

Affichage VGA

1 Le matériel

VGA (Video Graphics Adapter) est une norme de signal vidéo que l'on trouve principalement dans les ordinateurs individuels. Ce signal vidéo est constitué de 5 signaux, 3 analogiques avec des niveaux de 0.7 à 1.0 volt pour chacune des couleurs Rouge, Vert, Bleu (RGB) accompagnés de 2 signaux logiques que sont synchros horizontale et verticale.

1.1 Le moniteur de visualisation

Le principal composant du moniteur est le tube cathodique dont l'écran est la face visible pour l'utilisateur. Le faisceau d'électron est dévié selon un parcours horizontal de gauche à droite de l'écran, puis retour à la ligne avec décalage vers la bas et ceci jusqu'à atteindre le point le plus bas à droite. Dans ce parcours, l'information RGB est utilisée pour contrôler l'intensité du faisceau d'électrons. Cela donnera selon le cas un point de couleur rouge, bleu ou vert (couleurs primaires) plus ou moins intenses, l'œil humain percevant une couleur résultante de ces trois points très proches physiquement.

En VGA standard, l'écran contient 640 par 480 **pixels** (picture elements) qui doivent être balayés à une fréquence supérieure à 30 Hz si l'on veut éviter un effet de scintillement. On trouve en général sur les PC une fréquence de rafraîchissement de 60 Hz (60 images par seconde). Les moniteurs du commerce fonctionnent tous dans une certaine plage autour de cette fréquence de référence. Si l'on veut évaluer approximativement la fréquence pixel, il suffit de diviser 1/60 par le produit 680X480. Cela donne 50 ns environ entre 2 pixels. Bien que l'horloge de la carte Memec soit à 24 MHz, on fera les calculs de timing à 25 Mhz, le moniteur s'accordant sur la fréquence qu'il recevra.

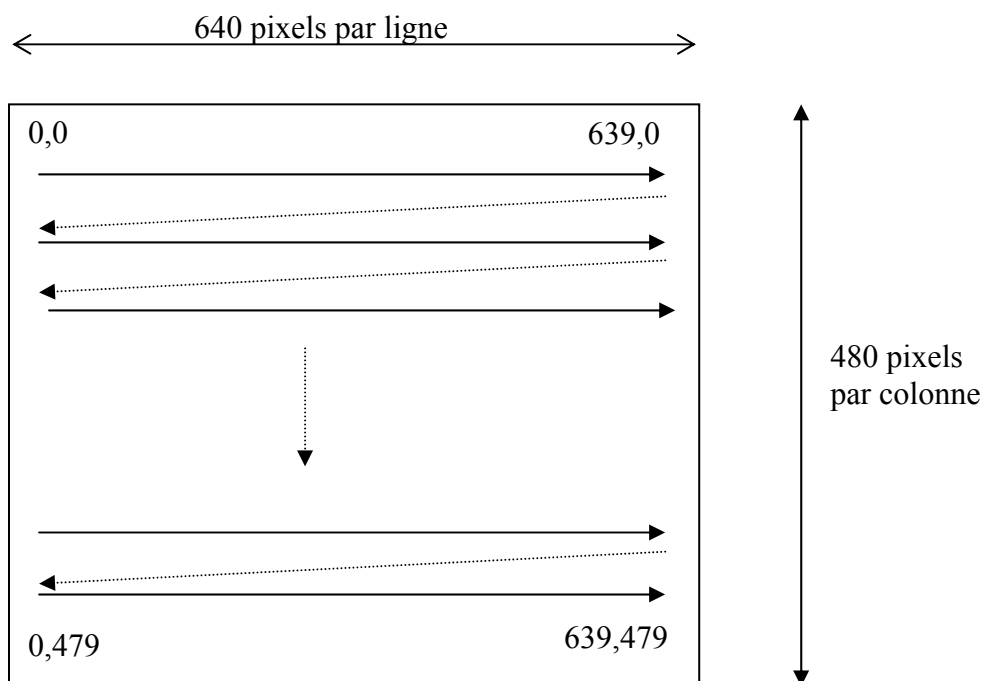


Figure 1 :Image 640 X 480

La trame VGA étant découpée en 525 lignes une fréquence image de 60 Hertz entraîne une fréquence ligne de 31,5 kHz. Une ligne correspond pour une fréquence pixel à 25 MHz à un comptage de 794 points.

