Information for using/developing external clock setting programs

The XFree86 Project, Inc. 16 December 1994

Contents

1 Using an external clock setting program

XFree86 provides a facility for setting the clock frequency on a graphics card by an external program. This is provided to make it possible to deal with cards that use clock selection methods not supported by the standard drivers.

This facility is enabled by adding a `ClockProg` line to the `Device` section of the `XF86Config` file. The format of this line is:

`ClockProg "`*commandpath*`"`

where *commandpath* is the full pathname of the clock setting program. No flags are allowed in *commandpath*.

When using this option, a Clocks line is required in the Device section of the `XF86Config` file to tell the server which clock frequencies are available to it. In the case of a card that has a fixed set of preset clocks, the ordering of the clocks in the `Clocks` line should correspond to what the card/program expects. In the case of a card which has a fully programmable clock (like the SS24) the ordering is not important, and the values chosen can be anything in the range supported by the card. Up to 32 clock values may be specified.

The server calls the external program when it needs to change the clock frequency. This occurs at startup and when switching modes with the hot-key sequences. The command is passed two command-line arguments. The first is the clock frequency in MHz (as a floating point number – currently specified to the nearest 0.1 MHz). The second argument is the index of the clock value in the Clocks list (the first clock is index 0). Cards with a fixed set of clocks would probably make use of the index, while cards with a fully programmable clock would use the frequency argument.

2 Developing an extern clock setting program

When such an external program is being used, the server does not change any register fields related to clock selection, and the external program must be careful to only modify clock selection fields. The program is run with stdin and stdout set to xf86Info.consoleFd – which is the fd to use for display-related ioctl() operations if required. Stderr is the same as the server's stderr – so error or warning messages should be directed there. The program is run with the uid set to the real user's ID – so if it needs to use privileged system calls it should be suid-root. The program does not inherit any I/O access privileges, so it will need to do whatever is required to enable them.

The program is expected to return an exit status 0 when successful, and a status in the range 1-254 when it fails. If the external program fails during the server initialisation stage, the server exits. If it fails for a mode switch, the mode switch is aborted (the server assumes that the clock frequency hasn't been changed) and the server keeps running. If necessary an exit status may be specified in the future for which the server would exit if the program fails for a mode switch.

A sample clock switching program is provided for use with many ET4000 cards that have a fixed set of 8 clocks. This program is only intended as a sample, and it is not intended for general use (the internal server code handles this type of card). The program is `xc/programs/Xserver/hw/xfree86/etc/et4000clock.c` in the source tree, and `/usr/X11R6/lib/X11/etc/et4000clock.c` in the binary tree.

The idea of using an external clock program was suggested by Frank Klemm <*pfk@rz.uni-jena.de*>

`$XConsortium: clkprog.sgml,v 1.1 95/01/06 20:29:55 kaleb Exp $`
`$XFree86: xc/programs/Xserver/hw/xfree86/doc/sgml/clkprog.sgml,v 3.2 1995/01/28 16:03:06 dawes Exp $`