XFree86 Video Timings HOWTO

Eric Steven Raymond

Thyrsus Enterprises

<esr@thyrsus.com>

Traduction française : Guillaume Allègre

Version 5.0

$Date: 2001/08/11 00:37:48 $

Historique des versions		
Version 6.0	2001-08-09	esr
Explication plus précise de DDC et EDID. Cet Howto est maintenant intrinsèquement obsolète.		
Version 5.0	2000-08-22	esr
Première version DocBook.		

Résumé

Cet Howto est maintenant obsolète. Les versions courantes de XFree86 (4.0.1 et supérieures) calculent automatiquement les lignes de mode optimales pour les résolutions spécifiées dans le fichier de configuration de X, section Modes.

Comment définir une ligne de mode pour votre couple carte graphique/moniteur sous . La distribution XFree86 inclut maintenant les éléments requis pour configurer la plupart des combinaisons standard ; le but premier de ce document est de vous apprendre à mettre au point une ligne de mode personnalisée pour un moniteur à haute performance ou du matériel très inhabituel. Il vous aidera aussi à utiliser kvideogen pour créer des lignes de mode (*modelines*), ou xvidtune pour effectuer le réglage fin d'un mode standard qui n'est pas encore optimal pour votre moniteur.

Table des matières

1. Mise en garde

L'utilisation des informations reprises ci-dessous se fait *EXCLUSIVEMENT A VOS RISQUES ET PERILS*. Cadencer votre moniteur en dehors des limites fixées par les spécifications du fabricant peut présenter un danger pour votre matériel et pour vous-mêmes. Lisez la section Usage du moniteur en surcapacité [p 21] pour un avertissement détaillé. Tout dommage infligé à votre personne ou à votre moniteur dû à un usage en surcapacité relève de votre seule responsabilité.

La plus récente version de ce HOWTO est disponible à la page Web du Linux Documentation Project.

Prière de transmettre vos commentaires, critiques, et suggestions à <`esr@snark.thyrsus.com`> (en anglais). S'il vous plaît, *n'envoyez pas*de courrier électronique implorant une solution miracle à vos problèmes personnels de moniteur, dans la mesure où agir de la sorte ne servira qu'à perdre mon temps et à vous frustrer -- tout ce que je sais sur le sujet est contenu ici.

2. Pourquoi cet Howto est obsolète

Avec les versions 4.0.0 et supérieures de XFree86, vous n'avez plus du tout besoin de générer les lignes de mode. Elles sont calculées automatiquement par le serveur au démarrage, d'après la résolution spécifiée dans le fichier de configuration de XFree86, section Screen, sous-section Modes, et les capacités de votre moniteur que le serveur X récupère via une requête EDID au moniteur.

Pour changer la résolution écran et la profondeur des couleurs, il suffit d'éditer ou de créer une section Display les décrivant. Voici un exemple de description Screen de mon fichier de configuration pour X :

```
Section "Screen"
        Identifier   "Screen0"
        Device       "ATI Rage Mobility"
        Monitor      "Monitor0"
        DefaultDepth    16

        Subsection "Display"
                Depth       16
                Modes       "1024x768"
        EndSubsection

EndSection
```

Normalement, tout ce que vous devrez faire sera de changer les valeurs dans l'entrée Modes ; X fera le reste. Si vous demandez une résolution impossible, il calculera la plus proche résolution supportable par votre moniteur, d'après les donnée de la requête EDID.

De ce fait, les informations contenues dans le reste de cet Howto ne vous seront utiles que si (a) vous avez un vieux moniteur, antérieur à la norme EDID, ou si (b) votre pilote de carte graphique ne permet pas d'interroger le moniteur, ou si (c) vous utilisez une ancienne version de XFree86. Dans ce dernier cas, vous devriez le mettre à jour pour régler tous vos problèmes.

3. Introduction

Le serveur XFree86 permet aux utilisateurs de configurer leur sous-système vidéo, ce qui favorise une utilisation optimale du matériel existant. Le but de ce guide est de vous apprendre à créer vos propre valeurs d'horloge pour faire le meilleur usage de votre carte vidéo et moniteur.

Nous présenterons une méthode pour obtenir une configuration de base utilisable, ensuite nous vous montrerons comment, au départ de celle-ci, vous pouvez vous livrer à des expériences pour déterminer un ensemble de valeurs qui l'adapte à vos préférences.

Si vous disposez déjà d'un mode qui fonctionne presque (en particulier, si l'un des modes VESA pré-définis vous donne une image stable mais décentrée vers la gauche ou la droite, trop petite, ou trop grande) vous pouvez passer immédiatement à la section intitulée Résoudre les problèmes affectant l'image [p ??] . Celle-ci vous apprendra diverses manières de manipuler les valeurs d'horloge en vue d'obtenir certains résultats précis.

Ne partez pas du principe que vous allez avoir de fastidieux ajustements de modes à faire, juste parce qu'à votre premier lancement de X après installation, vous obtenez un écran brouillé. Il se peut que presque toutes les lignes de modes soient bonnes, mais que le serveur utilise par défaut justement celle qui ne convient pas à votre matériel. Commencez plutôt par parcourir le cycle des modes installés avec CTRL-ALT-KP+. Si certains donnent de bons résultats, essayez de les commenter tous, sauf un mode 640x480, et vérifiez que ce mode fonctionne. Si c'est le cas, décommentez aussi une paire d'autres modes, par exemple un 800x600 et un 1024x768 à une fréquence que votre moniteur devrait aussi gérer.

4. Outils de calcul automatique

Si votre moniteur est postérieur à 1996, il supporte probablement la norme EDID. Les moniteurs compatibles EDID (parfois appelés "Plug'n'Play" dans la littérature marketing de Microsoft) peuvent déclarer leurs capacités à l'ordinateur.

De nombreux modules pilotes de XFree86 4.0 supportent le système DDC, VESA Display Data Channel facility (canal de données d'affichage VESA). Un module de pilote de carte compatible DDC est capable d'envoyer au moniteur une requête EDID, et de se configurer automatiquement d'après la réponse. Ainsi, avec XFree 4.0 et un moniteur récent, vous avez de bonnes chances de n'avoir aucune configuration à faire.

Si votre module de pilote de carte n'est pas compatible DDC, mais que votre moniteur est compatible EDID, vous pouvez tout de même utiliser le programme read-edid, qui interrogera le moniteur sur ses capacités, et calculera pour vous une ligne de mode. Voir http://altern.org/vii/programs/linux/read-edid/.

Depuis la version 3.2, XFree86 s'accompagne du programme **XF86Setup** qui simplifie grandement la création interactive d'un mode graphique valable, sans devoir manipuler directement les valeurs d'horloge vidéo. Ainsi, dans la plupart des cas, il ne devrait pas vous être nécessaire de calculer un mode graphique de base. Malheureusement, **XF86Setup** a ses limites ; il ne connaît que les modes graphiques standards jusqu'à 1280x1024. Si vous disposez d'un moniteur à très haute performance capable d'afficher 1600x1200 ou plus, il vous faudra malgré tout encore calculer votre mode graphique de base vous-mêmes.

Il y a un utilitaire KDE appelé KVideoGen qui calcule des lignes de mode à partir des caractéristiques du moniteur et de la carte. J'ai bien généré des lignes de modes avec, mais je ne les ai pas testées. Notez que ses paramètres "refresh rate" horizontal et vertical sont la même chose que les fréquences de synchronisation HSF et VSF que nous décrivons plus loin. Le nombre "horizontal sync pulse" semble être la largeur de l'impulsion de synchronisation en microsecondes, HSP (l'outil supposant des valeurs corrigées de HGT1 "front porch" et HGT2 "back porch"). Si vous ne connaissez pas votre "horizontal sync pulse", il vaut mieux laisser la valeur par défaut.

Un autre générateur de lignes de mode pour XFree86 se trouve ici. Vous pouvez soit récupérer le script Python script, soit utiliser le formulaire CGI.

Les versions récentes de XFree86 fournissent un outil appelé **xvidtune** que vous trouverez sans doute très utile pour tester et affiner les modes graphiques. Il commence par un avertissement effrayant relatif aux possibles conséquences d'un usage abusif. Si vous accordez à ce document une attention scrupuleuse et apprenez ce qui se cache derrière les jolies valeurs dans les écrans de xvidtune, vous serez capables d'utiliser ce programme efficacement et en toute confiance.

Si **xvidtune**(1) est présent, il vous sera possible d'essayer de nouveaux modes "au vol", sans modifier votre fichier de configuration X, sans même redémarrer votre serveur X. Dans le cas contraire, XFree86 vous permet d'utiliser des raccourcis clavier pour sélectionner parmi les différents modes définis dans Xconfig (voyez XFree86.man pour de plus amples détails). Exploitez cette capacité pour vous éviter des ennuis ! Lorsque vous souhaitez tester un nouveau mode, donnez-lui un nom unique et ajoutez-le à la *fin* de votre liste de raccourcis. Gardez toujours un mode que vous savez bon comme défaut, de façon à avoir une position de repli si le mode en cours de test ne marche pas.

A la fin de ce document, vous trouverez un script modeplot que vous pourrez utiliser pour produire un graphe analogique des modes disponibles. Ce n'est pas directement utile pour générer des lignes de mode, mais ça peut vous aider à comprendre les relations qui permettent de les définir.

5. Comment fonctionnent les écrans vidéo

Savoir comment fonctionne l'écran est essentiel pour comprendre quelles valeurs placer dans les différents champs du fichier Xconfig. Le serveur XFree86 utilise ces valeurs pour obtenir le contrôle de plus bas niveau sur l'écran.

L'écran crée une image à partir de ce qu'on peut considérer comme une série de points. Ces points sont juxtaposés de gauche à droite pour créer des lignes. Ces lignes sont à leur tour juxtaposées de haut en bas pour créer l'image. Les points émettent de la lumière lorsqu'ils sont frappés par les faisceaux d'électrons à l'intérieur du tube cathodique, un faisceau par couleur primaire, Rouge, Vert, Bleu. Pour faire en sorte que chaque faisceau frappe tous les points pendant une durée égale, le faisceau balaye l'écran suivant un itinéraire immuable, appelée trame.

Nous avons écrit "ce qu'on peut considérer comme une série de points" car les points de la trame ne correspondent pas aux points de phosphore physiques. Les points de trame sont beaucoup plus gros, et regroupent beaucoup de points de phosphore. Ils doivent l'être, car autrement l'affichage souffrirait d'un sévère effet de moiré. Les points de la trame correspondent réellement à l'échantillonnage du signal analogique du pilote vidéo, et sont affichés sous la forme d'une grille de points simplement parce que les pics et les vallées du signal sont espacés suffisamment régulièrement et finement.

