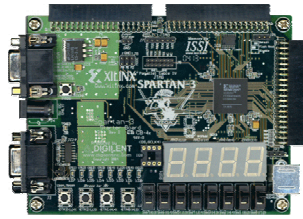


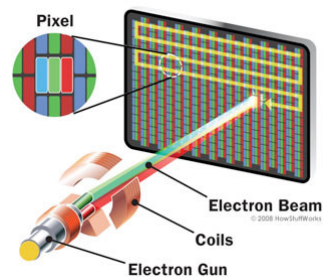
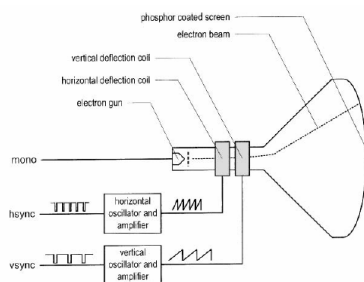
Periféricos II

VGA



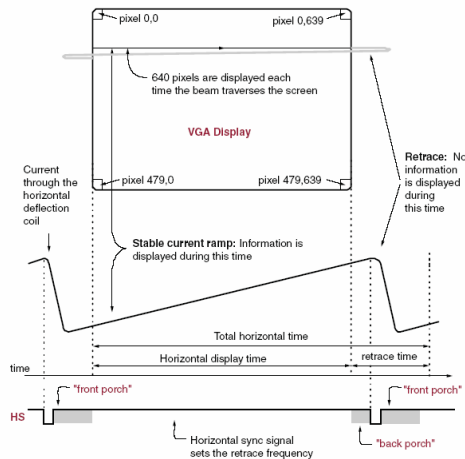
Lucas Leiva – INTIA
lleiva@exa.unicen.edu.ar
UNICEN – Octubre 2012

Operación Básica CRT



- ❑ *Dos cátodos direccionan los electrones del cañón*
- ❑ *Estos cátodos permiten generar imágenes a partir del barrido horizontal y vertical*

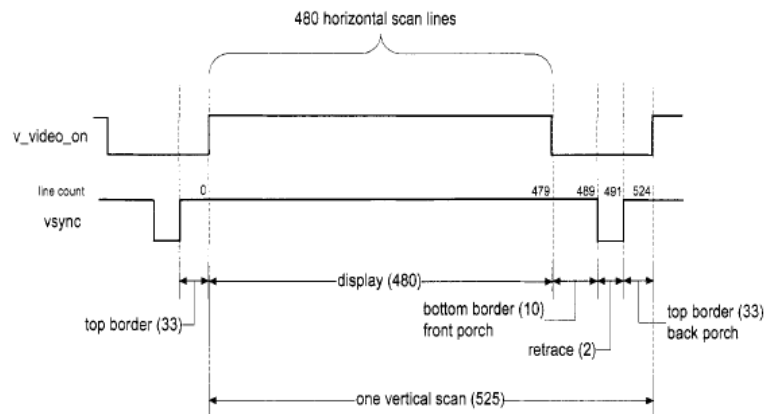
Sincronismo horizontal (640x480)



Sincronismo horizontal - timing

- ❑ **Display**
 - región de presentación de valores de imagen (640 pixels).
- ❑ **Retrace**
 - región de retorno a la izquierda de la pantalla. La señal de video debe deshabilitarse con negro (96 pixels).
- ❑ **Right border**
 - región que forma el borde derecho. También conocido como *front porch*. La señal de video debe estar inactiva (16 pixels).
- ❑ **Left border**
 - Región que forma el borde izquierdo. También conocida como *backporch*. La señal de video debe estar inactiva (48 pixels).

Sincronismo vertical (640x480)



Sincronismo vertical - timing

- **Display**
 - región de presentación de las líneas en la pantalla (480 líneas).
- **Retrace**
 - región de retorno a la parte superior de la pantalla. La señal de video debe estar inactiva (2 líneas).
- **Bottom border**
 - región que forma el borde inferior de la imagen. También conocida como *frontporch*. La señal de video debe estar inactiva (10 líneas).
- **Top border**
 - región que forma el borde superior de la imagen. También conocida como *backporch*. La señal de video debe estar inactiva (33 líneas).

