

Корневая файловая система

Процесс init:

1. Написал код с выводом **Hello ARM Kernel** и входом в бесконечный цикл while.

```
#include <stdio.h>

Windsurf: Refactor | Explain | Generate Function Comment | X
int main(void){
    printf("-----Hello ARM Kernel-----\n");
    while(1);
    return 0;
}
```

2. Компиляция кода под arm. Использовал флаг **-static** для того, чтобы нужные библиотеки уже были в исполняемом файле.

```
timofei@Acer-Aspire-E5-575G:~/Desktop$ arm-linux-gnueabi-gcc -static init.c -o init

timofei@Acer-Aspire-E5-575G:~/Desktop$ file ./init
./init: ELF 32-bit LSB executable, ARM, EABI5 version 1 (GNU/Linux), statically linked, BuildID[sha1]=e566fbed957a1e1f51b0ee7f2c0f45ad141477bc, for GNU/Linux 3.2.0, not stripped
```

3. Создаю initramfs образ. Передаем содержимое ./init утилите cpio для создания архива, и при помощи утилиты gzip сжимаем в файл initramfs.cpio.gz

```
timofei@Acer-Aspire-E5-575G:~/Desktop$ echo ./init | cpio -o -H newc | gzip > initramfs.cpio.gz
951 блок
```

4. Запуск

```
timofei@Acer-Aspire-E5-575G:~/Coding/test_kernels/arm$ QEMU_AUDIO_DRV=node qemu-system-arm -M vexpress-a9 -kernel zImage -initrd initramfs.cpio.gz -dtb vexpress-v2p-ca9.dtb -nographic -append "console=ttyAMA0"
```

```
-----Hello ARM Kernel-----
[ 15.203525] amba 1000f000.wdt: deferred probe pending: (reason unknown)
[ 15.203686] amba 100e0000.memory-controller: deferred probe pending: (reason unknown)
[ 15.203707] amba 100e1000.memory-controller: deferred probe pending: (reason unknown)
[ 15.203721] amba 100e5000.watchdog: deferred probe pending: (reason unknown)
```

Корневая файловая система с busybox:

1. Собираем defconfig busybox для arm архитектуры

```
timofei@Acer-Aspire-E5-575G:~/Coding/busybox/busybox-1.36.1$ ARCH=arm make defconfig
HOSTCC scripts/basic/fixdep
HOSTCC scripts/basic/split-include
HOSTCC scripts/basic/docproc

Circular Buffer support (FEATURE_IPC_SYSLOG) [Y/n/?] (NEW) y
Circular buffer size in Kbytes (minimum 4KB) (FEATURE_IPC_SYSLOG_BUFFER_SIZE) [16] (NEW) 16
Linux kernel printk buffer support (FEATURE_KMSG_SYSLOG) [Y/n/?] (NEW) y
timofei@Acer-Aspire-E5-575G:~/Coding/busybox/busybox-1.36.1$
```

2. Проверяем все флаги и выставляем свои через menuconfig

2.1. Выставил флаг для статического компилирования и прописал префикс для кросс компиляции

```
Build static binary (no shared libs)
Force NOMMU build
[arm-linux-gnueabihf-) Cross compiler prefix
() Path to svroot
```

2.2. При make install все бинарники для корневой файловой системы под arm будут собираться в директорию ./_install предотвращая затирания текущих бинарников моей системы

```
(./_install) Destination path for 'make install'
```

```
timofei@Acer-Aspire-E5-575G:~/Coding/busybox/busybox-1.36.1$ ARCH=arm make menuconfig
HOSTLD scripts/kconfig/mconf
HOSTCC scripts/kconfig/lxdialog/checklist.o
HOSTCC scripts/kconfig/lxdialog/inputbox.o
HOSTCC scripts/kconfig/lxdialog/lxdialog.o
HOSTCC scripts/kconfig/lxdialog/menubox.o
HOSTCC scripts/kconfig/lxdialog/msgbox.o
HOSTCC scripts/kconfig/lxdialog/textbox.o
HOSTCC scripts/kconfig/lxdialog/util.o
HOSTCC scripts/kconfig/lxdialog/yesno.o
HOSTLD scripts/kconfig/lxdialog/lxdialog
scripts/kconfig/mconf Config.in
#
# using defaults found in .config
#

*** End of configuration.
*** Execute 'make' to build the project or try 'make help'.
```

3. Сборка

```
Static linking against glibc, can't use --gc-sections
Trying libraries: m resolv rt
  Library m is needed, can't exclude it (yet)
  Library resolv is needed, can't exclude it (yet)
  Library rt is not needed, excluding it
  Library m is needed, can't exclude it (yet)
  Library resolv is needed, can't exclude it (yet)
Final link with: m resolv
```

Проверяем, что собралось статично под ARM.

```
timofei@Acer-Aspire-E5-575G:~/Coding/busybox/busybox-1.36.1$ file ./busybox
./busybox: ELF 32-bit LSB executable, ARM, EABI5 version 1 (GNU/Linux), statically
ly linked, BuildID[sha1]=aa0564d5e96575e7c1e5e52de4ad2304a9748e45, for GNU/Linux
3.2.0, stripped
```

4. make install

```
-----
You will probably need to make your busybox binary
setuid root to ensure all configured applets will
work properly.
-----
```

5. Упаковка содержимого _install

```
timofei@Acer-Aspire-E5-575G:~/Coding/busybox/busybox-1.36.1/_install$ find . | cpio -o -H newc | gzip > in
itramfs.cpio.gz
3197 блоков
```

6. Запуск

```
timofei@Acer-Aspire-E5-575G:~/Coding/test_kernels/arm$ QEMU_AUDIO_DRV=node qemu-system
-arm -M vexpress-a9 -kernel zImage -initrd initramfs.cpio.gz -dtb vexpress-v2p-ca9.dtb
-nographic -append "console=ttyAMA0 rdinit=/bin/ash"
```

```
~ # whoami
whoami: unknown uid 0
~ # ls
bin          initramfs.cpio.gz  root          usr
dev          linuxrc            sbin
```