ConfigXF86

A Configuration program for XFree86 version 2.x

By Steven T. Zwaska
March 28, 1994

Table of Contents

Table of Contents 2
Before you start: 3
Setting Up Your Environment 3
Running the Configuration Program: 3
Configuring the Mouse 4
Specifying the Video Card and Monitor 5
Specifying other Options 5
Virtual Desktop Size: 5
Viewport Location: 6
Server Options: 6
Font Server Specification: 7
Selecting the Active Modes: 8
Anatomy of the Video Card and Monitor Specification: 10
The Xconfig file 11
The Driver Section: 11
The Video Card Dot Clocks: 12
The Modes Line: 12
The ModeDB Section: 13
Horizontal timing in simple terms: 14
Vertical timing in simple terms: 16
The Tuning and Testing Process 17
Saving the Xconfig File: 20
Discrete Scan Line Tuning: 21
Registering New Monitor Descriptions: 23
Using MakeCard: 24

Before you start:

You will want to have several pieces of information at your finger tips:

Your mouse type, port location (tty number) and port baud rate.

Your video card type (brand)

Your monitor manual

This program cannot be run from inside the X environment. Be sure you are working from a character based screen.

Setting Up Your Environment

Make sure your *XWINHOME* environment variable is set. For most systems this should be */usr/X386.*

To set the environment variable in Bourne shell or Korn shell type:

```
XWINHOME=/usr/X386; export XWINHOME
```

For C shell type:

```
setenv XWINHOME /usr/X386
```

Running the Configuration Program:

If you are running this program from the command line as opposed to from inside an install script, type the following:

```
cd $XWINHOME/lib/X11/etc/ConfigXF86
./ConfigXF86
```

An introduction screen appears while the program is loading.

After the introduction screen clears you will be presented with a list of mouse types.

Configuring the Mouse

A sample screen is shown below

```
The first step in the configuration is to configure your mouse
driver, input port, and/or port speed...

              Select your mouse type:
      1)      xqueue          (Use this for SVR4 or SVR4.2
      2)      busmouse        (Use this for ANY busmouse)
      3)      logitech        (Logitech serial mouse only)
      4)      microsoft       (Serial mouse only)
      5)      mmseries
      6)      mouseman        (Serial mouse only)
      7)      mousesystems    (Serial mouse only)
      8)      ps/2
      9)      mmhittab        (Hitachi in MM mode)
           >
```

If you are working with UnixWare or another SVR4 (Dell, Consensys, ESIX etc.) select the first entry (xqueue) for true simplicity. This allows the operating system's mouse manager to control the mouse, and requires no further information about the mouse. If you are NOT working with one of the SVR4 operating systems enter your mouse type. Use the second selection (busmouse) for ALL bus type mice. If you are using a 2 button mouse you may opt to answer [y] to the next prompt regarding three button emulation. This allows you to use the combination of the left and right mouse buttons to simulate the middle button of a 3 button mouse. If your mouse type requires additional information you will be asked about the tty port location and baud rate. In the case you don't know what your baud rate is, enter [h] at the baud rate prompt for some examples.

Once the your mouse has been specified the system will want to know about your video hardware.

Specifying the Video Card and Monitor

First, you will be presented with a list of video cards that are known to the system. Pressing return will move you through the list in a circular fashion. Select the card that most closely resembles yours. If you don't see one press [Delete] or [Ctrl - C] (which ever your system uses as an interrupt key) to bail out of the program. You will need to run MakeCard to produce a new Card. See the section on how to produce a new card class and submit it to the XFree86 corporation for inclusion into the database.

When you find the card class that resembles yours, enter the class number (Along the left edge of the screen).

The system will now need to know about your Monitor. Scroll through the list to find a monitor that resembles yours. If you can't find one that matches yours, you can select the first record (1 VESA-Generic Modes), and use it as a starting point.

Once you've selected a monitor class the system will ask you about certain options available to you:

Specifying other Options

Virtual Desktop Size:

XFree86 supports a virtual desktop - a default root window that may be larger than the physical displayed resolution on your screen (refer to the manual page for more information). The default Virtual desktop size is normally unset. Leaving the virtual size unset will cause the server to calculate the correct size based on the specified display modes. You can press [Enter] to accept this default, or enter new horizontal and vertical numbers to change the defaults. The Virtual Desktop Size is entered in the following format:

<Width> <Height>

for example, for a virtual screen size of 1280 X 1024 you would enter:

```
1280 1024
```

Viewport Location:

The Viewport defines the initial upper left corner of the displayed portion of the virtual root window, if it is larger than the physical screen resolution. The default ViewPort is:

0 0

You can press [Enter] to accept this, or enter new X and Y coordinates in the format:

<Pixel Column> <Pixel Row>

for example, to center an actual screen size of 1024 X 768 on a Virtual Desktop of 1280 X 1024 you would enter a location of:

```
128 128
```

this number is arrived at by the following formula:

((Virtual Desktop Width) - (Actual Screen Width)) ÷ 2 = First digit
((1280) - (1024)) ÷ 2 = 128

((Virtual Desktop Height) - (Actual Screen Height)) ÷ 2 = Second digit
((1024) - (768)) ÷ 2 = 128

Server Options:

The video driver for your chipset/board may support option flags, as defined in the appropriate server manual page. It is advised that you NOT change the defaults unless you know what you are doing. The default list of options for your chipset/card is:

none

You can accept this default list by pressing [Enter], or you can create your own list of options by entering them here. If you want to add more than one option separate them with a space. Do not enclose the options in quotation marks, as this is done automatically in the configuration program.

Font Server Specification:

In addition to the default font directory list, you may have access to a font server on your network. If you would like to have the X server access a font server answer [y] to the next prompt, if you are not sure or don't know the TCP/IP port number of the font server answer [n] to this prompt and you may add this feature to your Xconfig file manually at a later time (check the Xconfig(4/5) manual page for instructions and examples). If you do know this information, answer [y] and when prompted enter the host name and port number in the format: *<hostname>:<tcp_port_number>* , for example if your font server is on the machine named 'zok' at port number 7000 your entry would be:

```
zok:7000
```

You may now choose to have the server check the font server either before or after the default font directories answer either [b] (before) or [a] (after) to the next prompt.

Selecting the Active Modes:

The program now runs a set of calculations against the selected data to find corresponding video card dot clocks and monitor clock rates to see what resolutions will be supported by your selections. It builds a temporary Xconfig file in the /tmp directory, and presents you with a list of supported video modes. You can switch between modes in XFree86 servers using CTRL-ALT-[+] and CTRL-ALT-[-]. In order to take advantage of this feature you must have more than one mode defined. An example screen is shown below:

```
Here are the modes I was able discern from your hardware:
     1  "1024x768" - Clock=65MHz, Horiz. sync=48kHz, Vert. refresh=60Hz
     2  "1024x768" - Clock=75MHz, Horiz. sync=56kHz, Vert. refresh=70Hz
     3  "1024x768i" - Clock=43MHz, Horiz. sync=34kHz, Vert. refresh=84Hz
     4  "1024x768i" - Clock=44MHz, Horiz. sync=35kHz, Vert. refresh=86Hz
     5  "1024x768i" - Clock=45MHz, Horiz. sync=35kHz, Vert. refresh=86Hz
     6  "640x480" - Clock=25MHz, Horiz. sync=31kHz, Vert. refresh=60Hz
     7  "640x480" - Clock=31MHz, Horiz. sync=37kHz, Vert. refresh=72Hz
     8  "800x600" - Clock=36MHz, Horiz. sync=35kHz, Vert. refresh=56Hz
     9  "800x600" - Clock=40MHz, Horiz. sync=38kHz, Vert. refresh=61Hz
    10  "800x600" - Clock=50MHz, Horiz. sync=48kHz, Vert. refresh=72Hz

Order them as you would like them (Leave out the mode numbers you
don't want) put the default mode number first.

Separate the mode numbers with SPACES.

Pressing [Enter] without entering any numbers will save the list as is.

Entering only a 'u' will re-set the list to the original order and
allow you to start over:
```

Select the modes you want to run (and whose clock values are supported by your monitor in the case of a Vesa/Generic monitor selection), in the order you want them available. For example if you want the following modes available in the following order:

1 "1024x768" - Clock=65MHz, Horiz. sync=48kHz, Vert. refresh=60Hz
4 "1024x768i" - Clock=44MHz, Horiz. sync=35kHz, Vert. refresh=86Hz
9 "800x600" - Clock=40MHz, Horiz. sync=38kHz, Vert. refresh=61Hz
6 "640x480" - Clock=25MHz, Horiz. sync=31kHz, Vert. refresh=60Hz

Then your input line would look like this:

```
1 4 9 6
```

Now the screen will clear and come back with your changes:

```
Here are the modes you selected:

     1  "1024x768" - Clock=65MHz, Horiz. sync=48kHz, Vert. refresh=60Hz
     4  "1024x768i" - Clock=44MHz, Horiz. sync=35kHz, Vert. refresh=86Hz
     9  "800x600" - Clock=40MHz, Horiz. sync=38kHz, Vert. refresh=61Hz
     6  "640x480" - Clock=25MHz, Horiz. sync=31kHz, Vert. refresh=60Hz

Order them as you would like them (Leave out the mode numbers you
don't want) put the default mode number first.

Separate the mode numbers with SPACES.

Pressing [Enter] without entering any numbers will save the list as is.

Entering only a 'u' will re-set the list to the original order and
allow you to start over:
```

By entering a [u] you can bring back the original list and re-enter a different set if you didn't get what you wanted the first time, or you can delete or re-order from the current list by entering the numbers in a different order (leaving out any you don't want).