Le balayage vertical se décompose en une synchro (2 lignes) suivi de 32 lignes noires suivi des 480 lignes de pixel suivies de 11 lignes noires.

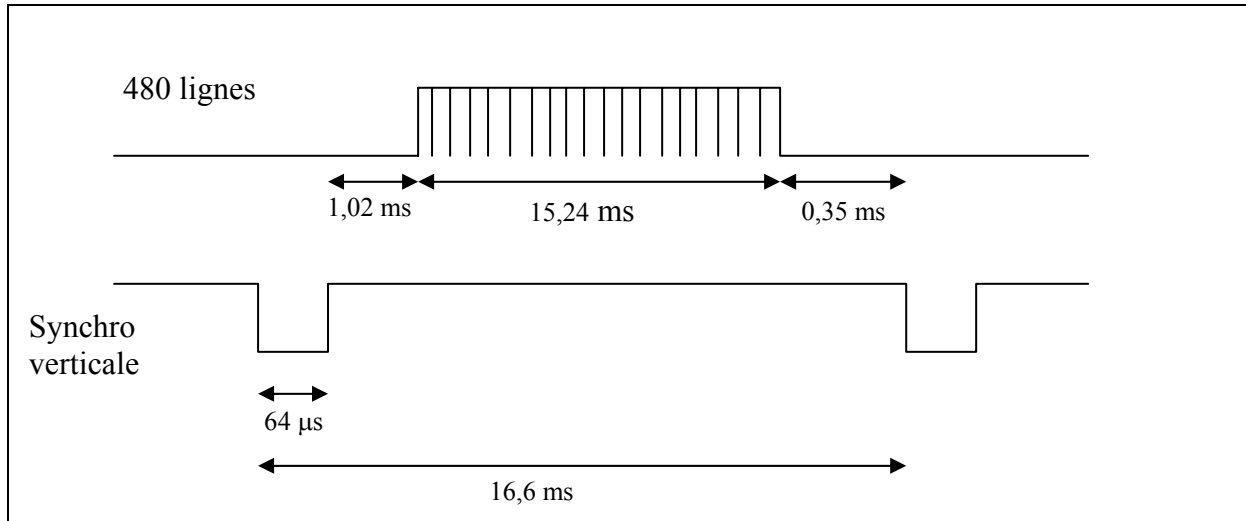


Figure 2 : timing de la synchro verticale à 60 Hz

Le balayage horizontal se décompose en un signal de synchronisation (95 points) un pallier avant noir (43 points), les 640 pixels et un pallier arrière (16 points).

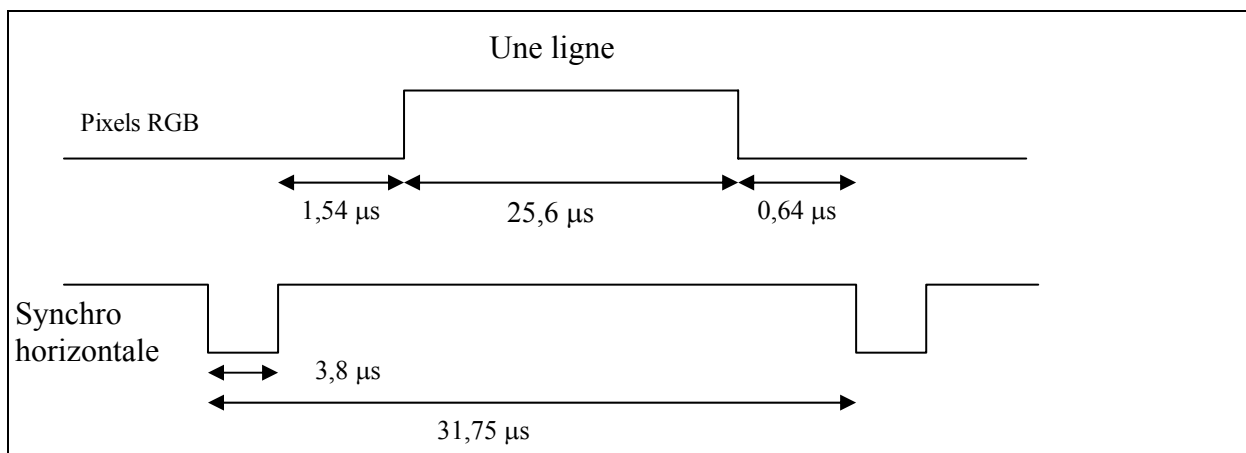
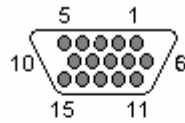