Le tracé de cet itinéraire commence dans le coin supérieur gauche, traverse l'écran vers la droite en une ligne presque horizontale, imperceptiblement descendante. Le faisceau est alors envoyé du côté gauche de l'écran, mais une ligne plus bas. Cette nouvelle ligne est parcourue de gauche à droite tout comme la première. Ce schéma est répété jusqu'à ce que la dernière ligne de l'écran ait été parcourue. A ce moment, le faisceau est renvoyé du coin inférieur droit au coin supérieur gauche, et la manoeuvre recommence.

Il existe une variante de ce schéma, appelée mode entrelacé (interlacing) : dans ce cas, seule une ligne sur deux est parcourue pendant la première demi-trame et les autres sont traitées lors d'un deuxième parcours de demi-trame.

Le départ du faisceau dans le coin supérieur gauche de l'écran est appelé le début de trame. La trame se termine lorsque le faisceau retrouve sa position de départ, revenant du coin inférieur droit. Une trame se compose de toutes les lignes que le faisceau a parcourues entre le haut et le bas de l'écran.

Si les faisceaux d'électrons étaient allumés en permanence pendant leur parcours de la trame, tous les points de l'écran seraient illuminés. Il n'y aurait pas de marges noires autour de la zone affichable. Aux bords de l'écran, l'image serait distordue car il est difficile de contrôler les faisceaux à cet endroit. Afin de réduire cette distorsion, les points en dehors de la zone affichable ne sont pas illuminés par les faisceaux, alors que les faisceaux traversent bien ces zones, mais éteints. C'est ainsi que la taille de la zone affichable est inférieure à la surface totale de l'écran.

Un autre concept important à comprendre est ce qu'il advient des faisceaux lorsqu'aucun point n'est illuminé à ce moment dans la zone affichable. Le temps pendant lequel le faisceau aurait pu illuminer les marges latérales de la zone affichable est utilisé pour renvoyer le faisceau du côté droit au côté gauche. De la même façon, le temps pendant lequel le faisceau aurait pu illuminer les marges inférieure et supérieure de la zone affichable est utilisé pour déplacer le faisceau du coin inférieur droit de l'écran au coin supérieur gauche.

Le rôle de la carte graphique est de générer les signaux qui commanderont à l'écran d'allumer ou d'éteindre les faisceaux d'électrons pour chaque point, selon la couleur désirée, créant ainsi l'image. La carte contrôle aussi le moment où l'écran déplace les faisceaux du côté droit au début de la ligne suivante en émettant ce que l'on appelle le signal de synchronisation horizontale (horizontal sync pulse). Un signal de synchronisation horizontale est émis à la fin de de chaque ligne. La carte graphique émet aussi un signal de synchronisation verticale (vertical sync pulse) qui commande à l'écran de renvoyer le faisceau dans le coin supérieur gauche. Un signal de synchronisation verticale est émis à la fin de chaque trame.

De courts temps de pause sont nécessaires immédiatement avant et après l'émission des signaux de synchronisation horizontale et verticale de façon à ce que les positions des faisceaux puissent se stabiliser. Sans cela, l'image ne sera pas stable.

Pour plus d'informations, il y a une page (en anglais) TV and Monitor Deflection Systems.

Dans une section ultérieure, nous reviendrons sur ces bases avec des définitions, des formules et des exemples pour vous aider à les utiliser.

6. Principes fondamentaux relatifs à votre écran et votrecarte graphique

Il y a quelques principes fondamentaux qu'il vous faut comprendre avant de bricoler une entrée dans le fichier XF86config. Ceux-ci sont :

- les fréquences de synchronisation horizontale et verticale de votre moniteur
- la bande passante de votre moniteur
- les fréquences d'horloge pilote de votre carte graphique, ou "dot clocks"

6.1. Les fréquences de synchronisation du moniteur

La fréquence de synchronisation horizontale représente simplement le nombre de fois par seconde que le moniteur peut parcourir une ligne horizontale ; c'est, de toutes, la valeur la plus importante concernant votre moniteur. La fréquence de synchronisation verticale représente le nombre de fois par seconde que le faisceau peut entièrement traverser l'écran verticalement.

Les fréquences de synchronisation seront généralement fournies à la page "spécifications techniques" du manuel de votre moniteur. La fréquence de synchronisation verticale est typiquement exprimée en Hz (cycles par seconde), la valeur horizontale en kHz (kilo-cycles par seconde). Les plages de valeurs se situent habituellement entre 50 et 150 Hz verticalement, et entre 31 et 135 kHz horizontalement.

Si vous avez un moniteur dit multi-fréquences (multisync), ces fréquences seront données sous forme de plages. Certains moniteurs, spécialement les modèles bas de gamme, ne supportent qu'une série de fréquences fixes. Ceux-ci peuvent aussi être configurés, mais votre marge de manoeuvre sera sérieusement limitée par les caractéristiques physiques du moniteur. Choisissez la plus haute paire de fréquences pour obtenir la meilleure résolution. Et soyez prudents : essayer de piloter un moniteur à fréquences fixes à une fréquence supérieure à celle pour laquelle il a été conçu peut aisément l'endommager.

Des versions précédentes de ce guide traitaient de façon fort légère l'utilisation de moniteurs multi-fréquences en surcapacité, les poussant au-delà du maximum nominal de leur fréquence de synchronisation verticale dans le but d'obtenir de meilleures performances. D'autres arguments ont depuis lors été portés à notre connaissance, incitant à la prudence en ce domaine ; nous reviendrons sur ce sujet dans la section Usage du moniteur en surcapacité [p 21] ci-dessous.

6.2. La bande passante vidéo du moniteur

La page des spécifications techniques du votre manuel de votre moniteur devrait mentionner sa bande passante. Si ce n'est pas le cas, jetez un oeil à la résolution maximale annoncée pour le moniteur. En première approximation, voici une table de conversion de résolution en estimation de bande passante (il s'agit donc de limites supérieures approximatives pour la fréquence pilote (dot clock) que vous pouvez utiliser) :

```
640x480                 25
800x600                 36
1024x768                65
1024x768 interlaced     45
1280x1024               110
1600x1200               185
```

Au demeurant, cette table n'a rien de magique ; ces valeurs sont simplement les fréquences pilotes les plus basses par résolution dans la base de données des modes standard XFree86 (exception faite de la dernière, que j'ai extrapolée). La bande passante de votre moniteur peut en réalité être plus élevée que le minimum requis pour sa résolution maximale, aussi ne craignez pas d'essayer une fréquence pilote de quelques MHz supérieure.

Notez aussi que la bande passante doit rarement être prise en ligne de compte aux fréquences pilotes inférieures à 65 MHz environ. Avec une carte SVGA et la plupart des moniteurs haute résolution, vous ne pourrez même pas approcher la limite de la bande passante de votre moniteur. En voici quelques exemples :

```
Brand                           Video Bandwidth
----------                      ---------------
NEC 4D                          75Mhz
Nano 907a                       50Mhz
Nano 9080i                      60Mhz
Mitsubishi HL6615               110Mhz
Mitsubishi Diamond Scan         100Mhz
IDEK MF-5117                    65Mhz
IOCOMM Thinksync-17 CM-7126     136Mhz
HP D1188A                       100Mhz
Philips SC-17AS                 110Mhz
Swan SW617                      85Mhz
Viewsonic 21PS                  185Mhz
PanaSync/Pro P21                220Mhz
```

Même les moniteurs bas de gamme n'ont en général pas de contrainte terrible au niveau de leur bande passante aux résolutions annoncées. Le NEC Multisync II en est un exemple parfait --- si l'on en croit les spécifications du constructeur, il ne parvient même pas à afficher en 800x600. Il affichera au maximum en 800x560. Pour des résolutions si basses, il n'est pas nécessaire de disposer de hautes fréquences pilotes ou une large bande passante ; le mieux que vous puissiez faire sera probablement 32 MHz ou 36 MHz, l'un comme l'autre ne s'écartant pas trop de la bande passante annoncée du moniteur : 30 MHz.

A ces deux fréquences pilotes, il se peut que l'image affichée ne soit pas aussi nette qu'elle devrait l'être, mais certainement d'une qualité tolérable. Il serait plus agréable, bien sûr, que le NEC Multisync II dispose d'une bande passante vidéo supérieure à 36 Mhz, par exemple. Mais ceci n'est pas indispensable pour des tâches de base comme l'édition de texte, pour autant que l'écart ne soit pas à ce point important qu'il occasionne une forte distorsion de l'image (vos yeux vous le feraient sentir immédiatement si cela devait être le cas).

6.3. La fréquence pilote (dot clock) de la carte graphique

La page "spécifications techniques" du manuel de votre carte graphique mentionne sans doute le dot clock maximum de la carte (c'est à dire, le nombre total de pixels que la carte peut envoyer à l'écran par seconde).

Si vous ne possédez pas cette information, le serveur X la trouvera pour vous. Les versions récentes des serveurs X supportent tous une option --probeonly qui imprime cette information et termine sans réellement démarrer X ni changer le mode vidéo.

Si cette option n'est pas supportée dans votre cas, ne perdez pas espoir. Même si X bloque votre moniteur, il émettra une ligne de valeurs d'horloge et d'autres informations vers la sortie erreur standard (stderr). Si vous re-dirigez cela vers un fichier, les informations devraient y être conservées, même s'il vous faut redémarrer la machine pour obtenir à nouveau l'accès à la console.

Les résultats de cette détection du matériel ou les messages de démarrage devraient ressembler à l'un des exemples suivants :

Si vous utilisez XFree86 :

```
  Xconfig: /usr/X11R6/lib/X11/Xconfig
  (**) stands for supplied, (--) stands for probed/default values
  (**) Mouse: type: MouseMan, device: /dev/ttyS1, baudrate: 9600
  Warning: The directory "/usr/andrew/X11fonts" does not exist.
           Entry deleted from font path.
  (**) FontPath set to "/usr/lib/X11/fonts/misc/,/usr/lib/X11/fonts/75dpi/"
  (--) S3: card type: 386/486 localbus
  (--) S3: chipset:   924
                      ---
      Chipset -- le modèle précis du processeur (ici, un ancien masque du 86C91
1)

  (--) S3: chipset driver: s3_generic
  (--) S3: videoram:  1024k
                      -----
           Taille de la mémoire RAM tampon de trame embarquée

  (**) S3: clocks:  25.00  28.00  40.00   3.00  50.00  77.00  36.00  45.00
  (**) S3: clocks:   0.00   0.00  79.00  31.00  94.00  65.00  75.00  71.00
                    ------------------------------------------------------
                              Fréquences pilotes autorisées en MHz

  (--) S3: Maximum allowed dot-clock: 110MHz
                                      ------
                                 Largeur de bande
  (**) S3: Mode "1024x768": mode clock =  79.000, clock used =  79.000
  (--) S3: Virtual resolution set to 1024x768
```

```
  (--) S3: Using a banksize of 64k, line width of 1024
  (--) S3: Pixmap cache:
  (--) S3: Using 2 128-pixel 4 64-pixel and 8 32-pixel slots
  (--) S3: Using 8 pages of 768x255 for font caching
```

Si vous utilisez SGCS ou X/Inside X :

```
WGA: 86C911 (mem: 1024k clocks: 25 28 40 3 50 77 36 45 0 0 79 31 94 65 75 71)
---  ------       -----         --------------------------------------------
   |     |            |              Fréquences pilotes autorisées en MHz
   |     |            +-- Taille de la mémoire RAM tampon de trame embarquée
   |     +-- Modèle du processeur
   +-- Type du serveur
```

Note : effectuez ce test sur votre machine lorsqu'elle n'est pas chargée (si possible). Dans la mesure où X est une application, ses boucles de temporisation peuvent être perturbées par l'activité sur le disque, rendant les valeur mentionnées ci-dessus imprécises. Effectuez le test plusieurs fois et veillez à ce que ces valeurs se stabilisent ; si cela ne se produit pas, éliminez autant de processus actifs que nécessaire. En particulier, le processus démon qui pilote votre souris risque fortement de vous causer des problèmes (il s'agit de gpm pour les utilisateurs Linux, de mousemgr pour les utilisateurs de SVr4).