If everything is fine simply press [Enter] to save the list as it is.

The screen will clear and the following message will be displayed
(The ATI 8514 Ultra on my system was used as an example thus displaying the XF86_Mach8 server your message will be different depending on which graphics card you selected):

```
The new X Server will be XF86_Mach8


You may now save the new Xconfig file as it is, or, you may run the tune
and test utility 'Fiddle'.  It is HIGHLY recommended that you run through
the tuning procedures, especially if you have chosen to use the
VESA/Generic modes.

Selecting [v]iew lets you view the current Xconfig file.

Selecting [s]ave saves the new Xconfig file somewhere useful and enables the
new server.

Selecting [t]une lets you fiddle with the settings then brings you back here
to save them.

Selecting [e]xit will leave the new Xconfig file in /tmp/Xconfig.test and
will not enable the new server, you'll have to do this by hand by using
the command:

        cp /usr/X/bin/XF86_Mach8 /usr/X/bin/X

So what do you want to do? ([v]iew, [s]ave, [t]une or [e]xit)?
```

If you found a monitor selection that matched yours exactly you can save the Xconfig file now and everything should be fine. If you have anything other than a perfect match you will probably want to select the [t] option to test and tune your image. If you can save at this time skip to the **Saving Xconfig** section below.

Anatomy of the Video Card and Monitor Specification:

Before we delve into the fascinating subject of image tuning lets take a look at the structure of the video card and monitor sections of the Xconfig file. If you don't want to know about all of this techno-junk (you really should) then skip to the **Tuning and Testing Process** section.

The numbers along the left side are for reference purposes only and do not show up in the actual Xconfig file, they are only here for the purposes of this discussion.

The Xconfig file

(Only the Video Card and Monitor sections are shown)

```
1       # **********************************************************************
2       # Graphics drivers
3       # **********************************************************************
4       #
5       #
6       accel

7       #----------------------------------------------------------
8       # Video Card Definition
9       # Video Card ID 5
10      # Video Card Name:      ATI 8514 Ultra
11      #                       Should also work for Graphics Ultra
12      # Contributed By:       Steven T. Zwaska <stz@netcom.com>
13      # Last Change:          03-16-94
14      # Chipset Mach8
15      #
16      # For boards with a programmable clock generator, you use a line like:
17      #
18      # Clocks "icd2061a"
19      # Then the Modes line may contain any mode the Monitor section has

20      Clocks 43.00 48.80 0.00 36.00 50.30 56.70 0.00 44.90 30.20 32.00 37.50
21      39.00 40.00 89.50 75.50 65.00 21.50 24.40 0.00 18.00 25.15 28.35 0.00
22      22.45 15.10 16.00 18.75 19.50 20.00 44.75 37.75 32.50
23      Modes  "1024x768" "1024x768i" "800x600" "640x480"

24      #----------------------------------------------------------
25      # Monitor Definition
26      # Monitor ID            1
27      # Monitor Name:         VESA-Generic Modes
28      # Contributed By:       David Wexelblat <dwex@goblin.org >
29      # Last Change:          03-03-94
30      ModeDB
31      "1024x768" 65 1024 1080 1224 1344 768 776 778 808
32      "1024x768i" 44 1024 1040 1216 1264 768 768 778 819 Interlace
33      "800x600" 40 800 840 968 1056 600 601 605 628
34      "640x480" 25 640 664 760 800 480 491 493 525
```

The Driver Section:

The entry on line 6 tells the server which driver we'll be using. Each class of video requires a specific type of driver. Normal SVGA video cards would use the 'vga256' driver, cards that only have 16 colors would use the 'vga16'. All Mach 8, Mach 32, 8514 and S3 based video cards would use the 'accel' driver.

Lines 7-14 are a description of the video card itself and the name of the person who submitted the definition.

The Video Card Dot Clocks:

At Line 20 we start getting to the meat of the matter. This list of numbers is the list of available dot clocks that the video card can generate. The list can be reproduced by running the following command:

(With the correct server in place as *$XWINHOME/bin/X* type:)

```
X -probeonly >xinit.out 2>&1
```

When you are returned to a prompt the file x*init.out* will contain a bunch of information that the server was able to discern from the hardware installed. Among that data is a list of the available dot clocks in the form:

(--) Mach8: clocks: 43.00 48.80 0.00 36.00 50.30 56.70 0.00 44.90
(--) Mach8: clocks: 30.20 32.00 37.50 39.00 40.00 89.30 75.50 65.00
(--) Mach8: clocks: 21.50 24.40 0.00 18.00 25.15 28.35 0.00 22.45
(--) Mach8: clocks: 15.10 16.00 18.75 19.50 20.00 44.65 37.75 32.50

Where (--) means that these are probed values as opposed to (**) which means that the 'Clocks' values were supplied in an *Xconfig* file - If the 'Clocks' line is omitted from the *Xconfig* file the server will probe the hardware for available clocks at runtime. Mach8 is the server that was running (this may be different for your hardware setup).