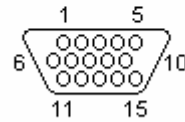
Figure 3 : timing de la synchro horizontale (31,5 kHz)

1.2 Le connecteur

Il permet de relier le système générateur de vidéo (le plus souvent l'ordinateur) au moniteur



(Connecteur VGA Femelle)



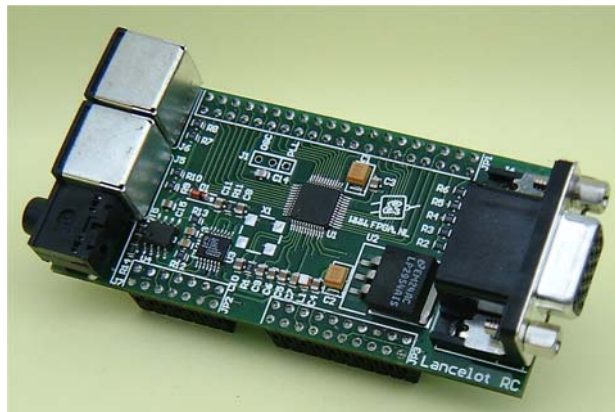
(Connecteur VGA Mâle)

Les signaux véhiculés sont ceux du tableau ci-après :

Broche	Nom	Direction	Description
1	RED	→	Vidéo rouge (75 ohms, 0.7 volt)
2	GREEN	→	Vidéo verte (75 ohms, 0.7 volt)
3	BLUE	→	Vidéo bleu (75 ohms, 0.7 volt)
4	RES		Réservé
5	GND	—	Masse
6	RGND	—	Masse rouge
7	GGND	—	Masse verte
8	BGND	—	Masse bleu
9	+5V	→	+5V continu
10	SGND	—	Masse synchro
11	ID0	←	Monitor ID Bit 0 (optionnel)
12	SDA	↔	Ligne de données série DDC
13	HSYNC ou CSYNC	→	Synchro horizontale ou composite
14	VSYNC	→	Synchro verticale
15	SCL	↔	Ligne d'horloge DDC (Display Data Channel)