De façon à éviter toute imprécision lors de la détection des fréquences d'horloge, vous pouvez simplement copier la liste de fréquences et la placer dans votre fichier XF86config comme valeur de la propriété "Clocks" --- ceci supprime la boucle de temporisation et fournit à X une liste précise des valeurs d'horloge qu'il peut utiliser. En utilisant les données de l'exemple ci-dessus, cela donnerait :

```
wga
        Clocks  25 28 40 3 50 77 36 45 0 0 79 31 94 65 75 71
```

Sur des systèmes dont la charge de travail varie fortement, ceci peut vous aider à éviter de mystérieux échecs au démarrage de X. Il peut arriver que X démarre, obtienne une mauvaise temporisation à cause de la charge excessive du système, et ne soit dès lors pas capable de trouver une fréquence correspondante dans sa base de données de configuration --- ou choisisse la mauvaise !

6.4. Ce que contrôlent ces données de base

Les plages de fréquences de synchronisation de votre moniteur, ainsi que la fréquence pilote de votre carte graphique, déterminent la résolution maximale qu'il vous sera loisible d'atteindre. Mais c'est le rôle du pilote de périphérique d'exploiter le potentiel de votre équipement. Du matériel haut de gamme sans pilote spécialisé est un gaspillage d'argent. D'un autre côté, avec un pilote de périphérique flexible et du matériel moins puissant, vous pouvez gagner un peu de marge de manoeuvre par rapport aux contraintes physiques. Telle est la philosophie qui présida à la conception de XFree86.

Il vous faudra sélectionner une fréquence d'horloge adaptée à la bande passante vidéo de votre moniteur. Vous bénéficierez ici d'une large marge de manoeuvre, cependant --- certains moniteurs ont une capacité réelle supérieure de 30% par rapport à leur bande passante nominale. Le risque ici est de dépasser la fréquence de synchronisation verticale annoncée du moniteur ; nous discuterons ceci en détail plus loin.

Cette connaissance de la bande passante vous permettra d'effectuer des choix plus avertis entre diverses configurations possibles. Elle peut en effet influencer la qualité visuelle de votre écran (spécialement la précision dans les petits détails).

7. Comprendre les spécifications de base

Cette section explique la signification des spécifications mentionnées précédemment, ainsi que certains autres éléments qu'il vous sera utile de connaître. Tout d'abord, quelques définitions. A côté de chaque terme défini est mentionné entre parenthèses le nom de la variable que nous utiliserons pour le représenter dans nos formules.

fréquence de synchronisation horizontale (HSF - horizontal sync freq.)

> Nombre de parcours horizontaux par seconde (voir ci-dessus).

fréquence de synchronisation verticale (VSF - vertical sync freq.)

> Nombre de parcours verticaux par seconde (voir ci-dessus). Principalement important comme limite supérieure de la fréquence de rafraîchissement.

fréquence pilote (DCF - driving clock freq. = dot clock)

> La fréquence du cristal ou VCO de votre carte graphique --- le nombre maximum de points par seconde qu'elle peut émettre.

bande passante vidéo (VB - video bandwith)

> La fréquence la plus élevée que vous puissiez appliquer à l'entrée vidéo de votre moniteur en conservant une chance raisonnable d'obtenir une image intelligible. Si vous vous représentez le signal émis par votre carte graphique comme une succession rapide d'états allumés/éteints, sa fréquence la plus basse est égale à la moitié de DCF, de sorte qu'en théorie, la bande passante n'a de sens qu'à partir de DCF/2. Pour obtenir à l'écran un affichage suffisamment net des petits détails, cependant, vous ne souhaiterez pas qu'elle soit de loin inférieure à votre DCF maximale ; il vaudrait même mieux qu'elle lui soit supérieure.

longueur de trame (HFL, VFL)

> La longueur de trame horizontale (HFL - horizontal frame length) est le nombre de tics d'horloge de votre carte graphique dont le canon à électrons de votre moniteur a besoin pour parcourir une ligne horizontale, *y compris les marges gauche et droite inactives*. La longueur de trame verticale (VFL - vertical frame length) est le nombre de lignes parcourues dans l'image *entière*, y compris les marges inférieure et supérieure inactives.

fréquence de rafraîchissement de l'écran (RR - refresh rate)

> Le nombre de fois par seconde que votre image est redessinée (ceci est aussi appelé la "fréquence de trame" - frame rate). Plus cette fréquence est élevée, meilleure est l'image, dans la mesure où cela diminue l'effet de clignotement. 60 Hz est bon, mais le standard VESA (72 Hz) est meilleur. Calculez-la sur base de la formule suivante :

```
RR = DCF / (HFL * VFL)
```

> Notez que le produit au dénominateur n'est *pas* égal à la résolution visible du moniteur, mais typiquement légèrement plus grand. Nous entrerons dans les détails de ceci plus loin.

Les fréquences pour lesquelles sont généralement mentionnés les modes dits entrelacés (comme "87Hz Interlaced") sont en fait des fréquences de demi-trame : un écran de ce type semble avoir une fréquence de rafraîchissement supérieure aux autres modèles de sa catégorie, mais chaque ligne individuelle n'est rafraîchie qu'une fois sur deux.

Dans le cadre de nos calculs, nous prendrons en compte la fréquence de rafraîchissement d'un moniteur entrelacé par trame complète, c'est-à-dire 43.5 Hz dans l'exemple cité plus haut. La qualité d'un mode entrelacé est meilleure que celle d'un mode non-entrelacé à fréquence de rafraîchissement par trame complète égale, mais définitivement plus médiocre que le mode non-entrelacé correspondant à la fréquence de rafraîchissement par demi-trame.

7.1. A propos de la bande passante

Les fabricants de moniteurs aiment vanter la large bande passante offerte par leur matériel car elle détermine la netteté des transitions d'intensité et de couleur à l'écran. Une large bande passante signifie des petits détails bien visibles.

Votre moniteur utilise des signaux électroniques pour offrir à vos yeux l'image qu'ils contemplent. Des signaux de ce type se présentent toujours sous la forme d'une onde une fois que l'information digitale a été convertie en un signal analogique. Ils peuvent être perçus comme une combinaison de nombreuses ondes plus simples, chacune ayant une fréquence fixe, pour la plupart dans la bande des MHz, par exemple, 20 MHz, 40 MHz, voire même 70 MHz. La bande passante de votre moniteur n'est autre que le signal analogique de plus haute fréquence qu'il peut gérer sans distorsion.

Dans le cas qui nous occupe, la bande passante vidéo est essentiellement importante comme limite supérieure approximative de la fréquence pilote (dot clock) qu'il vous est possible d'utiliser.

7.2. Fréquences de synchronisation et fréquence de rafraîchissement

Chaque ligne parcourue horizontalement à l'écran n'est que la partie visible d'un parcours de toute la longueur de la trame. A tout moment, il n'y a en fait qu'un seul point actif à l'écran, mais avec une fréquence de rafraîchissement suffisamment élevée, la persistance rétinienne de vos yeux vous permet de percevoir l'image complète.

Voici quelques schémas qui vous aideront :

```
 _______________________
|                       |      La fréquence de synchronisation horizontale
|->->->->->->->->->->-> |      représente le nombre de fois
|                      )|      par seconde que le faisceau d'électrons
|<-----<-----<-----<--- |      du moniteur peut parcourir
|                       |      un chemin semblable à celui-ci
|                       |
|                       |
|                       |
|_______________________|

 _______________________
|        ^              |      La fréquence de synchronisation verticale
|       ^ |             |      représente le nombre de fois par
|       | v             |      seconde que le faisceau d'électrons du
|       ^ |             |      moniteur peut parcourir un chemin
|       | |             |      semblable à celui-ci
```

```
    |       ^ |             |
    |       | v             |
    |       ^ |             |
    |_______|_v_____________|
```

Souvenez-vous que le balayage de la zone affichable (raster scan) adopte en réalité la forme d’un zigzag très serré ; cela signifie que le faisceau se déplace de gauche à droite et en même temps de haut en bas.

Nous comprenons maintenant comment fréquence pilote (dot clock) et taille de trame sont liées à la fréquence de rafraîchissement. Par définition, un hertz (Hz) équivaut à un cycle par seconde. Ainsi, si votre longueur de trame horizontale est représentée par HFL et votre longueur de trame verticale par VFL, il vous faut alors (HFL * VFL) tics pour couvrir l’entièreté de l’écran. Puisque votre carte émet DCF tics par seconde par définition, il s’ensuit bien entendu que le (les) canon(s) à électrons de votre moniteur peuvent parcourir l’écran de gauche à droite et retour et de haut en bas et retour DCF / (HFL * VFL) fois par seconde. Ceci représente la fréquence de rafraîchissement de votre écran, car c’est le nombre de fois que le contenu de votre écran peut être mis à jour (donc *rafraîchi*) par seconde !

Il vous est nécessaire d’intégrer ce concept si vous voulez pouvoir mettre sur pied une configuration qui sacrifie de la résolution pour gagner de la stabilité (réduction de l’effet de clignotement) de la manière qui vous convienne le mieux.

