

Инструкция по работе с “SSS Rescue CD” и программой CF_repair

(Для CF_repair версии 1.18 и более новых)

1. Назначение “SSS Rescue CD”

Компакт диск (CD) **SSS Rescue CD** предназначен для использования в качестве инструментального средства при создании системы на Compact Flash (CF), либо при восстановлении информации на повреждённых CF, которые устанавливаются в процессорных системах (модулях форматов PC/104 и ETX), поставляемых ООО «СПЕЦСТРОЙ-СВЯЗЬ» в составе своего оборудования. Этот CD является загрузочным и содержит операционную систему **Linux** со всеми программами и архивами, необходимыми для проведения восстановления информации на CF. Система на **SSS Rescue CD** способна загрузиться на подавляющем большинстве конфигураций персональных компьютеров (PC) с обычным BIOS. (U)EFI BIOS пока не поддерживается. Для сокращения объёма **SSS Rescue CD**, в нём исключена возможность работы в графическом режиме (нет X Window System), но поддерживается работа консоли в псевдографическом режиме через frame buffer. Такой режим работы консоли включается по умолчанию.

Для удобства работы пользователей в данной системе произведена русификация экрана и клавиатуры. Русификация экрана выполняется знакогенератором (font) **koi8-r**, а русификация клавиатуры выполняется с помощью **keymap**-файла (преобразователя) **ru-ms**. Переключение режимов **RUS/LAT** в этом преобразователе выполняется правой клавишей <Ctrl>.

Данная инструкция предполагает, что пользователь имеет начальные познания об организации данных на внешних носителях, в частности, на жестких дисках. Некоторые пояснения приведены в пункте 4.

Далее последовательно описываются все действия, которые должен выполнить пользователь, для восстановления ОС **Linux** и прикладной системы на CF. Некоторые из этих этапов можно пропустить, о чём оговаривается дополнительно. Причём, программа **CF_repair** восстанавливает только операционную систему и начальное состояние прикладной системы. Все настройки прикладных систем (конфигурации, маршрутизации и т. п.) пользователю придётся впоследствии выполнить вручную или восстановить с резервных копий, если таковые имеются. Следует заметить, что эту версию **Compact Flash Repair Tool** рекомендуется использовать только в тех случаях, когда файловую систему на CF невозможно восстановить (отремонтировать) другими способами. Но следует также помнить о том, что повреждение CF может быть более существенным, например, физический выход CF из строя, тогда и программа **CF_repair** не поможет её восстановить. В этих случаях CF следует заменить на работоспособную.

Дополнительно следует заметить, что программа **CF_repair** будет формировать гарантированно работоспособные CF только при работе в среде **SSS Rescue CD**. Использование этой программы в других системах не гарантирует её правильное функционирование и формирование полностью работоспособных систем на CF.

2. Загрузка ОС Linux с “SSS Rescue CD”

Основное требование для выполнения этого этапа: BIOS данного компьютера должен поддерживать загрузку с CD-ROM.

Для работы с **SSS Rescue CD** необходимо установить загрузку с CD-ROM в Setup системного BIOS компьютера и выполнить процедуру загрузки или перезапуска системы (естественно, что этот CD должен при этом находиться в приводе устройства чтения компакт дисков). Подключение Card Reader'a к разъёму USB и установку CF в Card Reader

рекомендуется выполнить до запуска программы **CF_repair**. В противном случае она может не обнаружить ни Card Reader, ни CF.

Внимание! В данной версии **SSS Rescue CD** после вывода экрана меню необходимо в течение 15 секунд нажать любую клавишу, в противном случае будет выполнена загрузка с первого жесткого диска компьютера.

Внимание! Необходимо заметить, что программа **CF_repair** поддерживает только Card Reader'ы, подключаемые к системе посредством шины USB. Устройства, подключаемые через интерфейс PCMCIA (PC Card), не поддерживаются.

Загрузчик **isolinux** после старта выводит на экран меню, показанное на рисунке 1. Все пункты меню отображаются символами серого цвета, а выбранный пункт (вариант загрузки) символами черного цвета. Выбор требуемого варианта загрузки выполняется с помощью клавиш ↑ (стрелка вверх) и ↓ (стрелка вниз), либо с помощью «горячих клавиш», символы которых выделены зеленым и розовым цветом. Последующее нажатие клавиши <Enter> вызовет загрузку соответствующего варианта.

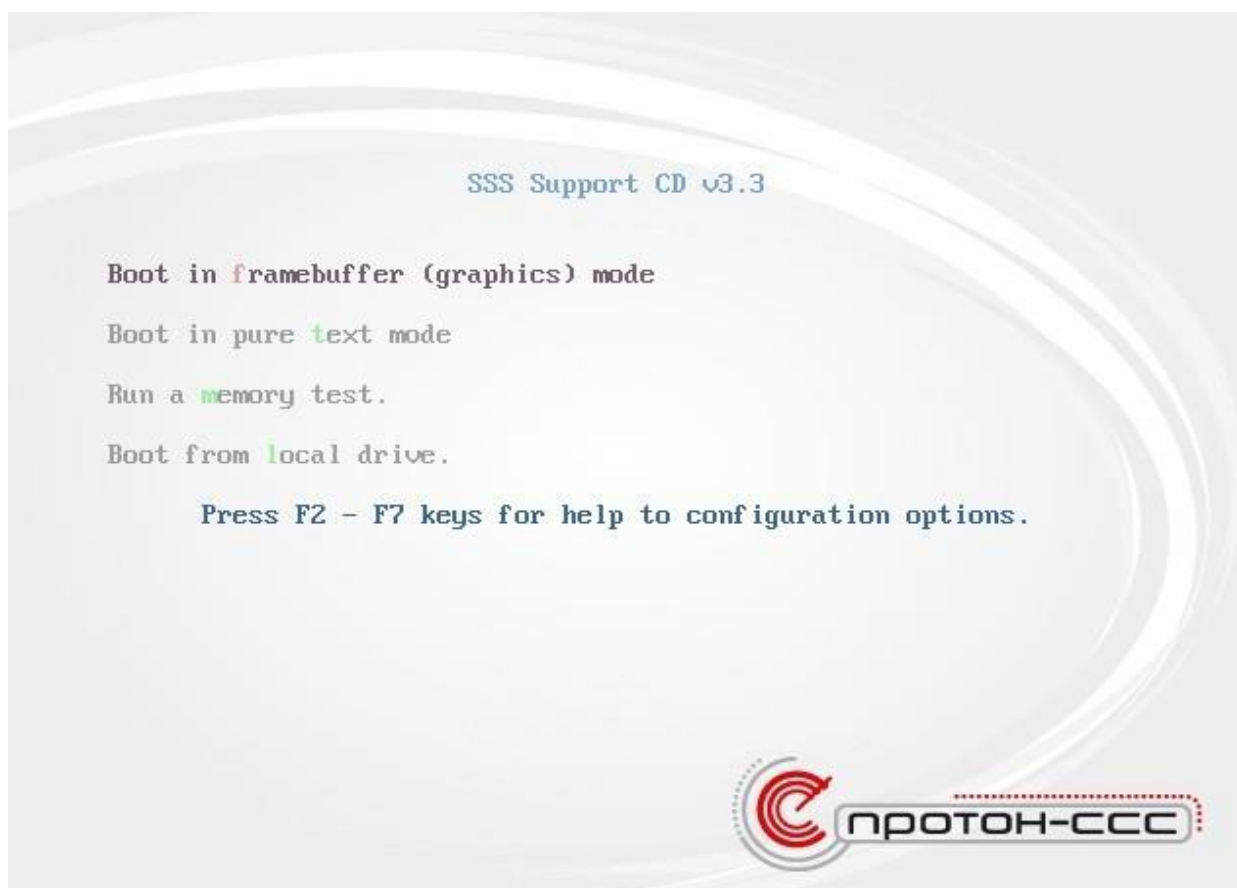


Рис. 1. Экран с меню загрузчика **isolinux**.

Выбор пункта **Boot in framebuffer (graphics) mode** вызовет продолжение загрузки системы и инициализацию консоли в графическом режиме (размер экрана при этом, как правило, устанавливается в 48 строк по 128 символов, если видеоадаптер и монитор позволяют установить видеорежим 1024x768x16). Данный пункт является вариантом загрузки по умолчанию.

Выбор пункта **Boot in pure text mode** вызовет продолжение загрузки системы и инициализацию консоли в текстовом режиме (размер экрана 25 строк по 80 символов). Этот вариант следует использовать, если при загрузке в предыдущем варианте по каким-либо причинам ядро не смогло успешно проинициализировать графический режим для консоли, что может быть, если BIOS видеоадаптера не совместим со спецификацией VESA версии 2.0 или выше.

Выбор пункта **Run a memory test** вызовет загрузку и выполнение теста оперативной памяти компьютера.

Выбор пункта **Boot from local drive** вызовет загрузку системы с первого жесткого диска вашего компьютера. Полезно в том случае, если возникают проблемы с изъятием диска после перезапуска системы. Тот же эффект получится, если на клавиатуре ничего не нажимать в течение 15 секунд – это вариант загрузки по умолчанию для данной версии **SSS Rescue CD**.

Нажатие клавиши **<Tab>** на выбранном пункте вызывает отображение в нижней части экрана команды, которая будет выполнена загрузчиком для этого пункта меню. Эту команду можно редактировать (добавлять параметры). Убирать параметры не рекомендуется, потому что в таком случае система может не загрузиться. Это имеет смысл только для первых двух пунктов меню.

Нажатие клавиш **<F2>** – **<F7>** вызывает отображение дополнительных подсказок по возможным параметрам ядра и режимам инициализации системы. Их можно добавить к команде загрузки, отображенной после нажатия клавиши **<Tab>**, для изменения поведения системы как при инициализации, так и во время функционирования. Полный перечень этих параметров и режимов инициализации также приведен в файле **README.txt**, который находится в корневом каталоге CD. Указанные параметры и режимы инициализации используются в основном для устранения конфликтов системы с экзотическим внешним оборудованием, и их следует использовать только тогда, когда процесс загрузки и инициализации системы не может успешно завершиться. Очень полезными в таком случае могут оказаться режимы **nohotplug** или **nocoldplug**, отключающие определение дополнительных устройств.

После успешной загрузки и инициализации различных служб и сервисов система переходит в режим ввода команд с консоли (клавиатуры), и на экране должно появиться приглашение:

```
root#
```

Если этого не происходит то, скорее всего для ядра при его загрузке требуется ввод дополнительных параметров (см. выше), либо в компьютере установлено оборудование, которое не поддерживается ядром **Linux**, имеющимся на **SSS Rescue CD**. Также возможен вариант механического повреждения CD, либо CD плохо воспринимается приводом (особенно, если данный CD был «нарезан» в другом приводе).

