

HowTo по настройке видеосистемы в Xfree86.

Eric S. Raymond, esr@thyrsus.com

Русский перевод Ilgiz Kalmeteв, ilgiz@mail.rb.ru

v3.1, 31 October 1997

Данный документ описывает определение параметров видеокарты/монитора в конфигурационном файле Xfree86. Дистрибуция Xfree86 имеет сейчас хорошие средства настройки для большинства стандартных случаев. Этот документ особенно полезен в случае, если вы хотите настроить видеорежим "под себя" для высококачественного монитора или необычного оборудования. Кроме того, здесь вы можете найти помощь в использовании программы xvidtune для "подгонки" стандартного режима под ваш монитор.

Содержание

1	Заявление	1
2	Предисловие	2
3	Как работает монитор	2
4	Полоса пропускания монитора	5
5	На что это влияет:	6
6	Интерпретация основных терминов	6
7	???? при конфигурации вашей системы	9
8	Требования к памяти	10
9	Расчет размер кадра	10
10	Черная магия и синхроимпульсы	11
11	Подводя итог	13
12	Предельные параметры вашего монитора.	14
13	Использование черезстрочных режимов	15
14	Вопросы о ответы	16
15	Разрешение проблем с изображением.	17
16	Графическое изображение характеристик монитора.	18
17	Благодарности.	21

1 Заявление

Материал представленный ниже может быть использован на ваш собственный страх и риск. Существует возможность нанести вред себе и испортить монитор, если выйти за границы его заводских настроек. Для более подробной информации читайте главу "Предельная настройка".

Любые последствия, связанные с ущербом нанесенный вам или вашему монитору касаются только вас.

Свежую версию этого документа можно взять на странице, посвященной проекту по документированию Линукса - <http://sunsite.unc.edu/LDP>.

Замечания, критику и предложения по улучшению этого How-to, пожалуйста, направляйте по адресу esg@snark.thyrsus.com. И, пожалуйста, не присылайте мне письма с просьбами решить за вас волшебным образом ту или иную проблему, связанную с вашим монитором, этим вы только отнимите у меня время и расстроите себя - все, что мне известно по теме, я уже представил ниже.

2 Предисловие

Сервер XFree86 позволяет пользователям конфигурировать видеосистему, позволяя добиваться лучшего использования установленного оборудования. Это руководство поможет вам узнать, как получать необходимые параметры для оптимального использования видеокарты и монитора.

Мы представим метод получения рабочей конфигурации и затем покажем, как можно экспериментировать опираясь на эту базовую установку и получить параметры максимально приближенные к вашим вкусам.

Начиная с версии 3.2, XFree86 снабжается программой XF86Setup(1), которая позволяет интерактивно настраивать режимы, не путаясь цифрах. Поэтому, в большинстве случаев, вам не надо рассчитывать основные видеорежимы вручную. К сожалению, XFree86(1) имеет некоторые ограничения; она знает только о стандартных режимах до 1280x1024. Если у вас в наличии высококачественный монитор способный поддерживать разрешение 1600x1200 и более, вам скорее всего придется рассчитать основные параметры самим.

Последние версии XFree86 имеют в своем составе инструмент под названием xvidtune(1), который вы, скорее всего, найдете полезным для проверки и настройки режимов монитора. Запуск этой программы начинается с печального предупреждения о возможных последствиях связанных с ошибками при ее использовании. Внимательное чтение этого документа поможет вам узнать, что означают эти цифры, которыми оперирует программа и позволит пользоваться xvidtune более эффективно и уверенно.

Если у вас запущен какой-нибудь из режимов и даже как-то работает (в особенности, если один из по умолчанию установленных VESA режимов дает устойчивую картинку, но изображение смещено вправо или влево, или оно слишком маленькое или, наоборот, слишком большое), можете сразу переходить к разделу "Разрешение проблем с изображением". Этом раздел научит вас подправлять параметры синхронизации для получения более тонких настроек.

Если у вас есть программа xvidtune(1), вы сможете тестировать новые режимы "на лету" не изменяя конфигурационные файлы X или даже не перегружая X сервер. Кроме того, XFree86 позволяет вам назначать горячие клавиши в Xconfig для переключения между различными режимами (подробнее см. XFree86.man). Используйте эту возможность - это очень удобно. Если вы хотите протестировать новый режим, назначьте ему уникальное название и разместите его в конце вашего списка горячих клавиш. Оставьте проверенный режим по умолчанию для того, чтобы можно было откатиться назад в случае, если с новым режимом возникнут проблемы.

3 Как работает монитор

Для того, чтобы определить правильные параметры в Xconfig, важно понимать, как работает монитор. Эти параметры используются X сервером на нижнем уровне управления дисплеем. Картинка, формируемая на дисплее состоит из линеек точек. Эти линейки формируют массив точек, образуя изображение. Точки испускают свет, когда на них попадает электронный луч внутри кинескопа. Чтобы луч находился подсвечивал каждую точку одинаковое количество времени, луч перемещается внутри кинескопа единообразно.

Луч начинает свое движение с верхней левой части экрана, перемещаясь по прямой слева направо и ненадолго задерживается в правой части экрана. Затем, луч движется влево, но на одну строку

ниже. Процесс повторяется как и в предыдущем случае до тех пор, пока не будет достигнута нижняя часть экрана. Затем, луч перемещается в левый верхний угол и все начинается сначала. Есть небольшое отступление от описанной выше схемы - это чересстрочная развертка (interlacing). При этом один полукадр заполняется каждая вторая строчка, а остальные строки заполняются во время следующего полукадра.

Начальная положение луча в верхнем левом углу называется началом кадра. Кадр завершается в той же точке после того, как луч пройдя левый нижний угол и вернется назад. Кадр состоит из всех линий, которые формирует луч проходя от верхней части дисплея до нижней.

Если электронный луч будет включен все то время, что он движется в течении кадра то, все точки дисплея будут светиться. В этом случае, вокруг видимого изображения не будет черной границы. На границах изображение будет искажено т.к. здесь лучом сложно управлять. Чтобы уменьшить искажения точки по краю дисплея не светятся даже если луч проходит по ним. Таким образом видимая зона уменьшается.

Важно понимать, что происходит в то время, когда луч не рисует картинку. Время, которое электронный луч затрачивал бы для подсвечивания граничных точек, используется для перемещения его от правого края дисплея обратно и смещения на одну линию вниз. Время, которое электронный луч затрачивал бы на подсвечивания верхних и нижних граничных точек, используется для перемещения его из нижнего правого угла дисплея в верхний левый угол.

Видеокарта генерирует сигналы, заставляющие монитор включать электронный луч над каждой точкой экрана - таким образом формируется изображение. Карта, кроме того, определяет в какой момент времени луч начинает двигаться от правой стороны дисплея в левую и вниз на одну линию, генерируя сигнал называемый горизонтальным синхроимпульсом (этот процесс называется синхронизацией). В конце каждой линии появляется один горизонтальный синхроимпульс. Видеокарта генерирует также один вертикальный синхроимпульс, который заставляет луч перемещаться в левый верхний угол дисплея. Вертикальный синхроимпульс генерируется в конце каждого кадра.

Дисплей требует небольшой задержки перед и после горизонтального и вертикального синхроимпульсов для того, чтобы луч мог стабилизироваться. Если луч не успеет стабилизироваться, изображение не будет стабильным. Мы еще вернемся к этому ниже, определив эти основные положения более точно на основе формул и примеров для лучшего понимания.

§ 4. Основные положения, которые нужно знать о вашем мониторе и видеокарте.

