

Руководство по
MPlayer - Медиа Проигрыватель и Mencoder
(кроме приложений)

Оглавление

http://www.mplayerhq.hu.....	7
Как читать эту документацию.....	7
Глава 1. Введение.....	7
Глава 2. Установка.....	8
2.1. Требуемые программы.....	8
2.2. Возможности.....	8
2.3. Как насчёт GUI?.....	9
2.4. Шрифты и OSD.....	9
2.4.1. TrueType шрифты.....	9
2.4.2. Растровые шрифты.....	9
2.4.3. OSD меню.....	10
Примечание.....	10
2.5. Codec installation.....	10
2.5.1. Xvid.....	10
2.5.2. x264.....	10
2.5.3. AMR кодеки.....	10
2.6. RTC.....	10
Глава 3. Использование.....	11
3.1. Командная строка.....	11
3.2. Субтитры и OSD.....	11
3.3. Управление.....	12
3.3.1. Конфигурация управления.....	12
3.3.2. Управление через LIRC.....	13
3.3.3. Подчинённый ("рабский") режим.....	13
3.4. Сетевые потоки и каналы.....	13
Примечание.....	14
3.4.1. Сохранение потоковых данных.....	14
3.5. Приводы CD/DVD.....	14
3.5.1. Linux.....	14
3.5.2. FreeBSD.....	14
3.6. Воспроизведение DVD.....	14
Примечание.....	15
3.6.1. Региональный код.....	15
3.7. Воспроизведение VCD.....	15
Примечание.....	15
3.8. Редактируемые списки решений [Edit Decision Lists] (EDL).....	16
3.8.1. Использование EDL файлов.....	16
3.8.2. Создание EDL файлов.....	16
3.9. Расширенные возможности аудио.....	16
3.9.1. Окружающее/Многоканальное[Surround/Multichannel] воспроизведение.....	16
3.9.1.1. DVD'шники.....	16
3.9.1.2. Воспроизведение стерео звука на четырех колонках.....	17
3.9.1.3. Передача AC-3/DTS.....	17
3.9.1.4. Передача MPEG аудио.....	17
3.9.1.5. Matrix-кодированное[matrix-encoded] аудио.....	17
3.9.1.6. Эмуляция окружающего звука в наушниках.....	17
3.9.1.7. Решение проблем.....	18
3.9.2. Манипуляции с каналами.....	18
3.9.2.1. Общая информация.....	18
3.9.2.2. Воспроизведение моно на двух колонках.....	18
3.9.2.3. Копирование/перемещение каналов.....	19
Пример: один канал на две колонки.....	19
Пример: левый канал на две колонки (сокращение).....	19
Пример: дублирование передних каналов на задние.....	19
3.9.2.4. Микширование каналов.....	19
Пример: один канал на две колонки.....	20
Пример: левый канал на две колонки (сокращение).....	20
Пример: сведение 6-канального PCM.....	20
Пример: Воспроизведение звука 5.1 на больших колонках без сабвуфера.....	20

3.9.3. Программная подстройка звука.....	21
3.10. TV вход.....	21
3.10.1. Компиляция.....	21
3.10.2. Советы по использованию.....	21
3.10.3. Примеры.....	22
3.11. Телетекст.....	23
3.11.1. Замечания реализации.....	23
3.11.2. Использование телетекста.....	23
3.12. Радио.....	23
3.12.1. Радио вход.....	23
3.12.1.1. Компиляция.....	23
3.12.1.2. Советы по использованию.....	23
3.12.1.3. Примеры.....	23
Примечание.....	24
Глава 4. Устройства вывода видео.....	24
4.1. Настройка MTRR.....	24
4.2. Xv.....	24
Примечание.....	25
4.2.1. 3dfx карты.....	25
4.2.2. S3 карты.....	25
Примечание.....	25
4.2.3. nVidia карты.....	25
4.2.4. ATI карты.....	26
4.2.5. NeoMagic карты.....	26
4.2.6. Trident карты.....	26
4.2.7. Cyro/PowerVR карты.....	26
4.2.8. Карты Intel.....	26
4.3. DGA.....	26
Риск безопасности.....	27
4.4. SDL.....	28
4.5. SVGAlib.....	29
Примечание.....	29
Примечание.....	29
4.6. Вывод во фреймбуфер (FBdev).....	29
Примечание.....	30
4.7. Matrox фреймбуфер (mga_vid).....	30
4.8. Поддержка 3Dfx YUV.....	30
4.9. tdfx_vid.....	30
4.10. OpenGL вывод.....	31
4.11. AAlib - отображение в текстовом режиме.....	31
Примечание.....	31
4.12. libcasa - Цветная ASCII Art библиотека	32
4.13. VESA - вывод в VESA BIOS.....	32
Важно.....	33
4.14. X11.....	33
4.15. VIDIX.....	34
4.15.1. ATI карты.....	35
4.15.2. Matrox карты.....	35
4.15.3. Trident карты.....	35
4.15.4. 3DLabs карты.....	35
4.15.5. nVidia карты.....	35
4.15.6. SiS карты.....	35
4.16. DirectFB.....	35
4.17. DirectFB/Matrox (dfbmga).....	35
Примечание.....	36
4.18. MPEG декодеры.....	36
4.18.1. DVB ввод и вывод.....	36
4.18.2. DXR2.....	38
4.18.3. DXR3/Hollywood+.....	38
4.19. Другое оборудование вывода видео.....	39
4.19.1. Zr.....	39

4.19.2. Blinkenlights[Мерцающие огни?]	40
4.20. Поддержка TV-выхода	40
4.20.1. Matrox G400 карты	40
Важно	40
4.20.2. Matrox G450/G550 карты	41
4.20.3. ATI карты	41
4.20.4. nVidia	41
4.20.5. NeoMagic	42
Глава 5. Портинг	42
5.1. Linux	42
5.1.1. Debian пакеты	42
5.1.2. RPM пакеты	43
5.1.3. ARM	43
5.2. *BSD	43
5.2.1. FreeBSD	43
5.2.2. OpenBSD	43
5.2.3. Darwin	43
5.3. Коммерческие Unix	43
5.3.1. Solaris	43
5.3.2. HP-UX	45
5.3.3. AIX	45
5.3.4. QNX	46
5.4. Windows	46
Примечание	47
5.4.1. Cygwin	47
5.4.2. MinGW	47
5.5. Mac OS	47
5.5.1. MPlayer OS X GUI	48
Глава 6. Основы использования MEncoder	48
6.1. Выбор кодеков и формата файлов	48
6.2. Выбор входного файла или устройства	50
6.3. Двухпроходное кодирование MPEG-4 ("DivX")	50
6.4. Кодирование в Sony PSP видео формат	51
6.5. Кодирование в MPEG формат	51
Подсказка:	51
6.6. Масштабирование фильмов	51
6.7. копирование потока	52
6.8. Кодирование из нескольких входных файлов изображений (JPEG, PNG, TGA, SGI)	52
Примечание	52
6.9. Извлечение DVD субтитров в файл VOBsub	53
6.10. Сохранение пропорций	53
Глава 7. Кодирование с MEncoder	53
7.1. Создание высококачественного MPEG-4 ("DivX") рипа из DVD фильма	53
7.1.1. Подготовка к кодированию: Идентификация исходного материала и кадровой частоты	54
7.1.1.1. Определение кадровой частоты источника	54
7.1.1.2. Идентификация исходного материала	54
Подсказка:	55
7.1.2. Постоянный квантователь в сравнении с многопроходностью	56
Замечание:	56
7.1.3. Ограничения для эффективного кодирования	57
7.1.4. Усечение и масштабирование	58
7.1.5. Выбор разрешения и битпотока	59
7.1.5.1. Расчёт разрешения	60
7.1.6. Фильтрация	60
7.1.7. Чересстрочная развёртка и телесин	61
7.1.8. Кодирование чересстрочного видео	62
7.1.9. Замечания об аудио/видео синхронизации	62
7.1.10. Выбор видекодека	62
7.1.11. Аудио	63
7.1.12. Мультиплексирование	64

7.1.12.1. Улучшение мультиплексирования и надёжности A/V синхронизации.....	65
7.1.12.2. Ограничения контейнера AVI.....	65
7.1.12.3. Мультиплексирование в контейнер Matroska (Матрёшка).....	65
7.2. Как работать с телесинком и чересстрочной развёрткой на NTSC DVD.....	66
7.2.1. Введение.....	66
7.2.2. Как распознать тип Вашего видео.....	67
7.2.2.1. Построчная развёртка.....	67
7.2.2.2. Телесин.....	67
7.2.2.3. Чересстрочная развертка.....	67
7.2.2.4. Смешанные построчная развертка и телесин.....	68
7.2.2.5. Смешанные построчная и чересстрочная развертки.....	68
7.2.3. Как кодировать каждую категорию.....	68
7.2.3.1. Построчная развертка.....	68
7.2.3.2. Телесин.....	68
7.2.3.3. Чересстрочная развертка.....	68
7.2.3.4. Смешанные построчная развертка и телесин.....	69
7.2.3.5. Смешанные построчная и чересстрочная развертки.....	69
7.2.4. Примечания.....	70
7.3. Кодирование семейством кодеков libavcodec	71
7.3.1. Видео кодеки libavcodec.....	71
7.3.2. Аудио кодеки libavcodec.....	72
7.3.2.1. Дополнительная таблица PCM/ADPCM форматов.....	72
7.3.3. Опции кодирования libavcodec.....	73
7.3.4. Примеры настроек кодирования.....	75
7.3.5. Нестандартные inter/intra матрицы.....	75
7.3.6. Пример.....	76
7.4. Кодирование кодеком Xvid.....	77
7.4.1. Какие опции следует использовать для получения лучших результатов?.....	77
7.4.2. Опции кодирования Xvid.....	77
7.4.3. Профили кодирования.....	78
7.4.4. Примеры настроек кодирования.....	81
7.5. Кодирование кодеком x264.....	81
7.5.1. Опции кодирования x264.....	81
7.5.1.1. Введение.....	81
7.5.1.2. Опции, затрагивающие, в основном, скорость и качество.....	81
Замечание:.....	82
7.5.1.3. Опции, относящиеся к различным предпочтениям.....	84
7.5.2. Примеры настроек кодирования.....	85
7.6. Кодирование семейством кодеков Video For Windows	86
7.6.1. Поддерживаемые кодеки Video for Windows.....	86
7.6.2. Использование vfw2menc для создания файла настроек кодека.....	87
7.7. Использование MEncoder для создания совместимых с QuickTime файлов.....	87
7.7.1. Зачем необходимо создавать совместимые с QuickTime файлы?.....	87
7.7.2. Ограничения QuickTime 7.....	88
7.7.3. Обрезание.....	88
7.7.4. Масштабирование.....	88
7.7.5. A/V синхронизация.....	88
7.7.6. Битпоток.....	88
7.7.7. Пример кодирования.....	89
7.7.8. Ремультимплексирование в MP4.....	89
7.7.9. Добавление тегов метаданных.....	90
7.8. Использование MEncoder для создания VCD/SVCD/DVD-совместимых файлов.....	90
7.8.1. Ограничения формата.....	90
7.8.1.1. Ограничения форматов.....	90
7.8.1.2. Ограничения на размер GOP.....	91
7.8.1.3. Ограничения на битпоток.....	91
7.8.2. Опции вывода.....	91
7.8.2.1. Пропорции.....	91
7.8.2.2. Сохранение A/V синхронизации.....	91
7.8.2.3. Преобразование частоты дискретизации.....	91
7.8.3. Использование libavcodec для VCD/SVCD/DVD кодирования.....	92

7.8.3.1. Введение.....	92
7.8.3.2. lavcopts.....	92
7.8.3.3. Примеры.....	92
7.8.3.4. Расширенные опции.....	92
7.8.4. Кодирование звука.....	93
7.8.4.1. toolame.....	93
7.8.4.2. twolame.....	93
7.8.4.3. libavcodec.....	93
7.8.5. Собирая все вместе.....	93
7.8.5.1. PAL DVD.....	93
7.8.5.2. NTSC DVD.....	93
7.8.5.3. PAL AVI, содержащий AC-3 звук, в DVD.....	93
7.8.5.4. NTSC AVI, содержащий AC-3 звук, в DVD.....	93
7.8.5.5. PAL SVCD.....	93
7.8.5.6. NTSC SVCD.....	93
7.8.5.7. PAL VCD.....	94
7.8.5.8. NTSC VCD.....	94
Глава 8. Часто Задаваемые вопросы.....	94
8.1. Разработка.....	94
8.2. Компиляция и установка.....	94
8.3. Общие вопросы.....	95
8.4. Проблемы воспроизведения.....	97
8.5. Проблемы драйверов вывода видео/аудио (vo/ao).....	99
8.6. Воспроизведение DVD.....	100
8.7. Просьбы о новых возможностях.....	101
8.8. Кодирование.....	101

Авторские права © 2000-2015 MPlayer team

License

MPlayer is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

MPlayer is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with MPlayer; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Как читать эту документацию

Если Вы устанавливаете программу в первый раз, прочитайте все от начала до конца секции "Установка", и просматривайте ссылки, которые Вы обнаружите. Если у Вас все ещё остались вопросы, вернитесь к [Оглавлению](#) и поищите там, прочтите [FAQ](#), или попытайтесь про'grep'ить файлы. На большую часть вопросов Вы найдёте ответы здесь, а остальные, наверное, уже спрашивались в наших [рассылках](#). Проверьте [по архивам](#), в которых можно найти много ценной информации.

Глава 1. Введение

MPlayer — это проигрыватель фильмов для LINUX'a (работает на многих других UNIX'ах и не-x86 CPU, см. [Портинг](#)). Он проигрывает большинство MPEG, VOB, AVI, Ogg/OGM, VIVO, ASF/WMA/WMV, QT/MOV/MP4, FLI, RM, NuppelVideo, yuv4mpeg, FILM, RoQ, PVA, Matroska файлов, опираясь на множество "родных", XAnim'овских, RealPlayer'овских и Win32 DLL кодеков. Вы также можете смотреть VideoCD, SVCD, DVD, 3ivx, RealMedia, Sorenson, Theora, и MPEG-4 (DivX) фильмы. Другой важной особенностью MPlayer'a является широкий спектр поддерживаемых устройств вывода. Он работает с X11, Xv, DGA, OpenGL, SVGAlib, fbdev, AALib, libcaca, DirectFB, и кроме того Вы можете использовать GGI and SDL (и таким образом все их драйверы) и также несколько низкоуровневых драйверов для конкретных карт (для Matrox, 3DFx и Radeon, Mach64, Permedia3)! Большинство из них поддерживают программное или аппаратное масштабирование, поэтому Вы можете насладиться просмотром фильма на полном экране. MPlayer поддерживает некоторые аппаратные MPEG декодеры, такие как [DVB](#) и [DXR3/Hollywood+](#). А как насчёт славных больших сглаженных затенённых субтитров (14 поддерживаемых типов) с Европейскими/ISO 8859-1,2 (венгерский, английский, чешский, и т.п.), кириллическими, корейскими шрифтами и вывода информации на экран [On Screen Display (OSD)]?

Плеер без проблем проигрывает повреждённые MPEG файлы (полезно для некоторых VCD), плохие AVI файлы, которые не проигрываются известным Windows Media Player. Даже AVI файлы без индекса являются проигрываемыми. Вы можете временно сделать индекс с помощью ключа `-idx` или permanently с помощью MEncoder'a, таким образом получив возможность перемещаться по фильму! Как видите стабильность и качество — наиболее важные вещи, но скорость также изумительна. Кроме того имеется мощная система фильтров для манипуляции с видео и аудио.

MEncoder (Кодировщик фильмов MPlayer 'а) — это простой кодировщик фильмов предназначенный для кодирования фильмов, проигрываемых MPlayer'ом AVI/ASF/OGG/DVD/VCD/VOB/MPG/MOV/VIV/FLI/RM/NUV/NET/PVA в другие проигрываемые MPlayer'ом форматы (см. ниже). Он может кодировать такими разными кодеками, как MPEG-4 (DivX4) (1 или 2 прохода), libavcodec, PCM/MP3/VBR MP3 звук.

Возможности MEncoder'a

- Кодирование из широкого спектра форматов файлов и декодеров MPlayer'a
- Кодирование во все кодеки FFmpeg'овской библиотеки libavcodec
- Кодирование видео с V4L совместимых TV тюнеров
- Кодирование/мультиплексирование в "слоёные"[interleaved] AVI файлы с соответствующим индексом
- Создание файлов с аудио потоком из внешнего файла
- Кодирование в 1, 2 или 3 прохода
- **VBR** MP3 аудио
- PCM аудио
- Копирование потоков

- Входная A/V синхронизация (основана на PTS, может быть отключена с помощью ключа `-mc 0`)
- Коррекция FPS[кадров/сек] ключом `-ofps` (полезно при кодировании 30000/1001 fps VOB в 24000/1001 fps AVI)
- Использование нашей очень мощной системы плагинов (обрезание[`crop`], расширение[`expand`], отражение[`flip`], пост-обработка[`postprocess`], поворот[`rotate`], масштабирование[`scale`], RGB/YUV преобразования)
- Может кодировать DVD/VOBsub и текстовые субтитры в один выходной файл
- Может извлекать DVD субтитры в VOBsub формат

MPlayer и MEncoder могут распространяться в соответствии с GNU General Public License Version 2.

Глава 2. Установка

В файле **README** вы сможете найти короткое руководство по установке. Пожалуйста, сначала прочтите его, а затем вернитесь к оставшимся неясными деталям.

В этой главе я постараюсь провести Вас через процесс компиляции и конфигурации MPlayer'a. Это не просто, но это не обязательно будет сложно. Если Вы заметите какие-то отклонения, от того, что я объясняю, пожалуйста, поищите в этой документации и Вы найдёте ответ. Если Вы увидите ссылку, пожалуйста, проследуйте по ней и внимательно прочитайте её содержимое. Это займёт некоторое время, но это **ДЕЙСТВИТЕЛЬНО** того стоит

Вам нужна современная система. Под Linux'ом рекомендуются ядра 2.4.x.

2.1. Требуемые программы:

- **binutils** - рекомендуемая версия - это **2.11.x**.
- **gcc** - рекомендуемые версии: 2.95 и 3.4+. Известно, что 2.96 и 3.0.x генерируют испорченный код. С 3.1 и 3.2 были проблемы, с 3.3. тоже были небольшие проблемы. На PowerPC используйте 4.x+.
- **XOrg/XFree86** - рекомендуемая версия - 4.3 или более поздняя. Убедитесь, что также установлен **пакет разработки**, иначе это не будет работать. Вам необязательно иметь X, некоторые драйверы вывода видео работают и без него.
- **make** - рекомендуемая версия - 3.79.x или более поздняя. Для сборки XML документации требуется 3.80.
- **FreeType** - версия 2.0.9 или более поздняя требуется для OSD и субтитров.
- **ALSA** - необязательно, для поддержки вывода звука через ALSA Требуется минимум 0.9.0rc4.
- **libjpeg** - необходима для опционального драйвера вывода видео JPEG.
- **libpng** - необходима для опционального драйвера вывода видео PNG.
- **directfb** - необязательно, версия 0.9.13 или более поздняя требуется для драйвера вывода видео directfb.
- **lame** - рекомендуется 3.90 или новее, необходимо для кодирования MP3 аудио с MEncoder,
- **zlib** - рекомендуется, необходима для сжатых MOV заголовков и поддержки PNG.
- **[LIVE555 Streaming Media](#)** - необязательно, необходимо для некоторых RTSP/RTP потоков.
- **cdparanoia** - необязательно, для поддержки CDDA
- **libxmms** - необязательно, для поддержки входных плагинов XMMS. Требуется минимум 1.2.7.
- **libsmb** - необязательно, для поддержки сетевого протокола SMB.

2.2. Возможности:

- Решите, нужен ли Вам GUI. Если да, прочитайте до компиляции секцию [GUI](#).
- Если Вы хотите установить MEncoder (наш великолепный многоцелевой кодировщик), читайте [секцию MEncoder](#).
- Если у Вас есть V4L совместимый **TV тюнер**, и Вы хотите смотреть/захватывать и кодировать MPlayer'ом фильмы, читайте секцию [TV вход](#).
- Если у вас есть V4L совместимый **Radio тюнер**, и вы хотите слушать/записывать MPlayer'ом радиопередачи, читайте секцию [Радио](#).
- Существует изящное **OSD Меню** готовое для использования. Проверьте секцию [OSD Меню](#).

Теперь соберите MPlayer:

```
./configure
make
```


make install

В этот момент, MPlayer готов к использованию. Проверьте, не содержится ли файл `codecs.conf` в Вашем домашнем каталоге (`~/.mplayer/codecs.conf`) оставленный от предыдущих версий MPlayer'a, и удалите его.

Обратите внимание на то, что если у Вас в `~/.mplayer/` есть файл `codecs.conf`, то встроенный и системный файлы `codecs.conf` будут полностью игнорированы. Не делайте этого, если только Вы не собираетесь развлекаться со внутренностями MPlayer'a, поскольку это может вызвать множество проблем. Если Вы хотите поменять порядок подбора кодеков, используйте опции `-vc`, `-ac`, `-vfm`, и `-afm` либо в командной строке, либо в Вашем конфигурационном файле (см. страницу руководства).

Пользователи Debian могут сами создать .deb пакеты, это очень просто. Просто запустите `fakeroot debian/rules binary`

в корневом каталоге MPlayer'a. Более подробные инструкции см. в разделе [Создание Debian пакетов](#).

Всегда просматривайте вывод `./configure`, и файл `config.log`, они содержат информацию о том, что будет собрано, а что нет. Возможно Вы захотите просмотреть файлы `config.h` и `config.mak`. Если у Вас стоят какие-то библиотеки, которые не определяются `./configure`, проверьте, что у Вас установлены соответствующие заголовки[header files] (обычно это -dev пакеты) и их версии совпадают. Файл `config.log` Обычно сообщит Вам, чего не хватает для сборки.

Хотя это не обязательно, но чтобы получить функционирующие OSD и субтитры, должны быть установлены шрифты. Рекомендуемый метод - установка TTF шрифта и указание MPlayer'у использовать его. Подробности, см. в секции [Субтитры и OSD](#).

2.3. Как насчёт GUI?

Для GUI требуется GTK 1.2.x или GTK 2.0 (он не целиком GTK'шный, но панельки - да). Шкуры хранятся в PNG формате, поэтому GTK, libpng (и их части для разработчиков, обычно они называются `gtk-dev` и `libpng-dev`) должны быть установлены. Вы можете собрать GUI, указав `--enable-gui` в `./configure`. Затем, чтобы использовать GUI, Вы должны запускать **gmplayer**.

Поскольку MPlayer не содержит ни одной шкуры, Вы должны скачать их, если Вы хотите использовать GUI. См. [download page\[страницу зачек\]](#). Они должны быть извлечены в системный каталог (`$PREFIX/share/mplayer/skins`), или в `$HOME/.mplayer/skins`. По умолчанию, MPlayer ищет каталог `default` в этих каталогах, но вы можете использовать опцию `-skin newskin`, или директиву конфигурационного файла `skin=newskin`, чтобы использовать шкуру из каталога `*/skins/newskin`.

2.4. Шрифты и OSD

Чтобы насладиться OSD и субтитрами, надо указать MPlayer какой шрифт использовать. Подойдет любой TrueType или специальный растровый шрифт. Тем не менее, рекомендуется использовать TrueType шрифты, поскольку они лучше выглядят, корректно масштабируются до размера фильма, у лучше справляются с различными кодировками.

2.4.1. TrueType шрифты

Есть два способа заставить работать TrueType шрифты. Первый - указать опцию `-font`, чтобы указать путь к TrueType шрифту из командной строки. Эта опция будет хорошим кандидатом на помещение в файл конфигурации (для подробностей смотрите страницу `man` руководства). Второй - создание символической ссылки с именем `subfont.ttf` на выбранный файл шрифта. Либо

```
ln -s /path/to/sample_font.ttf ~/.mplayer/subfont.ttf
```

индивидуально для каждого пользователя, либо одну общесистемную

```
ln -s /path/to/sample_font.ttf $PREFIX/share/mplayer/subfont.ttf
```

Если MPlayer был скомпилирован с поддержкой `fontconfig`, эти методы не будут работать, вместо этого, опция `-font` ожидает `fontconfig` название шрифта, и по умолчанию это шрифт `sans-serif`. Пример:

```
mplayer -font 'Bitstream Vera Sans' anime.mkv
```

Чтобы получить список шрифтов, известных `fontconfig`, используйте **fc-list**

2.4.2. Растровые шрифты

Если по каким-то причинам вам необходимо использовать растровый шрифт, скачайте набор таковых с нашего сайта. Вы можете выбирать между различными [ISO шрифтами](#) и набором шрифтов [созданных пользователями](#) в различных кодировках.

Распакуйте скачанный файл в `~/.mplayer` или `$PREFIX/share/mplayer`. После этого переименуйте каталог или создайте символическую ссылку на него с именем `font`, например:

```
ln -s ~/.mplayer/arial-24 ~/.mplayer/font
```

```
ln -s $PREFIX/mplayer/arial-24 $PREFIX/mplayer/font
```

Шрифты должны иметь соответствующий им `font.desc` файл, отображающий позиции юникод шрифта в кодовую страницу текста субтитров. Другое решение - иметь субтитры, кодированные в UTF-8 и использовать опцию `-utf8` или просто дать файлу с субтитрами такое же, как у видео файла, имя с расширением `.utf`, положив его в один каталог с фильмом.

2.4.3. OSD меню

У MPlayer'a существует целиком определяемый пользователем интерфейс OSD меню.

Примечание

меню Preferences[Настройки] в настоящий момент НЕ НАПИСАНО!

Установка

1. скомпилируйте MPlayer, указав `./configure` параметр `--enable-menu`
2. убедитесь, что у Вас установлен OSD шрифт
3. скопируйте `etc/menu.conf` в Ваш каталог `.mplayer`
4. скопируйте `etc/input.conf` в Ваш каталог `.mplayer`, или в системный конфигурационный каталог MPlayer'a (по умолчанию: `/usr/local/etc/mplayer`)
5. проверьте и отредактируйте `input.conf`, чтобы включить кнопки перемещения по меню (это здесь описано).
6. запустите MPlayer как в следующем примере:
`mplayer -menu file.avi`
7. нажмите любую меню-кнопку, которую Вы определили

2.5. Codec installation

2.5.1. Xvid

[Xvid](#) свободный, MPEG-4 ASP совместимый видео кодек, особенностями которого являются двухпроходное кодирование и полная поддержка MPEG-4 ASP. Имейте в виду, что Xvid не нужен для декодирования Xvid-кодированного видео. `libavcodec` используется по-умолчанию, т.к. обеспечивает более высокую скорость.

Установка Xvid

Как и большинство ПО с открытым исходным кодом, он доступен как в виде [официальных релизов](#), так и в виде CVS версии. Как правило, CVS версия достаточно стабильна для использования, т.к. в большинстве случаев ее особенностью является отсутствие ошибок, присутствующих в релизах. Далее описывается как заставить работать Xvid CVS с MEncoder:

1. `cvs -z3 -d:pserver:anonymous@cvs.xvid.org:/xvid login`
2. `cvs -z3 -d:pserver:anonymous@cvs.xvid.org:/xvid co xvidcore`
3. `cd xvidcore/build/generic`
4. `./bootstrap.sh`
5. `./configure`

Вам может потребоваться добавить некоторые опции (посмотрите вывод `./configure --help`).

6. `make && make install`
7. Перекомпилируйте MPlayer.

2.5.2. x264

[x264](#) is a library for creating H.264 video. MPlayer sources are updated whenever an x264 API change occurs, so it is always suggested to use MPlayer from Subversion.

If you have a GIT client installed, the latest x264 sources can be gotten with this command:

```
git clone git://git.videolan.org/x264.git
```

Then build and install in the standard way:

```
./configure && make && make install
```

Now rerun `./configure` for MPlayer to pick up x264 support.

2.5.3. AMR кодеки

Речевой кодек Adaptive Multi-Rate используется в мобильных телефонах третьего поколения (3G). Исходная реализация доступна с [The 3rd Generation Partnership Project](#) (бесплатна для личного использования). Чтобы включить поддержку, скачайте и установите библиотеки поддержки для [AMR-NB](#) и [AMR-WB](#), следуя инструкциям на указанной странице.

2.6. RTC

У MPlayer'a есть три метода синхронизации.

- **Чтобы использовать старый метод синхронизации**, Вам ничего не надо делать. Он использует `usleep()`, чтобы подстроить A/V синхронизацию, с точностью +/- 10ms. Однако, иногда требуется даже большая точность синхронизации.
- **Новый таймер** использует RTC (Real Time Clock[часы истинного времени]) для этой задачи потому, что это таймер точностью 1ms. Он включается опцией `-rtc`, но требует правильно настроенного ядра. Если Вы работаете с ядром 2.4.19pre8 или более

поздним, Вы можете настроить максимальную частоту RTC для обычных пользователей через файловую систему /proc. Используйте одну из этих двух команд, чтобы сделать RTC доступным для обычных пользователей:

```
echo 1024 > /proc/sys/dev/rtc/max-user-freq  
sysctl dev/rtc/max-user-freq=1024
```

Можно сделать эти изменения постоянными, добавив последнюю в `/etc/sysctl.conf`.

Вы увидите эффективность нового кода таймера по строке состояния. Функции управления потребляемой мощностью[`power management`] у некоторых ноутбуковских BIOS'ов со `speedstep-CPU` плохо взаимодействуют с RTC. Аудио и видео могут десинхронизироваться. Вероятно, если Вы вставите штекер внешнего питания до включения ноутбука, то это поможет. В некоторых аппаратных комбинациях (подтверждено при использовании не-DMA DVD драйва с материнской платой ALi1541) использование RTC таймера приводит к "прыгающему"[`skippy`] проигрыванию. В этом случае рекомендуется использовать третий метод.

- **Третий код таймера** включается опцией `-softsleep`. У него эффективность RTC, но он не использует RTC. С другой стороны, он сильнее использует CPU.

Глава 3. Использование

3.1. Командная строка

MPlayer использует составное дерево воспроизведения. Опции, указанные в командной строке, могут применяться ко всем файлам/URL или только к некоторым, в зависимости от местоположения. Например

```
mplayer -vfm ffmpeg movie1.avi movie2.avi
```

, будет использовать FFmpeg декодеры для каждого из двух файлов, а

```
mplayer -vfm ffmpeg movie1.avi movie2.avi -vfm dmo
```

будет воспроизводить второй файл, используя DMO декодер.

Вы можете группировать файлы/URL'ы вместе, используя { и }. Это полезно, например, с опцией `-loop`:

```
mplayer { 1.avi -loop 2 2.avi } -loop 3
```

Эта команда проиграет файлы в таком порядке: 1, 1, 2, 1, 1, 2, 1, 1, 2.

Воспроизведение файла:

```
mplayer [опции] [путь/]имя_файла
```

Другой способ:

```
mplayer [опции] file:///uri-escaped-path
```

Воспроизведение множества файлов:

```
mplayer [общие опции] [путь/]имя_файла1 [опции для имя_файла1] имя_файла2 [опции для имя_файла2] ...
```

Воспроизведение VCD:

```
mplayer [опции] vcd://номер_дорожки [-cdrom-device /dev/cdrom]
```

Воспроизведение DVD:

```
mplayer [опции] dvd://номер_ролика [-dvd-device /dev/dvd]
```

Воспроизведение из WWW:

```
mplayer [опции] http://site.com/file.asf
```

(так же можно использовать и списки проигрывания (плейлист[`playlist`]))

Воспроизведение по RTSP:

```
mplayer [опции] rtsp://server.example.com/streamName
```

Примеры:

```
mplayer -vo x11 /mnt/Films/Contact/contact2.mpg  
mplayer vcd://2 -cdrom-device /dev/hdc  
mplayer -afm 3 /mnt/DVDtrailers/alien4.vob  
mplayer dvd://1 -dvd-device /dev/hdc  
mplayer -abs 65536 -delay -0.4 -nobps ~/movies/test.avi
```

3.2. Субтитры и OSD

Вместе с фильмом MPlayer может показывать и субтитры. В настоящий момент поддерживаются следующие форматы:

- VOBsub
- OGM
- CC (closed caption[скрытые титры])
- MicroDVD
- SubRip
- SubViewer

- Sami
- VPlayer
- RT
- SSA
- PJS (Phoenix Japanimation Society)
- >MPsub
- AQTitle
- [JACosub](#)

MPlayer может конвертировать вышеперечисленные форматы субтитров (**кроме первых трёх**) в следующие форматы с помощью соответствующих опций:

- MPsub: `-dumpmpsub`
- SubRip: `-dumpsrtsub`
- MicroDVD: `-dumpmicrodvds`
- JACosub: `-dumpjacosub`
- Sami: `-dumpsami`

MEncoder может выдавать DVD субтитры в [VOBsub](#) формате.

Опции значительно различаются для различных форматов:

VOBsub субтитры. VOBsub субтитры состоят из большого (несколько мегабайт) **.SUB** файла, и необязательных **.IDX** и/или **.IFO** файлов. Использование: если у Вас, например, есть файлы **sample.sub**, **sample.ifo** (необязательно), **sample.idx** - Вы должны указать MPlayer'у опции `-vobsub sample [-vobsubid id]` (можно указать полный путь). Опция `-vobsubid` похожа на `-sid` для DVD, с её помощью, Вы можете выбирать между дорожками субтитров (языками). В случае, если `-vobsubid` пропущена, MPlayer попытается использовать языки, полученные через опцию `-slang` и перейти к пункту `langidx` в **.IDX** файле, чтобы выбрать язык субтитров. Если и это не удаётся, то субтитров не будет.

Другие субтитры. Прочие форматы субтитров состоят из единого текстового файла, содержащего информацию о синхронизации, местоположении и тексте субтитра. Использование: Если у Вас есть, например, файл **sample.txt**, Вы должны указать опцию `-sub sample.txt` (можно указать полный путь).

Регулировка синхронизации и местоположения субтитров:

`-subdelay sec`

Задерживает субтитры на **sec** секунд. Это значение может быть отрицательным.

`-subfps RATE`

Указывает количество кадров/сек для файла субтитров (вещественное число)

`-subpos 0-100`

Указывает позицию субтитров.

Если Вы наблюдаете увеличивающуюся задержку между фильмом и субтитрами, используя файл субтитров в формате MicroDVD, наиболее вероятно, что частота кадров у фильма и файла субтитров не совпадают. Пожалуйста, обратите внимание, что формат MicroDVD использует абсолютные номера кадров для синхронизации, но в нем нет информации частоте кадров и поэтому опция `-subfps` не может использоваться с этим форматом. Вы должны, если хотите навсегда решить эту проблему, вручную конвертировать частоту кадров файла субтитров. MPlayer может сделать для вас это преобразование:

```
mplayer -dumpmicrodvds -fps subtitles_fps -subfps avi_fps \
-sub subtitle_filename dummy.avi
```

О DVD субтитрах, читайте в секции [DVD](#).

3.3. Управление

MPlayer использует полностью конфигурируемый, основанный на понятии команды, уровень управления, позволяющий манипулировать MPlayer'ом с клавиатуры, мышью, джойстиком или с пульта дистанционного управления (используя LIRC). Полный список кнопок для управления с клавиатуры см. на map-странице.

3.3.1. Конфигурация управления

MPlayer позволяет повесить любую MPlayer'овскую команду на любую кнопку, используя простой конфигурационный файл. Синтаксис файла состоит из имени кнопки, сопровождающегося командой. По умолчанию конфигурационный файл находится в **\$HOME/.mplayer/input.conf**, но это можно изменить, указав опцию `-input conf` (относительный путь указывается относительно **\$HOME/.mplayer**).

Пример 3.1. Простой файл конфигурации ввода

```
##
## MPlayer input control file
```

```
##
```

```
RIGHT seek +10  
LEFT seek -10  
- audio_delay 0.100  
+ audio_delay -0.100  
q quit  
> pt_step 1  
< pt_step -1  
ENTER pt_step 1 1
```

3.3.2. Управление через LIRC

Linux Infrared Remote Control[Инфракрасное Удалённое Управление Linux'ом] — используйте легко собираемый в домашних условиях IR-приёмник, (почти) любой пульт управления и управляйте с их помощью Вашей Linux машиной. Подробности на www.lirc.org.

Если у Вас установлен ракет `lirc`, `configure` само его обнаружит. Если Всё прошло хорошо, MPlayer при старте напечатает сообщение, похожее на "Setting up lirc support...". Если произошла ошибка, он Вам сообщит. Если он не говорит ничего про LIRC, то поддержка LIRC не была скомпилирована. Вот так :-)

Имя приложения для MPlayer — угадайте — `mplayer`. Вы можете использовать все команды `mplayer`'а и даже можете использовать более одной команды, разделив их символами `\n`. Не забудьте включить флаг `repeat`[повтор] в `.lircrc`, когда это имеет смысл (перемещение, громкость и т.п.). Вот выдержка из моего `.lircrc`:

```
begin
```

```
    button = VOLUME_PLUS  
    prog = mplayer  
    config = volume 1  
    repeat = 1
```

```
end
```

```
begin
```

```
    button = VOLUME_MINUS  
    prog = mplayer  
    config = volume -1  
    repeat = 1
```

```
end
```

```
begin
```

```
    button = CD_PLAY  
    prog = mplayer  
    config = pause
```

```
end
```

```
begin
```

```
    button = CD_STOP  
    prog = mplayer  
    config = seek 0 1\npause
```

```
end
```

Если Вам не нравится стандартное место Вашего конфигурационного файла `lirc` (`~/.lircrc`), используйте опцию `-lircconf filename`, чтобы указать другой файл.

3.3.3. Подчинённый ("рабский") режим

Наличие подчинённого режима позволяет Вам создавать простые приложения к MPlayer'у. Когда режим включён (опцией `-slave`), MPlayer читает со стандартного входа команды, разделяемые символом конца строки (`\n`). Команды документированы в файле [slave.txt](http://www.mplayerhq.hu/DOCS/ENGLISH/SLAVE.TXT).

3.4. Сетевые потоки и каналы

MPlayer может проигрывать файлы по сети, используя HTTP, FTP, MMS или RTSP/RTP протокол. Проигрывание включается добавлением URL'а в командную строку. Также, MPlayer учитывает переменную среды `http_proxy` и использует прокси[proxy], если это возможно. Также можно заставить использовать прокси:

```
mplayer http_proxy://proxy.micorsops.com:3128/http://micorsops.com:80/stream.asf
```

MPlayer может считывать данные со стандартного входа (а не из поименованных каналов). Это может, например, использоваться при проигрывании файлов по FTP:

```
wget ftp://micorsops.com/something.avi -O - | mplayer -
```

Примечание

Мы рекомендуем включать `-cache` при проигрывании из сети:

```
wget ftp://micorsops.com/something.avi -O - | mplayer -cache 8192 -
```

3.4.1. Сохранение потоковых данных

Как только вам удалось воспроизвести любимый интернет-поток, вы можете воспользоваться опцией `-dumpstream`, чтобы сохранить его в файл. Например:

```
mplayer http://217.71.208.37:8006 -dumpstream -dumpfile stream.asf
```

сохранит данные из потока `http://217.71.208.37:8006` в `stream.asf`. Это работает для всех протоколов, поддерживаемых MPlayer, таких как MMS, RSTP, и других.

3.5. Приводы CD/DVD

Современные приводы CD-ROM могут работать на очень высоких скоростях, некоторые из них способны регулировать скорость чтения. Несколько аргументов за использование этой возможности:

- На высоких оборотах возрастает вероятность ошибки при чтении, особенно с плохо штампованных дисков. Уменьшение скорости может предотвратить потерю данных в этом случае.
- Многие CD-ROM приводы ужасно шумят, а снижение скорости может привести к уменьшению шума.

3.5.1. Linux

Вы можете уменьшить скорость вращения IDE CD-ROM приводов программами **hdparm**, **setcd** или **cdctl**. Это работает так:

```
hdparm -E [скорость] [устройство cdrom]
setcd -x [скорость] [устройство cdrom]
cdctl -bS [скорость]
```

Если используется эмуляция SCSI, Вам следует применять настройки к реальному IDE устройству, а не сэмулированному SCSI.

Если у Вас есть привилегии администратора, следующая команда тоже может оказаться полезной:

```
echo file_readahead:2000000 > /proc/ide/[устройство cdrom]/settings
```

Таким образом, предварительно считывается 2 мегабайта (полезно при дисках с царапинами). Если поставить слишком большое значение, то постоянный запуск и остановка вращения диска ужасно снизят эффективность. Рекомендуется также подстроить привод, используя **hdparm**:

```
hdparm -d1 -a256 -u1 [устройство cdrom]
```

Этой командой включается прямой доступ к памяти [DMA], предварительное чтение и размаскировка IRQ (прочтите man-страницу **hdparm**, с более подробным описанием). Обратитесь к `"/proc/ide/[устройство cdrom]/settings"` для подстройки Вашего CD-ROM привода.

Вы можете настроить скорость SCSI CD-ROM приводов с помощью **sdparm**, необходима версия 1.03 или выше:

```
sdparm --command=speed=[скорость в кБ/с] [устройство cdrom]
```

Скорость должна быть указана в килобайтах в секунду, привод округлит её надлежащим образом. Пожалуйста, обратитесь с странице руководства **sdparm** для деталей.

There is also a dedicated tool that works for [Plextor SCSI drives](#).

3.5.2. FreeBSD

Скорость:

```
cdcontrol [-f устройство] speed [скорость]
```

DMA:

```
sysctl hw.ata.atapi_dma=1
```

3.6. Воспроизведение DVD

Полный список возможных опций можно прочитать в man странице. Синтаксис для воспроизведения стандартного DVD таков:

```
mplayer dvd://<ролик> [-dvd-device устройство]
```

Пример:

```
mplayer dvd://1 -dvd-device /dev/hdc
```

Если вы собрали MPlayer с поддержкой dvdnav, синтаксис тот же, просто укажите `dvdnav://` вместо `dvd://`.

Устройство DVD по умолчанию — это `/dev/dvd`. Если Ваши настройки отличаются, создайте символическую ссылку или укажите правильное устройство в командной строке, используя опцию `-dvd-device`.

MPlayer использует библиотеки `libdvdread` и `libdvdcss` для воспроизведения и расшифровки DVD. Эти две библиотеки содержатся в дереве исходного кода MPlayer'a, так что отдельно устанавливать их не нужно. Вы также можете использовать уже установленные в системе версии библиотек, но это **не рекомендуется**, так как может приводить к ошибкам, несовместимости и потере скорости.

Примечание

В случае проблем с декодированием DVD, попробуйте отключить `supermount` или другие подобные удобства. Некоторые RPC-2 устройства могут требовать установку кода региона.

Расшифровка DVD. Расшифровка DVD осуществляется библиотекой `libdvdcss`. Метод может быть указан переменной окружения `DVDCSS_METHOD`, подробную информацию смотрите на странице руководства `man`.

3.6.1. Региональный код

Современные DVD приводы имеют дурацкое ограничение, называемое [региональным кодом](#). Это — способ заставить DVD приводы воспроизводить DVD диски, созданные для одного из шести различных регионов, на которые разделен мир. Как может группа людей сесть за стол, придумать подобное и при этом ожидать, что мир 21-го века поклонится их воле — это за пределами человеческого понимания.

Приводы, реализующие региональную защиту исключительно при помощи программного обеспечения, известны как RPC-1 приводы, реализующие ее аппаратно — RPC-2. RPC-2 приводы позволяют пять раз изменить код региона, после чего он фиксируется навсегда. В Linux Вы можете воспользоваться утилитой [regionset](#) для установки регионального кода Вашего DVD привода.

К счастью, возможна переделка RPC-2 приводов в RPC-1, через обновление прошивки. Укажите модель Вашего DVD привода в Вашем любимом поисковике или посмотрите на форуме и разделах загрузок на "[Странице прошивок](#)". Хотя обычные предостережения, касающиеся обновления прошивки, остаются в силе, опыт избавления от региональной защиты в основном положителен.

3.7. Воспроизведение VCD

Полный список возможных опций можно прочитать в `man`. Синтаксис для обычного Видео-CD (VCD):

```
mplayer vcd://<дорожка> [-cdrom-device <устройство>]
```

Пример:

```
mplayer vcd://2 -cdrom-device /dev/hdc
```

Устройство VCD по умолчанию — `/dev/cdrom`. Если Ваши настройки отличаются, создайте символическую ссылку добавьте правильное название в командной строке после опции `-cdrom-device`.

Примечание

По крайней мере Plextor'ы и некоторые Toshiba SCSI CD-ROM приводы показывают ужасную производительность при чтении VCD'ов. Это объясняется тем, что `CDROMREADRAW ioctl` на этих приводах реализован не полностью. Если Вы имеете некоторые познания в сфере программирования SCSI, пожалуйста [помогите нам](#) в написании поддержки SCSI generic для VCD.

В настоящий момент Вы можете извлечь данные с VCD, используя [readvcd](#), и воспроизвести получившийся файл MPlayer 'ом.

Структура VCD. VCD составлен из секторов CD-ROM XA, т.е. дорожек CD-ROM mode 2 form 1 и form 2:

- Первая дорожка записана в mode 2 form 2 формате, что, в частности, означает использование коррекции ошибок L2. Дорожка содержит файловую систему ISO-9660 с секторами по 2048 байт. Там содержатся метаданные VCD, и картинки, часто используемые в меню. Здесь также могут храниться фрагменты MPEG для меню, но каждый из них должен быть разбит на кусочки по 150 секторов. Еще файловая система может хранить файлы или программы, не имеющие отношения к работе с VCD.
- Вторая и остальные дорожки содержат MPEG-поток секторами по 2324 байта, по одному пакету MPEG PS на сектор вместо файловой системы. Это дорожки в формате mode 2 form 1 и хранят больше информации на один сектор за счет потери возможности некоторой коррекции ошибок. После первой дорожки также допустимо присутствие дорожки CD-DA. В некоторых ОС используются различные трюки, чтобы сделать эти не-ISO-9660 дорожки видимыми в файловой системе. Но Linux — это не тот случай (пока).

Здесь MPEG данные **не могут быть смонтированы**. (Вы когда-нибудь монтировали аудио диск для того, чтобы его воспроизвести?) Так как большинство фильмов находится именно на таких дорожках, попробуйте сначала `vcd://2`.

- Существуют VCD диски без первой дорожки (единственная дорожка без файловой системы). Они проигрываются, но не монтируются.
- Описание стандарта Video CD называется Philips "White Book" и, как правило, недоступна в онлайн, т.к. должна приобретаться у Philips. Более подробная информация о Video CD может быть найдена в [документации vcdimager](#).

Про файлы .DAT. Файл примерно в 600 мегабайт на первой дорожке не настоящий! Это так называемый ISO-переход, созданный, чтобы позволить Windows обрабатывать эти дорожки (Windows вообще запрещает приложениям использовать прямой доступ к устройствам). Под Linux Вы не можете копировать эти файлы (они выглядят, как мусор). Под Windows это возможно, поскольку там iso9660 эмулирует прямой доступ к дорожкам через этот файл. Чтобы проигрывать .DAT файл Вам нужен драйвер, из Linux версии PowerDVD. Это модифицированный драйвер iso9660 файловой системы (`vcdfs/isofs-2.4.X.0`), который способен эмулировать прямой доступ к дорожкам через этот файл. Если Вы замонтируете диск, используя их драйвер, Вы можете копировать и даже проигрывать .DAT файлы MPlayer'ом. Но это не будет работать со стандартным драйвером iso9660 из ядра Linux! Используйте вместо этого `vcd://`. Альтернативами для копирования VCD может послужить новый драйвер `cdfs` (не входит в официальное ядро) который показывает дорожки[сессии] на диске как файлы образов и `cdrdao`, приложение для побитового чтения/копирования CD.

3.8. Редактируемые списки решений [Edit Decision Lists] (EDL)

Система редактируемых списков решений [edit decision list] (EDL) позволяет Вам автоматически пропускать или заглушать части при воспроизведении, основываясь на специфичном для каждого фильма конфигурационном файле.