Tableau 1 : Signaux sur connecteur VGA

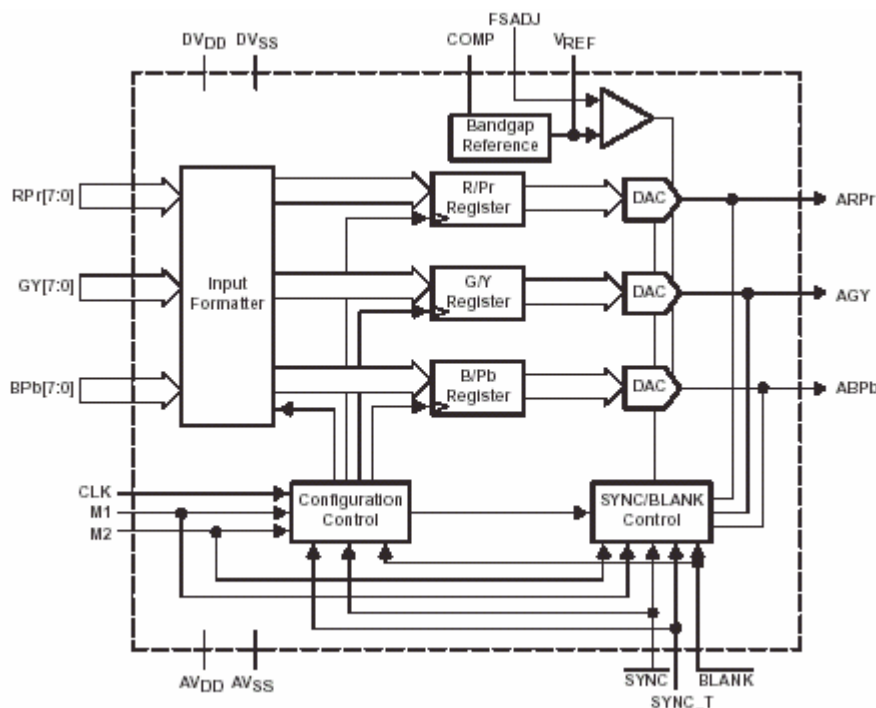
1.3 L'interface Lancelot



Lancelot (<http://www.fpga.nl>) est une plaque conçue comme extension possible à la carte de développement EXCALIBUR (System on programmable chip) de la société Altera. Elle comporte deux connecteurs PS2 (clavier + souris), un circuit d'alimentation 5 volt et pour la partie VGA un triple convertisseur 8 bits vidéo de type THS8134.

Chaque convertisseur reçoit les 8 bits de la couleur qu'il doit sortir sous forme analogique. Des signaux particuliers permettent de contrôler l'ensemble de l'opération effectuée par ce circuit. Ce sont :

- Les 2 entrées de mode m1 et m2 qui seront laissées à 0 (mode 4 :4 :4 RGB les entrées seront capturées sur front montant de video_clk)
- Les signaux sync_n et sync_t servent à créer des vidéos composites. Ils seront neutralisés en prenant sync_n = 1 et sync_t = 0
- Le signal blank (actif à 0) sert à inhiber les trois sorties analogiques et doit être affirmé quand il n'y a pas de pixel à afficher.



La carte Lancelot est utilisée ici en association avec la carte Spartan-IIe de Memec.

2 Génération du signal vidéo

2.1 Entité interface_vga

On doit interfacer l'ensemble convertisseur vidéo + connecteur VGA de la plaque Lancelot et le circuit Spartan de la carte Memec. Ceci conduit à définir l'entité suivante (fichier `~nouel/pub/projet2003/interface_vga.vhd`) :

```
ENTITY interface_vga IS
PORT (
  init      : IN  std_ulogic;           -- reset de l'interface
  clk_24Mhz : IN  std_ulogic;         -- horloge pixel
  rgb       : IN  std_logic_vector(23 DOWNTO 0); -- 3 couleurs
  red       : OUT std_logic_vector(7 DOWNTO 0); -- entree 1er dac
  green     : OUT std_logic_vector(7 DOWNTO 0); -- entree 2eme dac
```

```

    blue      : OUT std_logic_vector(7 DOWNTO 0); -- entree 3eme dac

    ligne_pix : OUT std_logic_vector(9 DOWNTO 0); -- compteur lignes pixels
    colonne_pix : OUT std_logic_vector(9 DOWNTO 0); -- compteur colonnes
pixels
    m1, m2    : OUT std_ulogic;           -- mode select dac
    blank_n   : OUT std_ulogic;           -- commande DAC
    sync_n    : OUT std_ulogic;           -- commande DAC
    sync_t    : OUT std_ulogic;           -- commande DAC
    video_clk : OUT std_ulogic;           -- commande DAC
    vga_vs    : OUT std_ulogic;           -- sur connecteur VGA
    vga_hs    : OUT std_ulogic;           -- sur connecteur VGA
);
END interface_vga;