Timing para 640x480, 60 Hz

Symbol	Parameter	Vertical Sync			Horizontal Sync	
		Time	Clocks	Lines	Time	Clocks
T_S	Sync pulse time	16.7 ms	416,800	521	32 μ s	800
T_{DISP}	Display time	15.36 ms	384,000	480	25.6 μ s	640
T_{PW}	Pulse width	64 μ s	1,600	2	3.84 μ s	96
T_{FP}	Front porch	320 μ s	8,000	10	640 ns	16
T_{BP}	Back porch	928 μ s	23,200	29	1.92 μ s	48

Constantes VGA

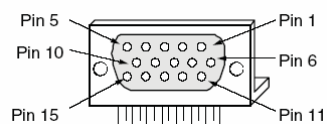
Format	Pixel Clock (MHz)	Horizontal (in Pixels)				Vertical (in Lines)			
		Active Video	Front Porch	Sync Pulse	Back Porch	Active Video	Front Porch	Sync Pulse	Back Porch
640x480, 60Hz	25.175	640	16	96	48	480	11	2	31
640x480, 72Hz	31.500	640	24	40	128	480	9	3	28
640x480, 75Hz	31.500	640	16	96	48	480	11	2	32
640x480, 85Hz	36.000	640	32	48	112	480	1	3	25
800x600, 56Hz	38.100	800	32	128	128	600	1	4	14
800x600, 60Hz	40.000	800	40	128	88	600	1	4	23
800x600, 72Hz	50.000	800	56	120	64	600	37	6	23
800x600, 75Hz	49.500	800	16	80	160	600	1	2	21
800x600, 85Hz	56.250	800	32	64	152	600	1	3	27
1024x768, 60Hz	65.000	1024	24	136	160	768	3	6	29
1024x768, 70Hz	75.000	1024	24	136	144	768	3	6	29
1024x768, 75Hz	78.750	1024	16	96	176	768	1	3	28
1024x768, 85Hz	94.500	1024	48	96	208	768	1	3	36

Imágenes en RGB (8 colores)

- 3 bits (RGB) que permiten generar imágenes de 8 colores
- Si el DAC posee más bits de profundidad, el espacio de colores se incrementa

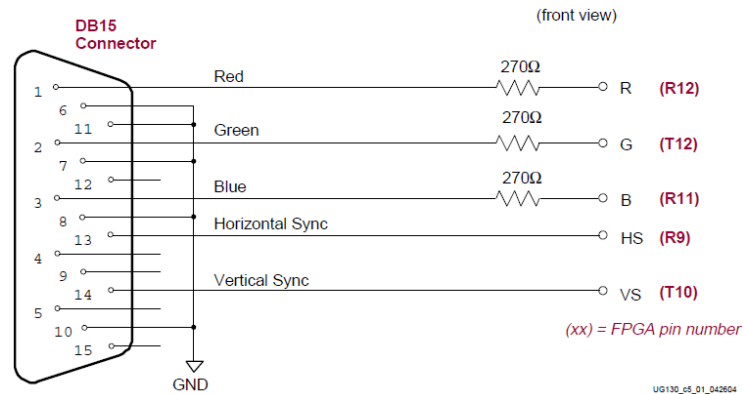
VGA_RED	VGA_GREEN	VGA_BLUE	Resulting Color
0	0	0	Black
0	0	1	Blue
0	1	0	Green
0	1	1	Cyan
1	0	0	Red
1	0	1	Magenta
1	1	0	Yellow
1	1	1	White

Conector VGA



PIN	SIGNAL NAME	DESCRIPTION
1	RED	Red video signal
2	GREEN	Green video signal
3	BLUE	Blue video signal
4	MONID(0)	Monitor ID signal 0
5	GND DDC	Return
6, 7, 8	AGND_VID	Analog video ground
9	+5V_IO 5 V	Power for I/O device
10	GND	HSYNC and VSYNC ground
11	VGA_ID	VGA ID signal
12	MONID(2)	Monitor ID signal 2
13	HSYNC	Horizontal synchronization signal
14	VSYNC	Vertical synchronization signal
15	MONID(1)	Monitor ID signal 1