Это полезно для тех, кто может захотеть посмотреть фильм в "семейном" режиме. Вы можете исключить любые проявления насилия, ненормативной лексики, Jar-Jar Binks, и т. п. из фильмов, сообразуясь с Вашими личными предпочтениями. Помимо этого, существуют другие применения, например автоматический пропуск рекламы при просмотре фильмов.

Формат EDL файлов пока элементарен. в каждой строке находится одна команда, которая указывает, что делать (пропустить/выключить звук) и когда (используя pts в секундах).

3.8.1. Использование EDL файлов

Включите опцию `-edl <имя_файла>`, когда Вы запускаете MPlayer, с именем EDL файла, который Вы хотите использовать с видео.

3.8.2. Создание EDL файлов

Текущий формат файлов EDL:

[начальная секунда] [конечная секунда] [действие]

Где секунды - это числа с плавающей точкой (вещественные числа), а действие - это или 0 для пропуска или 1 для заглушения звука. Пример:

```
5.3 7.1 0
15 16.7 1
420 422 0
```

Это вызовет пропуск видео с 5.3 секунды до 7.1 секунды, затем заглушит звук на 15 секунде, включит обратно в 16.7 секунд и пропустит видео с 420 по 422 секунды. Эти действия будут происходить, когда таймер проигрывания достигнет указанных в файле значений.

Чтобы начать создать EDL файл, используйте опцию `-edlout <filename>`. При проигрывании, когда Вы хотите отметить предыдущие две секунды для пропуска, нажмите `i`. Соответствующая запись для этого времени будет добавлена в файл. Затем Вы можете вернуться и подстроить сгенерированный EDL файл также как и поправить действие по умолчанию (пропуск блока, указанного в строке).

3.9. Расширенные возможности аудио

3.9.1. Окружающее/Многоканальное[Surround/Multichannel] воспроизведение

3.9.1.1. DVD'шники

Большинство DVD и многие другие файлы содержат окружающий звук. MPlayer поддерживает воспроизведение такого звука, но не задействует его по-умолчанию, поскольку стерео оборудование более распространено. Для воспроизведения файла с более чем двумя каналами звука, используйте опцию `-channels`. Например, для воспроизведения DVD со звуком 5.1:

```
mplayer dvd://1 -channels 6
```

Имейте ввиду, что несмотря на название "5.1" на самом деле присутствует шесть независимых каналов. Если у вас есть соответствующее оборудование, вы спокойно можете добавить

опцию `channels` в конфигурационный файл `~/.mplayer/config` MPlayer'a. Например, для установки по умолчанию квадрофонического[quadraphonic] воспроизведения, добавьте такую строку:

```
channels=4
```

MPlayer будет выводить четырехканальный звук, если все четыре есть в проигрываемом файле.

3.9.1.2. Воспроизведение стерео звука на четырех колонках

По-умолчанию, MPlayer не дублирует никакие каналы, как и большинство звуковых драйверов. Если вы хотите, сделайте это самостоятельно:

```
mplayer filename -af channels=2:2:0:1:0:0
```

Объяснения ищите в разделе [копирование каналов](#).

3.9.1.3. Передача AC-3/DTS

DVD, как правило, имеют окружающий звук, кодированный в AC-3 (Dolby Digital) или DTS (Digital Theater System[система цифрового кинотеатра]) формате. Некоторое современное аудио оборудование способно самостоятельно декодировать эти форматы. MPlayer может быть сконфигурирован, чтобы передавать данные без их декодирования. Это работает только для наличие в вашей звуковой карте разъема S/PDIF (Sony/Philips Digital Interface[цифровой интерфейс Sony/Philips]).

Если ваше оборудование может декодировать и AC-3, и DTS, вы можете спокойно задействовать передачу для обоих форматов. В противном случае включайте передачу только того формата, который поддерживается оборудованием.

Чтобы включить передачу из командной строки:

- только для AC-3, используйте `-ac hwac3`
- только для DTS, используйте `-ac hwdts`
- для AC-3 и DTS, используйте `-afm hwac3`

Чтобы включить передачу в файле настроек MPlayer:

- только для AC-3: `ac=hwac3,`
- только для DTS: `ac=hwdts,`
- для AC-3 и DTS: `afm=hwac3`

Заметьте, что в конце `ac=hwac3,` и `ac=hwdts,` присутствует запятая (","). В этом случае MPlayer вернется к кодеку, который он обычно использует, при воспроизведении файла, не содержащего звука AC-3 или DTS. `afm=hwac3` запятой не требует; Когда указано семейство аудио, MPlayer вернется к нужному кодеку так или иначе.

3.9.1.4. Передача MPEG аудио

Передачики цифрового ТВ (такие как DVB и ATSC) и некоторые DVD, обычно имеют аудио потоки MPEG (в частности MP2). Некоторые аппаратные MPEG декодеры, такие как полнофункциональные DVB карты и DXR2 адаптеры имеют встроенную возможность декодирования этого формата. MPlayer может быть настроен для передачи аудио данных без из декодирования.

Для использования этого кодека:

```
mplayer -ac hwmpa
```

3.9.1.5. Matrix-кодированное[matrix-encoded] аудио

TODO

Этот раздел пока не написан, и не может быть завершен, пока кто-нибудь не предоставит нам образцы файлов для тестирования. Если у вас есть matrix-кодированные файлы, знаете где их можно достать или имеете информацию, которая может быть полезна, отошлите сообщение в рассылку [MPlayer-DOCS](#). Укажите в теме письма [matrix-encoded audio].

Если не появятся файлы или какая-нибудь информация, этот раздел будет удален.

Полезные ссылки:

- <http://electronics.howstuffworks.com/surround-sound5.htm>
- <http://www.extremetech.com/article2/0,1697,1016875,00.asp>

3.9.1.6. Эмуляция окружающего звука в наушниках

MPlayer содержит плагин HRTF (Head Related Transfer Function), основанный на [проекте MIT](#) откуда взяты измерения от микрофонов, вмонтированных в макет человеческой головы.

Хотя точная имитация системы окружающего звука[surround system] невозможна, MPlayer'овский плагин HRTF производит более пространственный окружающий звук на 2-х канальных наушниках. Обычное сведение, просто объединяет каналы в два; кроме объединения каналов, `hrtf` создает хитрое эхо, слегка увеличивает разделение стерео, и меняет громкость некоторых частот. Лучше ли звучит HRTF, зависит от исходного звука, и

является делом личного вкуса, но его определенного стоит попробовать.

Для воспроизведения DVD с HRTF:

```
mplayer dvd://1 -channels 6 -af hrtf
```

hrtf работает хорошо тоько с 5-ю или 6-ю каналами. Также, **hrtf** тербуется 48 kHz звук. DVD аудио уже kHz, но если у вас есть файл, который вы хотите воспроизвести при помощи **hrtf**, с другой частотой сэмплирования, необходимо его ресэмплировать[**resample**]:

```
mplayer filename -channels 6 -af resample=48000,hrtf
```

3.9.1.7. Решение проблем

Если вы ничего не слышите при использовании окружающего звука, проверьте настройки вашего микшера при помощи такой как **alsamixer** программы; очень часто по-умолчанию выходной звук выключен или его уровень установлен в ноль.

3.9.2. Манипуляции с каналами

3.9.2.1. Общая информация

К сожалению, нет стандарта, описывающего порядок следования каналов. Порядки, указанные ниже, таковые из AC-3 и довольно типичны; попробуйте их и увидите совпадают ли они с вашим источником. Каналы нумеруются с нуля.

mono[моно]

0. center[центральный]

stereo[стерео]

0. left[левый]

1. right[правый]

quadraphonic[квардрофонический]

0. left front[левый передний]

1. right front[правый передний]

2. left rear[левый задний]

3. right rear[правый задний]

surround 4.0[окружение 4.0]

0. left front[левый передний]

1. right front[правый передний]

2. center rear[центральный задний]

3. center front[центральный передний]

surround 5.0[окружение 5.0]

0. left front[левый передний]

1. right front[правый передний]

2. left rear[левый задний]

3. right rear[правый задний]

4. center front[центральный передний]

surround 5.1[окружение 5.1]

0. left front[левый передний]

1. right front[правый передний]

2. left rear[левый задний]

3. right rear[правый задний]

4. center front[центральный передний]

5. subwoofer[сабвуфер]

Опция **-channels** используется для запроса количества каналов у аудио декодера. Некоторые аудио кодеки используют указанное количество каналов для определения необходимо ли сведение каналов. Заметьте, что это не всегда отражается на количестве выходных каналов. Например, используя **-channels 4** для проигрывания стерео MP3 файла будет по-прежнему выводить звук на два канала, поскольку MP3 кодек не создает дополнительных каналов.

Аудио плагин **channels** может использоваться для создания или удаления каналов, и полезен для управления количеством каналов, отсылаемых на звуковую карту. Смотрите следующие разделы для получения информации о манипуляции каналами.

3.9.2.2. Воспроизведение моно на двух колонках

Моно звук намного лучше звучит, при воспроизведении на двух колонках - особенно при использовании наушников. Аудиофайлы, реально имеющие один канал, автоматически проигрываются через две колонки; к сожалению, множество файлов с моно звуком кодированы как стерео с тишиной в одном из каналов. Простейший и безопасный способ воспроизведения одинакового звука на обеих колонках состоит в использовании плагина **extrastereo**:

```
mplayer filename -af extrastereo=0
```

Он усредняет оба канала, делая каждый в два раза тише изначального. В следующих

разделах приводятся другие способы сделать то же самое без уменьшения громкости, но они сложнее и требуют указания различных опций в зависимости от того, какой канал остается. Если вам действительно требуется управлять громкостью, будет проще поэкспериментировать с плагином `volume` и определить верное значение. Например:

```
mpplayer filename -af extrastereo=0,volume=5
```

3.9.2.3. Копирование/перемещение каналов

Плагин `channels` может переместить любой или все каналы. Установка всех подопций плагина `channels` не так проста и требует определенной аккуратности.

1. Определитесь, сколько выходных каналов вам необходимо. Это первая подопция.
2. Посчитайте количество перемещаемых каналов. Это вторая подопция. Каждый канал может быть перемещен в несколько отличных каналов одновременно, но учтите, что исходный канал (даже при перемещении в одно место) будет пуст, пока в него не переместится какой-либо другой. Для копирования канала, оставляя исходный неизменным, просто переместите канал одновременно в требуемый и исходный. Например:

канал 2 --> канал 3

канал 2 --> канал 2

3. Запишите копии каналов в виде пары подопций. Заметьте, что первый канал - это 0, второй - 1 и т.д. Порядок следования значений не имеет, пока они правильно сгруппированы в пары *исходный:результатирующий*.

Пример: один канал на две колонки

Это пример другого способа воспроизвести один канал на обе колонки. В нем предполагается, что левый канал должен воспроизводиться, а правый надо отбросить. Выполняем шаги, описанные выше:

1. Для создания по каналу на каждую из колонок, первая подопция должна быть 2.
2. Левый канал надо переместить на правый и на себя, чтобы он не оставался пуст. Всего два перемещения, делаем вторую подопцию тоже равной "2".
3. Для перемещения левого канала (канал 0) в правый (канал 1) пара подопций имеет вид "0:1", "0:0" перемещает левый канал на себя.

Собираем все вместе:

```
mpplayer filename -af channels=2:2:0:1:0:0
```

Преимущество этого примере перед `extrastereo` состоит в том, что громкость каждого канала такая же как у исходного. Недостаток заключается в необходимости изменить подопции на "2:2:1:0:1:1", если желаемый канал - правый. К тому же его труднее запомнить и набрать.

Пример: левый канал на две колонки (сокращение)

На самом деле есть более простой способ использования плагина `channels` для воспроизведения левого канала на обеих колонках:

```
mpplayer filename -af channels=1
```

Второй канал отбрасывается и, при отсутствии других подопций, остается единственным. Драйвер звуковой карты автоматически воспроизводит одноканальный звук на обеих колонках. Но это сработает только если желаемый канал - левый.

Пример: дублирование передних каналов на задние

Другая обычная операция - это дублирование передних каналов и воспроизведение их на задних колонках при квадрофонической настройке.

1. Выходных каналов должно быть четыре. Первая подопция равна "4".
2. Каждый из передних каналов надо переместить на соответствующий задний и на себя. Это четыре перемещения, так что вторая подопция равна "4".
3. Левый передний (канал 0) надо переместить на левый задний (канал 2): "0:2". Левый передний также надо переместить на себя: "0:0". правый передний (канал 1) перемещается на правый задний (канал 3): "1:3", и на себя: "1:1".

Собираем все и получаем:

```
mpplayer filename -af channels=4:4:0:2:0:0:1:3:1:1
```

3.9.2.4. Микширование каналов

Плагин `pan` может микшировать каналы в указанных пользователем пропорциях. Он может делать все, что `channels`, и даже больше. К сожалению, подопции намного сложнее.

1. Определите со сколькими каналами будете работать. Вам необходимо указать это при помощи `-channels` и/или `-af channels`. Дальнейшие примеры покажут когда какую использовать.
2. Решите, сколько каналов скормить `pan` (дополнительные декодированные каналы

отбрасываются). Это первая подопция, она также определяет сколько каналов готовится к выводу).

3. Оставшиеся подопции указывают какая часть каждого входного канала микшируется в в каждый выходной. Это самая сложная часть. Для решения задачи, разделите подопции на несколько наборов, по одному на каждый выходной канал. Каждая подопция в наборе относится к входному каналу.

`pan` принимает значения от 0 до 512, давая от 0% до 51200% громкости исходного канала. Будьте осторожны, используя значения больше 1, если вы превысите диапазон сэмплинга вашей звуковой карты, вы услышите противный треск и скрежет. Если хотите, можете вслед за `pan` указать `,volume` для задействования обрезки, но лучше держать значения `pan` достаточно низкими, чтобы не требовалось обрезание.

Пример: один канал на две колонки

Это еще один пример воспроизведения левого канала на двух колонках. Следуя инструкциям выше:

1. `pan` должен выдать два канала, т.о. первая подопция равна "2".
2. Поскольку входных каналов два, будет два набора подопций. Так как выходных каналов тоже два, то будет по две подопции в каждом наборе. Левый канал из файла должен перейти с полной громкостью в новые левый и правый. Таким образом, первый набор подопций будет "1:1". правый канал должен быть отброшен, поэтому второй набор равен "0:0". Любые значения 0 в конце могут быть опущены, но для более легкого понимания мы их оставим.

Соединение опций дает:

```
mpplayer filename -af pan=2:1:1:0:0
```

Если вместо правого канала нужен левый, подопции для `pan` будут "2:0:0:1:1".

Пример: левый канал на две колонки (сокращение)

Как и с `channels`, существует сокращенный вариант, который работает только для левого канала:

```
mpplayer filename -af pan=1:1
```

Поскольку `pan` имеет только один входной канал (остальные отбрасываются), будет только одна подопция, указывающая, что единственный канал получает 100% собственной громкости.

Пример: сведение 6-канального PCM

Декодер MPlayer'a для 6-канального PCM не способен сводить каналы. Здесь описан способ сведения PCM, используя `pan`:

1. Количество выходных каналов равно 2, значит первая подопция равна "2".
2. С шестью входными каналами будем иметь шесть наборов подопций. К счастью, поскольку мы беспокоимся о выводе только первых двух, достаточно создать два набора; оставшиеся можно опустить. Имейте ввиду, что не все многоканальные имеют одинаковый порядок каналов. пример показывает как свести файл с порядком как у AC-3 5.1:

0 - передний левый
1 - передний правый
2 - задний левый
3 - задний правый
4 - центральный передний
5 - сабвуфер

В первом наборе указаны проценты от исходной громкости, в соответствующем порядке, которую каждый выходной канал получит от переднего левого канала: "1:0". Правый передний должен перейти в правый: "0:1". То же для задних: "1:0" и "0:1" Центральный должен попасть в оба с половинной громкостью: "0.5:0.5", и сабвуфер переходит в оба канала с полной громкостью: "1:1".

Все вместе:

```
mpplayer 6-channel.wav -af pan=2:1:0:0:1:1:0:0:1:0.5:0.5:1:1
```

Проценты, указанные выше всего лишь пример. подстраивайте их как вам удобно.

Пример: Воспроизведение звука 5.1 на больших колонках без сабвуфера

Если у вас есть пара огромных передних колонок, нет надобности тратиться на сабвуфер для полноценной системы 5.1. Если использовать `-channels 5` для запроса к `liba52` на декодирование 5.1 аудио в 5.0, канал сабвуфера просто отбрасывается. Если вы хотите самостоятельно распределить канал сабвуфера, то потребуется ручное сведение при помощи `pan`:

1. Поскольку `pan` надо анализировать все шесть каналов, укажите `-channels 6`, чтобы

liba52 декодировал их все.

2. `pan` выводит только пять каналов, первая подопция равна 5.
3. Шесть входных каналов означает шесть наборов по пять подопций в каждом.
 - Левый передний дублируется только на себя: "1:0:0:0:0"
 - То же для правого переднего: "0:1:0:0:0"
 - То же для левого заднего: "0:0:1:0:0"
 - И то же для правого заднего: "0:0:0:1:0"
 - Центральный передний, тоже: "0:0:0:0:1"
 - И, наконец, мы должны решить что же делать с сабвуфером, например, половина на передний правый и половина на передний левый: "0.5:0.5:0:0:0"

Собирая все подопции вместе, получаем:

```
mplayer          dvd://1          -channels          6          -af
pan=5:1:0:0:0:0:0:1:0:0:0:0:0:1:0:0:0:0:0:1:0:0:0:0:0:1:0:0:0:0:0:1:0.5:0.5:0:0:0
```

3.9.3. Программная подстройка звука

Некоторые звуковые дорожки без усиления слишком тихие для комфортного прослушивания. Это становится проблемой, если звуковое оборудование не способно усиливать сигнал. Опция `-softvol` указывает MPlayer'у использовать встроенный микшер. После этого вы можете использовать клавиши подстройки громкости (по-умолчанию 9 и 0) чтобы достичь значительно более громкого звучания. Заметьте, что это не исключает использования микшера вашей карты; MPlayer всего лишь усиливает сигнал перед отправкой его на звуковую карту. Следующим пример может являться неплохим началом:

```
mplayer quiet-file -softvol -softvol-max 300
```

Опция `-softvol-max` указывает максимально допустимый уровень звука в процентах от исходного. Например, `-softvol-max 200` позволит увеличивать громкость вдвое по сравнению с оригинальным звуком. Использование больших значений с `-softvol-max`; высокий уровень громкости не будет достигнут без использования клавиш регулирования громкости. Единственный минус больших значений заключается в том, что, поскольку MPlayer регулирует громкость в процентах от максимума, вы не будете иметь той же точности при использовании клавиш регулирования громкости. Используйте меньшее значение `-softvol-max` и/или укажите `-volstep 1` если нужна повышенная точность.

Опция `-softvol` работает, управляя аудио плагином `volume`. Если вам надо воспроизвести файл с определенной громкостью от начальной, можете указать `volume` вручную:

```
mplayer quiet-file -af volume=10
```

Будет воспроизведен файл в усилении в 10 децибел. Будьте осторожны, используя плагин `volume` - вы можете легко повредить ваши уши слишком громким звуком. Начните с маленьких значений и постепенно увеличивайте, пока не почувствуете, что достаточно. Также, если указать черезчур высокие значения, `volume` может потребоваться обрезать звук, чтобы избежать отправления на карту данных, превышающих допустимые значение; это приведет к искажению звука.

3.10. TV вход

В этой секции описывается, как включить **просмотр/захват с V4L-совместимого TV тюнера**. См. `man` страницу, для описания TV опций и кнопок управления.

3.10.1. Компиляция

1. Во-первых, Вам нужно перекомпилировать MPlayer, `./configure` автоматически обнаружит заголовки ядра, относящиеся к v4l, и наличие `/dev/video*` устройств. Если они существуют, будет собрана поддержка TV (см. вывод `./configure`).
2. Убедитесь, что Ваш тюнер работает с другими TV приложениями под Linux, например XawTV.

3.10.2. Советы по использованию

Полный список опций доступен на страницах руководства (`man`). Вот всего несколько советов:

- Используйте опцию `channels`. Пример:
`-tv channels=26-MTV1,23-TV2`
Объяснение: При использовании такой опции, будут использоваться только каналы 26 и 23, и, кроме того, будет приятный OSD текст при переключении между каналами, отображающий название канала. Пробелы в названиях каналов должны быть заменены символом `"_"`.
- Выберите разумные размеры изображения. Размеры полученного изображения должны делиться на 16.
- Если Вы захватываете видео с вертикальным разрешением выше половины полного

разрешения (т.е. 288 для PAL или 240 для NTSC), то получаемые вами 'кадры' на самом деле будут чередующимися[interleaved] парами полей. В зависимости от того, что вы собираетесь делать с видео, можно оставить их в таком виде, произвести разрушающую (с возможной потерей качества) построчную развёртку, либо разделить пары обратно в отдельные поля.

Иначе Вы получите фильм с сильными искажениями в сценах с быстрыми движениями, и управление битпотокком, скорее всего, будет даже не в состоянии поддерживать необходимый уровень битпоттока, поскольку артефакты чересстрочной развёртки создают огромное количество мелких деталей и поэтому отнимают большую часть полосы пропускания. Вы можете включить преобразование в построчную развёртку, используя `-vf pp=DEINT_TYPE`. Обычно `pp=lb` работает хорошо, но это уже субъективное мнение. Другие алгоритмы преобразования в построчную развёртку см. на [map-странице](#) и попробуйте их.

- Обрежьте пустое пространство. Когда вы захватываете видео, зоны по краям, как правило, черны или содержат просто шум. Это опять съедает часть битпоттока. Точнее, это не сами чёрные зоны, а контрастный переход от чёрного к более светлому видео, но это сейчас не важно. Прежде чем Вы начнёте захватывать, подстройте аргументы опции `crop`, чтобы обрезать весь мусор по краям. Ещё раз, не забудьте сохранить получившиеся размеры изображения разумными.
- Отслеживайте загрузку CPU. Она не должна пересекать 90% границу большую часть времени. Если у Вас большой размер буфера захвата, MEncoder переживёт перегрузку в течение нескольких секунд, но не более того. Лучше отключить 3D OpenGL, хранители экрана и другую подобную гадость.
- Не меняйте системные часы. MEncoder использует системные часы для A/V синхронизации. Если Вы переведёте системные часы (особенно назад), MEncoder запутается, и Вы начнёте терять кадры. Это особенно важный вопрос, если Вы подключены к сети и используете какие-нибудь программы синхронизации времени, в духе NTP. Вы должны отключить NTP во время захвата, если Вы действительно хотите сделать хорошую запись.
- Изменяйте значение `outfmt` только если Вы знаете, что Вы делаете, или Ваши карта/драйвер не поддерживают значение по умолчанию (пространство цветов YV12). В старых версиях MPlayer/ MEncoder было необходимо выставлять нужное значение формата вывода. Эта проблема должна быть решена в текущих версиях и опция `outfmt` больше не требуется, поскольку значение по умолчанию подходит в большинстве случаев. Например если Вы будете захватывать в DivX, используя libavcodec и укажете `outfmt=RGB24` для улучшения качества полученного изображения, то Вы увидите, что в действительности, изображение все равно будет перекодировано в YV12, поэтому все что Вы получите, это огромная загрузка CPU.
- Чтобы использовать пространство цветов I420 (`outfmt=i420`), Вы должны указать опцию `-vc rawi420` в связи с конфликтом fourcc с видео кодеком Intel Indeo.
- Есть несколько путей захвата аудио. Вы можете получить звук, либо используя Вашу звуковую карту и внешний кабель, соединяющий видео карту и линейный вход[line-in], либо используя встроенный АЦП на чипе bt878. В этом случае, Вы должны загрузить драйвер `btaudio`. Читайте файл [linux/Documentation/sound/btaudio](#) (в дереве ядра, не MPlayer'a) с некоторыми инструкциями по использованию этого драйвера.
- Если MEncoder не может открыть аудио устройство, убедитесь, что оно действительно доступно. Возможны некоторые трудности со звуковыми серверами, например arts (KDE) и esd (GNOME). Если у Вас полнодуплексная звуковая карта (почти все современные карты это поддерживают), и Вы используете KDE, попробуйте отметить галочку "full duplex" в меню настроек звукового сервера.

3.10.3. Примеры

Фиктивный вывод, AALib :)

```
mplayer -tv driver=dummy:width=640:height=480 -vo aatv://
```

Ввод со стандартного V4L:

```
mplayer -tv driver=v4l:width=640:height=480:outfmt=i420 -vc rawi420 -vo xv tv://
```

Более изощрённый пример. Это заставляет MEncoder захватывать полное PAL изображение, обрезать края и изменить развёртку картинки на построчную, используя алгоритм линейного смешивания. Аудио сжимается до постоянного битпоттока 64 кБ/с, используя LAME кодек. Эти установки подходят для захвата фильмов.

```
mencoder -tv driver=v4l:width=768:height=576 -oac mp3lame -lameopts cbr:br=64\
```

```
-ovc lavc -lavcopts vcodec=mpeg4:vbitrate=900 \
```

```
-vf crop=720:544:24:16,pp=lb -o output.avi tv://
```

Здесь, изображение будет дополнительно масштабировано до 384x288 и сжато с битпотокком 350 кБ/с в режиме высокого качества. Опция `vqmax` даёт волю квантайзеру и позволяет компрессору видео действительно достичь столь низкого битпоттока, правда ценой качества. Это может быть полезно для захвата длинных TV серий, где качество не особенно важно.

```
mencoder -tv driver=v4l:width=768:height=576 \  
-ovc lavc -lavcopts vcodec=mpeg4:vbitrate=350:vhq:vqmax=31:keyint=300 \  
-oac mp3lame -lameopts cbr:br=48 -sws 1 -o output.avi \  
-vf crop=720:540:24:18,pp=lb,scale=384:288 tv://
```

Также возможно указать меньшие размеры изображения в опции `-tv` и пропустить программное масштабирование, но приведённый подход использует максимальное доступное количество информации и чуть более устойчив к шуму. Чипы `bt8x8` из-за аппаратных ограничений могут усреднять пиксели только по горизонтали.

3.11. Телетекст

На данный момент телетекст доступен только в MPlayer для v4l и v4l2 драйверов.

3.11.1. Замечания реализации

MPlayer поддерживает обычный текст, псевдографику и навигационные ссылки. К сожалению, цветные страницы поддерживаются пока не полностью - все страницы отображаются оттенками серого. Страницы с субтитрами (еще известные как Closed Captions) тоже поддерживаются.

MPlayer начинает кэшировать все страницы телетекста с момента начала просмотра TV, так что вам не потребуется ожидать загрузки интересующий страницы.

Замечание: Использование телетекста с `-vo xv` приводит к появлению странных цветов.

3.11.2. Использование телетекста

Чтобы включить декодирование телетекста, вы должны указать VBI устройство, из которого следует читать данные (обычно `/dev/vbi0` в Linux). Это можно сделать, указав `tdevice` в вашем файле конфигурации:

```
tv=tdevice=/dev/vbi0
```

Вам может потребоваться указать код языка телетекста для вашей страны. Чтобы получить список всех доступных языковых кодов, используйте

```
tv=tdevice=/dev/vbi0:tlang=-1
```

Вот пример для Русского:

```
tv=tdevice=/dev/vbi0:tlang=33
```

3.12. Радио

3.12.1. Радио вход

В этой секции описывается как включить прослушивание радио при помощи V4L совместимого Радио тюнера. Смотрите `man` страницу для описания опций и кнопок управления.

3.12.1.1. Компиляция

1. Во-первых, вам необходимо перекомпилировать MPlayer при помощи `./configure` с указанием опций `--enable-radio` и (если хотите включить поддержку записи) `--enable-radio-capture`.
2. Убедитесь, что ваш тюнер работает с другими приложениями в Linux, например XawTV.

3.12.1.2. Советы по использованию

Полный список опций доступен на страницах руководства (`man`). Вот всего несколько советов:

- Использование `channels` опции. Пример:
`-radio channels=104.4-Sibir,103.9-Maximum`
Объяснение: при указании этой опции, будут доступны только радиостанции 104.4 и 103.9. Кроме того, будет приятный OSD текст при переключении между каналами, отображающий название канала. Пробелы в названиях каналов должны быть заменены символом `"_"`
- Есть несколько путей захвата аудио. Вы можете получить звук, либо используя Вашу звуковую карту и внешний кабель, соединяющий видео карту и линейный вход[`line-in`], либо используя встроенный ADC на чипе `saa7134`. В этом случае, Вы должны загрузить драйвер `saa7134-alsa` или `saa7134-oss`.
- MEncoder не может быть использован для захвата звука, поскольку он требует обязательного наличия видео-потока. Таким образом, вы можете производить захват либо используя программу `arecord` из проекта ALSA, либо используя `-ao pcm:file=file.wav`. Во втором случае вы не будете слышать ничего во время захвата (за исключением случая, когда вы используете `line-in` кабель, и слушаете звук непосредственно с линейного входа).

3.12.1.3. Примеры

Вход со стандартного V4L (используя line-in кабель, запись отключена.):

```
mplayer radio://104.4
```

Вход со стандартного V4L (используя line-in кабель, запись отключена. Используется интерфейс V4Lv1):

```
mplayer -radio driver=v4l radio://104.4
```

прослушивание второй радиостанции из списка:

```
mplayer -radio channels=104.4=Sibir,103.9=Maximm radio://2
```

Получение звука через шину pci с внутреннего ADC радио тюнера. В этом примере тюнер используется как вторая звуковая карта (ALSA устройство hw:1,0). Для карт, основанных на saa7134, либо saa7134-alsa, либо saa7134-oss модуль должен быть загружен.

```
mplayer -rawaudio rate=32000 radio://2/capture \
```

```
-radio adevice=hw=1.0:arate=32000:channels=104.4=Sibir,103.9=Maximm
```

Примечание

При использовании имен устройств ALSA, двоеточия необходимо заменить на равенства, запятые - на точки.

Глава 4. Устройства вывода видео

4.1. Настройка MTRR

ОЧЕНЬ важно проверить, правильно ли установлены MTRR регистры, поскольку они могут дать большой прирост производительности.

Выполните **cat /proc/mtrr**:

```
--($:~)-- cat /proc/mtrr
```

```
reg00: base=0xe4000000 (3648MB), size= 16MB: write-combining, count=9
```

```
reg01: base=0xd8000000 (3456MB), size= 128MB: write-combining, count=1
```

Здесь все верно, показана моя Matrox G400 с 16Мб памяти. Я выполнил это из-под XFree 4.x.x, который автоматически устанавливает регистры MTRR.

Если ничего не сработало, вам придется сделать это вручную. Во-первых, вы должны определить базовый адрес. Существует три способа выяснить его:

1. из сообщений запуска X11, например:

```
(--) SVGA: PCI: Matrox MGA G400 AGP rev 4, Memory @ 0xd8000000, 0xd4000000
```

```
(--) SVGA: Linear framebuffer at 0xd8000000
```

2. из **/proc/pci** (используйте команду **lspci -v**):

```
01:00.0 VGA compatible controller: Matrox Graphics, Inc.: Unknown device 0525
```

```
Memory at d8000000 (32-bit, prefetchable)
```

3. из сообщений драйвера ядра `mga_vid` (используйте **dmesg**):

```
mga_mem_base = d8000000
```

Теперь давайте найдем объем памяти. Это очень просто, просто преобразуйте размер видео памяти в шестнадцатеричный формат, или используйте таблицу:

1 MB	0x100000
2 MB	0x200000
4 MB	0x400000
8 MB	0x800000
16 MB	0x1000000
32 MB	0x2000000

Вы знаете базовый адрес и размер памяти, так давайте настроим регистры MTRR! Например, для вышеуказанной карты Matrox (base=0xd8000000) с 32Мб памяти (size=0x2000000) просто выполните:

```
echo "base=0xd8000000 size=0x2000000 type=write-combining" > /proc/mtrr
```

Не все CPU имеют MTRR. Например, старый K6-2 (около 266MHz, стейпинг 0) не имеет MTRR, но у стейпинга 12 они уже есть (запустите **cat /proc/cpuinfo** чтобы это проверить).

4.2. Xv

Под XFree86 4.0.2 или новее, вы можете использовать функции работы с YUV[YUV routines] вашей видеокарты, используя расширение XVideo, то, которое используется при указании опции **-vo xv**. К тому же этот драйвер поддерживает управление яркостью/контрастностью/цветностью/и т.д. (кроме случая использования старого ,медленного DivX кодека DirectShow, который везде это поддерживает), смотрите страницу [map](#).

Чтобы заставить его работать, убедитесь, что выполняется следующее:

1. Требуется использовать XFree86 4.0.2 или новее (предыдущие версии не меют XVideo)
2. Ваша карта действительно поддерживает аппаратное ускорение (современные - да)
3. X загружают расширение XVideo, это похоже на:

(II) Loading extension XVideo
в [/var/log/XFree86.0.log](#)

Примечание

Это всего лишь загружается расширение XVideo. При нормальной установке оно грузится всегда, это не означает, что загружена **аппаратная** поддержка XVideo.

4. Ваша карта имеет поддержку Xv для Linux. Чтобы это проверить, запустите **xvinfo**, являющуюся частью дистрибутива XFree86. Она должна выдать на экран длинный текст, похожий на этот:

```
X-Video Extension version 2.2
screen #0
```

```
Adaptor #0: "Savage Streams Engine"
```

```
number of ports: 1
```

```
port base: 43
```

```
operations supported: PutImage
```

```
supported visuals:
```

```
depth 16, visualID 0x22
```

```
depth 16, visualID 0x23
```

```
number of attributes: 5
```

(...)

```
Number of image formats: 7
```

```
id: 0x32595559 (YUY2)
```

```
guid: 59555932-0000-0010-8000-00aa00389b71
```

```
bits per pixel: 16
```

```
number of planes: 1
```

```
type: YUV (packed)
```

```
id: 0x32315659 (YV12)
```

```
guid: 59563132-0000-0010-8000-00aa00389b71
```

```
bits per pixel: 12
```

```
number of planes: 3
```

```
type: YUV (planar)
```

(...и т.д....)

Карта должна поддерживать YUY2 пакованные и YUV12 планарные[planar] форматы пикселей, чтобы быть полезной в MPlayer.

5. И, наконец, проверьте, что MPlayer собран с поддержкой 'xv'. Выполните **mplayer -vo help | grep xv**. Если поддержка 'xv' включена, то увидите похожую строку:

```
xv    X11/Xv
```

4.2.1. 3dfx карты

Хорошо известно, что старые 3dfx драйвера имеют проблемы с ускорением XVideo, они не поддерживают ни YUY2 ни YV12, ни т.п. Проверьте, что у вас XFree86 версии 4.2.0 или новее, он может работать с YV12 и YUY2, в то время как предыдущие версии, включая 4.1.0, **с YV12 приводят к краху**. Если вы столкнулись со странными эффектами при использовании **-vo xv**, попробуйте SDL (он тоже имеет поддержку XVideo), и посмотрите поможет ли это. Обратитесь к разделу [SDL](#) за подробностями.

ИЛИ, попробуйте НОВЫЙ **-vo tdfxfb** драйвер! Смотрите раздел [tdfxfb](#).

4.2.2. S3 карты

S3 Savage3D, должны прекрасно работать, но для Savage4 используйте XFree86 версии 4.0.3 или выше (в случае проблем с изображением попробуйте 16bpp). По поводу S3 Virge: она поддерживает xv, но карта сама по себе слишком медленная, так что лучше будет ее продать. Существует родной драйвер фреймбуфера для карт S3 Virge, аналогичный tdfxfb. Настройте ваш фреймбуфер (например, укажите ядру **"vga=792 video=vesa:mtrr"**) и воспользуйтесь **-vo s3fb (-vf yuy2** и **-dr** тоже помогут).

Примечание

Пока не ясно в каких моделях Savage отсутствует поддержка YV12, и преобразование осуществляется драйвером (медленно). Если вы грешите на свою карту, возьмите свежий драйвер или вежливо спросите в списке рассылки MPlayer-users о драйвере с поддержкой MMX/3DNow!.

4.2.3. nVidia карты

nVidia под Linux - не всегда хороший выбор ... Открытые драйвера XFree86 поддерживают

большинство этих карт, но в некоторых случаях придется использовать закрытый бинарный драйвер от nVidia, доступный на [сайте nVidia](#). Этот драйвер также всегда необходим для задействования 3D ускорения.

Карты Rival128 не имеют поддержки XVideo с драйвером nVidia от XFree86 :(Подайте жалобу nVidia.

Тем не менее, MPlayer имеет [VIDIX](#) драйвер для большинства карт nVidia. Сейчас он в стадии беты и имеет некоторые недостатки. За подробностями обращайтесь к разделу [nVidia VIDIX](#).

4.2.4. ATI карты

[Драйвер GATOS](#) (который стоит использовать, если у вас не Rage128 или Radeon) по умолчанию имеет включенную опцию VSYNC. Это значит, что скорость декодирования (!) синхронизирована с частотой обновления монитора. Если воспроизведение кажется медленным, попробуйте как-нибудь отключить VSYNC или установите частоту обновления в $n \cdot (\text{fps}[\text{кадров/с} \text{ фильма}]) \text{ Гц}$.

Radeon VE - если нужен X, используйте XFree86 4.2.0 или новее. Нет поддержки TV-выхода. Конечно, с MPlayer вы можете успешно получить **ускоренное** отображение, с или без **TV-выхода**, без каких-либо библиотек X. Читайте раздел [VIDIX](#).

4.2.5. NeoMagic карты

Эти карты можно найти во многих ноутбуках. Вы должны использовать XFree86 4.3.0 или более новый, или использовать [Xv-совместимые драйвера](#) от Stefan Seyfried. Просто выберите подходящий для вашей версии XFree86.

XFree86 4.3.0 включает поддержку Xv, недавно Bohdan Horst отослал небольшой [патч](#) для исходников XFree86, ускоряющий операции с фреймбуфером (и XVideo) в четыре раза. Патч был включен в XFree86 CVS и должен быть в следующем релизе после 4.3.0.

Чтобы сделать возможным воспроизведение фильмов DVD разрешения поправьте ваш XF86Config как указано здесь:

Section "Device"

```
[...]
Driver "neomagic"
Option "OverlayMem" "829440"
[...]
```

EndSection

4.2.6. Trident карты

Если хотите использовать Xv с картой Trident, учитывая, что они не работают с 4.1.0, установите XFree 4.2.0. 4.2.0 добавляет поддержку полноэкранного Xv с картой Cyberblade XP. Другой вариант: MPlayer имеет [VIDIX](#) драйвер для карт Cyberblade/i1.

4.2.7. Kyro/PowerVR карты

Если хотите использовать Xv с картами на базе Kyro (например, Hercules Prophet 4000XT), следует скачать драйверы с [сайта PowerVR](#).

4.2.8. Карты Intel

Эти карты можно обнаружить во многих ноутбуках. Рекомендуются Xorg последней версии. Для воспроизведения контента размера DVD (и более) поправьте ваш XF86Config/xorg.conf как указано здесь:

Section "Device"

```
[...]
Driver "intel"
Option "LinearAlloc" "6144"
[...]
```

EndSection

Отсутствие этой опции скорее всего приведет к появлению ошибки

X11 error: BadAlloc (insufficient resources for operation)

при попытке использовать `-vo xv`.

4.3. DGA

ПРЕАМБУЛА. Этот документ пытается сказать несколько слов о том, что такое DGA в целом и что может сделать DGA драйвер для MPlayer (а что нет).

ЧТО ТАКОЕ DGA. DGA это сокращение от *Direct Graphics Access* [*Прямой Доступ к Графике*] и означает обход программами X сервера и прямое изменение ими памяти фреймбуфера. Говоря техническим языком, это происходит при помощи отображения[mapping] памяти фреймбуфера в адресное пространство вашего процесса. Это позволяет ядром, только если у вас есть привилегии суперпользователя. Вы можете получить их либо войдя в систему под именем root, либо установив SUID бит на исполняемый файл MPlayer (**не рекомендуется**).

Есть две версии DGA: DGA1 используется XFree 3.x.x и DGA2, появившийся в XFree 4.0.1.

DGA1 предоставляет только прямой доступ в фреймбуферу, как описано выше. Для переключения видеорежимов придется обратиться в расширению XVidMode.

DGA2 объединяет возможности расширения XVidMode и, к тому же, позволяет изменять глубину цвета отображения. Таким образом, вы можете, работая, в основном, в X с 32-х битной глубиной цвета, переключиться на глубину 15 бит и наоборот.

Однако DGA имеет некоторые недостатки. Похоже, оно каким-то образом зависит от используемого графического чипа и реализации видеодрайвера сервера X, управляющего этим чипом. Так что он работает не на всех системах.

УСТАНОВКА ПОДДЕРЖКИ DGA ДЛЯ MPLAYER. Во-первых, убедитесь, что X загружает расширение DGA, смотрите в `/var/log/XFree86.0.log`:

(II) Loading extension XFree86-DGA

Смотрите, **крайне рекомендуется** XFree86 4.0.x или старше! DGA драйвер программы MPlayer определяется автоматически скриптом `./configure`, или можете принудительно указать его использование опцией `--enable-dga`.

Если драйвер не смог переключиться на меньшее разрешение, поэкспериментируйте с опциями `-vm` (только для X 3.3.x), `-fs`, `-bpp`, `-zoom` чтобы найти видеорежим в который поместиться фильм. Конвертера Пока что нет :(

Получите права root. DGA требует права root для прямой записи в видеопамять. Если хотите запускать от имени обычного пользователя, установите бит SUID на MPlayer:

```
chown root /usr/local/bin/mplayer
chmod 750 /usr/local/bin/mplayer
chmod +s /usr/local/bin/mplayer
```

Теперь это работает и под обычным пользователем.

Риск безопасности

В этом заключается **большой** риск безопасности! **Никогда** не делайте этого на сервере или компьютере, к которому имеют доступ другие люди, т.к. они могут получить права root через MPlayer с битом SUID.

Теперь используйте опцию `-vo dga`, и вперед! (мы надеемся :) Можете попробовать, работает ли у вас опция `-vo sdl:driver=dga`! Это намного быстрее!

ПЕРЕКЛЮЧЕНИЕ РЕЖИМОВ. DGA драйвер позволяет переключать режимы (менять разрешение) выходного сигнала. Это позволяет избежать (медленного) программного масштабирования и в то же время предоставить полноэкранное изображение. В идеале следует переключаться в режим с таким же (необязательно с сохранением пропорций) как у видеоданных разрешением, но X сервер позволяет переключаться в режимы, предопределенные в `/etc/X11/XF86Config` (`/etc/X11/XF86Config-4` для XFree 4.X.X соответственно). Они определяются так называемыми моделаинами[modelines] и зависят возможностей вашей видеокарты. X сервер читает этот файл при старте и отключает режимы, недопустимые для вашего оборудования. Вы можете определить какие режимы остались, посмотрев лог файл X11. Он может быть найден в: `/var/log/XFree86.0.log`.

Вот значения, про которые известно, что они работают с чипом Riva128 при использовании X драйвера nv.o.

Section "Modes"

```
Identifier "Modes[0]"
Modeline "800x600" 40      800 840 968 1056  600 601 605 628
Modeline "712x600" 35.0   712 740 850 900   400 410 412 425
Modeline "640x480" 25.175 640 664 760 800   480 491 493 525
Modeline "400x300" 20      400 416 480 528   300 301 303 314 Doublescan
Modeline "352x288" 25.10  352 368 416 432   288 296 290 310
Modeline "352x240" 15.750 352 368 416 432   240 244 246 262 Doublescan
Modeline "320x240" 12.588 320 336 384 400   240 245 246 262 Doublescan
```

EndSection

DGA & MPLAYER. DGA используется программой MPlayer двумя способами: можно указать SDL использовать его (`-vo sdl:driver=dga`) и с помощью DGA драйвера (`-vo dga`). Все сказанное выше верно для обоих; в следующих разделах будет рассказано как работает DGA драйвер для MPlayer.

ВОЗМОЖНОСТИ. DGA запускается указанием `-vo dga` в командной строке. По-умолчанию, он пытается переключить режим с ближайшим к оригинальному видео разрешением. Он преднамеренно игнорирует опции `-vm` и `-fs` (переключение видеорежимов и полноэкранный режим) - он всегда старается занять как можно большую площадь экрана переключением видеорежима, избегая таким образом использования дополнительных тактов CPU для масштабирования изображения. Если выбранный режим вам не нравится, можете принудительно указать использовать разрешение ближайшее к указанному вами опциями `-x`

и **-y**. При указании опции **-v**, DGA драйвер выведет, кроме множества других вещей, список всех поддерживаемых режимов, указанных в **XF86Config**. Имея DGA2 вы также можете указать использование определенной глубины цвета при помощи опции **-bpp**. Допустимыми значениями являются 15, 16, 24 и 32. Зависит от оборудования, какие значения поддерживаются аппаратно, а для каких необходимо производить (возможно медленное) преобразование.

Если вам повезло иметь достаточно свободной памяти[**offscreen memory**], чтобы поместить туда изображение целиком, DGA драйвер будет использовать двойную буферизацию, что приведет к более плавному воспроизведению фильма. Он сообщит вам включена ли двойная буферизация или нет.

Двойная буферизация означает, что каждый следующий кадр вашего фильма рисуется в некоторую память[**offscreen memory**], пока отображается текущий кадр. Когда следующий кадр готов, графическому чипу сообщается его расположение в памяти, и чип просто выбирает оттуда данные для отображения. В это время новыми видео данными заполняется другой участок буфера.

Двойная буферизация может быть задействована опцией **-double** и отключена при помощи **-nodouble**. В данный момент двойной буфер по-умолчанию отключен. При использовании DGA драйвера, экранное отображение (OSD) работает только с двойной буферизацией. Однако, включение двойной буферизации может привести к существенному снижению скорости (на моем K6-II+ 525 оно использует дополнительные 20% времени CPU!) в зависимости от реализации DGA для вашего оборудования.

ПРОБЛЕМЫ БЫСТРОДЕЙСТВИЯ. Проще говоря, DGA доступ к фреймбуферу должен быть настолько быстр, насколько быстр используемый X11 драйвер с дополнительной выгодой[**benefit**] получения полноэкранного изображения. Процентные значения скорости, выводимые MPlayer, должны интерпретироваться с некоторой осторожностью, например, с драйвером X11 они не включают время, используемое сервером X11 непосредственно для прорисовки. Подключите терминал к последовательному порту и запустите **top**, чтобы увидеть, что на самом деле происходит.

Проще говоря, ускорение, полученное от использования DGA относительно 'обычного' использования X11, сильно зависит от видео карты и того, насколько хорошо оптимизирован модуль X11 для него.

Если у вас медленная система, лучше использовать глубину 15 или 16 бит, поскольку это потребует половину пропускной способности памяти 32-х битного дисплея.

Использование глубины 24 бита - хорошая идея, даже если ваша карта аппаратно поддерживает только 32 бита, поскольку передается на 25% меньше данных по сравнению с режимом 32/32.

Приходилось видеть, как некоторые AVI файлы воспроизводились на Pentium MMX 266. AMD K6-2 CPU может работать начиная с 400 МГц и выше.

ИЗВЕСТНЫЕ ОШИБКИ. Ну, по мнению некоторых разработчиков XFree, DGA - это немного монстр. Они говорят, что лучше его не использовать. Его реализация не безупречна для любого существующего драйвера XFree. изъянов.

- С XFree 4.0.3 и **nv.o** существует ошибка приводящая к странным цветам.
- ATI драйвер требует неоднократного переключения режима после завершения использования DGA.
- Некоторые драйвера просто не в состоянии переключиться обратно в нормальный режим (используйте **Ctrl+Alt+Keypad +** и **Ctrl+Alt+Keypad -** для нормального переключения).
- Некоторые драйвера просто отображают странные цвета.
- Некоторые драйвера неверно сообщают о количестве памяти, которое они отобразили в адресное пространство процесса, так что **vo_dga** не будет использовать двойную буферизацию (SIS?).
- Некоторые драйвера, похоже, не могут сообщить даже об одном верном режиме. В этом случае DGA рухнет, сообщая о невероятном режиме 100000x100000 или о чем-нибудь похожем.
- OSD работает только с задействованным двойным буфером (иначе он моргает).