3. Некоторые замечания о Card Reader'ах и шине USB

В ОС **Linux** все устройства Flash-памяти, подключаемые к компьютеру посредством шины USB, обслуживаются в системе драйвером **usb-storage**. Драйвер **usb-storage** представляет в системе любые USB-устройства массовой памяти, в том числе и Card Reader'ы, как SCSI-диски, причем, при назначении устройств выбирает первое свободное на данный момент устройство этого типа. Для пользователей эти устройства доступны через специальные файлы устройств в каталоге **/dev**. Для SCSI-, SATA- и PATA-дисков в ОС **Linux** зарезервированы следующие имена специальных файлов устройств: **sda**, **sdb** и т. д. Если к компьютеру не подключены SCSI- и/или SATA- диски и никакие другие USB-устройства массовой памяти, то Card Reader будет устройством **sda**. В мультиформатных Card Reader'ах разъем для CF часто определяется, как **sdb**, но могут быть и исключения. Один из примеров распределения устройств в мультиформатном Card Reader'е приведен на рисунке 2. Несмотря на то, что этот драйвер работает надлежащим образом с подавляющим большинством подобных устройств, иногда встречаются Card Reader'ы, которые не определяются данным драйвером должным образом. Как правило, это дешёвые китайские подделки под известные фирмы (brand), либо просто *нечто*, сделанное “на коленке” и не соответствующее ни нормам, ни стандартам CFA (Compact Flash Association). К сожалению, в последнее время в продаже появляется больше именно таких, прельщающих своей дешевизной,

Card Reader'ов. Но в данном случае следует помнить народную мудрость “скупой платит дважды”, и не гнаться за сверхдешёвыми устройствами. Многолетний опыт работы с различными Card Reader'ами говорит о том, что на корректное определение Compact Flash в Card Reader'е существенное влияние оказывает качество кабеля, которым Card Reader подключается к разъему USB, а также нагрузочная способность по питанию порта USB, к которому подключен этот разъем. Для некоторых типов таких Card Reader'ов может помочь указание параметра **scsi_mod.max_luns=4** при загрузке системы. Указание этого параметра имело значение в ранних версиях **SSS Rescue CD**, в последних версиях он включен в ядре по умолчанию. В этом случае на приглашение загрузчика системы '**boot:**' (см. пункт 2) следует набрать:

```
rescue scsi_mod.max_luns=4
```

Если это не помогает, то без дополнительных ухищрений с такими Card Reader'ами работать невозможно. Сообщения драйвера о назначении устройств можно увидеть, набрав команду **dmesg**, либо **dmesg | less** или **dmesg | more**.

Следует отметить еще некоторые не очень приятные моменты, связанные с Card Reader'ами, которые стали проявляться при работе с CF больших объемов. Было замечено, что некоторые типы Card Reader'ов некорректно работают с CF объемом более одного Гигабайта. Это проявляется по-разному. Одни Card Reader'ы могут вообще не определять такие CF, другие могут неверно записывать информацию на такие CF. Последний вариант особенно неприятен, поскольку он выявляется только при монтировании CF после создания на ней файловой системы (см. пункт б).

Поскольку программа **CF_repair** работает только с Card Reader'ами, подключаемыми к шине USB персонального компьютера, то необходимо прояснить некоторые моменты, касающиеся этой шины. На настоящий момент имеется три версии стандарта (спецификации) шины USB: USB 1.1, USB 2.0 и USB 3.0. В настоящее время широко используются только первые два. В первую очередь эти стандарты различаются максимальной скоростью передачи данных (для USB 1.1

— 60 Мбит/с, а для USB 2.0—480 Мбит/с), что существенно для пользователей. Современные компьютеры, как правило, имеют два типа хост-контроллеров шины USB: USB 1.1 и USB 2.0. В свою очередь сами Card Reader'ы могут являться устройствами USB 1.1 или USB 2.0. В этих условиях возможны 4 варианта подключения Card Reader'а к компьютеру причем максимальную скорость передачи данных будет определять более медленное устройство или хост-контроллер. Для CF объемом менее одного гигабайта это не очень существенно. Но для CF больших объёмов в случае использования стандарта USB 1.1 время работы программы **CF_repair** может измеряться часами.

4. Предварительная проверка и подготовка CF

В ответ на приглашение командного процессора shell (**root:**) пользователь может выполнить любую программу, из имеющихся на CD. Перед выполнением программы **CF_repair** рекомендуется убедиться в том, что Master Boot Record (MBR) на CF удовлетворяет всем условиям, которые проверяет программа перед использованием носителя. Эти условия заключаются в следующем: наличие сигнатуры MBR, таблица разделов в MBR не испорчена и содержит хотя бы один основной раздел. Необходимо, чтобы этот раздел был “родным” (native) разделом **Linux**, то есть он должен иметь идентификатор системы – 0x83 (шестнадцатеричная нотация в языке C). Последняя проверка дополнительно контролирует правильное указание устройства при запуске программы **CF_repair**. Если не выполнится любое из этих условий, кроме последнего, которое программа **CF_repair** может выполнить сама, она завершается с выдачей соответствующего сообщения (см. пункт б). Выполнение этих требований можно проверить при помощи программы **fdisk** (редактор таблиц разделов жёстких дисков), которая запускается следующей командой:

```
fdisk [-c=dos] [-u=cylinders] /dev/sdx
```

, где **x** – обозначает букву конкретного устройства, соответствующего CF, может принимать значения строчных латинских букв от **a** до **z**. При таком варианте запуска программа

работает в диалоговом режиме и имеет исчерпывающий набор команд для обслуживания таблиц разделов (дополнительно выводится геометрия устройства). Ключ **-c=dos** необходим при работе с программой **fdisk** последних версий. Без этого ключа **fdisk** работает по новым правилам и не совместима с нашим подходом к разбиению носителя на разделы. Ключ **-u=cylinders** можно указывать для того, чтобы единицей отображения были цилиндры, а не сектора, как принято в последних версиях **fdisk**. Для данных ключей имеются команды **c** и **u**, соответственно. Эти ключи актуальны для **SSS Rescue CD** версии 3.3 и выше. Вывод таблицы разделов выполняется командой **p**, изменение типа раздела (идентификатора системы) – **t**, а интерактивная подсказка – команда **m**. Перевод программы в экспертный режим – **x**. Выход из программы по командам **w**, если надо сохранить изменения в MBR на носителе, либо **q** в остальных случаях. Более подробные сведения о программе **fdisk** можно получить из файлов **fdisk.txt** и **README.fdisk**, находящихся в этом же каталоге.

| Производитель | N Cyls | N Hdrs | Sec/Trk | Примечание |
|---------------------------|--------------|----------|----------|------------------------|
| Объём CF = 16 Mb | | | | |
| Nicon | 490 | 2 | 32 | |
| Feiya | 61 | 16 | 32 | |
| Объём CF = 32 Mb | | | | |
| PQI | 489 | 4 | 32 | |
| PQI Industrial | 489 | 4 | 32 | |
| Объём CF = 64 Mb | | | | |
| Digitex | 978 | 4 | 32 | |
| Adata | 978 | 4 | 32 | |
| TwinMOS | 978 | 4 | 32 | |
| KINGMAX | 1001 | 4 | 32 | |
| Transcend | 496 | 8 | 32 | |
| Объём CF = 128 Mb | | | | |
| SanDisk | 980 | 8 | 32 | |
| Kingston | 978 | 8 | 32 | |
| PQI (f1/40x) | 994 | 8 | 32 | |
| TwinMOS | 984 | 16 | 16 | |
| Digitex | 978 | 8 | 32 | |
| Samsung | 496 | 16 | 32 | |
| PRETEC | 978 | 8 | 32 | |
| Transcend | 978 | 8 | 32 | |
| Transcend 45x | 980 | 8 | 32 | |
| ICF 1000 (iEi) | 492 | 16 | 32 | UDMA33 |
| Объём CF = 256 Mb | | | | |
| SMART | 980 | 16 | 32 | MWDMA2 |
| Kingston | 502 | 16 | 63 | MWDMA2 |
| ICF 1000 (iEi) | 1000 | 16 | 32 | |
| swissbit | 980 | 16 | 32 | UDMA66 |
| Объём CF = 512 Mb | | | | |
| Kingmax | 993 | 16 | 63 | |
| Объём CF = 1024 Mb (1 Gb) | | | | |
| Transcend 80x | 1007 2014 | 32 16 | 63 63 | LBA в BIOS Реальная |
| Kingston | 1007 2015 | 32 16 | 63 63 | LBA в BIOS Реальная |
| Объём CF = 4096 Mb (4 Gb) | | | | |
| Kingston elite pro (133X) | 7785 | 16 | 63 | До UDMA100 |
| Kingston CF | 7745 | 16 | 63 | |
| Transcend CF4Gb/TR133 | 971 | 128 | 63 | До UDMA66 |
| Transcend CF4Gb (133X) | 7769 | 16 | 63 | UDMA33 |

| | | | | |
|---------------------------|-------|----|----|--------|
| Объём CF = 8192 Mb (8 Gb) | | | | |
| Kingston | 15636 | 16 | 63 | UDMA33 |

Табл. 1. Геометрия известных типов Compact Flash.

По поводу геометрии CF следует поговорить отдельно, потому что несоответствие геометрии в разных вариантах подключения их (CF) к компьютеру является довольно частой причиной неработоспособности всей системы. Прежде всего, следует заметить, что CF может поддерживать на своей внешней шине два типа интерфейсов (режима работы): **True IDE** и PCMCIA (PC Card). В процессорных модулях форматов PC/104 или ETX, используемых нами, разъём для установки CF всегда подключен к одному из IDE- или PATA-интерфейсов, поэтому в этих модулях CF всегда представляется в системе как IDE- или SCSI-диск. Исторически сложилось так, что любые диски и дискеты, в том числе и IDE, имеют ряд параметров, определяющих их объём, и называемых геометрией. К геометрии относятся: количество цилиндров, количество головок (дорожек) и число секторов на дорожку. В интерфейсе IDE для дисков имеется специальная команда **Identify Drive**, которая выдаёт все параметры накопителя, в том числе и геометрию. Эту геометрию производитель устанавливает в контроллере CF, и на основании её контроллер осуществляет доступ к секторам на CF, поэтому назовем её реальной геометрией. Драйвер дисков операционной системы, установленной на CF, видит как раз реальную геометрию и определяет все параметры носителя исходя из этой геометрии. Однако параметры разделов определяются на основании информации из таблицы разделов в MBR. В случае подключения CF к компьютеру посредством Card Reader'a она представляется как SCSI-диск с транспортным протоколом USB, и драйвер определяет её геометрию на основании информации из таблицы разделов в MBR, потому что в данном случае SCSI- транспорт не имеет команды подобной **Identify Drive**.

В последнее время наблюдается тенденция к существенному увеличению объёмов почти всех типов флэш-карт (CF, MMC, SD, MS). При этом флэш-карты меньших объёмов просто перестают производить и продавать. Приобретение CF объёмом до одного гигабайта сейчас является почти невыполнимой задачей. Дополнительно следует отметить, что производители в основном предполагают использование флэш-карт в устройствах бытовой электроники (фотоаппараты, MP3 плееры и т.п.). В соответствии с этим данные в таблице разделов формируются таким образом, чтобы свести к минимуму противоречие, вызванное ограничением разрядности полей, которые определяют геометрию раздела в элементе таблицы разделов, и объёмом раздела на флэш-карте (в частности на CF). Ограничения разрядности полей параметров геометрии исторически существуют с момента появления компьютеров PC/XT и накладываются функциями системного BIOS. Они составляют 1024 цилиндра, 255 дорожек (головок) и 63 сектора на дорожку, что даёт максимально возможное количество секторов на носителе равное 16450560 секторов или 8422686720 байтов (8,4 Гб). Если на CF объёмом до одного или двух гигабайтов это противоречие практически не сказывается, то для CF больших объёмов (4, 8, 16 и т. д. Гб) избежать его уже не удастся. И, если в таблице разделов в MBR не указать реальные параметры геометрии, то можно получить совсем неожиданные результаты и CF с неработоспособной системой.

