

Три простых шага для запуска Linux на отладочной плате с процессором Blackfin

Алексей КОМАРОВ,
к. т. н.
Alexey.Komarov@eltech.spb.ru

Можно ли сделать серьезный коммерческий проект на платформе Blackfin, не приобретая никаких средств разработки? Чтобы дать ответ на этот достаточно актуальный сейчас вопрос, давайте попробуем сравнить те возможности, которые предоставляют бесплатные и коммерческие инструменты разработчика.

Коммерческие программные средства включают бесплатную 90-дневную версию VisualDSP++, содержащую исходные коды драйверов всей периферии Blackfin, различные демонстрационные примеры, ядро реального времени VDK и стек TCP/IP. Кроме этого, в разделе www.analog.com/software можно скачать SDK, включающий различные бесплатные и коммерческие коды, функции обработки и анализа изображений, видео и многое другое.

С другой стороны, существует бесплатная и открытая среда Linux, портированная на Blackfin. Речь идет о uClinux — операционной системе для встраиваемых приложений без аппаратных средств управления виртуальной памятью. Проект uClinux содержит всю необходимую цепочку средств разработки для Blackfin: от GNU C-компилятора и boot-rom загрузчика до готовых инструментов работы в среде uClinux.

Если коммерческие средства разработки освещены в документации достаточно подробно (в том числе на русском языке), то бесплатные приводят неподготовленного человека в состояние замешательства. Обилие различной информации о средствах компиляции ядра, о драйверах и приложениях для хост-машины и целевой платформы не дает быстро определить те базовые шаги, которые дадут возможность начать работу uClinux. Мы постарались создать простую инструкцию установки uClinux на отладочную плату Blackfin вообще без использования коммерческих средств разработки и аппаратных отладочных средств.

Итак, рассмотрим ситуацию, когда у нас нет ничего кроме отладочной платы Blackfin: нет ни JTAG-эмулятора, ни VDSP++. В нашем случае это будет одна из самых дешевых плат BF537-STAMP, на которой нет On-Board-Debug-Agent, поэтому подключить ее к VDSP++ нам все равно не удастся.

Шаг 1. Загрузка платы по последовательному порту

Для того чтобы заставить эту плату работать, нам необходимо каким-то образом загрузить в память код и выполнить его. Тут нам на помощь придет способность процессоров Blackfin загружаться по последовательному порту UART. Итак, для запуска uClinux нам достаточно иметь трехжильный COM-портовый кабель. Для этих целей подойдет практически любой RS-232 кабель с разъемами DB-9, в конце концов, можно спаять или каким-то образом соединить 2-й и 3-й контакты UART с соответствующими пинами разъема COM-порта вашего персонального компьютера (рис. 1).

Кроме того, нам понадобится программное обеспечение, поддерживающее необходимый для загрузки Blackfin протокол передачи. На главной странице проекта blackfin.uclinux.org в разделе GNU Toolchain можно найти полную инструментальную цепочку разработки, в том числе работающую под Windows — `blackfin-toolchain-win32-2008R1.5.exe`. Из этого пакета нам пока потребуется только инструмент для загрузки процессора по UART, после установки пакета он будет находиться здесь: `Путь установки\GNU Toolchain\2008R1.5\elf\bin\bfin-elf-loader.exe`.

Наконец, тот код, который нам необходимо выполнить на Blackfin, также можно скачать с главной страницы blackfin.uclinux.org из раздела Das U-Boot. U-Boot — это загрузчик, стартующий перед запуском uClinux. Он необходим для инициализации памяти и загрузки образа ядра uClinux из внешних источников. В соответствии с нашей платформой и методом загрузки выбираем файл `u-boot-bf537-stamp-uart-2008R1.5.ldr`.

Для того чтобы выбрать нужный режим загрузки платы BF537-STAMP, как правило, достаточно бывает повернуть переключатель SW16 на плате в режим 7 (BOOT FROM UART HOST), перезагрузить плату и запустить про-