4.4. SDL

SDL (Simple Directmedia Layer) - это, в основном, унифицированный видео/аудио интерфейс. Программы, его использующие, знают только про SDL, а не про то, какой видео или аудио драйвер реально используется. Например, порт игры Doom, используя SDL, может запуститься на **svgalib**, **aalib**, **X**, **fbdev** и других, вам придется только указать (для примера) используемый видеодрайвер при помощи переменной окружения **SDL_VIDEODRIVER**. Ну, в теории.

Для карт/драйверов, не поддерживающих XVideo, мы использовали в MPlayer собственные возможности программного масштабирования SDL'вских X11 драйверов, пока не написали наш собственный (более быстрый и изящный) программный модуль масштабирования. Также мы использовали его aalib вывод, но теперь у нас есть свой, более удобный. До некоторых пор его DGA режим был лучше нашего. Хотите получить его прямо сейчас ? :)

Он также помогает с некоторыми сбойными драйверами/картами, если видео прерывается (если это не проблема недостаточного быстродействия), или заикается звук.

SDL видео вывод поддерживает отображения субтитров внизу, на черной полосе (если она есть).

4.5. SVGAlib

УСТАНОВКА. потребуется установить svgalib и ее пакет разработки, чтобы MPlayer собрал свой SVGAlib драйвер (определяется автоматически, но можно включить принудительно), и отредактировать `/etc/vga/libvga.config` в соответствии с вашей картой и монитором.

Примечание

Убедитесь, что не используете опцию `-fs`, поскольку она включает использование программного масштабирования и работает медленно. Если вам действительно это необходимо, используйте опцию `-sws 4`, которая будет давать плохое качество, но несколько быстрее.

ПОДДЕРЖКА EGA (4BPP). SVGAlib включает в себя EGAlib, и MPlayer имеет возможность выводить любой фильм в 16-ти цветах. Используется в таких вариантах:

- EGA карта с EGA монитором: 320x200x4bpp, 640x200x4bpp, 640x350x4bpp
- EGA карта с CGA монитором: 320x200x4bpp, 640x200x4bpp

Значение bpp (бит на пиксел) должно быть вручную установлено в 4: `-bpp 4`

Возможно потребуется отмасштабировать фильм, чтобы уместить в размеры EGA режима:

```
-vf scale=640:350
```

or

```
-vf scale=320:200
```

Для масштабирования требуется быстрый алгоритм с плохим качеством:

```
-sws 4
```

Возможно надо отключить автоматическую коррекцию пропорций:

```
-noaspect
```

Примечание

Как показывает мой опыт, лучшее качество на EGA экране получается при небольшом уменьшении яркости: `-vf eq=-20:0`. Мне также пришлось уменьшить частоту дискретизации [samplerate] на моей машине, поскольку звук не работал на 44кГц. `-srate 22050`.

Вы можете включить OSD и субтитры только с плагином `expand`, смотрите страницу руководства man для точных параметров.

4.6. Вывод во фреймбуфер (FBdev)

Надо ли собирать FBdev автоматически определяется скриптом `./configure`. Прочтите документацию на фреймбуфер в исходниках ядра (`Documentation/fb/*`) для более подробной информации.

Если ваша карта не поддерживает стандарт VBE 2.0 (старые ISA/PCI карты, такие как S3 Trio64), а только VBE 1.2 (или еще старше?): ну, VESAfb все же будет доступна, но вам потребуется загрузить SciTech Display Doctor (она же UniVBE) до загрузки Linux. Используйте загрузочный диск DOS или что либо другое. И не забудьте зарегистрировать ваш UniVBE ;))

Драйвер FBdev вместе с прочими принимает несколько дополнительных параметров:

```
-fb указывает какой устройство фреймбуфера использовать (по-умолчанию: /dev/fb0)
```

```
-fbmode название используемого режима (в соответствии с /etc/fb.modes)
```

```
-fbmodeconfig конфигурационный файл с режимами (по-умолчанию: /etc/fb.modes)
```

```
-monitor-hfreq, -monitor-vfreq, -monitor-dotclock
```

важные значения, смотрите `example.conf`

При желании переключиться в особый режим используйте:

```
mplayer -vm -fbmode название_режима имя_файла
```

- `-vm` в одиночку выберет наиболее подходящий режим из `/etc/fb.modes`. Также может использоваться совместно с `-x` и `-y`. Опция `-flip` поддерживается, только если формат точки фильма совпадает с форматом точки видеорежима. Обратите внимание на значение bpp, fbdev пытается использовать текущий или указанный вами опцией `-bpp`.
- Опция `-zoom` не поддерживается (используйте `-vf scale`). Вы не можете использовать

- режимы с 8bpp (или меньше).
- Вы, возможно, захотите отключить курсор:
`echo -e '\033[?25l'`
или
`setterm -cursor off`
и хранитель экрана:
`setterm -blank 0`
Чтобы снова включить курсор:
`echo -e '\033[?25h'`
или
`setterm -cursor on`

Примечание

Смена режимов FBdev *не работает* с VESA фреймбуфером. Не просите об этом, т.к. это не ограничения MPlayer.

4.7. Matrox фреймбуфер (mga_vid)

mga_vid - это комбинация драйвера вывода и модуля ядра Linux, использующая модуль видео масштабирования/оверлея Matrox G200/G400/G450/G550 для выполнения YUV->RGB преобразования цветового пространства и произвольного масштабирования. mga_vid имеет аппаратную поддержку VSYNC с тройной буферизацией. Работает как во фреймбуфер консоли, так и под X, но только с Linux 2.4.x.

Версию этого драйвера для Linux 2.6.x ищите на <http://attila.kinali.ch/mga/>.

Установка:

1. чтобы использовать его, придется, во-первых, скомпилировать **mga_vid.o**:
`cd drivers`
`make`
2. Затем запустите (под root)
`make install`
что должно установить модуль и создать для Вас файл устройства. Загрузите драйвер:
`insmod mga_vid.o`
3. Вам следует проверить определение размера памяти, используя команду **dmesg**. Если он неверен, укажите опцию **mga_ram_size** (но сначала **rmmod mga_vid**), указав размер памяти в Мб:
`insmod mga_vid.o mga_ram_size=16`
4. чтобы в случае необходимости загружать его автоматической, сначала добавьте следующую строку в конец файла **/etc/modules.conf**:
`alias char-major-178 mga_vid`
5. Теперь надо (пере)скомпилировать MPlayer, **./configure** определит **/dev/mga_vid** и соберет драйвер 'mga'. Использование его в MPlayer осуществляется опцией **-vo mga**, если используете консоль matroxfb, или опцией **-vo xmga** из-под XFree86 3.x.x или 4.x.x.

Драйвер mga_vid работает совместно с Xv.

Из файла устройства **/dev/mga_vid** можно получать некоторую информацию, например, командой

```
cat /dev/mga_vid
```

В него можно писать для изменения яркости:

```
echo "brightness=120" > /dev/mga_vid
```

В том же каталоге есть тестовая программа, называемая **mga_vid_test**. Если все работает нормально, она должна рисовать на экране изображения 256x256.

4.8. Поддержка 3Dfx YUV

Этот драйвер использует ядерный драйвер фреймбуфера tdfx для воспроизведения фильмой с YUV ускорением. Вам потребуется ядро с поддержкой tdfxfb и перекомпиляция с

```
./configure --enable-tdxfb
```

4.9. tdfx_vid

Это комбинация модуля Linux ядра и драйвера вывода видео, аналогичный [mga_vid](#). Вам потребуется 2.4.x ядро с драйвером agpgart, поскольку tdfx_vid использует AGP. Укажите **configure** опцию **--enable-tdxfb** для сборки драйвера вывода видео и соберите модуль ядра, как указано далее.

Установка tdfx_vid.o модуля ядра:

1. Скомпилируйте **tdfx_vid.o**
`cd drivers`
`make`
2. Запустите (от root):

make install

что должно установить модуль и создать для Вас файл устройства. Загрузите драйвер:
insmod tdfx_vid.o

3. Чтобы сделать его загружающимся/выгружающимся автоматически, сначала вставьте следующую строку в конец `/etc/modules.conf`:
alias char-major-178 tdfx_vid

В том же каталоге есть тестовая программа, называемаяся **tdfx_vid_test**. Она должна вывести некоторую полезную информацию, если все работает нормально.

4.10. OpenGL вывод

MPlayer поддерживает воспроизведение фильмов через OpenGL, но если ваша платформа/драйвер поддерживает xv, как в случае PC с Linux, лучше используйте xv, производительность OpenGL значительно ниже. Если у вас реализация X11 без поддержки xv, OpenGL жизнеспособная замена.

К сожалению, не все драйвера поддерживают эту возможность. Драйвера Utah-GLX (для XFree86 3.3.6) имеют ее для всех карт. Подробности установки смотрите <http://utah-glx.sf.net>. XFree86(DRI) 4.0.3 или новее поддерживает OpenGL с картами Matrox и Radeon, 4.2.0 или более поздние поддерживают и Rage128. Инструкции по закачиванию и установке смотрите на <http://dri.sf.net>.

Подсказка от одного из наших пользователей: видео вывод GL может использоваться для получения синхронизированного по вертикали TV вывода. Необходимо установить переменную окружения (как минимум для nVidia):

```
export __GL_SYNC_TO_VBLANK=1
```

4.11. AAlib - отображение в текстовом режиме

AAlib - это библиотека для отображения графики в текстовом режиме, используя мощный ASCII рендерер[renderer]. Существует множество программ уже поддерживающих ее, такие как Doom, Quake, и т.д. MPlayer содержит очень удобный драйвер для нее. Если `./configure` обнаруживает установленную aalib, будет собран libvo драйвер для aalib.

В AA Window можно использовать некоторые клавиши для изменения настроек рендеринга:

Клавиша	Действие
1	уменьшить контрастность
2	увеличить контрастность
3	уменьшить яркость
4	увеличить яркость
5	включение/отключение быстрого рендеринга
6	установка режима зашумления[dithering] (отсутствие, распределение ошибки[error distribution], Floyd Steinberg)
7	инвертировать изображение
8	переключение между управлением aa и MPlayer

Могут быть использованы следующие опции командной строки:

```
-aaosdcolor=V
```

Смена цвета OSD

```
-aasubcolor=V
```

Смена цвета субтитров

где V может быть: 0 (нормальный), 1 (темный), 2 (жирный), 3 (жирный шрифт), 4 (реверсный[reverse]), 5 (специальный).

AAlib сама предоставляет большое количество опций. Вот некоторые из важных::

```
-aadriver
```

Установить рекомендуемый aa драйвер (X11, curses, Linux).

```
-aaextended
```

Использовать все 256 символов.

```
-aaeight
```

Использовать восьмибитную ASCII.

```
-aahelp
```

Выводит все опции aalib.

Примечание

Рандеринг очень сильно загружает CPU, особенно при использовании AA-on-X (использование

aalib под X), и меньше при использовании стандартной не-фреймбуфер консоли. Используйте SVGATextMode, чтобы настроить большой текстовый режим и наслаждайтесь! (второй выход карт Hercules рулит[secondary head Hercules cards rock] :)), но, ИМНО, вы можете использовать опцию `-vf lbpp`, чтобы получить графику на hgafb :)

Используйте опцию `-framedrop`, если ваш компьютер недостаточно быстр для отрисовки всех кадров!

При воспроизведении на терминале, вы получите лучшую скорость и качество при использовании драйвера Linux, а не curses(`-aadrivers linux`). Но при этом вы должны иметь право записи в `/dev/vcsa<терминал>`! Это не определяется aalib автоматически, но `vo_aa` вместо нее пытается определить лучший режим. Смотрите <http://aa-project.sf.net/tune> для дальнейших задач тюнинга.

4.12. libcasa - Цветная ASCII Art библиотека

Библиотека `libcasa` - это графическая библиотека, выводящая чекст вместо пикселей, так что она может работать на старых видео картах или текстовых терминалах. Она не такая, как знаменитая AAlib. libcasa требует терминал для своей работы, так что она будет работать на всех unix системах (включая Mac OS X), используя библиотеку slang или ncurses, под DOS используя библиотеку conio.h, и под windows, используя либо slang, либо ncurses (через Cugwin эмуляцию), либо conio.h. Если скрипт `./configure` определяет libcasa, то casa libvo драйвер будет собран.

Отличия от AAlib следующие:

- 16 доступных цветов для вывода символов (256 цветовых пар)
- зашумление[dithering] цветного изображения

Но libcasa также имеет следующие ограничения:

- нет поддержки яркости, контрастности, гаммы

Вы можете использовать следующие клавиши в окне casa для изменения опций рендеринга:

Клавиш	Действие
a	Перключение методов зашумления[dithering] libcasa.
a	Переключение сглаживания[antialiasing] libcasa.
b	Переключение фона libcasa.

libcasa также анализирует следующие переменные окружения:

CACA_DRIVER

Установить рекомендуемый casa драйвер, например ncurses, slang, x11.

CACA_GEOMETRY (только X11)

Указывает количество строк и столбцов, например, 128x50.

CACA_FONT (только X11)

Указывает используемый шрифт, например, fixed, nexus.

Используйте опцию `-framedrop`, если ваш компьютер недостаточно быстр для рендеринга всех кадров.

4.13. VESA - вывод в VESA BIOS

Этот драйвер был разработан и представлен как **универсальный драйвер** для любых карт с VESA VBE 2.0 совместимым BIOS. Другое преимущество этого драйвера заключается в том, что он пытается принудительно включить TV вывод. *VESA BIOS EXTENSION (VBE) Версия 3.0 Дата: 16 сентября, 1998* (Страница 70) гласит:

Dual-Controller Designs. VBE 3.0 поддерживает дизайн с двумя контроллерами, предполагая, что поскольку оба контроллера обычно имеют одного производителя, и управляются единственной BIOS ROM на той же карте, то возможно скрыть от приложения факт наличия на самом деле двух контроллеров. Это ограничивает их независимое одновременное использование, но позволяет приложениям, выпущенным до VBE 3.0 нормально работать. VBE функция 00h (Вернуть информацию о контроллере) возвращает комбинированную информацию двух контроллеров, включая объединенный список доступных режимов. Когда приложение выбирает режим, активируется соответствующий контроллер. Каждая из остальных VBE функций затем работает с активным контроллером.

Так что у вас есть шанс получить работающий TV выход, используя этот драйвер. (Предполагается, что TV-выход - отдельный контроллер[standalone head] или отдельный выход как минимум.)

ПРЕИМУЩЕСТВА

- У вас есть шанс смотреть фильмы, **даже если Linux не знает** ваше видео

оборудование.

- Вам не требуется устанавливать в Linux ничего, относящегося к графике (вроде X11 (он же XFree86), fbdev и т.п.). Этот драйвер может запускаться из **текстового режима**.
- У вас есть шанс получить **работающий TV-выход**. (Это известно как минимум для ATI карт).
- Этот драйвер вызывает обработчик `int 10h`, так что это не эмуляция - он вызывает **реальные** вещи *реального BIOS в реальном режиме* (на самом деле `vm86` режим).
- С ним вы можете использовать VIDIX, получая ускоренное отображение видео, и TV вывод одновременно! (Рекомендуется для ATI карт.)
- Если у вас есть VESA VBE 3.0+, и вы где-то указали `monitor-hfreq`, `monitor-vfreq`, `monitor-dotclock` (в файле конфигурации или в командной строке), то получите наибольшую возможную частоту обновления. (Используя General Timing Formula). Чтобы задействовать эту возможность, вы должны указать **все** опции вашего монитора.

НЕДОСТАТКИ

- Это работает только на **x86 системах**.
- Может использоваться только пользователем `root`.
- В данный момент доступно только для **Linux**.

Важно

Не используйте этот драйвер с **GCC 2.96**! Он не будет работать!

ОПЦИИ КОМАНДНОЙ СТРОКИ, ДОСТУПНЫЕ ДЛЯ VESA

`-vo vesa:опции`

данный момент распознаются: `dga` для включения режима `dga` и `nodga` для его отключения. В `dga` режиме вы можете включить двойную буферизацию опцией `-double`.
Замечание: вы можете опустить эти параметры для **автоопределения** режима `dga`.

ИЗВЕСТНЫЕ ПРОБЛЕМЫ И СПОСОБЫ ИХ РЕШЕНИЯ

- Если вы установили **NLS** шрифт на вашем Linux и запускаете VESA драйвер из текстового режима, то после завершения MPlayer у вас окажется загруженным **ROM шрифт** вместо национального. Вы можете загрузить национальный шрифт снова, воспользовавшись утилитой `setsysfont` из дистрибутива Mandrake/Mandriva, например. (**Подсказка:** Та же утилита используется для локализации `fbdev`).
- Некоторые **графические драйверы Linux** не обновляют активный **BIOS режим** в DOS памяти. Таким образом, если у вас подобная проблема - всегда используйте VESA драйвер только из **текстового режима**. Иначе в любом случае будет активирован текстовый режим (`#03`) и вам придется перезагружать компьютер..
- Часто после завершения работы VESA драйвера вы получаете **черный** экран. Чтобы вернуться в обычный режим просто переключитесь на другую консоль (нажав `Alt+F<x>`) затем переключитесь обратно тем же способом.
- Для получения **работающего TV выхода** необходимо, чтобы TV разъем был подключен до включения вашего PC, т.к. видео BIOS инициализирует себя только один раз во время POST процедуры.

4.14. X11

Избегайте, если возможно. Вывод в X11 (используя расширение разделяемой памяти) - без какого-либо ускорения. Поддерживается (MMX/3DNow/SSE ускоренное, но все равно медленное) программное масштабирование, используйте опции `-fs -zoom`. Большинство карт имеют поддержку масштабирования, для этого используйте вывод `-vo xv`, или `-vo xmg` для карт Matrox.

Проблема в том, что большинство драйверов карт не поддерживают аппаратное ускорение на втором мониторе/TV. В этом случае вы увидите окно зеленого/синего цвета вместо изображения, и этот драйвер будет полезным, но требуется мощный CPU для программного масштабирования. Не используйте программный вывод+масштабирование SDL драйвера, он имеет худшее качество картинки!

Программное масштабирование очень медленное, лучше попробуйте вместо этого изменить видео режим. Это очень просто. Смотрите [раздел DGA режимы](#) и вставьте соответствующие строки в ваш `XFree86Config`.

- Если у вас XFree86 4.x.x: используйте `-vm` опцию. Она переключится в режим с подходящим разрешением. Если нет:
- С XFree86 3.x.x: циклически переключайтесь между разными разрешениями с помощью клавиш `Ctrl+Alt+Доп. клавиша плюс` и `Ctrl+Alt+Доп. клавиша минус`.

Если не находите вставленные видео режимы, просмотрите вывод `Xfree86`. Некоторые драйверы не могут использовать низкие частоты пикселизации (количество отрисовываемых

пикселей в секунду), необходимые для видео режимов с низким разрешением.

4.15. VIDIX

ПРЕАМБУЛА. VIDIX - это аббревиатура для **VIDeo Interface for *niX** (Видео интерфейс для Unix). VIDIX разработан и введен как интерфейс для быстрых драйверов пространства пользователя[user-space], обеспечивающих такую же производительность, как mga_vid для карт Matrox. Они к тому же хорошо переносимы.

Этот интерфейс был разработан, чтобы уместить существующие интерфейсы ускорения видео (известные как mga_vid, rage128_vid, radeon_vid, pm3_vid) в единую схему. Он предоставляет высокоуровневый интерфейс к чипам, известным как BES (BackEnd scalers) или OV (Video Overlays). Он не предоставляет низкоуровневого интерфейса к вещам, известным как графические серверы (Я не хочу соревноваться с командой X11 в переключении режимов.) Т.е. основная цель интерфейса - ускорить до максимума скорость воспроизведения видео.

ИСПОЛЬЗОВАНИЕ

- Вы можете использовать отдельный видеодрайвер: **-vo xvidix**. Этот драйвер был разработан как X11 фронтенд к VIDIX технологии. Он требует X сервер и может работать только под ним. Имейте ввиду, что поскольку этот драйвер напрямую обращается к оборудованию в обход драйвера X, то могут быть повреждены растровые изображения, кешированные в памяти видеокарты. Вы можете предотвратить это, ограничив размер видеопамати, используемой X, XF86Config опцией "VideoRam" в разделе устройств. Вам следует установить этот параметр в количество установленной видеопамати минус 4Мб. Если у вас меньше 8Мб видеопамати, вместо этого можно использовать опцию "XaaNoPixmapCache" в разделе экранов.
- Существует консольный VIDIX драйвер: **-vo cvidix**. Для большинства карт требуется работающий и инициализированный фреймбуфер (в противном случае просто испортите изображение на экране), и вы будете иметь тот же эффект, что и с **-vo mga** или **-vo fbdev**. Карты nVidia, тем не менее, способны выводить полностью графическое видео в настоящей текстовой консоли. Смотрите раздел [nvidia_vid](#) для более подробной информации. Чтобы избавиться от такста на полях и мерцающего курсора попробуйте нечто подобное
setterm -cursor off > /dev/tty9
(предполагая, что tty9 ранее не использовался) и затем переключитесь на tty9. С другой стороны, **-colorkey 0** должна дать вам видео, работающее "на фоне", однако правильность работы этого зависит от функциональности colorkey.
- Вы можете использовать подустройство VIDIX, примененное к различным драйверам видео вывода, например: **-vo vesa:vidix** (**только Linux**) и **-vo fbdev:vidix**.

Это действительно неважно, какой драйвер вывода видео используется с **VIDIX**.

ТРЕБОВАНИЯ

- Видеокарта должна находиться в графическом режиме (кроме карт nVidia с драйвером **-vo cvidix**).
- Драйвер вывода видео MPlayer должен знать текущий видеорежим и быть способным сообщить VIDIX некоторые видео характеристики сервера.

СПОСОБЫ ИСПОЛЬЗОВАНИЯ. Когда VIDIX используется в качестве **подустройства** (**-vo vesa:vidix**), настройка видеорежима производится драйвером вывода видео (короче говоря **vo_server**). Следовательно, вы можете передать в командную строку MPlayer те же ключи, что и для vo_server. Дополнительно он понимает ключ **-double** как глобально видимый параметр. (Я рекомендую использовать этот ключ с VIDIX как минимум для карт ATI). **-vo xvidix** дополнительно понимает следующие опции: **-fs -zoom -x -y -double**.

Вы можете напрямую указать VIDIX драйвер третьим параметром к командной строке:

```
mplayer -vo xvidix:mga_vid.so -fs -zoom -double file.avi
```

или

```
mplayer -vo vesa:vidix:radeon_vid.so -fs -zoom -double -bpp 32 file.avi
```

Но это опасно, и вам не следует этого делать. В этом случае принудительно запускается указанный драйвер и результат может быть непредсказуемым (он может **подвесить** ваш компьютер). Вам следует это делать ТОЛЬКО если вы абсолютно уверены, что он работает и MPlayer не использует его автоматически. Пожалуйста, сообщите об этом разработчикам. Правильный путь - использование VIDIX без аргументов для задействования автоопределения драйвера.

Поскольку VIDIX требует прямой доступ к оборудованию, вы можете либо запускать его от имени root, либо установить SUID бит на исполняемый файл MPlayer ((**Внимание: Это большой риск безопасности**)). Как вариант, вы можете использовать специальный модуль ядра, как этот:

1. Скачайте [разрабатываемую версию](#) svgalib (например 1.9.17), **ИЛИ** версию, созданную Alex специально для использования с MPlayer (она не требует наличия исходников svgalib для компиляции) [отсюда](#).
2. Скомпилируйте модуль в каталоге `svgalib_helper` (он может быть найден внутри каталога `svgalib-1.9.17/kernel/`, если вы скачали исходники с сайта svgalib) и выполните `insmod` для него.
3. Для создания необходимых устройств в каталоге `/dev`, выполните от имени root команду `make device` в `svgalib_helper`.
4. Переместите каталог `svgalib_helper` в подкаталог `vidix` дерева исходных текстов MPlayer.
5. Удалите комментарий перед строкой CFLAGS, содержащий строку "svgalib_helper" в файле `vidix/Makefile`.
6. Перекомпилируйте.

4.15.1. ATI карты

В данный момент для большинства карт ATI, начиная от Mach64 и заканчивая последними Radeon, имеется встроенная поддержка.

Существует два скомпилированных бинарных файла: `radeon_vid` для Radeon и `rage128_vid` для карт Rage 128. Вы можете принудительно использовать один из них или позволить VIDIX автоматически опробовать все доступные драйверы.

4.15.2. Matrox карты

Сообщалось, что работают Matrox G200, G400, G450 и G550.

Драйвер поддерживает видео эквалайзеры и должен быть столь же быстр, как и [Matrox фреймбуфер](#)

4.15.3. Trident карты

Существует драйвер для чипсета Trident Cyberblade/i1, который можно найти на материнских платах VIA Eria.

Драйвер написан и поддерживается [Alastair M. Robinson](#).

4.15.4. 3DLabs карты

Хотя драйвер для чипов 3DLabs GLINT R3 и Permedia3 существует, никто его не тестировал, так что отчеты приветствуются.

4.15.5. nVidia карты

Уникальная особенность драйвера `nvidia_vid` заключается в способности отображать видео в **простой, чисто текстовой консоли** - без какого бы то ни было фреймбуфера или магии с X. Для этой цели мы будем использовать драйвер вывода видео `cvidix`, как показывает следующий пример:

```
mplayer -vo cvidix example.avi
```

4.15.6. SiS карты

Это, как и `nvidia_vid`, весьма экспериментальный код.

Он тестировался на SiS 650/651/740 (наиболее распространенный чипсет, используемый в SiS версиях байрбонов[barebones] "Shuttle XPC")

Отчеты ожидаются!

4.16. DirectFB

"DirectFB - это графическая библиотека, которая была разработана с учетом особенностей встроенных систем. Она предоставляет максимум производительности при минимуме используемых ресурсов и накладных расходов." - процитировано с <http://www.directfb.org>

Я исключу описание возможностей DirectFB из этого раздела.

Несмотря на то, что MPlayer не поддерживается в DirectFB как "video провайдер", этот драйвер вывода видео задействует воспроизведение видео через DirectFB. Он будет - конечно - работать с ускорением, на моей Matrox G400 скорость DirectFB такая же как у XVideo.

Всегда старайтесь использовать последнюю версию DirectFB. Вы можете использовать опции DirectFB в командной строке, при помощи `-dfbopts`. Выбор слоя производится методом подустройства, например.: `-vo directfb:2` (по-умолчанию -1: автоопределение)

4.17. DirectFB/Matrox (dfbmga)

Прочтите [основной DirectFB](#) раздел для общей информации.

Этот драйвер вывода видео задействует CRTC2 (на втором мониторе[second head]) на картах Matrox G400/G450/G550, отображающий видео **независимо** от первого монитора[first head].

Ville Syrjala имеет [README](#) и [HOWTO](#) на своей странице, описывающие как задействовать

вывод DirectFB TV на картах Matrox.

Примечание

Первая версия DirectFB, на которой удалось это сделать, была 0.9.17 (имеющая ошибки, требует surfacemanager патч с указанного выше URL). Портирование кода CRT2 в [mga_vid](#) планируется уже несколько лет, [патчи](#) приветствуются.

4.18. MPEG декодеры

4.18.1. DVB ввод и вывод

MPlayer поддерживает карты с чипсетом Siemens DVB и таких производителей, как Siemens, Technotrend, Galaxis или Hauppauge. Последние DVB драйверы доступны с [сайта Linux TV](#). Если вы собираетесь делать программное транскодирование, у вас должен быть как минимум 1ГГц CPU.

Скрипт configure должен определить вашу DVB карту. Если нет, принудительно укажите определение с помощью

```
./configure --enable-dvb
```

Если заголовочные файлы ost находятся не в стандартных каталогах, укажите путь с

```
./configure --extra-cflags=каталог исходников DVB/ost/include
```

Затем компилируйте и устанавливайте как обычно.

ИСПОЛЬЗОВАНИЕ. Аппаратное декодирование потоков, содержащих MPEG-1/2 видео и/или MPEG аудио, может быть выполнено следующей командой:

```
mplayer -ao mpegpes -vo mpegpes file.mpg|vob
```

Декодирование любых других видео потоков требует транскодирования в MPEG-1, поэтому оно медленно и, возможно, не стоит неприятностей, особенно если ваш компьютер медленный. Его можно добиться, используя команду:

```
mplayer -ao mpegpes -vo mpegpes yourfile.ext  
mplayer -ao mpegpes -vo mpegpes -vf expand yourfile.ext
```

Имейте в виду, что DVB карты поддерживают высоту изображения только 288 и 576 для PAL и 240 и 480 для NTSC. Для других значений высоты вы **должны** отмасштабировать изображение, добавив `scale=ширина:высота` к опции `-vf` с желаемыми значениями ширины и высоты. DVB карты допускают различные значения ширины: 720, 704, 640, 512, 480, 352 и т.д. и производят аппаратное масштабирование по горизонтали, так что в большинстве случаев масштабировать по горизонтали не нужно. Для 512x384 (пропорции 4:3) MPEG-4 (DivX) попробуйте:

Если у вас есть широкоформатный фильм и вы не хотите растягивать его на полную высоту, используйте плагин `expand=w:h` для добавления черных полос. Чтобы посмотреть 640x384 MPEG-4 (DivX), попробуйте:

```
mplayer -ao mpegpes -vo mpegpes -vf expand=640:576 file.avi
```

Если ваш CPU слишком слаб для полноразмерного 720x576 MPEG-4 (DivX), попробуйте уменьшить размер:

```
mplayer -ao mpegpes -vo mpegpes -vf scale=352:576 file.avi
```

Если скорость не увеличилась, попробуйте уменьшить размер по вертикали тоже:

```
mplayer -ao mpegpes -vo mpegpes -vf scale=352:288 file.avi
```

Для OSD и субтитров используйте возможности OSD плагина expand. Так, вместо `expand=w:h` или `expand=w:h:x:y`, используйте `expand=w:h:x:y:1` (5-й параметр `:1` в конце для включения OSD рендеринга). Вы можете слегка подвинуть изображение вверх, чтобы получить большую черную область для субтитров. Вы также можете переместить субтитры вверх, если они выходят за пределы TV экрана, используйте для этого `-subpos <0-100>` (`-subpos 80` - хороший выбор).

Чтобы воспроизвести не-25fps фильмы на PAL TV или на машине со слабым CPU, добавьте опцию `-framedrop`.

Для сохранения пропорций MPEG-4 (DivX) файлов и получения оптимальных параметров масштабирования (аппаратное горизонтальное и программное вертикальное масштабирование с сохранением пропорций), используйте плагин `dvbscale`:

for a 4:3 TV: `-vf dvbscale,scale=-1:0,expand=-1:576:-1:-1:1`

for a 16:9 TV: `-vf dvbscale=1024,scale=-1:0,expand=-1:576:-1:-1:1`

Цифровое TV (DVB драйвер ввода). Вы можете использовать DVB карту для просмотра Цифрового TV.

Вы должны иметь установленные программы **scan** и **szap/tzap/czap/azap**; все они входят в пакет драйверов.

Проверьте, что ваши драйверы правильно работают с такими программами, как [dvbstream](#) (это основа DVB драйвера ввода).

Теперь вам следует создать файл `~/mplayer/channels.conf`, с понятным **szap/tzap/czap/azap** синтаксисом, или позволить **scan** сделать это для вас.

Если у вас несколько разнотипных карт (например, для спутникового TV, наземного, кабельного и ATSC), можете сохранить файлы как `~/mplayer/channels.conf.sat`, `~/mplayer/channels.conf.ter`, `~/mplayer/channels.conf.cbl`, и `~/mplayer/channels.conf.atsc`, соответственно, с тем, чтобы можно было явно указать MPlayer какой файл использовать вместо стандартного `~/mplayer/channels.conf`, и какую карту с ним использовать.

Убедитесь, что в `channels.conf` находятся каналы !-- FIXME Free to Air --> *только* для обычного телевидения[Free to Air], иначе MPlayer будет ожидать передачи незашифрованных данных.

В полях аудио и видео вы можете использовать расширенный синтаксис: `...:pid[+pid]:...` (в каждом максимум 6 pid'ов); в этом случае MPlayer включит в поток все указанные pid плюс pid 0 (содержащий PAT). В каждую строку всегда следует включать PMT и PCR pid'ы для соответствующего канала (если вы их знаете). Также можно указать 8192, при этом будут выбраны все pid на этой частоте, и Вы сможете потом переключаться между программами при помощи TAB. Это может потребовать большей пропускной способности, однако дешевые карты всегда пересылают все каналы как минимум до ядра, так что в этом случае разница будет небольшой. Другие возможные применения: телетекст pid, вторая аудио дорожка, и т.д.

Если MPlayer часто жалуется на

`Too many video/audio packets in the buffer`

или если вы заметили растущую рассинхронизацию между звуком и видео проверьте наличие PCR pid'a в вашем потоке (требующегося для соблюдения модели буферизации передатчика) и/или попробуйте воспользоваться libavformat MPEG-TS декодером, добавив `-demuxer lavf -lavfdopts probesize=128` в командную строку.

Для показа первого из доступных каналов, запустите

```
mplayer dvb://
```

Если вы хотите посмотреть определенный канал, например R1, запустите

```
mplayer dvb://R1
```

Если у вас больше одной карты, также надо указать номер той, с которой доступен канал:

```
mplayer dvb://2@R1
```

Для смены канала, нажимайте клавиши h (следующий) и k (предыдущий), или используйте [OSD меню](#).

Для временного отключения аудио или видео потока скопируйте следующие строки в `~/mplayer/input.conf`:

```
% set_property switch_video -2
```

```
& step_property switch_video
```

```
? set_property switch_audio -2
```

```
^ step_property switch_audio
```

(Заменяя горячие клавиши по своему усмотрению.) При нажатии на клавишу, соответствующую `switch_x -2`, поток будет закрыт; при нажатии на клавишу, соответствующую `step_x`, поток будет открыт снова. Имейте в виду, что этот механизм переключения не будет работать как следует, когда в мультиплексе присутствует несколько аудио или видео потоков.

Во время воспроизведения (не во время записи) для предотвращения заикания и таких ошибок, как 'Your system is too slow', имеет смысл добавить

```
-mc 10 -speed 0.97 -af scaletempo
```

к опциям командной строки, исправив параметры `scaletempo` по своему усмотрению.

Если ваш `~/mplayer/menu.conf` содержит запись `<dvbset>`, как в файле с примерами `etc/dvb-menu.conf` (можете использовать его чтобы перезаписать `~/mplayer/menu.conf`), главное меню будет отображать подменю, позволяющее вам выбрать один из присутствующих в `channels.conf` каналов, возможно, с предшествующим ему подменю с DVB выбором карт, доступных MPlayer.

Если хотите записать какую-то программу на диск, используйте

```
mplayer -dumpfile r1.ts -dumpstream dvb://R1
```

Если хотите вместо этого записать ее в другом формате (перекодировать), следует использовать такую команду:

```
mencoder -o r1.avi -ovc xvid -xvidencopts bitrate=800 \  
-oac mp3lame -lameopts cbr:br=128 -pp=ci dvb://R1
```

Полный список опций для DVB драйвера ввода можно найти на странице руководства `man`.

БУДУЩЕЕ. Если у вас есть вопросы, или вы хотите получать сообщения о новых возможностях

и принять участие в дискуссиях на эту тему, подпишитесь на список рассылки [MPlayer-DVB](#). Помните, что язык рассылки - Английский.

В будущем вы можете рассчитывать на возможность отображения OSD и субтитров, используя встроенные OSD возможности DVB карт.

4.18.2. DXR2

MPlayer поддерживает аппаратное ускорение воспроизведения с картами Creative DXR2.

Прежде всего вам потребуется правильно установленные DXR2 драйверы. Их и руководство по установке можно найти на сайте [Ресурсного Центра DXR2](#).

ИСПОЛЬЗОВАНИЕ

`-vo dxr2`

Включить TV выход.

`-vo dxr2:x11 or -vo dxr2:xv`

Включить вывод через оверлей в X11.

`-dxr2 <option1:option2:...>`

Эта опция используется для управления драйвером DXR2.

Чипсет оверлея, использовавшийся на DXR2 довольно плохого качества, но с настройками по умолчанию будет работать у всех. OSD может работать с оверлеем (не на TV), отрисовывая себя в ключевом цвете[`colorkey`]. С настройками ключевого цвета[`colorkey`] по умолчанию, вы можете получить разные результаты, скорее всего увидите ключевой цвет[`colorkey`] вокруг символов или другие забавные эффекты. Но при правильных настройках, можно получить вполне приемлемый результат.

Смотрите страницу `map` руководства для списка доступных опций.

4.18.3. DXR3/Hollywood+

MPlayer поддерживает аппаратно ускоренное воспроизведение картами Creative DXR3 и Sigma Designs Hollywood Plus. Обе эти карты используют MPEG декодер на чипе em8300 от Sigma Designs

Прежде всего вам потребуются правильно установленные драйвера DXR3/H+, версии 0.12.0 или выше. Драйверы и инструкции по установке могут быть найдены на сайте [DXR3 & Hollywood Plus для Linux configure](#) должен автоматически определить вашу карту, компиляция должна пройти без проблем.

ИСПОЛЬЗОВАНИЕ

`-vo dxr3:prebuf:sync:norm=x:device`

Опция `overlay` активирует оверлей вместо TV-out. Для корректной работы она требует его правильной настройки. Самый простой способ правильно его настроить - сначала запустить `autocal`, затем запустить `mplayer` с драйвером `dxr3` и выключенным оверлеем, потом запустить `dxr3view`. В `dxr3view` вы можете менять настройки оверлея и видеть результат в реальном времени, может быть, в будущем эта возможность будет поддерживаться в MPlayer GUI. После того как оверлей правильно настроен, надобность в `dxr3view` отпадает. Опция `prebuf` включает пребуферинг. Пребуферинг - это возможность чипа em8300, позволяющая ему хранить более одного кадра видео в каждый момент времени. Это значит, что при включенном пребуферинге MPlayer будет пытаться всегда держать буфер заполненным данными. Если у вас медленная машина, то MPlayer будет забирать практически 100% CPU. Это особенно часто случается, если вы воспроизводите чистые MPEG потоки (DVD, SVCD и т.д.), т.к., поскольку MPlayer не должен перекодировать их в MPEG, то он заполняет буфер очень быстро. С пребуферингом воспроизведение видео **значительно** меньше зависит от остальных программ, прибирающих к рукам CPU, он не будет терять кадры, кроме случая, когда приложения захватит CPU на довольно длительное время. При запуске без пребуферинга, em8300 намного более чувствителен к загрузке CPU, так что настоятельно рекомендуется включить `-framedrop` опцию MPlayer для избежания потери синхронизации. `sync` включит алгоритм синхронизации. Пока что это еще экспериментальная возможность. С включенной `sync` возможностью будут постоянно проверяться встроенные часы em8300, и если начинается отклонение от часов MPlayer чип будет сброшен, что приведет к пропуску всех запаздывающих кадров.

`norm=x` установит TV стандарт DXR3 карты без необходимости использования внешних утилит вроде `em8300setup`. Допустимые стандарты: 5 = NTSC, 4 = PAL-60, 3 = PAL. Специальные стандарты 2 (автонастройка, используя PAL/PAL-60) и 1 (автонастройка используя PAL/NTSC) решают какой стандарт использовать на основе частоты кадров. `because they decide which norm to use by looking at the frame rate of the movie.` `norm = 0` (по умолчанию) не изменяет текущий стандарт.

`device` = номер устройства, если их у вас несколько em8300 карт. Любые из этих опций

могут быть опущены.

`:prebuf:sync`, похоже, отлично работает при воспроизведении фильмов MPEG-4 (DivX). Пользователи сообщали о проблемах при использовании `prebuf` опции при воспроизведении файлов MPEG-1/2. Вы можете попробовать запустить программу сначала вообще без указания опций, если же имеете проблемы с синхронизацией или с DVD субтитрами, попробуйте с `:sync`.

`-ao oss:/dev/em8300_ma-X`

Для вывода звука, где *X* - номер устройства (0, если карта одна).

`-af resample=xxxxx`

em8300 не может воспроизводить частоты дискретизации ниже 44100Hz. Если частота ниже 44100Hz, выберите либо 44100Hz, либо 48000Hz в зависимости от того, какая частота ближайшая. Т.е. если фильм использует 22050Hz укажите 44100Hz, т.к. $44100 / 2 = 22050$, если 24000Hz, используйте 48000Hz, т.к. $48000 / 2 = 24000$ и т.д. Это не работает с цифровым выводом звука (`-ac hwac3`).

`-vf lavc`

Для просмотра не-MPEG контента на em8300 (например MPEG-4 (DivX) или RealVideo) вы должны указать MPEG-1 видеоплагин, такой как libavcodec (lavc). Дополнительную информацию о `-vf lavc` смотрите на странице руководства man. На данный момент неизвестно способа установить значение fps для em8300, что означает фиксированную величину, равную 30000/1001 fps, поэтому крайне рекомендуется использовать `-vf lavc=quality:25`, особенно, если используете пребуферинг. Почему 25, а не 30000/1001? Ну, причина в том, что при использовании 30000/1001 изображение начинает слегка дрожать. Причина нам неизвестна. Если вы ставите это значение где-то между 25 и 27 картинка стабилизируется. Пока что мы можем только принять это как факт

`-vf expand=-1:-1:-1:-1:1`

Хотя драйвер DXR3 может помещать некоторый OSD в MPEG-1/2/4 видео, он имеет намного более плохое качество, чем традиционный OSD MPlayer, и имеет несколько проблем при обновлении к тому же. Команда, указанная выше, сначала отконвертирует входное видео в MPEG-4 (это обязательно, извините), затем применит плагин expand, который не будет ничего расширять (-1: по-умолчанию), но наложит нормальное OSD на картинку (это все, что делает "1" в конце строки).

`-ac hwac3`

em8300 поддерживает воспроизведение звука AC-3 (окружающий звук) через цифровой аудио выход карты. Смотрите опцию `-ao oss` выше, она должна использоваться для указания вывода через DXR3 вместо звуковой карты.

4.19. Другое оборудование вывода видео

4.19.1. Zr

Это видеодрайвер (`-vo zr`) для некоторого количества MJPEG карт захвата/воспроизведения (тестировался на DC10+ и Vuz, но также должен работать для LML33, DC10). Драйвер работает, кодируя кадр в JPEG, и отправляя его карте. Для кодирования JPEG используется и требуется библиотека libavcodec. Со специальным режимом *cinerama*, вы можете смотреть фильмы на действительно широком экране, который можно получить, имея два проектора и две MJPEG карты. В зависимости от разрешения и настроек качества, этот драйвер может потребовать существенной мощности CPU, не забывайте указывать `-framedrop`, если ваша машина слишком медленная. Замечание: Мой AMD K6-2 350МГц является (с `-framedrop`) вполне подходящим для просмотра материала размера VCD, и фильмов с уменьшенным разрешением.

Этот драйвер общается с драйвером ядра, доступном на <http://mjpeg.sf.net>, так что сначала должен корректно заработать второй. Наличие MJPEG карты автоматически определяется скриптом `configure`, если этого не происходит, включите принудительное определение при помощи

```
./configure --enable-zr
```

Вывод может управляться несколькими опциями, подробные описания можно найти на странице руководства man, короткий список опций получите, выполнив `mplayer -zrhelр`

Такие вещи как масштабирование и OSD не осуществляются этим драйвером, но могут быть сделаны, используя видеоплагины. Например, предположим, что вы имеете фильм с разрешением 512x272 и хотите просмотреть его в полноэкранном режиме на DC10+. Есть три главные возможности: вы можете отмасштабировать фильм до ширины 768, 384 или 192. По причинам производительности и качества я бы выбрал масштабирование до 384x204, используя быстрый билинейный программный модуль. Командная строка:

```
mplayer -vo zr -sws 0 -vf scale=384:204 movie.avi
```

Обрезка может быть выполнена плагином **crop** и самим драйвером. Предполагая, что фильм слишком широк для отображения на Vuz, и вы хотите использовать **-zrcrop** для уменьшения ширины, то необходимо применить следующую команду:

```
mplayer -vo zr -zrcrop 720x320+80+0 benhur.avi
```

Если вы хотите использовать плагин **crop**, выполните

```
mplayer -vo zr -vf crop=720:320:80:0 benhur.avi
```

Дополнительное указание **-zrcrop** активизирует режим *cinerama*, т.е. вы можете распределить фильм на несколько TV или проекторов для создания большего экрана. Предположим у вас два проектора. Левый подключен к Vuz на **/dev/video1**, а правый подключен к DC10+ на **/dev/video0**. Фильм имеет разрешение 704x288. Также предположим, что вы хотите выводить на правый проектор в черно-белом цвете, а левый должен иметь JPEG кадры качества 10. в этом случае вы должны указать:

```
mplayer -vo zr -zrdev /dev/video0 -zrcrop 352x288+352+0 -zrxdoof 0 -zrbw \
```

```
-zrcrop 352x288+0+0 -zrdev /dev/video1 -zrquality 10 \  
movie.avi
```

Можно видеть, что опции, встречающиеся до второго **-zrcrop**, применяются только к DC10+, а опции после второго применяются только к Vuz. Максимальное количество карт для режима *cinerama* равно четырем, так что вы можете построить 2x2 видеостену.

Наконец важное замечание: Не запускайте и не останавливайте XawTV на устройстве воспроизведения, это может повредить ваш компьютер. Тем не менее, можно прекрасно **СНАЧАЛА** запустить XawTV, **ЗАТЕМ** запустить MPlayer, подождать, пока MPlayer завершит работу и **ЗАТЕМ** остановить XawTV.

4.19.2. Blinkenlights[Мерцающие огни?]

Этот драйвер способен воспроизводить, используя Blinkenlights UDP протокол. Если не знаете, что такое [Blinkenlights](#) или его преемник [Arcade](#), выясните это. Хотя это, возможно, реже всего используемый драйвер, без сомнения, это самая клевая вещь, предлагаемая MPlayer. Просто посмотрите некоторые из [видеороликов документации Blinkenlights](#). На видео Arcade вы можете видеть драйвер видеовывода Blinkenlights в действии в 00:07:50.

4.20. Поддержка TV-выхода

4.20.1. Matrox G400 карты

Под Linux есть два способа получить работающий TV выход на G400:

Важно

инструкции по TV-выходу для Matrox G450/G550, смотрите в следующем разделе!

XFree86

Используя драйвер и HAL модуль, доступный с [сайта Matrox](#). Это даст вам X на TV.

Этот метод не дает ускоренного воспроизведения как под Windows! Второй выход имеет только YUV фреймбуфер, *BES* (Back End Scaler, модуль YUV масштабирования на картах G200/G400/G450/G550) на нем не работает! Драйвер windows как-то это обходит, возможно используя 3D движок для масштабирования, а YUV фреймбуфер - для вывода отмасштабированного изображения. Если вы действительно хотите использовать X, используйте опции **-vo x11 -fs -zoom**, но это будет **МЕДЛЕННО**, и будет иметь включенную защиту от копирования **Macrovision** (можно "обойти" Macrovision используя этот [скрипт на perl](#)).

Framebuffer

Используя **модули matroxfb** в 2.4 ядрах. 2.2 ядра не имеют в этих модулях возможности работы с TV-out, так что для нашего дела непригодны. Вы должны включить ВСЕ matroxfb-относящиеся возможности во время компиляции (кроме MultiHead), и скомпилировать их в **модули**! Вам также необходима задействованная I2C.

1. Войдите в **TVout** и наберите **./compile.sh**. Установите **TVout/matroxset/matroxset** куда-нибудь в ваши PATH.
2. Если вы еще не имеете установленного **fbset**, поместите **TVout/fbset/fbset** куда-нибудь в ваши PATH.
3. Если **con2fb** у вас еще не установлен, поместите **TVout/con2fb/con2fb** куда-нибудь в ваши PATH.
4. Затем войдите в каталог **TVout/** в исходниках MPlayer, и запустите **./modules** от имени root. Ваша консоль из текстового режима переключится в режим фреймбуфера (обратно не получится!).
5. Затем, **ОТРЕДАКТИРУЙТЕ** и запустите скрипт **./matroxtv**. Он покажет вам очень простое меню. Нажмите 2 и Enter. Теперь вы должны иметь одинаковую картинку на мониторе и TV. Если картинка на TV (PAL по-умолчанию) имеет некоторые странные полосы, значит

скрипт не смог корректно установить разрешение (на 640x512 по-умолчанию). Попробуйте другие разрешения из меню и/или поэкспериментируйте с fbset.