```

2.2 Structure associée

Coté circuit, la première chose à implanter est la fonction de génération des signaux nécessaires au fonctionnement du VGA. La conception de l'interface ne pose pas de problèmes particuliers. La génération des signaux de synchronisation et de contrôle avec leur timing n'est qu'un problème de comptage. On dispose d'une horloge de 24 Mhz sur la plaque Memec, les calculs de temporisation effectués à 25MHz ne seront cependant pas modifiés. Le circuit d'interface ne fait aucun traitement sur les couleurs et se contente de les transmettre au convertisseur DAC. Celui-ci demande aussi une horloge de conversion qui sera l'horloge 24MHz.

Pour les applications futures, il est utile de disposer du comptage ligne (0..479) et colonne (0..679) des pixels correspondant au signal blank infirmé.

2.3 Simulation VHDL

Le fichier `~nouel/pub/projet2003/moniteur_vga.vhd` permet de simuler le moniteur VGA en amont du FPGA.

```

LIBRARY std;
USE std.textio.ALL;
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;

ENTITY moniteur_vga IS
    GENERIC (nom_fichier : string := "vga.ppm";
             en_tete     : IN  string := "P3";
             largeur     : IN  natural := 640;
             hauteur     : IN  natural := 480 ;
             couleur_max : IN  natural := 255);
    PORT (
        video_clk : IN std_logic;           -- horloge pour regler le debit de
        sortie
        rouge     : IN  std_logic_vector(7 DOWNTO 0 ) ; -- pixels
        vert      : IN  std_logic_vector(7 DOWNTO 0 ) ; -- pixels
        bleu      : IN  std_logic_vector(7 DOWNTO 0 ) ; -- pixels
        blank_n   : IN  std_logic);         -- validation pixels
END moniteur_vga;

```

Une image au format PPM est générée à partir des 640x480 premiers pixels validés par *blank_n* au rythme de l'horloge *video_clk*. Cette image pourra après simulation être visualisée avec un outil adéquat tel que XV (`~goudal/bin/xv`).

2.4 1^{er} test : Affichage d'une palette de couleurs

Limiter le nombre de couleurs à 256 exige de définir une palette qui à 8 bits d'entrée fera correspondre les 24 bits rgb entrant dans la composition de cette couleur. La palette sera réalisé techniquement par un circuit de type ROM (fichier *~nouel/pub/projet2003/palette.vhd*).

```
ENTITY palette IS
PORT (
    couleur      : IN  std_logic_vector(7 DOWNT0 0); -- entrée 8 bits
    rgb          : OUT std_logic_vector(24 DOWNT0 0));-- vers interface
END;
```

Dans le cas d'une échelle de 256 niveaux de gris, la palette n'est plus nécessaire car il suffit de reporter l'entrée 8 bits sur les 3 composantes : rouge=bleu=vert=entrée.

Un test simple de l'interface VGA en vraie grandeur consiste à afficher sur 1 ligne chaque couleur de la palette. Pour cela il suffit de reboucler le signal *ligne_pix* sur l'entrée couleur de la palette. On devrait ainsi voir toute la palette (256 lignes) puis de nouveau les 224 (=480-256) lignes de la même palette.

3 Applications

3.1 Contraintes

Ce qui va limiter nos applications VGA ce sont les limites du circuit Spartan-IIIE de la carte Memec. En effet, pour une application graphique VGA pleine capacité, il faudrait 480x640x3 octets de mémoire par image soient 921600 octets ou 900K octets pour une image en 17 millions de couleurs. Or le circuit Spartan ne dispose que d'un maximum de 64 kilo-bits (8 kilo-octets) et il n'y a pas de mémoire externe sur la carte Memec. En conséquence, il est absolument nécessaire de limiter les performances escomptées et donc les ressources. Le passage en 256 couleurs divise par trois la mémorisation d'une image mais 300K-octets est encore beaucoup trop important.

