

**ПАКЕТ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ РАСШИРЕННОЙ
ПОДДЕРЖКИ СБИС 1888TX018. АДАПТИРОВАННОЕ ЯДРО
ОПЕРАЦИОННОЙ СИСТЕМЫ LINUX С ПОДДЕРЖКОЙ РАСШИРЕННОГО
НАБОРА УСТРОЙСТВ, ПРИСУТСТВУЮЩИХ В КРИСТАЛЛЕ СБИС
1888TX018, ДЛЯ ПЛАТЫ MB115.01**

Инструкция по применению исходных кодов

ЮФКВ.30169-02 93 01

Листов 13

Инд. № подл.	Подп. и дата	Взам. инв. №	Инд. № дубл.	Подп. и дата

АННОТАЦИЯ

Настоящая инструкция распространяется на адаптированное ядро операционной системы Linux с поддержкой расширенного набора устройств, присутствующих в кристалле СБИС 1888ТХ018 для платы МВ115.01 (далее – Ядро Linux, программа).

Документ содержит описание исходных кодов программы, а также инструкцию по сборке базового программного обеспечения для платы МВ115.01 на СБИС 1888ТХ018.

СОДЕРЖАНИЕ

1. Назначение программы.....	4
1.1 Общие сведения.....	4
2. Назначение и состав комплекта файлов	6
2.1 Назначение комплекта файлов.....	6
2.2 Состав комплекта файлов.....	6
3. Применение комплекта файлов	7
3.1 Компиляция исходных кодов.....	7
Перечень принятых сокращений	12

1. НАЗНАЧЕНИЕ ПРОГРАММЫ

1.1 Общие сведения

1.1.1. Адаптированное ядро операционной системы Linux с поддержкой расширенного набора устройств, присутствующих в кристалле СБИС 1888TX018, для платы MB115.01 входит в состав пакета программного обеспечения (далее по тексту ППО) расширенной поддержки СБИС 1888TX018, разработки НТЦ «Модуль», предназначенного для выполнения на аппаратных ресурсах платы MB115.01 на кристалле СБИС 1888TX018.

1.1.2. Адаптированное ядро Linux с поддержкой базового набора устройств обеспечивает следующие основные возможности:

– Работа с двумя ядрами PowerPC в SMP режиме;

– Поддержка драйверов:

- UART;
- PCI Express;
- VDU;
- MAUDIO;
- Подсистема MTD;
- Watch Dog timer;
- Ethernet;
- SDIO;
- GPIO;
- I2C;
- SPI;
- PINMux;

- USB (host/device);
- Cpufreq;
- Temp_sensor.

1.1.3. Используемая версия исходных кодов ядра Linux из официальной ветки – 5.5.

2. НАЗНАЧЕНИЕ И СОСТАВ КОМПЛЕКТА ФАЙЛОВ

2.1 Назначение комплекта файлов

2.1.1. Комплект файлов исходных кодов операционной системы Linux для платы MB115.01 на кристалле СБИС 1888TX018 предназначен для сборки бинарных образов ядра операционной системы и для дальнейшей разработки программного обеспечения для данного ядра. Его полное обозначение и наименование: ЮФКВ.30169-02 «Адаптированное ядро операционной системы Linux с поддержкой расширенного набора устройств, присутствующих в кристалле СБИС 1888TX018, для платы MB115.01».

2.2 Состав комплекта файлов

2.2.1. В качестве артефактов поставки исходных кодов ядра Linux предоставляются:

а) kernel-sources.tgz – архив исходных кодов ядра Linux версии 5.5 адаптированного для исполнения в кристалле СБИС 1888BM18 для платы MT150.02.

б) rootfs-sources.tgz – исходные коды системы сборки образа файловой системы Linux.

в) tx018-kernel-image.tgz – архив собранных бинарных образов ядра Linux с модулями

г) tx018-rootfs.tgz – архив собранной основной файловой системы Linux.

3. ПРИМЕНЕНИЕ КОМПЛЕКТА ФАЙЛОВ

3.1 Компиляция исходных кодов

3.1.1. Компиляция исходных кодов производится на персональном компьютере, функционирующем под управлением операционной системы Linux (Debian GNU/Linux 10 (buster)) со всеми установленными пакетами из дистрибутива.