If you have a graphics card with a supported programmable clock chip (i.e. Diamond Stealth 24, Elsa Winner 1000, Number 9 GXE to name a few) you will handle the 'Clocks' line totally differently. Rather than a list of available clock rates, the clock chip name is entered in the form:

```
Clocks "icd2061a"
      or
Clocks "sc11412" (for the SPEA Mercury card)
```

The Modes Line:

Then the Modes line may contain any mode the Monitor section has.

Line 23 in our example is the 'Modes' line. It contains a list of modes you want to have available in the order you want them selected (via CTRL-ALT-[+] and CTRL-ALT-[-]). If you only intend to run in a single mode then only put that mode on this line. The mode name MUST match a compatible mode in the 'ModeDB' section (we'll discuss the ModeDB section later).

By 'compatible' I mean that the clock rate of that mode (the second field of the ModeDB mode entry) corresponds to one of the dot clocks in the 'Clocks' line.

For example, the following 'Clocks' line and 'ModeDB' entry are compatible since the 'ModeDB' entry has a clock rate of 44 and the 'Clocks' line has an entry of 44.90:

Clocks 43.00 48.80 0.00 36.00 50.30 56.70 0.00 **44.90** 30.20 32.00 37.50

"1024x768i" **44** 1024 1040 1216 1264 768 768 778 819 Interlace

However, a 'ModeDB' entry clock rate of 45 would also work because the server can round up and down.

The ModeDB Section:

The ModeDB entries are the most complicated thing about the Xconfig file. Once you understand how they work, however. tuning your image becomes a fairly simple process even if done by hand.

We'll use the following ModeDB line as an example:

"1024x768" 65 1024 1080 1224 1344 768 776 778 808

The first two fields are the mode name and the clock rate that we discussed in the previous section. We'll ignore those. We'll also for the time being ignore the 'flags' section (in the case of the previous example, the 'Interlace' flag). Lets take a look at the rest of the line from above:

1024 1080 1224 1344 768 776 778 808

It is actually two sets of four numbers each:

Horizontal Timings	Vertical Timings
1024 1080 1224 1344	768 776 778 808

These represent the Horizontal timings and the Vertical timings respectively. Each number is measured in 'pixels' (screen dots). The first number of each is the actual resolution (1024 and 768), the stuff you actually see on the screen. The other three fields are what happens inside the monitor between drawing the visible stuff.

Horizontal timing in simple terms:

The monitor squirts color juice at each pixel location in the form of an electron beam for a period of time. It starts at the upper left corner of the monitor, and works its way to the right and then down to the next line. One complete line of dots is called a 'scan line'. After one complete scan line the beam is moved to the beginning (left side) of the next line.

Each pixel takes, in this case, one 65 millionth (.000,000,015) of a second to draw (Clock rate of 65 (MHz)).

After all 1024 visible pixels are drawn, the monitor rests for a moment (called interestingly enough 'front porch' time).

This is the time between the first number (1024 pixels) and the second number (1080 pixels) which is the beginning of the 'sync pulse' .

The sync pulse time is when the beam is being moved back to the beginning of the next scan line. This take some time to do. That time is the difference between the second number and the third number (Start and End of sync pulse).

The monitor rests for another moment (called, of course, 'back porch' time). This rest time is the difference between the third number and the fourth number, which is the Horizontal Frame Length (or total cycle time).

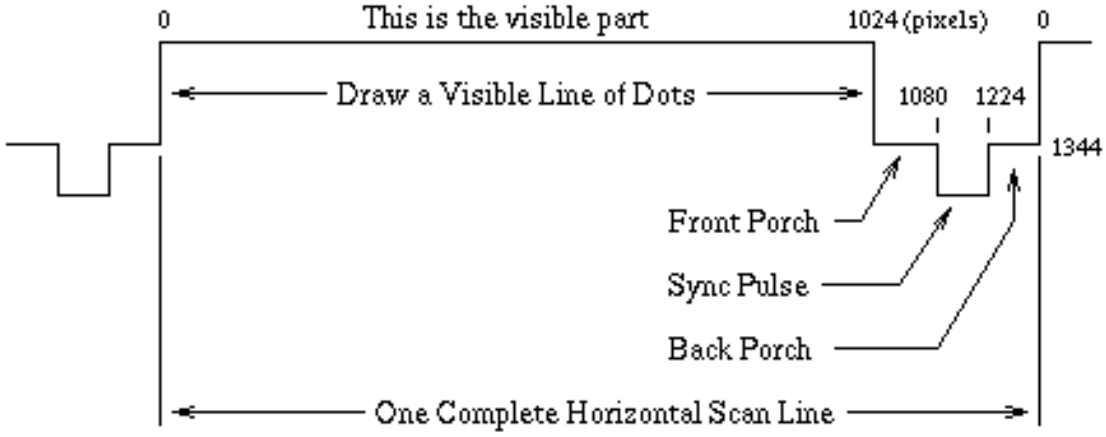
The monitor has now done one complete scan line, and is back at the beginning of the next line ready to start all over again. The time it took to produce the complete scan line is calculated as:

1 ÷ (Clock rate ÷ Horizontal Frame Length)

OR

1 ÷ (65 million ÷ 1344) = 1 ÷ 48,000 = 0.000,021 seconds (48kHz)

Here is a crude graphic representation of the scan line (sort of) showing where the numbers come from:



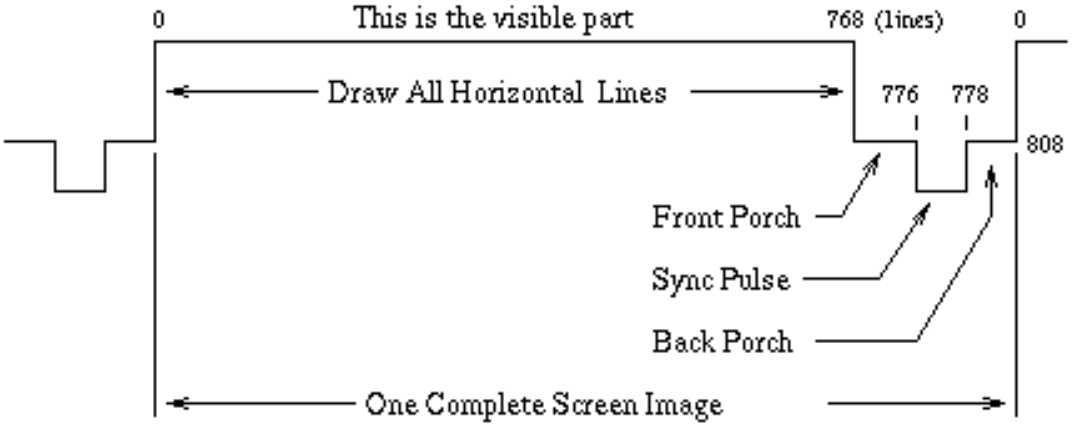
Timing for the Horizontal values : "1024x768" 65 **1024 1080 1224 1344** 768 776 778 808

Vertical timing in simple terms:

Now let's look at the second set of four numbers. The meaning of the numbers is essentially the same, except that rather than representing pixel times, the numbers represent horizontal scan line times.

The monitor repeats the horizontal line drawing cycle for each visible line. The number of visible lines is the first number of the second set. Once all of the visible lines have been drawn the monitor waits for a moment (Again called Front Porch Time), a sync pulse is generated to move the beam back to the upper left corner of the display, waits another moment (Back Porch Time) and starts the horizontal line drawing process all over again.

This time the numbers represent horizontal scan line times as opposed to pixel times.



Timing for the Vertical values : "1024x768" 65 1024 1080 1224 1344 **768 776 778 808**

Tuning and Testing the Image:

What does all of the above mean to you? Well, if your video image is off center or does not fill the screen properly these numbers can be altered to 'tune' the image.

By manipulating the porch times, sync times and frame length (the last three numbers of each set) you can center and resize the image. This is made easier by our menu system.

The Tuning and Testing Process

Lets go back to the screen menu:

You should be at the prompt:

```
So what do you want to do? ([v]iew, [s]ave, [t]une or [e]xit)?
```

Select the [t] (Tune) option.

The system breaks up your temporary Xconfig file (the one you just specified) into several parts. It then runs the server to glean certain information from the hardware.

The ModeDB entries are parsed by the program and a list of available modes is presented in a selection list. Below is an example from my system:

```
These are your available modes:

        1 "1024x768"
        2 "1024x768i"
        3 "800x600"
        4 "640x480"

Enter the number of the resolution mode to fiddle with:
```