Если информация в MBR испорчена или не совсем корректна, то драйвер устанавливает геометрию в соответствии со спецификацией **X3T10** (SCSI-2 Common access method transport and SCSI interface module), которая может не соответствовать реальной геометрии. В этом случае необходимо установить реальную геометрию. Это можно сделать при помощи программы **fdisk** в экспертном режиме, как указано ниже.

Для создания таблицы разделов в MBR, в случае её отсутствия, или корректировки неправильной таблицы разделов сначала необходимо определить реальную геометрию этой CF, а затем выполнить программу **fdisk**. Определение реальной геометрии в случае отсутствия такой информации можно выполнить в **Setup** системного BIOS того процессорного модуля, где предполагается использование CF в качестве твердотельного заменителя “жесткого” диска. Информация о геометрии отображается в пункте меню определения параметров “жесткого” диска. Геометрия известных автору типов CF приведена в таблице 1. Если в таблице разделов на CF имеется хотя бы один раздел, то перед корректировкой геометрии все существующие разделы

необходимо уничтожить командой **d**. В диалоге экспертного режима программы **fdisk** (вход в этот режим выполняется по команде **x**) устанавливается требуемая геометрия; командами **s**, **h** и **c** устанавливается число секторов на дорожке, головок (дорожек) и цилиндров, соответственно, после чего выполняется переход в обычный режим (команда **r**). В обычном режиме создается, как минимум, один раздел (команда **n**) и сформированная MBR сохраняется на CF командой **w**. Команда **w** автоматически завершает выполнение программы **fdisk**. Если не создать новый раздел и не сохранить его на CF, то информация о геометрии будет утеряна, и все действия по формированию новой MBR придется повторить. Пример сеанса работы с программой **fdisk** для установки требуемой геометрии приведен в приложении 1.

5. Работа с программой CF_repair

Внимание! Внимание! Внимание!

Во избежание всяких неприятностей, связанных с порчей информации на основном диске компьютера, пользователям компьютеров, имеющих SCSI-, SATA- или PATA-диски, **необходимо с особым вниманием отнестись к выбору устройства для CF** в самом начале работы программы и на диагностику программы на этапе проверки состояния CF. В Linux SATA- и PATA-диски представляются в системе как SCSI, и система назначает им такие же имена устройств, как и для SCSI-дисков.

Запуск программы восстановления CF (**CF_repair**) выполняется следующей командой (можно ввести **CF**, после чего нажать клавишу <Tab>):

```
CF_repair [-cxh]
```

```
CF_repair [-ch] -b <имя-конф-файла>
```

В обоих вариантах запуска ключ **-h** указывает вывод справочного сообщения и выход из программы. Ключ **-c** задаёт пропись раздела или разделов CF нулями перед созданием файловой системы или файловых систем. Ключ **-x** указывает включение “экспертного” режима. Экспертный режим предназначен для сокращения времени работы программы **CF_repair** в ситуациях, когда предыдущий запуск завершился обнаружением какой-либо ошибки после выполнения части шагов формирования CF и рекомендуется только опытным пользователям. Если ключи **-c** и **-x** указаны одновременно, то действие ключа **x** отменяется. Второй вариант запуска (указан ключ **-b**) определяет работу программы в так называемом “пакетном” режиме. В этом режиме программа читает все параметры настройки из текстового файла, имя которого указывается параметром **<имя-конф- файла>**. Содержимое и формат этого файла приведены в разделе 5.2. В пакетном режиме также отменяется действие ключа **-x**.

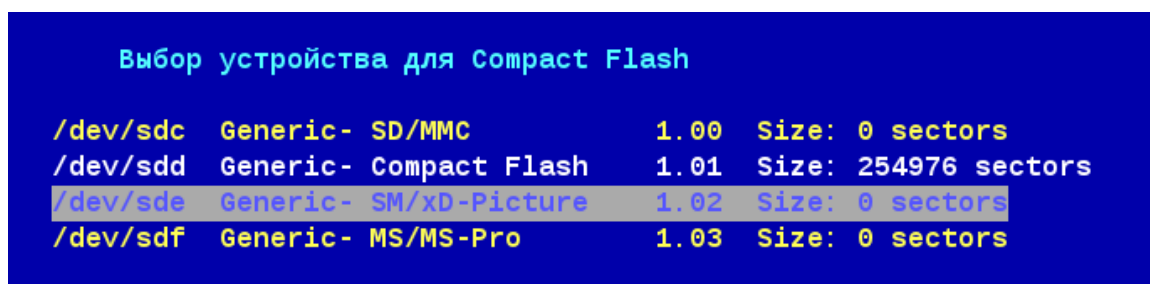
В общих чертах работа программы **CF_repair** заключается в следующем. Сначала программа выполняет проверки состояния носителя (CF) и занятость раздела или разделов CF системой. В зависимости от результатов этих проверок она либо прекращает свою работу с выдачей соответствующей диагностики (см. пункт 6), либо переходит к этапу указания параметров конфигурации. На этом этапе пользователь указывает конфигурацию требуемых ему типа процессорного модуля, операционной и прикладной систем, а также определяет конфигурацию локальной сети, в которой этот модуль будет функционировать. В соответствии с выбранными и введёнными параметрами **CF_repair** создаёт на CF файловую систему или файловые системы, заполняет её или их необходимыми программами и прочими файлами и выполняет соответствующие настройки для всей системы и локальной сети.

5.1. Работа с программой в диалоговом режиме

В этом режиме работы программа предоставляет визуальный оконный интерфейс с пользователем. Это более удобно и понятно, чем ввод параметров из командной строки.

Выбор устройства для CF выполняется в диалоге в самом начале работы программы. Такой

вариант выбора устройства для CF вместо указания имени устройства в командной строке при запуске программы, как это было сделано в предыдущих версиях программы, реализован по простой причине – исключить потерю данных на дисках пользователей. Многие пользователи слабо ориентируются в терминологии ОС Linux, и все эти **sda**, **sdb** и т. п. являются для них шаманскими заклинаниями, а некоторым пользователям просто лень внимательно прочитать эту инструкцию. В основном эти причины несмотря на настойчивые и многочисленные предупреждения, довольно часто приводят к потере данных на жестких дисках пользователей. В верхней части главного окна программы выводится список SCSI-дисков, подключенных к компьютеру посредством шины USB, а в нижней подсказка по использованию клавиш. Каждая строка этого списка представляет отдельное устройство. В первом столбце выводится имя специального файла устройства, через него программа будет работать с CF. Во втором столбце выводится имя производителя устройства, а все поля до слова **Size** определяют название устройства и его версию. Для CF будут выведены имя производителя и обозначение устройства для Card Reader'a, а не самой CF. Пример показан на рисунке 2. Весь список устройств отображается символами желтого цвета на синем фоне. Текущее выбираемое устройство выделяется символами синего цвета на белом фоне. Выбор устройства выполняется нажатием клавиши **<Enter>** на устройстве, являющемся выбираемым в данный момент. Выбранное пользователем устройство отображается символами белого цвета на синем фоне.



```
Выбор устройства для Compact Flash
/dev/sdc  Generic- SD/MMC          1.00  Size: 0 sectors
/dev/sdd  Generic- Compact Flash     1.01  Size: 254976 sectors
/dev/sde  Generic- SM/xD-Picture     1.02  Size: 0 sectors
/dev/sdf  Generic- MS/MS-Pro         1.03  Size: 0 sectors
```

Рис. 2. Образец списка устройств для указания устройства для CF

Внимание! При указании неверного устройства для CF имеется реальная опасность потерять все данные на этом устройстве.

После успешной проверки состояния CF, на экран выводится главное окно, в левой верхней части которого указаны основные параметры настройки системы. В правой верхней части отображаются геометрия CF и размеры общего объема и свободного места в её файловой системе или файловых системах. Геометрия выводится для дополнительного визуального контроля. В двух нижних строках окна находится краткая подсказка по использованию клавиатуры при указании параметров или выполнении других действий. При этом автоматически выводится окно для выбора типа процессорного модуля, его скриншот приведен на рисунке 4. Скриншот одного из вариантов формата главного окна в случае успешного завершения программы **CF_repair** приведен на рисунке 3.

В любой момент выполнение программы можно прервать нажатием клавиш **<Ctrl+C>** (эта запись означает, что при нажатой клавише **<Ctrl>** нужно нажать клавишу **<C>**), после чего в двух нижних строках главного окна выводятся следующие сообщения:

Interrupt: Выполнение программы прервано пользователем!
Нажмите любую клавишу для завершения CF_repair.

После нажатия пользователем любой клавиши происходит немедленное завершение работы программы. Если пользователь прервал работу программы в тот момент, когда началась запись на CF, то при следующем запуске программы для этой CF процесс восстановления рекомендуется выполнить с самого начала (не указывать экспертный режим).

Поля для указания параметров делятся на два типа: постоянные – для выбора заведомо известных вариантов и переменные – для ввода настраиваемых параметров с клавиатуры.

Постоянные параметры указываются в пяти первых строках (см. рис. 3). Указание постоянных параметров выполняется в отдельных, специальных окнах и более подробно будет объяснено позже. Изначально переменные поля пустые и значения в них должен ввести пользователь, руководствуясь информацией о своей локальной сети. Позиция ввода следующего символа в переменных полях указывается курсором. Перемещение между полями выбора и/или ввода выполняется клавишами стрелок и <Tab>. Текущее выбираемое или изменяемое поле выделяется символами синего цвета на белом фоне. Для переменных полей такое цветовое выделение показывает максимальный размер поля. Фиксация значений постоянных параметров и завершение ввода переменных параметров выполняется нажатием клавиши <Enter> в выбранном или введенном поле. Зафиксированные пользователем значения параметров отображаются символами белого цвета на голубом фоне. Если в переменном поле были введены ошибочные символы, то имеется возможность их исправления. В этом случае клавишами <Backspace> или из конца поля удаляются все ошибочные символы до последнего правильного, после чего можно продолжить ввод новых символов в этом поле. При вводе запрещённых символов и/или нажатии на необработываемые служебные клавиши программа не выполняет никаких действий, но в области диагностики, расположенной в нижней части основного окна, выводится сообщение **Необработываемый код клавиши:** и шестнадцатеричное значение кода нажатой клавиши.

```

Compact Flash (CF) Repair Tool v 1.17 (by VEV)
Wed Apr 6 2011 10:46:27

Тип процессорного модуля: LX                ID диска: 2527A2C7
Серия ядра Linux: 6                         Геометрия /dev/sdb:
Прикладная система: DGW                   Цилиндров: 492
Вариант системы: Classic                   Дорожек: 16
Сетевой boot протокол: Статич. IP         Сект./дор: 32
Имя хоста (Hostname): ttdgw               Всего секторов: 251904
Имя домена: proton-sss.ru                 /dev/sdb1
IP адрес: 192.168.12.222                  Общий объем: 121972 Kib
Маска (под)сети: 255.255.0.0             Свободно: 58732 Kib
IP адрес шлюза для INet: 192.168.147.3
IP адрес сервера DNS: 192.168.147.8,192.168.147.3

Чтобы Вы могли проверить файлы конфигурации и другие данные на вновь
созданной Compact Flash, её файловая система всё ещё примонтирована к
каталогу: /mnt/cf1.
Если в проверке нет необходимости, можно завершить программу клавишей <F10>.
В противном случае перед изъятием Compact Flash из Card Reader'a или перед
перезапуском системы необходимо обязательно демонтировать её файловую
систему вручную, потому что она НЕ БУДЕТ ДЕМОНТИРОВАНА автоматически.
Для демонтажа следует выполнить команду 'umount /mnt/cf1'.
После чего можно изъять Compact Flash из Card Reader'a и перезапустить
компьютер командой 'reboot' или кнопкой 'Reset'.

CF_repair успешно выполнена
Любая клавиша - выход из CF_repair, <F10> - выход с демонтаживанием ФС на CF.
Copyright (c) 2011 SSS

```

Рис. 3. Формат главного окна программы CF_repair.