Для того, чтобы вносить изменения в Xconfig, вам надо знать следующее:

- горизонтальную и вертикальную частоту вашего монитора.
- частоту видеокарты (dot clock)
- полосу пропускания монитора

Частоты синхронизации монитора:

Частота горизонтальных синхроимпульсов - это просто число горизонтальных линий, которое монитор способен генерировать за секунду. Это наиболее важный параметр вашего монитора. Частота вертикальных синхроимпульсов - число вертикальных перемещений луча за секунду времени.

Частоты синхронизации обычно перечислены в разделе спецификации руководства на ваш монитор. Вертикальная синхронизация обычно обозначается в герцах (Гц) - циклов в секунду, горизонтальная частота в килогерцах - 1000 циклов секунду (КГц). Частота вертикальной синхронизации обычно лежит в пределах 50-150 Гц, горизонтальной - в пределах 31-135 КГц.

Если вы обладаете мультисканирующим монитором, то для этих частот будут указаны пределы. Некоторые мониторы, особенно из дешевых, имеют линейку фиксированных частот. Они тоже могут быть настроены, но возможности будут сильно ограничены в пределах характеристик вашего монитора. Для высокого разрешения выбирайте наибольшие частоты синхронизации. И будьте осторожны - выбрав для такого монитора большую частоту развертки, чем та на которую он рассчитан, можно легко его испортить.

Для того, чтобы добиться лучших результатов ранние версии этого руководства были довольно бесцеремонны относительно выбора предельных значений развертки для мультисканирующих мониторов. В этой связи, у нас появились с тех пор новые соображения, касающиеся осторожного обращения с оборудованием. Эту тему мы раскроем ниже в разделе "Предельные параметры вашего монитора".

Частота видеокарты:

В руководстве на вашу видеокарту обычно описывается ее частота (dot clock) т.е. количество точек в секунду, которое карта способна нарисовать на экране. Если у вас нет этой информации, то X сервер вам ее предоставит. Даже если X сервер заблокирует монитор, он покажет на терминале линейку частот и другую информацию. Если вы перенаправите эту информацию в файл, надо ее сохранить даже, если вам придется перезагрузиться, чтобы снова получить доступ к консоли. (Все последние версии X серверов поддерживают опцию `-probeonly`, которые высвечивают эту информацию и заканчивают работу не загружая X-ы и, даже, не изменяя видеорежима).

Когда начинает загружаться X сервер сообщения выглядят примерно так:

Если вы используете 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
       ---
           Точное обозначение чипсета; прототип для 86C911

(--) S3: chipset driver: s3_generic
(--) S3: videoram: 1024k
       -----
           Объем страничной памяти RAM

(**) 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
       -----
                   Возможные частоты в МГц

(--) S3: Maximum allowed dot-clock: 110MHz
       -----
                   Полоса пропускания

(**) 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
```

Если вы используете SGCS или X/Inside X:

```
WGA: 86C911 (mem: 1024k clocks: 25 28 40 3 50 77 36 45 0 0 79 31 94 65 75 71)
-----
|      |      |      |      |      |      |      |      |      |
                   Допустимые частоты в МГц
```

```

|          |          +-- Объем памяти
|          +-- Тип чипа
+-- Тип сервера

```

Примечание: выполняйте эту операцию при минимальной загрузке машины (насколько это возможно). Поскольку X сервер это приложение, его циклы измерения могут быть неточными из-за дисковой активности. Поэтому проделайте это несколько раз и добейтесь одинаковых результатов. Если не получается, начинайте убивать процессы пока не добьетесь желаемого. Пользователям SVr4: процесс mousemgr наверняка вам все будет портить.

Чтобы избежать неточностей при измерении, вам следует взять значения, полученные при тестировании, и поместить их в раздел Clocks конфигурационного файла Xconfig - это подавляет измерительный цикл и дает X-ам точный список значений частот, которые могут тестироваться X-ми. Используем значения из приведенного выше примера:

```

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

```

В системах с большой нагрузкой этот метод позволит избежать таинственных проблем связанных с невозможностью запуска X сервера. X сервер может взять неправильные значения из-за большой нагрузки системы, а затем он будет не способен найти подходящую частоту карты или найдет неправильную частоту!

4 Полоса пропускания монитора

Если вы пользуетесь XFree86, ваш сервер протестирует видеокарту и сообщит вам наибольшую возможную частоту на которую она рассчитана.

В любом случае, наибольшая доступная частота примерно равна полосе пропускания монитора. Хотя здесь нужно уточнить, что некоторые мониторы способны обеспечивать на 30% большую полосу пропускания по сравнению с номинальной. Опасность с которой здесь можно столкнуться это превысить вертикальную частоту синхронизации; более подробно бы обсудим это ниже.

Знание полосы пропускания позволяет вам более тонко выбирать настройки. Она может повлиять на качество изображение (особенно точность передачи деталей).

Полоса пропускания вашего монитора должна быть указана в спецификации руководства для монитора. Если это не так сверьтесь с наибольшей разрешающей способностью монитора. Как правило, для грубой оценки полосы пропускания (а следовательно и верхней границы частоты видеокарты, которую вы можете использовать) исходя из разрешения можно использовать следующую таблицу:

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

ВТW, никакой тайны в этой таблице нет; эти числа просто нижняя граничная частота для каждого разрешения в стандартной базе видеорежимов XFree86 (за исключением последней строчки - это значение я экстраполировал). На самом деле, полоса пропускания монитора может быть выше минимально требуемой для максимального разрешения вашего монитора, поэтому не бойтесь установить частоту карты на несколько МГц выше.

Обратите внимание, также, что полоса пропускания редко бывает проблемой для частот приблизительно до 65 МГц. Для SVGA карты и большинства мониторов с высоким разрешением вы можете взять любое значение в пределах полосы пропускания монитора. Например:

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

Даже в низкокачественных мониторах не сильно ограничена полоса пропускания по сравнению с оцененной для максимального разрешения. Хорошим примером является NEC Multisync II - он даже не держит 800x600 согласно его спецификации. Он может выдать только 800x560. Для такого низкого разрешения вам не надо высокую частоту видеокарты или большую полосу пропускания; скорее всего лучше что выможете себе позволить это частота 32 МГц или 36 МГц, эти два значения все равно не слишком отличаются от заявленной полосы пропускания в 30 МГц.

Для этих двух значений изображение не будет таким четким, каким оно могло бы быть, но все же терпимое. Конечно, было бы лучше если бы NEC Multisync II имел бы полосу выше, чем, скажем, 36 МГц. Но это не выжно для обычных применений как , скажем, редактирование текстов, поскольку разница не слишком значительна, чтобы вызвать большие видеоискажения (ваши глаза сразу бы заметили это, если бы это было так).

5 На что это влияет:

Пределы частоты синхронизации монитора вместе с частотой видеокарты определяют максимальное разрешение, которое вы можете использовать. Но только драйвер ответственен за использование максимального потенциала железа. Хорошее оборудование без соответствующего ему драйвера пустая трата денег. С другой стороны, вряд ли вы много добьетесь с хорошим драйвером для плохого оборудования. Это философия разработки XFree86.

6 Интерпретация основных терминов

В этом разделе объясняется, что означают все те, термины о которых мы рассказывали и некоторые другие вещи, которые вам следует знать. Но, прежде некоторые определения. Далее описываются переменные (в скобках), которые мы будем использовать в наших вычислениях.

частота горизонтальной развертки (HSF)

число горизонтальных сканирований в секунду (см. выше)

частота вертикальной развертки (VSF)

число вертикальных сканирований в секунду (см. выше). Наиболее важный параметр т.к. является верхней границей рифреша.

частота видеокарты (dot clock - DCF)

частота чипа (VCO) на видекарте - максимальное количество генерируемых им точек в секунду.

полоса пропускания (VB)

максимальная частота, которую вы можете подать на видеовход вашего монитора и при этом увидеть что-либо осмысленное на экране. If your adaptor produces an alternating on/off pattern (здесь я не знаю, как перевести, прим. переводчика), его нижняя частота составляет половину DCF, поэтому, в теории полоса, пропускания имеет смысл начиная с DCF/2. Но для нормального отображения деталей вы вряд ли захотите чтобы она была значительно ниже максимальной DCF, лучше - выше.

размеры кадра (HFL, VFL)

горизонтальная длина кадра (HFL) - число тиков необходимых для прохождения лучом всех точек одной линии включая неактивные точки по правой и левой границе. Вертикальная длина кадра - число линий необходимых для всего изображения включая неактивные границы кадра.

частота обновления изображения или рефреш (RR)

Число раз обновления изображения в секунду (этот параметр еще называют частотой кадров). Большая частота лучше т.к. уменьшает мерцание экрана. 60 Гц - хорошо, но стандарт VESA в 72 Гц еще лучше. Вычисляется этот параметр так:

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

Обратите внимание, что значение в делителе это не тоже самое что видимое разрешение монитора (обычно разрешение монитора меньше). Мы рассмотрим это подробнее ниже.

Частоты для которых обычно определяются черезстрочные режимы (interlaced modes), как, например черезстрочный режим 87 Гц на самом деле являются частотами для полукадров: частота мерцания кажется почти такой же как для обычного режима, но каждая строка обновляется в два раза реже.

Для расчетных целей мы будем считать для режима с черезстрочной разверткой полную частоту обновления кадров т.е. 43,5 Гц. Качество изображения с черезстрочным режимом лучше, чем с построчным при одной и той же частоте полных кадров, но определенно хуже, чем построчный режим при полукадровой частоте.

6.1 О полосе пропускания

Производители мониторов любят афишировать высокую полосу пропускания монитора потому, что от нее зависит контрастность и цветовой диапазон. Более высокая полоса означает видимость более мелких деталей.

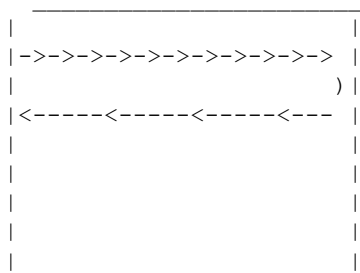
Для изображения картинки ваш монитор использует электрические сигналы. Такие сигналы, будучи преобразованы из цифровой формы в аналоговую всегда представляю собой волновой процесс. Они могут рассматриваться как комбинация более простых волн каждая из которых имеет определенную частоту. Многие из этих простых волн имеют частоту в мегагерцовом диапазоне т.е. 20МГц, 40МГц или даже 70МГц. Полоса пропускания вашего монитора - это наибольшая частота аналогового сигнала, который может обработан монитором без искажения.

Для наших целей полоса пропускания главным образом важна как точка ограничивающая наибольшую возможную частоту видекарты.

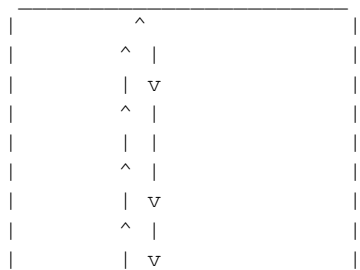
6.2 Частоты синхронизации и частота обновления :

Каждая горизонтальная линия на дисплее - это просто видимая часть горизонтальной длины кадра. На самом деле, в каждый момент времени на экране активна только одна точка, но при достаточно большой частоте обновления и благодаря инерционности ваших глаз, вы можете наблюдать цельное изображение.

Для пояснения несколько рисунков:



Горизонтальная частота синхронизации - число, перемещений электронного луча в секунду как это показано на рисунке.



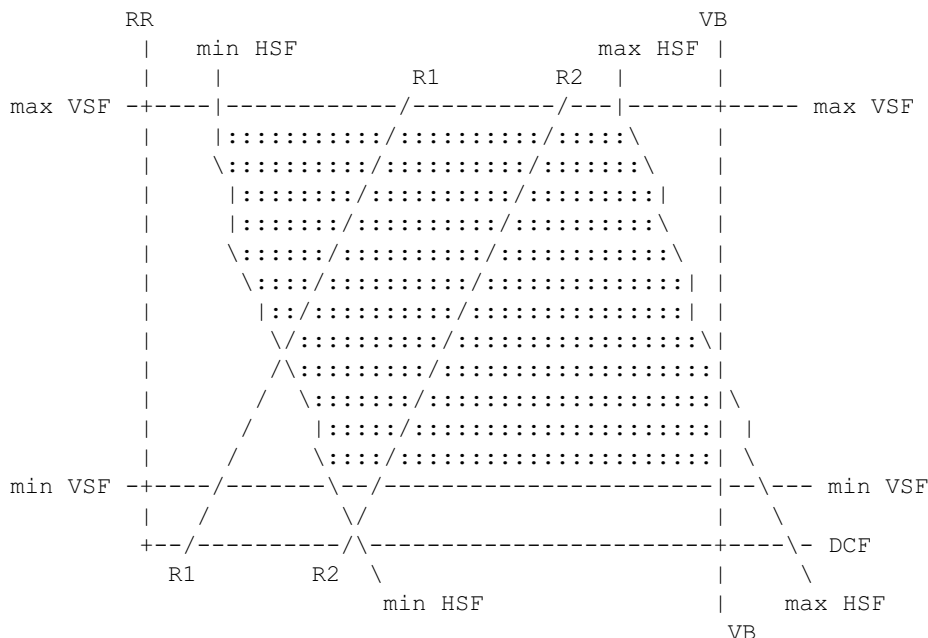
Вертикальная частота синхронизации - число, перемещений электронного луча в секунду, как это показано на рисунке.

Помните, что на самом деле плотность зигзагов при формировании растра гораздо выше; т.е. луч перемещается влево-вправо и одновременно вверх-вниз.