We'll select the first resolution ("1024x768") as the one we want to test first, so we enter '1'. This process should be run for each resolution mode you intend to use. After some munching, the program gives us a work screen that looks like this:

```
 Mode Info: "1024x768" - Clock=65MHz, H sync=48kHz, V refresh=60Hz
  HORIZONTAL VALUES                    |        VERTICAL VALUES
 Resolution Sync    Sync    Frame      |  Resolution Sync    Sync    Frame
            Start   End     Length     |             Start   End     Length
-----------------------------------+-----------------------------------
1024      1080    1224    1344          768       776    778      808
 Porch Vals    56      144      120                    8      2      30
Image Tuning ----- Action ------ Units --+-------- Action ------ Units -
L)  Move Image Left         (by n*4)    | U)  Move Image Up         (by n*1)
R)  Move Image Right        (by n*4)    | D)  Move Image Down       (by n*1)
W)  Widen Image             (by n*8)    | H)  Heighten Image        (by n*2)
N)  Narrower Image          (by n*8)    | S)  Shorten Image         (by n*2)
Scan Line Tuning - Action ---- Units --+------ Action ------------ Units
1)  increment Sync Start    (by n*4)    | a)  increment Sync Start (by n*1)
2)  decrement Sync Start    (by n*4)    | b)  decrement Sync Start (by n*1)
3)  increment Sync End      (by n*4)    | c)  increment Sync End   (by n*1)
4)  decrement Sync End      (by n*4)    | d)  decrement Sync End   (by n*1)
5) increment Frame Length (by n*8)      | e)increment Frame Length (by n*1)
6) decrement Frame Length (by n*8)      | f)decrement Frame Length (by n*1)
            To add hsync or vsync polarization enter [P]
[n]ew mode  [h]elp  [q]uit  [t]est  [s]ave&exit  [r]eset to orig.values
                    What next? >
```

By selecting [t] first, we can see what the image looks like as is. Always do this first, since you can't fix the image if you can't see what's wrong with it.

After entering [t] the screen clears and the server is brought up with a test pattern. The test pattern stays up for 5 seconds then brings you back to the work screen.

The top half of the options menu is fairly self explanatory. These options are what I call macro tuning or Image tuning actions whereas the bottom half of the options menu is for discrete scan line tuning. The 'units' column might not be clear, however. The mode line numbers in the Horizontal section must be altered by certain increments or decrements. So for instance if you want to move the image to the right (by entering [R]) the system will prompt you for 'how many units'. If you enter '2' in response to the prompt the screen image will be shifted right by (2*4 pixels) 8 pixels. If you want to widen the image (by entering [W]) and you enter '2' in response to the units prompt the image will be narrowed by (2*8 pixel) 16 pixels. So the '(by n*number)' column in the work screens tells you the granularity of your tuning capabilities.

As you alter these values and move the image around test (using the [t] option) early and often, to see the effects of your work. If you get things way out of whack, enter [u] to reset the timing lines back to their original values. It may be interesting to watch the numbers in the mode line change with each action (the sixth and seventh lines of the display).

There are two sets of numbers to watch: the mode line and the 'Porch Vals' line. the mode line has already been discussed in the previous section, the Porch Vals are the 'front porch', 'sync pulse width' and 'back porch' respectively.

The bottom half of the option menu requires some knowledge of the effects of discrete changes to the timing values. The upper half does these same things in 'sets'. In other words, selecting an action from the top half of the menu causes one or more of the options in the bottom half to be executed. If you want to see exactly what these 'Scan Line Tuning' options do, see the section on discrete scan line tuning. This section also discusses exactly what happens to the numbers as the 'Image Tuning' options are executed.

If you know the sync polarization of the video mode you are working with you can set it by entering [P]. You will be prompted to enter the horizontal and or vertical sync polarization. Some example entries would be:

```
+hsync  +vsync
-hsync
+vsync
-hsync  +vsync
```

Note that if you specify both you must separate the two with a space.

Once you have the display for this mode line looking the way you want it, select [n] to work on the next mode. You will be returned to the mode selection screen explained above.

Saving the Xconfig File:

When all of the modes are tuned and tested select [s] to save the changes you've made. This rebuilds the temporary working *Xconfig.test* file in the */tmp* directory. You will be returned to the

```
So what do you want to do? ([v]iew, [s]ave, [t]une or [e]xit)?
```

prompt and you will need to select the [s] option to save the *Xconfig* file to some place useful. At this point you may want to select the [v] option before you save the Xconfig file. This option pages you through the file you've just created. You cannot edit it at this time, however you may want to make notes as to the changes you may wish to make after the file is saved. When you return from viewing the file, select the [s] option and the program will prompt you for a location to save it to - the menu looks like this:

```
Where would you like to save the NEW Xconfig file:

1)      (Personal Xconfig)      $HOME/Xconfig

2)      (System Xconfig)        /etc/Xconfig

3)      (Host Xconfig)          $XWINHOME/lib/X11/Xconfig.hostname

4)      (DEFAULT)               $XWINHOME/lib/X11/Xconfig

5)      Leave it where it is (/tmp/Xconfig.test)
  ------------------------------------------------
          Enter Your Selection:
```

If your not sure you want to commit it without editing it first for changes, select option 5. This will leave the file in */tmp/Xconfig.test* and you may do with it as you see fit. For most normal installations (single user work stations), saving it to either '$*HOME/Xconfig*' (option 1) or '*/usr/X/lib/X11/Xconfig*' (option 4) will work fine. To render a default user *Xconfig* file that everyone on the system gets (if they don't have one in their home directory) select option 2. For a host specific version select option 3.