Configuración VGA Spartan 3 SK



Controlador VGA

```
entity vga_sync is
  port(
    clk, reset: in std_logic;
    hsync, vsync: out std_logic;
    video_en, p_click: out std_logic;
    pixel_x, pixel_y: out std_logic_vector (9 downto 0)
  );
end vga_sync;

architecture arch of vga_sync is
  -- VGA 640-by-480 sync parameters
  constant HD: integer:=640; --horizontal display area
  constant HF: integer:=16; --h. front porch
  constant HB: integer:=48; --h. back porch
  constant HA: integer:=56; --h. retrace
  constant VD: integer:=480; --vertical display area
  constant VF: integer:=10; --v. front porch
  constant VB: integer:=38; --v. back porch
  constant VA: integer:=2; --v. retrace
  -- mod-2 counter
  signal mod2_reg, mod2_next: std_logic;
  -- sync counters
  signal v_count_reg, v_count_next: unsigned(9 downto 0);
  signal h_count_reg, h_count_next: unsigned(9 downto 0);
  -- output buffer
  signal v_sync_reg, h_sync_reg: std_logic;
  signal v_sync_next, h_sync_next: std_logic;
  -- status signal
  signal h_end, v_end, pixel_click: std_logic;
end arch;
```

- mod2**
 - señal de clock a 50 Mhz
- v_count y h_count**
 - contadores de sincronismos
- h_end y v_end**
 - indican el fin de linea o imagen

Controlador VGA - Registros

```
process (clk, reset)
begin
  if reset='1' then
    mod2_reg <= '0';
    v_count_reg <= (others=>'0');
    h_count_reg <= (others=>'0');
    v_sync_reg <= '0';
    h_sync_reg <= '0';
  elsif (clk'event and clk='1') then
    mod2_reg <= mod2_next;
    v_count_reg <= v_count_next;
    h_count_reg <= h_count_next;
    v_sync_reg <= v_sync_next;
    h_sync_reg <= h_sync_next;
  end if;
end process;
```

□ Proceso que actualiza los registros del controlador

□ Señales h_sync y v_sync registradas para evitar glitches

Controlador VGA - Señales

```
-- mod-2 circuit to generate 25 MHz enable tick
mod2_next <= not mod2_reg;

-- 25 MHz pixel tick
pixel_tick <= '1' when mod2_reg='1' else '0';

-- status
h_end <= -- end of horizontal counter
'1' when h_count_reg=(HD+HF+HB+HR-1) else --799
'0';
v_end <= -- end of vertical counter
'1' when v_count_reg=(VD+VF+VB+VR-1) else --524
'0';
h_sync_next <=
'1' when (h_count_reg>=(HD+HF)) --656
and (h_count_reg<=(HD+HF+HR-1)) else --751
'0';
v_sync_next <=
'1' when (v_count_reg>=(VD+VF)) --490
and (v_count_reg<=(VD+VF+VR-1)) else --491
'0';
-- video on/off
video_on <=
'1' when (h_count_reg<HD) and (v_count_reg<VD) else
'0';
```

Controlador VGA – Contador hsync

```
process (h_count_reg,h_end,pixel_tick)
begin
  if pixel_tick='1' then -- 25 MHz tick
    if h_end='1' then
      h_count_next <= (others=>'0');
    else
      h_count_next <= h_count_reg + 1;
    end if;
  else
    h_count_next <= h_count_reg;
  end if;
end process;
```

- En cada pulso de píxel incrementa el contador.
 - Si terminó la línea lo reinicializa
-

Controlador VGA – Contador vsync

```
process (v_count_reg,h_end,v_end,pixel_tick)
begin
  if pixel_tick='1' and h_end='1' then
    if (v_end='1') then
      v_count_next <= (others=>'0');
    else
      v_count_next <= v_count_reg + 1;
    end if;
  else
    v_count_next <= v_count_reg;
  end if;
end process;
```

- Al final de cada línea incrementa el contador
 - Si terminó el barrido, lo reinicializa
-

Problemas en Spartan 3 SK

- Solo posee 192 kbits de Block RAM
 - Imagen de 640x480 en 8 colores requiere de 900 kbits
 - Solución:
 - Utilizar bancos de memorias externas (8Megabits)
 - Graficar solo una porción de la pantalla
-