6. Йоу. Следующая задача - убрать курсор с tty1 (или где он есть), и выключить гашение экрана. Запустите следующие команды:

```
echo -e '\033[?25l'  
setterm -blank 0  
or  
setterm -cursor off  
setterm -blank 0
```

Возможно вы захотите поместить вышеуказанное в скрипт, добавив очистку экрана. Чтобы вернуть курсор назад

```
echo -e '\033[?25h'  
или  
setterm -cursor on
```

7. Готово. Запускайте воспроизведение
`mplayer -vo mga -fs -screenw 640 -screenh 512 filename`
(Если используете X, переключитесь теперь в matroxfb при помощи, например Ctrl+Alt+F1.) Замените 640 и 512, если установили другое разрешение...
8. **Наслаждайтесь супер-быстрым, супер-навороченным выводом Matrox TV (лучше чем Xv)!**

Создание кабеля Matrox TV-out. Никто не дает никаких гарантий и не несет никакой ответственности за возможное нанесение ущерба, вызванное выполнением инструкций, указанных в этой документации.

Кабель для G400. Четвертый контакт CRTC2 коннектора - это композитный видео сигнал. Земля - шестой, седьмой и восьмой контакты. (информация получена от Balázs Rácz)

Кабель для G450. Первый контакт CRTC2 коннектора - это композитный видео сигнал. Земля - пятый, шестой, седьмой и пятнадцатый (5, 6, 7, 15) контакты. (информация получена от Balázs Kerekes)

4.20.2. Matrox G450/G550 карты

Поддержка TV выхода для этих карт была добавлена недавно, и пока отсутствует в основном ядре. В данный момент **mga_vid** не может быть использован AFAIK, поскольку драйвер G450/G550 работает только в одной конфигурации: первый чип CRTC (с наибольшим количеством возможностей) на первом экране (мониторе), и второй чип CRTC (без **BES** - для объяснения, что такое BES, смотрите раздел о G400 выше) на TV. Так что в настоящий момент вы можете использовать только драйвер вывода *fbdev* программы MPlayer.

Первый CRTC не может быть перенаправлен на второй выход на текущий момент. Автор драйвера ядра matroxfb - Petr Vandrovcs - возможно добавит поддержку для этого, отображая вывод первого CRTC одновременно на два выхода, как в данный момент и рекомендуется для G400, смотрите раздел выше.

Необходимый патч для ядра и детальное HOWTO можно скачать с http://www.bglug.ca/matrox_tvout/

4.20.3. ATI карты

ПРЕАМБУЛА. Сейчас ATI не хочет поддерживать ни один из ее TV-out чипов под Linux, по причине технологии лицензированной ими у Macrovision.

СТАТУС ПОДДЕРЖКИ TV-OUT ДЛЯ КАРТ ATI ПОД LINUX

- **ATI Mach64**: поддерживается [GATOS](#).
- **ASIC Radeon VIVO**: поддерживается [GATOS](#).
- **Radeon** и **Rage128**: поддерживается MPlayer! Смотрите разделы [VESA драйвер](#) и [VIDIX](#).
- **Rage Mobility P/M, Radeon, Rage 128, Mobility M3/M4**: поддерживается [atitvout](#).

Для других карт просто используйте [VESA](#) драйвер, без VIDIX. Конечно, требуется мощный CPU.

Единственная вещь, которую надо сделать - **Иметь TV коннектор подключенным до загрузки вашего PC**, поскольку видео BIOS инициализирует себя только один раз во время POST процедуры.

4.20.4. nVidia

Во-первых, вы ДОЛЖНЫ скачать закрытые драйверы с <http://nvidia.com>. Я не буду описывать процесс установки и настройки, поскольку это выходит за рамки данной документации.

После того, как XFree86, XVideo, и 3D ускорение заработает правильно, отредактируйте раздел Device для вашей карты в файле **XF86Config**, в соответствии с указанным ниже примером (адаптируйте к вашей карте/TV):

Section "Device"

```

Identifier      "GeForce"
VendorName      "ASUS"
BoardName       "nVidia GeForce2/MX 400"
Driver          "nvidia"
#Option         "NvAGP" "1"
Option          "NoLogo"
Option          "CursorShadow" "on"

Option          "TwinView"
Option          "TwinViewOrientation" "Clone"
Option          "MetaModes" "1024x768,640x480"
Option          "ConnectedMonitor" "CRT, TV"
Option          "TVStandard" "PAL-B"
Option          "TVOutFormat" "Composite"

```

EndSection

Конечно же важная часть - это TwinView.

4.20.5. NeoMagic

Чип NeoMagic найден на различных ноутбуках, некоторые из которых оснащаются простым аналоговым TV кодером, некоторые имеют более продвинутый.

- **Чип аналогового кодера:** Сообщалось, что надежный TV выход можно получить, используя `-vo fbdev` или `-vo fbdev2`. Вам требуется иметь `vesafb` скомпилированный в вашем ядре и передать следующие параметры в командной строке ядра: `append="video=vesafb:ywrap,mtrr" vga=791`. Вам следует запустить X, затем переключитесь в консольный режим при помощи, например, `Ctrl+Alt+F1`. Если вы не запустите X до запуска MPlayer в консоли, видео станет медленным и дрожащим[chorpy] (объяснения приветствуются). Залогиньтесь в консоли и запустите следующую команду:

```
clear; mplayer -vo fbdev -zoom -cache 8192 dvd://
```

Теперь вы должны увидеть фильм, запущенный в консольном режиме, заполняющий примерно половину LCD экрана вашего ноутбука. Для переключения в TV нажмите `Fn+F5` три раза. Тестировался на Tecra 8000, 2.6.15 ядре с `vesafb`, ALSA v1.0.10.

- **Chrontel 70xx чип кодирования:** Найден на IBM Thinkpad 390E и, возможно, других Thinkpad или ноутбуках.

Необходимо использовать `-vo vesa:neotv_pal` для PAL или `-vo vesa:neotv_ntsc` для NTSC. Это даст TV выход, работающий в следующих 16 bpp и 8 bpp режимах:

- NTSC 320x240, 640x480 and maybe 800x600 too.
- PAL 320x240, 400x300, 640x480, 800x600.

Режим 512x384 не поддерживается в BIOS. Вы должны масштабировать изображение в другое разрешение для задействования TV выхода. Если вы видите изображение на экране в разрешении 640x480 или 800x600, но не 320x240 или другом меньшем разрешении, вам требуется заменить две таблицы в `vbelib.c`. Смотрите функцию `vbeSetTV` для подробностей. Пожалуйста, свяжитесь автором в этом случае.

Известные проблемы: только VESA, не реализованы различные настройки, такие как яркость, контрастность, уровень черного, фильтр дрожания[flickfilter].

Глава 5. Портинг

5.1. Linux

Основная платформа разработки — это Linux на x86, хотя MPlayer работает под многими другими портами Linux. Бинарные пакеты MPlayer'a доступны из нескольких источников. Тем не менее, **ни один из этих пакетов не поддерживается**. Сообщайте о проблемах их авторам, а не нам.

5.1.1. Debian пакеты

Чтобы создать Debian пакет, выполните следующие команды в каталоге с исходным кодом MPlayer'a:

```
fakeroot debian/rules binary
```

Если вы хотите передать дополнительные опции `configure`, установите соответствующее значение переменной окружения `DEB_BUILD_OPTIONS`. В частности, если хотите поддержку GUI и OSD, укажите:

```
DEB_BUILD_OPTIONS="--enable-gui --enable-menu" fakeroot debian/rules binary
```

Вы также можете передать некоторые переменные в Makefile. Например, если желаете компилировать gcc 3.4 даже если это не основной компилятор:

CC=gcc-3.4 DEB_BUILD_OPTIONS="--enable-gui" fakeroot debian/rules binary

Для очистки дерева исходных текстов воспользуйтесь командой:

```
fakeroot debian/rules clean
```

В качестве root'a Вы затем можете установить **.deb** пакет:

```
dpkg -i ../mplayer_версия.deb
```

Какое-то время Christian Marillat собирал неофициальные Debian пакеты с MPlayer, MEncoder и бинарными кодеками, так что вы можете их скачать (выполнить apt-get) с его [сайта](#).

5.1.2. RPM пакеты

Dominik Mierzejewski поддерживает официальные Fedora Core RPM пакеты MPlayer'a. Они доступны в [репозитории Livna](#).

Mandrake/Mandriva RPM пакеты доступны с [P.L.F.](#) SuSE включала искаленную версию MPlayer'a в дистрибутив. Из последних релизов они убрали эти пакеты. Вы можете взять работающие RPM с [links2linux.de](#).

5.1.3. ARM

MPlayer работает на Linux PDA с ARM процессором, например Sharp Zaurus, Compaq Ipaq. Простейший способ получить MPlayer — это скачать его с пакетных репозиториев [OpenZaurus](#). Если Вы хотите скомпилировать его самостоятельно, обратите внимание на каталоги [mplayer](#) и [libavcodec](#) в корне сборки дистрибутива OpenZaurus. Там всегда найдутся свежий Makefile и патчи, используемые для сборки SVN MPlayer'a вместе с libavcodec. Если Вам нужен GUI, используйте встроенный в xhms

5.2. *BSD

MPlayer работает на всех известных семействах BSD. Существуют портированные[ports]/пакеты сорцов[pkgsrsrcs]/fink/ и т.п. версии MPlayer, которые, наверное, проще использовать, чем просто исходный код.

Чтобы собрать MPlayer, Вам понадобится GNU make (gmake — родной BSD make не будет работать) и свежая версия binutils.

Если MPlayer ругается, что он не может найти **/dev/cdrom** или **/dev/dvd**, создайте соответствующую ссылку:

```
ln -s /dev/Ваше_cdrom_устройство /dev/cdrom
```

Чтобы использовать Win32 DLL'и с MPlayer'ом, Вам необходимо перекомпилировать ядро с "option USER_LDT" (если только у Вас не FreeBSD-CURRENT, где это включено по умолчанию).

5.2.1. FreeBSD

Если Ваш CPU поддерживает SSE, перекомпилируйте ядро с "options CPU_ENABLE_SSE" (необходимо FreeBSD-STABLE или патчи к ядру).

5.2.2. OpenBSD

В связи с ограничениями в различных версиях gas (конфликт настройки адресов и MMX), Вы должны будете компилировать в два шага: сначала убедитесь, что не родной as — первый в Вашем \$PATH и выполните **gmake -k**, затем убедитесь, что будет использоваться родная версия и запустите **gmake**.

Начиная с OpenBSD 3.4 подобный хак больше не нужен.

5.2.3. Darwin

См. секцию [Mac OS](#).

5.3. Коммерческие Unix

MPlayer был портирован на некоторые коммерческие варианты Unix. Поскольку окружения разработки этих систем отличаются от свободных Unix'ов, вам придется самостоятельно произвести некоторые действия, чтобы сборка заработала.

5.3.1. Solaris

MPlayer должен работать под Solaris 2.6 и более новыми версиями. Для звука используйте звуковой драйвер SUN с опцией **-ao sun**.

На **UltraSPARC**'ах, MPlayer использует преимущество их расширения **VIS** (эквивалент MMX), но (в настоящий момент) только в libmpeg2, libvo and libavcodec, но не в mp3lib. Вы сможете просматривать VOB'ы на 400MHz CPU. Вам потребуется установленная [mLib](#).

Предостережение:

- **mediaLib** в данный момент отключена по умолчанию в MPlayer из-за поломанности. Пользователи SPARC, компилировавшие MPlayer с mediaLib сообщали об избытии зелёного оттенка в видео, кодируемом и декодируемом libavcodec. Если хотите, можете включить ее:

```
$ ./configure --enable-mlib
```

Вы делаете это на свой страх и риск, пользователи x86 не жолжны **никогда** использовать mediaLib, поскольку это очень сильно скажется на производительности MPlayer.

Чтобы собрать программу, Вам потребуется GNU make (**gmake**, `/opt/sfw/gmake`), родной Solaris make не будет работать. Типичная ошибка которую Вы будете получать при использовании Solaris make, вместо GNU make:

```
% /usr/ccs/bin/make
```

```
make: Fatal error in reader: Makefile, line 25: Unexpected end of line seen
```

На Solaris SPARC, Вам потребуется GNU C/C++ Compiler; при этом не имеет значения, был ли GNU C/C++ компилятор сконфигурирован с или без GNU ассемблера.

На Solaris x86, Вам потребуются GNU ассемблер и GNU C/C++ компилятор, сконфигурированный, чтобы использовать GNU ассемблер! На x86 платформах код MPlayer'a использует много MMX, SSE и 3DNOW! инструкций, которые Sun'овский ассемблер `/usr/ccs/bin/as` не может скомпилировать.

Скрипт **configure** пытается обнаружить, какой ассемблер используется Вашей командой "gcc" (в том случае, если автоопределение не сработает, используйте опцию `--as=/там/где/у/Вас/установлен/gnu-as`, чтобы сообщить скрипту **configure**, где можно обнаружить GNU "as" на Вашей системе).

Решение общих проблем:

- Сообщения об ошибках **configure** на Solaris x86 системах при использовании GCC без GNU ассемблера:

```
% configure
```

```
...
```

```
Checking assembler (/usr/ccs/bin/as) ... , failed
```

```
Please upgrade(downgrade) binutils to 2.10.1...
```

(Решение: Установите и используйте gcc, сконфигурированный с `--with-as=gas`)

Типичная ошибка при сборке GNU C компилятором, который не использует GNU as:

```
% gmake
```

```
...
```

```
gcc -c -lloader -llibvo -O4 -march=i686 -mcpu=i686 -pipe -ffast-math
```

```
-fomit-frame-pointer -I/usr/local/include -o mplayer.o mplayer.c
```

```
Assembler: mplayer.c
```

```
"(stdin)", line 3567 : Illegal mnemonic
```

```
"(stdin)", line 3567 : Syntax error
```

```
... more "Illegal mnemonic" and "Syntax error" errors ...
```

- MPlayer может сообщить о нарушении сегментации при кодировании и декодировании видео, использующего win32codecs:

```
...
```

```
Trying to force audio codec driver family acm...
```

```
Opening audio decoder: [acm] Win32/ACM decoders
```

```
sysi86(SI86DSCR): Invalid argument
```

```
Couldn't install fs segment, expect segfault
```

MPlayer interrupted by signal 11 in module: init_audio_codec

```
...
```

Это из-за изменений в sysi86() в Solaris 10 и пре-Solaris Nevada b31 релизах. Исправлено в Solaris Nevada b32; тем не менее Sun еще следует портировать исправление обратно на Solaris 10. Проект MPlayer осведомил Sun об этой проблеме и патч в данный момент готовится для Solaris 10. Больше информации об этой ошибке ищите на: http://bugs.opensolaris.org/bugdatabase/view_bug.do?bug_id=6308413.

- В связи с ошибками в Solaris 8, Вы не сможете проигрывать DVD диски, размером больше 4 Гб:
 - Под Solaris 8 x86 драйвер sd(7D) содержит ошибку, проявляющуюся при доступе к дискам, размером > 4 Гб на устройствах с логическим размером блока != DEV_BSIZE (например CD-ROM и DVD диски). Из-за целочисленного 32-х битного переполнения, происходит доступ к дисковому адресу по модулю 4 Гб (<http://groups.yahoo.com/group/solarisonintel/message/22516>). Проблема отсутствует в SPARC версиях Solaris 8.
 - Похожая ошибка существует в коде файловой системы hfs(7FS) (ISO9660), hfs может не поддерживать разделы/диски больше 4 Гб, доступ к данным происходит по модулю 4 Гб (<http://groups.yahoo.com/group/solarisonintel/message/22592>). Проблемы с hfs могут быть исправлены установкой патча 109764-04 (sparc) / 109765-04 (x86).

5.3.2. HP-UX

Joe Page на своей домашней странице держит подробное [HOWTO](#) по MPlayer на HP-UX, написанное Martin Gansser. С этими инструкциями сборка должна работать "прямо из коробки". Следующая информация взята оттуда.

Вам потребуется GCC 3.4.0 или более поздней версии, GNU make версии 3.80 или новее и SDL 1.2.7 или более новый. HP cc не может создать работоспособную программу, предыдущие версии GCC глючат. Для функционирования OpenGL необходимо установить Mesa, после чего должны заработать драйвера вывода видео gl и gl2, хотя, в зависимости от быстродействия CPU, скорость может быть ужасной. GNU esound является хорошей заменой довольно бедной звуковой системе HP-UX.

Произведите сканирование шины SCSI на предмет наличия DVD устройства:

```
# ioscan -fn
```

Class	I	H/W	Path	Driver	S/W	State	H/W Type	Description
...								
ext_bus	1	8/16/5	c720	CLAIMED INTERFACE	Built-in	SCSI		
target	3	8/16/5.2	tgt	CLAIMED DEVICE				
disk	4	8/16/5.2.0	sdisk	CLAIMED DEVICE			PIONEER DVD-ROM DVD-305	 /dev/dsk/c1t2d0 /dev/rdisk/c1t2d0
target	4	8/16/5.7	tgt	CLAIMED DEVICE				
ctl	1	8/16/5.7.0	sctl	CLAIMED DEVICE	Initiator			 /dev/rscsi/c1t7d0 /dev/rscsi/c1t7l0 /dev/scsi/c1t7l0
...								

Вывод показывает, что по адресу 2 шины SCSI находится Pioneer DVD-ROM. Экземпляр карты для аппаратного пути 8/16 равен 1.

Создайте ссылку от сырого устройства к DVD устройству.

```
ln -s /dev/rdisk/c<SCSI bus instance>t<SCSI target ID>d<LUN> /dev/<device>
```

Пример:

```
ln -s /dev/rdisk/c1t2d0 /dev/dvd
```

Далее следуют решения некоторых общих проблем:

- Крах при запуске с таким сообщением об ошибке:
/usr/lib/dld.sl: Unresolved symbol: finite (code) from /usr/local/lib/gcc-lib/hppa2.0n-hp-hpux11.00/3.2/./././libGL.sl
Это значит, что функция .finite(). недоступна в стандартной математической библиотеке HP-UX. Вместо этого используйте .isfinite().. Решение: Используйте последнюю версию Mesa из репозитория.
- Крах при воспроизведении со следующей ошибкой:
/usr/lib/dld.sl: Unresolved symbol: sem_init (code) from /usr/local/lib/libSDL-1.2.sl.0
Решение: Используйте опцию extralibdir программы configure **--extra-ldflags="/usr/lib -lrt"**
- MPlayer вылетает с нарушением сегментации и сообщением вроде этого:
Pid 10166 received a SIGSEGV for stack growth failure.
Possible causes: insufficient memory or swap space, or stack size exceeded maxssiz.
Segmentation fault
Решение: Ядро HP-UX по-умолчанию для каждого процесса имеет размер стека равный 8МВ(?). (11.0 и новые патчи для 10.20 позволяют вам увеличить maxssiz вплоть до 350МВ для 32-х битных программ). Вы должны расширить maxssiz и перекомпилировать ядро (и перезагрузиться). Чтобы сделать это, можно использовать SAM. (Находясь в нем, проверьте параметр maxdsiz на предмет максимального количества данных, которые могут использоваться программами. 64 Мб по умолчанию может хватить или не хватить в зависимости от Ваших приложений.)

5.3.3. AIX

MPlayer успешно собирается на AIX 5.1, 5.2, и 5.3, используя GCC 3.3 или новее. Сборка MPlayer не проверена на AIX 4.3.3 и более ранних. Крайне рекомендуется собирать MPlayer используя GCC 3.4 или старше, и, как минимум, GCC 4.0, если собираете на POWER5.

Убедитесь, что используете GNU make ([/opt/freeware/bin/gmake](#)) для сборки MPlayer, поскольку столкнетесь с проблемами при использовании [/usr/ccs/bin/make](#).

По-прежнему ведется работа над кодом определения CPU. Проверены следующие архитектуры:

- 604e

- POWER3
- POWER4

На следующих архитектурах не проверялось, но должно работать:

- POWER
- POWER2
- POWER5

Вывод звука через Ultimedia Services не поддерживается, т.к. Ultimedia была убрана из AIX 5.1; таким образом, остается единственный вариант: использовать драйвер AIX Open Sound system (OSS) от 4Front Technologies с <http://www.opensound.com/aix.html>. Для некоммерческого использования 4Front Technologies распространяет драйвер OSS под AIX 5.1 бесплатно; несмотря на это, на текущий день нет драйверов вывода звука для AIX 5.2 или 5.3. **drivers for AIX 5.2 or 5.3. Это означает, что сейчас AIX 5.2 и 5.3 несовместимы с выводом звука MPlayer.**

Решения для общих проблем:

- Если вы столкнулись с такой ошибкой **configure**:

```
$ ./configure
```

```
...
```

```
Checking for iconv program ... no
```

```
No working iconv program found, use
```

```
--charset=US-ASCII to continue anyway.
```

```
Messages in the GTK-2 interface will be broken then.
```

Это из-за того, что AIX использует нестандартные имена кодировок; поэтому перекодировка сообщений в данный момент не работает. Решение - использовать:

```
$ ./configure --charset=noconv
```

5.3.4. QNX

Вам нужно скачать и установить SDL для QNX. Затем запустите MPlayer с опциями **-vo sdl:photon** и **-ao sdl:nto**, и все будет работать быстро.

Вывод **-vo x11** будет ещё медленнее, чем под Linux, поскольку под QNX X'ы эмулируются, что ОЧЕНЬ медленно.

5.4. Windows

Да, MPlayer работает под Windows под [Cygwin](#) и [MinGW](#). Пока ещё нет официального GUI, но версия командной строки полностью функциональна. Обратитесь к списку рассылки [MPlayer-cygwin](#) за помощью и дополнительной информацией. Официальные бинарники под Windows могут быть найдены на [странице загрузки](#). Пакеты установки и простые GUI фронтенды доступны из внешних источников, мы собрали их в разделе Windows на [нашей странице проектов](#).

При нежелании использовать командную строку поможет простой трюк: поместите на рабочий стол ссылку, со следующим содержимым в секции execute:

```
c:\путь\к\mplayer.exe %1
```

Это позволит MPlayer воспроизводить любой фильм, который вы перетащите на созданный ярлык. Добавьте **-fs** для полноэкранного режима.

Лучшие результаты получаются при использовании родного DirectX видео вывода (**-vo directx**). Альтернативой является использование OpenGL или SDL, но производительность OpenGL сильно зависит от машины, а SDL на некоторых системах искажает видео или вылетает. Если изображение искажено, попробуйте отключить аппаратное ускорение, указав **-vo directx:noaccel**. Скачайте [файлы заголовков DirectX 7](#), чтобы скомпилировать видео драйвер DirectX. Кроме того, вам потребуется установленный DirectX 7, чтобы работал DirectX видеодрайвер.

VIDIX теперь доступен и под Windows, как **-vo winvidix**, хотя это ещё экспериментально и требует небольшой ручной установки. Скачайте [dhahelper.sys](#) или [dhahelper.sys \(с поддержкой MTRR\)](#) и скопируйте его в каталог **vidix/dhahelperwin** в Вашем дереве исходного кода MPlayer'a. Откройте консоль и перейдите в этот каталог. Теперь наберите

```
gcc -o dhasetup.exe dhasetup.c
```

```
и запустите
```

```
dhasetup.exe install
```

под Администратором. Теперь Вам нужно перезагрузить машину.

Для получения наилучших результатов, MPlayer должен использовать пространство цветов, аппаратно поддерживаемое Вашей видеокарткой. К сожалению, многие графические драйверы под Windows ошибочно сообщают, что некоторые пространства цветов поддерживаются аппаратно. Чтобы найти какие именно, попробуйте

```
mplayer -benchmark -nosound -frames 100 -vf format=colospace movie
```

, где *colorspace* может быть любым пространством цветов из вывода опции `-vf format=fmt=help`. Если Вы найдёте пространство цветов, которое Ваша карта особенно плохо поддерживает, опция `-vf noformat=colorspace` помешает его использованию. Добавьте это в ваш конфигурационный файл, чтобы это пространство цветов больше никогда не использовалось.

Существуют специальные пакеты кодеков для Windows, доступные на нашей [странице загрузки](#), позволяющие воспроизводить форматы, для которых пока нет родной поддержки. Поместите их куда-нибудь в пути или укажите `configure` опцию `--codecsdir=c:/path/to/your/codecs` (или `--codecsdir=/path/to/your/codecs`, но только под Cygwin). У нас были сообщения о том, что Real DLL должны быть доступны пользователю, запускающему MPlayer, для записи, но только на некоторых системах (NT4). Если у Вас проблемы с ними, попробуйте сделать их доступными на запись.

Вы можете воспроизводить VCD, проигрывая `.DAT` или `.MPG` файлы, которые Windows показывает на VCD. Вот как это работает (указывайте букву диска Вашего CD-ROM):

```
mplayer d:/mpegav/avseq01.dat
```

В качестве альтернативы вы можете напрямую воспроизводить VCD дорожки, указав:

```
mplayer vcd://<дорожка> -cdrom-device d:
```

DVDs также работают, укажите `-dvd-device` с буквой Вашего DVD-ROM:

```
mplayer dvd://<title> -dvd-device d::
```

Консоль Cygwin/MinGW весьма медленная. Перенаправление вывода или использование опции `-quiet` улучшает производительность на некоторых системах. Прямой рендеринг (`-dr`) также может помочь. Если воспроизведение прерывисто, попробуйте `-autosync 100`. Если какие-то из этих опций Вам помогут, стоит поместить их в конфигурационный файл.

Примечание

Если у Вас Pentium 4 и Вы заметили крахи при использовании кодеков RealPlayer, попробуйте отключить hyperthreading.

5.4.1. Cygwin

Для компиляции MPlayer требуется запустить Cygwin версии 1.5.0 или старше.

Файлы заголовков DirectX надо распаковать в `/usr/include/` или `/usr/local/include/`.

Вы можете найти инструкции и файлы для запуска SDL под Cygwin на [сайте libsdl](#).

5.4.2. MinGW

Прежде, установка версии MinGW, способной скомпилировать MPlayer, была сложновата, но сейчас все работает с самого начала. Просто установите MinGW 3.1.0 или более новый и MSYS 1.0.9 или старше и укажите постустановщику MSYS, что MinGW установлен.

Распакуйте файлы заголовков DirectX в `/mingw/include/`.

Для поддержки сжатых заголовоков MOV необходима `zlib`, которую MinGW по умолчанию не предоставляет. Сконфигурируйте её, указав `--prefix=/mingw` и установите её до компиляции MPlayer'a.

Полные инструкции по сборке MPlayer и необходимых библиотек могут быть найдены на странице [MPlayer MinGW HOWTO](#).

5.5. Mac OS

MPlayer не работает на Mac OS версий меньше 10, но компилируется "из коробки" на Mac OS X 10.2 и старше. Предпочитаемым компилятором является версия Apple GCC 3.x или более позднего. Вы можете получить начальное окружение для компиляции, установив Apple'овский `Xcode`. Если у вас Mac OS X 10.3.9 или выше и QuickTime 7, можете использовать драйвер видео вывода `corevideo`.

К сожалению, основное окружение не позволяет получить преимущество от всех приятных возможностей MPlayer. В частности, чтобы иметь включенную поддержку OSD, потребуются установленные в системе библиотеки `fontconfig` и `freetype`. В отличие от остальных Unix'ов, таких как Linux и клоны BSD, OS X не имеет поставляющейся с ОС систему управления пакетами.

Есть как минимум два на выбор: [Fink](#) и [MacPorts](#). Они оба предоставляют одинаковый сервис (т.е. огромное количество пакетов для установки, разрешение зависимостей, возможность простой установки/обновления/удаления пакетов и т.д.). Fink предлагает как предкомпилированные бинарные пакеты, так и сборку всего из исходников, в то время как MacPorts предлагает только собирать из исходных текстов. Автор данного руководства выбрал MacPorts исходя из того простого соображения, что его базовая установка легче. Последующие примеры будут основаны на MacPorts.

В частности для компиляции MPlayer с поддержкой OSD:

```
sudo port install pkgconfig
```

Это установит `pkg-config`, который является системой управления флагами компиляции/сборки

библиотек. Скрипт `configure` программы MPlayer использует его для правильного обнаружения библиотек. Тем же способом можно установить `fontconfig`:

```
sudo port install fontconfig
```

Затем можно продолжить, запустив MPlayer'овский `configure` скрипт (задайте переменные окружения `PKG_CONFIG_PATH` и `PATH` так, чтобы `configure` мог найти библиотеки, установленные при помощи MacPorts):

```
PKG_CONFIG_PATH=/opt/local/lib/pkgconfig/ PATH=$PATH:/opt/local/bin/ ./configure
```

5.5.1. MPlayer OS X GUI

Вы можете получить родной GUI для MPlayer вместе с предкомпилированными бинарниками MPlayer для Mac OS X из проекта [MPlayerOSX](#), но предупреждаем: этот проект давно не развивается.

К счастью, MPlayerOSX был подхвачен членом команды MPlayer. Предварительные релизы доступны с нашей [страницы загрузки](#) и скоро ожидается официальный релиз.

Чтобы самостоятельно собрать MPlayerOSX из исходный текстов, вам потребуется `mplayerosx`, `main` и копию `main` SVN модуля, называющегося `main_noaltivec`. `mplayerosx` - это GUI frontend, `main` - это MPlayer, а `main_noaltivec` - это MPlayer собранный без поддержки AltiVec.

Для извлечения модулей из SVN:

```
svn checkout svn://svn.mplayerhq.hu/mplayerosx/trunk/ mplayerosx
```

```
svn checkout svn://svn.mplayerhq.hu/mplayer/trunk/ main
```

Чтобы собрать MPlayerOSX потребуется настроить что-то вроде этого:

```
MPlayer_source_directory
```

```
|
|--->main          (MPlayer Subversion исходники)
|
|--->main_noaltivec (MPlayer Subversion исходники, сконфигурированные с --disable-
altivec)
|
\--->mplayerosx    (MPlayer OS X Subversion исходники)
```

Сначала надо собрать `main` и `main_noaltivec`.

Для начала, чтобы добиться максимальной обратной совместимости, установите переменную окружения:

```
export MACOSX_DEPLOYMENT_TARGET=10.3
```

Затем сконфигурируйте:

Если конфигурируете для G4 или более позднего CPU с поддержкой AltiVec, делайте так:

```
./configure --disable-gl --disable-x11
```

Если конфигурируете для машины с G3 без AltiVec, используйте:

```
./configure --disable-gl --disable-x11 --disable-altivec
```

Вам может потребоваться отредактировать `config.mak` и изменить `-mcpu` и `-mtune` с `74XX` на `G3`.

Продолжайте с

```
make
```

после чего идите в каталог `mplayerosx` и там наберите:

```
make dist
```

Это создаст сжатый архив `.dmg` с котовым к использованию бинарником.

Также можно использовать проект Xcode 2.1; более старый Xcode 1.x больше не работает.

Глава 6. Основы использования MEncoder

Полный список доступных опций MEncoder и примеры смотрите на странице руководства `man`. Ряд наглядных примеров и подробные руководства по использованию отдельных параметров кодирования, можно узнать прочтя [советы по кодированию](#), которые мы собрали из отдельных нитей[threads] рассылки MPlayer-users. В архивах [здесь](#) и, особенно по поводу старых вещей, [здесь](#) найдется множество дискуссий, посвященных всех аспектам и проблемам кодирования при помощи MEncoder.

6.1. Выбор кодеков и формата файлов

Аудио и видео кодеки для кодирования выбираются опциями `-oac` и `-ovc`, соответственно.

Наберите для примера:

```
mencoder -ovc help
```

чтобы получить список всех видео кодеков поддерживаемых версией MEncoder, установленной на вашей машине. Доступны следующие варианты:

Аудио кодеки:

Название кодека	Описание
mp3lame	Кодируйте в VBR, ABR или CBR MP3 с LAME
lavc	Используйте один из кодеков библиотеки libavcodec
faac	FAAC AAC аудио кодер
toolame	MPEG Audio Layer 2 кодер
twolame	MPEG Audio Layer 2 кодер, основанный на tooLAME
pcm	Несжатый PCM звук
copy	Не перекодировать, просто копировать сжатые кадры

Видео кодеки:

Название кодека	Описание
lavc	Используйте один из кодеков библиотеки libavcodec
xvid	Xvid, MPEG-4 Advanced Simple Profile (ASP) кодек
x264	x264, MPEG-4 Advanced Video Coding (AVC), АКА[он же] H.264 кодек
nuv	nuvel видео, используемое некоторыми приложениями реального времени
raw	Несжатые видео кадры
copy	Не перекодировать, просто скопировать сжатые кадры
frameno	Использовался для 3-х проходного кодирования (не рекомендуется)

Формат выходных файлов выбирается опцией **-of**. Наберите:

`mencoder -of help`

чтобы получить список всех форматов, поддерживаемых версией MEncoder, установленного на вашей машине. Доступны следующие варианты:

Форматы файлов:

Название формата	Описание
lavf	Один из форматов, поддерживаемых библиотекой <code>libavformat</code>
avi	'Слоеное' Аудио-Видео
mpeg	MPEG-1 и MPEG-2 PS
rawvideo	сырой видео поток (без мультиплексирования - только видео поток)
rawaudio	сырой аудио поток (без мультиплексирования - только аудио поток)

AVI является родным форматом для MEncoder, что означает наилучшую его поддержку, MEncoder изначально разрабатывался для этого формата. Как замечено выше, другие форматы тоже пригодны, но вы можете столкнуться с проблемами при их использовании.

форматы файлов библиотеки `libavformat`:

Если вы выбрали `libavformat` для мультиплексирования выходного файла (используя **-of lavf**), подходящий формат файла будет определен по расширению выходного файла. Вы можете заставить использовать конкретный формат опцией **format** библиотеки `libavformat`.

название формата <code>libavformat</code>	Описание
mpg	MPEG-1 и MPEG-2 PS
asf	Advanced Streaming Format

название формата libavformat	Описание
avi	'Слоеное' Аудио-Видео
wav	Waveform Аудио
swf	Macromedia Flash
flv	Macromedia Flash видео
rm	RealMedia
au	SUN AU
nut	открытый формат NUT (экспериментальный и пока не полностью соответствующий спецификации)
mov	QuickTime
mp4	MPEG-4 формат
dv	Sony Digital Видео формат
mkv	Открытый аудио/видео контейнер Matroska

Как видите, libavformat позволяет MEncoder мультиплексировать во множество форматов. К сожалению, поскольку MEncoder изначально не разрабатывался для поддержки форматов, отличных от AVI, вам следует относиться к результирующему файлу с определенной долей паранойи. Убедитесь, что в порядке Аудио/видео синхронизация, и файл воспроизводится не только в MPlayer.

Пример 6.1. Кодирование в формат Macromedia Flash

Создание видео Macromedia Flash, подходящего для воспроизведения в веб браузере плагином Macromedia Flash:

```
mencoder input.avi -o output.flv -of lavf \
  -oac mp3lame -lameopts abr:br=56 -srate 22050 -ovc lavc \
  -lavcopts vcodec=flv:vbitrate=500:mbd=2:mv0:trell:v4mv:cbp:last_pred=3
```

6.2. Выбор входного файла или устройства

MEncoder может кодировать из файлов или непосредственно с DVD или VCD диска. Просто укажите имя файла в командной строке для кодирования из файла, или `dvd://номер_ролика` или `vcd://номер_дорожки` для кодирования DVD ролика или VCD дорожки. Если вы уже скопировали DVD на жесткий диск (может воспользоваться утилитой вроде `dvdbackup`, доступной на многих системах), и желаете кодировать из копии, влдует по-прежнему использовать `dvd://` синтаксис, вместе с `-dvd-device` с последующим путем к корню копии DVD. Опции `-dvd-device` и `-cdrom-device` также могут быть использованы для переопределения путей к файлам устройств для чтения прямо с диска, если значения по умолчанию `/dev/dvd` и `/dev/cdrom` не подходят для вашей системы.

При кодировании с DVD, часто бывает желательно выбрать раздел или диапазон разделов для кодирования. для этой цели можно использовать опцию `-chapter`. Например, `-chapter 1-4` будет кодировать только разделы DVD с 1-го по 4-й. Это особенно полезно при кодировании 1400Мб с целью уместить их на 2 CD, так как вы можете разбить фильм точно на границе раздела, вместо середины некоторой сцены.

Если у вас есть поддерживаемая карта TV захвата, вы также можете кодировать с TV входа. Используйте `tv://номер_канала` в качестве имени файла, и опцию `-tv` для настройки различных параметров захвата. DVB вход работает аналогично.

6.3. Двухпроходное кодирование MPEG-4 ("DivX")

Название происходит из того факта, что кодирование файла производится *дважды*. Первое кодирование (дублирующий проход) создает несколько временных файлов (`*.log`) размером в несколько мегабайт, не удаляйте их пока (вы можете удалить AVI или вообще не создавать видеофайл, перенаправив его в `/dev/null`). На втором проходе, с использованием данных о битпотокке из временных файлов, формируется готовый выходной. Получившийся файл будет иметь намного лучшее качество картинки. Если слышите об этом в первый раз, обратитесь к руководствам, которые можно найти в интернет.

Пример 6.2. копирование звуковой дорожки

Кодирование (двухпроходное) второй дорожки DVD в MPEG-4 ("DivX") AVI с копированием звуковой дорожки.

```
mencoder dvd://2 -ovc lavc -lavcopts vcodec=mpeg4:vpas=1 -oac copy -o /dev/null
```

```
mencoder dvd://2 -ovc lavc -lavcopts vcodec=mpeg4:mbd=2:trell:vpass=2 \  
-oac copy -o output.avi
```

Пример 6.3. кодирование звуковой дорожки

Кодирование (в два прохода) DVD в MPEG-4 ("DivX") AVI с кодированием звуковой дорожки в MP3. Будьте аккуратны, используя этот метод, так как в некоторых случаях это может привести к рассинхронизации аудио/видео.

```
mencoder dvd://2 -ovc lavc -lavcopts vcodec=mpeg4:vpass=1 \  
-oac mp3lame -lameopts vbr=3 -o /dev/null
```

```
mencoder dvd://2 -ovc lavc -lavcopts vcodec=mpeg4:mbd=2:trell:vpass=2 \  
-oac mp3lame -lameopts vbr=3 -o output.avi
```

6.4. Кодирование в Sony PSP видео формат

MEncoder поддерживает кодирование в видео формат Sony PSP, но, в зависимости от ревизии программного обеспечения PSP, с различными ограничениями. Можете не беспокоиться, если не будете нарушать следующие ограничения:

- **Битовый поток:** не должен превышать 1500кбит/с, тем не менее, последние версии очень хорошо поддерживали любой битрейт, пока заголовок не требовал черезчур большого значения.
- **Размеры:** ширина и высота PSP видео должна быть кратна 16, а произведение ширина * высота не должно превышать 64000. В некоторых случаях возможно воспроизведение видео большего размера.
- **Звук:** частота дискретизации должна быть 24кГц для MPEG-4, и 48кГц для H.264.

Пример 6.4. Кодирование для PSP

```
mencoder -ofps 30000/1001 -af lavcresample=24000 -vf harddup -oac lavc \  
-ovc lavc -lavcopts aglobal=1:vglobal=1:vcodec=mpeg4:acodec=libfaac \  
-of lavf -lavfopts format=psp \  
input.video -o output.psp
```

Заметьте, что можно задать заголовок видео опцией `-info name=Заголовок_Фильма`.

6.5. Кодирование в MPEG формат

MEncoder может создавать файлы формата MPEG (MPEG-PS). MPEG-1 или MPEG-2 обычно используют по причине кодирования в более ограниченные форматы, такие как SVCD, VCD или DVD. Особые требования для этих форматов описаны в [руководстве по созданию VCD и DVD section](#).

Чтобы сменить формат выходного файла MEncoder, используйте опцию `-of mpeg`.

Пример:

```
mencoder input.avi -of mpeg -ovc lavc -lavcopts  
vcodec=mpeg1video \  
-oac copy other_options -o output.mpg
```

Создается файл MPEG-1 пригодный для воспроизведения на системах с минимальной поддержкой мультимедиа, таких как только что установленные Windows:

```
mencoder input.avi -of mpeg -mpegopts format=mpeg1:tsaf:muxrate=2000 \  
-o output.mpg -oac lavc -lavcopts acodec=mp2:abitrage=224 -ovc lavc \  
-lavcopts vcodec=mpeg1video:vbitrate=1152:keyint=15:mbd=2:aspect=4/3
```

То же, но используя libavformat MPEG мультиплексор:

```
mencoder input.avi -o VCD.mpg -ofps 25 -vf scale=352:288,harddup -of lavf \  
-lavfopts format=mpg -oac lavc -lavcopts acodec=mp2:abitrage=224 -ovc lavc \  
-lavcopts vcodec=mpeg1video:vrc_buf_size=327:keyint=15:vrc_maxrate=1152:vbitrate=1152:vmax_b_frames=0
```

Подсказка:

Если по каким-то причинам видео после второго прохода вас не устраивает, можно снова запустить кодирование с другими значениями битпотока, при условии, что вы сохранили статистику предыдущего прохода. Это возможно, потому что основная задача файла со статистикой - записывать сложность каждого кадра, которая жестко с битпотокком не связана. Следует иметь в виду, что, несмотря на это, лучшее качество получается если значения результирующего битпотока всех проходов не сильно отличаются.

6.6. Масштабирование фильмов

Часто возникает потребность изменить размер изображения у фильма. Причин может быть

много: уменьшение размера файла, пропускная способность сети, и т.д. Большинство производят масштабирование даже при конвертации DVD или SVCD в AVI. Если есть желание провести масштабирование, прочтите раздел [Сохранение пропорций](#).

Процесс масштабирования осуществляется плагином `scale`: `-vf scale=ширина:высота`. Качество может быть установлено опцией `-sws`. Если не указано, MEncoder будет использовать 2: бикубическое.

Использование:

```
mencoder input.mpg -ovc lavc -lavcopts vcodec=mpeg4:mbd=2:trell \
-vf scale=640:480 -o output.avi
```

6.7. копирование потока

MEncoder может обрабатывать входные потоки двумя способами: **кодировать** или **копировать** их. Этот раздел о **копировании**.

- **Видео поток** (опция `-ovc copy`): можно делать классные вещи :) Вроде помещения (не преобразования!) FLI или VIVO или MPEG-1 видео в AVI файл! Конечно, проиграть такие файлы сможет только MPlayer :) И, возможно, никакого реальной пользы в этом нет. Реально: копирование видеопотока может быть полезно, если надо кодировать только аудио поток (например, несжатый PCM в MP3).
- **Аудио поток** (опция `-oac copy`): straightforward. Возможно взять внешний файл (MP3, WAV) и уплотнить[mux] его в выходной поток. Воспользуйтесь опцией `-audiofile имя_файла`, чтобы сделать это.

Использование `-oac copy` для копирования из одного формата в другой может потребовать указания `-fafmttag` для сохранения тэга аудио формата из оригинального файла. Например, если вы преобразовываете NSV файл со звуком AAC в формат AVI, аудио формат будет неверен и должен быть изменен. Чтобы получить список тэгов аудио формата проверьте `codecs.conf`.

Пример:

```
mencoder input.nsv -oac copy -fafmttag 0x706D \
-oac lavc -lavcopts vcodec=mpeg4:mbd=2:trell -o output.avi
```

6.8. Кодирование из нескольких входных файлов изображений (JPEG, PNG, TGA, SGI)

MEncoder может создавать фильмы из одного или более JPEG, PNG, TGA или других файлов изображений. Простым копированием кадров он может создавать MJPEG (Motion JPEG), MPNG (Motion PNG) или MTGA (Motion TGA) файлы.

Разъяснение процесса:

1. MEncoder декодирует изображение(я) с помощью `libjpeg` (при декодировании PNG, он будет использовать `libpng`).
2. MEncoder затем скормливает декодированное изображение выбранному видео компрессору (`DivX4`, `Xvid`, `FFmpeg msmpeg4`, и .т.д).

примеры. The explanation of the `-mf` option is in the man page.

Создание файла MPEG-4 из всех JPEG файлов текущего каталога:

```
mencoder mf://*.jpg -mf w=800:h=600:fps=25:type=jpg -ovc lavc \
```

```
-lavcopts vcodec=mpeg4:mbd=2:trell -oac copy -o output.avi
```

Создание файла MPEG-4 из некоторых JPEG файлов текущего каталога:

```
mencoder mf://frame001.jpg,frame002.jpg -mf w=800:h=600:fps=25:type=jpg \
-oac lavc -lavcopts vcodec=mpeg4:mbd=2:trell -oac copy -o output.avi
```

Создание файла MPEG-4 из явного списка JPEG файлов (`list.txt` в текущем каталоге содержит список файлов, используемых в качестве источника, по одному в строке):

```
mencoder mf://@list.txt -mf w=800:h=600:fps=25:type=jpg \
-oac lavc -lavcopts vcodec=mpeg4:mbd=2:trell -oac copy -o output.avi
```

Вы можете смешивать различные типы изображений независимо от используемого метода — отдельные файлы, маска или файл со списком — при условии, конечно, что они имеют одинаковое разрешение. Так что вы можете, например, взять титульный кадр из PNG файла, а затем поместить слайдшоу из JPEG фотографий.

Создание файла Motion JPEG (MJPEG) из всех JPEG файлов текущего каталога:

```
mencoder mf://*.jpg -mf w=800:h=600:fps=25:type=jpg -ovc copy -oac copy -o output.avi
```

Создание несжатого файла из всех PNG файлов текущего каталога:

```
mencoder mf://*.png -mf w=800:h=600:fps=25:type=png -ovc raw -oac copy -o output.avi
```

Примечание

Ширина должна быть целым числом, кратным 4, это ограничение формата RAW RGB AVI.

Создание файла Motion PNG (MPNG) из всех PNG файлов текущего каталога:

```
mencoder mf://*.png -mf w=800:h=600:fps=25:type=png -ovc copy -oac copy -o output.avi
```

Создание файла Motion TGA (MTGA) из всех TGA файлов текущего каталога:

```
mencoder mf://*.tga -mf w=800:h=600:fps=25:type=tga -ovc copy -oac copy -o output.avi
```

6.9. Извлечение DVD субтитров в файл VOBsub

MEncoder способен извлекать субтитры из DVD в файлы формата VOBsub. Они состоят из пары файлов, оканчивающихся на `.idx` и `.sub` и обычно упакованы в один `.rar` архив. MPlayer может воспроизводить их при помощи опций `-vobsub` и `-vobsubid`.

Вы указываете базовое имя (т.е. без расширения `.idx` или `.sub`) выходных файлов с помощью `-vobsubout` и индекс этих субтитров в результирующем файле при помощи `-vobsuboutindex`.

Если источником является не DVD следует использовать `-ifo` для указания файла `.ifo`, необходимого для создания результирующего `.idx` файла.

Если источником является не DVD и у вас нет `.ifo` файла, используйте опцию `-vobsubid` для указания, какой ид языка следует записать в `.idx` файл.

При каждом запуске субтитры будут добавляться в конец файлов, если `.idx` и `.sub` уже существуют. Так что вам следует удалять их перед началом.