In any case once the destination option has been selected the newly specified server is activated by copying the appropriate server to the filename '$*XWINHOME/bin/X*' and the *Xconfig* file is copied to the user selected destination and the program terminates.

The **ConfigXF86** configuration program may be run as often as you like, whenever you like, just make sure you have a backup of your working *Xconfig* file or be sure you select option 5 in the save menu to leave the new *Xconfig*.test file in */tmp.*

Discrete Scan Line Tuning:

(Shamelessly lifted from the $XWINHOME/lib/X11/etc/VideoModes.doc - read it - it's groovy)

To see what goes on inside of the **Image Tuning** options, and to get a feel of what to do with the **Scan Line Tuning** options, I have 'borrowed' the text by Eric S. Raymond upon which this program is based.

Remember the ModeDB line is made up of two sets of numbers, horizontal timings and vertical timings. Also remember that the number sets represent the following values:

Visible_Resolution, Sync_Pulse_Start, Sync_Pulse_End, and Frame_Length

You **move** the image by changing the sync pulse timing. You **scale** it by changing the frame length (you need to move the sync pulse to keep it in the same relative position, otherwise scaling will move the image as well). Here are some more specific recipes:

The horizontal and vertical positions are independent. That is, moving the image horizontally doesn't affect placement vertically, or vice-versa. However, the same is not quite true of scaling. While changing the horizontal size does nothing to the vertical size or vice versa, the total change in both may be limited. In particular, if your image is too large in both dimensions you will probably have to go to a higher dot clock to fix it. Since this raises the usable resolution, it is seldom a problem!

The image is displaced to the left or right

To fix this, move the horizontal sync pulse. That is, increment or decrement (by a multiple of 8) the middle two numbers of the horizontal timing section that define the leading and trailing edge of the horizontal sync pulse.

If the image is shifted left (right border too large, you want to move the image to the right) decrement the numbers. If the image is shifted right (left border too large, you want it to move left) increment the sync pulse.

The image is displaced up or down

To fix this, move the vertical sync pulse. That is, increment or decrement the middle two numbers of the vertical timing section that define the leading and trailing edge of the vertical sync pulse.

If the image is shifted up (lower border too large, you want to move the image down) decrement the numbers. If the image is shifted down (top border too large, you want it to move up) increment the numbers.

The image is too wide (too narrow) horizontally

To fix this, increase (decrease) the horizontal frame length. That is, change the fourth number in the first timing section. To avoid moving the image, also move the sync pulse (second and third numbers) half as far, to keep it in the same relative position.

The image is too deep (too shallow) vertically

To fix this, decrease (increase) the vertical frame length. That is, change the fourth number in the second timing section. To avoid moving the image, also move the sync pulse (second and third numbers) half as far, to keep it in the same relative position.

Any distortion that can't be handled by combining these techniques is probably evidence of something more basically wrong, like a calculation mistake or a faster dot clock than the monitor can handle.

Finally, remember that increasing either frame length will decrease your refresh rate, and vice-versa.

Registering New Monitor Descriptions:

If the monitor you are using was NOT in the database and you have successfully built a well tuned monitor specification, you may want to submit your 'ModeDB' section to the XFree86 group for inclusion into the Monitor database. Here are the ground rules for such a submission:

1 The minimum identification information acceptable includes:
Monitor name
Your name
Your Email address (if you have one)
Date of Creation

2 The modeDB section should include at least three modes:
640x480 @ 25MHz
800x600 @ either 36MHZ or 40MHz
and either (both may be submitted)
1024x768i (interlaced) @ 44MHz
OR
1024x768 @ either 65MHz or 75MHz

Other modes may also be submitted but at least three of the above four modes MUST be included for the submission to be included into the database. Modes other than these might not be included into the database.

Send your submissions via Email to:
stz@netcom.com

The format for the submission is shown below:

```
# Monitor Name:         Acme Model 17
# Contributed By:       David Wexelblat <dwex@goblin.org >
# Last Change:          03-03-94
ModeDB
"1024x768" 65 1024 1080 1224 1344 768 776 778 808
"1024x768i" 44 1024 1040 1216 1264 768 768 778 819 Interlace
"800x600" 40 800 840 968 1056 600 601 605 628 +hsync
"640x480" 25 640 664 760 800 480 491 493 525
```

Using MakeCard:

MakeCard is a program for generating new records for the Video card database (*Cards.dat*). This is an abbreviated description of how to use it to automagically generate a data record for your hardware. To start the program, first make sure you are NOT in the X environment. You MUST execute this program from a character screen. Now do the following:

```
cd $XWINHOME/lib/X11/etc/ConfigXF86

./MakeCard
```

After the screen clears you will see the prompt:

```
Want me to probe your hardware first? (y/n):
```

Answer 'y' to this prompt so that the program can scan your hardware and fill in much of the data that the record requires.