Pour ceux d’entre vous qui comprennent mieux un petit dessin qu’un long discours, en voici un :

```
        RR                                      VB
         |   min HSF                     max HSF |
         |    |             R1        R2  |      |
max VSF -+----|------------/----------/---|------+----- max VSF
         |    |:::::::::::/::::::::::/:::::\     |
         |    \::::::::::/::::::::::/:::::::\    |
         |     |::::::::/::::::::::/:::::::::|   |
         |     |:::::::/::::::::::/::::::::::\   |
         |     \::::::/::::::::::/::::::::::::\  |
         |      \::::/::::::::::/::::::::::::::| |
         |       |::/::::::::::/:::::::::::::::| |
         |        \/::::::::::/:::::::::::::::::\|
         |        /\:::::::::/:::::::::::::::::::|
         |       /  \:::::::/::::::::::::::::::::|\
         |      /    |:::::/:::::::::::::::::::::| |
         |     /     \::::/::::::::::::::::::::::| \
min VSF -+----/-------\--/-----------------------|--\--- min VSF
         |   /         \/                        |   \
         +--/----------/\------------------------+----\- DCF
           R1        R2  \                       |     \
                          min HSF                |    max HSF
                                                 VB
```

Ce graphique appelé diagramme de mode traduit les capacités d’un moniteur classique. Sur l’axe des x sont placées les fréquences pilotes (DCF), sur l’axe des y les fréquences de rafraîchissement (RR). La région noircie du diagramme représente les capacités du moniteur : chaque point à l’intérieur de cette région est un mode graphique possible.

Les lignes marquées ‘R1’ et ‘R2’ symbolisent une résolution fixe (telle 640x480) ; le but de leur présence est de montrer comment une résolution donnée peut être obtenue par l’utilisation de nombreuses combinaisons différentes de fréquences pilotes et de fréquences de rafraîchissement. La ligne R2 représentera donc une plus haute résolution que R1.

Les frontières haute et basse de la région noircie sont de simples lignes horizontales qui représentent les valeurs limites de la fréquence de synchronisation verticale (min VSF et max VSF). La bande passante vidéo (VB) est une limite supérieure à la fréquence pilote et apparaît donc comme une ligne verticale limitant la région noircie sur la droite.

À la section Représentation graphique des capacités du moniteur [p 26] , vous trouverez un programme qui vous aidera à établir un diagramme semblable à celui-ci (mais beaucoup plus joli, avec des graphiques X) pour votre moniteur personnel. Ce chapitre couvrira aussi ce sujet fascinant : comment déduire les limites résultant des fréquences de synchronisation horizontale minimum et maximum.

8. Concessions lors de la configuration du système

Il existe une autre façon de formuler l'équation développée plus haut

```
DCF = RR * HFL * VFL
```

C'est-à-dire que vous considérez votre fréquence pilote comme fixe. Vous pouvez ensuite convertir ces points par seconde que vous venez d'économiser en fréquence de rafraîchissement, résolution horizontale, ou résolution verticale. Si l'un de ces paramètres augmente, l'un ou les deux autres doivent diminuer.

Notez, cependant, que votre fréquence de rafraîchissement ne peut excéder la fréquence de synchronisation verticale maximum de votre moniteur. Ainsi, pour tout moniteur à une fréquence pilote donnée, il existe un produit de longueurs de trame minimum (HFL * VFL) en dessous duquel vous ne pourrez le faire descendre.

Lorsque vous choisirez vos paramètres, souvenez-vous de ceci : si vous prenez une valeur de RR trop basse, vous serez gênés par un effet de clignotement. Il n'est pas recommandé de faire descendre votre fréquence de rafraîchissement en dessous des 60 Hz. Ceci représente la fréquence d'oscillation des néons aux USA (50 Hz en Europe, puisque c'est la fréquence du courant électrique) ; si vous êtes sensibles à ceux-ci, il vous faudra de préférence conserver les 72 Hz, le standard ergonomique VESA.

L'effet de clignotement est très éprouvant pour les yeux, bien que l'oeil humain soit adaptable et que la tolérance individuelle au phénomène varie fortement. Si vous faites face à votre moniteur à un angle de 90%, que vous utilisez une couleur de fond sombre et une couleur en fort contraste pour l'avant-plan, et que vous vous contentez d'une intensité basse à moyenne, il se *peut* qu'une fréquence de rafraîchissement aussi basse que 45 Hz vous semble confortable.

Et maintenant le test qui tue : ouvrez un xterm avec un fond blanc et un avant-plan noir avec la commande **xterm -bg white -fg black** et modifiez sa taille de façon à ce qu'il recouvre l'entièreté de la zone affichable. Réglez ensuite votre moniteur aux 3/4 de son intensité maximum, et détournez votre regard du moniteur. Essayez de le regarder en biais (de façon à forcer l'utilisation des cellules rétiniennes périphériques, plus sensibles). Si vous ne percevez aucun effet de clignotement, ou si vous considérez celui-ci tolérable, cela signifie que la fréquence de rafraîchissement vous convient. Dans le cas contraire il serait préférable que vous configuriez une fréquence de rafraîchissement plus élevée, car ce clignotement à la limite du perceptible fatiguera terriblement vos yeux et causera des maux de tête, même si l'image semble parfaite en vision normale.

Dans le cas des modes entrelacés, l'importance du clignotement dépend de plus de facteurs différents tels la résolution verticale choisie et le type d'image affiché. Il ne vous reste qu'à procéder à vos propres expériences. Quoi qu'il en soit, je ne vous conseille pas de descendre beaucoup sous la limite des 85 Hz (fréquence de demi-trame).

Partons du principe que vous avez choisi une fréquence de rafraîchissement représentant un minimum acceptable. Vous aurez alors quelqu'espace de manoeuvre dans le choix de vos HFL et VFL.

9. Exigences en terme de mémoire

La mémoire tampon de trame (frame-buffer RAM) disponible peut limiter la résolution qu'il vous sera possible d'obtenir sur des écrans couleur ou à niveaux de gris. Cela ne joue sans doute pas de rôle par contre sur des écrans qui ne peuvent afficher que deux couleurs, noir et blanc sans dégradé de gris.

Pour des images en 256 couleurs, un octet de mémoire vidéo est nécessaire pour chaque point visible à afficher. Cet octet contient l'information qui définit quel mélange de rouge, vert et bleu est utilisé pour son point. Pour calculer la quantité de mémoire requise, multipliez le nombre de points visibles par ligne par le nombre de lignes visibles. Pour un écran d'une résolution de 1024x768, cela ferait 1024 x 768 = 786432, ce qui correspond au nombre de points visibles sur l'écran. Cela représente aussi, à raison d'un octet par point, le nombre d'octets de mémoire vidéo requise sur votre carte graphique.

Ainsi, vos exigences en terme de mémoire seront typiquement de (HR * VR)/1024 Ko de VRAM, arrondis à l'unité supérieure (768 Ko exactement dans l'exemple précédent). Si vous disposez de plus de mémoire qu'il n'est strictement nécessaire, il vous sera possible d'utiliser l'excédent par la création d'un écran virtuel d'une surface supérieure à celle de votre écran physique.

Cependant, si votre carte graphique n'est équipée que de 512K, il ne vous sera pas possible d'atteindre cette résolution. Même si vous possédez un bon moniteur, à défaut d'une quantité suffisante de mémoire vidéo, votre ne pourrez exploiter pleinement les capacités de votre écran. D'un autre côté, si votre carte SVGA est dotée d'un Méga de RAM, mais que votre écran ne peut afficher plus de 800x600, les hautes résolutions sont malgré tout hors de votre portée. Voyez la section Utilisation des modes entrelacés [p 21] pour une solution possible.

Ne vous inquiétez pas si vous disposez de plus de mémoire que nécessaire ; XFree86 en fera bon usage en vous permettant de faire dérouler votre zone affichable (voyez la documentation du fichier Xconfig concernant les paramètres de taille de l'écran virtuel). Souvenez-vous aussi qu'une carte équipée de 512 Ko de mémoire ne dispose pas en réalité de 512000 octets, mais bien de 512 x 1024 = 524288 octets.

Si vous utilisez X/Inside avec une carte S3, et que vous acceptez de vous contenter de 16 couleurs (4 bits par pixel), vous pouvez employer le paramètre depth=4 dans Xconfig et effectivement doubler la résolution que votre carte pourra gérer. Les cartes S3, par exemple, offrent normalement 1024x768x256. Vous pouvez les configurer en 1280x1024x16 en forçant la profondeur d'image à 4 bits.

10. Calcul de la taille de trame

Avertissement : cette méthode a été développée pour les moniteurs multi-fréquences (multisync). Elle s'appliquera sans doute aussi aux moniteurs à fréquence fixe, mais c'est sans garantie !

Commencez par diviser DCF par la plus haute valeur de HSF disponible pour obtenir une longueur de trame horizontale.

Par exemple, supposons que vous possédiez une carte graphique SVGA de type Sigma Legend dont la fréquence pilote est de 65 MHz, et que votre moniteur ait une fréquence de synchronisation horizontale de 55 kHz. La valeur (DCF / HSF) est alors de 1181 (65 MHz = 65000 kHz ; 65000/55 = 1181).

Et maintenant notre première astuce de magie noire. Il vous faut arrondir le résultat obtenu au plus proche multiple de 8. Cela s'explique par l'architecture du contrôleur VGA en usage sur les cartes SVGA et S3 ; ce dernier utilise un registre sur 8 bits, après décalage à gauche de 3 bits, pour représenter une valeur qui normalement occupe 11 bits. D'autres modèles de carte comme l'ATI 8514/A n'exigent sans doute pas cet arrondi, mais nous ne sommes pas sûrs et l'utiliser ne peut pas nuire. Aussi, arrondissez la longueur de trame horizontale utilisable vers le bas pour obtenir 1176.

Ce résultat (DCF / HSF arrondi à un multiple de 8) représente la valeur minimum de HFL utilisable. Il est possible d'obtenir de plus grandes valeurs de HFL (et donc, probablement, plus de points horizontalement à l'écran) en jouant sur la fréquence de synchronisation pour diminuer HSF, mais en le payant par un rafraîchissement plus lent et donc un effet de clignotement accentué.

En règle générale, 80% de la longueur de trame horizontale est disponible pour la résolution horizontale, autrement dit la partie visible du parcours horizontal du faisceau (ceci tient compte, globalement, des marges et du temps de retour -- c'est-à-dire le temps requis par le faisceau pour se déplacer du côté droit de l'écran au côté gauche). Dans cet exemple, cela représente 940 tics d'horloge.

Dès lors, pour donner à votre image ses proportions normales de 4:3, fixez la résolution verticale aux 3/4 de la valeur que vous venez juste de calculer pour la résolution horizontale. Toujours dans notre exemple, cela représente 705 tics d'horloge. Pour obtenir votre VFL réel, multipliez ce chiffre par 1.05, ce qui vous donne 740 tics.