Пример 6.5. копирование двух субтитров из DVD при выполнении двухпроходного кодирования

```
rm subtitles.idx subtitles.sub
```

```
mencoder dvd://1 -oac copy -ovc lavc -lavcopts vcodec=mpeg4:vpass=1 \
```

```
-vobsubout subtitles -vobsuboutindex 0 -sid 2
```

```
mencoder dvd://1 -oac copy -ovc lavc -lavcopts vcodec=mpeg4:mbd=2:trell:vpass=2 \
```

```
-vobsubout subtitles -vobsuboutindex 1 -sid 5
```

Пример 6.6. Копирование французских субтитров из MPEG файла

```
rm subtitles.idx subtitles.sub
```

```
mencoder movie.mpg -ifo movie.ifo -vobsubout subtitles -vobsuboutindex 0 \
```

```
-vobsuboutid fr -sid 1 -nosound -ovc copy
```

6.10. Сохранение пропорций

DVD и SVCD (т.е. MPEG-1/2) файлы содержат информацию о пропорции, которая описывает как проигрыватель должен масштабировать видео поток, чтобы люди не становились яйцеголовыми. (напр.: $480 \times 480 + 4:3 = 640 \times 480$). Хотя при кодировании в AVI (DivX) файлы вы избавлены от этой проблемы, т.к. заголовки AVI не содержат это значение. Масштабирование изображения отвратительно и расточительно, есть лучший путь!

There is

MPEG-4 имеет уникальную возможность: видео поток может хранить требуемые ему пропорции. Да, в точности как MPEG-1/2 (DVD, SVCD) и H.263 файлы. К сожалению, немного проигрывателей кроме MPlayer поддерживают этот MPEG-4 атрибут.

Эта возможность может использоваться только с libavcodec'овским mpeg4 кодеком. Имейте в виду: хотя MPlayer корректно воспроизведет файл, другие проигрыватели могут использовать неверные пропорции.

Вы серьезно должны обрезать черные полосы выше и ниже изображения. Смотрите страницу руководства man по использованию cropdetect и crop плагинов.

Использование:

```
mencoder sample-svcd.mpg -vf crop=714:548:0:14 -oac copy -ovc lavc \
```

```
-lavcopts vcodec=mpeg4:mbd=2:trell:autoaspect -o output.avi
```

Глава 7. Кодирование с MEncoder

7.1. Создание высококачественного MPEG-4 ("DivX") рипа из DVD фильма

Одним часто задаваемым вопросом является "Как мне сделать рип самого высокого качества для заданного размера?". Другой вопрос "Как мне создать DVD рип с самым высоким возможным качеством? Я не беспокоюсь о размере файла, мне нужно лишь наилучшее качество."

Последний вопрос, похоже, отчасти неверно сформулирован. В конце концов, если Вы не беспокоитесь о размере файла, почему бы просто не скопировать весь MPEG-2 видео поток с DVD? Конечно, Ваш AVI файл будет занимать около 5GB, но если Вы желаете наилучшее качество и не волнуетесь о размере, то это, несомненно, лучшее решение.

В действительности, причиной, по которой Вы хотите перекодировать DVD в MPEG-4, является именно Ваше **беспокойство** о размере файла.

Сложно дать универсальный рецепт о создании DVD рипа очень высокого качества. Необходимо рассмотреть несколько факторов, и Вы должны понимать эти детали, иначе Вы,

скорее всего, разочаруетесь своими результатами. Ниже мы исследуем некоторые из этих вопросов, а затем рассмотрим пример. Мы предполагаем, что Вы используете libavcodec для кодирования видео, хотя теория также применима и к другим кодекам.

Если это кажется для Вас слишком сложным, то Вам, пожалуй, следует использовать один из многочисленных неплохих фронтендов, указанных в [разделе MEncoder](#) нашей страницы родственных проектов. Так Вы должны получить высококачественные рипы без особых размышлений, поскольку большинство этих утилит разработаны для принятия умных решений за Вас.

7.1.1. Подготовка к кодированию: Идентификация исходного материала и кадровой частоты
Прежде, чем даже задумываться о кодировании фильма, Вам необходимо выполнить некоторые предварительные действия.

Первым и наиболее важным шагом перед кодированием должно быть определение типа содержимого, с которым Вы работаете. Если источником Ваших исходных материалов является DVD или широкоэвещательное/кабельное/спутниковое TV, оно будет содержаться в одном из двух форматов: NTSC для Северной Америки и Японии, PAL для Европы и т.д.. Однако, важно понимать, что это только форматирование для показа на телевидении, и оно часто **не** соответствует исходному формату фильма. Опыт показывает, что NTSC материал существенно более сложен для кодирования, т.к. в нём содержится больше элементов, которые нужно идентифицировать. Для проведения удачного кодирования, Вам необходимо знать исходный формат. Отказ от принятия этого во внимание приведёт к различным дефектам в Вашем кодировании, включая безобразные гребешки (артефакты чересстрочной развёртки) и повторяющиеся или даже потерянные кадры. Кроме ухудшения картинки, артефакты так же уменьшают эффективность кодирования: Вы получите худшее качество на единицу битпотока.

7.1.1.1. Определение кадровой частоты источника

Вот список, содержащий общие типы исходных материалов, где они, преимущественно, встречаются и их свойства:

- **Стандартный фильм:** Производятся для театральных показов на 24 fps [кадр/сек].
- **PAL видео:** Записывается с помощью PAL видеокамеры при 50 полях в секунду. Поле состоит только из чётных или нечётных линий кадра. Телевидение было разработано для обновления этих полей попеременно, что используется как вид дешёвого аналогового сжатия. Человеческий глаз, предположительно, компенсирует это, но однажды поняв чересстрочную развёртку, Вы научитесь видеть её и на TV и Вам больше никогда не понравится телевидение. Два поля **не** составляют целый кадр, поскольку они снимаются с задержкой в 1/50 секунды и, следовательно, не формируют одно изображение, за исключением случая полного отсутствия движения.
- **NTSC видео:** Записывается с помощью NTSC видеокамеры при 60000/1001 полях в секунду, или 60 полях в секунду в эпоху чёрно-белого TV. В других отношениях аналогично PAL.
- **Анимация:** Обычно рисуется на 24 fps, но также существуют разновидности со смешанной кадровой частотой.
- **Компьютерная графика (CG):** Может быть с любой частотой кадров, но некоторые встречаются чаще остальных; 24 и 30 кадров в секунду типичны для NTSC, и 25 fps типично для PAL.
- **Старый фильм:** Различные низкие кадровые частоты.

7.1.1.2. Идентификация исходного материала

Фильмы, состоящие из кадров, называются фильмами с построчной (или прогрессивной) развёрткой, а состоящие из независимых полей — фильмами с чересстрочной развёрткой или просто видео; однако, последний термин двусмысленный.

Из-за дальнейших усложнений, некоторые фильмы будут смесью нескольких, указанных выше.

Наиболее важным различием между всеми этими форматами является то, что одни из них основаны на кадрах, а другие — на полях. **Любой** фильм, подготовленный для просмотра на телевидении (включая DVD), преобразуется в формат, основанный на полях. Различные методы, с помощью которых это может быть сделано, совокупно называются "телесин" (англ. telecine), одним из вариантов которого является отвратительный NTSC "3:2 пулдаун" (англ. pulldown). За исключением случаев, когда формат исходного материала был также основан на полях (и с такой же частотой полей), Вы получите фильм в формате отличном от исходного.

Существует несколько общих типов пулдауна:

- **PAL 2:2 пулдаун:** Наилучший из всех. Каждый кадр показывается за время длительности двух полей путем извлечения чётных и нечётных строк и их

попеременного показа. Если в исходном материале 24 fps, то это ускоряет воспроизведение фильма на 4%.

- **PAL 2:2:2:2:2:2:2:2:2:2:3 пулдаун:** Каждый 12-й кадр показывается за время длительности трёх полей, вместо двух. Это помогает избежать проблемы 4%-го ускорения, но делает обращение процесса существенно более сложным. Такие вещи обычно наблюдаются в музыкальных произведениях, где изменение скорости на 4% существенно повредит музыкальную партитуру.
- **NTSC 3:2 телесин:** Кадры показываются попеременно за время длительности 3-х полей или 2-х полей. Это даёт частоту полей в 2.5 раза больше исходной частоты кадров. Результат также очень незначительно замедляется от 60 до 60000/1001 полей в секунду для поддержания частоты полей NTSC.
- **NTSC 2:2 пулдаун:** Используется для отображения материала с 30 fps на NTSC. Так же мил, как и 2:2 PAL пулдаун.

Так же существуют методы для преобразования между NTSC и PAL видео, но подобные темы выходят за рамки данного руководства. Если Вам попался такой фильм, и Вы хотите кодировать его, лучшим решением будет найти копию в исходном формате. Преобразование между этими двумя форматами вносит большие потери и не может быть точно обращено, так что Ваше кодирование существенно пострадает, если оно делается из преобразованного источника.

Когда видео находится на DVD, последовательные пары полей группируются как кадр, даже если они не предназначены для одновременного отображения. Стандарт MPEG-2, используемый на DVD и цифровом TV предоставляет возможность одновременно кодировать исходные кадры с построчной развёрткой и сохранять число полей, в течении которых кадр должен быть показан, в его заголовке. Если был использован такой метод, фильм часто будет называться как "мягкий телесин", т.к. процесс только указывает DVD-плееру о необходимости применения пулдауна к фильму, не изменяя при этом сам фильм. Этот случай существенно предпочтителен, т.к. он может быть легко обращён (в действительности, проигнорирован) кодером и т.к. он сохраняет максимальное качество. Однако, многие широкоэвещательные и DVD студии не используют надлежащую технологию кодирования и вместо этого производят фильмы с "жёстким телесином", где поля в действительности повторяются в кодированном MPEG-2.

Порядок действия в таких случаях будет описан [позже в данном руководстве](#). Сейчас мы дадим Вам несколько советов по идентификации типа материала, с которым Вы работаете:

Регионы NTSC:

- Если при просмотре Вашего фильма MPlayer выводит, что частота кадров была изменена до 24000/1001 и она никогда не меняется обратно, то это почти наверняка содержимое с построчной развёрткой, которое было подвергнуто "мягкому телесину".
- Если MPlayer отображает попеременные переключения частоты кадров между 24000/1001 и 30000/1001, и Вы иногда видите "гребешки", есть несколько возможностей. Сегменты с 24000/1001 fps почти наверняка являются "мягко телесиненным" содержимым с построчной развёрткой, но части с 30000/1001 fps могут быть как "жёстко телесиненным" содержимым с 24000/1001 fps, так и NTSC видео с 60000/1001 полями в секунду. Используйте два нижеследующих руководства для определения того, с каким случаем Вы имеете дело.
- Если MPlayer никогда не показывает изменения кадровой частоты и каждый отдельный кадр, где есть движение, оказывается гребёнкой, Ваш фильм есть NTSC видео с 60000/1001 полями в секунду.
- Если MPlayer никогда не показывает изменения кадровой частоты и два кадра из каждых пяти оказываются гребёнкой, Ваш фильм представляет собой "жёстко телесиненное" содержимое с 24000/1001 fps.

Регионы PAL:

- Если Вы не видите никакой гребёнки, Ваш фильм есть 2:2 пулдаун.
- Если Вы видите попеременную гребёнку каждые полсекунды, Ваш фильм представляет собой 2:2:2:2:2:2:2:2:2:2:3 пулдаун.
- Если Вы всегда видите гребёнки во время движения, значит Ваш фильм является PAL видео с 50 полями в секунду.

Подсказка:

MPlayer может замедлить воспроизведение фильма с опцией `-speed` или воспроизводить его покадрово. Попробуйте использовать опцию `-speed 0.2` для очень медленного просмотра фильма или нажимайте клавишу "." для воспроизведения одного кадра за раз и идентифицируйте образец, если не можете его увидеть на полной скорости.

7.1.2. Постоянный квантователь в сравнении с многопроходностью

Возможно кодировать Ваш фильм, широко варьируя качество. С современными видеокодерами и небольшим сжатием перед кодированием (уменьшением размера и шумов) возможно достичь очень хорошего качества при размере 700 МБ для 90-110-минутного широкоэкранный фильма. Более того, всё, кроме самых длинных фильмов, может быть кодировано с почти безупречным качеством на 1400 МБ.

Есть три подхода при кодировании видео: постоянный битпоток (CBR), постоянный квантователь и многопроходность (ABR или усреднённый битпоток).

Сложность кадров фильма и, таким образом, число битов, нужных для их сжатия может существенно отличаться от одной сцены к другой. Современные видеокодеры могут подстраиваться под это в процессе работы и варьировать битпоток. Однако, в таких простых режимах как CBR кодеры не знают загруженность битпотока в последующих сценах и т.о. не могут превысить затребованный битпоток для больших промежутков времени. Более совершенные режимы, такие как многопроходный режим, могут учитывать статистику предыдущих проходов; это решает проблему, упомянутую выше.

Замечание:

Большинство кодеков, поддерживающих ABR кодирование, поддерживают только двупроходный режим, в то время как некоторые другие, такие как x264, Xvid и libavcodec поддерживают многопроходность, несколько улучшающую качество на каждом проходе, однако, это улучшение не измеримо и не заметно после 4-го прохода или около того. Поэтому, в данном разделе дву- и многопроходность будут использоваться взаимозаменяемо.

В каждом из этих режимов видеокодек (такой как libavcodec) разбивает видеокадр на макроблоки размером 16x16 пикселей и потом применяет квантователь к каждому макроблоку. Чем меньше квантователь, тем лучше качество и выше битпоток. Метод, используемый видео кодером для определения того, какой квантователь использовать для данного макроблока, варьируется и подлежит тонкой настройке. (Это крайнее упрощение реального процесса, но основная концепция полезна для понимания.)

Когда Вы указываете постоянный битпоток, видеокодек будет кодировать видео, отбрасывая детали столько, сколько необходимо и настолько мало, насколько это возможно с целью оставаться ниже заданного битпотока. Если Вас действительно не волнует размер файла, Вы можете также использовать CBR и указать бесконечный битпоток. (На практике это означает значение, достаточно большое для обозначения отсутствия предела, например, 10000 Кбит.) В результате, без реального ограничения битпотока, кодек использует наименьший возможный квантователь для каждого макроблока (как указано опцией `vqmin` для libavcodec, равной 2 по умолчанию). Как только Вы укажете настолько низкий битпоток, что кодек будет вынужден использовать более высокий квантователь, Вы почти наверняка испортите качество Вашего видео. Чтобы избежать этого, Вам, вероятно, придётся уменьшить размеры Вашего видео, согласно методу, описанному далее в этом руководстве. В общем, Вам следует избегать CBR совсем, если Вы заботитесь о качестве.

С постоянным квантователем кодек использует для всех макроблоков один и тот же квантователь, указанный в опции `vqscale` (для libavcodec). Если Вы хотите рип наивысшего возможного качества, снова не взирая на битпоток, Вы можете использовать `vqscale=2`. Это приведёт к тому же битпотoku и PSNR (пику отношения сигнала к шуму), что и CBR с `vbitrate=бесконечности` и значением по умолчанию `vqmin`, равным 2.

Проблема с постоянным квантованием заключается в том, что кодек использует заданный квантователь вне зависимости от того, требуется это для макроблока или нет. То есть возможно использование большего квантователя для макроблока без ухудшения видимого качества. Зачем тратить биты на излишне низкий квантователь? У Вашего процессора есть столько тактов, сколько есть времени, но имеется лишь ограниченное число битов на жёстком диске.

При двупроходном кодировании первый проход создаст рип фильма так, как будто это был CBR, но сохранит лог свойств для каждого кадра. Эта информация затем будет использована во время второго прохода для принятия интеллектуальных решений о том, какой квантователь следует использовать. Во время быстрого движения или сцен с высокой детализацией с большой вероятностью будут использованы большие квантователи, а во время медленного движения или сцен с низкой детализацией — меньшие. Обычно количество движения играет существенно более важную роль, чем количество деталей.

Если Вы используете `vqscale=2`, то Вы теряете биты. Если Вы используете `vqscale=3`, то Вы не получаете рип наивысшего качества. Предположим, Вы делаете рип DVD, используя `vqscale=3`, результат получается 1800 Кбит. Если Вы сделаете двупроходное кодирование с `vbitrate=1800`, получившееся видео будет обладать **лучшим качеством** для **того же**

битпотока.

После того, как Вы сейчас убедились, что два прохода — это путь к действию, возникает вопрос о том, какой битпоток использовать? Ответ таков, что нет единого ответа. В идеале, Вы хотите выбрать битпоток, при котором достигается наилучший баланс между качеством и размером файла. Здесь возможны вариации в зависимости от исходного видеоматериала.

Если размер не важен, хорошей отправной точкой для рипа очень высокого качества будет 2000 Кбит +/- 200 Кбит. Для видеоматериала с быстрым движением или высокой детализацией или просто если у Вас очень разборчивый глаз, Вы можете использовать 2400 или 2600. Для некоторых DVD Вы не заметите разницы на 1400 Кбит. Хорошей идеей является экспериментирование со сценами на разных битпотоках, чтобы почувствовать разницу.

Если Вашей целью является определённый размер, Вам нужно как-нибудь вычислить битпоток. Но перед этим, Вам нужно знать, сколько места нужно зарезервировать по аудио дорожке(и), так что Вам необходимо сперва извлечь их. Вы можете рассчитать битпоток с помощью следующей формулы: $\text{битпоток} = (\text{конечный_размер_в_МБайт} - \text{размер_звука_в_МБайт}) * 1024 * 1024 / \text{длительность_в_секундах} * 8 / 1000$. Например, для сжатия двухчасового фильма в 702 МБ CD, с 60 МБ аудио дорожкой, битпоток видео должен составлять: $(702 - 60) * 1024 * 1024 / (120 * 60) * 8 / 1000 = 740$ кбит/сек.

7.1.3. Ограничения для эффективного кодирования

Из-за особенностей MPEG-подобного сжатия, существуют различные ограничения, которым Вы должны следовать для достижения максимального качества. MPEG разбивает видео на квадраты 16x16, называемые макроблоками. Каждый макроблок состоит из 4 блоков 8x8 с информацией о люме (интенсивности) и двух блоков 8x8 с информацией о хrome (цвете) половинного разрешения (один для красно-бирюзовой оси и другой для жёлто-голубой оси). Даже если ширина и высота Вашего фильма не кратны 16, кодер всё равно использует нужное количество макроблоков 16x16 для покрытия всей области картинка, дополнительная область будет впустую потрачена. Так что в интересах максимизации качества при фиксированном размере файла, не стоит использовать размеры, не кратные 16.

У большинства DVD также есть определённое подобие чёрных полос на краях. Если Вы их оставите, это может **сильно** повредить качество несколькими путями.

1. MPEG-подобное сжатие очень чувствительно к преобразованиям частотных интервалов, в частности, к дискретному косинусному преобразованию (DCT), которое аналогично преобразованию Фурье. Этот вид сжатия эффективен для представления образов и сглаженных переходов, но у него возникают проблемы с острыми краями. Для кодирования последних Вам нужно гораздо больше битов, а иначе у Вас появится артефакт, известный как ореолы.

Частотные преобразования (DCT) выполняются независимо для каждого макроблока (на самом деле, для каждого блока), так что эта проблема возникает только в случае попадания острого края внутрь блока. Если Ваши чёрные поля возникают точно на границах, кратных 16 пикселям, это не проблема. Однако, чёрные полосы на DVD редко хорошо расположены, так что на практике Вам всегда придётся усекавать стороны для избежания этих проблем.

В дополнение к преобразованиям частотных интервалов, MPEG-подобное сжатие использует векторы движения для отображения изменений от одного кадра к другому. Векторы движения, естественно, работают существенно менее эффективно для новых объектов, идущих от краёв картинка, поскольку они отсутствуют в предыдущих кадрах. Пока картинка простирается вплоть до края кодируемой области, у векторов движения не возникает проблем с движением объектов за пределы картинка. Однако, при наличии черных полей могут возникнуть проблемы:

2. Для каждого макроблока MPEG-подобное сжатие сохраняет вектор, определяющий какая часть предыдущего кадра должна быть скопирована в этот макроблок как основа для предсказания следующего кадра. Кодированию подлежит только оставшаяся разность. Если макроблок простирается до края картинка и содержит часть чёрной полосы, то векторы движения других частей картинка переписут чёрную полосу. Это означает, что много битов нужно потратить либо на повторное чернение переписанной полосы, либо (что более вероятно) вектор движения не будет использован вовсе и все изменения для этого макроблока будут явно кодированы. Так или иначе, эффективность кодирования существенно уменьшается.

Ещё раз, эта проблема возникает только в случае, если чёрные полосы не укладываются в границы, кратные 16.

3. Наконец, предположим, что у нас есть находящийся внутри картинка макроблок и объект движется в этот блок от края изображения. MPEG-подобное кодирование не

может сказать "скопируй ту часть, что внутри картинки, но не чёрную полосу". Так что чёрная полоса также будет скопирована внутрь, в результате чего масса битов будет потрачена на кодирование части изображения, которое должно быть на месте полосы.

Для случаев, когда всё изображение движется к краю кодируемой области, у MPEG есть специальные оптимизации для повторяющегося копирования пикселей к краю картинки, когда вектор движения идёт извне области кодирования. Эта возможность становится бесполезной, если у фильма есть чёрные полосы. В отличии от случаев 1 и 2, выравнивание границ до кратности 16 здесь не поможет.

- Несмотря на то, что границы полностью чёрные и никогда не изменяются, существуют, как минимум, определённые накладные расходы, связанные с наличием большего числа макроблоков.

Благодаря всем этим причинам, рекомендуется полностью урезать чёрные полосы. Более того, если есть области шумов/искажений на краях картинки, то их урезание также поспособствует улучшению качества кодирования. Видеофилы, желающие сохранить оригинал как можно более точно, могут возражать против такого усечения; но если Вы не планируете кодировать при постоянном квантователе, качество, полученное при усечении, существенно превысит потери информации на краях.

7.1.4. Усечение и масштабирование

Вспомните из предыдущего раздела, что конечный размер картинки, подлежащей кодированию, должен быть кратен 16 (как высота, так и ширина). Это может быть достигнуто усечением, масштабированием или комбинацией того и другого.

Есть несколько рекомендаций для усечения, которым необходимо следовать для избежания повреждения фильма. Обычный формат YUV, 4:2:0, сохраняет цветность (информацию о цвете) половинной дискретизации, т.е. цветность сохраняется в два раза реже в каждом направлении, чем яркость (информация об интенсивности). Рассмотрите следующую диаграмму, где L обозначает точки дискретизации яркости и C — цветности.

L	L	L	L	L	L	L	L
C		C		C		C	
L	L	L	L	L	L	L	L
L	L	L	L	L	L	L	L
C		C		C		C	
L	L	L	L	L	L	L	L

Как Вы видите, строки и столбцы изображения естественным образом идут в парах. Поэтому смещения и размеры усечения *должны* быть чётными числами. Иначе цветность перестанет правильно соответствовать яркости. Теоретически возможно усечение с нечётными смещениями, но оно потребует переэквантования цветности, что потенциально является операцией с потерей качества и не поддерживается фильтром усечения сторон crop.

Более того, видео с чересстрочной развёрткой дискретизируется следующим образом:

Верхнее поле								Нижнее поле							
L	L	L	L	L	L	L	L								
C		C		C		C									
								L	L	L	L	L	L	L	L
L	L	L	L	L	L	L	L								
								C		C		C		C	
								L	L	L	L	L	L	L	L
L	L	L	L	L	L	L	L								
C		C		C		C									
								L	L	L	L	L	L	L	L
L	L	L	L	L	L	L	L								
								C		C		C		C	
								L	L	L	L	L	L	L	L

Как Вы видите, структура повторяется только после 4 строк. Так что для чересстрочного видео Ваше u-смещение и высота усечения должны быть кратны 4.

Естественные разрешения DVD составляют 720x480 для NTSC и 720x576 для PAL, но существует флаг соотношения сторон, который указывает является ли видео полноэкранным (4:3) или широкоэкранным (16:9). Многие (если не большинство) широкоэкранных DVD не точно соответствуют формату 16:9 и могут быть как 1.85:1, так и 2.35:1 (кинематографический формат). Это означает, что в видео будут чёрные полосы, которые нужно усечь.

MPlayer предоставляет фильтр обнаружения усечения, который определяет прямоугольник, до которого нужно усечь (`-vf cropdetect`). Запустите MPlayer с `-vf cropdetect` и он выдаст настройки усечения для удаления полей. С целью получения точных параметров усечения, Вы должны проигрывать фильм достаточно долго для того, чтоб была использована вся область изображения.

Затем проверьте значения, полученные с помощью MPlayer, используя командную строку, выведенную `cropdetect`, и подстройте прямоугольник при необходимости. Фильтр `rectangle` может быть полезен, позволив Вам интерактивно менять прямоугольник усечения для Вашего фильма. Не забывайте следовать указанным выше руководствам по делимости, чтобы не испортить выравнивание цветности.

В ряде случаев масштабирование может быть нежелательным. Масштабирование по вертикальному направлению затруднено для чересстрочного видео, и если Вы хотите сохранить чересстрочность, Вам, как правило, будет необходимо воздерживаться от масштабирования. Если Вы не будете масштабировать, но всё ещё желаете размеры, кратные 16, то Вам придётся проводить излишнее усечение. Не проводите неполное усечение, поскольку чёрные полосы очень плохи для кодирования!

Поскольку MPEG-4 использует макроблоки 16x16, Вы должны убедиться, что каждое измерение кодируемого видео кратно 16; иначе Вы ухудшите качество, особенно на малых битпотоках. Вы можете сделать это, округлив ширину и высоту прямоугольника усечения до ближайшего меньшего целого, кратного 16. Как указано выше, при усечении Вам необходимо увеличить смещение по Y на половину разности старой и новой высоты, так что полученное видео будет братья из центра кадра. И из-за способа дискретизации DVD видео, убедитесь, что смещение есть чётное число. (Фактически, возьмите за правило никогда не использовать нечётные величины для любых параметров усечения или масштабирования видео.) Если Вы беспокоитесь из-за нескольких излишне отброшенных битов, возможно, Вы предпочтёте взамен масштабировать видео. Мы рассмотрим это ниже в нашем примере. В действительности, Вы можете доверить фильтру `cropdetect` сделать для Вас всё вышеупомянутое, т.к. у него есть необязательный параметр округления `round`, равный 16 по умолчанию.

Также будьте осторожны с "полутёмными" пикселями на краях. Убедитесь, что они тоже отрезаются, иначе Вы будете тратить биты, которым есть лучшее применение.

После всего выше сказанного и сделанного, Вы, вероятно, получите видео не точно формата 1:85.1 или 2.35:1, а с чем-то близким к этому. Вы можете вычислить новый коэффициент соотношения сторон вручную, но MEncoder предоставляет опцию для libavcodec, называемую `autoaspect`, которая сделает это для Вас. Ни в коем случае не увеличивайте размер этого видео с целью квадратизации пикселей, если Вы не желаете впустую потратить место на жёстком диске. Масштабирование должно выполняться при воспроизведении, и плеер использует коэффициент соотношения сторон, сохранённый в AVI, для определения правильного разрешения. К сожалению, не все плееры используют эту информацию автомасштабирования, поэтому Вам всё ещё может быть необходимо перемасштабирование.

7.1.5. Выбор разрешения и битпотока

Если Вы не собираетесь кодировать в режиме постоянного квантователя, Вам нужно выбрать битпоток. Понятие битпотока очень просто: это среднее число битов, которые будут использованы для сохранения Вашего фильма, в секунду. Обычно битпоток измеряется в килобитах (1000 бит) в секунду. Размер Вашего фильма на диске есть битпоток, умноженный на длительность фильма, плюс небольшие накладные расходы (см. раздел [контейнер AVI](#) для примера). Остальные параметры, такие как масштабирование, усечение и т.п. **не** изменят размер файла, пока Вы также не измените битпоток!

Битпоток изменяется **не** пропорционально разрешению. То есть файл разрешением 320x240 с 200 кбит/сек не будет того же качества, что этот же фильм разрешением 640x480 и 800 кбит/сек! Для этого есть две причины:

1. **Восприятие:** Вы сильнее замечаете MPEG артефакты, если они больше! Артефакты возникают на масштабе блоков (8x8). Ваш глаз не увидит ошибки в 4800 маленьких блоков так же легко, как и в 1200 больших блоков (предполагая масштабирование обоих фильмов на полный экран).

2. **Теоретическая:** Когда Вы уменьшаете размер изображения, но продолжаете использовать блоки того же размера (8x8) для пространственных частотных преобразований, Вы перемещаете больше данных в высокочастотные полосы. Грубо говоря, каждый пиксель содержит больше деталей, чем раньше. Так что несмотря на то, что Ваша картинка с уменьшенным масштабом содержит 1/4 информации в пространственных направлениях, она всё ещё может содержать большую часть информации в частотных интервалах (предполагая, что высокие частоты были не использованы в оригинальном 640x480 изображении).

Последние руководства рекомендовали выбор битпотока и разрешения, основываясь на приближении "бит на пиксель", но это обычно не верно из-за упомянутых выше причин. Похоже, лучшей оценкой является рост битпотока пропорционально квадратному корню разрешения, так что 320x240 и 400 кбит/сек должно быть сравнимо с 640x480 и 800 кбит/сек. Однако, это не было строго проверено теоретически или эмпирически. Кроме того, из-за существенного отличия фильмов по уровню шума, деталей, количества движения и т.п., тщетно давать общие рекомендации для "битов на длину диагонали" (аналог битов на пиксель, используя квадратный корень).

Таким образом, мы обсудили сложность выбора битпотока и разрешения.

7.1.5.1. Расчёт разрешения

Следующие шаги помогут Вам рассчитать разрешение для Вашего кодирования без слишком сильного искажения видео, учитывая несколько видов информации об исходном видео. Прежде всего, Вам необходимо рассчитать коэффициент соотношения сторон для кодированного видео: $ARc = (Wc \times (ARa / PRdvd)) / Hc$

где:

- Wc и Hc — ширина и высота усечённого видео,
- ARa — коэффициент соотношения сторон изображения, обычно 4/3 или 16/9,
- $PRdvd$ — отношение пикселей DVD, что равно 1.25=(720/576) для PAL DVD и 1.5=(720/480) для NTSC DVD.

Затем Вы можете рассчитать разрешение по X и Y, согласно определённому фактору качества сжатия (CQ): $ResY = INT(SQRT(1000 * \text{Битпоток} / 25 / ARc / CQ)) / 16 * 16$ и $ResX = INT(ResY * ARc / 16) * 16$.

Хорошо, но что такое CQ? CQ соответствует числу битов на пиксель и на кадр для кодирования. Грубо говоря, чем больше CQ, тем меньше вероятность увидеть артефакты кодирования. Однако, если у Вас есть заданный размер для Вашего фильма (например, 1 или 2 CD), есть ограниченное общее число битов, которые Вы можете потратить; поэтому важно найти хороший компромисс между сжимаемостью и качеством.

CQ зависит от битпотока, эффективности видеокodeка и разрешения фильма. Обычно, в целях увеличения CQ, Вам нужно будет уменьшить размер фильма, при условии, что битпоток, вычисленный как функция конечного размера, и длина фильма постоянны. С MPEG-4 ASP кодеками, такими как Xvid и libavcodec, CQ меньше 0.18 обычно приводит к изображению с большим числом сегментов "квадратиками", из-за недостаточного числа битов для кодирования информации в каждом макроблоке. (MPEG4, как и многие другие кодеки, группирует пиксели в блоки по несколько пикселей для сжатия изображения; если битов не хватает, границы этих блоков заметны.) Следовательно, благоразумно выбрать CQ в диапазоне от 0.20 до 0.22 для рипа на 1 CD и 0.26-0.28 для рипа на 2 CD при использовании стандартных опций кодирования. Более продвинутое опции кодирования, такие как указанные для [libavcodec](#) и [Xvid](#) должны сделать возможным получение того же качества с CQ в диапазоне от 0.18 до 0.20 для рипа на 1 CD и 0.24-0.26 для рипа на 2 CD. Используя MPEG-4 AVC кодеки, такие как x264, Вы можете использовать CQ в диапазоне от 0.14 до 0.16 со стандартными опциями кодирования и должны суметь достичь таких низких значений, как 0.10 — 0.12 с помощью [продвинутых опций кодирования x264](#).

Пожалуйста, обратите внимание, что CQ — лишь показательная величина, т.к. она зависит от кодируемого содержимого; CQ 0.18 может хорошо смотреться для Бергмана (Bergman), в отличии от такого фильма как Матрица (The Matrix), содержащего много сцен с быстрым движением. С другой стороны, бесполезно увеличивать CQ выше 0.30, т.к. Вы будете тратить биты без заметного увеличения качества. Так же обратите внимание, что, как было указано выше в данном руководстве, фильмам с низким разрешением (например, по сравнению с DVD) необходим более высокий CQ для того, чтоб они выглядели хорошо.

7.1.6. Фильтрация

Изучение использования видео фильтров MEncoder важно для получения хороших результатов кодирования. Вся обработка видео выполняется посредством фильтров: усечение, масштабирование, подстройка цвета, удаление шума, увеличение чёткости, деинтерлейс

(преобразование видео из чересстрочной развёртки в построчную), телесин, обратный телесин и удаление блочной сегментации — и это лишь некоторые из них. Вместе с огромным количеством поддерживаемых входных форматов, разнообразие фильтров, доступных в MEncoder, является одним из его основных достоинств над другими аналогичными программами.

Фильтры загружаются в цепочку с помощью опции -vf:

-vf фильтр1=опции,фильтр2=опции,...

Большинство фильтров используют численные значения опций, разделённые двоеточиями, но синтаксис этих параметров различается у разных фильтров, так что читайте мануал для детальной информации о фильтрах, которые Вы желаете использовать.

Фильтры действуют на видео в порядке их загрузки. Например, следующая цепочка:

-vf crop=688:464:12:4,scale=640:464

сперва усечёт область изображения до 688x464 с верхним левым углом (12,4), а затем масштабирует результат до 640x464.

Некоторые фильтры нужно загружать в начале цепочки фильтров (или рядом с ним) с целью получения преимущества от использования информации после видеодекодера, которая будет потеряна или искажена другими фильтрами. Важнейшими примерами являются: **pp** (постобработка, только при выполнении операций удаления блочной сегментации (deblocking) или увеличения чёткости краёв (deringing)), **spp** (другой фильтр постобработки, служащий для удаления артефактов MPEG), **pullup** (обратный телесин), и **softpulldown** (для преобразования мягкого телесина в жёсткий).

В общем случае, Вам следует делать настолько мало фильтрации, насколько это возможно, для того чтоб остаться близко к оригинальному DVD источнику. Усечение часто необходимо (как описано выше), но избегайте масштабирования видео. Несмотря на то, что уменьшение размера иногда предпочтительно использованию бОльших квантователей, нужно избегать и того, и другого: помните, что мы с самого начала решили обменять биты на качество.

Также не корректируйте гамму, контрастность, яркость и т.п.. То, что хорошо выглядит на Вашем мониторе, может плохо выглядеть на других. Эти коррекции должны выполняться только при воспроизведении.

Однако, есть одна вещь, которую Вы, быть может, захотите сделать — это пропустить видео через очень слабый фильтр удаления шумов, такой как **-vf hqdn3d=2:1:2**. Ещё раз, причиной этому является то, что этим битам можно найти лучшее применение: зачем тратить их, кодируя шум, если Вы просто можете вернуть этот шум в процессе воспроизведения? Увеличение параметров для **hqdn3d** дополнительно улучшит сжимаемость, но увеличив значения слишком сильно, Вы рискуете ухудшить различимость изображения. Рекомендованные выше значения (**2:1:2**) слегка консервативны; не бойтесь экспериментировать с более высокими значениями и самостоятельно оценивать результаты.

7.1.7. Чересстрочная развёртка и телесин

Почти все фильмы снимаются при 24 fps [кадр/сек]. Поскольку в NTSC используется 30000/1001 fps, нужно выполнить некоторую обработку для такого 24 fps видео, чтобы оно корректно воспроизводилось при кадровой частоте NTSC. Этот процесс называется 3:2 пулдаун, обычно называемый телесин (поскольку пулдаун часто применяется в процессе показа телевизионного фильма, англ. telecine); и в упрощенном описании это работает путём замедления фильма до 24000/1001 fps и повтора каждого четвёртого кадра.

Однако, никакой специальной обработки не выполняется для видео на PAL DVD, которое воспроизводится при 25 fps. (Технически PAL может быть подверженным телесину, называемому 2:2 пулдаун, но на практике это не применяется). 24 fps фильм просто проигрывается на 25 fps. В результате фильм воспроизводится слегка быстрее, но если Вы не пришелец, то, вероятно, не заметите разницы. У большинства PAL DVD аудио скорректировано по высоте звука так, что, воспроизводясь при 25 fps, оно звучит нормально, несмотря на то, что аудиодорожка (и, следовательно, весь фильм) проигрываются на 4% быстрее, чем NTSC DVD.

Поскольку видео на PAL DVD не переделывается, Вам не стоит беспокоиться о частоте кадров. У источника 25 fps и у Вашего рипа будет 25 fps. Однако, если Вы делаете рип NTSC DVD фильма, Вам, быть может, придётся выполнить обратный телесин.

Для фильмов, снятых на 24 fps, видео на NTSC DVD идёт либо с телесином 30000/1001, либо с построчной развёрткой 24000/1001 fps и предназначается для телесина на лету с помощью DVD плеера. С другой стороны, TV сериалы идут обычно только с чересстрочной развёрткой, но без телесина. Это не строгое правило: есть сериалы с чересстрочной развёрткой (например, Баффи, Убийца Вампиров [Buffy the Vampire Slayer]), в то время как другие представляют собой смесь построчной и чересстрочной развёртки (такие как Ангел [Angel] или

24).

Настоятельно рекомендуется прочитать раздел о [работе с телесиноном и чересстрочной развёрткой в NTSC DVD](#) для изучения способов обработки в разных ситуациях.

Однако, если Вы преимущественно делаете рипы фильмов, Вы, скорее всего, имеете дело с 24 fps видео либо с построчной развёрткой, либо с подвергнутым телесину; в последнем случае Вы можете использовать `pullup` фильтр: `-vf pullup,softskip`.

7.1.8. Кодирование чересстрочного видео

Если Вы желаете кодировать фильм с чересстрочной развёрткой (NTSC или PAL видео), Вам нужно решить, будете ли Вы его преобразовывать в построчную развёртку или нет. Хотя такое преобразование (деинтерлейс) сделает Ваш фильм пригодным для дисплеев с построчной развёрткой, таких как компьютерные мониторы и проекторы, это будет иметь свою цену: частота полей уменьшится вдвое от 50 или 60000/1001 до 25 или 30000/1001 поля в секунду, и примерно половина информации в Вашем фильме будет потеряна в сценах со значительным движением.

Поэтому, если Вы кодируете для высококачественных архивных целей, не рекомендуется делать деинтерлейс. Вы всегда можете преобразовать развёртку фильма в процессе воспроизведения (при воспроизведении на устройствах с построчной развёрткой). Мощность современных компьютеров вынуждает плееры использовать фильтр деинтерлейса, что слегка ухудшает качество изображения. Но плееры будущего будут способны имитировать дисплей TV с чересстрочной развёрткой, выполняя деинтерлейс на полной частоте полей и интерполируя 50 или 60000/1001 кадров в секунду для чересстрочного видео.

С чересстрочным видео нужно работать особым образом:

1. Высота усечения и смещение по оси у должны быть кратны 4.
2. Любое вертикальное масштабирование должно выполняться в режиме чересстрочной развёртки.
3. Фильтры постобработки и удаления шума могут не работать как ожидается, только если Вы особо не позаботитесь об их применении к одному полю за раз, иначе они могут повредить видео при неверном использовании.

Учитывая вышесказанное, вот наш первый пример:

```
mencoder захват.avi -mc 0 -oac lavc -ovc lavc -lavcopts \
vcodec=mpeg2video:vbitrate=6000:ilme:ildct:acodec=mp2:abitrage=224
```

Обратите внимание на опции `ilme` и `ildct`.

7.1.9. Замечания об аудио/видео синхронизации

Алгоритмы аудио/видео (A/V) синхронизации MEncoder были разработаны с целью восстановления файлов с повреждённой синхронизацией. Однако, в ряде случаев они могут привести к ненужному пропуску или повторению кадров и, возможно, к лёгкой A/V рассинхронизации корректных входных данных (конечно, проблемы A/V синхронизации возникают только при обработке или копировании аудиотрека при кодировании видео, что настоятельно рекомендуется). Поэтому Вы можете переключиться на базовую A/V синхронизацию с помощью опции `-mc 0` или разместить это в конфигурационном файле `~/mplayer/mencoder`, если Вы работаете только с хорошими источниками (DVD, TV-захват, высококачественные MPEG-4 рипы и т.п.), а не с повреждёнными файлами ASF/RM/MOV.

Если Вы хотите дополнительно защититься от странных пропусков и повторений кадров, Вы можете одновременно использовать опции `-mc 0` и `-noskip`. Это предотвратит *любую* A/V коррекцию, и будет копировать кадры один в один, так что Вы не сможете это использовать, если будете применять какие-либо фильтры, которые непредсказуемо добавляют или отбрасывают кадры, либо если у Вашего входного файла переменный битопоток! Поэтому использование `-noskip` в общем случае не рекомендуется.

Сообщалось о том, что так называемое трёхпроходное аудиокодирование, поддерживаемое MEncoder, вызывало A/V рассинхронизацию. Это наверняка произойдёт при использовании совместно с некоторыми фильтрами, поэтому сейчас *не* рекомендуется использовать трёхпроходный аудио режим. Эта возможность оставлена только для совместимости и для опытных пользователей, понимающих когда это безопасно, а когда нет. Если Вы ранее никогда не слышали о трёхпроходном режиме, забудьте даже о том, что мы его упоминали! Также были сообщения об A/V рассинхронизации при кодировании со стандартного ввода (stdin) с помощью MEncoder. Не делайте этого! Всегда взамен используйте файл или CD/DVD и т.п. устройство.

7.1.10. Выбор видеокодека

То, какой видеокодек лучше выбрать, зависит от нескольких факторов, таких как размер, качество, устойчивость к ошибкам, практичность и распространённость, многие из которых сильно зависят от личных предпочтений и технических ограничений.

- Эффективность сжатия:** Достаточно очевидно, что большинство кодеков нового поколения разработаны для увеличения качества и степени сжатия. Поэтому, авторы данного руководства и многие другие люди полагают, что Вы не можете ошибиться [\[1\]](#), выбирая MPEG-4 AVC кодеки (например, x264) вместо таких MPEG-4 ASP кодеков, как libavcodec MPEG-4 или Xvid. (Опытные разработчики кодеков могут быть заинтересованы в ознакомлении с точкой зрения Михаэля Нидермайера (Michael Niedermayer) "[почему MPEG4-ASP отстой](#)".) Аналогично, Вы должны получить лучшее качество с MPEG-4 ASP, по сравнению с MPEG-2 кодеками. Однако, новые кодеки, находящиеся в интенсивной разработке, могут страдать от ещё не замеченных ошибок, которые могут испортить кодирование. Просто это плата за использование передовых технологий. Более существенно то, что для начала использования нового кодека необходимо потратить время на изучение его опций так, чтобы Вы знали, что нужно подстраивать для достижения заданного качества изображения.
- Аппаратная совместимость:** Обычно необходимо длительное время для включения поддержки последних видеокодеков в автономные видеоплееры. В итоге, большинство поддерживает только MPEG-1 (наподобие VCD, XVCD и KVCD), MPEG-2 (например, DVD, SVCD и KVCD) и MPEG-4 ASP (например, DivX, libavcodec LMP4 и Xvid) (Осторожно: обычно поддерживаются не все возможности MPEG-4 ASP). Пожалуйста, обратитесь к технической спецификации Вашего плеера (если она доступна) или к гугл (google) для детальной информации.
- Лучшее соотношение качества и времени кодирования:** Кодеки, уже использующиеся определённое время (например, libavcodec MPEG-4 и Xvid) обычно сильно оптимизированы всевозможными остроумными алгоритмами и ассемблерным SIMD кодом. Поэтому они обладают тенденцией достижения лучшего соотношения качества к времени кодирования. Однако, у них могут быть некоторые очень продвинутые опции, которые, будучи включенными, сделают кодирование очень медленным ради несущественного выигрыша. Если Вам нужна высокая скорость, примерно придерживайтесь настроек видеокодека по умолчанию (хотя Вам стоит попробовать другие опции, упоминаемые в иных разделах данного руководства). Вы так же можете рассмотреть вариант использования многопоточного кодека, хотя это полезно только для пользователей машин с несколькими процессорами. libavcodec MPEG-4 позволяет это, но выигрыш в скорости ограничен и есть небольшой отрицательный эффект для качества картинки. Многопоточное кодирование Xvid, включаемое опцией `threads`, может использоваться для ускорения кодирования (на примерно 40-60% в типичных случаях) с небольшим ухудшением картинки или вообще без него. x264 также позволяет многопоточное кодирование, что обычно ускоряет процесс на 94% для каждого CPU ядра с уменьшением PSNR от 0.005 дБ до 0.01 дБ при типичных настройках.
- Личные предпочтения:** Здесь всё становится почти неразумным: из-за тех же причин, по которым одни придерживаются DivX 3 в течении лет, в то время как новые кодеки уже творят чудеса, другие люди предпочитают Xvid или libavcodec MPEG-4 использованию x264. Вам нужно принимать решение самостоятельно; не слушайте советов людей, признающих только один кодек. Сделайте несколько образцов клипов из искомым источников и сравните разные опции кодирования и кодеки, с целью выбора того, что Вам наиболее подходит. Лучший кодек — это тот, которым Вы сами овладели, и который выглядит лучше всего для Ваших глаз на Вашем дисплее [\[2\]!](#)

Пожалуйста, обратитесь к разделу [выбор кодеков и форматов контейнера](#) для получения списка поддерживаемых кодеков.

7.1.11. Аудио

Аудио — это гораздо более простая проблема: если Вы беспокоитесь о качестве, просто оставьте всё как есть. Даже потоки AC-3 5.1 не более чем 448 Кбит/с и они стоят каждого бита. Вы можете соблазниться перекодированием аудио в высококачественный Vorbis (он же ogg формат), но лишь то, что у Вас сегодня нет A/V приёмника для пропускания AC-3, не означает, что у Вас не будет его завтра. Для жизнеспособности Ваших DVD рипов в будущем, сохраняйте поток AC-3. Вы можете сохранить поток AC-3, копируя его непосредственно в видеопоток [в процессе кодирования](#). Вы также можете извлечь AC-3 поток с целью мультимплексирования его в контейнеры наподобие NUT или Matroska (Матрёшка).

```
mp1ayer файл_источника.vob -aid 129 -dumpaudio -dumpfile звук.ac3
```

сохранит в файл `звук.ac3` аудиодорожку с номером 129 из файла `файл_источника.vob` (Обратите внимание: DVD VOB файлы обычно используют нумерацию аудио, отличную от стандартной, что означает, что аудиодорожка VOB 129 — это вторая аудиодорожка файла).

Но иногда у Вас действительно нет иного выбора, чем далее сжимать звук для того, чтоб больше битов могло быть потрачено на видео. Большинство людей предпочитают сжимать звук с помощью MP3 или Vorbis аудиокодеков. Последний является очень эффективным, но MP3 лучше поддерживается аппаратными плеерами, хотя эта тенденция меняется.

Не используйте `-nosound` при кодировании файла с аудио, даже если позже Вы будете отдельно кодировать и мультимплексировать аудио. Хотя это может работать в идеальных случаях, использование `-nosound` обычно скрывает ряд проблем в Ваших настройках кодирования в командной строке. Другими словами, наличие звуковой дорожки в процессе кодирования гарантирует Вам, что в случае отсутствия сообщений, подобных «Слишком много аудиопакетов в буфере», у Вас будет получена правильная синхронизация.