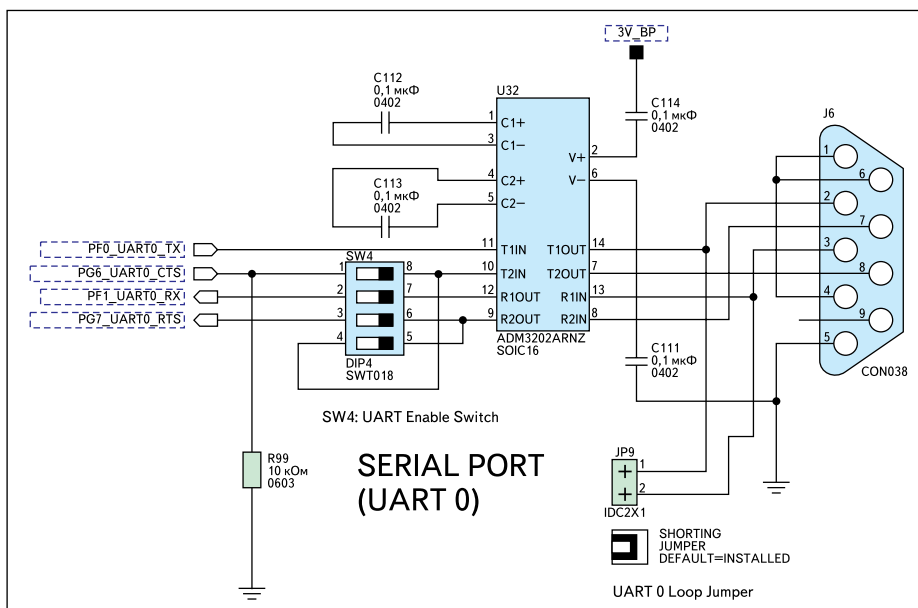


Рис. 1. Схема подключения порта UART на плате BF537-STAMP

цесс передачи файла **bfin-elf-ldr.exe -l u-boot-bf537-stamp-uart-2008R1.5.ldr** через COM-порт персонального компьютера:

```
bfin-elf-ldr.exe -l u-boot-bf537-stamp-uart-2008R1.5.ldr /COM1
Loading LDR D:\u-boot-bf537-stamp-uart-2008R1.5.ldr... OK!
Opening /COM1... OK!
Configuring terminal I/O... OK!
Trying to send autobaud... OK!
Trying to read autobaud... OK!
Checking autobaud... OK!
Autobaud result: 115200bps 49.766mhz (header:0xBF DLL:0x1B
DLH:0x00 fin:0x00)
Sending blocks of DXE 1... [5/8] (62%)...
```

Перед запуском рекомендуем все-таки проверить положение всех джамперов и переключателей на плате, ведь по умолчанию UART замкнут сам на себя (рис. 1), и для его нормальной работы нужно снять джампер JP9. Иногда выводы UART просто отключены от процессора переключателем SW4 (рис. 1).

Более подробную информацию о переключателях платы BF537-STAMP можно найти в [1].

Шаг 2. Настройка платы для загрузки из встроенной флэш-памяти

После того как нами сделан первый шаг и загрузчик успешно стартовал, с ним можно работать с помощью любой удобной терминальной программы. Проще всего воспользоваться стандартной программой **HyperTerminal** — **C:\Program Files\Windows NT\hypertrm.exe**. Настройки параметров порта указаны на рис. 2.

Установив сессию терминального соединения, вы увидите приглашение загрузчика U-Boot, предназначенного для старта ядра uClinux или других приложений. Загрузчик инициализирует SDRAM, умеет обращаться к флэш-памяти, поддерживает передачу файлов по TCP/IP и последовательному порту. Подробное описание команд можно найти в [2].

Для того чтобы заставить нашу плату загружаться непосредственно с флэш-памяти, нужно скопировать туда загрузчик и вернуть переключатель SW16 в положение 1 (BOOT FROM

16-BIT FLASH MEMORY). Предварительно очищаем флэш-память командами:

```
bfin> protect off all
bfin> erase all
```

Версию загрузчика, запускающегося с флэш-памяти, можно скачать с blackfin.uclinux.org, опять же из раздела **Das U-Boot**. Полученный файл **u-boot-bf537-stamp-para-2008R1.5.ldr** загрузим на плату командой **loadb**. А в **HyperTerminal** в меню «Передача» выбираем «Отправить файл» в режиме Kermit (рис. 3).

Если загрузка файла прошла успешно, последует сообщение:

```
## Total Size = 0x00021362 = 136034 Bytes
## Start Addr = 0x01000000
```

Затем просто копируем эти 136 034 байта из ОЗУ во флэш-память:

```
bfin> cp. b 1000000 20000000 21362
Copy to Flash... done
```

Обратите внимание на суффикс команды копирования «. b», он означает работу с байтовыми данными.

Мы сделали второй шаг, в результате которого наша плата стала уже полноценным устройством, способным загружаться без хост-компьютера. Не будем, однако, забывать, что наша конечная цель — Linux, поэтому, опуская несущественные детали, сразу загрузим ядро uClinux по уже установленному соединению.

Шаг 3. Загрузка ядра uClinux

Подходящий образ ядра можно взять из раздела **Linux Kernel** сайта blackfin.uclinux.org. В командной строке загрузчика вводим команду **loadb**, а в **HyperTerminal** в меню «Передача» выбираем: отправить файл **uImage-bf537-stamp** в режиме Kermit.

Процесс передачи файла по последовательному соединению может продлиться довольно долго, поэтому нетерпеливым читателям мы предлагаем попробовать самим настроить TCP/IP-соединение и загрузить образ командой **tftpboot**. Большинство отладочных устройств Blackfin имеют сетевой интерфейс, и все, что вам понадобится, это сетевой кабель с перекрестным обжатием, правильное положение переключателей на плате, разрешающее работу сетевого интерфейса, а также любой TFTP-сервер для вашего персонального компьютера.