Ce rapport 4:3 n'est revêtu d'aucun pouvoir magique ; rien ne vous empêche de définir des proportions qui s'écartent de la Règle d'Or si cela doit vous permettre d'exploiter au mieux votre espace affichable. Le seul avantage de ces proportions classiques est de faciliter le calcul de la hauteur et largeur de trame à partir de la taille diagonale : multipliez simplement la diagonale par 0.8 pour obtenir la largeur et par 0.6 pour obtenir la hauteur.

Ainsi, nous avons HFL=1176 et VFL=740. Si nous divisons 65 MHz par le produit de ces deux valeurs, nous obtenons une fréquence de rafraîchissement -- parfaitement saine -- de 74.6 Hz. Excellent ! Mieux que le standard VESA ! Et vous obtenez une surface affichable de 944 x 705, soit plus que les 800 x 600 que vous attendiez probablement. Pas mal du tout !

Il est même possible d'augmenter encore la fréquence de rafraîchissement, la poussant presque jusqu'à 76 Hz, en exploitant le fait que les moniteurs supportent souvent une fréquence de synchronisation horizontale supérieure de 2 kHz environ à leur maximum annoncé, et en diminuant quelque peu VFL (c'est-à-dire, en utilisant moins de 75% de 940 dans l'exemple ci-dessus). Mais avant de vous lancer dans cette tentative d'usage en surcapacité, si vous vous y décidez, *assurez-vous* que le canon à électrons de votre moniteur accepte des fréquences verticales allant jusqu'à 76 Hz. (Le très répandu NEC 4D, par exemple, ne le supporte pas. Il ne va que jusqu'à 75 Hz VSF). (Reportez-vous à la section Usage du moniteur en surcapacité [p 21] pour une discussion plus générale de ce sujet.)

Jusqu'ici, la majeure partie de ce que nous avons exposé n'est que simple arithmétique et connaissance de base des écrans d'affichage. Pas besoin de la moindre magie noire !

11. Magie noire et impulsions de synchronisation

OK, vous venez de calculer les valeurs de HFL/VFL pour la fréquence pilote que vous avez choisie, vous avez trouvé une fréquence de rafraîchissement acceptable, et vérifié que vous disposiez d'une quantité suffisante de mémoire VRAM. Passons maintenant à la vraie magie noire : il vous faut déterminer où et quand placer vos signaux de synchronisation.

Les signaux de synchronisation contrôlent en fait les fréquences de balayage horizontale et verticale du moniteur. Les valeurs HSF et VSF que vous avez extraites des spécifications techniques sont des quantités nominales, approximatives pour les fréquences de synchronisation maximum. L'impulsion de synchronisation présente dans le signal de la carte graphique fournit au moniteur sa vitesse d'exécution effective.

Vous souvenez-vous des deux schémas présentés ci-dessus ? Seule une partie du temps requis pour que le faisceau balaye une trame complète est utilisé pour construire l'image affichée (autrement dit, votre résolution).

11.1. Synchronisation horizontale

En fonction de notre définition précédente, il faut HFL tics d'horloge pour effectuer le parcours d'une ligne horizontale. Appelons HR (horizontal resolution) le nombre de tics accomplis en mode visible (c'est-à-dire la résolution horizontale de votre écran). Il s'ensuit, par définition, que HR < HFL. Pour rester pratique, supposons que les deux quantités démarrent au même instant comme illustré ci-dessous :

```
|___ __ __ __ __ __ __ __ __ __ __ __ __
|_ _ _ _ _ _ _ _ _ _ _ _                |
|_______________________|_______________|_____
0                       ^               ^     unité : tic
                        |   ^       ^   |
                        HR  |       |  HFL
                        |   |<----->|   |
                        |<->|  HSP  |<->|
                        HGT1         HGT2
```

Maintenant, nous voudrions placer un signal de synchronisation de longueur HSP comme illustré ci-dessus, c'est-à-dire entre la fin des tics d'horloge utilisés pour l'affichage de l'image et la limite des tics consacrés à la trame complète. Pourquoi cela ? Parce que si nous parvenons à ce résultat, l'image affichée ne sera pas décaclée à gauche ou à droite. Elle occupera sa place assignée sur l'écran, recouvrant le centre de la zone affichable.

Qui plus est, il est préférable de conserver environ 30 tics "de sécurité" (guard time) de part et d'autre de l'impulsion de synchronisation. Ces valeurs sont représentées par HGT1 et HGT2. Dans une configuration typique HGT1 != HGT2, mais si vous êtes en train de construire une configuration à partir de zéro, il vaudra mieux que vous commenciez vos expériences avec des valeurs égales (c'est-à-dire, avec l'impulsion de synchronisation centrée).

Le symptôme d'une impulsion de synchronisation mal placée consiste en un déplacement de l'image sur l'écran, l'une des marges élargie à l'extrême tandis que l'autre côté de l'image est réfléchi sur la paroi latérale du tube, ce qui résulte en une ligne blanche à la limite de la zone affichable et une bande d'image "fantôme" du côté intérieur de cette ligne. Une impulsion de synchronisation verticale excessivement mal placée peut aller jusqu'à provoquer un saut cyclique de l'image similaire à ce qu'on observe sur un poste de télévision dont on aurait déréglé le bouton d'ajustement vertical (c'est en fait le même phénomène qui est à l'oeuvre ici).

Si vous avez de la chance, la largeur des signaux de synchronisation de votre moniteur sera documentée à la page des spécifications techniques. Dans le cas contraire, c'est ici que l'on entre dans la vraie magie noire...

Pour cette section-ci, il vous faudra en partie procéder par essais et erreurs. Mais la plupart du temps, vous pouvez sans danger partir du principe qu'une impulsion de synchronisation a une durée d'environ 3.5 à 4.0 microsecondes.

Toujours pour rester concret, supposons qu'HSP vaille 3.8 microsecondes (ce qui, soit dit en passant, n'est pas une mauvaise valeur de départ lorsque l'on expérimente).

Dans cette hypothèse, sur la base de la fréquence pilote de 65 Mhz mentionnée plus haut, nous obtenons que HSP est égal à 247 tics d'horloge (= 65 * 10^6 * 3.8 * 10^-6) [souvenez-vous : Méga=10^6, micro=10^-6].

Certains fabricants aiment mentionner leurs paramètres de tramage horizontal sous forme de durée plutôt que de largeur exprimée en hertz. Vous pourriez chez eux rencontrer les termes suivants :

durée utile (HAT - Horizontal Active Time)

Equivalent à HR, mais en millisecondes. HAT * DCF = HR.

durée hors service (HBT - Horizontal Blanking Time

Equivalent à (HFL - HR), mais en millisecondes. HBT * DCF = (HFL - HR).

fenêtre avant (HFP - Horizontal Front Porch)

Synonyme d'HGT1.

durée de synchronisation

Synonyme d'HSP.

fenêtre arrière (HBP - Horizontal Back Porch)

Synonyme d'HGT2.

11.2. Synchronisation verticale

En nous référant à l'illustration précédente, comment placerons-nous les 247 tics d'horloge comme indiqué dans le graphe ?

Toujours sur base du même exemple, nous avons HR égal à 944 et HFL égal à 1176. La différence entre les deux est 1176 - 944 = 232 < 247 ! De toute évidence, il nous faudra ici procéder à quelques ajustements. Que peut-on faire ?

La première étape est de porter 1176 à 1184, et de réduire 944 à 936 (NdT : choix des arrondis aux multiples de 8). Maintenant la différence = 1184 - 936 = 248. Hmm, on approche.

Ensuite, au lieu d'utiliser 3.8, nous prendrons 3.5 pour calculer HSP ; ce qui nous donne 65 * 3.5 = 227. C'est déjà beaucoup mieux. Mais 248 n'est pas beaucoup plus élevé que 227. Il est normalement requis de laisser environ 30 tics d'horloge entre HR et le début de SP, et la même remarque est valable entre la fin de SP et HFL. ET ces valeurs doivent être multiples de huit ! Sommes-nous dans l'impasse ?

Non. Procédons comme suit, 936 % 8 = 0, (936 + 32) % 8 = 0 également. Mais 936 + 32 = 968, 968 + 227 = 1195, 1195 + 32 = 1227. Hmm... ceci n'a pas l'air trop mal. Mais ce n'est pas un multiple de huit, aussi arrondissons-le à 1232.

Des ennuis potentiels se profilent hélas à l'horizon car l'impulsion de synchronisation n'est plus située juste au milieu de l'espace h - H. Heureusement, grâce à l'aide de notre calculette nous trouvons que 1232 - 32 = 1200 est aussi un multiple de 8 et que (1232 - 32) - 968 = 232, ce qui correspondrait à utiliser une impulsion de synchronisation d'une durée de 3.57 microsecondes, une valeur qui est encore raisonnable.

De plus, 936 / 1232 ~ 0.76 ou 76%, ce qui ne s'éloigne pas trop des 80%, il ne devrait donc pas y avoir de problème.

Qui plus est, l'usage de la longueur de trame horizontale présente requiert de notre moniteur qu'il effectue sa synchronisation à 52.7 kHz (= 65 MHz / 1232) ce qui se situe dans les limites de ses capacités. Pas de problème.

Sur la base des règles générales mentionnées plus haut, notre nouvelle résolution verticale sera de 936 * 75% = 702. Notre nouvelle longueur de trame verticale vaudra 702 * 1.05 = 737.

Fréquence de rafraîchissement de l'écran = 65 MHz / (737 * 1232) = 71.6 Hz. Cette valeur est toujours excellente.

La représentation de l'impulsion de synchronisation verticale est très semblable :

```
|___ __ __ __ __ __ __ __ __ __ __ __ __
|_ _ _ _ _ _ _ _ _ _ _ _ _ _ _          |
|_______________________|_______________|______
0                      VR              VFL      unit: ticks
                        ^   ^       ^
                        |   |       |
                        |<->|<----->|
                         VGT    VSP
```

Nous commençons l'impulsion de synchronisation juste après la fin des cycles verticaux consacrés à l'affichage réel de données. VGT est l'intervalle de sécurité (vertical guard time) requis pour l'impulsion de synchronisation. La plupart des moniteurs supportent sans broncher une valeur de VGT de 0 (pas d'intervalle de sécurité) et nous tirerons parti de cette possibilité dans cet exemple. Un petit nombre, cependant, nécessitent l'emploi d'un intervalle de sécurité de 2 ou 3 tics, et cela ne cause généralement pas grand tort de les ajouter.

Revenons à notre exemple : puisque, suivant la définition du concept de longueur de trame, un tic (ou cycle) vertical représente le temps nécessaire pour dessiner une trame HORIZONTALE complète, il s'ensuit dans notre exemple qu'il vaut 1232 / 65 MHz = 18.95 µs.

L'expérience prouve qu'une impulsion de synchronisation verticale devrait se situer quelque part entre 50 µs et 300 µs. En guise d'illustration, prenons 150 µs, ce qui correspond à 8 tics d'horloge verticaux (150 µs / 18.95 µs ~ 8).