Вам необходимо обработать звук с помощью MEncoder. Например, Вы можете копировать исходную звуковую дорожку в процессе кодирования с помощью `-oac copy` или преобразовать её в "лёгкий" 4 кГц моно WAV PCM с помощью `-oac pcm -channels 1 -srate 4000`. Иначе, в ряде случаев, будет создаваться видео файл, рассинхронизированный с аудио. Такие случаи происходят, когда число кадров видео исходного файла не совпадает с полной длиной кадров аудио, или когда были разрывы/сшивания потока, где появились пропущенные или излишние аудиокадры. Правильным решением подобных проблем является вставка тишины или усечение аудио в таких точках. Однако, MPlayer не может это сделать и если Вы демультимплексируете AC-3 аудио и кодируете его отдельным приложением (или создаёте дамп в PCM с помощью MPlayer), сшивания останутся нескорректированными и единственный способ их исправить — пропускать/дублировать видеокадры в местах сшивки. Пока MEncoder видит аудио при кодировании видео, он может выполнять этот пропуск/дублирование (что обычно не вызывает проблем, т.к. происходит при полностью чёрных кадрах или при смене сцен), но если MEncoder не доступно аудио, он просто будет обрабатывать все кадры "как есть" и они не будут совпадать с окончательным аудиопотоком, когда Вы, например, объедините аудио и видео дорожки в Matroska файл.

Прежде всего, Вам необходимо преобразовать DVD звук в WAV файл, который может использоваться аудиокодеком в качестве входных данных. Например:

```
mplayer исходный_файл.vob -ao pcm:file=звук.wav  
-vc dummy -aid 1 -vo null
```

сохранит вторую аудиодорожку из файла `исходный_файл.vob` в файл `звук.wav`. Возможно, Вы захотите нормализовать звук перед кодированием, поскольку аудиодорожки DVD обычно записываются с маленькой громкостью. Вы можете использовать, например, утилиту `normalize`, доступную в большинстве дистрибутивов. Если Вы пользуетесь Window\$, утилита `BeSweet` делает то же самое. Вы можете сжать в Vorbis или MP3. Например:

```
oggenc -q1 звук.wav
```

кодирует `звук.wav` с качеством 1, что примерно эквивалентно 80 Кб/с и является минимальным качеством, при котором Вам нужно кодировать, если Вы заботитесь о качестве. Пожалуйста, обратите внимание, что MEncoder на данный момент не поддерживает мультимплексирование аудиопотоков Vorbis в выходной файл, поскольку он поддерживает только AVI и MPEG контейнеры для выходных файлов, использование каждого из которых может привести к проблемам A/V синхронизации с некоторыми плеерами, в случае когда AVI файл содержит VBR аудиопотоки наподобие Vorbis. Не беспокойтесь, в данном документе будет рассказано как Вы можете это сделать с помощью сторонних программ.

7.1.12. Мультимплексирование

Теперь, после того как Вы кодировали видео, скорее всего, Вы захотите мультимплексировать его с одним или несколькими аудиопотоками в такие видео контейнеры как AVI, MPEG, Matroska или NUT. На данный момент встроенная поддержка вывода аудио и видео в MEncoder есть только для форматов контейнеров MPEG и AVI. Например:

```
mencoder -oac copy -ovc copy -o выходной_фильм.avi \  
-audiofile исходный_звук.mp2 исходное_видео.avi
```

Это объединит видеофайл `исходное_видео.avi` и аудиофайл `исходный_звук.mp2` в AVI файл `выходной_фильм.avi`. Эта команда работает с MPEG-1 слой I, II и III (более известный как MP3) аудио, WAV, а также с некоторыми иными форматами аудио.

MEncoder обладает экспериментальной поддержкой `libavformat` — библиотеки из проекта `FFmpeg`, поддерживающей мультимплексирование и демультимплексирование множества контейнеров. Например:

```
mencoder -oac copy -ovc copy -o выходной_фильм.asf \  
-audiofile исходный_звук.mp2 исходное_видео.avi
```



```
-audiofile исходный_звук.mp2 исходное_видео.avi \  
-of lavf -lavfopts format=asf
```

Это сделает то же самое, что и предыдущий пример, но выходным контейнером будет ASF. Пожалуйста, обратите внимание, что эта поддержка весьма экспериментальна (но становится лучше с каждым днём), и будет работать только в случае компиляции MPlayer с включенной поддержкой libavformat (что означает, что в большинстве случаев бинарная версия из пакетов не будет работать).

7.1.12.1. Улучшение мультимплексирования и надёжности A/V синхронизации

Вы можете столкнуться с некоторыми серьёзными проблемами A/V синхронизации при попытке мультимплексирования Вашего видео с некоторыми аудиодорожками, где, как бы Вы не подбирали задержку аудио, никогда не получается правильная синхронизация. Это может происходить при использовании некоторых видеофильтров, пропускающих или дублирующих некоторые кадры, например фильтров обратного телесина. Настоятельно рекомендуется добавлять видеофильтр **harddup** в конце цепочки фильтров для избежания подобных проблем. Без опции **harddup**, в случае когда MEncoder хочет дублировать кадр, он полагается на то, что мультимплексор расположит отметку в контейнере таким образом, что последний кадр будет повторен для достижения синхронизации без реальной записи кадра. С опцией **harddup**, MEncoder вместо этого просто ещё раз поместит последний кадр в цепочку фильтров. Это означает, что кодер получит *точно* такой же кадр дважды и сожмёт его. Это приведёт у несколько большему файлу, но избавит от проблем при демультимплексировании или ремультимплексировании с другими форматами контейнеров.

Также у Вас может не быть иного выбора, как использовать **harddup** с форматами контейнеров, которые не слишком плотно связаны с MEncoder, например, с форматами, поддерживаемыми с помощью libavformat, которые могут не поддерживать дублирование кадров на уровне контейнера.

7.1.12.2. Ограничения контейнера AVI

Хотя это самый широко распространённый формат контейнера после MPEG-1, он также обладает некоторыми существенными недостатками. Пожалуй, они наиболее очевидны в его избыточности. Для каждой цепочки AVI файла теряется 24 байта на заголовки и индекс. Это приводит к чуть более 5 МБ/час или 1.0-2.5% избыточности для 700 МБ фильма. Это не кажется большим, но может означать разницу между возможностью использования 700 кбит/сек или 714 кбит/сек в случаях, когда каждый бит на счету.

В дополнение к малой эффективности, AVI также обладает следующими серьёзными ограничениями:

1. Может быть сохранено только содержимое с фиксированной частотой кадров. В частности, это особенно ограничивает, когда Ваш исходный материал смешанного содержимого: например, является смесью NTSC видео и киноматериала. В действительности, есть хаки, позволяющие сохранять содержимое с переменным fps в AVI, но они увеличивают (и без того большую) избыточность впятеро или более того и поэтому непрактичны.
2. Аудио в AVI файлах должно быть или с постоянным битпотокком (CBR) или с постоянным размером кадра (т.е. все кадры декодируются в одно и то же число выборок). К сожалению, самый эффективный кодек, Vorbis, не удовлетворяет ни одному из данных требований. Поэтому, если Вы планируете сохранять Ваш фильм в AVI, Вы должны использовать менее эффективный кодек, такой как MP3 или AC-3.

Сказав всё это, отметим, что MEncoder на данный момент не поддерживает вывод с переменным fps или Vorbis кодирование. Поэтому Вы можете не рассматривать всё это как ограничения, если MEncoder — это единственный инструмент, который Вы используете для кодирования. Однако, возможно использовать MEncoder только для кодирования видео и затем использовать внешние утилиты для кодирования аудио и мультимплексирования его в контейнер другого формата.

7.1.12.3. Мультимплексирование в контейнер Matroska (Матрёшка)

Matroska — это свободный, открытый стандарт формата контейнера, нацеленный на предоставление большого количества продвинутых возможностей, которые старые контейнеры (наподобие AVI) не поддерживают. Например, Matroska поддерживает аудиосодержимое с переменным битпотокком (VBR), переменные частоты кадров (VFR), разделы, файловые вложения, код обнаружения ошибок (EDC) и современные A/V кодеки, такие как "Продвинутое Аудио Кодирование" ("Advanced Audio Coding", AAC), "Vorbis" или "MPEG-4 AVC" (H.264), также не поддерживаемые AVI.

Утилиты, необходимые для создания Matroska файлов, сообщаются называются mkvtoolnix, и доступны для большинства Unix платформ, так же как и для Window\$. Поскольку Matroska —

открытый формат, Вы можете найти иные утилиты, которые лучше Вам подходят, но поскольку mkvtoolnix — наиболее общие и поддерживаются самой командой разработчиков Matroska, мы будем обсуждать только их использование.

Возможно, самым простым способом начать использовать Matroska является использование MMG, графической оболочки, поставляемой с mkvtoolnix. Следуйте [руководству к mkvmerge GUI \(mmg\)](#).

Также Вы можете мультиплексировать аудио и видео файлы используя командную строку:

```
mkvmerge -o выходной_файл.mkv входное_видео.avi входное_аудио1.mp3 входное_аудио2.ac3
```

Это объединит видеофайл *входное_видео.avi* и два аудиофайла *входное_аудио1.mp3* и *входное_аудио2.ac3* в Matroska файл *выходной_файл.mkv*. Как было отмечено ранее, Matroska способна реализовать гораздо большее, например, множественные аудиодорожки (включая тонкую настройку аудио/видео синхронизации), разделы, субтитры, разбиение и т.д.. Пожалуйста, обратитесь к документации на эти приложения для деталей.

7.2. Как работать с телесинем и чересстрочной развёрткой на NTSC DVD

7.2.1. Введение

Что такое телесин? Если Вы не понимаете многое из того, что здесь написано, прочтите [статью Википедии о телесине](#). Это понятное и разумно обширное описание того, что такое телесин.

Замечание о числах. Многие документы, включая указанную выше статью, ссылаются на количество полей в секунду 59.94 для NTSC видео и соответствующие кадровые частоты 29.97 (для подвёрнутого телесину и чересстрочного видео) и 23.976 (для построчного). Для простоты в ряде статей эти числа даже округляются до 60, 30 и 24 соответственно.

Строго говоря, все эти числа являются аппроксимациями. Чёрно-белое NTSC видео было точно с 60 полями в секунду, но позже была выбрана частота 60000/1001 для адаптации цветовой информации с одновременным сохранением совместимости с чёрно-белым телевидением. Цифровое NTSC видео (такое как на DVD) также с 60000/1001 полями в секунду. Отсюда возникла кадровая частота 30000/1001 кадр/сек для чересстрочного и телесиненного видео; построчное видео идёт с 24000/1001 кадр/сек.

Старые версии документации MEncoder и много архивных сообщений из списков рассылки ссылаются на 59.94, 29.97 и 23.976. Вся документация MEncoder была обновлена для использования дробных значений, и Вам так же следует их использовать.

-ofps 23.976 — неправильно. Взамен нужно использовать **-ofps 24000/1001**.

Как используется телесин. Всё видео, предназначенное для просмотра на NTSC телевидении должно быть с 60000/1001 полями в секунду. Фильмы, сделанные для показа на TV часто снимаются непосредственно при 60000/1001 полей в секунду, но большинство кино снимается на 24 или 24000/1001 кадрах в секунду. В процессе создания DVD с кинофильмом, видео преобразуется для телевидения с помощью процесса, называемого телесин.

В действительности, видео никогда не хранится на DVD с 60000/1001 полями в секунду. Для видео, оригинально являющегося 60000/1001, каждая пара полей объединяется для формирования кадра, приводя к 30000/1001 кадрам в секунду. Затем аппаратные DVD плееры читают флаг, включенный в видеопоток, для определения того какие, чётные или нечётные строки должны формировать первый кадр.

Обычно, содержимое с частотой кадров 24000/1001 остаётся неизменным при кодировании на DVD и DVD плеер должен выполнить телесин на лету. Однако, иногда видео подвергается телесину до записи на DVD; и хотя оно изначально было с 24000/1001 кадр/сек, видео становится с 60000/1001 полями в секунду. Когда оно сохраняется на DVD, пары полей объединяются для формирования 30000/1001 кадров в секунду.

При рассмотрении отдельных кадров, полученных из 60000/1001 полей в секунду, телесиненных или наоборот, чересстрочная развёртка чётко видна в случае, если есть какое-либо движение, поскольку одно поле (скажем, с чётными номерами строк) отображает момент времени на $1/(60000/1001)$ секунды позже, чем другое поле. Воспроизведение чересстрочного видео на компьютере выглядит скверно по двум причинам: монитор обладает более высоким разрешением и видео показывается покадрово, вместо отображения по полям.

Замечания:

- Этот раздел применим только к NTSC DVD, а не к PAL.
- Примеры командных строк MEncoder в данном разделе **не** предназначены для реального использования. Они просто являются минимально необходимым требованием для кодирования соответствующей категории видео. То, как сделать хорошие DVD рипы или тонко настроить libavcodec для достижения максимального качества, не входит в рамки данного раздела; обратитесь к другим разделам [Руководства по кодированию с MEncoder](#).

- Есть несколько сносок, специфичных для данного руководства, обозначенных следующим образом: [\[1\]](#)

7.2.2. Как распознать тип Вашего видео

7.2.2.1. Построчная развёртка

Видео с построчной развёрткой изначально записывается на 24000/1001 fps и сохраняется на DVD без чередования.

При воспроизведении DVD с построчной развёрткой в MPlayer, MPlayer выведет следующую строку при начале воспроизведения фильма:

`demux_mpg: обнаружено 24000/1001 кадра/сек NTSC содержимое с построчной развёрткой, переключаю частоту кадров.`

Начиная с этого момента, `demux_mpg` никогда не должен сообщать о том, что найдено "30000/1001 кадров/сек NTSC содержимое".

При просмотре видео с построчной развёрткой Вы не должны никогда наблюдать чересстрочность. Однако, будьте осторожны, поскольку иногда есть небольшая примесь телесина там, где Вы этого не ожидаете. Мной наблюдались DVD с TV-шоу, у которых была одна секунда телесина при каждой смене сцен или в случайных на вид местах. Однажды я видел DVD, у которого одна половина была с построчной развёрткой, а вторая — телесиненной. Если Вы желаете быть *действительно* уверенными, Вы можете просканировать весь фильм:

```
mplayer dvd://1 -nosound -vo null -benchmark
```

Использование `-benchmark` позволяет MPlayer воспроизводить фильм столь быстро, сколь это возможно; тем не менее, в зависимости от Вашего железа, это может занять некоторое время. Всякий раз, когда `demux_mpg` будет сообщать об изменении частоты кадров, строка прямо над сообщением покажет Вам время, при котором произошло изменение.

Иногда видео на DVD с построчной развёрткой называют "мягким телесином", поскольку предполагается, что телесин будет выполнен DVD плеером.

7.2.2.2. Телесин

Телесиненное видео изначально снимается на 24000/1001 кадр/сек, но подвергается телесину до записи на DVD.

MPlayer не (всегда) сообщает об изменении частоты кадров при воспроизведении телесиненного видео.

При просмотре телесиненного видео, Вы будете видеть "мерцающие" артефакты чересстрочной развёртки: они будут многократно повторяться и исчезать. Вы можете детально это рассмотреть следующим образом:

1. `mplayer dvd://1`
2. Переместитесь в часть фильма с движением.
3. Используйте клавишу `.` для покадровой перемотки вперёд.
4. Наблюдайте за последовательностью кадров с чересстрочной и построчной развёрткой. Если Вы видите следующую структуру: ЧЧЧПП,ЧЧЧПП,ЧЧЧПП,... (где Ч — чересстрочные, а П — построчные кадры), значит видео телесиненное. Если Вы наблюдаете иную структуру, видео может быть телесиненным, используя какой-либо нестандартный метод; MEncoder не может преобразовать без потерь нестандартный телесин в построчную развёртку. Если Вы не видите вообще никакой структуры, значит наиболее вероятно, что видео с чересстрочной развёрткой.

Иногда подверженное телесину видео на DVD называют "жестким телесином". Поскольку жесткий телесин уже имеет 60000/1001 полей в секунду, DVD проигрыватель, воспроизводя его, не делает никаких преобразований.

Другой способ выяснить, был Ваш источник подвержен телесину или нет, заключается в воспроизведении исходного материала с опциями командной строки `-vf pullup` и `-v`, чтобы увидеть, как `pullup` сопоставляет кадры. Если источник был телесиненным, Вы должны увидеть в консоли 3:2 структуру с чередующимися $0+.1.+2$ и $0++1$. Преимущество этой техники состоит в том, что не требуется просматривать исходный материал для его идентификации, это может быть полезно для автоматизации процедуры кодирования или выполнения вышеуказанной процедуры удаленно через медленное соединение.

7.2.2.3. Чересстрочная развертка

Чересстрочное видео изначально снималось на 60000/1001 полями в секунду, и сохранялось на DVD с 30000/1001 кадрами в секунду. Эффект чересстрочности (часто называемый "гребёнкой") — результат объединения пары полей в кадры. Поля сдвинуты друг относительно друга на $1/(60000/1001)$ секунды, и, когда отображаются одновременно, разница заметна.

Как и с подверженным телесину видео, MPlayer не должен сообщать о каких-либо изменениях

частоты кадров при воспроизведении чересстрочного содержимого.

Внимательно, кадр за кадром (при помощи клавиши .) рассматривая чересстрочное видео, Вы увидите, что каждый отдельный кадр — чересстрочный.

7.2.2.4. Смешанные построчная развертка и телесин

Все видео со "смешанными построчной разверткой и телесином" изначально было с 24000/1001 кадрами в секунду, но некоторые его части оказались подвержены телесину.

Когда MPlayer воспроизводит эту категорию, он будет (как правило, периодически) переключаться между "30000/1001 кадров/сек NTSC содержимым" и "24000/1001 кадра/сек NTSC содержимым с построчной развёрткой". Смотрите конец вывода MPlayer, чтобы увидеть эти сообщения.

Вам следует проверить разделы с "30000/1001 кадров/сек NTSC содержимым", чтобы убедиться, что видео действительно телесиненное, а не просто чересстрочное.

7.2.2.5. Смешанные построчная и чересстрочная развертки

В содержимом со "смешанными построчной и чересстрочной развертками", построчное и чересстрочное видео переплетаются друг с другом.

Эта категория выглядит также, как и "смешанные построчная развертка и телесин", до тех пор, пока не проверите разделы 30000/1001 кадр/сек и не увидите, что структура телесина отсутствует.

7.2.3. Как кодировать каждую категорию

Как уже было сказано выше, последующие примеры командных строк MEncoder **не** означают, что надо использовать именно их; они всего лишь примеры минимального набора параметров для правильного кодирования каждой категории.

7.2.3.1. Построчная развертка

Видео с построчной разверткой не требует специальной обработки для кодирования. Единственный нужный Вам для уверенности параметр — это `-ofps 24000/1001`. В противном случае MEncoder будет пытаться кодировать с 30000/1001 кадрами в секунду и создаст дублирующиеся кадры.

```
mencoder dvd://1 -oac copy -ovc lavc -ofps 24000/1001
```

Частый случай, однако, когда видео, выглядящее построчным, на самом деле содержит очень короткие подверженные телесину части. Если Вы не уверены, безопаснее будет считать его как видео со [смешанными построчной разверткой телесином](#). Потеря скорости невелика[3].

7.2.3.2. Телесин

Телесин может быть обращён для получения оригинального 24000/1001 содержимого при помощи процесса, называемого обратный телесин. MPlayer содержит несколько фильтров для выполнения этого; лучший из них, `pullup` описан в разделе [смешанные построчная развертка и телесин](#).

7.2.3.3. Чересстрочная развертка

На практике в большинстве случаев невозможно получить полностью построчное видео из чересстрочного содержимого. Единственный способ сделать это без потери половины вертикального разрешения - это удвоить частоту кадров и попытаться "угадать", что должно составить соответствующие линии каждого поля (этот способ имеет недостатки, смотрите метод 3).

1. Кодировать видео в чересстрочной форме. Обычно это наносит вред способности кодировщика хорошо сжимать, но libavcodec имеет два параметра специально для чуть лучшего сохранения чересстрочного видео: `ildct` и `ilme`. К тому же, настоятельно рекомендуется использовать `mbd=2`[2], потому что при этом макроблоки в местах без движения будут кодированы как нечересстрочные. Имейте в виду, что `-ofps` здесь НЕ нужна.

```
mencoder dvd://1 -oac copy -ovc lavc -lavcopts ildct:ilme:mbd=2
```

2. Используйте фильтр деинтерлейсинга перед кодированием. Существует несколько таких фильтров на выбор, каждый имеет свои преимущества и недостатки. Обратитесь к `mplayer -pphelp` и `mplayer -vf help` для определения доступных (grep по "deint"), прочтите [Сравнение фильтров деинтерлейсинга](#) Майкла Нидермайера (Michael Niedermayer), и поищите в [списках рассылки MPlayer](#), чтобы найти множество обсуждений различных фильтров. И опять, частота кадров не меняется, поэтому никаких `-ofps`. к тому же деинтерлейсинг следует производить после обрезания [1] и до масштабирования.

```
mencoder dvd://1 -oac copy -vf yadif -ovc lavc
```

3. К сожалению, эта опция сбивает с MEncoder; она должна хорошо работать с MEncoder G2, но его пока нет. Вы можете столкнуться с крахами. Как бы то ни было, назначение опции `-vf tfields` — создать полный кадр из каждого поля, что делает частоту кадров

равной 60000/1001. Преимущество этого подхода в том, что никакие данные не теряются; однако, т.к. каждый кадр получается только из одного поля, недостающие строки должны как-то интерполироваться. Не существует очень хороших методов генерации недостающих данных, поэтому результат будет выглядеть несколько похожим на применение некоторых фильтров деинтерлейсинга. Генерация недостающих строк также создает другие проблемы, просто потому что количество данных удваивается. Таким образом, для сохранения качества требуются более высокие значения битпотока, и больше ресурсов процессора используется как для кодирования, так и для декодирования. `tfields` имеет несколько различных опций, определяющих способ создания недостающих строк каждого кадра. Если выбрали этот способ, обратитесь к руководству и выберите ту опцию, которая лучше подходит для Вашего материала. Имейте в виду, что при использовании `tfields` Вы **должны** указать как `-fps`, так и `-ofps`, установив им значение, равное удвоенной частоте исходного материала.

```
mencoder dvd://1 -oac copy -vf tfields=2 -ovc lavc \
-fps 60000/1001 -ofps 60000/1001
```

4. Если планируете сильно уменьшать размер изображения, можно извлекать и декодировать только одно поле из двух. Конечно, Вы потеряете половину вертикального разрешения, но если планируется уменьшать размер как минимум вдвое, потеря будет не сильно заметна. В результате получится построчной развертки файл с 30000/1001 кадрами в секунду. Процедура следующая: `-vf field`, затем обрезание [1] и масштабирование соответствующим образом. Помните, что потребуются скорректировать масштабирование для компенсации уменьшенного вдвое вертикального разрешения.

```
mencoder dvd://1 -oac copy -vf field=0 -ovc lavc
```

7.2.3.4. Смешанные построчная развертка и телесин

Для преобразования видео со смешанными построчной разверткой и телесином в полностью построчное необходимо к подверженным телесину частям применить обратный телесин. Есть три описанных ниже способа добиться этого. Заметьте, что следует **всегда** применять обратный телесин до какого-либо масштабирования; за исключением случая, когда Вы точно знаете, что делаете, выполняйте обратный телесин также до обрезания [1]. `-ofps 24000/1001` здесь необходима, поскольку видео на выходе будет с 24000/1001 кадрами в секунду.

- `-vf pullup` разработана для обратного телесина материала, телесину подверженного, оставляя построчные данные как есть. Для правильной работы после `pullup` **должен** следовать фильтр `softskip`, иначе произойдет крах MEncoder. `pullup` является, однако, самым чистым и точным методом, доступным для кодирования и телесина, и "смешанного построчного с телесином".

```
mencoder dvd://1 -oac copy -vf pullup,softskip \
-ovc lavc -ofps 24000/1001
```

- `-vf filmdint` похожа на `-vf pullup`: оба фильтра пытаются сопоставить пару полей для формирования полного кадра. Однако `filmdint` будет производить деинтерлейсинг одиночных полей, которым не может найти пару, в то время как `pullup` попросту их отбросит. Вдобавок фильтры имеют различные алгоритмы анализа, и `filmdint` имеет тенденцию к более частому нахождению соответствий. Какой фильтр будет лучше работать зависит от исходного видео и личного вкуса; не бойтесь экспериментировать с тонкой настройкой опций фильтров, если у Вас возникли проблемы с любым из них (подробности смотрите на странице руководства man). Для большинства качественного исходного видео, однако, оба фильтра работают достаточно хорошо, так что начинать работать можно с любым из них.

```
mencoder dvd://1 -oac copy -vf filmdint -ovc lavc -ofps 24000/1001
```

- Более старый метод заключается не в применении обратного телесина к телесиненным частям, а, наоборот, в телесине не подверженных телесину частей и последующем применении обратного телесина ко всему видео. Звучит запутанно? `softpulldown` — это фильтр, проходящий по видео и делающий телесиненным весь файл. Если следом за `softpulldown` указать либо `detc`, либо `ivtc`, финальный результат будет полностью построчным. `-ofps 24000/1001` необходима.

```
mencoder dvd://1 -oac copy -vf softpulldown,ivtc=1 -ovc lavc -ofps 24000/1001
```

7.2.3.5. Смешанные построчная и чересстрочная развертки

Существует две опции для этой категории, каждая из которых — это компромисс. Вы должны выбрать, исходя из продолжительности/положения каждого типа.

- Рассматривайте видео как построчное. Чересстрочные части будут выглядеть чересстрочными, и потребуются удаление некоторых из чересстрочных полей, что даст в результате некоторое скачкообразное дрожание. Вы можете использовать фильтр постобработки, если хотите, но это может несколько ухудшить построчные части. Эта опция определенно не должна использоваться, если Вы хотите со временем отображать видео на чересстрочном устройстве (с помощью TV карты, например). Если у Вас есть чересстрочные кадры в видео с 24000/1001 кадрами в секунду, к ним, как и к прогрессивным, будет применен телесин. Половина их чересстрочных "кадров" будут отображаться с длительностью трех полей (3/(60000/1001) секунд), давая в результате неприятно выглядящий эффект "прыжка назад во времени". Даже если Вы пробуете это, Вы **должны** использовать фильтр деинтерлейсинга, такой как `lb` или `l5`. Для отображения на построчном дисплее это тоже может быть плохой идеей. Будут отбрасываться пары последовательных чересстрочных полей, приводя к разрывам, которые могут быть заметнее, чем при использовании второго метода, отображающего некоторые построчные кадры дважды. Чересстрочное видео с 30000/1001 кадрами в секунду уже несколько прерывисто, потому что в реальности оно должно отображаться с 60000/1001 полями в секунду, так что дублирующиеся кадры не так сильно выделяются. Так или иначе, лучше всего проанализировать Ваше содержимое и как Вы его собираетесь показывать. Если видео на 90% построчное и Вы никогда не будете показывать его на TV, Вам следует отдать предпочтение построчному варианту. Если оно только наполовину построчное, Вы, возможно, захотите кодировать его, как если бы оно было чересстрочным.
- Считайте его чересстрочным. Некоторые кадры построчной части потребуют дублирования, что даст в результате некоторое скачкообразное дрожание. И снова, фильтры деинтерлейсинга могут несколько ухудшить построчные части.

7.2.4. Примечания

1. Об усечении сторон: Видеоданные на DVD хранятся в формате, называемом YUV 4:2:0. В YUV видео, люма ("яркость") и хрома ("цвет") хранятся отдельно. Поскольку человеческий глаз отчасти менее чувствителен к цвету, чем к яркости, в YUV 4:2:0 изображении присутствует только один цветностный пиксел на четыре яркостных. В изображении с построчной развёрткой каждый квадрат из четырёх яркостных пикселов (два на два) имеют один общий цветностный пиксел. Вы должны обрезать построчный YUV 4:2:0 до чётных размеров и использовать чётные смещения. Например, `crop=716:380:2:26` — правильно, а `crop=716:380:3:26` — нет.

Когда имеете дело с чересстрочным YUV 4:2:0, ситуация чуть более сложная. Вместо разделения одного цветностного пиксела четырьмя яркостными пикселями в *кадре*, каждые четыре яркостных пиксела каждого *поля* разделяют цветностный пиксел. Когда поля объединены в кадр, каждая строка имеет высоту в один пиксел. Теперь, вместо квадрата из четырех яркостных пикселов мы имеем два соседних пиксела, а два других расположены на две строки ниже. Два яркостных пиксела следующей строки принадлежат другому полю, и, поэтому, разделяют другой пиксел цветности с двумя пикселями на две строки дальше. Вся эта неразбериха требует, чтобы вертикальные размеры и смещения обрезания были кратны четырем. Горизонтальные могут оставаться четными.

Для телесиненного видео я рекомендую производить обрезание после обратного телесина. Так как видео построчное, достаточно обрезать только по четным размерам. Если же действительно хотите получить небольшую прибавку к скорости, которую может дать обрезка, Вам придется производить усечение с вертикальными размерностями, кратными четырем. В противном случае фильтр обратного телесина не будет иметь правильных данных.

Для чересстрочного (не подверженного телесину) видео, Вы всегда должны производить усечение с вертикальными размерностями, кратными четырем, если только не используете `-vf field` перед усечением.

2. О параметрах кодирования и качестве: Если я здесь рекомендую `mbd=2`, это еще не значит, что эту опцию не следует использовать где-либо еще. Совместно с `trell`, `mbd=2` является одной из двух опций `libavcodec`, которые значительно увеличивают качество. Вам всегда следует использовать как минимум эти две, за исключением случая, когда потеря скорости кодирования недопустима (например, кодирование в реальном времени). Есть множество других `libavcodec` опций, улучшающих качество (и замедляющих кодирование), но их описание выходит за рамки этого документа.

- О производительности pullup: Использование `pullup` (совместно с `softskip`) для видео с построчной развёрткой вполне безопасно и обычно является хорошей идеей, если только про источник не известно достоверно, что он полностью построчный. Потеря скорости мала в большинстве случаев. В минимальном варианте кодирования `pullup` замедляет MEncoder на 50%. Добавление обработки звука и продвинутых `lavcopts` опций затмевает эту разницу, уменьшая падение производительности от использования `pullup` до 2%.

7.3. Кодирование семейством кодеков libavcodec

libavcodec предоставляет возможность простого кодирования в множество интересных видео и аудио форматов. Вы можете кодировать следующими кодеками (более или менее свежий список):

7.3.1. Видео кодеки libavcodec

Название видео кодека	Описание
mjpeg	Motion JPEG
ljpeg	JPEG без потери качества
jpegls	JPEG LS
targa	Targa рисунок
gif	GIF рисунок
bmp	BMP рисунок
png	PNG рисунок
h261	H.261
h263	H.263
h263p	H.263+
mpeg4	ISO стандарт MPEG-4 (DivX, Xvid совместимый)
msmpeg4	вариант пре-стандарта MPEG-4 от MS, v3 (он же DivX3)
msmpeg4v2	вариант пре-стандарта MPEG-4 от MS, v2 (используемый в старых ASF файлах)
wmv1	Windows Media Video, версия 1 (он же WMV7)
wmv2	Windows Media Video, версия 2 (он же WMV8)
rv10	RealVideo 1.0
rv20	RealVideo 2.0
mpeg1video	MPEG-1 видео
mpeg2video	MPEG-2 видео
huffyuv	сжатие без потерь
ffvhuff	huffyuv без потерь, модифицированный FFmpeg
asv1	ASUS Видео v1
asv2	ASUS Видео v2
ffv1	видео кодек без потерь из FFmpeg
svq1	Sorenson видео 1
flv	Sorenson H.263 используемый в Flash Видео
flashsv	Flash Screen Video
dvvideo	Sony Digital Video
snow	экспериментальный кодек FFmpeg, основанный на вейвлетах
zmbv	Zip Motion Blocks Video

Название видео кодека	Описание
dnxhd	AVID DNxHD

Первый столбец содержит названия кодеков, которые следует указывать после `vcodec` опции, например: `-lavcopts vcodec=msmpeg4`

Пример с MJPEG сжатием:

```
mencoder dvd://2 -o title2.avi -ovc lavc -lavcopts vcodec=mjpeg -oac copy
```

7.3.2. Аудио кодеки libavcodec

Название аудио кодека	Описание
ac3	Dolby Digital (AC-3)
adpcm_*	Форматы Adaptive PCM, смотрите дополнительную таблицу
flac	Free Lossless Audio Codec (FLAC)
g726	G.726 ADPCM
libamr_nb	3GPP Adaptive Multi-Rate (AMR) узкополосный
libamr_wb	3GPP Adaptive Multi-Rate (AMR) широкополосный
libfaac	Advanced Audio Coding (AAC) - используя FAAC
libgsm	ETSI GSM 06.10 full rate
libgsm_ms	Microsoft GSM
libmp3lame	MPEG-1 audio layer 3 (MP3) - используя LAME
mp2	MPEG-1 audio layer 2 (MP2)
pcm_*	PCM форматы, смотрите дополнительную таблицу
roq_dpcm	Id Software RoQ DPCM
sonic	экспериментальный кодек от FFmpeg с потерями (lossy)
sonicls	экспериментальный кодек от FFmpeg без потерь (lossless)
vorbis	Vorbis
wmav1	Windows Media Audio v1
wmav2	Windows Media Audio v2

Первый столбец содержит названия кодеков, которые следует указывать после `acodec` опции, например: `-lavcopts acodec=ac3`

Пример с AC-3 сжатием:

```
mencoder dvd://2 -o title2.avi -oac lavc -lavcopts acodec=ac3 -ovc copy
```

В отличие от видео кодеков `libavcodec`, ее аудио кодеки не очень разумно используют отданные им биты, в силу неудачной реализации некоторой минимальной психоакустической модели (если она вообще есть), которая является характерной чертой большинства остальных реализаций кодеков. Однако заметьте, что все эти аудио кодеки очень быстры и работают прямо из коробки везде, где MEncoder скомпилирован с `libavcodec` (а почти всегда так оно и есть), и не зависят от внешних библиотек.

7.3.2.1. Дополнительная таблица PCM/ADPCM форматов

Название PCM/ADPCM кодека	Описание
pcm_s32le	signed 32-bit little-endian
pcm_s32be	signed 32-bit big-endian
pcm_u32le	unsigned 32-bit little-endian
pcm_u32be	unsigned 32-bit big-endian

Название PCM/ADPCM кодека	Описание
pcm_s24le	signed 24-bit little-endian
pcm_s24be	signed 24-bit big-endian
pcm_u24le	unsigned 24-bit little-endian
pcm_u24be	unsigned 24-bit big-endian
pcm_s16le	signed 16-bit little-endian
pcm_s16be	signed 16-bit big-endian
pcm_u16le	unsigned 16-bit little-endian
pcm_u16be	unsigned 16-bit big-endian
pcm_s8	signed 8-bit
pcm_u8	unsigned 8-bit
pcm_alaw	G.711 A-LAW
pcm_mulaw	G.711 μ -LAW
pcm_s24daud	signed 24-bit D-Cinema Audio формат
pcm_zork	Activision Zork Nemesis
adpcm_ima_qt	Apple QuickTime
adpcm_ima_wav	Microsoft/IBM WAVE
adpcm_ima_dk3	Duck DK3
adpcm_ima_dk4	Duck DK4
adpcm_ima_ws	Westwood Studios
adpcm_ima_smjpeg	SDL Motion JPEG
adpcm_ms	Microsoft
adpcm_4xm	4X Technologies
adpcm_xa	Phillips Yellow Book CD-ROM eXtended Architecture
adpcm_ea	Electronic Arts
adpcm_ct	Creative 16->4-bit
adpcm_swf	Adobe Shockwave Flash
adpcm_yamaha	Yamaha
adpcm_sbpro_4	Creative VOC SoundBlaster Pro 8->4-bit
adpcm_sbpro_3	Creative VOC SoundBlaster Pro 8->2.6-bit
adpcm_sbpro_2	Creative VOC SoundBlaster Pro 8->2-bit
adpcm_thp	Nintendo GameCube FMV THP
adpcm_adx	Sega/CRI ADX

7.3.3. Опции кодирования libavcodec

В идеале, Вы, наверное, хотели бы иметь возможность просто сказать кодировщику переключиться на "высокое качество" и начать кодирование. Это было бы замечательно, но, к сожалению, трудно реализуемо, поскольку различные опции кодирования, в зависимости от исходного материала, дают в результате различное качество. Так происходит потому, что сжатие зависит от визуальных свойств видео. Например, аниме и живая съемка имеют сильно отличающиеся свойства и, поэтому, требуют разные опции для получения оптимального результата. Хорошая новость состоит в том, что некоторые опции, такие как `mbd=2`, `trell` и `v4mv`, никогда не следует опускать. Детальное описание основных опций кодирования смотрите ниже.

Опции для настройки:

- **vmax_b_frames**: хороши 1 или 2, в зависимости от фильма. Заметьте, если хотите, чтобы Ваш фильм декодировался DivX5, Вы должны активировать поддержку закрытых GOP, используя опцию `cgop libavcodec`, но также должны деактивировать определение сцен, что не является хорошей идеей, поскольку несколько вредит эффективности.
- **vb_strategy=1**: помогает в высокодинамичных сценах. Для некоторых видео файлов `vmax_b_frames` может повредить качеству, но `vmax_b_frames=2` вместе с `vb_strategy=1` поможет в этом случае.
- **dia**: диапазон поиска движения. Большие значения лучше и медленнее. Отрицательные значения — это совершенно другая шкала. Хорошими значениями являются -1 для быстрого кодирования или 2-4 — для медленного.
- **predia**: предпроход поиска движения. Не так важен, как `dia`. Хорошими являются значения от 1 (по умолчанию) до 4. Требуется `preme=2`, чтобы быть действительно полезным.
- **cmp, subcmp, precmp**: Функция сравнения для поиска движения. Поэкспериментируйте со значениями 0 (по умолчанию), 2 (hadamard), 3 (dct), и 6 (соотношение сигнал-шум). 0 — самый быстрый и достаточен для `precmp`. В случае `cmp` и `subcmp`, 2 является хорошим для аниме, а 3 для живой съемки. 6 может оказаться лучше, а может и нет, но он медленнее.
- **last_pred**: Количество предсказателей движения, берущихся из предыдущего кадра. 1-3 или около того помогут Вам ценой небольшой потери в скорости. Большие значения медленны и не дают дополнительного улучшения.
- **cbp, mv0**: Контролирует выбор макроблоков. Незначительное снижение скорости с небольшим приростом в качестве.
- **qprd**: адаптивное квантование, основанное на сложности макроблока. Может сделать лучше или хуже в зависимости от видео и других опций. Она также может привести к появлению артефактов, если Вы не установите `vqmax` в некоторое разумно малое значение (хорошо — 6, может быть даже 4); `vqmin=1` также может помочь.
- **qns**: очень медленно, особенно в комбинации с `qprd`. Эта опция укажет кодировщику минимизировать шум от артефактов сжатия вместо создания закодированного видео, полностью соответствующего исходному. Не используйте ее, если только не перепробовали настроить все, что было возможно, а результат все таки недостаточно хорош.
- **vqcomp**: Настраивает управление битпоток. Какие значения являются хорошими, зависит от фильма. Если хотите, можете без опаски оставить значение по умолчанию. Уменьшение `vqcomp` отдает больше бит в сцены с низкой сложностью, увеличение его передает биты в очень сложные сцены (по умолчанию: 0.5, диапазон: 0-1, рекомендуемый диапазон: 0.5-0.7).
- **vlelim, vcelim**: Устанавливает порог отбрасывания одиночного коэффициента для яркостной и цветностной плоскостей. Они кодируются независимо во всех MPEG-подобных алгоритмах. Идея этих опций заключается в использовании некоторой хорошей эвристики для определения момента, когда изменения в блоке ниже указанного Вами порога, и что его стоит кодировать как "блок без изменений". Это сохраняет биты и, возможно, ускоряет кодирование. `vlelim=-4` и `vcelim=9` выглядят неплохими для живой съемки, но, скорее всего, не помогут для аниме; при кодировании анимации Вам, возможно, следует оставить эту опцию неизменной.
- **qpel**: Четвертьпиксельная оценка движения. По-умолчанию, MPEG-4 использует полупиксельную точность для поиска движения, следовательно, эта опция вносит дополнительные накладные расходы, поскольку сохраняет больше информации в закодированном файле. Улучшение/ухудшение степени сжатия зависит от фильма, но обычно эта опция не очень эффективна для аниме. `qpel` всегда вносит значительный вклад в CPU время декодирования (+25% на практике).
- **psnr**: не влияет на сам процесс кодирования, но выводит в файл тип/размер/качество каждого кадра, а также итоговый PSNR (Peak Signal to Noise Ratio, пиковое отношения сигнала к шуму) в конце процесса.

Опции, с которыми играть не стоит:

- **vme**: Значение по умолчанию является лучшим.
- **lumi_mask, dark_mask**: Психовизуальное адаптивное квантование. Не стоит играть с этими опциями, если заботитесь о качестве. Разумные значения могут быть эффективными в Вашем случае, но имейте в виду, что это весьма субъективно.
- **scplx_mask**: Пытается предотвратить появление квадратов, но лучше выполнить

постобработку.

7.3.4. Примеры настроек кодирования

Следующие настройки — это примеры различных комбинаций опций кодирования, которые влияют на соотношение скорость-качество при той же величине целевого битпотока. Все настройки кодирования проверялись на тестовом видео 720x448 @30000/1001 fps с целевым битпотокм 900кбит/сек, на машине AMD-64 3400+ с 2400 МГц и 64 битном режиме. Для каждой настройки кодирования указаны измеренная скорость кодирования (в кадрах в секунду) и потеря PSNR (в дБ) по сравнению с настройкой "очень высокое качество". Поймите, пожалуйста, что в зависимости от Вашего материала, типа машины, прогресса разработки Вы можете получить сильно отличающиеся результаты.

Описание	Опции кодирования	скорость (в fps)	Относительная потеря PSNR (в дБ)
Очень высокое качество	<code>vcodec=mpeg4:mbd=2:mv0:trell:v4mv:cbp:last_pred=3:predia=2:dia=2:vmax_b_frames=2:vb_strategy=1:precmp=2:cmp=2:subcmp=2:preme=2:qns=2</code>	6fps	0дБ
Высокое качество	<code>vcodec=mpeg4:mbd=2:trell:v4mv:last_pred=2:dia=-1:vmax_b_frames=2:vb_strategy=1:cmp=3:subcmp=3:precmp=0:vqcomp=0.6:turbo</code>	15fps	-0.5дБ
Быстрое	<code>vcodec=mpeg4:mbd=2:trell:v4mv:turbo</code>	42fps	-0.74дБ
Реального времени	<code>vcodec=mpeg4:mbd=2:turbo</code>	54fps	-1.21дБ

7.3.5. Нестандартные inter/intra матрицы

С этой возможностью libavcodec, Вы можете установить нестандартные inter (I-кадры/ключевые) и intra (P-кадры/предсказанные) матрицы. Это поддерживается многими кодеками: В mpeg1video и mpeg2video также заявлена поддержка.

Обычное использование этой опции — установить матрицы, предпочитаемые спецификациями [KVCD](#).

KVCD Матрица Квантования "Notch":

Intra:

8 9 12 22 26 27 29 34 9 10 14 26 27 29 34 37

12 14 18 27 29 34 37 38
22 26 27 31 36 37 38 40
26 27 29 36 39 38 40 48
27 29 34 37 38 40 48 58
29 34 37 38 40 48 58 69
34 37 38 40 48 58 69 79

Inter:

16 18 20 22 24 26 28 30
18 20 22 24 26 28 30 32
20 22 24 26 28 30 32 34
22 24 26 30 32 32 34 36
24 26 28 32 34 34 36 38
26 28 30 32 34 36 38 40
28 30 32 34 36 38 42 42
30 32 34 36 38 40 42 44

Использование:

```
mencoder input.avi -o output.avi -oac copy -ovc lavc \
-lavcopts inter_matrix=...:intra_matrix=...
mencoder input.avi -ovc lavc -lavcopts \
```

```
vcodec=mpeg2video:intra_matrix=8,9,12,22,26,27,29,34,9,10,14,26,27,29,34,37,\
12,14,18,27,29,34,37,38,22,26,27,31,36,37,38,40,26,27,29,36,39,38,40,48,27,\
29,34,37,38,40,48,58,29,34,37,38,40,48,58,69,34,37,38,40,48,58,69,79\
:inter_matrix=16,18,20,22,24,26,28,30,18,20,22,24,26,28,30,32,20,22,24,26,\
28,30,32,34,22,24,26,30,32,32,34,36,24,26,28,32,34,34,36,38,26,28,30,32,34,\
36,38,40,28,30,32,34,36,38,42,42,30,32,34,36,38,40,42,44 -oac copy -o svcd.mpg
```

7.3.6. Пример

Итак, Вы только что купили новенькую, блестящую копию фильма "Гарри Поттер и Тайная Комната" (в широкоэкранный формат, конечно) и хотите сделать рип этого DVD так, чтобы добавить его к Домашнему кинотеатру на PC. Это DVD первого региона, поэтому NTSC. Пример ниже также применим и для PAL, за исключением того, что надо будет опустить `-ofps 24000/1001` (поскольку частота кадров на выходе такая же, как и на входе), и, конечно, границы обрезания будут другими.

После запуска `mplayer dvd://1` мы следуем процессу, детально описанному в разделе [Как работать с телесиноном и чересстрочностью в NTSC DVD](#), и выясняем, что это 24000/1001 fps построчное видео, а значит, использовать фильтры обратного телесина, такие как `pullup` или `filmdint` не нужно.

Далее, мы хотим определить верные границы обрезания, поэтому используем фильтр `cropdetect`:

```
mplayer dvd://1 -vf cropdetect
```

Убедитесь, что переместились к полностью заполненному кадру (например, к светлой сцене после пропущенных начальных титров и логотипов), Вы должны увидеть в консоли MPlayer:

```
crop area: X: 0..719 Y: 57..419 (-vf crop=720:362:0:58)
```

Затем снова воспроизводим фильм с этим фильтром для проверки его корректности:

```
mplayer dvd://1 -vf crop=720:362:0:58
```

И убеждаемся, что все выглядит прекрасно. Далее, проверяем, что ширина и высота делятся на 16. С шириной все в порядке, а с высотой — нет. Поскольку мы не заваливали математику в 7-ом классе, то знаем, что ближайшее целое, меньшее 362 и кратное 16, равно 352.

Мы могли бы просто использовать `crop=720:352:0:58`, но будет лучше отрезать понемногу от верха и низа, чтобы центр остался на месте. Мы уменьшили высоту на 10 пикселей, но не хотим увеличивать смещение по y на 5, поскольку это нечетное число и отрицательно скажется на качестве. Вместо этого, мы увеличим y на 4:

```
mplayer dvd://1 -vf crop=720:352:0:62
```

Другая причина, по которой мы урезаем пиксели сверху и снизу, заключается в том, что мы хотим убедиться, что удалены все наполовину черные пиксели, если они есть. Если Ваше видео подвержено телесину, убедитесь, что фильтр `pullup` (или любой другой фильтр обратного телесина, который Вы решили использовать) находится в цепочке до фильтра `crop`. Если оно чересстрочное, то перед обрезкой проведите деинтерлейсинг. (Если решили сохранить чересстрочность видео, убедитесь, что вертикальный сдвиг обрезания кратен 4.)

Если Вас действительно беспокоит потеря этих 10 пикселей, Вы можете вместо этого отмасштабировать фильм, уменьшив размерности до ближайших кратных 16 значений. Цепочка фильтров будет выглядеть примерно так:

```
-vf crop=720:362:0:58,scale=720:352
```

Подобное уменьшение изображения будет означать потерю небольшого количества деталей, хотя это, возможно, окажется незаметным. Масштабирование изображения в сторону увеличения даст худшее качество (если Вы не увеличиваете битпоток). Обрезка же полностью выбросит те пиксели. Это компромисс, идти на который или нет, придется решать в каждом частном случае. Например, если DVD видео было создано для телевидения, Вы можете захотеть избежать вертикального масштабирования, поскольку дискретизация строк соответствует тому, как содержимое изначально записывалось.

При проверке видим, что наш фильм имеет немного движения и большое количество деталей, так что выбираем для битпотока значение 2400Кбит/сек.