Теперь мы можем понять, как связаны частота видеокарты и размер кадра с частотой обновления. По определению, один Герц (Гц) - это один цикл в секунду. Таким образом, если горизонтальная длина кадра это HFL, а вертикальная длина кадров это VFL, то для того чтобы покрыть весь экран нам понадобится $(HFL * VFL)$ тиков. Поскольку ваша карта генерирует DCF тиков в секунду по определению то, очевидно, что частота, с которой луч может перемещаться слева направо и обратно, снизу вверх и обратно равна $DCF / (HFL * VFL)$. Это и есть частота обновления экрана т.е. сколько раз в секунду изображение обновляется.

????? Для того, чтобы установить хорошее разрешение при приемлимой частоте мерцания экрана вам необходимо понимать этот принцип.

Тем из вас, кто лучше понимают объяснения в графическом представлении поможет следующий рисунок:



Эта диаграмма для условного монитора. Ось x представляет частоту видеокарты (DCF), ось y - частоту обновления (RR). Закрашенная область представляет возможности монитора: каждая точка внутри заштрихованной области один из возможных видеорежимов. Линии помеченные R1 и R2 представляют определенные разрешения (такие как 640x480); они служат для иллюстрации того, как могут различные комбинации частоты карты и частоты обновления представлять одно и тоже разрешение. Линия R2 означает более высокое разрешение, чем R1.

Верхняя и нижняя границы очерченной области просто горизонтальные линии, определяющие граничные значения для вертикальной частоты синхронизации. Полоса пропускания это верхняя граница частоты видеокарты и, следовательно, представлена правой вертикальной линией заштрихованной области.

В разделе "Диаграмма возможностей монитора" вы найдете программу, которая поможет нарисовать похожую диаграмму (только гораздо красивее, в графическом режиме) конкретно для вашего монитора. Кроме того, в этом разделе обсуждаются такие интересные вещи как зависимость границ в зависимости от пределов горизонтальной частоты синхронизации.

7 ??? при конфигурации вашей системы

Рассмотрим формулу, которую мы получили, иначе:

$$DCF = RR * HFL * VFL$$

т.е. случай, когда мы имеем неизменную частоту видеокарты. Вы можете ориентироваться на этот параметр для выбора частоты обновления, горизонтального разрешения или вертикального разрешения. Если один из этих параметров увеличивается, два или один из оставшихся уменьшаются.

Тем не менее, обратите внимание, что частота обновления не может быть больше максимальной частоты вертикальной синхронизации вашего монитора. Таким образом, для каждого данного монитора при данной частоте адаптера, существует минимальное значение произведения длин кадра ниже которого вы не сможете опуститься.

Выбирая значения, помните: если RR слишком мала, изображение будет мерцать. Скорее всего вы не захотите сделать частоту обновления ниже 60 Гц. Это частота мерцания люминофора; если вы плохо воспринимаете изображение при такой частоте, попробуйте 72 Гц, эргономический стандарт VESA.

Мерцание очень утомительно, хотя глаза могут хорошо приспособляться и эта способность различна для разных людей. Если ваш монитор расположен таким образом, чтобы обеспечивать угол зрения в 90 градусов, фон используется темный, символы контрастны, а монитор настроен на среднюю яркость вы можете чувствовать себя комфортно и при 45 Гц.

Хорошим тестом может быть следующий: запустите xterm с белым фоном и черными символами используя команду `xterm -bg white -fg black`, увеличьте размер окна на весь экран. Теперь увеличьте яркость монитора до 3/4 от максимальной, отведите взгляд в сторону от монитора. Теперь попробуйте взглянуть на изображение боковым зрением (таким образом, задействовав более чувствительное периферическое зрение). Если вы не заметите мерцания или почувствуете, что мерцание терпимо, значит частота обновления нормальна для вас. В противном случае, лучше установить более высокую частоту, потому, что невидимая прямым зрением мерцание будет сильно утомлять глаза и вызовет головную боль, даже если не первый взгляд изображение выглядит нормальным.

Для чересстрочных режимов уровень мерцания зависит от большего числа факторов таких как установленное вертикальное разрешение и характер изображения. Так что, поэкспериментируйте. Тем не менее, мы все же не хотим опускаться значительно ниже полукадровой частоты в 85 Гц. Итак, предположим мы выбрали минимально приемлемую частоту обновления. Выбирая значение для HFL и VFL нам нужно запас для маневра.

8 Требования к памяти

Доступный объем памяти может ограничить выбираемое разрешение для цветных или черно-белых мониторов. Очевидно, что это не является ограничением для дисплеев, которые имеют только два цвета - черный и белый т.е. не позволяют отображать оттенки черного.

Для дисплея способных отображать 256 цветов, на каждую видимую точку необходимо по одному байту. Этот байт определяет в какой пропорции генерируются красный, зеленый и синий цвета для соответствующей точки. Чтобы получить требуемое количество памяти, умножьте количество видимых точек по горизонтали на количество видимых линий по вертикали. Для дисплея с разрешением 800x600 это составит $800 \times 600 = 480\,000$ - количество видимых точек на экране. При одной точке на байт, это число определяет количество необходимой памяти на видеокarte.

Таким образом, округленно, требования к памяти составят $(HR * VR) / 1024$ Кб. Если у вас больше памяти, чем минимально необходимо, излишек памяти будет использован виртуальными экранами. Тем не менее, если у вас установлено 512 Кб, вы не сможете использовать это разрешение. Даже если у вас хороший монитор, без достаточного количества памяти, вы не сможете использовать потенциальные возможности монитора. С другой стороны, если у вас 1 Мб, но ваш монитор способен только на 800x600, вы все равно не сможете использовать высокое разрешение (возможное разрешение этой проблемы обсуждается в разделе "Использование черезстрочных режимов")

Не беспокойтесь, если у вас больше памяти, чем требуется; XFree86 использует ее для прокрутки изображения (см. документацию к Xconfig по размерам виртуальных экранов). Помните, также, что карта с 512Кб реально имеет не 512 000 байт, а $512 \times 1024 = 524,288$ байт.

Если вы запускаете SGCS X (теперь называется X/Inside) и используете карту S3 и при этом желаете иметь 16 цветов (4 бита на пиксел), вы можете установить глубину цвета 4 пиксела в Xconfig, таким образом удвоив разрешение, на которое видеокarta рассчитана. Например, карты S3 поддерживают 1024x768x256. Вы можете вытянуть из них 1280x1024x16 при глубине цвета 4 бита.

9 Расчет размер кадра

Внимание: этот метод работает для мультисканирующих мониторов. Возможно, он будет работать для мониторов с фиксированными частотами, но без гарантий!

Начните с того, что разделите DCF на максимально доступную HSF, чтобы получить горизонтальную длину кадра.

Например, предположим, что у вас Sigma Legend SVGA с частотой 65 МГц и монитор у вас с вертикальной частотой синхронизации 55КГц. Тогда, результат (DCF / HSF) будет равен 1181 ($65\text{МГц} = 65000\text{КГц}$; $65000 / 55 = 1181$).

А теперь немножко черной магии. Вам нужно округлить это число до ближайшего, делимого на 8. Это связано с контролером VGA, используемом SVGA и S3 картами; они используют 8-ми битные регистры со сдвигом на три бита влево, что реально составляет 11-ти битное значение. Другие типы видеокарт такие как ATI 8514/A могут в этом не нуждаться, но наверняка мы этого не знаем и коррекция не помешает. Поэтому, округлите округлите горизонтальную длину кадра до 1176.

Это число (DCF / HSF) округленное до делимого на 8) минимальная HFL, которую вы можете использовать. Вы можете получить большие значения HFL (и, таким образом, большее количество точек по горизонтали), уменьшая HSF. Но, за это вы заплатите меньшей частотой обновления и, следовательно, большим мерцанием.

Как правило, 80% процентов горизонтальной длины кадра можно использовать для горизонтального разрешения - видимой части горизонтальной части раstra (остальная часть составляет границы и обратное время сканирования - т.е. время необходимое лучу переместиться от правого края экрана до левого края следующей линии раstra). В нашем примере, это составляет 944 тиков.