Certains fabricants aiment mentionner leurs paramètres de tramage vertical sous forme de durée plutôt que de largeur exprimée en pixels. Vous pourriez chez eux rencontrer les termes suivants :

durée utile (VAT - Vertical Active Time)

Equivalent à VR, mais en millisecondes. VAT * VSF = VR.

durée hors service (VBT - Vertical Blanking Time)

Equivalent à (VFL - VR), mais en millisecondes. VBT * VSF = (VFL - VR).

fenêtre avant (VFP - Vertical Front Porch)

Synonyme de VGT.

durée de synchronisation

Synonyme de VSP.

fenêtre arrière (VBP - Vertical Back Porch)

Une seconde temporisation de sécurité après l'impulsion de synchronisation verticale. Souvent nulle.

12. Synthèse

La Table des Modes Vidéo du fichier Xconfig contient des séquences de nombres, chaque ligne définissant de façon complète un mode opératoire du serveur X. Les champs sont groupés en quatre sections, à savoir respectivement le nom, la fréquence pilote, les paramètres horizontaux et les paramètres verticaux.

La section consacrée au nom ne contient qu'un champ, le nom du mode vidéo défini par le reste de la ligne. Il sera fait référence à ce nom dans les lignes "Modes" de la section de configuration du pilote de carte graphique du fichier Xconfig. Le champ nom peut être omis si la valeur nom d'une ligne précédente s'applique à la ligne courante.

La section de la ligne de mode consacrée à la fréquence pilote ne contient que ce champ-là (c'est la quantité que nous avions baptisée DCF). La valeur placée dans ce champ spécifie quelle fréquence pilote a été utilisée pour générer les valeurs des sections suivantes.

La section des paramètres horizontaux se compose de quatre champs qui définissent comment chaque ligne horizontale de l'écran doit être construite. Le premier champ de la section contient le nombre de points par ligne qui seront illuminés pour créer l'image (la quantité que nous avons appelée HR). Le second champ de la section (SH1) décrit le point où commencera l'impulsion de synchronisation horizontale. Le troisième champ (SH2) décrit le point où se terminera cette impulsion de synchronisation. Le quatrième champ définit la longueur de trame horizontale totale (HFL).

La section des paramètres verticaux contient aussi quatre champs. Le premier champ contient le nombre de lignes visibles qui apparaîtront sur l'écran (VR). Le second champ (SV1) indique le numéro de la ligne où commencera l'impulsion de synchronisation verticale. Le troisième champ (SV2) définit le numéro de ligne où se terminera cette impulsion de synchronisation. Le quatrième champ contient la longueur de trame verticale totale (VFL).

Exemple :

```
#Nom de mode  horloge  valeurs horizontales  valeurs verticales

"752x564"     40       752 784  944 1088     564 567 569 611
              44.5     752 792  976 1240     564 567 570 600
```

(Note : la version de base de X11R5 ne permet pas l'emploi de fréquences pilotes fractionnaires.)

Pour Xconfig, toutes les valeurs dont nous venons de parler - le nombre de points illuminés sur une ligne, le nombre de points séparant ceux qui sont illuminés du début de l'impulsion de synchronisation, le nombre de points représentant la durée de cette impulsion, et le nombre de points situés après la fin de l'impulsion de synchronisation - sont additionnés pour fournir le nombre de points par ligne. Le nombre de points horizontaux doit être divisible par huit.

Exemple de valeurs horizontales : 800 864 1024 1088

Cette ligne d'exemple comporte le nombre de points illuminés (800) suivi par la coordonnée horizontale du point où commence l'impulsion de synchronisation (864), suivi par la coordonnée horizontale du point où finit cette impulsion (1024), suivi par la coordonnée horizontale du dernier point sur la ligne (1088).

Remarquez encore une fois que chacune des valeurs horizontales (800, 864, 1024, et 1088) est divisible par huit ! Les valeurs verticales ne sont pas frappées de la même exigence.

Le nombre de lignes comprises entre le sommet de la zone affichable et le bas de celle-ci compose la trame. Le signal d'horloge de base, en ce qui concerne la trame, est la ligne. Un certain nombre de lignes composeront l'image. Après que la dernière ligne illuminée a été affichée, un délai correspondant à un nombre précis de lignes sera respecté avant que l'impulsion de synchronisation verticale soit émise. Cette impulsion durera alors quelques lignes, et finalement les dernières lignes de la trame, correspondant au délai requis après l'impulsion, seront générées. Les valeurs qui définissent ce mode opératoire sont fournies d'une manière semblable à l'exemple suivant.

Exemple de valeurs verticales : 600 603 609 630

Cet exemple définit une image composée de 600 lignes visibles, où l'impulsion de synchronisation verticale débute sur la 603ème ligne et se termine sur la 609ème, et composée d'un total de 630 lignes.

Remarquez que les valeurs verticales ne doivent pas nécessairement être divisibles par huit !

Revenons à l'exemple sur lequel nous travaillions. Suivant ce que nous venons d'exposer, tout ce qu'il nous reste à faire est de placer nos résultats dans le fichier Xconfig selon le format suivant :

```
<nom>   DCF     HR  SH1 SH2   HFL   VR  SV1 SV2 VFL
```

où SH1 représente le tic de début de l’impulsion de synchronisation horizontale et SH2 son tic final ; de la même manière, SV1 représente le tic initial de l’impulsion de synchronisation verticale et SV2 son tic final.

Pour définir ces valeurs, souvenez-vous de la discussion relative à la magie noire et aux impulsions de synchronisation que nous avons tenue plus haut. SH1 est le point qui marque le front avant de l’impulsion de synchronisation horizontale ; donc, SH1 = HR + HGT1. SH2 représente le front arrière ; donc, SH2 = SH1 + HSP. De la même manière, SV1 = VR + VGT (mais VGT vaut la plupart du temps 0) et SV2 = SV1 + VSP.

```
#nom     dotclock valeurs horizontales   valeurs verticales   drapeau
936x702   65       936 968 1200 1232   702 702 710 737
```

Aucun drapeau (flag) spécial n’est nécessaire ici ; il s’agit d’un mode non-entrelacé. Maintenant nous avons réellement terminé.

13. Usage du moniteur en surcapacité

Il vous est *fermement déconseillé* de tenter d’imposer à votre moniteur l’usage d’une fréquence de balayage supérieure s’il s’agit d’un moniteur à fréquence fixe. Votre écran peut tout simplement finir grillé ! Avec un moniteur multi-fréquences en surcapacité, il peut survenir des problèmes potentiellement plus subtils, dont il est bon que vous soyez conscient.

Par contre, utiliser une fréquence pilote supérieure à la bande passante maximum du moniteur est relativement inoffensif. C’est le dépassement des fréquences maximales de synchronisation qui est problématique. Les moniteurs modernes ont des circuits de protection qui éteignent l’écran aux fréquences dangereuses, mais ne comptez pas trop dessus. En particulier, il y a des vieux moniteurs multi-fréquences (comme le Multisync II) qui n’utilisent qu’un transformateur horizontal. Ces moniteurs n’offrent pas tellement de protection contre leur usage en surcapacité. Même s’il y a forcément des circuits de régulation haute tension (qui ne sont pas nécessaires dans un moniteur à fréquences fixes), ils ne couvriront pas forcéments tous les intervalles de fréquences possibles, particulièrement dans les modèles bon marché. Il n’y a pas seulement des risques pour les circuits, mais cela risque aussi d’accélérer le vieillissement de la couche de phosphore, et d’augmenter la dose de radiations (dont les rayons X) émise par le moniteur.

Cependant, la valeur qui est la source principale des problèmes est la fréquence d’oscillation (slew rate) - la "pente" des signaux vidéo - à la sortie de la carte graphique, et cela ne dépend normalement pas de la fréquence pilote courante, mais (si le constructeur de votre carte se préoccupe de ces questions) est lié à la fréquence pilote maximum supportée par la carte.

Aussi, soyez prudents...

14. Utilisation des modes entrelacés

(Cette section est dûe en majeure partie à David Kastrup <dak@pool.informatik.rwth-aachen.de>)

A une fréquence pilote donnée, un écran entrelacé manifestera un effet de clignotement considérablement moins perceptible qu’un écran non- entrelacé, si les circuits verticaux de votre moniteur sont capables de supporter cette fréquence de façon stable. C’est la raison pour laquelle ces modes entrelacés furent inventés au départ.

Les modes entrelacés doivent leur mauvaise réputation au fait que leur qualité est inférieure au mode non-entrelacé équivalent à la même fréquence de balayage verticale, VSF (celle qui est généralement mentionnée dans les publicités). Mais ils sont certainement d'une qualité supérieure à la même fréquence de balayage horizontale, et c'est là que se situent les limites décisives de votre couple moniteur/carte graphique.

À une *fréquence de rafraîchissement* donnée (ou fréquence de demi-trame, ou VSF), l'affichage entrelacé manifestera un clignotement plus fort : un affichage entrelacé à 90 Hz sera d'une qualité inférieure à un affichage non-entrelacé à 90 Hz. Il ne nécessitera, cependant, que la moitié de la bande passante vidéo et la moitié de la fréquence de balayage horizontale. Si vous le comparez à un mode non-entrelacé à la même fréquence pilote et aux mêmes fréquences de balayage, vous le trouverez de loin supérieur : 45 Hz non-entrelacé est intolérable. Avec 90 Hz entrelacé, j'ai travaillé des années sur mon Multisync 3D (à 1024x768) et j'en suis tout à fait satisfait. J'ai l'impression qu'il vous faudrait au moins du 70 Hz non-entrelacé pour atteindre un niveau de confort équivalent.

Voici quelques éléments auxquels il vous faudra prêter attention, cependant : n'employez les modes entrelacés qu'à des résolutions élevées, de sorte que les lignes illuminées alternativement soient proches les unes des autres. Vous avez aussi la possibilité de jouer sur la largeur et la position de l'impulsion de synchronisation pour obtenir le positionnement des lignes le plus stable. Si des lignes sont alternativement claires et foncées, l'entrelacement va vous "sauter à la figure". J'utilise un programme qui emploie une configuration de points semblable pour l'arrière-plan d'un menu (il s'agit d'XCept - aucun autre programme à ma connaissance ne fait cela, heureusement).

Pour la même raison, utilisez au moins des polices 100 dpi, ou toute autre police dont les barres horizontales font au moins deux lignes d'épaisseur (dans le cas de hautes résolutions, cela n'a pas de sens d'utiliser quoi que ce soit d'autre de toute façon).

Et bien sûr, n'utilisez jamais de mode entrelacé si votre matériel est capable de supporter l'emploi d'un mode non-entrelacé à la même fréquence de rafraîchissement.