Теперь мы готовы произвести двухпроходное кодирование. Проход первый:

```
mencoder dvd://1 -ofps 24000/1001 -oac copy -o Harry_Potter_2.avi -ovc lavc \
-vlc -lavcopts
vcodec=mpeg4:vbitrate=2400:v4mv:mbd=2:trell:cmp=3:subcmp=3:autoaspect:vpass=1 \
-vf pullup,softskip,crop=720:352:0:62,hqdn3d=2:1:2
```

И второй проход с теми же параметрами, за исключением `vpass=2`:

```
mencoder dvd://1 -ofps 24000/1001 -oac copy -o Harry_Potter_2.avi -ovc lavc \
-vlc -lavcopts
vcodec=mpeg4:vbitrate=2400:v4mv:mbd=2:trell:cmp=3:subcmp=3:autoaspect:vpass=2 \
-vf pullup,softskip,crop=720:352:0:62,hqdn3d=2:1:2
```

Опции `v4mv:mbd=2:trell` значительно улучшат качество ценой времени кодирования. Нет никаких оснований отключать эти опции, когда главным критерием является качество. Опции `cmp=3:subcmp=3` выбирают функцию сравнения, дающую лучшее качество, чем стандартная. Вы можете поэкспериментировать с этим параметром (возможные значения смотрите на map странице), поскольку разные функции могут давать разный прирост в качестве в зависимости

от исходного материала. Например, если Вы замечаете, что libavcodec производит слишком много блочных артефактов (квадратиков), то можете попытаться выбрать экспериментальный NSSE в качестве функции сравнения при помощи опции `*cmp=10`.

Для этого фильма полученный AVI будет 138 минут длинной и размером около 3Гб. И, поскольку Вы сказали, что размер файла значения не имеет, это вполне приемлемый результат. Однако, если все-таки хотите получить меньший размер файла, можете попробовать уменьшить битпоток. Увеличение битпотока имеет снижающийся эффект, поэтому, хотя мы можем ясно видеть улучшение от 1800Кбит/сек до 2000Кбит/сек, оно может быть не столь заметно выше 2000Кбит/сек.

Так как мы пропустили исходное видео через фильтр удаления шума, то, возможно, захочется вернуть какую-то его часть во время воспроизведения. Это, совместно с фильтром постобработки `spp`, существенно улучшит воспринимаемое качество и поможет избежать блочных артефактов в видео. Опцией `autoq` MPlayer'a Вы можете изменять величину производимой фильтром `spp` постобработки в зависимости от доступных ресурсов CPU. Вдобавок, на этом этапе Вы можете захотеть применить коррекцию гаммы и/или цвета для лучшего соответствия Вашему монитору. Например:

```
mpplayer Harry_Potter_2.avi -vf spp,noise=9ah:5ah,eq2=1.2 -autoq 3
```

7.4. Кодирование кодеком Xvid

Xvid — это свободная библиотека для кодирования MPEG-4 ASP видео потоков. Перед тем, как начать кодирование, Вам потребуется [настроить MEncoder для его поддержки](#).

Это руководство в основном нацелено на особенности применения тех же методов, что описаны в руководстве по кодированию с помощью x264. Поэтому, сначала прочтите, пожалуйста, [первую часть](#) того руководства.

7.4.1. Какие опции следует использовать для получения лучших результатов?

Пожалуйста, начните с просмотра раздела Xvid man страницы MPlayer. Этот раздел предполагается как дополнение к man странице.

Настройки по умолчанию Xvid уже являются хорошим выбором между скоростью и качеством, поэтому Вы можете без опасений придерживаться их, если следующий раздел Вас озадачивает.

7.4.2. Опции кодирования Xvid

- **vhq** Эта опция влияет на алгоритм принятия решений о макроблоке, чем выше значение, тем мудрее будут решения. Значение по умолчанию можно без опаски использовать для любого кодирования, в то время, как более высокие значения всегда улучшат PSNR, но будут работать значительно медленнее. Заметьте, пожалуйста, что лучший PSNR не обязательно означает лучше выглядящую картинку, но говорит, что она ближе к оригиналу. Отключение этой опции заметно ускоряет кодирование; это может быть достойным компромиссом, если скорость Вам критична.
- **bvhq** То же, что и `vhq`, но для В-кадров. Имеет незначительное влияние на скорость и слегка улучшает качество (около +0.1дБ).
- **max_bframes** Больше число допустимых последовательных В-кадров обычно улучшает сжимаемость, хотя оно может также привести к большему количеству блочных артефактов (квадратиков). Значение по умолчанию — хороший выбор между сжимаемостью и качеством, но Вы можете увеличить его до 3, если стеснены величиной битпотока. Вы также можете уменьшить это значение до 1 или 0, если печетесь об отличном качестве, впрочем в этом случае Вы должны убедиться, что целевой битпоток достаточно высок, дабы кодировщик не увеличивал значение квантователя, сохраняя нужную величину битпотока.
- **bf_threshold** Управляет чувствительностью кодировщика к В-кадрам, где большие значения приводят к использованию большего количества В-кадров (и наоборот). Опция должна использоваться совместно с `max_bframes`; если Вы стеснены величиной битпотока, то должны увеличить и `max_bframes`, и `bf_threshold`, в том время как увеличение `max_bframes` и уменьшение `bf_threshold` позволят кодировщику использовать больше В-кадров в местах, где это **действительно** необходимо. Низкое количество `max_bframes` и высокое значение `bf_threshold` — это, возможно, не самое мудрое решение, поскольку оно принудит кодировщик размещать В-кадры в местах, которые никак не выигрывают от этого, тем самым ухудшая визуальное качество. Однако, если Вам требуется совместимость с аппаратными проигрывателями, поддерживающими только старые DivX профили (которые поддерживают только 1

последовательный В-кадр), это единственный способ увеличить сжимаемость при помощи В-кадров.

- **trellis** Оптимизирует процесс квантования для получения оптимального соотношения между PSNR и битпоток, что позволяет существенно экономить биты. Эти биты впоследствии будут потрачены на другие части видео, что приведет к увеличению общего качества. Следует всегда оставлять эту опцию включенной, поскольку ее влияние на качество огромно. Даже если Вы заботитесь о скорости, не отключайте ее до тех пор, пока не выставили **vhq** и остальные более CPU-прожорливые опции на минимум.
- **hq_ac** Активирует более точный метод оценки стоимости коэффициентов, что уменьшает размер файла примерно на 0.15 - 0.19% (соответствует увеличению PSNR меньше, чем на 0.01дБ), имея несущественное влияние на скорость. Поэтому, рекомендуется всегда держать эту опцию включенной.
- **cartoon** Разработана для лучшего кодирования мультфильмов и не влияет на скорость, поскольку всего-лишь настраивает эвристики принятия решений о режимах для этого типа содержимого.
- **me_quality** Это опция для настройки точности оценки движения. Чем выше **me_quality**, тем точнее будет оценка оригинального движения и тем лучше получающийся отрывок будет фиксировать оригинальное движение. Настройка по умолчанию лучше во всех случаях, поэтому не рекомендуется ее выключать, если только Вы действительно не гонитесь за скоростью, поскольку биты, сэкономленные хорошей оценкой движения, могут быть использованы где-нибудь еще, увеличивая общее качество. Таким образом, не используйте значения ниже 5, да и его — только в крайнем случае.
- **chroma_me** Улучшает оценку движения, дополнительно принимая во внимание информацию о цвете, тогда как одна **me_quality** использует только яркость. Это замедляет кодирование на 5-10%, но несколько улучшает визуальное качество, уменьшая эффект блочности и сокращая размер файла примерно на 1.3%. Если Вас интересует скорость, следует попробовать отключить эту опцию, прежде чем решите уменьшать значение **me_quality**.
- **chroma_opt** Эта опция служит для увеличения качества цветного изображения вокруг чисто черных/белых границ вместо улучшения сжатия. Она также может помочь против эффекта "красных ступенек".
- **lumi_mask** Пытается отдать меньший битпоток областям изображения, которые человеческий глаз не в состоянии увидеть достаточно хорошо, что позволит кодировщику потратить сэкономленные биты на более важные части картинки. Качество закодированного материала, привнесенное этой опцией, сильно зависит от личных предпочтений и от типа и настроек монитора, использовавшегося для просмотра (обычно результат выглядит не очень хорошо, если он яркий, или является TFT монитором).
- **qpel** Увеличивает количество предполагаемых векторов движения, путём повышения точности оценки движения с полупиксельной до четвертьпиксельной. Идея состоит в том, чтобы найти лучшие векторы движения, которые взамен уменьшат битпоток (тем самым увеличивая качество). Однако, векторы движения с четвертьпиксельной точностью требуют большего количества дополнительных бит для кодирования, а векторы-кандидаты не всегда дают (значительно) лучшие результаты. Достаточно часто кодек тратит дополнительные биты на повышенную точность впустую, а взамен получает или вообще ничего, или небольшое увеличение качества. К сожалению, нет способа предсказать возможные улучшения от **qpel**, так что Вам придется сделать кодирование с ней и без нее, чтобы знать наверняка. **qpel** может почти удвоить время кодирования и требует, как минимум, на 25% большей мощности при декодировании. Она поддерживается не всеми аппаратными проигрывателями.
- **gmc** Пытается сэкономить биты в панорамных сценах, используя один вектор движения для всего кадра. Это почти всегда увеличивает PSNR, но заметно замедляет кодирование (так же как и декодирование). Поэтому Вас следует использовать ее, только когда Вы включили **vhq** на максимум. GMC Xvid'a является более сложным, чем у DivX'a, но поддерживается только некоторыми аппаратными проигрывателями.

7.4.3. Профили кодирования

Xvid поддерживает профили кодирования через опцию **profile**, которая используется для накладывания ограничений на значения видео потока Xvid таким образом, что он будет воспроизводиться на всем, что поддерживает выбранный профиль. Ограничения относятся к

разрешению, битпоток и некоторым возможностям MPEG-4. Следующая таблица показывает, что поддерживает тот или иной профиль.

DivX	HDTV	1280	720	30	9708.4	16000	2		X	X		
	Домашний кинотеат	720	576	25	4854	8000	1		X	X		
	Домашний кинотеат	720	480	30	4854	8000	1		X	X		
	Портативный PAL	352	288	25	4854	8000	1		X			
	Портативный NTSC	352	240	30	4854	8000	1		X			
	Карманный	176	144	15	537.6	800	0		X			
	5	720	576	30	8000			X	X	X	X	
	4	352	576	30	3000			X	X	X	X	
	3	352	288	30	768			X	X	X	X	
	2	352	288	15	384			X	X	X	X	
Расширенный простой	1	176	144	30	128			X	X	X	X	
	0	176	144	30	128			X	X	X	X	
	3	352	288	15	384		0					
Простой	2	352	288	15	128		0					
	1	176	144	15	64		0					
	0	176	144	15	64		0					
	Название профиля	Ширина [пикселей]	Высота [пикселей]	Частота кадров [fps]	Максимальный средний битпоток [кбит/сек]	Пиковое значение средней величины	Макс. В-кадров	МPEG квантование	Адаптивное квантование	Чересстрочное кодирование	Четвертьпиксельная точность	Глобальная компенсация движения

7.4.4. Примеры настроек кодирования

Последующие настройки — это примеры различных комбинаций опций кодирования, которые влияют на соотношения скорость-качество при той же величине целевого битпотока.

Все настройки кодирования проверялись на тестовом видео 720x448 @30000/1001 fps с целевым битпотокм 900кбит/сек, на машине AMD-64 3400+ с 2400 МГц и 64 битном режиме. Для каждой настройки кодирования указаны измеренная скорость кодирования (в кадрах в секунду) и потеря PSNR (в дБ) по сравнению с настройкой "очень высокое качество". Поймите, пожалуйста, что в зависимости от Вашего материала, типа машины, прогресса разработки, Вы можете получить сильно отличающиеся результаты.

Описание	Опции кодирования	скорость (в fps)	Относительная потеря PSNR (в дБ)
Очень высокое качество	<code>chroma_opt:vhq=4:bvhq=1:quant_type=mpeg</code>	16fps	0дБ
Высокое качество	<code>vhq=2:bvhq=1:chroma_opt:quant_type=mpeg</code>	18fps	-0.1дБ
Быстрое	<code>turbo:vhq=0</code>	28fps	-0.69дБ
Реального времени	<code>turbo:nochroma_me:notrellis:max_bframes=0:vhq=0</code>	38fps	-1.48дБ

7.5. Кодирование кодеком x264

x264 — это свободная библиотека для кодирования H.264/AVC видео потоков. Перед началом кодирования Вы должны [настроить MEncoder для его поддержки](#).

7.5.1. Опции кодирования x264

Начните, пожалуйста с просмотра раздела x264 map страницы MPlayer'a. Этот раздел предполагается быть дополнением к странице map. Здесь Вы найдете быстрые подсказки о том, какие опции чаще всего интересуют большинство людей. Страница map более лаконична, но также более полна и порой намного лучше преподносит технические детали.

7.5.1.1. Введение

Это руководство рассматривает две главные категории опций кодирования:

1. Опции, в основном влияющие на соотношение скорость-качество.
2. Опции, которые могут быть полезны для удовлетворения различных пользовательский предпочтений и специальных требований.

В конце концов, только Вы можете решать какие опции являются лучшими для Ваших целей. Решение для первого класса опций очень простое: надо только определить, считаете ли Вы, что разница в качестве оправдывает разницу в скорости. Для второго класса опций предпочтения могут быть значительно более субъективными и зависеть от большего числа факторов. Имейте в виду, что некоторые из опций категории "пользовательских предпочтений и специальных требований" могут все же иметь большое влияние на скорость или качество, но это не основное их предназначение. Часть опций из "пользовательских предпочтений" могут даже привести к изменениям, которые выглядят лучше для одних людей и хуже — для других. Перед тем как продолжить, Вам придется понять, что это руководство использует только одну метрику качества: глобальный PSNR. Краткое описание того, что такое PSNR, смотрите в [статье Википедии о PSNR](#). Глобальный PSNR — это последнее значение PSNR, выводимое на консоль, когда в `x264encopts` включена опция `psnr`. Каждый раз, когда Вы читаете утверждения о PSNR, за ними скрывается предположение, что используются одинаковые значения битпотока.

Почти все комментарии этого руководства предполагают, что Вы используете два прохода. Есть две основные причины использовать двухпроходное кодирование при сравнении опций. Во-первых, использование двух проходов увеличивает PSNR примерно на 1дБ, что является очень хорошим значением. Во-вторых, тестирование опций прямым сравнением качества при однопроходном кодировании вводит основной сбивающий фактор: зачастую битпоток значительно меняется при каждом кодировании. Не всегда можно с легкостью сказать, изменилось ли качество в основном за счет изменения опций, или оно по большей части отражает случайные изменения в полученном битпоток.

7.5.1.2. Опции, затрагивающие, в основном, скорость и качество

- **subq**: Из всех опций, позволяющих выбирать между скоростью и качеством, `subq` и `frameref` (смотрите ниже), пожалуй, самые важные. Если Вы заинтересованы в тонкой настройке либо скорости, либо качества, эти две — первое, с чего Вам стоит начать. С точки зрения скорости, опции `frameref` и `subq` очень жестко взаимодействуют друг с

другом. Опыт показывает, что с одним ссылочным кадром `subq=5` (настройка по умолчанию) расходует на 35% больше времени, чем `subq=1`. С 6 ссылочными кадрами эта величина достигает 60%. Эффект `subq` на PSNR выглядит довольно постоянным, в отличие от количества ссылочных кадров. Как правило, `subq=5` достигает значения глобального PSNR на 0.2-0.5 дБ большего, чем при `subq=1`. Обычно этого достаточно, чтобы заметить.

`subq=6` — медленнее и дает лучшее качество при разумной цене. Если сравнивать с `subq=5`, он обычно дает на 0.1-0.4 дБ больший глобальный PSNR ценой потери 25%-100% скорости. В отличие от остальных уровней `subq`, поведение `subq=6` не так сильно зависит от `frameref` и `me`. Вместо этого, эффективность `subq=6` по большей части зависит от количества используемых В-кадров. При обычном использовании это означает, что `subq=6` в сложных, высокодинамичных сценах имеет большое влияние как на скорость, так и на качество, но в сценах с малым количеством движения она не имеет такого эффекта. Имейте в виду, что по-прежнему рекомендуется всегда устанавливать `bframes` в значение, отличное от нуля (смотрите далее).

`subq=7` — самый медленный режим с наилучшим качеством. По сравнению с `subq=6` он, обычно, улучшает общий PSNR на 0.01-0.05 дБ ценой потери 15%-30% скорости. Поскольку соотношение качества и времени кодирования очень невелико, Вам следует использовать этот режим, только если боретесь за каждый бит, и время кодирования Вас не волнует.

- **frameref:** `frameref` по умолчанию установлена в 1, но это не значит, что ее стоит устанавливать в 1. Только увеличение `frameref` до 2 дает прирост PSNR примерно на 0.15дБ за счет уменьшения скорости на 5-10%; похоже, что это неплохая цена. `frameref=3` дает примерно 0.25дБ PSNR сверх `frameref=1`, что должно быть видимой разницей. `frameref=3` медленнее примерно на 15%, чем `frameref=1`. К сожалению, улучшение очень быстро сходит на нет. От `frameref=6` можно ожидать прироста PSNR лишь на 0.05-0.1 дБ по сравнению с `frameref=3` с дополнительной потерей 15% скорости. Выше `frameref=6` качество обычно увеличивается очень незначительно (хотя на всем протяжении этой дискуссии Вам следует иметь в виду, оно может значительно изменяться в зависимости от исходного материала). В довольно типичном случае `frameref=12` улучшит глобальный PSNR всего на 0.02дБ по сравнению с `frameref=6`, ценой 15%-20% скорости. При таких высоких значениях `frameref`, единственная действительно хорошая вещь, о которой может быть сказано, состоит в том, что дальнейшее ее увеличение почти никогда не будет **вредить** PSNR, но увеличение качества будет трудно даже измерить, не говоря уже о его заметности.

Замечание:

Увеличение `frameref` до чрезмерно высоких значений **может** и **обычно наносит** вред эффективности кодирования, если CABAC отключен. С задействованным CABAC (настройка по умолчанию), возможность установки `frameref` "слишком высоким" на данный момент выглядит слишком далекой, чтобы об этом беспокоиться, а в будущем оптимизации могут вообще убрать такую возможность.

Если Вас заботит скорость, разумным компромиссом будет использовать низкие значения `subq` и `frameref` в первом проходе, а затем увеличить их во втором. Обычно, это обладает ничтожным отрицательным эффектом на конечное качество: Вы, возможно, потеряете вплоть до 0.1дБ PSNR, что должно быть слишком малой разницей, чтобы её заметить. Однако, различные значения `frameref` могут иногда повлиять на решение о выборе типа кадра. Скорее всего, это довольно редкие крайние случаи, но если Вы хотите быть точно уверенными, посмотрите, содержит ли Ваше видео полноэкранные периодически вспыхивающие изображения или очень большие паузы, которые могут стать причиной принудительной вставки I-кадра. Настройте `frameref` в первом проходе так, чтобы она была достаточно большой для содержания длительности цикла вспыхивания (или паузы). Например, если сцены вспыхивают и гаснут между двумя изображениями в течении трёх кадров, установите `frameref` равным 3 или выше. Эта проблема, возможно, очень редко появляется для живой съемки, но она иногда возникает при записи видео игр.

- **me:** Эта опция используется для выбора метода оценки движения. Изменение этой опции оказывает прямое влияние на соотношение скорость-качество. `me=dia` лишь на несколько процентов быстрее, чем поиск по умолчанию, ценой не больше 0.1дБ глобального PSNR. Значение по умолчанию (`me=hex`) — разумный выбор между скоростью и качеством. `me=umh` немного, вплоть до 0.1дБ, улучшает глобальный PSNR, соответствующее падение скорости меняется в зависимости от `frameref`. С высокими

значениями `frameref` (например, 12 или около того), `me=umh` примерно на 40% медленнее, чем настройка по умолчанию `me=hex`. С `frameref=3`, падение скорости уменьшается до 25%-30%.

`me=esa` использует исчерпывающий поиск, который работает слишком медленно для практического применения.

- **partitions=all**: Эта опция задействует использование сегментов 8x4, 4x8 и 4x4 в предсказанных макроблоках (в дополнение к стандартным). Ее включение приведет к довольно постоянной 10%-15% потере в скорости. Эта опция практически бесполезна для исходного материала, содержащего только небольшое движение, тем не менее, для некоторого высокодинамичного материала, особенно с большим количеством мелких движущихся объектов, следует ожидать прироста около 0.1дБ.
- **bframes**: Если Вы занимались кодированием с другими кодеками, то могли заметить, что В-кадры не всегда полезны. В H.264 это изменилось: есть новые техники и типы блоков, возможные в В-кадрах. Обычно, даже примитивный алгоритм выбора В-кадров может дать значимую выгоду для PSNR. Интересно заметить, что использование В-кадров обычно отчасти ускоряет второй проход, а также может ускорить однопроходное кодирование, если отключено адаптивное принятие решения о В-кадрах.
С отключенным адаптивным принятием решения о В-кадрах (`nob_adapt` в `x264encopts`), оптимальное значение этой опции обычно не превышает `bframes=1`, иначе могут пострадать высокодинамичные сцены. С включенным адаптивным принятием решения о В-кадрах (поведение по умолчанию), можно безопасно использовать более высокие значения; кодировщик уменьшит количество В-кадров в сценах, где они повредят сжатию. Кодировщик редко решает использовать больше, чем 3 или 4 В-кадра; установка этой опции в любое более высокое значение не будет иметь большого эффекта.
- **b_adapt**: Заметьте: она включена по умолчанию.
Когда эта опция включена, кодировщик будет использовать разумно быстрый процесс принятия решения для уменьшения количества В-кадров, используемых в сценах, которые от этого не сильно выиграют. Вы можете использовать `b_bias` для тонкой настройки того, насколько "счастлив" будет кодировщик использованию В-кадров. Потеря в скорости при использовании адаптивных В-кадров на данный момент весьма невелика, но таково же и потенциальное улучшение качества. Тем не менее, хуже от этого обычно не становится. Заметьте, что эта опция влияет на скорость и решение о типе кадра только в первом проходе. `b_adapt` и `b_bias` не имеют эффекта в последующих проходах.
- **b_pyramid**: С тем же успехом Вы можете включить эту опцию, если используете ≥ 2 В-кадров; Вы получите небольшое улучшение качества без потери в скорости, как и говорит `map` руководство. Имейте в виду, что такое видео не может быть прочитано основанными на `libavcodec` декодерами, созданными ранее, чем примерно 5 Марта 2005.
- **weight_b**: В обычных случаях эта опция не дает большого улучшения. Однако, в проявляющихся или затухающих сценах взвешенное предсказание дает довольно большую экономию битпотока. В MPEG-4 ASP затухание обычно лучше кодируется последовательностью дорогих I-кадров; использование взвешенного предсказания в В-кадрах делает возможным преобразовать хотя бы часть из них в значительно более меньшие В-кадры. Потери в скорости кодирования минимальны, поскольку не требуется делать дополнительные принятия решений. Вдобавок, вопреки расхожему мнению, взвешенное предсказание не сильно влияет на требования декодера к CPU при прочих равных условиях.
К сожалению, текущий алгоритм адаптивного принятия решений о В-кадрах имеет твердую склонность к избеганию использования В-кадров при затуханиях. До тех пор, пока это не изменится, хорошей идеей, возможно, будет добавить `nob_adapt` к `x264encopts`, если предполагаете, что затухания будут давать существенный вклад в Вашем конкретном видеоклипе.
- **threads** Эта опция позволяет породить потоки для параллельного кодирования на нескольких CPU. Вы можете вручную выбрать количество создаваемых потоков или, что лучше, установить `threads=auto` и позволить `x264` определить количество доступных CPU и выбрать соответствующее количество потоков. Если у Вас многопроцессорная машина, Вам следует всерьез задуматься об использовании этой опции, так как она может увеличить скорость кодирования линейно в зависимости от числа CPU ядер (около 94% на ядро), незначительно уменьшая PSNR (примерно 0.005 дБ для

двухпроцессорной, 0.01 дБ — для четырехпроцессорной машины).

7.5.1.3. Опции, относящиеся к различным предпочтениям

- **Двухпроходное кодирование:** Выше советовалось всегда использовать кодирование в два прохода, но все же существуют причины этого не делать. Например, если Вы захватываете TV трансляцию и кодируете в реальном времени, придется использовать однопроходный режим. К тому же один проход очевидно быстрее, чем два; если Вы используете точно такой же набор опций в обоих случаях, двухпроходной режим медленнее почти вдвое.

Все же существует очень хорошие причины использовать кодирование в два прохода. Во-первых, управление битпоток однопроходного режима не является телепатом и часто делает необоснованный выбор, потому что не может видеть общую картину. Например, предположим, что Вы имеете двухминутное видео, состоящее из двух независимых частей. Первая половина — очень динамичная сцена, продолжающаяся 60 секунд и требующая сама по себе битпоток примерно 2500 кбит/сек, чтобы прилично выглядеть. Сразу за ней следует гораздо менее требовательная 60-секундная сцена, которая хорошо выглядит при 300 кбит/сек. Предположим, Вы запросили битпоток 1400 кбит/сек; в теории этого достаточно для удовлетворения потребностей обеих сцен. В этом случае управление битпоток в однопроходном режиме сделает пару "ошибок". Во-первых, оно установит битпоток в 1400 кбит/сек для обеих частей. Первая часть может оказаться чрезмерно квантованной, что приведет к недопустимо выглядящему и неоправданно блочному изображению. Вторая часть будет существенно недостаточно квантованной; она может выглядеть отлично, но цена битпотока для этого качества будет полностью неоправданной. Чего намного труднее избежать, так это проблемы перехода между двумя сценами. В первых секундах малодинамичной части квантователь будет чрезвычайно превышен, потому что управление битпоток все еще ожидает встретить такие же требования к битпоток как и в первой части. Этот "ошибочный период" с чрезвычайно превышенным квантованием будет выглядеть раздражающе неприятно и использовать на самом деле меньше, чем 300 кбит/сек, требуемых ему для того, чтобы прилично выглядеть. Существуют способы смягчить эффект от подобных подводных камней однопроходного режима, но они могут иметь склонность к усилению неверного предсказания битпотока.

Многопроходное кодирование может предложить огромные преимущества по сравнению с однопроходным. Используя статистику, собранную при первом проходе, кодировщик может оценить, с разумной точностью, "стоимость" (в битах) кодирования любого заданного кадра при любом заданном квантователе. Это делает возможным намного более рациональное, лучше спланированное распределение битов между дорогими (высокодинамичными) и дешевыми (малодинамичными) сценами. Смотрите [qcomp](#) ниже, чтобы узнать некоторые идеи о том, как можно это распределение настроить по Вашему вкусу.

Более того, два прохода занимают не двойное время по сравнению с одним. Вы можете настроить опции первого прохода на более быструю скорость и низкое качество. Если хорошо выберете опции, Вы получите очень быстрый первый проход. Полученное качество во втором проходе будет несколько ниже, потому что предсказание размера менее точно, но разница в качестве обычно слишком мала, чтобы быть заметной. Попробуйте, например, добавить `subq=1:frameref=1` в `x264encopts` первого прохода. Затем, при втором проходе, используйте более медленные, с лучшим качеством опции: `subq=6:frameref=15:partitions=all:me=umh`

- **Кодирование в три прохода?** `x264` предоставляет возможность делать желаемое количество последовательных проходов. Если Вы указали `pass=1` при первом проходе, используйте затем `pass=3` в последующем проходе, этот проход будет одновременно читать статистику предыдущего прохода и записывать свою собственную. Дополнительный проход, следующий за этим, будет иметь очень хорошую основу для осуществления очень точных предсказаний размеров кадров при выбранном квантователе. На практике, общее улучшение качества от использования этого режима близко к нулю и, вполне возможно, третий проход приведет к немного худшему глобальному PSNR, чем у предыдущего прохода. При обычном использовании три прохода помогают, если Вы при двух проходах получаете либо плохое предсказание битпотока, либо плохо выглядящие переходы между сценами. Это отчасти то, что наверняка будет происходить на очень коротких клипах. Существуют также особые случаи, когда три (или более) проходом удобны для продвинутых пользователей, но, для краткости, это руководство не включает в себя описание этих особых случаев.

- **qcomp:** `qcomp` управляет соотношением количества бит, отданных "дорогим" высокодинамичным и "дешевым" малодинамичным кадрам. Один крайний случай: `qcomp=0`, предназначен для истинно постоянного битпотока. Обычно это сделает высокодинамичные сцены выглядящими просто ужасно, в то время как малодинамичные сцены будут, возможно, выглядеть абсолютно великолепно, но при этом будут использовать во много раз больший битпоток, чем им необходимо, чтобы выглядеть лишь превосходно. Другая крайность: `qcomp=1`, добивается примерно одинакового параметра квантования (QP). Постоянный QP не выглядит плохо, но большинство людей думают, что более разумно частично снизить битпоток в сильно дорогих сценах (где потеря качества не очень заметна) и перераспределить их в сцены, которые легче закодировать с отличным качеством. `qcomp` по умолчанию установлена в 0.6, что по мнению многих людей может быть несколько мало (также часто используется 0.7-0.8).
- **keyint:** `keyint` — единственная возможность выбора между удобством перемещения по файлу и эффективностью кодирования. По-умолчанию `keyint` установлена в 250. В материале с 25fps это гарантирует возможность перемещения с точностью до 10 секунд. Если Вы считаете, что более важным и полезным будет перемещение с точностью до 5 секунд, установите `keyint=125`; это немного ухудшит качество/битпоток. Если Вы заботитесь только о качестве, но не о перемещаемости, Вы можете установить значение этой опции в более высокое значение (понимая, что улучшение будет убывающим, вплоть до исчезающе малого или даже нулевого). Видео поток по-прежнему будет иметь точки перемещения, пока в нем есть какие-то изменения сцен.
- **deblock:** Этот раздел может быть несколько спорным. H.264 определяет простую процедуру удаления блочности в I-блоках, которая использует предустановленные степени обработки и пороговые значения в зависимости от QP рассматриваемого блока. По-умолчанию, блоки с высоким QP обрабатываются сильнее, а в блоках с низким QP удаление блочности вообще не производится. Предустановленные степени обработки, определенные стандартом, тщательно подобраны и имеют хорошие шансы быть PSNR-оптимальными для любого видео, которое Вы пытаетесь кодировать. Опция `deblock` позволяет указать смещения предустановленных пороговых значений деблокинга. Похоже, многие думают, что хорошей идеей является значительное уменьшение силы воздействия фильтра деблокинга (читай, -3). Это, однако, почти никогда не является хорошей идеей, и, люди, это делающие, в большинстве случаев не совсем хорошо понимают, как работает удаление блочности по умолчанию. Первая и самая важная вещь, которую нужно знать о in-loop фильтре удаления блочности состоит в том, что пороговые значения по умолчанию практически всегда PSNR-оптимальны. В редких случаях, где они неоптимальны, идеальное смещение будет плюс минус 1. Изменение параметров деблокинга на большие значения фактически гарантирует ухудшение PSNR. Усиление фильтра размажет больше деталей; ослабление — оставит больше квадратов. По определению плохая идея уменьшать пороги деблокинга, если Ваш исходный материал в основном имеет небольшую пространственную сложность (т.е. не имеет множества деталей или шума). In-loop фильтр делает весьма неплохую работу по сокрытию появляющихся артефактов. Однако, если исходный материал имеет высокую пространственную сложность, артефакты будут практически незаметны. Это происходит потому, что ореолы имеют склонность выглядеть как детали или шум. Зрительное восприятие легко замечает отсутствие деталей, но ему не так легко обратить внимание на неверно изображенный шум. Когда речь идет о субъективном качестве, шум и детали в некоторой степени взаимозаменяемы. Уменьшая силу фильтра удаления блочности, Вы, скорее всего, увеличиваете ошибку, добавляя ореолы, но глаз этого не замечает, поскольку он путает артефакты с деталями. Однако, это **по-прежнему** не оправдывает уменьшение силы фильтра. Вы в большинстве случаев можете получить более качественный шум при помощи постобработки. Если результат кодирования при помощи H.264 выглядит слишком смазанным или размытым, попробуйте поиграть с `-vf noise`, при воспроизведении закодированного фильма. `-vf noise=8a:4a` должна скрыть большинство мелких артефактов. Ее результат почти наверняка будет выглядеть лучше, чем полученный при помощи махинаций с фильтром удаления блочности.

7.5.2. Примеры настроек кодирования

Последующие настройки — это примеры различных комбинаций опций кодирования, которые влияют на соотношения скорость-качество при той же величине целевого битпотока.

Все настройки кодирования проверялись на тестовом видео 720x448 @30000/1001 fps с целевым битпотокм 900кбит/сек, на машине AMD-64 3400+ с 2400 МГц и 64-х битном режиме. Для каждой настройки кодирования указаны измеренная скорость кодирования (в кадрах в секунду) и потеря PSNR (в дБ) по сравнению с настройкой "очень высокое качество". Поймите, пожалуйста, что в зависимости от Вашего материала, типа машины, прогресса разработки Вы можете получить сильно отличающиеся результаты.

Описание	Опции кодирования	скорость (в fps)	Относительная потеря PSNR (в дБ)
Очень высокое качество	<code>subq=6:partitions=all:8x8dct:me=umh:frameref=5:bframes=3:b_pyramid=normal:weight_b</code>	6fps	0дБ
Высокое качество	<code>subq=5:8x8dct:frameref=2:bframes=3:b_pyramid=normal:weight_b</code>	13fps	-0.89дБ
Быстро	<code>subq=4:bframes=2:b_pyramid=normal:weight_b</code>	17fps	-1.48дБ

7.6. Кодирование семейством кодеков Video For Windows

Video for Windows предоставляет простое кодирование при помощи бинарных видео кодеков. Вы можете кодировать следующими кодеками (если у Вас есть другие, сообщите нам!)

Имейте в виду, что поддержка этой возможности очень экспериментальная и некоторые кодеки могут не работать корректно. Некоторые кодеки могут работать только в определенных пространствах цветов, попробуйте `-vf format=bgr24` и `-vf format=yuy2`, если кодек выдает ошибку или кодирует неверно.

7.6.1. Поддерживаемые кодеки Video for Windows

Имя файла с видео кодеком	Описание (FourCC)	md5sum	Комментарий
aslcodec_vfw.dll	Alparysoft vfw кодек без потерь (ASLC)	608af234a6ea4d90cdc7246af5f3f29a	
avimsh.dll	AVImszh (MSZH)	253118fe1eedea04a95ed6e5f4c28878	необходим <code>-vf format</code>
avizlib.dll	AVIzlib (ZLIB)	2f1cc76bbcf6d77d40d0e23392fa8eda	
divx.dll	DivX4Windows-VFW	acf35b2fc004a89c829531555d73f1e6	
huffyuv.dll	HuffYUV (без потерь) (HFYU)	b74695b50230be4a6ef2c4293a58ac3b	
iccvid.dll	Cinepak Video (cvid)	cb3b7ee47ba7dbb3d23d34e274895133	
icmw_32.dll	Motion Wavelets (MWV1)	c9618a8fc73ce219ba918e3e09e227f2	
jp2avi.dll	ImagePower MJPEG2000 (IPJ2)	d860a11766da0d0ea064672c6833768b	<code>-vf flip</code>
m3jp2k32.dll	Morgan MJPEG2000 (MJ2C)	f3c174edcbaef7cb947d6357cdfde7ff	
m3jpeg32.dll	Morgan Motion JPEG Codec (MJPG)	1cd13fff5960aa2aae43790242c323b1	
mpg4c32.dll	Microsoft MPEG-4 v1/v2	b5791ea23f33010d37ab8314681f1256	
tscavid.dll	TechSmith Camtasia	8230d8560c41d444f249802a2700d	ошибка shareware в

Имя файла с видео кодеком	Описание (FourCC)	md5sum	Комментарий
	Screen Codec (TSCC)	1d5	windows
vp31vfw.dll	On2 Open Source VP3 Codec (VP31)	845f3590ea489e2e45e876ab107ee7d2	
vp4vfw.dll	On2 VP4 Personal Codec (VP40)	fc5480a482ccc594c2898dcc4188b58f	
vp6vfw.dll	On2 VP6 Personal Codec (VP60)	04d635a364243013898fd09484f913fb	
vp7vfw.dll	On2 VP7 Personal Codec (VP70)	cb4cc3d4ea7c94a35f1d81c3d750bc8d	-ffourcc VP70
ViVD2.dll	SoftMedia ViVD V2 Vfw кодек (GXVE)	a7b4bf5cac630bb9262c3f80d8a773a1	
msulvc06.DLL	MSU кодек без потерь (MSUD)	294bf9288f2f127bb86f00bfcc9ccdda	Может декодироваться Window Media Player, но не MPlayer (пока).
camcodec.dll	CamStudio lossless video codec (CSCD)	0efe97ce08bb0e40162ab15ef3b45615	sf.net/projects/camstudio

Первый столбец содержит имена кодеков, которые должны указываться после параметра `codec`, например: `-xvfwopts codec=divx.dll` FourCC код, используемый каждым кодеком, указан в скобках.

Пример конвертации ISO DVD ролика в VP6 flash видео файл с использованием `compdata` настроек битпотока:

```
mencoder -dvd-device zeiram.iso dvd:///7 -o trailer.flv \
```

```
-ovc vfw -xvfwopts codec=vp6vfw.dll:compdata=onepass.mcf -oac mp3lame \
-lameopts cbr:br=64 -af lavcresample=22050 -vf yadif,scale=320:240,flip \
-of lavf
```

7.6.2. Использование `vfw2menc` для создания файла настроек кодека.

Для кодирования кодеками Video for Windows, Вам потребуется установить величину битпотока и другие опции. Известно, что это работает на x86 и для *NIX, и для Windows.

Во-первых, Вы должны собрать программу `vfw2menc`. Она находится в подкаталоге **TOOLS** дерева исходников MPlayer. Для сборки под Linux, это можно сделать, воспользовавшись Wine: `winegcc vfw2menc.c -o vfw2menc -lwinmm -lole32`

Для сборки под Windows в MinGW или Cygwin используйте:

```
gcc vfw2menc.c -o vfw2menc.exe -lwinmm -lole32
```

Для сборки в MSVC Вам потребуется `getopt`. `Getopt` можно найти в исходном архиве `vfw2menc`, доступном в: Проект [MPlayer на win32](#).

Далее следует пример использования с VP6 кодеком.

```
vfw2menc -f VP62 -d vp6vfw.dll -s firstpass.mcf
```

Это откроет диалоговое окно кодека VP6. Повторите этот шаг для второго прохода, указав `-s secondpass.mcf`.

Пользователи Windows могут использовать `-xvfwopts codec=vp6vfw.dll:compdata=dialog`, для показа окна настроек кодека перед началом кодирования.

7.7. Использование MEncoder для создания совместимых с QuickTime файлов

7.7.1. Зачем необходимо создавать совместимые с QuickTime файлы?

Есть несколько причин, по которым создание QuickTime-совместимых файлов может быть желательным.

- Вы хотите, чтобы любой компьютерно неграмотный человек мог смотреть результат Вашего кодирования на большинстве платформ (Windows, Mac OS X, Unices ...).
- QuickTime позволяет воспользоваться преимуществами большего числа возможностей аппаратного и программного ускорения на Mac OS X, чем платформо-независимые плееры наподобие MPlayer или VLC. Это означает, что Ваше кодирование имеет шансы плавно воспроизводиться на старых машинах, оснащённых G4.

- QuickTime 7 поддерживает кодек нового поколения H.264, который даёт существенно лучшее качество изображения, чем предыдущие поколения кодеков (MPEG-2, MPEG-4 ...).

7.7.2. Ограничения QuickTime 7

QuickTime 7 поддерживает H.264 видео и AAC аудио, но не поддерживает их мультимплексирование в формат контейнера AVI. Однако, Вы можете использовать MEncoder для кодирования видео и аудио, а потом использовать внешнюю программу, такую как mp4creator (часть [пакета MPEG4!](#)) для ремультимплексирования видео и аудио дорожек в контейнер MP4.

Поддержка H.264 в QuickTime ограничена, так что Вам придётся отказаться от нескольких продвинутых возможностей. Если Вы кодируете видео с возможностями, не поддерживаемыми QuickTime 7, плееры, основанные на QuickTime, покажут Вам милый белый экран вместо ожидаемого Вами видео.

- **В-кадры:** QuickTime 7 поддерживает максимум 1 В-кадр, т.е. `-x264encopts bframes=1`. Это означает, что `b_pyramid` и `weight_b` не дадут эффекта, поскольку им необходимо, чтобы `bframes` было больше 1.
- **Макроблоки:** QuickTime 7 не поддерживает 8x8 DCT макроблоки. Эта опция (`8x8dct`) выключена по умолчанию, так что просто удостоверьтесь, что явно её не задали. Это также означает, что опция `i8x8` будет бесполезна, т.к. ей необходима `8x8dct`.
- **Коэффициент соотношения сторон:** QuickTime 7 не поддерживает информацию SAR (коэффициент пропорций пиксела, `sample aspect ratio`) в MPEG-4 файлах; он предполагает SAR=1. Прочтите [раздел о масштабировании](#) для обхода проблемы.

7.7.3. Обрезание

Предположим, что Вы хотите сделать рип свежкупленной копии "Хроник Нарнии" и Ваш регион DVD 1, что означает, что это NTSC. Пример ниже будет также применим к PAL, за исключением того, что Вам нужно будет опустить `-ofps 24000/1001` и использовать слегка отличающиеся размеры для `crop` и `scale`.

После запуска `mplayer dvd://1`, Вы следуете процессу, описанному в разделе [Как работать с телесином и чересстрочной развёрткой на NTSC DVD](#) и обнаруживаете, что это 24000/1001 fps видео с построчной развёрткой. Это несколько упрощает обработку, поскольку Вам не нужно использовать фильтр обратного телесина, такой как `pullup`, или фильтр деинтерлейса, такой как `yadif`.

Затем Вам необходимо усечь чёрные полосы сверху и снизу видео, как описано в [ЭТОМ](#) разделе.

7.7.4. Масштабирование

Следующий шаг действительно душераздирающий. QuickTime 7 не поддерживает MPEG-4 видео с коэффициентом соотношения сторон пиксела, отличным от 1. Так что Вам придётся масштабировать видео либо в сторону увеличения (что впустую потратит много места на диске), либо в сторону уменьшения (что приведёт к потере некоторых деталей источника) для квадратизации пикселов. Какой бы способ Вы не выбрали, это будет крайне неэффективным, но не может быть опущено, если Вы хотите, чтоб Ваше видео воспроизводилось с помощью QuickTime 7. MEncoder может применить необходимое увеличивающее или уменьшающее масштабирование, если ему указать `-vf scale=-10:-1` или `-vf scale=-1:-10` соответственно. Это отмасштабирует Ваше видео до корректной ширины для усечённой высоты, округлённой до ближайшего множителя 16 для оптимального сжатия. Помните, что если производите обрезание, то нужно сперва обрезать, а лишь затем масштабировать:

```
-vf crop=720:352:0:62,scale=-10:-1
```

7.7.5. A/V синхронизация

Поскольку Вы будете мультимплексировать в другой контейнер, Вы должны всегда использовать опцию `harddup`, чтобы убедиться, что дублирующиеся кадры будут действительно дублироваться в полученном видео. Без этой опции MEncoder будет просто располагать маркер в видеопотоке о том, что кадр был повторен, и будет полагаться на то, что клиентское программное обеспечение покажет кадр дважды. К сожалению, это "мягкое дублирование" не переживает ремультимплексирование, в результате чего аудио будет постепенно терять синхронизацию с видео.

В итоге, цепочка фильтров выглядит следующим образом:

```
-vf crop=720:352:0:62,scale=-10:-1,harddup
```

7.7.6. Битпоток

Как обычно, выбор битпотока зависит от технических свойств исходного материала, как объясняется [здесь](#), как, впрочем, и от личного вкуса. Этот фильм обладает небольшим количеством движения и большим количеством деталей, но H.264 видео хорошо выглядит на

существенно меньших битпотоках, чем XviD или другие MPEG-4 кодеки. После длительного экспериментирования, автор данного руководства решил кодировать фильм на 900 кбит/сек, и считает, что он выглядит очень хорошо. Вы можете уменьшить битпоток, если Вам нужно сохранить больше места, или увеличить, если Вам нужно улучшить качество.

7.7.7. Пример кодирования

Теперь Вы готовы к кодированию видео. Поскольку Вы заботитесь о качестве, Вы, разумеется, будете делать двупроходное кодирование. Для некоторого сокращения времени кодирования, Вы можете указать опцию **turbo** при первом проходе; это уменьшит **subq** и **frameref** до 1. Чтобы сохранить немного места на диске, Вы можете использовать параметр **ss** для отрезания первых нескольких секунд видео. (Я обнаружил, что, в частности, у данного фильма есть 32 секунды титров и логотипов.) **bframes** может быть 0 или 1. остальные опции описаны в разделе [Кодирование кодеком x264](#) и на map странице.

```
mencoder dvd://1 -o /dev/null -ss 32 -ovc x264 \
-x264encopts pass=1:turbo:bitrate=900:bframes=1:\
me=umh:partitions=all:trellis=1:qp_step=4:qcomp=0.7:direct_pred=auto:keyint=300 \
-vf crop=720:352:0:62,scale=-10:-1,harddup \
-oac faac -faacopts br=192:mpeg=4:object=2 -channels 2 -srate 48000 \
-ofps 24000/1001
```

Если у Вас многопроцессорная машина, не упустите шанс значительно ускорить кодирование задействованием [многопоточного режима x264](#), добавив **threads=auto** в **x264encopts** в командной строке.

Второй проход выполняется аналогично, за исключением того, что Вам нужно указать выходной файл и установить **pass=2**.

```
mencoder dvd://1 -o нарния.avi -ss 32 -ovc x264 \
-x264encopts pass=2:turbo:bitrate=900:frameref=5:bframes=1:\
me=umh:partitions=all:trellis=1:qp_step=4:qcomp=0.7:direct_pred=auto:keyint=300 \
-vf crop=720:352:0:62,scale=-10:-1,harddup \
-oac faac -faacopts br=192:mpeg=4:object=2 -channels 2 -srate 48000 \
-ofps 24000/1001
```

Получившееся AVI должно хорошо воспроизводиться в MPlayer, но, конечно же, QuickTime не сможет его воспроизвести, т.к. не поддерживает H.264, мультимплексированный в AVI. Так что следующий шаг — ремультимплексирование видео в контейнер MP4.

7.7.8. Ремультимплексирование в MP4

Существует несколько способов ремультимплексирования AVI файлов в MP4. Вы можете использовать **mp4creator**, являющийся частью [пакета MPEG4IP](#).

Сперва демультимплексируйте AVI в отдельные аудио и видео потоки с помощью MPlayer.

```
mplayer нарния.avi -dumpaudio -dumpfile нарния.aac
mplayer нарния.avi -dumpvideo -dumpfile нарния.h264
```

Имена файлов важны; для **mp4creator** необходимо, чтобы AAC аудио потоки назывались **.aac** и H.264 видео потоки назывались **.h264**.

Теперь используйте **mp4creator** для создания нового MP4 файла из аудио и видео потоков.

```
mp4creator -create=нарния.aac нарния.mp4
mp4creator -create=нарния.h264 -rate=23.976 нарния.mp4
```

В отличии от этапа кодирования, Вам нужно указать частоту кадров как десятичную (например, 23.976), а не целую (например, 24000/1001) дробь.