Перед тем как привести объяснение назначения параметров настройки необходимо сделать некоторые пояснения. Для процессорных модулей **MOPSlcdGX1**, **MSM586SEN** и **ETX-DC** каждая система собиралась и настраивалась только под конфигурацию оборудования этого конкретного модуля, и на других аналогах таких модулей может не выполнять некоторых функций. С процессорными модулями (ПМ) на базе **AMD Geode LX 800** и **AMD Geode GX** ситуация совсем иная. Имеется множество разновидностей (типов) модулей на базе этих процессоров, выполненных как в формате PC/104, так и в формате ETX. Дополнительно все модули на базе этих процессоров имеют встроенные аппаратные средства для измерения температуры процессора и самого модуля, которые называются **hardware monitor** или сокращенно **HW Mon**. Основной набор внешних устройств во всех реализациях процессорных модулей почти идентичен за исключением типа сетевого адаптера, способа подключения разъёма для CF и типа **HW Mon**. Поэтому для этих модулей была создана единая система, которая

учитывает особенности всех известных нам процессорных модулей на базе этих процессоров, а настройка на конкретный вариант выполняется дополнительным пунктом (в отдельном окне) после установки основных и, если было необходимо, сетевых параметров. Значения дополнительных параметров для возможных модулей типа **LX** приведены в таблице 2. В этой таблице после названия модуля в скобках указан его производитель. Серым фоном в таблице отмечены процессорные модули, серийно поставляемые в изделиях ООО «СПЕЦСТРОЙ-СВЯЗЬ».

Все окна для выбора постоянных параметров имеют похожий вид и идентичное управление. Список вариантов выбора выводится символами синего цвета на сером фоне. Текущее выбираемое значение параметра (маркер) выделяется символами белого цвета на синем фоне. Перемещение маркера по списку выполняется клавишами стрелок и **<Tab>**. Выбор требуемого значения параметра выполняется нажатием клавиши **<Enter>**, при этом выбранное значение отображается символами белого цвета на сером фоне. Фиксация выбранного значения выполняется нажатием клавиши **<F10>**, при этом текущее окно закрывается и автоматически открывается окно для выбора следующего постоянного параметра.

| Тип процессорного модуля | Формат исполнения | У-во загрузки | Тип Ethernet адаптера | HW Monitor |
|--------------------------|-------------------|--------------------------|-----------------------|-----------------|
| PM-GX (IEI) | PC/104 | hdb | 8139too | w83627ehf |
| MSM800BEV (DLAG) | PC/104 | hda/hdb ¹ | e100 | scx200_acb,lm83 |
| PM-6100 (Aewin) | PC/104 | hda | e100 | w83627hf |
| ETX-LX (Kontron) | ETX | sda/hda/hdb ² | e100 | w83627hf |
| IEM-LX (IEI) | ETX | hda | 8139too | w83627ehf |
| EmETX-a5363 (ARBOR) | ETX | hdb | e100 | w83627hf |
| ESM-2545 (Avalue) | ETX | hda | 8139too | w83627hf |

Табл. 2. Дополнительные параметры настройки для **LX**.

Далее приводится содержательное разъяснение основных параметров настройки и их возможных значений.

Параметр **Тип процессорного модуля** выбирается в отдельном окне, вид которого показан на рисунке 4. Выбор выполняется из значений:

MSM – для процессорных модулей **MSM586SEN** или **MSM586SEL** на базе микроконтроллера **Élan SC520** (для этих процессорных модулей рекомендуется выбирать ядро Linux серии 2);

MOPS – для процессорных модулей **MOPSlcdGX1** на базе процессора **GeodeGX1**;

LX – для процессорных модулей на базе новых процессоров **AMD Geode LX 800** или **AMD Geode GX** (для этого варианта автоматически устанавливается система на базе ядра 2.6.18);

atom – для процессорных модулей **ETX-DC** на базе новых процессоров **Intel Atom N270** (для этого варианта автоматически устанавливается система на базе ядра 2.6.34).

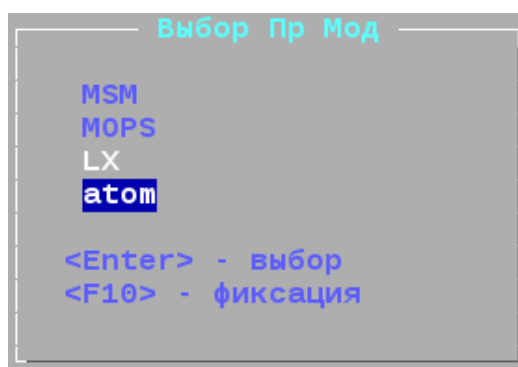


Рис. 4. Окно выбора типа процессорного модуля.

¹ Зависит от наличия/отсутствия на этом модуле перемычки J1 – **hda/hdb** соответственно.

² Зависит от того, в каком разъёме будет установлена CF: если на процессорном модуле, то – **sda**; если на БУКМ-Е или БУП-Е, то в зависимости от наличия/отсутствия перемычки **X26** – **hda/hdb** соответственно.

Для корректного восстановления CF пользователь должен точно знать тип процессорного модуля, установленного в его оборудовании. Если создать CF с несоответствующим типом процессорного модуля – она не будет работать.

Существенное замечание: для процессорных модулей **MSM586SEN** или **MSM586SEL** не рекомендуется использовать CF объёмом более 512 Мб, поскольку эти модули не могут работать с CF большего объёма по причине ограничений нестандартной реализации контроллера дисков IDE. Но лучше для этих модулей использовать CF объёмом не более 256 Мб.

Параметр **Серия ядра Linux** выбирается в отдельном окне, вид которого показан на рисунке 5. Выбор выполняется из значений: **2** – для ядра 2.2.16, **6** – для ядра 2.6.10 (2.6.18, 2.6.34) и **6i** – для ядра 2.6.10 (2.6.18, 2.6.34) с использованием **initramfs**. Здесь следует заметить, что для процессорных модулей **MSM586SEL** не следует устанавливать систему на базе ядер серии 2.6. В этих модулях контроллер IDE, к которому подключается CF, имеет очень специфическую реализацию и с драйвером жестких дисков из ядра 2.6 работает неустойчиво, что выражается большим количеством сбоев при работе с CF.



Рис. 5. Окно выбора серии ядра Linux.



Рис. 6. Окно выбора прикладной системы.

Параметр **Прикладная система** выбирается в отдельном окне, вид которого показан на рисунке 6. Выбор выполняется из значений: **DGW** – для работы в составе IP-шлюза Diamond GateWay, **diamond** – для работы в составе ЦАТС «Алмаз 1», **DIP** – для работы в составе ЦАТС «Diamond IP», **VIP** – для работы в составе ЦАТС «Вектор IP», либо **vector2m** – специальная реализация пульта оператора. Следует отметить, что варианты **VIP**, **DIP** и **vector2m** реализуются только на процессорных модулях формата ETX, поэтому для них следует устанавливать тип процессорного модуля **LX** или **atom**. В настройках процессорного модуля **LX** (см. ниже) следует указывать только последние четыре варианта, потому что первые три – это процессорные модули в формате PC/104. Соответствие этих параметров проверяется программой, и в случае неверного их указания выдается сообщение об ошибке.

Параметр **Вариант системы** выбирается в отдельном окне, вид которого показан на рисунке 7. Выбор выполняется из значений: **BusyBox** – для установки упрощённой системы, требующей малый объём (до 32 Мб) на Compact Flash (основу этой системы составляют так называемые “мультибинарники” **BusyBox** и **TinyLogin**), либо **Classic** – для установки обычной системы (такой как в большинстве дистрибутивов ОС **Linux**) с минимально необходимым требующимся набором системных программ и утилит. В последнем случае объём CF должен быть не менее 64 Мб. Программа отслеживает объём CF и вариант устанавливаемой системы, и после ввода параметров может выдать предупреждение о возможном дефиците памяти на CF. Возможные варианты систем для всех типов процессорных модулей и серий ядра Linux приведены в таблице 3. В таблице используются следующие сокращения: Cls – **Classic**, BB – **BusyBox**.

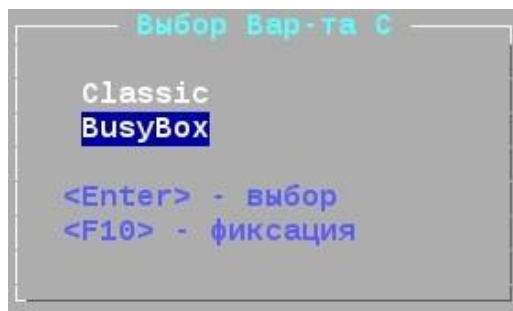


Рис. 7. Окно выбора варианта системы.

| | MSM | MOPS | LX | Atom |
|-----------------|-----------------|-----------------|-----------------|-----------------|
| 2 | Cls/BB | Cls/BB | – | – |
| 6 | Cls (2.6.10) | Cls (2.6.10) | Cls (2.6.18) | Cls (2.6.34) |
| 6i ³ | – | Cls (2.6.10) | Cls (2.6.18) | Cls (2.6.34) |

Табл. 3. Возможные варианты систем на **SSS Rescue CD**.

Способ конфигурации локальной сети (**Сетевой boot протокол**) выбирается в отдельном окне, вид которого показан на рисунке 8. Выбор выполняется из вариантов: **DHCP** – настройка сетевых параметров по протоколу DHCP посредством клиента **dhcpcd**, либо **Статич. IP** – ручная настройка с заранее известными параметрами (необходимыми IP-адресами).

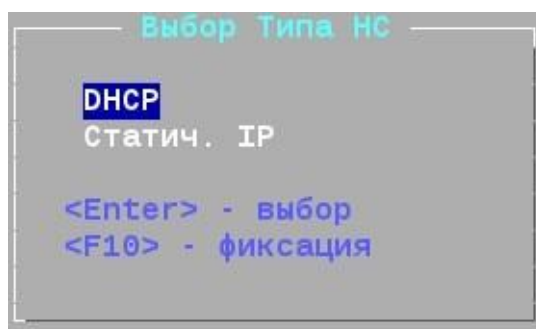


Рис. 8. Окно выбора сетевого boot-протокола.