Теперь, чтобы получить нормальное соотношение сторон дисплея т.е. 4:3, установите вертикальное разрешение на 3/4 от вертикального разрешения, которое вы только что вычислили. Для нашего примера, это 708 тиков. Чтобы получить реальную VFL умножьте это число на 1,05 и получите 743 тика.

С техничкой точки зрения, в соотношении 4:3 нет ничего волшебного; ничто не мешает вам использовать не Золотое отношение, а какое-нибудь другое, если вам так удобнее. Получить высоту и ширину кадра можно исходя из длины диагонали, просто умножьте длину диагонали на 0,8 для ширины, и на 0,6 для высоты.

Итак, $HFL = 1176$ и $VFL = 743$. Разделив 65МГц на произведение этих двух значений мы получим замечательную 74,4 Гц частоту обновления экрана. Здорово! И лучше чем стандарт VESA. И при этом вы получили 944x708, больше, чем ожидаемое 800x600. Совсем не плохо!

Вы можете даже улучшить частоту рефреша до почти 76 Гц, имея в виду тот факт, что мониторы способны часто обеспечивать примерно на 2 КГц большую частоту по сравнению с расчетной и уменьшая VFL (т.е. взяв 75% от 944). Но прежде чем экспериментировать с предельными значениями убедитесь, что монитор способен обеспечивать вертикальную синхронизацию до 76%. (популярный монитор NEC 4D, например, не может. Он способен только на 75Гц VSF). (Более подробно это обсуждается "Предельные настройки вашего монитора").

До сих пор, большинство наших рассуждений и расчетов относилось к растровым дисплеям. И они мало напоминают черную магию!

10 Черная магия и синхроимпульсы

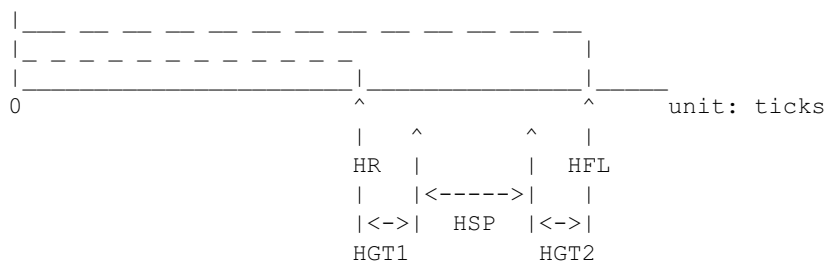
Ну, хорошо, теперь, когда вы рассчитали значения HFL/VFL для выбранной частоты карты, нашли приемлимую частоту рефреша и проверили достаточно ли для этого памяти, начинается настоящая магия - вам нужно знать когда и где разместить значения синхронизации.

Синхроимпульсы управляют вертикальными и горизонтальными развертками монитора. Значения HSF и VSF, выуженные из таблицы руководства на монитор, это номинальные значения, аппроксимирующие максимальные синхронизирующие частоты. Именно синхроимпульс исходящий от видеокарты определяет максимальную частоту.

Помните те картинки, которое мы рассматривали ? Только часть времени при формировании растра используется для изображения картинки (т.е. вашего разрешения).

10.1 Горизонтальная синхронизация:

Пользуясь предыдущим, определением, чтобы сгенерировать одну горизонтальную линию необходимо HFL тиков. Назовем видимое количество тиков (горизонтальное разрешение) HR. Тогда, очевидно, что $HR < HFL$ по определению. Для конкретности, допустим, что оба начинаются в один и тот же момент времени, как показано ниже:



Теперь, мы желали бы разместить синхроимпульс HSP как показано выше т.е. между последним тиком для данных изображения и последним тиком для целого кадра. Но почему? Потому, что если мы сможем этого достигнуть, тогда изображение не будет сдвинуто вправо или влево. Оно будет находиться там, где ему и положено быть, размещаясь в видимой области экрана.

Более того, нам нужен запас около 30 тиков с обеих сторон синхроимпульса. Это расстояния HGT1 и HGT2. В обычной конфигурации $HGT1 \neq HGT2$, но если вы настраиваете конфигурацию "с нуля", вы захотите начать эксперименты принимая их равными (т.е. синхроимпульс выровнен по центру). Признаком того, что синхроимпульс расположен неправильно является смещение изображения на экране, при этом с одной стороны экрана наблюдается широкая полоса, а с

другой стороны изображение завернуто относительно края, что создает эффект белой полосы или полосы с двойным изображением. Неправильно установленный вертикальный синхроимпульс может вызвать заворачивание изображения совсем так же как у телевизора при неправильно установленном вертикальном размере (собственно, принцип тот же).

Если вам повезло, то ширина синхроимпульса отражена в спецификации руководства на ваш монитор. А если нет, то вот тут то и начинается настоящая магия...

Вам придется в этом случае провести несколько испытаний. Но для большинства случаев, мы можем допустить, что синхроимпульс длиной от 3,5 до 4,0 микросекунд.

Опять же для конкретности, положим HSP равным 3,8 микросекунды (что является, BTW, неплохим значением для начала).

Используя уже известное значение 65МГц, мы знаем, что HSP эквивалентно 247 тикам ($= 65 * 10^{**6} * 3.8 * 10^{-6}$) [вспоминаем, что $M = 10^6$, $micro = 10^{-6}$].

Некоторые производители любят указывать горизонтальную частоту вместо ширины в точках. Вы можете встретиться со следующими терминами:

active time (HAT)

Связано с HR, но в миллисекундах. $HAT * DCF = HR$.

blanking time (HBT) (HBT)

Связано с (HFL - HR), но в миллисекундах. $HBT * DCF = (HFL - HR)$.

front porch (HFP)

Это просто HGT1.

sync time

Это просто HSP.

back porch (HBP)

Это просто HGT2.

10.2 Вертикальная синхронизация:

Возвращаясь к нашей картинке, как нам поместить 247 тиков так это показано на рисунке?

Используя наш пример, имеем, $HR = 944$ и $HFL = 1176$. Разница между этими двумя значениями $1176 - 944 = 232 < 247!$ Очевидно, что-то здесь нужно подкорректировать. Но что же?

Первым делом надо поднять 1176 до 1184 или опустить 944 до 936. Теперь разница составляет $1184 - 936 = 248$. Ну, что ж, уже ближе...

Затем вместо 3,8, мы используем 3,5 для вычисления HSP; теперь имеем $65 * 3.5 = 227$. Выглядит лучше. Но 248 не сильно отличается от 227. Обычно хватает приблизительно 30 тиков между HR и началом SP и тоже значение между концом SP и HFL. И они должны делиться на 8. Неужели все?

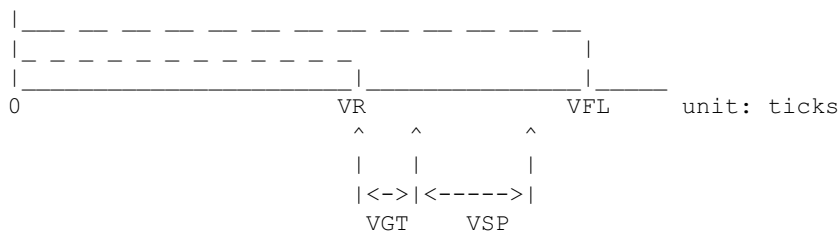
Нет. Теперь давайте сделаем следующее, $936 \% 8 = 0$, $(936 + 32) \% 8 = 0$ тоже. Но $936 + 32 = 968$, $968 + 227 = 1195$, $1195 + 32 = 1227$. Хм, неплохо. Но не делится на 8, поэтому округляем до 1232.

Но теперь, возможно, у нас появилась проблема, синхроимпульс больше не расположен прямо посередине между h и H. К счастью, взяв калькулятор, мы находим, что $1232 - 32 = 1200$ тоже делится на 8 и $(1232 - 32) - 968 = 232$ еще приемлемо, соответствуя синхроимпульсу длительностью 3,57 микросекунд.

Кроме того, $936/1232 = 0.76$ или 76%, не сильно отличается от 80%, так что все должно быть в порядке. Более того, при использовании текущей длины кадра, мы устанавливаем для монитора частоту синхронизации в 52.7КГц ($= 65\text{МГц}/1232$), что находится в пределах способностей нашего монитора. Без проблем.

Имея в виду упомянутые выше правила, $936 * 75\% = 702$, Это наше новое разрешение по вертикали. А $702 * 1.05 = 737$ - наше новая вертикальная длина кадра.

Частота обновления экрана = $65\text{МГц}/(737 \cdot 1232) = 71.6\text{Гц}$. Пока все хорошо.
 Диаграмма вертикального синхроимпульса аналогична:



Синхроимпульс начинается сразу же после данных. VGT - запас по времени для синхроимпульса. Большинство мониторов довольствуются $VGT=0$ (нет запаса) и этот случай мы будем использовать в нашем примере. Некоторым мониторам требуется 2 или 3 тика и ничего страшного, если придется их добавить.

Возвращаясь к нашему примеру: исходя из определения длины кадра, вертикальный тик это время необходимое для прохождения полного горизонтального кадра. Таким образом, в нашем примере он равен $1232/65\text{Mhz} = 18.95\text{us}$.

По опыту, вертикальный синхроимпульс находится в пределах от 50 до 300 мкс. Давайте, для примера возьмем 150 мкс, что составляет 8 вертикальных импульсов ($150\text{us}/18.95\text{us} \approx 8$).

Некоторые производители вместо количества точек любят приводить кадровые характеристики в виде временных параметров. Вы можете встретиться со следующими терминами:

active time (VAT)

Связанно с VR, но в миллисекундах. $VAT \cdot VSF = VR$.

blanking time (VBT)

Связанно с $(VFL - VR)$, но в миллисекундах. $VBT \cdot VSF = (VFL - VR)$.

front porch (VFP)

Это просто VGT.

sync time

Это просто VSP.

back porch (VBP)