Теперь файл **нарния.mp4** должен проигрываться с помощью любого QuickTime 7 приложения, например, QuickTime Player или iTunes. Если Вы планируете просмотр видео в вэб-браузере с помощью плагина QuickTime, Вам также необходимо модифицировать фильм таким образом, чтобы плагин QuickTime мог начать его воспроизведение ещё во время загрузки. **mp4creator** может создать эти вспомогательные дорожки (т.н. **hint tracks**):

```
mp4creator -hint=1 нарния.mp4
mp4creator -hint=2 нарния.mp4
mp4creator -optimize нарния.mp4
```

Вы можете проверить полученный результат, чтобы убедиться, что вспомогательные дорожки были успешно созданы.

```
mp4creator -list нарния.mp4
```

Вы должны увидеть список дорожек: 1 аудио, 1 видео и 2 вспомогательных.

```
Track Type Info
1 audio MPEG-4 AAC LC, 8548.714 secs, 190 kbps, 48000 Hz
2 video H264 Main@5.1, 8549.132 secs, 899 kbps, 848x352 @ 23.976001 fps
3 hint Payload mpeg4-generic for track 1
```

4 hint Payload H264 for track 2

7.7.9. Добавление тегов метаданных

Если Вы хотите добавить в видео теги, которые отображаются в iTunes, Вы можете использовать [AtomicParsley](#).

```
AtomicParsley нарния.mp4 --metaEnema --title "The Chronicles of Narnia" --year 2005 --stik Movie --freefree --overWrite
```

Опция `--metaEnema` удаляет любые существующие метаданные (mp4creator вставляет своё название в тег "утилиты кодирования") и `--freefree` высвобождает место, оставшееся от удалённых метаданных. Опция `--stik` устанавливает тип видео (например, Movie или TV Show), который используется iTunes для группировки родственных видеофайлов. Опция `--overWrite` перезаписывает исходный файл; без неё AtomicParsley создаст новый файл с автоматическим именем в том же каталоге и оставит исходный файл нетронутым.

7.8. Использование MEncoder для создания VCD/SVCD/DVD-совместимых файлов.

7.8.1. Ограничения формата

MEncoder способен создавать MPEG файлы VCD, SVCD и DVD формата, используя библиотеку libavcodec. Эти файлы затем могут быть использованы совместно с [vcdimager](#) или [dvdauthor](#) для создания дисков, которые будут воспроизводиться на стандартном видео проигрывателе. DVD, SVCD, и VCD форматы обладают жесткими ограничениями. Доступен только небольшой выбор размеров и пропорций кодируемого изображения. Если Ваш фильм пока не удовлетворяет этим ограничениям, придется изображение масштабировать, обрезать или добавлять к нему черные полосы, чтобы добиться совместимости.

7.8.1.1. Ограничения форматов

Формат	Разрешение	Вид. Кодек	Вид. Битпоток	Дискретизация	Ауд. Кодек	Ауд. Битпоток	FPS	Пропорции
NTSC DVD	720x480, 704x480, 352x480, 352x240	MPEG-2	9800 кбит/с	48000 Гц	AC-3, PCM	1536 кбит/с (макс.)	30000/1001, 24000/1001	4:3, 16:9 (только для 720x480)
NTSC DVD	352x240 <a>[a]	MPEG-1	1856 кбит/с	48000 Гц	AC-3, PCM	1536 кбит/с (макс.)	30000/1001, 24000/1001	4:3, 16:9
NTSC SVCD	480x480	MPEG-2	2600 кбит/с	44100 Гц	MP2	384 кбит/с (макс.)	30000/1001	4:3
NTSC VCD	352x240	MPEG-1	1150 кбит/с	44100 Гц	MP2	224 кбит/с	24000/1001, 30000/1001	4:3
PAL DVD	720x576, 704x576, 352x576, 352x288	MPEG-2	9800 кбит/с	48000 Гц	MP2, AC-3, PCM	1536 кбит/с (макс.)	25	4:3, 16:9 (только для 720x576)
PAL DVD	352x288 <a>[a]	MPEG-1	1856 кбит/с	48000 Гц	MP2, AC-3, PCM	1536 кбит/с (макс.)	25	4:3, 16:9
PAL SVCD	480x576	MPEG-2	2600 кбит/с	44100 Гц	MP2	384 кбит/с (макс.)	25	4:3
PAL VCD	352x288	MPEG-1	1152 кбит/с	44100 Гц	MP2	224 кбит/с	25	4:3

[\[a\]](#) Эти разрешения редко используются для DVD, поскольку имеют довольно низкое качество.

Если Ваш фильм имеет пропорции 2.35:1 (большинство современных фильмов с обилием движения), для создания DVD или VCD придется добавить черные полосы или обрезать фильм до 16:9. Добавляя черные полосы, пытайтесь выровнять их размеры на границу в 16 пикселей,

чтобы минимизировать влияние на производительность кодирования. К счастью, DVD имеет достаточно избыточный битпоток, чтобы не сильно беспокоиться об эффективности кодирования, но SVCD и VCD весьма ограничены в битпотоке и требуют определенных усилий для достижения приемлемого качества.

7.8.1.2. Ограничения на размер GOP

DVD, VCD, и SVCD также ограничивают Вас относительно низкими размерами GOP (Group of Pictures, Группа Изображений). Для материала с 30 fps максимальный допустимый размер GOP равен 18. Для 25 или 24 fps, максимум равен 15. Размер GOP устанавливается опцией `keyint`.

7.8.1.3. Ограничения на битпоток

VCD видео должно быть CBR с 1152 кбит/с. Это сильное ограничение усугубляется, к тому же, чрезвычайно низким размером `vbv` буфера, равным 327 килобит. SVCD допускает различные значения видео битпотока вплоть до 2500 кбит/с и не так сильно стесняющий размер `vbv` буфера, равный 917 килобит. У DVD видео битпоток может свободно меняться вплоть до 9800 kbps (хотя обычный поток равен примерно половине этого значения), а размер `vbv` буфера равен 1835 килобит.

7.8.2. Опции вывода

У MEncoder есть опции, управляющие выходным форматом. Используя их, можно дать указание создать файл корректного типа.

Для VCD и SVCD опции называются `xvcd` и `xsvcd`, потому что они являются расширенными форматами. Они не полностью совместимы, в основном, потому что не содержат смещений развёртки. Если нужно создать образ SVCD, следует передать выходной файл программе [vcdimager](#).

VCD:

```
-of mpeg -mpegopts format=xvcd
```

SVCD:

```
-of mpeg -mpegopts format=xsvcd
```

DVD (с временными метками на каждом кадре, если возможно):

```
-of mpeg -mpegopts format=dvd:tsaf
```

DVD с NTSC Pullup:

```
-of mpeg -mpegopts format=dvd:tsaf:telecine -ofps 24000/1001
```

Это делает возможным кодирование 24000/1001 fps построчного содержимого с частотой 30000/1001 fps, с одновременным сохранением совместимости с DVD.

7.8.2.1. Пропорции

Аргумент `aspect` в `-lavcopts` используется для кодирования коэффициента пропорций файла. Коэффициент пропорций используется в процессе воспроизведения для восстановления правильного размера видео.

16:9 or "Широкоэкранный"

```
-lavcopts aspect=16/9
```

4:3 or "Полноэкранный"

```
-lavcopts aspect=4/3
```

2.35:1 or "Кинематографический" NTSC

```
-vf scale=720:368,expand=720:480 -lavcopts aspect=16/9
```

Для вычисления правильного размера масштабирования используйте расширенную ширину NTSC $854/2.35 = 368$

2.35:1 or "Кинематографический" PAL

```
-vf scale=720:432,expand=720:576 -lavcopts aspect=16/9
```

Для вычисления правильного размера масштабирования используйте расширенную ширину PAL $1024/2.35 = 432$

7.8.2.2. Сохранение A/V синхронизации

Для того, чтобы сохранять аудио/видео синхронизацию на протяжении всего кодирования, MEncoder должен выбрасывать или дублировать кадры. Это довольно неплохо работает при мультиплексировании в AVI файл, но с другими мультиплексорами, такими как MPEG, почти гарантировано приведет к нарушению A/V синхронизации. Для избежания подобных проблем, необходимо добавить видео фильтр `harddup` в конец цепочки фильтров. Дополнительную техническую информацию о `harddup` можно найти в разделе [Улучшение мультиплексирования и надежности A/V синхронизации](#) или в тап руководстве.

7.8.2.3. Преобразование частоты дискретизации

Если частота дискретизации в оригинальном файле не совпадает с требуемой в целевом формате, необходимо преобразование. Его можно осуществить, совместно используя опцию `-srate` и аудио фильтр `-af lavcresample`.

DVD:

-srate 48000 -af lavcresample=48000

VCD и SVCD:

-srate 44100 -af lavcresample=44100

7.8.3. Использование libavcodec для VCD/SVCD/DVD кодирования

7.8.3.1. Введение

Используя соответствующие опции, можно применять libavcodec для создания VCD/SVCD/DVD совместимого видео.

7.8.3.2. lavcopts

Это список полей в `-lavcopts`, которые может потребоваться изменить, чтобы создать совместимый фильм для VCD, SVCD или DVD:

- **acodec**: `mp2` для VCD, SVCD, или PAL DVD; `ac3` наиболее часто используется для DVD. Для DVD также может использоваться PCM, но это, по большей части, бесполезная трата свободного места. Имейте в виду, что MP3 аудио не совместимо ни с одним из этих форматов, но, как бы то ни было, часто проигрыватели не испытывают никаких проблем с его воспроизведением.
- **abitrte**: 224 для VCD; вплоть до 384 для SVCD; вплоть до 1536 для DVD, но распространенным является диапазон значений от 192 кбит/с для стерео до 384 кбит/с для 5.1 звука.
- **vcodec**: `mpeg1video` для VCD; `mpeg2video` для SVCD; `mpeg2video` обычно используется для DVD, но Вы также можете использовать `mpeg1video` для CIF разрешений.
- **keyint**: Используется для установки размера GOP. 18 для 30fps материала или 15 для 25/24 fps материала. Коммерческие изготовители, похоже, предпочитают значение интервала ключевых кадров, равное 12. Можно значительно увеличить это значение и все еще сохранять совместимость с большинством проигрывателей. `keyint` равное 25 не должно вызывать никаких проблем.
- **vrc_buf_size**: 327 для VCD, 917 для SVCD и 1835 для DVD.
- **vrc_minrate**: 1152 для VCD. Может не указываться для SVCD и DVD.
- **vrc_maxrate**: 1152 для VCD; 2500 для SVCD; 9800 для DVD. Для SVCD и DVD Вы, возможно, пожелаете использовать меньшие значения в зависимости от Ваших личных требований и предпочтений.
- **vbtrate**: 1152 для VCD; вплоть до 2500 для SVCD; вплоть до 9800 для DVD. Для двух последних форматов, `vbtrate` следует установить на основании личных предпочтений. Например, если Вы настаиваете на размещении 20 или около того часов видео на DVD, можете использовать `vbtrate=400`. Качество получившегося видео, возможно, будет довольно плохим. Если Вы пытаетесь выжать максимально возможное качество на DVD, используйте `vbtrate=9800`, но имейте в виду, что это ограничит Вас менее чем одним часом видео на однослойном DVD.
- **vstrict**: `vstrict=0` следует использовать для создания DVD. Без этой опции MEncoder создает поток, который не может быть корректно декодирован некоторыми аппаратными DVD проигрывателями.

7.8.3.3. Примеры

Это обычный минимальный набор `-lavcopts` для кодирования видео:

VCD:

```
-lavcopts vcodec=mpeg1video:vrc_buf_size=327:vrc_minrate=1152:\
vrc_maxrate=1152:vbtrate=1152:keyint=15:acodec=mp2
```

SVCD:

```
-lavcopts vcodec=mpeg2video:vrc_buf_size=917:vrc_maxrate=2500:vbtrate=1800:\
keyint=15:acodec=mp2
```

DVD:

```
-lavcopts vcodec=mpeg2video:vrc_buf_size=1835:vrc_maxrate=9800:vbtrate=5000:\
keyint=15:vstrict=0:acodec=ac3
```

7.8.3.4. Расширенные опции

Для достижения более высокого качества кодирования, Вы также можете добавить опции `lavcopts`, улучшающие качество, такие как `trell`, `mbd=2` и другие. Обратите внимание, что `qpel` и `v4mv`, часто полезные с MPEG-4, не применимы к MPEG-1 или MPEG-2. Также, если Вы хотите выполнить очень высококачественное кодирование DVD, может быть полезным добавление `dc=10` в `lavcopts`. Это может помочь подавить появление блоков в однородно окрашенных областях. Подводя итог, вот пример настроек `lavcopts` для высококачественного DVD:

```
-lavcopts vcodec=mpeg2video:vrc_buf_size=1835:vrc_maxrate=9800:vbtrate=8000:\
keyint=15:trell:mbd=2:precmp=2:subcmp=2:cmp=2:dia=-10:predia=-10:cbp:mv0:\
vqmin=1:lmin=1:dc=10:vstrict=0
```

7.8.4. Кодирование звука

VCD и SVCD поддерживают звук MPEG-1 layer II, используя одну из toolame, twolame, или MP2 libavcodec кодировщик. libavcodec MP2 не так хорош, как остальные две библиотеки, однако, он должен быть всегда доступен для использования. VCD поддерживает только звук с постоянным битпоток (CBR), в то время как SVCD также поддерживает и переменный (VBR). Будьте осторожны, используя VBR, поскольку некоторые плохие аппаратные проигрыватели могут не очень хорошо его поддерживать.

Для DVD звука используется AC-3 кодек из libavcodec.

7.8.4.1. toolame

Для VCD и SVCD:

```
-oac toolame -toolameopts br=224
```

7.8.4.2. twolame

Для VCD и SVCD:

```
-oac twolame -twolameopts br=224
```

7.8.4.3. libavcodec

Для 2 канального DVD звука:

```
-oac lavc -lavcopts acodec=ac3:abitrage=192
```

Для DVD с 5.1 звуком:

```
-channels 6 -oac lavc -lavcopts acodec=ac3:abitrage=384
```

Для VCD и SVCD:

```
-oac lavc -lavcopts acodec=mp2:abitrage=224
```

7.8.5. Собирая все вместе

Этот раздел демонстрирует некоторые полные команды для создания VCD/SVCD/DVD совместимого видео.

7.8.5.1. PAL DVD

```
mencoder -oac lavc -ovc lavc -of mpeg -mpegopts format=dvd:tsaf \
```

```
-vf scale=720:576,harddup -srate 48000 -af lavcresample=48000 \  
-lavcopts vcodec=mpeg2video:vrc_buf_size=1835:vrc_maxrate=9800:vbitrate=5000:\
```

```
keyint=15:vstrict=0:acodec=ac3:abitrage=192:aspect=16/9 -ofps 25 \
```

```
-o фильм.mpg фильм.avi
```

7.8.5.2. NTSC DVD

```
mencoder -oac lavc -ovc lavc -of mpeg -mpegopts format=dvd:tsaf \
```

```
-vf scale=720:480,harddup -srate 48000 -af lavcresample=48000 \  
-lavcopts vcodec=mpeg2video:vrc_buf_size=1835:vrc_maxrate=9800:vbitrate=5000:\
```

```
keyint=18:vstrict=0:acodec=ac3:abitrage=192:aspect=16/9 -ofps 30000/1001 \
```

```
-o фильм.mpg фильм.avi
```

7.8.5.3. PAL AVI, содержащий AC-3 звук, в DVD

Если исходный материал уже содержит AC-3 звук, используйте -oac copy вместо перекодирования.

```
mencoder -oac copy -ovc lavc -of mpeg -mpegopts format=dvd:tsaf \
```

```
-vf scale=720:576,harddup -ofps 25 \  
-lavcopts vcodec=mpeg2video:vrc_buf_size=1835:vrc_maxrate=9800:vbitrate=5000:\
```

```
keyint=15:vstrict=0:aspect=16/9 -o фильм.mpg фильм.avi
```

7.8.5.4. NTSC AVI, содержащий AC-3 звук, в DVD

Если исходный материал уже содержит AC-3 звук и является NTSC @ 24000/1001 fps:

```
mencoder -oac copy -ovc lavc -of mpeg -mpegopts format=dvd:tsaf:telecine \
```

```
-vf scale=720:480,harddup -lavcopts vcodec=mpeg2video:vrc_buf_size=1835:\
```

```
vrc_maxrate=9800:vbitrate=5000:keyint=15:vstrict=0:aspect=16/9 -ofps 24000/1001 \
```

```
-o фильм.mpg фильм.avi
```

7.8.5.5. PAL SVCD

```
mencoder -oac lavc -ovc lavc -of mpeg -mpegopts format=xsvcd -vf \
```

```
scale=480:576,harddup -srate 44100 -af lavcresample=44100 -lavcopts \  
vcodec=mpeg2video:mbd=2:keyint=15:vrc_buf_size=917:vrc_minrate=600:\
```

```
vbitrate=2500:vrc_maxrate=2500:acodec=mp2:abitrage=224:aspect=16/9 -ofps 25 \
```

```
-o фильм.mpg фильм.avi
```

7.8.5.6. NTSC SVCD

```
mencoder -oac lavc -ovc lavc -of mpeg -mpegopts format=xsvcd -vf \
```

```
scale=480:480,harddup -srate 44100 -af lavcresample=44100 -lavcopts \  
vcodec=mpeg2video:mbd=2:keyint=18:vrc_buf_size=917:vrc_minrate=600:\
```

```
vbitrate=2500:vrc_maxrate=2500:acodec=mp2:abitrage=224:aspect=16/9 -ofps 30000/1001 \
```

```
-o фильм.mpg фильм.avi
```

7.8.5.7. PAL VCD

```
mencoder -oac lavc -ovc lavc -of mpeg -mpegopts format=xvcd -vf \
```

```
scale=352:288,harddup -srate 44100 -af lavcresample=44100 -lavcopts \  
vcodec=mpeg1video:keyint=15:vrc_buf_size=327:vrc_minrate=1152:\
```

```
vbitrate=1152:vrc_maxrate=1152:acodec=mp2:abitrage=224:aspect=16/9 -ofps 25 \
```

```
-o фильм.mpg фильм.avi
```

7.8.5.8. NTSC VCD

```
mencoder -oac lavc -ovc lavc -of mpeg -mpegopts format=xvcd -vf \
```

```
scale=352:240,harddup -srate 44100 -af lavcresample=44100 -lavcopts \  
vcodec=mpeg1video:keyint=18:vrc_buf_size=327:vrc_minrate=1152:\
```

```
vbitrate=1152:vrc_maxrate=1152:acodec=mp2:abitrage=224:aspect=16/9 -ofps 30000/1001 \
```

```
-o фильм.mpg фильм.avi
```

[1] Несмотря на это, будьте осторожны: для декодирования MPEG-4 AVC видео с DVD разрешением необходима быстрая машина (например, Pentium 4 свыше 1.5 ГГц или Pentium M свыше 1 ГГц).

[2] Один и тот же результат кодирования может не выглядеть таким же на чьем-либо другом мониторе или при воспроизведении с помощью другого декодера, так что проверяйте Ваши результаты кодирования на жизнеспособность, воспроизводя их в разных начальных условиях.

Глава 8. Часто Задаваемые вопросы

8.1. Разработка

Вопрос:	Как создать правильный патч для MPlayer?
Ответ:	Мы создали краткий документ (англ.), описывающий все необходимые детали. Пожалуйста, следуйте инструкциям.
Вопрос:	Как перевести MPlayer на новый язык?
Ответ:	Прочитайте HOWTO по переводу (англ.), там все описано. За дополнительной помощью Вы можете обратиться в рассылку MPlayer-translations .
Вопрос:	Как можно помочь развитию MPlayer?
Ответ:	Мы будем более чем счастливы принять пожертвования оборудованием и программным обеспечением. Они позволяют нам постоянно улучшать MPlayer.
Вопрос:	Как стать разработчиком MPlayer?
Ответ:	Мы всегда рады приветствовать людей, занимающихся написанием кода и документации. Для начала прочтите техническую документацию (англ.). Затем Вам следует подписаться на список рассылки MPlayer-dev-eng и начать написание кода. Если Вы хотите помочь с документацией, то подпишитесь на MPlayer-docs .
Вопрос:	Почему Вы не используете autoconf/automake?
Ответ:	У нас есть модульная, написанная вручную система сборки. Она хорошо справляется со своей работой, зачем же ее менять? Кроме того, нам, как и некоторым другим , не нравятся auto* утилиты.

8.2. Компиляция и установка

Вопрос:	Компиляция завершается с ошибкой и gcc вываливается с загадочным сообщением, содержащим фразу <code>internal compiler error</code> или <code>unable to</code>
---------	---

	find a register to spill или can't find a register in class 'GENERAL_REGS' while reloading 'asm'.
Ответ:	Вы столкнулись с ошибкой в gcc. Пожалуйста, сообщите об этом разработчикам gcc но не нам. В силу некоторых причин MPlayer, похоже, часто вызывает ошибки компилятора. Тем не менее, мы не можем их исправить, но и добавлять специальный код для обхода ошибок компилятора не будем. Чтобы избежать этой проблемы, либо используйте надежную и стабильную версию компилятора, либо часто обновляйтесь.
Вопрос:	Существуют ли бинарные (RPM/Debian) пакеты MPlayer?
Ответ:	Смотрите разделы Debian и RPM .
Вопрос:	Как собрать 32-х битную версию MPlayer на 64-х битном Athlon?
Ответ:	Попробуйте следующие опции configure: <pre>./configure --target=i386-linux --cc="gcc -m32" --as="as --32" --with-extralibdir=/usr/lib</pre>
Вопрос:	Configure завершается с указанным сообщением, и MPlayer не компилируется! <pre>Your gcc does not support even i386 for '-march' and '-mcpu'</pre>
Ответ:	Ваш gcc некорректно установлен, проверьте config.log для более подробной информации.
Вопрос:	Я имею Matrox G200/G400/G450/G550, как мне скомпилировать/использовать mga_vid драйвер?
Ответ:	Прочитайте раздел mga_vid .
Вопрос:	На этапе 'make', MPlayer жалуется на отсутствующие библиотеки X11. Я не понимаю, ведь X11 у меня <i>установлен!</i> ?
Ответ:	... но Вы не имеете установленных пакетов разработки X11. Либо они установлены некорректно. Они называются XFree86-devel* в Red Hat, xlibs-dev в Debian Woody и libx11-dev в Debian Sarge. Также проверьте существуют ли символические ссылки /usr/X11 и /usr/include/X11 .
Вопрос:	Сборка в Mac OS 10.3 приводит к нескольким ошибкам линковки.
Ответ:	Как правило, ошибка линковки выглядит так: <pre>ld: Undefined symbols: _LLCStyleInfoCheckForOpenTypeTables referenced from QuartzCore expected to be defined in ApplicationServices _LLCStyleInfoGetUserRunFeatures referenced from QuartzCore expected to be defined in ApplicationServices</pre> <p>Эта проблема — результат разработчиков Apple, использующих 10.4 для компиляции их программ и распространяющих бинарники пользователям 10.3 через Software Update. Неопределённые символы присутствуют в 10.4, но их нет в 10.3. Одно из решений — откат QuickTime до версии 7.0.1, но есть и лучшее.</p> <p>Загрузите более старую версию фреймворка. Вы получите архив, содержащий фреймворк QuickTime 7.0.1 и QuartzCore 10.3.9. Распакуйте файл куда-нибудь, кроме папки System (т.е. не устанавливайте эти программы в /System/Library/Frameworks! Эта старая копия нужна только для решения проблем с линковкой!)</p> <pre>gunzip < CompatFrameworks.tgz tar xvf -</pre> <p>В config.mak следует добавить -F/path/to/where/you/extracted к переменной OPTFLAGS. При использовании X-Code Вы можете просто выбрать эти фреймворки вместо системных. Получившийся бинарный MPlayer будет использовать фреймворк, установленный в Вашей системе, используя динамические ссылки времени исполнения. (Это можно проверить, запустив <code>otool -l</code>).</p>

8.3. Общие вопросы

Вопрос:	Есть ли списки рассылки, посвященные MPlayer?
Ответ:	Да. Смотрите раздел списки рассылки нашего сайта.
Вопрос:	Я обнаружил неприятную ошибку, появляющуюся при просмотре моего любимого фильма! Кому мне следует сообщить об этом?
Ответ:	Прочтите руководство по составлению отчетов об ошибках и следуйте инструкциям.
Вопрос:	У меня происходит дамп ядра при попытке создать дамп потоков, что не так?
Ответ:	Не паникуйте. Убедитесь, что знаете где Ваше полотенце. Серьёзно, обратите внимание на смайлик и ищите файлы, оканчивающиеся на .dump .
Вопрос:	В начале воспроизведения появляется следующее сообщение, хотя все вроде бы работает прекрасно:
	<pre>Ошибка инициализации Linux RTC в ioctl (rtc_pie_on): Permission denied</pre>
Ответ:	Чтобы использовать RTC тайминг, Вам необходимо специально настроенное ядро. Подробности ищите в разделе RTC .
Вопрос:	Как сделать скриншот??
Ответ:	Чтобы иметь возможность сделать скриншот, Вы должны использовать драйвер вывода видео, который не использует оверлей. В X11 это -vo x11 , в Windows используйте -vo directx:noaccel . Есть и другой способ. Запустите MPlayer с видео фильтром screenshot (-vf screenshot) , и нажмите клавишу s для создания скриншота.
Вопрос:	Каков смысл чисел в статусной строке?
Ответ:	Пример: <pre>A: 2.1 V: 2.2 A-V: -0.167 ct: 0.042 57/57 41% 0% 2.6% 0 4 49% 1.00x</pre> A: 2.1 позиция аудио в секундах V: 2.2 позиция видео в секундах A-V: -0.167 сдвиг аудио-видео в секундах (задержка) ct: 0.042 итоговая выполненная коррекция A-V синхронизации 57/57 кадров проиграно/декодировано (считая от места последней перемотки) 41% время CPU в процентах, используемое видео кодеком (для поблочного и прямого рендеринга сюда включается и video_out) 0% время CPU, используемое video_out 2.6% время CPU в процентах, используемое аудио кодеком 0 выброшено кадров для сохранения синхронизации A-V 4 текущий уровень постобработки (при использовании -autoq) 49% текущий уровень использования кеша. (нормальное значение — около 50%) 1.00x скорость воспроизведения (множитель от нормальной скорости) Большинство из них используются для отладки, используйте опцию

	-quiet чтобы их скрыть. Имейте ввиду, что для некоторых файлов использование CPU модулем video_out равно нулю (0%). Дело в том, что иногда он вызывается непосредственно из кодека и не может быть измерен отдельно. Если Вы хотите узнать скорость video_out в этом случае, сравните разницу при воспроизведении с -vo null и Вашим драйвером вывода видео.
Вопрос:	Появляется сообщение, что не найден файл <code>/usr/local/lib/codecs/ ...</code>
Ответ:	Загрузите и установите бинарные кодеки с нашей страницы загрузки .
Вопрос:	Как сделать, чтобы MPlayer запомнил мои особые опции для определенного файла, например <code>movie.avi</code> ?
Ответ:	Создайте файл с именем <code>movie.avi.conf</code> и Вашими опциями в нем, сохраните его в каталог <code>~/.mplayer</code> или в один каталог с медиа-файлом.
Вопрос:	Субтитры великолепные, лучшие из виденных когда-либо, но они замедляют воспроизведение! Я знаю, это необычно...
Ответ:	После запуска <code>./configure</code> , отредактируйте <code>config.h</code> и замените <code>#undef FAST_OSD</code> на <code>#define FAST_OSD</code> . Перекомпилируйте.
Вопрос:	Как запустить MPlayer в фоновом режиме?
Ответ:	Используйте: <pre>mplayer опции имя_файла < /dev/null &</pre>

8.4. Проблемы воспроизведения

Вопрос:	Не могу локализовать причину некоторой странной проблемы произведения.
Ответ:	Завалился файл <code>codecs.conf</code> в <code>~/.mplayer/</code> , <code>/etc/</code> , <code>/usr/local/etc/</code> или в аналогичном месте? Удалите его, Устаревший <code>codecs.conf</code> может стать причиной непонятных проблем и предназначается только для использования разработчиками, занимающимися поддержкой кодеков. Он переопределяет внутренние настройки кодеков MPlayer'a, что не сулит ничего хорошего, если в новых версиях программы внесены несовместимые изменения. При использовании неэкспертами, это верный путь к, на первый взгляд, случайным и неподдающимся локализации крахам и проблемам воспроизведения. Если этот файл присутствует где-либо в Вашей системе, удалите его.
Вопрос:	Как настроить субтитры для вывода на черных полях вокруг изображения?
Ответ:	Воспользуйтесь видео фильтром <code>expand</code> для увеличения области рендеринга по вертикали и разместите изображение у верхней кромки: <pre>mplayer -vf expand=0:-100:0:0 -slang de dvd://1</pre>
Вопрос:	Как выбрать дорожки звука/субтитров в DVD, OGM, Matroska или NUT файле?
Ответ:	Нужно использовать опцию -aid (аудио ID) или -alang (язык аудио), -sid (ID субтитров) или -slang (язык субтитров), например: <pre>mplayer -alang eng -slang eng example.mkv mplayer -aid 1 -sid 1 example.mkv</pre> Посмотреть список доступных: <pre>mplayer -vo null -ao null -frames 0 -v имя_файла grep sid mplayer -vo null -ao null -frames 0 -v имя_файла grep aid</pre>
Вопрос:	Пытаюсь воспроизвести поток через интернет, но ничего не получается.
Ответ:	Попробуйте воспроизвести поток с опцией -playlist .
Вопрос:	Я загрузил фильм из P2P сети, но он не работает!

Ответ:	У Вас испорченный или поддельный файл. Если Вы получили его от друзей, которые утверждают, что он работает, проверьте контрольную сумму md5sum.
Вопрос:	У меня проблемы с отображением субтитров, помогите!
Ответ:	Убедитесь, что шрифты установлены правильно. Снова пройдите все шаги части Шрифты и OSD раздела, посвященного установке. Если используете TrueType шрифты, проверьте, что установлена библиотека FreeType. Стоит также проверить Ваши субтитры при помощи текстового редактора или стороннего проигрывателя. Попробуйте преобразовать их в другой формат.
Вопрос:	Почему MPlayer не работает в Fedora Core?
Ответ:	<p>Из-за плохого взаимодействия между exec-shield, prelink и любым приложением, использующим Windows DLLs (таким как MPlayer). Проблема в том, что exec-shield делает случайными адреса загрузки всех системных библиотек. Смена адресов производится на стадии prelink (один раз каждые две недели).</p> <p>Когда MPlayer пытается загрузить Windows DLL, он хочет поместить ее по определенному адресу (0x400000). Если там уже находится важная системная библиотека, MPlayer рухнет. (Типичные симптомы — segmentation fault при попытке воспроизвести файлы Windows Media 9.)</p> <p>Если столкнулись с этой проблемой, то у Вас есть два варианта:</p> <ul style="list-style-type: none"> • Подождать пару недель. Все может снова заработать. • Слинковать все бинарники в системе с различными prelink опциями. Вот пошаговая инструкция: <ul style="list-style-type: none"> • Отредактируйте <code>/etc/syconfig/prelink</code> и измените <code>PRELINK_OPTS=-mR</code> на <code>PRELINK_OPTS="-mR --no-exec-shield"</code> • touch /var/lib/misc/prelink.force • <code>/etc/cron.daily/prelink</code> (Это перелинкует все приложения, что может занять длительное время.) • execstack -s /path/to/mplayer (Это отключит exec-shield для исполняемого файла MPlayer.)
Вопрос:	<p>MPlayer умирает с сообщением:</p> <pre>MPlayer прерван сигналом 4 в модуле: decode_video</pre>
Ответ:	Не используйте MPlayer на процессоре, отличном от того, на котором он компилировался, либо перекомпилируйте с определением CPU "на лету" (<code>./configure --enable-runtime-cpudetection</code>).
Вопрос:	При попытке захвата с тюнера, все работает, но цвета — неправильные. С другими приложениями все в порядке.
Ответ:	Ваша карта, вероятно, сообщает о некотором пространстве цветов как поддерживаемом, хотя на самом деле его не поддерживает. Попробуйте с YUY2 вместо YV12 по умолчанию (смотрите секцию TV).
Вопрос:	Получаю странные процентные значения (слишком большие) при проигрывании файлов на ноутбуке.
Ответ:	Это эффект системы управления энергопотреблением ноутбука (BIOS, не ядра). Подсоедините разъем внешнего питания до включения ноутбука. Также можно попробовать cpufreq (SpeedStep интерфейс для Linux), возможно это поможет.
Вопрос:	Аудио/видео теряют синхронизацию, при запуске MPlayer с правами root на ноутбуке. При запуске с правами обычного пользователя все работает нормально.
Ответ:	Это тоже эффект системы управления энергопотреблением (смотрите выше). Подсоедините разъем внешнего питания до включения ноутбука либо убедитесь, что не используется опция <code>-rtc</code> .

Вопрос:	При воспроизведении фильма изображение вдруг начинает дергаться и появляется следующее сообщение: Обнаружен плохо 'слоёный' AVI файл - переключаюсь в -ni режим...
Ответ:	Файлы с некорректным чередованием и опция <code>-cache</code> вместе работают не очень хорошо. Попробуйте <code>-nocache</code> .

8.5. Проблемы драйверов вывода видео/аудио (vo/ao)

Вопрос:	При запуске в полноэкранном режиме, наблюдаются черные полосы вокруг изображения вместо полноэкранного масштабирования.
Ответ:	Ваш драйвер вывода видео аппаратно не поддерживает масштабирование, и, поскольку программное масштабирование может быть ужасно медленным, MPlayer не задействует его автоматически. Скорее всего Вы используете драйвер вывода x11 вместо xv. Попробуйте указать <code>-vo xv</code> в командной строке или прочтите раздел видео , чтобы узнать о других драйверах вывода видео. Опция <code>-zoom</code> явно задействует программное масштабирование.
Вопрос:	Я только что установил MPlayer. Попытка открыть файл приводит к фатальной ошибке: Ошибка при открытии/инициализации выбранного устройства видеовывода (-vo). Как решить проблему?
Ответ:	Просто смените драйвер вывода видео. Введите следующую команду для получения списка доступных драйверов: <code>mplayer -vo help</code> Как только выберите правильный драйвер, добавьте его в файл конфигурации. Добавьте <code>vo = выбранный_драйвер</code> в <code>~/.mplayer/config</code> и/или <code>vo_driver = выбранный_драйвер</code> в <code>~/.mplayer/gui.conf</code> .
Вопрос:	У меня проблемы с [Ваш оконный менеджер] и полноэкранном режимом xv/xmga/sdl/x11 ...
Ответ:	Прочтите руководство по составлению отчетов об ошибках и отошлите соответствующий отчет нам. Попробуйте также поэкспериментировать в опцией <code>-fstype</code> .
Вопрос:	Теряется синхронизация звука при воспроизведении AVI файла.
Ответ:	Попробуйте опцию <code>-bps</code> или <code>-nobps</code> . Если это не помогает прочтите руководство по составлению отчетов об ошибках и загрузите этот файл на наш FTP.
Вопрос:	Как использовать dmix с MPlayer?
Ответ:	После настройки asoundrc используйте <code>-ao alsa:device=dmix</code> .
Вопрос:	Нет звука при воспроизведении видео и появляется ошибка подобная этой: A0: [oss] 44100Hz 2ch Signed 16-bit (Little-Endian) [A0 OSS] инициализация аудио: Не могу открыть аудиоустройство /dev/dsp: Device or resource busy Не могу открыть/инициализировать аудиоустройство -> без звука. Аудио: без звука Начало воспроизведения...
Ответ:	Используете KDE или GNOME со звуковыми службами aRts или ESD? Попробуйте отключить службу звука или используйте <code>-ao arts</code> или <code>-ao esd</code> опции, для указания MPlayer использовать aRts или ESD. Также

	возможно, что Вы используете ALSA без эмуляции OSS, попробуйте загрузить ALSA OSS модули ядра или добавьте опцию <code>-ao alsa</code> в командную строку, для использования ALSA напрямую.
Вопрос:	При запуске MPlayer в KDE наблюдается черный экран и ничего более. Примерно через минуту начинается воспроизведение.
Ответ:	aRts блокирует звуковое устройство. Либо подождите старта видео, либо отключите службу aRts в центре управления. Если Вы хотите использовать звук через aRts, укажите вывод звука через встроенный звуковой драйвер aRts (<code>-ao arts</code>). Если он не работает, либо MPlayer собран без него, попробуйте SDL (<code>-ao sdl</code>) и убедитесь, что SDL может работать со звуком aRts. Ещё один вариант — запуск MPlayer с <code>artsdsp</code> .
Вопрос:	Проблема с синхронизацией A/V. Некоторые из моих AVI файлов проигрываются нормально, но некоторые — с двойной скоростью!
Ответ:	У Вас неверно работающая звуковая карта/драйвер. Скорее всего, поддерживается работа только с 44100Гц, а Вы пытаетесь проиграть файл с 22050Гц звуком. Попробуйте звуковой фильтр <code>resample</code> .
Вопрос:	При проигрывании фильма происходит рассинхронизация звука и видео или MPlayer рушится с ошибкой: Слишком много (945 в 8390980 байтах) видеопакетов в буфере!
Ответ:	Причин может быть несколько. <ul style="list-style-type: none"> • Ваш CPU и/или видеокарта и/или шина слишком медленные. В этом случае MPlayer выводит соответствующее сообщение (и количество выброшенных кадров быстро увеличивается). • Если это AVI, возможно, у него проблемы с чередованием звука/видео, попробуйте опцию <code>-ni</code> для обхода проблемы. Или файл имеет неверный заголовок. В этом случае могут помочь опции <code>-nobps</code> и/или <code>-mc 0</code>. • Многие FLV файлы будут правильно воспроизводиться только с <code>-correct-pts</code>. К сожалению, MEncoder такой опции не имеет, но Вы можете попробовать вручную указать правильное значение <code>-fps</code>, если оно Вам известно. • У Вас неверно работает звуковой драйвер.
Вопрос:	Как избавиться от рассинхронизации при перематке потоков RealMedia?
Ответ:	Может помочь опция <code>-mc 0.1</code> .

8.6. Воспроизведение DVD

Вопрос:	Как насчет DVD навигации/Меню?
Ответ:	MPlayer не поддерживает DVD меню в силу серьезных архитектурных ограничений, которые мешают правильной обработке неподвижных изображений и интерактивного содержимого. Если хотите модные меню, Вам придется использовать другой проигрыватель, например <code>xine</code> , <code>vlc</code> или <code>Ogle</code> . Если Вы хотите получить DVD навигацию в MPlayer, реализуйте ее самостоятельно, но имейте ввиду, что это нелегкое и ответственное дело.
Вопрос:	Как насчет субтитров? MPlayer умеет их отображать?
Ответ:	Да. Смотрите DVD раздел .
Вопрос:	Как установить регион для DVD-привода? У меня нет Windows!
Ответ:	Воспользуйтесь утилитой regionset .
Вопрос:	DVD не проигрываются, MPlayer зависает или выводит ошибку "Encrypted VOB file!".
Ответ:	Код расшифровки CSS не работает с некоторыми приводами, если регион

	неверно установлен. Смотрите ответ на предыдущий вопрос.
Вопрос:	Нужны ли права root (setuid) для воспроизведения DVD?
Ответ:	Нет. Конечно, Вы должны иметь необходимые права на файл DVD устройства (в <code>/dev/</code>).
Вопрос:	Возможно ли проигрывать/кодировать только выбранные разделы?
Ответ:	Да, попробуйте опцию <code>-chapter</code> .
Вопрос:	Воспроизведение DVD медленное!
Ответ:	Используйте опцию <code>-cache</code> (описанную на странице <code>man</code>) и попробуйте включить DMA для DVD привода утилитой <code>hdparm</code> (описанной в разделе CD).
Вопрос:	Я скопировал DVD используя <code>vobcopy</code> . Как его проигрывать/кодировать с жесткого диска?
Ответ:	Используйте опцию <code>-dvd-device</code> , чтобы указать каталог, содержащий файлы: <pre>mpplayer dvd://1 -dvd-device /путь/к/каталогу</pre>

8.7. Просьбы о новых возможностях

Вопрос:	Если MPlayer приостановлен, и я пытаюсь перемотать фильм или нажимаю любую клавишу, воспроизведение продолжается. Хотелось бы иметь возможность перемотки во время паузы.
Ответ:	Это нелегко реализовать без потери синхронизации A/V. Пока ни одна попытка не увенчалась успехом, но патчи приветствуются.
Вопрос:	Хочется перематывать на +/- 1 кадр вместо 10 секунд.
Ответ:	Вы можете перемещаться на 1 кадр вперед, нажимая клавишу <code>..</code> После этого воспроизведение будет приостановлено (смотрите страницу руководства <code>man</code> для подробностей). Перемещение назад вряд ли будет реализовано в ближайшем будущем.

8.8. Кодирование

Вопрос:	Как производить кодирование?
Ответ:	Прочтите раздел MEncoder .
Вопрос:	Как сохранить весь ролик DVD в файл?
Ответ:	Как только определитесь с выбором ролика и проверите, что он проигрывается в MPlayer, воспользуйтесь опцией <code>-dumpstream</code> . Например: <pre>mpplayer dvd://5 -dumpstream -dumpfile dvd_dump.vob</pre> сохранит 5-й ролик в файл <code>dvd_dump.vob</code>
Вопрос:	Как автоматически создавать (S)VCD?
Ответ:	Попробуйте скрипт <code>mencvcd.sh</code> из каталога TOOLS . С его помощью Вы можете перекодировать DVD или другие фильмы в VCD или SVCD формат и даже записывать их на CD.
Вопрос:	Как создать (S)VCD?
Ответ:	Новые версии MEncoder могут самостоятельно создавать MPEG-2 файлы, которые можно использовать в качестве основы для VCD или SVCD, производящиеся "из коробки" на всех платформах (например, чтобы поделиться фильмом, снятым на видеокамеру, с компьютерно-неграмотными друзьями). Прочтите Использование MEncoder для создания VCD/SVCD/DVD-совместимых файлов .

Вопрос:	Как склеить два файла?
Ответ:	<p>MPEG файлы могут быть объединены в один, если сильно повезет. Для файлов AVI, можете воспользоваться встроенной возможностью MEncoder'a работать с несколькими файлами сразу:</p> <pre>mencoder -ovc copy -oac copy -o out.avi файл1.avi файл2.avi</pre> <p>Это будет работать только если файлы имеют одинаковое разрешение и сжаты одинаковым кодеком. Также попробуйте avidemux и avimerge (часть набора утилит transcode).</p>
Вопрос:	Как исправить AVI файлы с испорченным индексом или неверным чередованием?
Ответ:	<p>Избежать использования <code>-idx</code> и иметь возможность перемотки в AVI файлах с испорченным индексом, либо избежать <code>-ni</code> для воспроизведения файлов с неверным чередованием поможет команда</p> <pre>mencoder input.avi -idx -ovc copy -oac copy -o output.avi</pre> <p>Она скопирует видео и аудио в новый файл, генерируя новый индекс и исправляя чередование. Конечно, это не поможет исправить ошибки в видео и/или аудио потоке.</p>
Вопрос:	Как исправить пропорции в AVI файле?
Ответ:	<p>Вы можете это сделать при помощи опции MEncoder'a <code>-force-avi-aspect</code>, которая переопределяет информацию о пропорциях, хранящуюся в опции <code>vprp</code> AVI OpenDML заголовка. Например:</p> <pre>mencoder input.avi -ovc copy -oac copy -o output.avi -force-avi-aspect 4/3</pre>
Вопрос:	Как сделать копию и закодировать VOB файл с испорченным началом?
Ответ:	<p>Основная проблема при попытке кодировать испорченный VOB файл [3] состоит в том, что будет трудно закодировать файл с хорошей A/V синхронизацией. Один из способов заключается в отбрасывании испорченных данных и кодирования только корректной части. Сначала ищите начало корректных данных (меняя параметр "колво_байтов_для_пропуска"):</p> <pre>mplayer input.vob -sb колво_байтов_для_пропуска</pre> <p>Затем создаете новый файл, содержащий корректную часть:</p> <pre>dd if=input.vob of=output_cut.vob skip=1 ibs=колво_байтов_для_пропуска</pre>
Вопрос:	Не могу закодировать DVD субтитры в AVI!
Ответ:	Необходимо правильно указать опцию <code>-sid</code>
Вопрос:	Как кодировать только выбранные эпизоды из DVD?
Ответ:	Используйте опцию <code>-chapter</code> , например <code>-chapter 5-7</code> .
Вопрос:	Я хочу работать с файлами более 2Гб в файловой системе VFAT. Это возможно?
Ответ:	Нет. VFAT не поддерживает файлы более 2Гб.
Вопрос:	Что означают числа в статусной строке в процессе кодирования?
Ответ:	<p>Пример:</p> <pre>Pos: 264.5s 6612f (2%) 7.12fps Trem: 576min 2856mb A-V:0.065 [2156:192]</pre> <p>Pos: 264.5s временная позиция в кодируемом потоке 6612f количество закодированных кадров 2% размер закодированной части входного потока 7.12fps скорость кодирования</p>

	<p>Trem: 576min оценка времени, оставшегося до конца кодирования 2856mb оценка окончательного размера перекодированного файла A-V:0.065 текущая задержка между аудио и видео потоками [2156:192] средний видео битпоток (в кбит/с) и средний аудио битпоток (в кбит/с)</p>
Вопрос:	Почему рекомендуемый MEncoder'ом битрейт отрицателен?
Ответ:	Потому что аудио битрейт в процессе кодирования становится слишком велик, чтобы уместить фильм на CD. Проверьте, правильно ли установлена libmp3lame.
Вопрос:	Не могу перекодировать ASF файл в AVI/MPEG-4, потому что используется 1000fps.
Ответ:	Поскольку ASF использует переменную частоту кадров, а AVI — фиксированную, необходимо её установить вручную опцией -ofps .
Вопрос:	Как поместить субтитры в выходной файл?
Ответ:	Просто укажите MEncoder опцию -sub <имя_файла> (или -sid , соответственно).
Вопрос:	Как закодировать только звук из видеоклипа?
Ответ:	<p>Напрямую это невозможно, но можете попробовать такой способ (обратите внимание на & в конце команды mplayer):</p> <pre>mkfifo encode mplayer -ao pcm -aofile encode dvd:///1 & lame Ваши_опции encode music.mp3 rm encode</pre> <p>Это позволяет использовать любой кодер, не только LAME, просто замените lame на Ваш предпочитаемый кодер.</p>
Вопрос:	Почему сторонние проигрыватели не могут воспроизвести MPEG-4 файлы, сжатые MEncoder'ом версий старше 1.0pre7?
Ответ:	<p>libavcodec, родная библиотека кодирования MPEG-4, обычно поставляемая с MEncoder, устанавливала FourCC в 'DIVX' при сжатии MPEG-4 видео (FourCC — это тэг AVI, указывающий нужную программу сжатия и декодирования видео, т.е. кодек). Поэтому многие люди считают, что libavcodec была библиотекой кодирования DivX, когда на самом деле это совершенно другая библиотека кодирования MPEG-4, реализующая стандарт MPEG-4 значительно лучше, чем DivX. Поэтому новый FourCC, по умолчанию используемый libavcodec — это 'FMP4', но Вы можете переопределить это поведение, используя опцию MEncoder'a -ffourcc. Вы также можете сменить FourCC у существующего файла тем же способом:</p> <pre>mencoder input.avi -o output.avi -ffourcc XVID</pre> <p>Имейте ввиду, что этот пример устанавливает FourCC в XVID, а не DIVX. Это рекомендуется, т.к. DIVX FourCC означает DivX4, очень простой MPEG-4 кодек, в то время как DX50 и XVID оба — полные MPEG-4 (ASP). Таким образом, если Вы установите FourCC в DIVX, некоторые плохие программные или аппаратные проигрыватели могут быть в шоке от дополнительных возможностей, которые libavcodec поддерживает, а DivX — нет; с другой стороны Xvid по функциональности ближе к libavcodec и поддерживается любыми приличными проигрывателями.</p>
Вопрос:	Как перекодировать звуковой файл (без видео)?
Ответ:	Используйте aconvert.sh из подкаталога TOOLS дерева исходников MPlayer.
Вопрос:	Как воспроизвести субтитры внедренные в AVI?

Ответ:	Используйте <code>avisubdump.c</code> из подкаталога TOOLS или прочтите этот документ о извлечении/мультиплексировании субтитров внедренных в OpenDML AVI файлы .
Вопрос:	MPlayer не хочет...
Ответ:	Посмотрите коллекцию всевозможных скриптов и хаков в подкаталоге TOOLS . TOOLS/README содержит документацию.

[3] В какой-то мере некоторые формы защиты от копирования могут быть восприняты как испорченное содержимое.