3.1.2. Для компиляции исходных кодов необходимо выполнить следующие действия:

3.1.2.1. Установить комплект компилятора для платформы `powerpc-rcm-linux-gnu-` в доступное для исполнения из переменной окружения `PATH` директорию или использовать ранее установленный;

3.1.2.2. Создать в домашней директории пользователя каталог `src` или использовать существующий с таким же именем:

```
mkdir ~/src
```

3.1.2.3. Запустить терминал ОС Linux;

3.1.2.4. В командной строке терминала для распаковки архива выполнить следующие команды:

```
cd ~/src
```

```
tar xzf ~/kernel-sources.tgz
```

3.1.2.1. Выполнить компиляцию ядра Linux, для чего необходимо выполнить следующие команды:

```
cd ~/src
```

```
sudo apt-get install bc u-boot-tools
```

```
CROSS_COMPILE=powerpc-rcm-linux-gnu-      make      ARCH=powerpc  
44x/1888tx018_defconfig
```

```
CROSS_COMPILE=powerpc-rcm-linux-gnu-      make      ARCH=powerpc
INSTALL_MOD_PATH=./.modinst      uImage      mb115-01.dtb      modules
modules_install
```

3.1.3. После выполнения указанных команд будут созданы необходимые целевые бинарные файлы `arch/powerpc/boot/uImage` – образ ядра Linux и `arch/powerpc/boot/dts/mb115-01.dtb` – образ дерева оборудования платы MB115.01. Также в каталоге `.modinst` будет создано дерево бинарных файлов динамически загружаемых модулей ядра Linux.

3.2 Сборка корневой файловой системы Linux

3.2.1. Создать рабочий каталог и перейти в него (можно воспользоваться существующим каталогом):

```
mkdir ~/rootfs
cd ~/rootfs
```

3.2.2. Распаковать архив с исходными кодами системы сборки:

```
tar xzf rootfs-sources.tgz
```

3.2.3. Если в системе не установлен Docker, то установить его, следуя инструкции <https://docs.docker.com/engine/install/>;

3.2.4. Выполнить сборку образа docker, если образ до этого не собирался:

```
sudo docker build --tag builder docker
```

3.2.5. Выполнить сборку, запустив командный файл и указав платформу дистрибутива:

```
./build.sh tx018
```

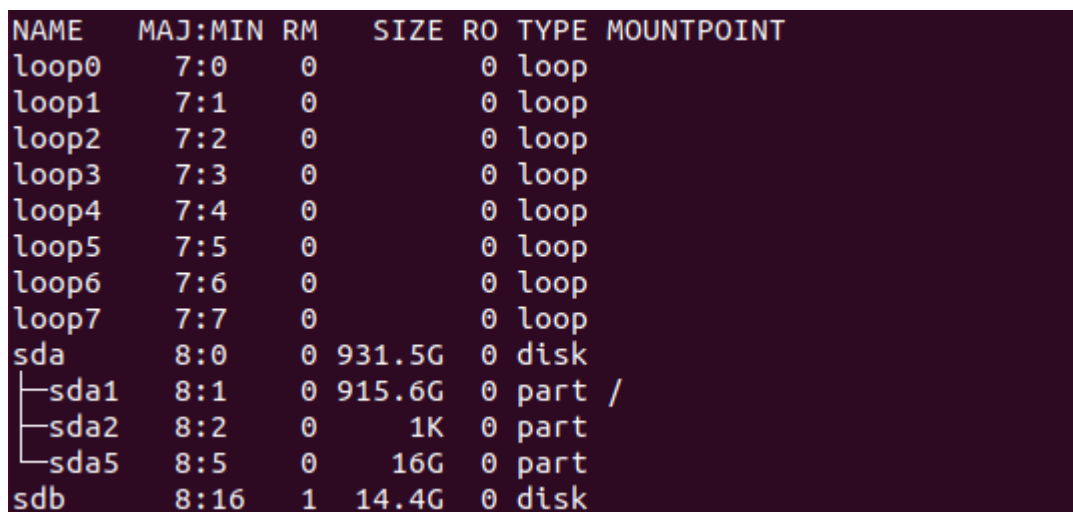
3.2.6. По окончании сборки в каталоге `tx018` будет находиться файл `rootfs.tgz` с основной файловой системой Linux.

3.3 Запись дистрибутива Linux на SD-карту

3.3.1. Для записи дистрибутива Linux на SD-карту необходимо выполнить следующие шаги:

3.3.1.1. Подготовить SD-карту с установленным загрузчиком U-Boot согласно документу «Пакет программного обеспечения расширенной поддержки СБИС 1888TX018. Адаптированный загрузчик U-Boot для платы MB115.01 на кристалле СБИС 1888TX018. Инструкция по применению исходных кодов».

3.3.1.2. Вставить подготовленную SD-карту в устройство чтения-записи. Для того чтобы узнать, как SD-карта определилась в системе, можно выполнить команду `lsblk -a`.



NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
loop0	7:0	0		0	loop	
loop1	7:1	0		0	loop	
loop2	7:2	0		0	loop	
loop3	7:3	0		0	loop	
loop4	7:4	0		0	loop	
loop5	7:5	0		0	loop	
loop6	7:6	0		0	loop	
loop7	7:7	0		0	loop	
sda	8:0	0	931.5G	0	disk	
├─sda1	8:1	0	915.6G	0	part	/
├─sda2	8:2	0	1K	0	part	
└─sda5	8:5	0	16G	0	part	
sdb	8:16	1	14.4G	0	disk	

Рисунок 1 – Внешний вид утилиты lsblk

3.3.1.3. В нашем случае это sdb, далее необходимо заменять sdb на полученное название.

3.3.1.4. Подготовить специальный раздел на SD-карте. Например, при помощи командной строки Linux:

```
sudo fdisk /dev/sdb
```

```
Welcome to fdisk (util-linux 2.31.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x8210c451.

Command (m for help): p
Disk /dev/sdb: 14.4 GiB, 15489564672 bytes, 30253056 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x8210c451

Command (m for help):
```

Рисунок 2 – Вид окна терминала после команды `sudo fdisk /dev/sdb`

3.3.1.5. Необходимо выполнить команду `n` и создать новый раздел:

```
Command (m for help): n
Partition type
  p   primary (0 primary, 0 extended, 4 free)
  e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 2
First sector (2048-30253055, default 2048):
Last sector, +sectors or +size{K,M,G,T,P} (2048-30253055, default 30253055): +4G

Created a new partition 2 of type 'Linux' and of size 4 GiB.

Command (m for help):
```

Рисунок 3 – Вид окна терминала при успешном создании раздела

3.3.1.6. Для записи сделанных изменений необходимо подать команду `w`.

```
Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.
```

Рисунок 4 – Сохранение сделанных изменений в разделах диска

3.3.2. Создать файловую систему на созданном разделе (необходимо заменить `/dev/sdb2` на реальное устройство отвечающее за созданный раздел):

```
sudo mkfs.ext4 /dev/sdb2
```

3.3.3. Смонтировать созданный раздел с файловую систему (необходимо заменить `/dev/sdb2` на реальное устройство, отвечающее за созданный раздел):

```
sudo mount /dev/sdb2 /mnt/sdcard
```

3.3.4. Распаковать архив файловой системы `rootfs.tgz` в смонтированный каталог:

```
cd ~/rootfs/tx018
```

```
sudo tar xzf rootfs.tgz -C /mnt/sdcard
```

3.3.5. Скопировать файлы образа ядра Linux и device tree в смонтированный каталог:

```
cd ~/src
```

```
sudo cp uImage /mnt/sdcard/boot/uImage-tx018.bin
```

```
sudo cp mb115-01.dtb /mnt/sdcard/boot/uImage-tx018.dtb
```

3.3.6. Скопировать модули ядра Linux в смонтированный каталог:

```
sudo cp -r .modinst/* /mnt/sdcard
```

3.3.7. Размонтировать созданный раздел:

```
sudo umount /mnt/sdcard
```

3.3.8. Освободить устройство `/dev/sdb`:

```
sudo eject /dev/sdb
```

ПЕРЕЧЕНЬ ПРИНЯТЫХ СОКРАЩЕНИЙ

EXT	– Extended File System (Расширенная файловая система)
FAT	– File Allocation Table (Таблица размещения файлов)
GPIO	– General-Purpose Input/Output (Интерфейс ввода/вывода общего назначения)
PCI Express	– Peripheral Component Interconnect Express (Компьютерная шина, использующая программную модель шины PCI и высокопроизводительный физический протокол, основанный на последовательной передаче данных)
SD	– Secure Digital (Формат карт памяти для использования в портативных устройствах)
SDIO	– Secure Digital Input Output (Стандарт, поддержка которого позволяет использовать со слотом расширения формата SD/MMC соответствующую периферию)
SMP	– Symmetric Multiprocessing (Симметричная мультипроцессорность)
SPI	– Serial Peripheral Interface (Последовательный периферийный интерфейс)
UART	– Universal Asynchronous Receiver-Transmitter (Универсальный асинхронный приёмопередатчик)
USB	– Universal Serial Bus (Универсальная последовательная шина)
VDU	– Visual Display Unit (Устройство визуального отображения)
НТЦ	– научно-технический центр
ОС	– операционная система
ППО	– пакет программного обеспечения

[illegible]