Примечание. При выборе варианта **DHCP** могут возникнуть проблемы с определением IP-адреса модуля при последующем обращении к нему по сети. Чтобы избежать этого следует обязательно назначить уникальное значение параметру **Имя хоста** потому, что, в дальнейшем, только посредством указанного сетевого имени можно будет обратиться к модулю по сети и определить все остальные сетевые настройки этого процессорного модуля (для этого можно использовать программы **nslookup**, **host** и т. п.).

Сетевое имя процессорного модуля (параметр **Имя хоста**) – это имя, под которым процессорный модуль можно идентифицировать в сети при наличии службы DNS. Если не указывается,

³ В процессорном модуле требуется наличие ОЗУ не менее 128 Мбайт.

то программа подставит в качестве этого параметра значение параметра **Прикладная система (DGW, VIP, DIP, diamond или vector2m)**.

Параметр **Имя домена** определяет имя подсети, в которой будет функционировать процессорный модуль, для которого формируется CF. При нажатии клавиши <**F5**> программа автоматически присваивает этому параметру значение **proton-sss**, которое является именем домена ООО «СПЕЦСТРОЙ-СВЯЗЬ». Для локальной сети пользователя значение этого параметра можно узнать у администратора сети. При указании неверного имени домена служба **DNS** будет сильно тормозить работу с сетью других сетевых сервисов, например, **FTP**.

Завершение ввода основных параметров указывается нажатием функциональной клавиши <**F10**>. Допускается не указывать значения параметров **Имя хоста** и/или **Имя домена**. Если для конфигурирования локальной сети используется ручная настройка с заранее известными параметрами (вариант **Статич. IP** параметра **Сетевой boot протокол**), то пользователь должен ввести дополнительные сетевые параметры, которые появляются на экране после завершения ввода основных параметров. Перечень сетевых параметров, необходимых для системы на CF приведен ниже:

- сетевой (**IP**) адрес процессорного модуля – параметр **IP адрес**;
- параметр **Маска (под)сети** определяет маску сети, в которой будет функционировать данный процессорный модуль;
- параметр **IP адрес шлюза для Inet** определяет сетевой адрес шлюза (компьютера), через который будет происходить выход в Интернет.
- параметр **IP адрес сервера DNS** определяет сетевой адрес **DNS**-сервера (допускается не указывать значение этого параметра).

Действительные значения сетевых параметров можно узнать у администратора вашей локальной сети. Если нажать клавишу <**F5**> или <**F6**>, то программа всем этим параметрам присвоит значения по умолчанию, но они имеют смысл только для локальной сети ООО «СПЕЦ - СТРОЙ-СВЯЗЬ». Клавиша <**F5**> устанавливает умолчания для подсетей 172.16.32.0 – 172.16.63.0, а <**F6**> – для подсети 192.168.0.0.

Все сетевые параметры представляют собой IP-адреса и должны вводиться в десятично-точечном формате. После нажатия клавиши <**Enter**>, которая является завершителем ввода параметра, он проверяется на корректность синтаксиса и допустимость десятичных значений (должны быть в пределах 0-255). В случае ввода некорректного значения параметра программа удаляет все некорректные символы до конца параметра и выводит предупреждающее сообщение в области диагностики. Если пользователь не исправит некорректный сетевой параметр, то программа **CF_repair** все равно обнаружит ошибку на этапе настройки сетевой подсистемы и не завершит должным образом восстановление CF. Завершение установки всех сетевых параметров указывается нажатием функциональной клавиши <**F10**>.

В том случае, когда для параметра **Тип процессорного модуля** было установлено значение **LX**, после ввода всех необходимых параметров на экране появляется окно **Выбор модуля LX/GX**, где пользователь должен установить конкретный тип процессорного модуля, для которого выполняется формирование CF. Вид этого окна приведен на рисунке 9. В этом окне выводится список названий типов процессорных модулей, который приведен в таблице 2. Список выводится символами синего цвета на сером фоне. Текущий выбираемый тип процессорного модуля (маркер) выделяется символами белого цвета на синем фоне. Перемещение маркера по списку выполняется клавишами стрелок и <**Tab**>. Выбор типа процессорного модуля выполняется

нажатием клавиши

<Enter>, при этом выбранный тип отображается символами белого цвета на сером фоне. Если для выбранного типа процессорного модуля имеется несколько вариантов устройств загрузки (см. таблицу 3), то справа от списка названий типов процессорных модулей выводится список возможных для выбранного процессорного модуля устройств загрузки, и пользователь должен выбрать нужное ему устройство. На рисунке это показано для модуля **ETX-LX**. Назначение клавиш для выбора устройства загрузки такое же, как для выбора типа процессорного модуля. Критерии для выбора устройств загрузки приведены в примечаниях к таблице 2. После выбора типа модуля, и если это требовалось, устройства загрузки в нижней строке окна выводится строка с выбранными параметрами. Для процессорных модулей, у которых нет вариантов выбора устройств загрузки, процедура их выбора не выполняется, и строка с параметрами выводится после выбора типа процессорного модуля. Завершение установки типа процессорного модуля и устройства загрузки для него происходит после нажатия клавиши <F10>, после чего это окно исчезает с экрана. Значения параметров «тип Eth адаптера» и «HW Monitor» устанавливаются автоматически по типу процессорного модуля.

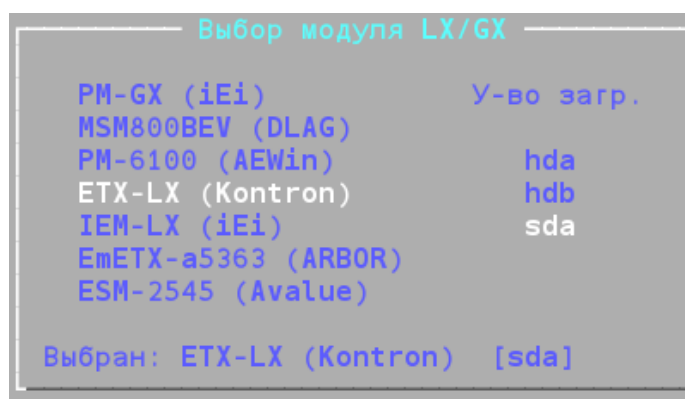


Рис. 9. Формат окна уточнения выбора процессорных модулей LX/GX.

Далее, если это необходимо, программа выполняет модификацию таблицы разделов в MBR, которая может заключаться в следующем: изменение идентификатора раздела или разделов, добавление второго раздела, удаление второго раздела или изменение размеров разделов. Следует помнить, что этот шаг может не выполняться, если не требуется коррекция таблицы разделов. Перед записью модифицированной MBR на CF выдается запрос на подтверждение модификации MBR. При этом в главном окне выводится сообщение:

Подтверждаете модификацию MBR? (1-да,0-нет)

Если пользователь подтверждает модификацию (вводит <1>, а затем нажимает клавишу <Enter>), программа записывает MBR на CF, выполняет повторную инициализацию CF и продолжает работу. В противном случае (пользователь вводит <0>, а затем нажимает клавишу <Enter>) работа программы прекращается.

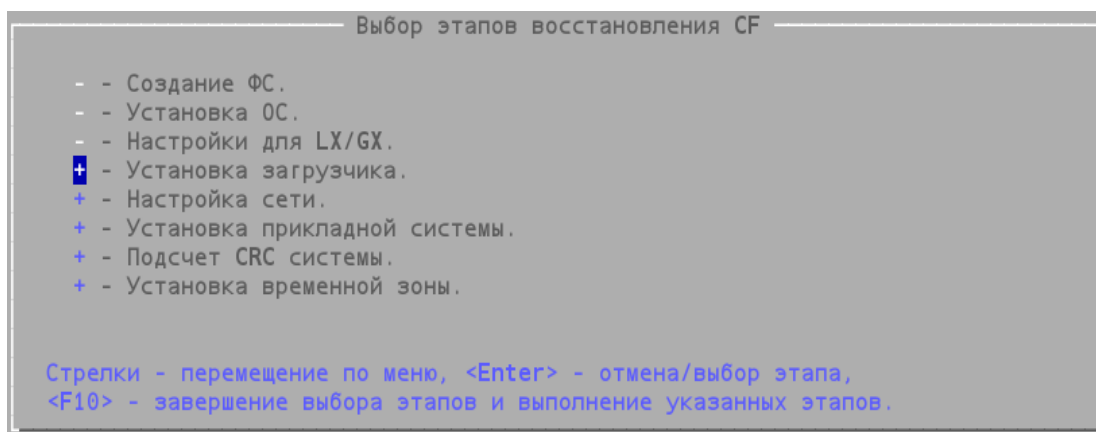


Рис. 10. Пример окна выбора шагов восстановления в случае обычной системы.

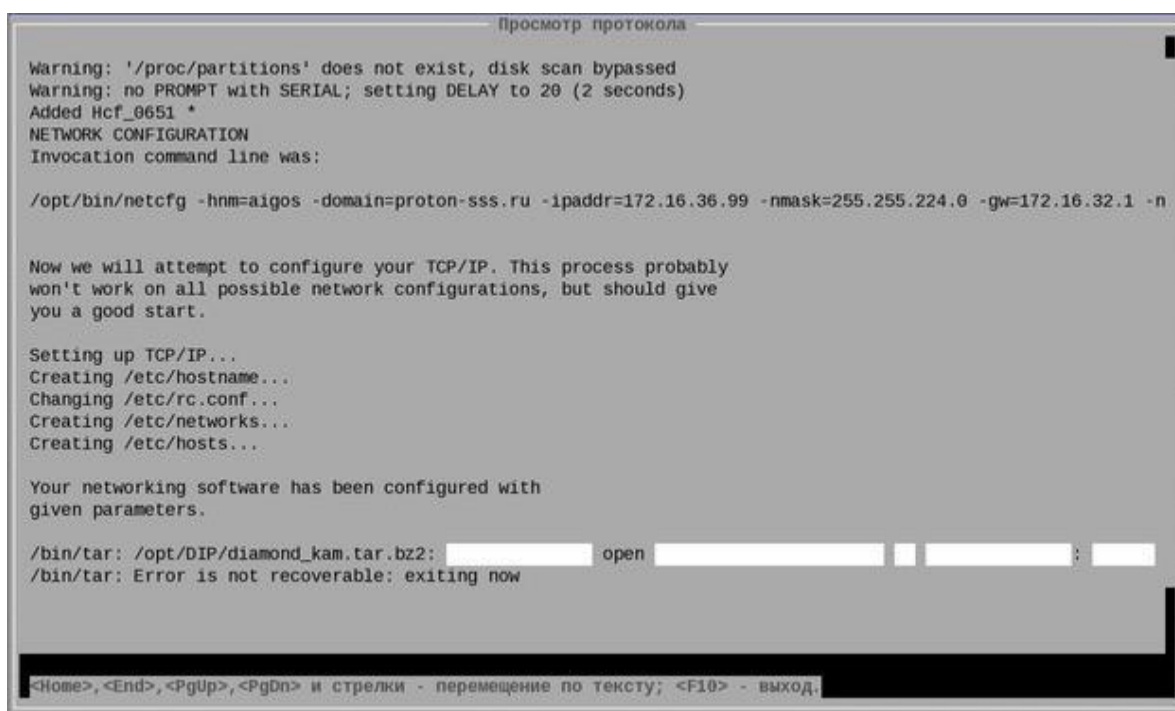
В случае указания “экспертного” режима после ввода всех настраиваемых параметров выводится окно меню для указания этапов (шагов) восстановления. Вид этого окна для создания обычной системы показан на рисунке 10 (в скобках показано название этапа для системы с **initramfs**). Сначала все этапы включены, что показывают знаки ‘+’ слева от названий этапов. Перемещаясь по меню и нажимая клавишу <Enter>, Пользователь может выключать ненужные и включать требуемые этапы. Выключенные этапы отмечаются знаками ‘-’ слева от названий этапов. Следует заметить, что нельзя выключать этапы в шахматном порядке. Работа программы построена так, что должны быть выполнены все этапы, начиная с первого выбранного и до конца списка меню.