Это временной "запас" после вертикального синхроимпульса. Часто равен 0.

11 Подводя итог

Файл Xconfig содержит таблицу видеорежимов в виде строчек цифр, каждая из которых является полным определением для одного из режимов X сервера. Поля сгруппированы в 4 секции, раздел названия, раздел частоты стробирования, разделы вертикальных и горизонтальных параметров.

Раздел названий содержит одно поле - название видеорежима, который определяется. Это название используется в поле "Modes" в секции Graphics Driver Setup в файле Xconfig. Название поля может быть опущено, если имя текущего поля такое же как у предыдущего.

Раздел частоты частоты стробирования содержит только этот параметр (то, что мы называли DCF) видеорежима. Значение этого поля определяет какая частота требуется для генерации чисел в следующих разделах.

Раздел горизонтальных параметров состоит из четырех полей, которые определяют как должны генерироваться горизонтальные линии. Число в первом поле показывает число точек в каждой линии составляющих картинку (то, что мы назвали HR). Второе поле, определяет точку для

генерации горизонтального синхроимпульса. Третье поле, определяет точку в которой закончиться горизонтальный синхроимпульс. Четвертое поле, определяет общую длину кадра (HFL).

Раздел вертикальных параметров также состоит из четырех полей. Первое поле содержит число видимых линий (VR). Второе поле определяет номер линии для генерации вертикального синхроимпульса. Третье поле определяет номер линии в которой вертикальный синхроимпульс закончится. Четвертое поле содержит число, определяющее общую высоты кадра (VFL).

Пример:

#Название	частота	горизонталь				вертикаль			
"752x564"	40	752	784	944	1088	564	567	569	611
	44.5	752	792	976	1240	564	567	570	600

(Примечание: стандарт X11R5 не поддерживает дробные частоты)

Для Xconfig все вышеприведенные числа - количество подсвеченных точки в линии, число точек между подсвеченной областью и началом синхроимпульса, число точек представляющих продолжительность импульса, число точек после окончания синхроимпульса - составляет общее количество точек на линию. Количество горизонтальных точек должно делится на 8 без остатка.

Пример для горизонтали: 800 864 1024 1088

В этом примере количество подсвеченных точек - 800, за ним следует число точек, когда начинается синхроимпульс - 864, синхроимпульс заканчивается на 1024 точке, последняя точка в линейке - 1088. Еще раз обратите внимание, что все числа (800 864 1024 и 1088) делятся на 8! Для вертикальных значений это не требуется.

Общее количество линий на дисплее формирует кадр. Главной единицей измерения для кадра является линия. Часть линий содержит картинку. После того как последняя видимая линия будет прорисована, произойдет задержка в несколько линий, перед началом вертикального синхроимпульса. Этот импульс продолжается несколько линий, после чего следует задержка в виде оставшихся до конца кадра линий. Числа описывающие этот режим выглядят примерно так:

Пример для вертикали: 600 603 609 630

Исходя из этого прмера на нашем дисплее 600 видимых линий, вертикальный синхроимпульс начинается на 603-м линии и заканчивается на 609, всего используется 630 линий.

Обратите внимание, что для вертикальных параметров нет необходимости быть кратными 8!

Давайте вернемся к нашему примеру. Теперь все, что нам нужно это подставить полученные значения в переменные Xconfig:

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

где SH1 стартовый тик горизонтального синхроимпульса, а SH2 конечный тик; аналогично SV1 стартовый тик иертикального синхроимпульса, а SH2 конечный тик.

#название	частота	горизонталь				вертикаль				флаг
936x702	65	936	968	1200	1232	702	702	710	737	

Указывать флаг не надо, ведь это построчный режим. Ну, теперь то мы закончили.

12 Предельные параметры вашего монитора.

Вам определенно не следует превышать параметры развертки вашего монитора, если монитор с фиксированными частотами. Вы можете сжечь его!

Для мультисканирующих мониторов существуют более тонкие потенциальные проблемы, которых вам следует остерегаться.

В тоже самое время, использование более высокой частоты видеокарты, чем полоса пропускания монитора для мультисканирующих мониторов достаточно безвредна. (замечание: теоретические пределы достигаются, когда частота стробирования превысит полосу пропускания в два раза).

Это прямое приложение теоремы Найквиста: рассматривайте пиксели как пространственно распределенные последовательности единиц стробирующих сигналов и вы поймете, почему.) (? :) прим. переводчика)

А вот превысить максимальную частоту синхронизации довольно сложно. Некоторые мониторы могут иметь защиту на схемном уровне, которая блокирует монитор при опасных синхронизирующих частотах, но не надейтесь на нее. Особенно, если это старые мультисканирующие мониторы (типа Multisync II), которые используют только один преобразователь для строчной развертки. Такие мониторы недостаточно защищены для таких случаев. Даже если в вашем мониторе есть схема управления высоким напряжением (которая может отсутствовать в мониторах в фиксированными развертками), она может не покрыть все возможные частоты, особенно в дешевых моделях. Это не только может увеличить нагрузку на электронную схему, но увеличить скорость старения люминофора и вызвать неожиданные излучения (в том числе и рентгеновское).

Еще один момент касающийся полосы пропускания это то, что входной импеданс рассчитанный только для определенного диапазона, при более высокой частоте может вызвать отражения изображения, и радиопомехи.

Тем не менее, наиболее важный момент в этом вопросе крутизна фронтов видеосигнала, которая обычно не зависит от реальной частоты видекарты, а (если производителя видекарты заботят такие проблемы) связана с максимальной частотой карты.

Так что будьте осторожны...

13 Использование черезстрочных режимов

(Существованием этого раздела мы обязаны главным образом David Kastrup, dak@pool.informatik.rwth-aachen.de)

При фиксированной частоте черезстрочный режим гораздо менее подвержен мерцанию, чем построчный, при условии, что схемы вертикальной развертки монитора способны поддерживать этот режим стабильным. Именно поэтому, черезстрочный режим был придуман первым. Черезстрочные режимы заслужили свою дурную репутацию потому, что они хуже своих построчных собратьев при той же вертикальной частоте развертки (VSF, которую обычно указывают в спецификациях). Но они определенно совершеннее при одной и той же частоте горизонтальной развертки, именно там, где лежат основные ограничения монитора/карты.

При фиксированной частоте (или при половинной кадровой частоте или VSF) черезстрочный режим более подвержен мерцанию: 90Гц черезстрочный дисплей будет хуже 90Гц построчного. Ему, тем не менее, потребуется только половина полосы пропускания и половина частоты горизонтальной развертки. Если сравнить его с построчным режимом при одинаковой частоте стробирования и одинаковой развертке, он будет гораздо лучше: 45Гц при построчной развертке непереносимо. С 90Гц и черезстрочной разверткой я работал годами на мониторе Multisync 3D (разрешение 1024x768). Думаю, вам бы понадобилось по крайней мере 70 Гц при построчной развертке, чтобы чувствовать себя также комфортно.

Тем не менее, вам надо иметь в виду два момента: используйте черезстрочные режимы только при высоких разрешениях, чтобы линии в различных полукадрах были ближе друг к другу. Можно было бы поэкспериментировать с шириной синхроимпульса и его положением для более стабильного изображения. Если чередующиеся строки белые и черные, вот тут то и проявятся недостатки черезстрочного режима. У меня есть одна программа, которая использует в качестве фона такой рисунок (XСерт, к счастью, единственная, насколько я знаю). Мне приходится переключаться в 800x600, чтобы не болели глаза.

По той же самой причине используйте шрифты по крайней мере 100dpi, или другие шрифты изображения горизонталей в которых занимает не менее 2 линий (других причин не использовать черезстрочный режим для высоких разрешений, нет).