После того как файл образа ядра uClinux так или иначе загружен в ОЗУ отладочного устройства, появится сообщение:

```
## Total Size = 0x00379f65 = 3645285 Bytes
## Start Addr = 0x01000000
```

Скопируем 3 645 285 байт, начиная с адреса 0x01000000, в свободную область флэш-памяти. Часть флэш-памяти с адреса 0x20000000 у нас уже занята загрузчиком U-Boot, поэтому в качестве адреса назначения указываем 0x20040000:

```
bfin> cp. b 1000000 20040000 379f65
Copy to Flash... done
```

Проверить содержимое флэш-памяти по адресу 0x20040000 можно командой:

```
bfin> md.b 20040000
20040000: 27 05 19 56 ed .V...EU&a.7%
20040010: 00 00 10 00 00. ....//....
20040020: 75 43 6c 69 6e. uClinux Kernel a
```

Теперь, убедившись, что там находится содержимое файла образа ядра, настроим автототзагрузку ядра uClinux, добавим в переменные окружения загрузчика U-Boot команды для автостарта:

```
bfin> setenv bootcmd bootm 20040000
bfin> saveenv
bfin> printenv bootcmd
bootcmd=bootm 20040000
```



Рис. 2. Параметры последовательного соединения

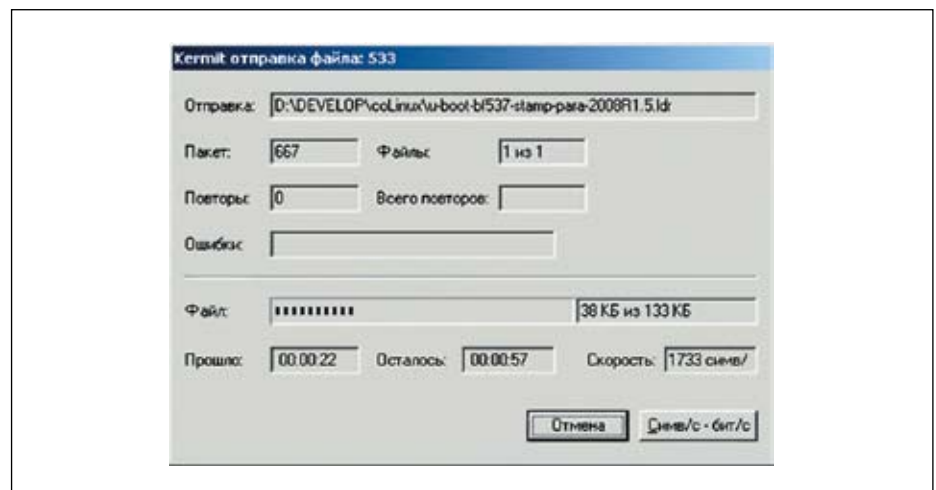


Рис. 3. Передача файла по последовательному соединению

Теперь каждый раз после перезагрузки U-Boot будет автоматически стартовать и uClinux — посредством команды bootm 2004000 (рис. 4).

Итак, мы успешно завершили все три шага установки uClinux. Обращаем ваше внимание на то, что мы не использовали ни коммерческие средства отладки с JTAG, ни VisualDSP++, а процесс установки на другие отладочные комплекты Blackfin ничем не отличается от данной инструкции. Такой упрощенный вариант может быть использован с целью ознакомления с операционной системой и ее базовыми возможностями. При этом время установки uClinux сопоставимо со временем развертывания VisualDSP++.

В следующих статьях мы постараемся более детально осветить вопросы конфигурирования ядра под ту или иную задачу, а также коснемся вопросов разработки собственных приложений под uClinux. ■

Литература

1. BF537 STAMP Board Manual — docs.blackfin.uclinux.org/doku.php?id=hw:boards:bf537-stamp
2. U-Boot Documentation — www.denx.de/wiki/U-Boot/Documentation

```

TCP cubic registered
NET: Registered protocol family 1
NET: Registered protocol family 17
rtc-bfin rtc-bfin: setting the system clock to 1970-01-01 02:43:06 (9786)
Freeing unused kernel memory: 6588k freed
dma_alloc_init: dma_page @ 0x007be000 - 256 pages at 0x03f00000

Welcome to the uClinux distribution
Linux (TM)
For embedded processors including
the Analog Devices Blackfin

For further information, check out:
- http://blackfin.uclinux.org/
- http://docs.blackfin.uclinux.org/
- http://www.uclinux.org/
- http://www.analog.com/blackfin

Have a lot of fun...

BusyBox v1.4.1 (2008-08-11 18:46:32 CST) Built-in shell (msh)
Enter 'help' for a list of built-in commands.

root:/>

```

Рис. 4. Окно терминала и командная строка uClinux