Перед созданием файловой системы на CF программа выводит запрос:

Выполнять проверку сбойных блоков? (1-да,0-нет)

При утвердительном ответе во время создания файловой системы на CF будет выполнена проверка наличия сбойных блоков, что занимает некоторое время (весьма значительное для CF больших объемов). Отказываться от этой проверки для сокращения времени имеет смысл только тогда, если CF достаточно новая или только что приобретена.

Затем программа начинает работу по формированию новой CF. При этом в нижней части экрана выводится название выполняемого этапа. Протоколы выполнения системных программ выводятся в дополнительном окне, которое располагается в нижней половине экрана между областью параметров и областью диагностики внизу главного окна. Окно протоколов имеет черный фон. После выполнения этапа проверки состояния CF справа от области ввода параметров выводится геометрия CF, её общий объём и свободное место на CF. Значение свободного места на CF корректируется после каждого последующего этапа, выполняющего запись файлов на CF. В общем случае программа рассчитана на полный вывод при ширине экрана в 128 символов. Но при загрузке системы без фрейм-буфера (**rescue-nofb**) ширина экрана будет 80 символов. В этом случае при создании системы с **initramfs** информация о свободном месте для второго раздела отображаться не будет.



```
Просмотр протокола
Warning: '/proc/partitions' does not exist, disk scan bypassed
Warning: no PROMPT with SERIAL; setting DELAY to 20 (2 seconds)
Added hcf_0651 *
NETWORK CONFIGURATION
Invocation command line was:

/opt/bin/netcfg -hnm=aigos -domain=proton-sss.ru -ipaddr=172.16.36.99 -nmask=255.255.224.0 -gw=172.16.32.1 -n

Now we will attempt to configure your TCP/IP. This process probably
won't work on all possible network configurations, but should give
you a good start.

Setting up TCP/IP...
Creating /etc/hostname...
Changing /etc/rc.conf...
Creating /etc/networks...
Creating /etc/hosts...

Your networking software has been configured with
given parameters.

/bin/tar: /opt/DIP/diamond_kam.tar.bz2:  open  : 
/bin/tar: Error is not recoverable: exiting now

<Home>, <End>, <PgUp>, <PgDn> и стрелки - перемещение по тексту; <F10> - выход
```

Рис. 11. Пример окна просмотра протокола.

Если в процессе работы программы **CF_repair** выявляется нештатная ситуация

(обнаруживается ошибка), то причина ошибки будет выведена либо в строке диагностики (третья снизу строка), либо в окне протоколов. В окне протокола, как правило, можно получить уточнение причины возникновения нештатной ситуации. В таких случаях программа выводит соответствующую диагностику и предупреждает о завершении работы после того, как пользователь нажмет любую клавишу. Если после последнего выполненного этапа был сохранен протокол, то дополнительно перед завершением работы программы пользователь может просмотреть его, нажав клавишу <F3>. Просмотр протокола полезен, если пользователю по какой-либо причине не удалось отследить диагностику ошибок в окне протокола. Пример окна просмотра протокола приведен на рисунке 11. Имеется возможность сохранения протокола работы в файл `/var/log/CF_repair.log`. Для этого программа выводит запрос:

Сохранить протокол работы в файл? (1-да,0-нет)

При утвердительном ответе буфер протокола записывается в файл и выводится сообщение:

Протокол работы сохранен в файле: `/var/log/CF_repair.log`

После чего следует нажать любую клавишу для продолжения завершения программы.

При выполнении этапа восстановления прикладной системы программа `CF_repair` выводит на экран список всех файлов, находящихся в каталоге выбранной прикладной системы (`DGW`, `VIP`, `DIP`, `diamond` или `vector2m`), предоставляя пользователю самому выбрать нужную версию программы. Это позволяет иметь несколько версий прикладных систем (программ) на CD. Выбор конкретной версии программы выполняется нажатием клавиши <Enter> на выделенном имени файла. Прикладная система устанавливается в каталог `/usr` на CF. Программа предполагает, что каждая версия прикладной системы хранится в виде отдельного `tar`-архива, упакованного программой сжатия `gzip`, `bzip2` или `lzma` (определяется по расширению файла архива). Такой способ хранения предпочтительнее по нескольким причинам: во-первых, компактность, во-вторых, сохранение исходных атрибутов файлов, характерных для UNIX-подобных систем. Это особенно важно в тех случаях, когда происходит перепись файлов из ОС `Linux` в файловые системы других ОС, например, `Windows` (при этом большинство атрибутов файла теряется), либо передача их по сети, в частности по протоколу `FTP`, который модифицирует атрибуты владельца, а также даты создания и последнего изменения файла. Скриншот одного из вариантов окна программы на этом этапе (выбрана прикладная система «IP-шлюз» – `DGW`) приведен на рисунке 12.

```
Compact Flash (CF) Repair Tool v 1.17 (by VEV)
Wed Apr 6 2011 10:46:27
Тип процессорного модуля: LX                ID диска: 2527A2C7
Серия ядра Linux: 6                        Геометрия /dev/sdb:
Прикладная система: DGW                  Цилиндров: 492
Вариант системы: Classic                  Дорожек: 16
Сетевой boot протокол: Статич. IP        Сект./дор: 32
Имя хоста (Hostname): ttdgw              Всего секторов: 251904
Имя домена: proton-sss.ru                 /dev/sdb1
IP адрес: 192.168.12.222                   Общий объем: 121972 Kib
Маска (под)сети: 255.255.0.0              Свободно: 58732 Kib
IP адрес шлюза для INet: 192.168.147.3
IP адрес сервера DNS: 192.168.147.8,192.168.147.3

Содержимое каталога '/opt/DGW/'
dgw-1_06-6-3.tar.bz2                      dgw-1_07-6-0_rc_10-10-25.tar.bz2
DGW_custom                                dgw-1_04-2-15.tar.bz2

Скрипт: /opt/DGW/DGW_custom выполнен успешно
<Tab>, стрелки вниз или вправо - выбор файла, <Enter> - установка файла,
<F10> - завершение установки прикладной системы и продолжение CF_repair.
Copyright (c) 2011 SSS
```


Рис. 12. Пример окна на этапе установки прикладной системы.

Внимание! При установке прикладной системы для IP-шлюза DGW, после установки файла с одной из версий программы шлюза, необходимо выполнить установку (запуск) файла DGW_custom, который является shell-скриптом, выполняющим дополнительную настройку всей системы под программу шлюза. Если этого не сделать, то программа IP-шлюза на CF будет неработоспособной. Аналогичное действие следует выполнить при установке прикладной системы для ЦАТС «Алмаз 1» (diamond), только в этом случае shell-скрипт называется diamond_custom. При установке прикладной системы для VIP (ЦАТС «Вектор IP») необходимо установить архивы diamond_VIP.tbz2 и dgwvip-1_07-6-0_rc.tbz2, после чего установить файл VIP_custom, который выполнит дополнительную настройку системы для VIP. При установке прикладной системы для DIP (ЦАТС «Diamond IP») необходимо установить архивы diamond_DIP.tar.bz2 и dgwdip-1_08-6-0.tbz2, после чего установить файл DIP_custom, который выполнит дополнительную настройку системы для DIP. При установке прикладной системы для vector2m необходимо установить архив vector2m.tbz2, после чего установить файл v2m_custom, который выполнит дополнительную настройку системы для vector2m.

После установки прикладной системы программа предлагает сменить пароль пользователя **root**. Это рекомендуется делать, если по каким-либо причинам исходный пароль (пароль, имеющийся в системе на **SSS Rescue CD**) этого пользователя забыт или неизвестен. При этом в главном окне выводится сообщение:

Желаете изменить пароль пользователя 'root'? (1-да, 0-нет)

В случае утвердительного ответа (пользователь нажимает <1>, а затем нажимает клавишу <Enter>) выполняется процедура смены пароля, в противном случае пароль пользователя **root** остается таким, каким он был при формировании макета системы для **SSS Rescue CD**. При смене пароля выводится окно, показанное на рисунке 13. Текущее поле ввода выделяется синим фоном, а позицию ввода определяет курсор. Пользователь должен ввести пароль в поле ввода пароля, желательно не менее 6 знаков, после чего нажать клавишу <Enter>. После этого поле ввода переключается на подтверждение пароля, и пользователю следует подтвердить пароль, введя те же символы в поле подтверждения пароля, и нажать клавишу <Enter>. Для завершения смены пароля необходимо нажать клавишу <F10>. Если смена пароля прошла успешно, окно ввода пароля исчезает, и происходит переход к установке временной зоны. В противном случае в нижней строке окна выводится сообщение об ошибке и предлагается эту ошибку исправить. После трех повторов исправлений и неисправленной ошибке программа считает, что смены пароля не было, и завершает эту процедуру. Следует учитывать, что новый пароль вводится и подтверждается в скрытом режиме (вместо введенных символов отображаются звездочки), и его нужно вводить очень внимательно, чтобы не ошибиться.

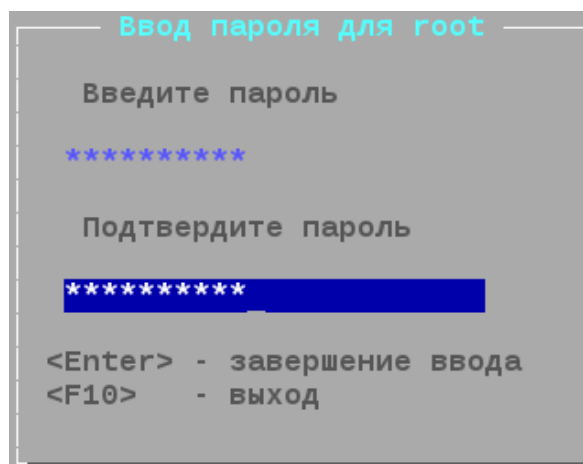


Рис. 13. Пример окна ввода пароля для пользователя **root**.

Затем программа предлагает установить или изменить временную зону (часовой пояс). По умолчанию в системе предустановлена временная зона московского времени (**Europe/Moscow**).

Если система используется в этой временной зоне, то её можно не менять. В противном случае для корректного ведения и отображения времени в системе временную зону рекомендуется изменить. Предложение изменения временной зоны выводится в главном окне программы и выглядит следующим образом:

Желаете изменить временную зону? (1-да, 0-нет)

При утвердительном ответе в центре экрана выводится окно со списком возможных временных зон, и пользователь может выбрать нужную ему зону. Перечень временных зон в создаваемой системе ограничен пространством бывшего СССР. Вид этого окна показан на рисунке 14. В этом окне не выводится никаких подсказок, поэтому назначение клавиш в процессе установки временной зоны приведено здесь. Список поддерживаемых временных зон выводится символами синего цвета на сером фоне. Поскольку весь список не вмещается в окне, то при достижении маркера верхней или нижней границы окна, выполняется скроллинг списка в соответствующем направлении. Текущая выбираемая зона (маркер) выделяется символами белого цвета на синем фоне. Как показано на рисунке – это **Europe/Moscow**. Перемещение маркера по списку выполняется клавишами стрелок и <Home>, <End>, <PgUp> и <PgDn>. Выбор временной зоны выполняется нажатием клавиши <Enter>. При этом выбранная зона выделяется символами белого цвета на сером фоне (на рисунке это **Europe/Kaliningrad**). Завершение установки выбранной временной зоны происходит после нажатия клавиши <F10>, и это окно выбора временной зоны исчезает с экрана монитора.

