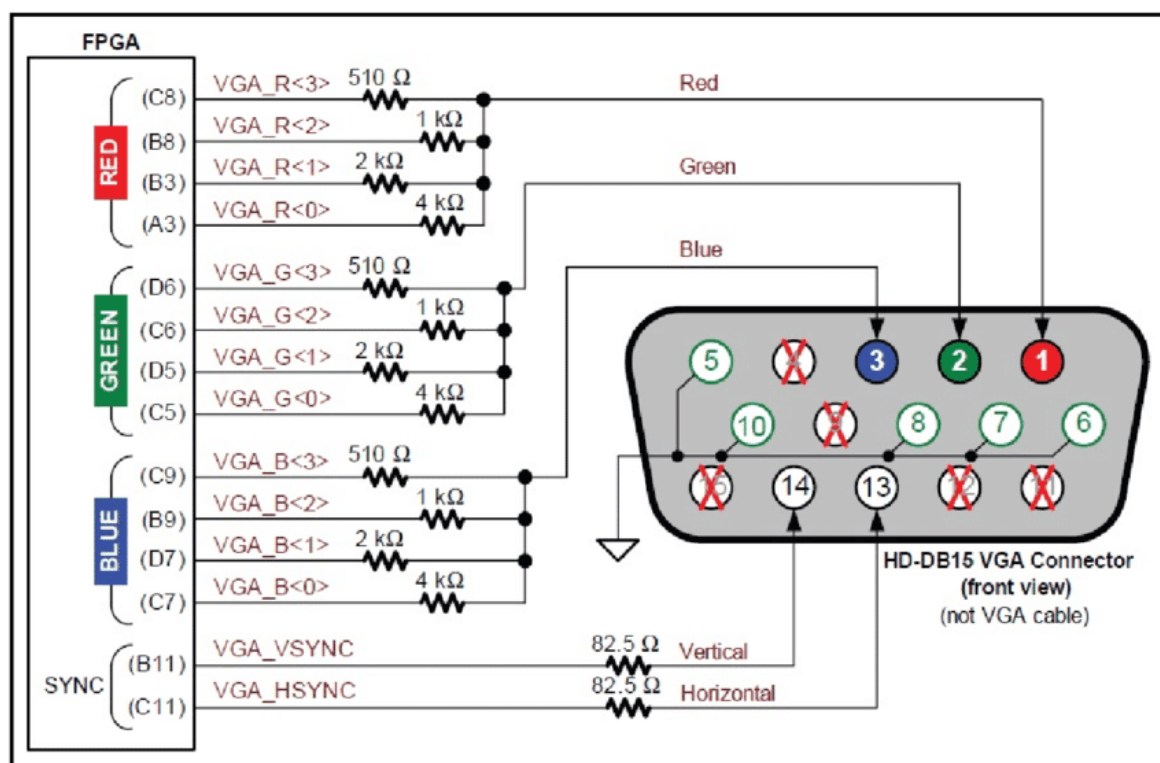


Imagen extraída de

https://www.researchgate.net/publication/312984160_FPGA_Based_Real_Time_Iris_Recognition_Signature

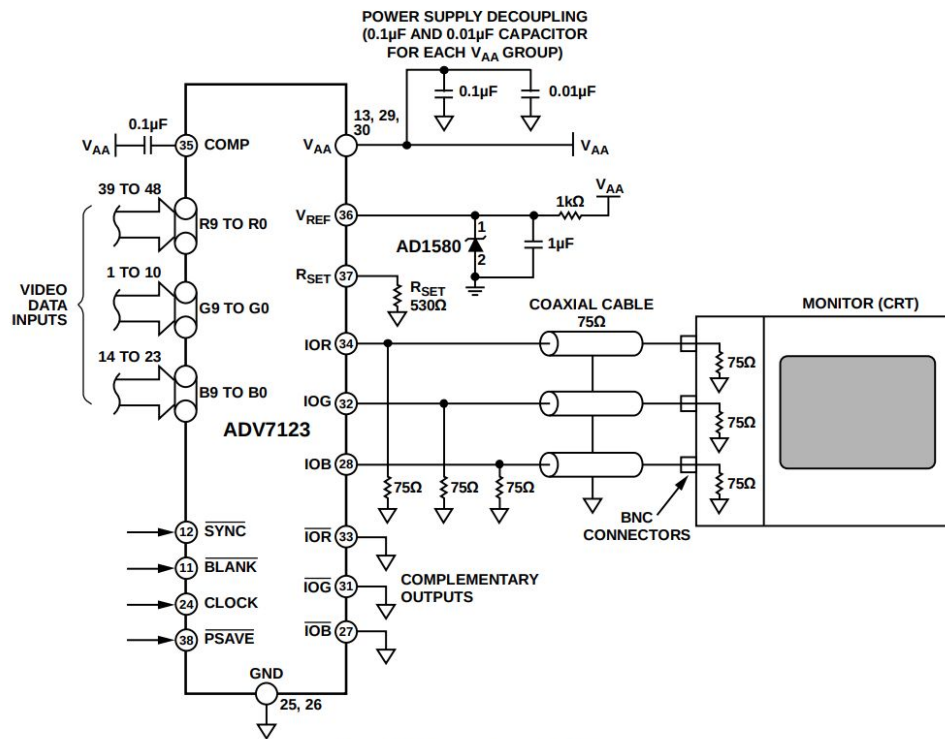


Imagen extraída del datasheet

<https://www.analog.com/media/en/technical-documentation/data-sheets/ADV7123.pdf>