Travailler en 2 couleurs (couleur sur fond blanc ou noir et blanc etc..) réduit à un seul bit l'information pixel, on arrive alors à 307200 bits soient 37 Kilo-octets environ ce qui est toujours insuffisant.

Les applications les plus économes sont celles qui n'obligent pas à mémoriser toute l'image mais seulement certains objets sur fond constant.

3.2 Affichage de caractères

Un caractère sur l'écran peut être localisé sur une matrice 8 par 8 de pixels et sa représentation mémorisée dans une table de 8 octets comme dans l'exemple suivant du caractère A:

Adresse	Donnée (font)
000	00011000
001	00111100
010	01100110
011	01111110
100	01100110
101	01100110
110	01100110
111	00000000

Chaque bit de donnée représente un pixel d'une même ligne. Un changement d'adresse correspond à la ligne suivante de balayage.

Sur ce principe, il ne faut que 512 octets pour avoir en mémoire 64 modèles de caractères. Ils pourront être implantés facilement dans un circuit de type ROM (fichier

~nouel/pub/projet2003/fonts.vhd)

```
ENTITY fonts IS
PORT (
  adresse_caractere : IN std_logic_vector(5 DOWNTO 0); -- 64 caracteres
  colonne_font      : IN std_logic_vector(2 DOWNTO 0); -- 1 bit parmi 8
  ligne_font       : IN std_logic_vector(2 DOWNTO 0); -- 1 bit parmi 8
  pixel            : OUT std_ulogic) ; -- allumé ou éteint
END ;
```

La table de fonts est conforme au tableau suivant :

Caractère	Adresse	Caractère	Adresse	Caractère	Adresse	Caractère	Adresse
@	0	P	16	Espace	32	0	48
A	1	Q	17	!	33	1	49
B	2	R	18	"	34	2	50
C	3	S	19	#	35	3	51
D	4	T	20	\$	36	4	52
E	5	U	21	%	37	5	53
F	6	V	22	&	38	6	54
G	7	W	23	'	39	7	55
H	8	X	24	(40	8	56
I	9	Y	25)	41	9	57
J	10	Z	26	*	42	A	58
K	11	[27	+	43	B	59
L	12	Flèche bas	28	,	44	C	60
M	13]	29	-	45	D	61
N	14	Flèche haut	30	.	46	E	62
O	15	Flèche gauche	31	/	47	F	63

3.3 2^{ème} test : Affichage des 8 premiers caractères

Cela permettra de tester la qualité des fonts, la maîtrise de leur taille, de leur couleur, de la couleur de fond.

3.4 3^{ème} test : Affichage d'un texte sur l'écran

Afficher un texte à l'écran implique une mémorisation du message à afficher. Celui-ci sera constitué d'un tableau d'adresses caractères. Il faudra aussi décider a priori de l'emplacement de la ou des messages à afficher, éventuellement de la taille des caractères(on peut facilement multiplier par 2).

Le balayage vga doit ensuite prendre en compte horizontalement (compteur colonne) et verticalement (compteur ligne) chaque pixel utile extrait du message relié au composant fonts

3.5 4^{ème} test : *Sujet libre*

Quelques autres idées (non limitatives) :

- ❑ Améliorer l'affichage précédent ; choix de la couleur d'affichage par switch , taille variable par interrupteur, affichage du nom de la couleur sur les afficheurs 7 segments.
- ❑ animer le message sur l'écran
- ❑ créer et utiliser d'autres fonts.
- ❑

4 Bibliographie

- ❑ MicroPC et Image VGA : Christophe Paris – PFE ENSEIRB 2002
- ❑ Rapid prototyping of digital systems: James O. Hamblen , Michael D. Fuman – Kluwer academic publishers
- ❑ Site <http://www.fpga.nl> pour Lancelot
- ❑ ~nouel/pub/doc/circuits pour le convertisseur dac vidéo
- ❑ ~nouel/pub/doc/xilinx pour la carte Memec et le circuit Spartan