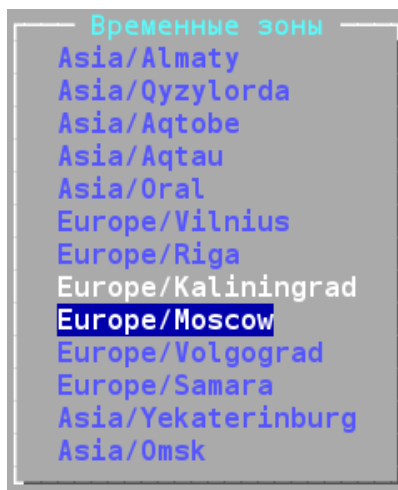


Рис. 14. Пример окна выбора временной зоны.

На финальном этапе программа выдает напоминание о демонтаже файловой системы или файловых систем на CF (см. рис. 3). Если визуальная проверка настроек после работы **CF_repair** не нужна, можно выполнить выход из программы **CF_repair** при помощи клавиши <F10>. В этом случае программа сама демонтирует файловую систему или файловые системы на CF. При нажатии любой другой клавиши для завершения **CF_repair** автоматическое демонтажное действие не выполняется, и перед извлечением CF из Card Reader'a после завершения работы программы **CF_repair** необходимо вручную демонтировать файловую систему на CF следующей командой:

```
umount /mnt/cf1
```

В случае создания CF с **initramfs** команда должна быть следующей:

```
umount /mnt/cf1 /mnt/cf2
```

Если этого не сделать, то файловая система на CF останется в несогласованном состоянии, что при последующем её монтировании может привести к неожиданным результатам (потере некоторых файлов и даже каталогов, перемешиванию содержимого каталогов и т.п.). При завершении **CF_repair** в результате обнаружения ошибки файловая система CF также может

остаться смонтированной к системе. В этих случаях **CF_repair** выдает соответствующее предупреждение перед завершением, и пользователь должен демонтировать файловую систему одной из приведенных выше команд (зависит от варианта устанавливаемой/восстанавливаемой системы). В противном случае **CF_repair** не сможет смонтировать файловую систему на CF при следующем запуске после устранения причин, вызвавших ошибочное завершение.

Убедиться в том, что файловая система CF демонтирована, можно, просмотрев файл **/proc/mounts**, в котором ядро отображает все смонтированные файловые системы. Это выполняется командой:

```
cat /proc/mounts
```

Исполнив такую команду до и после указанной выше команды **umount**, можно увидеть, какие файловые системы смонтированы в каждом случае. В первом случае последняя строка этого файла будет выглядеть примерно так:

```
/dev/sdx1 /mnt/cf1 ext2 rw,nogroups 0 0 .
```

В случае создания CF с **initramfs** таких строк в конце этого файла будет две:

```
/dev/sdx1 /mnt/cf1 ext2 rw,nogroups 0 0  
/dev/sdx2 /mnt/cf2 ext2 rw,nogroups 0 0 ,
```

где **x** – обозначает букву конкретного устройства, соответствующего CF, может принимать значения строчных латинских букв от **a** до **z**.

После выполнения команды **umount** этой строки или строк в файле **/proc/mounts** не будет.

После успешного создания или восстановления CF, как правило, требуется перезапустить или остановить компьютер. Обычно в ОС **Linux** для этого используются команды **reboot** или **halt**, либо **shutdown**, которые корректно завершают работу системы и приводят все файловые системы на внешних носителях в согласованное состояние. В нашем случае этого не требуется, поскольку файловая система на CD не модифицируется, и при последующих запусках она будет находиться в том же самом состоянии, как и во всех предыдущих. По этой причине после демонтажа файловой системы или файловых систем на CF компьютер можно перезапустить кнопкой аппаратного перезапуска – «Reset». Главное в этом случае успеть извлечь CD из привода, чтобы после перезапуска система не начала опять загружаться с CD-ROM. Если CD все-таки не удалось извлечь из привода, то не следует паниковать и снова перезагружать компьютер. Для загрузки системы с жесткого диска следует выбрать пункт меню **Boot from local drive**, и вместо загрузки системы с CD будет загружена система с первого жесткого диска вашего компьютера.

5.2. Работа программы в пакетном режиме

Функционально работа программы **CF_repair** в пакетном режиме полностью аналогична работе в диалоговом режиме за исключением того, что все параметры настройки указываются в файле конфигурационных параметров, имя которого передается программе при её запуске после ключа **-b** (см. пункт 5). Файл конфигурационных параметров является текстовым файлом, содержащим значения всех параметров, необходимых для формирования CF. Каждый параметр представляется в виде записи следующего формата:

```
<имя-параметра>=<значение-параметра>
```

Между именем параметра, знаком равно и значением параметра не должно быть пробелов и/или табуляций. Регистр символов (строчные и прописные буквы) также имеет значение.

Все параметры настройки являются полными аналогами параметров настройки для диалогового режима работы, поэтому содержательное их разъяснение смотрите в пункте 5.1. Здесь приводятся только названия и возможные значения. Имена параметров (выделены жирным шрифтом), которые можно указывать в файле конфигурационных параметров, приведены ниже:

PMod= – соответствует параметру **Тип процессорного модуля**. Можно указывать значения из следующего перечня: **MOPS, MSM, LX** или **atom**;

Kernel= – соответствует параметру **Серия ядра Linux**. Можно указывать из следующих значений: **2, 6** или **6i**;

SYS_Type= – соответствует параметру **Вариант системы**. Можно указывать из значений: **Classic** или **BusyBox**;

Target= – соответствует параметру **Прикладная система**. Можно указывать из следующих значений: **diamond, DGW, DIP, VIP** или **vector2m**;

NetCfgProto= – способ конфигурации локальной сети соответствует параметру **Сетевой boot протокол**. Можно указывать следующие значения: **DHCP** или **StaticIP**;

Host_Name= – соответствует параметру **Имя хоста (Hostname)**. Можно указывать произвольное имя;

Domain_Name= – соответствует параметру **Имя домена**. Следует указать то имя, которое присвоено вашей подсети. Если не указывается, то программа установит значение **proton-sss.ru**;

IPaddr= – соответствует параметру **IP адрес**. Не указывается, если параметр **NetCfgProto=DHCP**. Если не указывается в противном случае, то программа установит значение 192.168.12.99;

NetMask= – соответствует параметру **Маска (под)сети**. Не указывается, если параметр **NetCfgProto=DHCP**. Если не указывается в противном случае, то программа установит значение 255.255.0.0;

GW_IPaddr= – соответствует параметру **IP адрес шлюза для Inet**. Не указывается, если параметр **NetCfgProto=DHCP**. Если не указывается в противном случае, то программа установит значение 192.168.147.3;

DNS_IPaddr= – соответствует параметру **IP адрес сервера DNS**. Не указывается, если параметр **NetCfgProto=DHCP**. Если не указывается в противном случае, то программа установит значение 192.168.147.3,192.168.147.8;

Tgt_Dev= – имя устройства для Compact Flash. Следует указывать **sda, sdb, ...**

LX_Type= – конкретизация типа ПМ для LX. Этот параметр следует указывать, если **PMod=LX**. Если значение указывается с пробелом, то его необходимо заключать в двойные кавычки как показано далее. Если указывается сокращенное название, то двойные кавычки можно опускать. Значения параметра, следующие: "PM-GX (iEi)", "MSM800BEV (DLAG)", "PM-6100 (AEWin)", "ETX-LX (Kontron)", "IEM-LX (iEi)", "EmETX-a5363 (ARBOR)", "ESM-2545 (Avalue)", что соответствует названиям ПМ в таблице 2;

LX_BDev= – устройство загрузки для ПМ типа LX. Этот параметр следует указывать, если **PMod=LX**. Можно указывать значения из следующего списка: **hda, hdb, hdc** или **sda**. При выборе следует руководствоваться таблицей 2;

DGW_fname= – имя файла архива для системы DGW и vector2m. Для VIP и DIP необходимо также указывать параметр **PBX_fname**;

PBX_fname= – имя файла архива для системы Diamond. Для VIP и DIP необходимо также указывать параметр **DGW_fname**;

Cust_fname= – имя файла скрипта XXX_custom, если желаете использовать нестандартный скрипт, в противном случае будет использован скрипт из каталога, соответствующего устанавливаемой прикладной системе;

RootPwd= – пароль пользователя root (произвольный);

TZone= – название временной зоны. Можно указать любую из списка: Asia/Yerevan, Asia/Baku, Asia/Tbilisi, Asia/Bishkek, Asia/Almaty, Asia/Qyzylorda, Asia/Aqtobe, Asia/Aqtau, Asia/Oral, Asia/Yekaterinburg, Asia/Omsk, Asia/Novosibirsk, Asia/Krasnoyarsk, Asia/Irkutsk, Asia/Yakutsk, Asia/Vladivostok, Asia/Sakhalin, Asia/Magadan, Asia/Kamchatka, Asia/Anadyr, Asia/Dushanbe, Asia/Ashgabat, Asia/Samarkand, Asia/Tashkent, Europe/Vilnius, Europe/Tallinn, Europe/Riga, Europe/Minsk, Europe/Kaliningrad, Europe/Kiev, Europe/Uzhgorod, Europe/Zaporozhye, Europe/Simferopol, Europe/Moscow, Europe/Volgograd, Europe/Samara. Если этот параметр не указывается, то по умолчанию устанавливается **Europe/Moscow**.

BadBlk= – выполнять проверку сбойных секторов (Bad Blocks) при создании файловой системы на CF. Можно указывать **0** – не выполнять, либо **1** – выполнять. Более подробно см. пункт 5.1.

Во время разбора и проверки файла конфигурационных параметров программа дополнительно выполняет проверку допустимых значений большинства параметров, и в случае обнаружения ошибки, выдает сообщение об ошибке следующего формата:

<имя-файла>:<номер-строки>: <текстовая-часть-сообщения>

Если обнаружена хотя бы одна серьезная ошибка, то работа программы прекращается после разбора файла конфигурационных параметров. Текстовые части сообщений об ошибках приведены ниже. Собственно тексты сообщений выделены жирным шрифтом.

Синтаксическая ошибка '<строка>' – в указании параметра отсутствует знак '='.

Неизвестный параметр '<строка>' игнорируется – имя параметра указано неверно.

Нет закрывающей двойной кавычки в '<строка>' – в параметре **LX_Type** указана открывающая двойная кавычка, но не указана закрывающая.

Параметр 'LX_BDev' можно указывать только после 'LX_Type' – нарушен порядок указания параметров **LX_Type** и **LX_BDev**. Чтобы проверить корректность значения параметра **LX_BDev** программе нужно знать значение параметра **LX_Type**.