Si, au contraire, vous remarquez qu'à certaines résolutions vous poussez soit le moniteur soit la carte graphique jusqu'à ses limites, et que vous obtenez un clignotement désagréable ou une image délavée (signe que vous saturez la bande passante disponible), peut-être souhaiterez-vous essayer d'obtenir la même résolution par l'emploi d'un mode entrelacé. Ceci bien sûr n'a de sens que si le VSF de votre moniteur n'est pas encore proche de sa limite.

La mise au point d'un mode entrelacé est facile : procédez exactement comme pour un mode non-entrelacé. Il suffit de prendre en compte deux éléments supplémentaires : il vous faut un nombre total impair de lignes verticales (la dernière valeur de votre ligne de mode), et lorsque vous spécifiez le drapeau "Interlace", la fréquence de trame verticale réelle pour votre moniteur est doublée. Il faut que votre moniteur supporte une fréquence de trame de 90 Hz si le mode que vous spécifiez semble être du 45 Hz, le drapeau "Interlace" mis à part.

En guise d'exemple, ceci est ma ligne de mode pour du 1024x768 entrelacé : mon Multisync 3D supporte jusqu'à 90 Hz vertical et 38 kHz horizontal.

```
ModeLine "1024x768" 45 1024 1048 1208 1248 768 768 776 807 Interlace
```

Les deux limites sont pour ainsi dire atteintes avec ce mode. Mais si vous spécifiez le même mode, en omettant simplement le drapeau "Interlace", vous vous situez toujours à la limite de la capacité horizontale du moniteur (et, à strictement parler, un poil en dessous de la limite inférieure de la fréquence de balayage verticale), tandis que vous souffrirez d'un insupportable effet de clignotement de l'image.

Quelques règles de base : si vous avez mis au point un mode qui n'exploite que la moitié de la capacité verticale de votre moniteur, changez le nombre total de lignes en une valeur impaire et ajoutez le drapeau "Interlace". La qualité de l'image devrait s'en trouver grandement améliorée dans la majorité des cas.

Si vous utilisez un mode non-entrelacé qui par ailleurs dépasse les spécifications de votre moniteur tandis que la fréquence de balayage verticale se situe à 30% ou plus en dessous du maximum supporté par votre écran, mettre au point manuellement un mode entrelacé (qui offrira sans doute une résolution légèrement supérieure) peut fournir de meilleurs résultats, mais je ne peux rien promettre.

15. Questions et réponses

Q : L'exemple présenté plus haut met en scène une taille d'écran non standard. Puis-je l'utiliser ? [p 24]
Q : Est-ce là la seule résolution utilisable compte tenu d'une fréquence pilote de 65 MHz et d'un HSF de 55 kHz ? [p 24]
Q : Vous n'avez fait mention que de deux résolutions standard. Dans le fichier Xconfig, de nombreuses résolutions standard sont disponibles. Pouvez-vous me dire s'il y a une raison pour moi de chipoter avec les valeurs de synchronisation ? [p 24]
Q : Pourriez-vous résumer ce qui a été exposé jusqu'ici ? [p 24]

Q : L’exemple présenté plus haut met en scène une taille d’écran non standard. Puis-je l’utiliser ?

R : Pourquoi pas ? Il n’y a absolument AUCUNE raison qui vous force à vous cantonner aux tailles habituelles 640x480, 800x600, ou même 1024x768. Les serveurs XFree86 vous offrent énormément de liberté lors de la configuration de votre matériel. Il faut généralement deux ou trois essais pour obtenir la bonne. L’objectif à garder en ligne de mire est une fréquence de rafraîchissement élevée tout en conservant une zone affichable de taille raisonnable. Ne visez pas une résolution élevée si c’est au prix d’un clignotement qui vous ruinera les yeux !

Q : Est-ce là la seule résolution utilisable compte tenu d’une fréquence pilote de 65 MHz et d’un HSF de 55 kHz ?

R : Pas du tout ! Nous vous exhortons au contraire à suivre la procédure générale décrite plus haut et à vous livrer à quelques expériences afin d’obtenir une configuration qui vous plaise réellement. Vous livrer à ces expériences peut vous apporter énormément de plaisir. La plupart des configurations risquent de ne produire qu’une horrible neige, mais en pratique un écran moderne multi-fréquences n’est pas si facile à endommager. Soyez sûr, cependant, que votre moniteur peut supporter la fréquence de trame que vous voulez lui infliger avant de l’utiliser pour un temps considérable.

Mais méfiez-vous des moniteurs à fréquence fixe ! Ce type de manipulations hasardeuses peut les endommager très rapidement. Soyez sûr que vous utilisez une fréquence de rafraîchissement supportée lors de *chaque* tentative.

Q : Vous n’avez fait mention que de deux résolutions standard. Dans le fichier Xconfig, de nombreuses résolutions standard sont disponibles. Pouvez-vous me dire s’il y a une raison pour moi de chipoter avec les valeurs de synchronisation ?

R : Tout à fait ! Prenez par exemple le "standard" 640x480 mentionné dans le fichier Xconfig actuel. Il exploite une fréquence pilote de 25 Mhz, les longueurs de trame sont de 800 et 525 ce qui nous donne une fréquence de rafraîchissement d’environ 59.5 Hz. Pas trop mal. Mais 28 MHz est une fréquence pilote généralement disponible sur nombre de cartes SVGA. Si vous utilisiez celle-ci pour générer du 640x480, suivant la procédure illustrée plus haut, vous obtiendriez des longueurs de trame de 812 (arrondi à 808) et 505. La fréquence de rafraîchissement est maintenant portée à 68 Hz, ce qui représente une amélioration significative par rapport à la configuration standard.

Q : Pourriez-vous résumer ce qui a été exposé jusqu’ici ?

R : En bref :

- pour toute fréquence pilote donnée, une augmentation de la résolution maximum se paye par une diminution de la fréquence de rafraîchissement, ce qui induira un effet de clignotement accru.
- si l’obtention d’une résolution élevée s’avère nécessaire et si votre moniteur la supporte, essayez de vous procurer une carte SVGA qui fournisse une fréquence pilote (ou DCF) compatible. Plus elle est élevée, meilleur sera le résultat !

16. Résoudre les problèmes affectant l’image

OK, ainsi donc vous disposez maintenant des valeurs nécessaires à la configuration de X. Vous les avez placées dans votre fichier Xconfig, en choisissant pour le mode un champ "nom" qui l’identifie comme un test. Vous démarrez X, vous utilisez les raccourcis clavier pour sauter au nouveau mode, ... et l’image ne semble pas correcte. Que faire ? Ci-dessous se trouve une liste de distorsions d’image vidéo fréquentes et la façon d’y remédier.

(La réparation de ces distorsions mineures est réellement la situation où**xvidtune**(1) se distinguera brillamment.)

Vous *déplacerez* l’image en modifiant les coordonnées temporelles de l’impulsion de synchronisation. Vous *agirez sur sa taille* en jouant sur la longueur de trame (n’oubliez pas de déplacer l’impulsion de synchronisation en conséquence, de façon à la maintenir à la même position relative, sinon altérer la taille de l’image déplacera aussi celle-ci). Voici quelques recettes plus spécifiques :

Les positions horizontale et verticale sont indépendantes. Par cela, nous entendons que déplacer l’image horizontalement n’affecte pas sa position verticale, et réciproquement. Cependant, il n’en va pas tout à fait de même pour la taille. Alors que le fait de modifier la taille horizontale n’affecte en rien la dimension verticale et vice versa, la quantité globale de changement qui peut être apporté dans les deux directions peut être limité. En particulier, si votre image est trop large dans les deux sens il vous faudra probablement adopter une fréquence pilote plus élevée pour la faire rétrécir. Dans la mesure où cela se traduira par une augmentation de la résolution utilisable, c’est rarement un problème !

16.1. L’image est décentrée vers la gauche ou la droite

Pour y remédier, déplacez l’impulsion de synchronisation horizontale. C’est-à-dire, incrémentez ou décrémentez (par un multiple de 8) les deux valeurs médianes de la section des données horizontales qui déterminent les limites antérieure et postérieure de l’impulsion de synchronisation horizontale.

Si l’image est décentrée vers la gauche (la marge droite étant trop large, vous souhaitez faire glisser l’image vers la droite) décrémentez les valeurs. Si l’image est décentrée vers la droite (marge gauche trop large, vous voulez faire glisser l’image vers la gauche) incrémentez les coordonnées de l’impulsion de synchronisation.

16.2. L’image est décentrée vers le haut ou le bas

Pour résoudre ce problème, déplacez l’impulsion de synchronisation verticale. A savoir, incrémentez ou décrémentez les deux valeurs médianes de la section des données verticales qui déterminent les limites antérieure et postérieure de l’impulsion de synchronisation verticale.

Si l’image est décentrée vers le haut (marge inférieure trop large, vous souhaitez faire glisser l’image vers le bas) décrémentez les valeurs. Si l’image est décentrée vers le bas (marge supérieure trop large, vous souhaitez faire glisser l’image vers le haut) incrémentez les valeurs.

16.3. L’image est trop large dans les deux directions

Passez à une fréquence d’horloge supérieure sur la carte. Si vous disposez de nombreux modes différents dans votre fichier de configuration, il est possible que l’un de ceux qui exploitent une fréquence inférieure ait été activé par erreur.

16.4. L'image est trop large (ou trop étroite) horizontalement

Pour remédier à cela, augmentez (ou diminuez) la longueur de trame horizontale. C'est-à-dire, modifiez la quatrième valeur de la première section des coordonnées temporelles. Pour éviter de déplacer l'image par la même occasion, déplacez aussi l'impulsion de synchronisation (deuxième et troisième valeurs) de la moitié de la différence, de façon à la conserver à la même position relative.

16.5. L'image est trop grande (ou trop petite) verticalement

Pour résoudre ce problème, augmentez (ou diminuez) la longueur de trame verticale. C'est-à-dire, modifiez la quatrième valeur dans la deuxième section des coordonnées temporelles. Pour éviter de déplacer l'image, n'oubliez pas de déplacer aussi l'impulsion de synchronisation (deuxième et troisième valeurs) de la moitié de la différence, pour la garder à la même position relative.

Toute distorsion qui ne peut être éliminée en combinant ces techniques est probablement la preuve d'un problème plus profond, comme une erreur dans les calculs ou une fréquence pilote supérieure aux limites du moniteur.

En dernier lieu, souvenez-vous que si vous augmentez l'une des longueurs de trame, vous diminuerez du même coup votre fréquence de rafraîchissement, et réciproquement.