И конечно, никогда не используйте черезстрочный режим, если есть соответствующий построчный режим для вашей видекарты при той же частоте рефреша. Если все-таки вы обнаружите, что при некоторых разрешениях вы ограничите с предельными параметрами монитора или карты,

что проявляется в мерцании или смазывании (превышена полоса пропускания) изображения, попробуйте использовать черезстрочный режим. Это, конечно, бесполезно, если VSF монитора близка к пределу.

Настроить черезстрочный режим просто: действуйте так же как в случаях с построчными режимами. Только надо принимать во внимание два дополнительных соображения: вам понадобится нечетное общее количество линий (последнее число в строке, описывающей видеорежим) и когда вы определяете флаг "interlaced" реальная кадровая частота удваивается. Ваш монитор должен поддерживать 90Гц кадровой частоты, если режим который вы определили выглядит как 45Гц не смотря на флаг "interlaced".

Для примера вот моя конфигурация для черезстрочного 1024x768: мой монитор Multisync 3D поддерживает до 90Гц кадровой частоты и 38КГц строчной.

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

Обе границы на пределе в этом режиме. Определение того же режима, но без флага "interlace", также находится на пределе возможностей строчной развертки (и строго говоря, за пределами нижней частоты вертикальной развертки), что вызывает невыносимое мерцание экрана.

Основное правило: если вы определили режим с вертикальной разверткой, меньшей в два раза, чем возможная для монитора, сделайте количество линий нечетным и добавьте флаг "interlace". В большинстве случаев качество изображения должно значительно улучшиться.

С другой стороны, если при построчной развертке монитор работает на пределе своих возможностей т.е. когда частота вертикальной развертки на 30% или более ниже максимальной для монитора, установка черезстрочного режима (возможно, с более высоким разрешением) может обеспечить более высокий результат, но не гарантирую.

14 Вопросы о ответах

Q. В примере, который вы привели использованы нестандартные размеры экрана, могу ли я их использовать?

А. Почему нет? Нет никаких оснований вынуждающих вас использовать 640x480, 800x600, или даже 1024x768. Сервера XFree86 предоставляют большую свободу в конфигурировании оборудования. Обычно чтобы найти подходящую конфигурацию надо несколько попыток. Самое главное найти высокую частоту обновления при разумном размере видимой области, а не высокое разрешение за счет утомительного мерцания!

Q. Единственное ли это разрешение при частоте карты 65МГц и HSF=55КГц?

А. Конечно нет! Следуя обычным указаниям, вы вольны испытать несколько конфигураций, выбрав на свой вкус. Эксперименты могут доставить много удовольствия. Большинство попыток закончится лишь мусором на экране, практически современному мультисканирующему монитору сложно нанести вред. Только удостоверьтесь, что ваш монитор может поддерживать частоты развертки выбранного режима длительное время.

Помните о мониторах с фиксированной частотой! Такие эксперименты могут разрушить их очень быстро. Убедитесь в использовании допустимых для них частотах перед каждым экспериментом.

Q. Вы только что упомянули два стандартных разрешения. В Xconfig есть много стандартных разрешений.

А. Несомненно! Возьмите, например, "стандартный" 640x480 в Xconfig. Он использует 25МГц, размеры кадра 800 и 525 => частота обновления 59,5 Гц. Не плохо. Но 28 МГц - широко распространенная частота для многих SVGA карт. Если мы хотим использовать ее для 640x480, то следуя указаниям, которые мы только что обсудили мы получили бы размеры кадра примерно 802 и 505. Теперь частота обновления поднята до 68Гц, значительное улучшение по сравнению со стандартной конфигурацией.

Q. Можете ли вы выделить основные моменты того, что мы обсудили?

А. Вкратце:

1. Для любой определенной частоты, платой за повышение разрешения является снижение частоты обновления т.е. повышенное мерцание.
2. Если более высокое разрешение предпочтительно и ваш монитор поддерживает его, постарайтесь достать SVGA карту, обеспечивающую соответствующую частоту (DCF). Чем выше, тем лучше!

15 Разрешение проблем с изображением.

Итак, вы получили параметры для конфигурации X. Поместите их файл Xconfig и обозначьте как тестовый режим. Вы загрузили X, переключились в новый режим... и с изображением что-то не так. Что делать? Вот список распространенных видео искажений и как их исправить.

(С исправлением небольших искажений хорошо справляется xvidtune(1))

Переместить изображение помогут параметры синхроимпульса. Изменить размер можно изменяя длины кадра (вам надо переместить синхроимпульс, чтобы изображение оставалось в той же самой относительном положении, иначе вы также переместите и изображение). Вот более конкретные рекомендации:

Горизонтальные и вертикальные положения независимы. Т.е. перемещение изображение в горизонтальном направлении никак не повлияет на его вертикальное положение, и наоборот. Тем не менее, это не совсем так при изменении размера. Тогда как изменение размера по горизонтали не влияет на вертикальный размер или наоборот, изменение размера в обоих направлениях может быть ограничено. В особенности, когда изображение слишком велико в обоих измерениях, вам наверняка придется перейти на более высокую частоту карты. Но, это редко является проблемой, поскольку увеличение частоты стробирования увеличивает возможное разрешение.

15.1 Изображение смещено влево или вправо

Чтобы это исправить, переместите горизонтальный синхроимпульс. Т.е. увеличьте или уменьшите (оставляя кратными 8) средние 2 числа в секции горизонтальных параметров определяя тем самым начальный и конечный край горизонтального синхроимпульса.

Если изображение смещено влево (правая граница слишком широкая, и вы хотите сместить изображение вправо) уменьшите эти значения. Если изображение смещено вправо (левая граница слишком широкая, и вы хотите сместить ее влево) увеличьте значения.

15.2 Изображение смещено вверх или вниз

Чтобы это исправить, переместите вертикальный синхроимпульс. Т.е. увеличьте или уменьшите средние числа вертикальных параметров определяя тем самым начальный и конечный край вертикального синхроимпульса.

Если изображение смещено вверх (нижняя граница слишком велика, и вы хотите передвинуть изображение вниз) уменьшите числа. Если изображение смещено вниз (верхняя граница велика, и вы хотите передвинуть его выше), увеличьте значения.

15.3 Размер изображения слишком велик и по горизонтали и по вертикали

Переключитесь на более высокую частоту стробирования. Если у вас несколько определенных режимов, возможно, по ошибке вы выбрали режим с меньшей частотой.

15.4 Изображение слишком широкое (слишком узкое)

Чтобы это исправить, увеличьте (уменьшите) горизонтальную длину кадра. Т.е. измените 4-ое число в первой секции. Чтобы изображение не двигалось, измените также синхроимпульс (второе и третье числа) наполовину, чтобы сохранить относительное положение.

15.5 Изображение слишком растянуто (сужено) по вертикали

Чтобы это исправить увеличьте (уменьшите) вертикальную длину кадра. Т.е. измените 4-ую цифру во втором разделе. Чтобы изображение не смещалось, переместите также синхоримпульс (вторая и третья цифры) наполовину для сохранения относительного положения.

Любые искажения, которые не могут быть исправлены с помощью этих мер, говорят о более серьезных ошибках, наподобие ошибки в расчетах или же установка частоты большей, чем способен выдержать монитор.

И наконец, помните, увеличение размера кадра уменьшает частоту обновления, и наоборот.

16 Графическое изображение характеристик монитора.

Для того, чтобы построить диаграмму режимов монитора, вам понадобится gnuplot package (свободный графический язык для Unix систем) и программа под названием tool modeplot, шеловый скрипт для gnuplot для изображения диаграммы характеристик монитора, водимых в командной строке.

Вот копия modeplot:

```
#!/bin/sh
#
# modeplot -- генерация диаграммы возможных режимов монитора
#
# Запустите `modeplot -?' чтобы увидеть возможные опции программы.
#
# ($Id: XFree86-HOWTO.txt,v 1.1 1999/03/27 07:54:21 ott Exp $)

# Monitor description. Bandwidth in MHz, horizontal frequencies in kHz
# and vertical frequencies in Hz.
TITLE="Viewsonic 21PS"
BANDWIDTH=185
MINHSF=31
MAXHSF=85
MINVSF=50
MAXVSF=160
ASPECT="4/3"
vesa=72.5      # минимальная рекомендованная частота рефреша для 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
modeplot control switches:
```

```

-t "<description>"      name of monitor           defaults to "Viewsonic 21PS"
-b <nn>                 bandwidth in MHz          defaults to 185
-h <min> <max>          min & max HSF (kHz)      defaults to 31 85
-v <min> <max>          min & max VSF (Hz)      defaults to 50 160
-a <aspect ratio>      aspect ratio              defaults to 4/3
-g "<options>"          pass options to gnuplot

```

The -b, -h and -v options are required, -a, -t, -g optional. You can use -g to pass a device type to gnuplot so that (for example) modeplot's output can be redirected to a printer. See gnuplot(1) for details.

The modeplot tool was created by Eric S. Raymond <esr@thyrsus.com> based on analysis and scratch code by Martin Lottermoser <Martin.Lottermoser@mch.sni.de>

```

This is modeplot $Revision: 1.1 $
EOF

```

```

                exit;;
        esac
        shift
done

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

# Константы. Unfortunately, the plot is quite sensitive to changes in
# these, and they may fail to represent reality on some monitors. We need
# to fix values to get even an approximation of the mode diagram. These come
# from looking at lots of values in the ModeDB database.
F1 = 1.30      # multiplier to convert horizontal resolution to frame width
F2 = 1.05      # multiplier to convert vertical resolution to frame height

# Function definitions (multiplication by 1.0 forces real-number arithmetic)
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)

# Put labels on the axes
set xlabel 'DCF (MHz)'
set ylabel 'RR (Hz)' 6 # Put it right over the Y axis

# Generate diagram
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

```

Убедившись в наличии пакетов `modeplot` и `gnuplot`, вам понадобятся следующие характеристики монитора:

- полоса пропускания (VB)
- диапазон частот горизонтальной синхронизации (HSF)
- диапазон частот вертикальной синхронизации (VSF)

Программа изображения диаграмм делает некоторые допущения для упрощения, которые необязательно правильные. Вот почему результирующая диаграмма - это только грубое приближение. Вот эти допущения:

1. Все разрешения имеют только одно соотношение сторон экрана $AR = HR/VR$. Стандартные разрешения имеют $AR = 4/3$ или $AR = 5/4$. `Modeplot` принимает $4/3$ по умолчанию, но вы можете и изменить этот параметр.
2. Для рассматриваемых режимов, горизонтальная и вертикальная длины кадров жесткое произведение горизонтальных и вертикальных разрешений, соответственно:

$$\begin{aligned} HFL &= F1 * HR \\ VFL &= F2 * VR \end{aligned}$$

Для грубой прикидки возьмем $F1 = 1.30$ и $F2 = 1.05$ (см. "Расчет размера кадра").

Теперь возьмем частоту занимающую особое положение, HSF. Приведенные допущения лишь показывают, что каждое значение для DCF заранее определяет частоту обновления RR, т.е. для любого значения HSF существует функция $RR(DCF)$. Она может быть получена следующим образом:

Частота обновления равна частоте стробирования деленной на произведение длин кадра:

$$RR = DCF / (HFL * VFL) \quad (*)$$

С другой стороны, горизонтальная длина кадра равна частоте карты поделенной на частоту горизонтальной синхронизации:

$$HFL = DCF / HSF \quad (**)$$

Значение VFL может быть уменьшено до HFL с помощью вышеприведенных допущений:

$$\begin{aligned} VFL &= F2 * VR \\ &= F2 * (HR / AR) \\ &= (F2/F1) * HFL / AR \end{aligned} \quad (***)$$

Подставляя (**) и (***) в (*) мы получаем:

$$\begin{aligned} RR &= DCF / ((F2/F1) * HFL**2 / AR) \\ &= (F1/F2) * AR * DCF * (HSF/DCF)**2 \\ &= (F1/F2) * AR * HSF**2 / DCF \end{aligned}$$

При неизменных HSF, F1, F2 и AR, на нашей диаграмме это гипербола. Нарисовав две таких кривых для минимальной и максимальной горизонтальной частоты синхронизации, мы получили две оставшиеся границы допустимой области.

Прямые линии пересекающие допустимую область представляют особые разрешения. Это основано на (*) и следующем допущении:

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

Нарисовав эти линии для всех разрешении, которые нас интересуют, можно сразу же прочитать на диаграмме возможные соотношения между разрешением, частоты карты и частоты обновления допустимые для монитора. Обратите внимание, что эти линии не зависят от характеристик монитора, а зависят от второго допущения.

Программа modeplot предоставляет удобства для этого. Чтобы посмотреть возможные опции запустите modeplot -?. Типичная командная строка выглядит так:

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

Ключ -b определяет полосу пропускания; -v и -h устанавливают диапазоны для горизонтальной и вертикальной частот разверток.

Читая результаты работы modeplot, всегда имейте в виду, что это только приблизительные оценки. Например, она не рассматривает ограничения HFL, зависящей от минимально необходимой ширины синхроимпульса, она может быть точна лишь настолько, насколько точны допущения. Поэтому заменить точные вычисления (и шаманство) ничто не может что и описано в разделе "Подводя итог". Тем не менее, знание ее возможностей должно дать вам ощущение комфорта.

17 Благодарности.

Автор оригинального текста этого документа Chin Fang, fangchin@leland.stanford.edu .

Eric S. Raymond, esr@snark.thyrsus.com переработал и значительно переписал оригинал Chin Fang в попытке понять его. В процессе работы, он включил большУю часть совершенно другого how-to от Bob Crosson, crosson@cam.nist.gov .

Материал по чересстрочным режимам принадлежит главным образом David Kastrup, dak@pool.informatik.rwth-aachen.de .

Martin Lottermoser, Martin.Lottermoser@mch.sni.de предложил идею использовать gnuplot для построения диаграмм и сделал на ее основе математические выкладки. Распространяемый modeplot был переработан и обобщен ESR на основе оригинального кода от Martin.

Перевод на русский язык Плахотниченко Т.В., e-mail: taras@cartel.metrocom.ru fidonet: 2:5030/232.14, С.-Петербург, Сентябрь 1998г.