Неверное значение '<строка>' для параметра '<имя-параметра>' – для указанного параметра такого значения не может быть.

Повторное указание параметра '<имя-параметра>' игнорируется – указанный параметр приведен в конфигурационном файле более одного раза. Все указания параметра, кроме первого, игнорируются.

Не указано устройство для CF – в конфигурационном файле отсутствует параметр **Tgt_Dev**.

В системе не обнаружен USB диск /dev/sdX – имя устройства, указанное в параметре **Tgt_Dev** не соответствует действительности.

Для процессорного модуля '<название ПМ>' не указано устройство загрузки – в файле конфигурационных параметров указан параметр **LX_Type**, но не указан параметр **LX_BDev**.

Для прикладной системы '<имя ПС>' не указано имя файла архива – не указан параметр **DGW_fname** или **PBX_fname** в зависимости от <имя ПС>.

Для прикладной системы '<имя ПС>' не указано имя файла архива для параметра '<имя-параметра>' – для ПС **DIP** или **VIP** не указано имя одного из архивов, какого именно определяет <имя-параметра>.

6. Возможные проблемы и рекомендации по их устранению

Как было сказано выше, при обнаружении нештатной ситуации, программа выводит диагностику ошибки в третьей снизу строке экрана, а уточнение причины, вызвавшей нештатную ситуацию, можно увидеть в окне протоколов, если таковой был сохранен. При выдаче любого из приведенных ниже сообщений вместо символов **xxx** будет присутствовать имя реального устройства или раздела устройства, для которого выявлена ошибочная ситуация.

Абсолютно все ситуации возникновения ошибок предусмотреть и объяснить в этом документе невозможно, потому что их может быть очень много. Поэтому здесь приведены только самые существенные и часто встречающиеся ситуации. Наиболее серьезные ошибки выявляются на шаге проверки CF. Как правило, после выявления таких ситуаций программа завершает свою работу. Здесь возможны следующие диагностические сообщения (прочие возможности возникновения вероятных ошибок диагностируются предупреждающими сообщениями и приведены выше по тексту):

/dev/xxx: No such device or address – означает, что либо Card Reader не был подключён к компьютеру во время загрузки системы, либо не были загружены модули требуемых драйверов. Убедиться, что Card Reader подключён к компьютеру, либо перезагрузить компьютер, чтобы вся эта процедура выполнялась автоматически;

/dev/xxx: No medium found – означает, что либо в Card Reader'е нет CF, либо она повреждена настолько серьезно, что не подлежит восстановлению, либо неисправен Card Reader. Если CF нет в Card Reader'е, то необходимо вставить её в Card Reader. Во втором случае может потребоваться другая CF, а эту, скорее всего, придется выбросить. В последнем случае потребуется замена Card Reader'а;

/dev/xxx: Нет сигнатуры MBR. Следует выполнить 'fdisk' – означает, что испорчена информация в нулевом секторе CF (MBR). Необходимо выполнить программу **fdisk** и восстановить таблицу разделов и геометрию CF в MBR;

/dev/xxx: Отсутствует таблица разделов. Следует выполнить 'fdisk' – означает, что все записи в таблице разделов MBR пустые – не содержат смысловой информации (в полях типов разделов находятся нули). Это возможно в следующих случаях: либо при запуске **CF_repair** указано неверное устройство, что может привести к потере данных на другом SCSI-, SATA-, PATA-диске или USB-устройстве массовой памяти, либо CF отформатирована как дискета (смотри пояснение далее по тексту), либо CF вообще не отформатирована. В двух последних случаях необходимо выполнить программу **fdisk** и создать таблицу разделов на CF заново. В этой ситуации потребуется знание реальной геометрии CF (см. пункт 4).

Вывод любого из двух следующих предупреждений не приводит к завершению работы программы, но может в дальнейшем привести к серьезным проблемам с CF.

/dev/xxx: Возможно геометрия CF не соответствует реальной – означает, что либо вместо CF был указан другой SCSI-, SATA- или PATA-диск, либо в таблице разделов имеются несоответствия, либо этой таблицы вообще нет на CF;

/dev/xxx: Не Linux раздел(ы). Возможно неверное устройство – означает, что все записи в таблице разделов в MBR содержат значения отличные от 0x83 в поле идентификатора системы. Это возможно в следующих случаях: либо при запуске **CF_repair** указано неверное устройство, что может привести к потере данных на другом SCSI-, SATA-, PATA-диске или USB-устройстве массовой памяти, либо CF ранее использовалась в операционной системе, отличной от **Linux**. В первом случае следует отказаться от продолжения выполнения программы **CF_repair** и при последующем её запуске указать правильное устройство для CF. Данная версия программы может самостоятельно изменять типы разделов на требуемый (0x83 в шестнадцатеричной нотации), если все другие параметры в таблице разделов указаны корректно. Поэтому во втором случае можно просто принять к сведению эту информацию.

На шаге проверки раздела или разделов CF на предмет их занятости системой могут быть выведены следующие сообщения:

/dev/xxx: используется аппаратно (занят)

/dev/xxx: является устройством swap

/dev/xxx: смонтирован только на чтение

/dev/xxx: является корневой файловой системой

/dev/xxx: уже смонтирован на: ууу где **ууу** – точка монтирования.

Вывод любого из этих сообщений означает то, что указанное устройство, а точнее раздел устройства, уже занят операционной системой под другие нужды. Причиной этого может быть указание неверного устройства при запуске программы, либо тот факт, что раздел или разделы CF не были демонтированы после предыдущего запуска программы **CF_repair**.

На шаге модификации таблицы разделов в MBR могут быть выявлены следующие ошибки:

/dev/xxx: Слишком много разделов в MBR (должно быть <=2) – в таблице разделов устройства, указанного для CF обнаружено более двух разделов. Это может означать, что либо вместо CF был указан другой SCSI- или SATA-диск, либо указанная CF ранее использовалась в других устройствах. В последнем случае следует с помощью программы **fdisk** выполнить “переразбиение” разделов на CF, например, удалить все имеющиеся разделы и создать как минимум один раздел на всю CF. После этого программу **CF_repair** можно запустить повторно.

Все следующие сообщения указывают, что **SSS Rescue CD** плохо воспринимается приводом (устройством чтения компакт дисков), либо сам диск имеет механические повреждения.

/dev/xxx: Не могу вычислить размер раздела

/dev/xxx: Не могу вычислить размер архива

/dev/xxx: Неизвестный тип модификации MBR

Следует отметить еще один важный момент, который может привести к проблемам при создании системы на CF. В ОС **Linux** CF всегда представляется как “жёсткий” (hard) диск IDE или SCSI, в зависимости от того к какому интерфейсу её подключают. Это предопределяет на CF соответствующий формат данных, который включает таблицу разделов, находящуюся в MBR, и, собственно, сами разделы для хранения данных, где информация организована в определенных файловых системах. Однако в ОС **Windows**, по-видимому, имеется возможность форматирования CF как дискету для флоппи-диска. На практике встречались CF, отформатированные таким способом. В этом случае на носителе не содержится MBR, а в первом секторе находится загрузчик и BPB (BIOS Parameters Block). Если для формирования системы используется CF с таким форматом, то возникают определенные проблемы сначала с формированием MBR, а затем с установкой загрузчика. Дело в том, что программа **lilo**, устанавливающая одноименный загрузчик, может отказаться перезаписывать MBR, если там находится, например, NTLDR (загрузчик ОС **Windows**). В такой ситуации следует либо вручную удалить NTLDR, либо выполнять программу **lilo** с ключом **-F**.

Если после ввода основных параметров в строке ошибок появляется одно из следующих сообщений:

setup_params: указанный вариант системы не поддерживается.

setup_params: для DIP/VIP/V2M возможны только проц. модули LX и atom

setup_params: система с ядром 2.2 только для MOPS и MSM

то это означает, что пользователь указал несогласованные параметры **Тип процессорного модуля, Серия ядра Linux и Вариант системы**. При последующем запуске программы следует руководствоваться набором возможных вариантов систем, приведенным в таблице 3.

Во время выбора устройства для CF возможны следующие сообщения об ошибках:

select_target: нет доступа к sysfs – означает, что файловая система **sysfs** не смонтирована в точке монтирования **/sys**. Необходимо убедиться, что при старте системы была выполнена команда **mount none -t sysfs /sys**, и если нет, то выполнить ее вручную. После устранения причины

можно повторить запуск программы.

select_target: в системе не обнаружено USB дисков – означает, что к компьютеру не подключены ни Card Reader'ы, ни другие USB-устройства массовой памяти. После устранения причины можно повторить запуск программы.

Появление сообщения **mount /dev/xxx on /mnt/cf1: invalid argument** на этапе монтирования файловой системы говорит о том, что Card Reader не может корректно записать информацию на эту Compact Flash. Необходимо заменить Card Reader на такой, который будет корректно работать с этой Compact Flash, либо использовать другой тип Compact Flash.

Приложение 1. Листинг сеанса работы с программой fdisk по установке истинной геометрии для Compact Flash.

Примечание. Все команды и данные, которые вводятся пользователем, выделены жирным, подчеркнутым шрифтом. **<Enter>** - означает нажатие клавиши <Enter> (<Ввод>).

victor ~ # **fdisk -c=dos -u=cylinders /dev/sdb**

WARNING: DOS-compatible mode is deprecated. It's strongly recommended to switch off the mode (with command 'c').

WARNING: cylinders as display units are deprecated. Use command 'u' to change units to sectors.

Команда (m для справки): **p**

Диск /dev/sdb: 4017 МБ, 4017807360 байт
255 heads, 63 sectors/track, 488 cylinders
Units = цилиндры of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x6a5f7029

| Устр-во | Загр | Начало | Конец | Блоки | Id | Система |
|-----------|------|--------|-------|----------|----|---------|
| /dev/sdb1 | | 1 | 488 | 3919828+ | 83 | Linux |

Команда (m для справки): **d**

Выбранный раздел 1

Команда (m для справки): **x**

Команды эксперта (m для справки): **h**

Количество головок (1-256, по умолчанию 255): **16**

Команды эксперта (m для справки): **c**

Количество цилиндров (1-1048576, по умолчанию 488): **7785**

Команды эксперта (m для справки): **r**

Команда (m для справки): **n**

Действие команды

е расширенный

р основной раздел (1-4)

p

Номер раздела (1-4): **1**

Первый цилиндр (1-7785, по умолчанию 1): **<Enter>**

Используется значение по умолчанию 1

Last цилиндр, +цилиндры от +size{K,M,G} (1-7785, по умолчанию 7785): **<Enter>**

Используется значение по умолчанию 7785

Команда (m для справки): **a**

Номер раздела (1-4): **1**

Команда (m для справки): **p**

Диск /dev/sdb: 4017 МБ, 4017807360 байт
16 heads, 63 sectors/track, 7785 cylinders
Units = цилиндры of 1008 * 512 = 516096 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x6a5f7029

| Устр-во | Загр | Начало | Конец | Блоки | Id | Система |
|-----------|------|--------|-------|----------|----|---------|
| /dev/sdb1 | * | 1 | 7785 | 3923608+ | 83 | Linux |

Команда (m для справки): **w**

Таблица разделов была изменена!

Вызывается ioctl() для перечитывания таблицы разделов.

Синхронизируются диски.