Occasionnellement, vous pouvez corriger les petites distorsions d'image en jouant sur les réglages de votre moniteur. Le défaut de cette méthode est que si vous vous éloignez trop des réglages d'usine pour régler les problèmes du mode graphique, vous pouvez vous retrouver avec une image inutilisable en mode texte. Il vaut mieux régler vos lignes de modes.

17. Représentation graphique des capacités du moniteur

Pour tracer le diagramme de mode d'un moniteur, il vous faudra le programme gnuplot (un logiciel libre de tracé graphique) et l'outil **modeplot**, un script shell/gnuplot qui tracera le diagramme sur la base des caractéristiques de votre moniteur, fournies comme options sur la ligne de commande.

Voici le listing du script **modeplot** :

```
#!/bin/sh
#
# modeplot -- crée un graphe X des modes moniteurs disponibles
#
# Invoquez 'modeplot -?' pour afficher les options de contrôle.
#

# Description du moniteur. Bande passante en MHz, fréquences horizontales
# en kHz et fréquences verticales en Hz.
TITLE="Viewsonic 21PS"
BANDWIDTH=185
MINHSF=31
MAXHSF=85
MINVSF=50
MAXVSF=160
ASPECT="4/3"
vesa=72.5       # fréq. de rafraîchissement min. recommandée VESA

while [ "$1" != "" ]
do
```

```
        case $1 in
        -t) TITLE="$2"; shift;;
        -b) BANDWIDTH="$2"; shift;;
        -h) MINHSF="$2" MAXHSF="$3"; shift; shift;;
        -v) MINVSF="$2" MAXVSF="$3"; shift; shift;;
        -a) ASPECT="$2"; shift;;
        -g) GNUOPTS="$2"; shift;;
        -?) cat <<EOF
options de contrôle de modeplot :

-t "<description>"  nom du moniteur            défaut : "Viewsonic 21PS"
-b <nn>                 bande passante en MHz      défaut : 185
-h <min> <max>          HSF min & max (kHz)        défaut : 31 85
-v <min> <max>          VSF min & max (Hz)         défaut : 50 160
-a <aspect ratio>       proportions de l'image     défaut : 4/3
-g "<options>"      options à transmettre à gnuplot

La présence des paramètres -b, -h et -v est requise, -a, -t, -g sont
optionnels.  Vous pouvez utiliser -g pour transmettre un nom de
périphérique à gnuplot de façon à ce que (par exemple) les sorties
produites par modeplot puissent être redirigées vers une imprimante.
Voyez gnuplot(1) pour de plus amples détails.

L'outil modeplot a été conçu par Eric S. Raymond <esr@thyrsus.com> sur
base d'une analyse et d'un code original par Martin Lottermoser
<Martin.Lottermoser@mch.sni.de>

Voici modeplot $Revision: 1.20 $
EOF
                exit;;
        esac
        shift
done

gnuplot $GNUOPTS <<EOF
set title "$TITLE Mode Plot"

# Nombres magiques.  Malheureusement, le graphe est très sensible à toute
# modification de ceux-ci, et ils pourraient être loin de la vérité dans
# le cas de certains moniteurs.  Il nous faut déterminer des valeurs afin
# d'obtenir même une approximation du diagramme de mode. Ces valeurs
# proviennent d'une comparaison de nombreux exemples fournis dans la base
# de données ModeDB.
F1 = 1.30       # facteur de conversion résol. horiz. -> largeur de trame
F2 = 1.05       # facteur de conversion résol. vertic. -> hauteur de trame

# Définition de fonctions
# (multiplier par 1.0 force l'arithmétique en nombres réels)
ac = (1.0*$ASPECT)*F1/F2
refresh(hsync, dcf) = ac * (hsync**2)/(1.0*dcf)
dotclock(hsync, rr) = ac * (hsync**2)/(1.0*rr)
resolution(hv, dcf) = dcf * (10**6)/(hv * F1 * F2)

# Place les légendes le long des axes
set xlabel 'DCF (MHz)'
set ylabel 'RR (Hz)' 6  # Place la légende juste au-dessus de l'axe des Y

# Génère le diagramme
set grid
set label "VB" at $BANDWIDTH+1, ($MAXVSF + $MINVSF) / 2 left
set arrow from $BANDWIDTH, $MINVSF to $BANDWIDTH, $MAXVSF nohead
set label "max VSF" at 1, $MAXVSF-1.5
```

```
set arrow from 0, $MAXVSF to $BANDWIDTH, $MAXVSF nohead
set label "min VSF" at 1, $MINVSF-1.5
set arrow from 0, $MINVSF to $BANDWIDTH, $MINVSF nohead
set label "min HSF" at dotclock($MINHSF, $MAXVSF+17), $MAXVSF + 17 right
set label "max HSF" at dotclock($MAXHSF, $MAXVSF+17), $MAXVSF + 17 right
set label "VESA $vesa" at 1, $vesa-1.5
set arrow from 0, $vesa to $BANDWIDTH, $vesa nohead # style -1
plot [dcf=0:1.1*$BANDWIDTH] [$MINVSF-10:$MAXVSF+20] \
  refresh($MINHSF, dcf) notitle with lines 1, \
  refresh($MAXHSF, dcf) notitle with lines 1, \
  resolution(640*480,   dcf) title "640x480  " with points 2, \
  resolution(800*600,   dcf) title "800x600  " with points 3, \
  resolution(1024*768,  dcf) title "1024x768 " with points 4, \
  resolution(1280*1024, dcf) title "1280x1024" with points 5, \
  resolution(1600*1280, dcf) title "1600x1200" with points 6

pause 9999
EOF
```

Une fois que vous êtes sûr d'avoir correctement installé **modeplot** et le programme gnuplot, il vous faudra réunir les caractéristiques suivantes de votre moniteur :

- bande passante vidéo (VB)
- gamme de fréquences de synchronisation horizontale (HSF)
- gamme de fréquences de synchronisation verticale (VSF)

Le programme de tracé doit faire quelques suppositions simplificatrices qui ne sont pas nécessairement correctes. C'est la raison pour laquelle le diagramme résultant n'est qu'une description relativement grossière. Ces suppositions sont les suivantes :

- Toutes les résolutions n'ont qu'un seul rapport de proportions prédéfini AR = HR/VR. Les résolutions standard ont AR = 4/3 ou AR = 5/4. Le programme modeplot suppose 4/3 par défaut, Mais il vous est possible de modifier cela.
- Pour les modes pris en compte, les longueurs de trames horizontale et verticale sont des multiples fixés des résolutions horizontale et verticale, respectivement :

```
HFL = F1 * HR
VFL = F2 * VR
```

En première approximation, prenez F1 = 1.30 et F2 = 1.05 (voyez la section Calcul de la taille de trame [p 14]).

Maintenant considérez une fréquence de synchronisation particulière, HSF. Sur base des présupposés que nous venons de mentionner, la valeur que prendra la fréquence pilote DCF déterminera déjà la fréquence de rafraîchissement RR, c'est-à-dire que pour toute valeur de HSF il y a une fonction RR(DCF). Celle-ci peut s'obtenir comme suit.

La fréquence de rafraîchissement est égale à la fréquence pilote divisée par le produit des longueurs de trame :

```
RR = DCF / (HFL * VFL)          (*)
```

D'autre part, la longueur de trame horizontale est égale à la fréquence pilote divisée par la fréquence de synchronisation horizontale :

```
HFL = DCF / HSF                  (**)
```

Il est possible de réduire VFL à HFL au moyen des deux suppositions mentionnées plus haut :

```
VFL = F2 * VR
    = F2 * (HR / AR)
    = (F2/F1) * HFL / AR         (***)
```

En insérant (**) et (***) dans (*) nous obtenons :

```
RR = DCF / ((F2/F1) * HFL**2 / AR)
   = (F1/F2) * AR * DCF * (HSF/DCF)**2
   = (F1/F2) * AR * HSF**2 / DCF
```

Pour des valeurs fixes de HSF, F1, F2 et AR, cette formule se traduit par une hyperbole dans notre diagramme. Si nous traçons deux courbes de ce type pour les fréquences de synchronisation horizontale minimum et maximum nous obtenons les deux limites restantes de la région des solutions permises.

Les lignes droites qui traversent la région des capacités représentent des résolutions particulières. Ceci est basé sur (*) et la deuxième supposition :

```
RR = DCF / (HFL * VFL) = DCF / (F1 * HR * F2 * VR)
```

En traçant de telles droites pour chacune des résolutions qui vous intéressent, vous pourrez immédiatement extraire du graphe les relations possibles entre résolution, fréquence pilote et fréquence de rafraîchissement dont le moniteur est capable. Remarquez que ces lignes ne dépendent pas des caractéristiques réelles du moniteur, mais bien de notre seconde supposition.

L'outil **modeplot** vous offre une manière très simple de faire cela. Tapez **modeplot -?** pour afficher ses options de contrôle. Une invocation usuelle ressemble à ceci :

```
modeplot -t "Swan SW617" -b 85 -v 50 90 -h 31 58
```

Le paramètre -b spécifie la bande passante vidéo ; -v et -h définissent les gammes de fréquences de synchronisation horizontale et verticale.

Lorsque vous consulterez le graphique produit par **modeplot**, conservez toujours présent à l'esprit le fait qu'il n'offre qu'une description approximative. Par exemple, il ignore les limitations imposées à HFL dues à ce qu'une largeur d'impulsion de synchronisation minimum est requise, et sa précision ne peut dépasser celle de nos suppositions. Il ne peut dès lors remplacer un calcul détaillé (incluant une certaine dose de magie noire) tel celui que nous avons présenté dans la section Synthèse [p 19] . Il devrait, cependant, vous offrir une meilleure perception de ce qui est possible et des concessions que cela implique.

18. Crédits

L'ancêtre primordial de ce document est dû à Chin Fang <fangchin@leland.stanford.edu>.

Eric S. Raymond <esr@snark.thyrsus.com> a retravaillé, réorganisé, et largement ré-écrit l'original de Chin Fang dans le but de le comprendre. Au cours de cette opération, il y a inclus la plus grande part d'un autre Howto écrit par Bob Crosson <crosson@cam.nist.gov>.

Les informations consacrées aux modes entrelacés sont en grande partie dues à David Kastrup <dak@pool.informatik.rwth-aachen.de>

Nicholas Bodley <nbodley@alumni.princeton.edu> a corrigé et clarifié la section sur le fonctionnement des écrans.

Payne Freret <payne@freret.org> a corrigé quelques erreurs mineures sur la conception des moniteurs.

Martin Lottermoser <Martin.Lottermoser@mch.sni.de> a soumis l'idée d'utiliser gnuplot pour générer des diagrammes de mode et a réalisé l'analyse mathématique qui sous-tend **modeplot**. La version de modeplot telle qu'elle est actuellement distribuée a été repensée et généralisée par ESR à partir du code gnuplot original de Martin pour un cas précis.