A warning will appear and five seconds later SuperProbe will be run.

It is possible that your system may appear to hang, this happens on some systems. If this happens wait for 4 or 5 minutes then reboot.

The data should have been captured regardless of whether your system hung.

Start the program again and again answer 'y' to the above prompt. This time the program will notice that we're in a recovery state and will grab the previously scanned data without re-running the SuperProbe program.

Now the program will determine which X server is appropriate for your hardware and will inform you as to its selection:

```
We'll be using the XF86_Mach8 Server...
```

The screen clears again and a work screen is presented, my hardware produces the following work screen. Your hardware will provide different data:

```
             Graphics Card Description    CardID [0]
------------------------------------------------------------------------
      CardName    [NoCard]
      CardDesc    [0]
      Contributor [0]  LastEdit [03-29-94]

ChipSet   [Mach8] ClockChip [0]     Server [XF86_Mach8] (ie XF86_SVGA)
ClockProg [0]     RamDac    [0]     Driver [accel] (ie vga256)

ClockRates [43.00 48.80 0.00 36.00 50.30 56.70 0.00 44.90 30.20 32.00
37.50 39.00 40.00 89.00 75.50 65.00 21.50 24.40 0.00 18.00 25.15 28.35
0.00 22.45 15.10 16.00 18.75 19.50 20.00 44.50 37.75 32.50 ]

Vesa Fallbacks - At least ONE MUST be defined!
Vmode64  [25]     Vmode86   [40]    Vmode107i [0]
Vmode107 [65]     Vmode121i [0]     Vmode121  [0]

Options  [0]
VideoRam [1024]   BiosBase [0]      ViewPort [0]
Virtual  [0]      SpeedUp  [0]
------------------------------------------------------------------------
Changes? (Format: Fieldname=Value) [f]ile [t]est [s]ave [q]uit [e]xample
```

Note that many of the fields are already filled in for you (shown here in bold face) . However you MUST fill in certain fields to make this record useful. The following data is REQUIRED:

CardName	(A text name for your video card, this name is what will show up in the ConfigXF86 video card menu)
Contributor	(Your name and email address (if you have one) goes here)
Server	(should be filled in automatically)
Driver	(should be filled in automatically)
ClockRates	(should be filled in automatically)

Most of the other fields that are important will be filled in automatically.

At this point you should only need to enter the CardName and Contributor fields. To do this, enter the data in the following format (My name and hardware are used as examples):

```
CardName=ATI Graphics Ultra

Contributor=Steven T. Zwaska
```

Your screen should now look something like this:

```
                Graphics Card Description      CardID [0]
------------------------------------------------------------------------
        CardName    [ATI Graphics Ultra]
        CardDesc    [0]
        Contributor [Steven T. Zwaska]  LastEdit [03-29-94]

ChipSet   [Mach8]     ClockChip [0]   Server [XF86_Mach8] (ie XF86_SVGA)
ClockProg [0]   RamDac    [0]     Driver [accel] (ie vga256)

ClockRates [43.00 48.80 0.00 36.00 50.30 56.70 0.00 44.90 30.20 32.00
37.50 39.00 40.00 89.00 75.50 65.00 21.50 24.40 0.00 18.00 25.15 28.35
0.00 22.45 15.10 16.00 18.75 19.50 20.00 44.50 37.75 32.50 ]

Vesa Fallbacks - At least ONE MUST be defined!
Vmode64  [25]   Vmode86   [40]  Vmode107i [0]
Vmode107 [65]   Vmode121i [0]   Vmode121  [0]

Options  [0]
VideoRam [1024] BiosBase [0]    ViewPort [0]
Virtual  [0]    SpeedUp  [0]
------------------------------------------------------------------------
Changes? (Format: Fieldname=Value) [f]ile [t]est [s]ave [q]uit [e]xample
```

If you don't have any other changes to make (in most cases you won't) enter 's' to save the record. When you see the prompt:

```
Press [Enter] to save to
/home3/Xinstall/ConfigXF86/DATAFILES/<cardname>.CRD...
```

Press [Enter]
Then enter 'q' to quit

Your record will have been added to the database and a single record file named after your card will have been written to the *DATAFILES* subdirectory.

Note: for those with SVR3.x operating systems, your OS will probably have a problem with the long filenames generated by MakeCard - normally it will simply truncate the name to 14 characters.

Now re-run **ConfigXF86** and your card should appear at the bottom of the list.

If the configuration works well, mail the file it generated in the *DATAFILES* subdirectory to **stz@netcom.com** for inclusion into the next release of the database. This is very important since, if you don't mail the new file, the next release of the database will overwrite your database without your video card.