Задание 17. Кроскомпиляция ядра Linux с дальнейшей ошибки файловой системы

1. Собрал defconfig для архитектуры arm а именно multi_v7

```
timofei@Acer-Aspire-E5-575G:~/Coding/linux$ ARCH=arm make defconfig
  HOSTCC  scripts/basic/fixdep
  HOSTCC  scripts/kconfig/conf.o
  HOSTCC  scripts/kconfig/confdata.o
  HOSTCC  scripts/kconfig/expr.o
  LEX     scripts/kconfig/lexer.lex.c
  YACC    scripts/kconfig/parser.tab.[ch]
  HOSTCC  scripts/kconfig/lexer.lex.o
  HOSTCC  scripts/kconfig/menu.o
  HOSTCC  scripts/kconfig/parser.tab.o
  HOSTCC  scripts/kconfig/preprocess.o
  HOSTCC  scripts/kconfig/symbol.o
  HOSTCC  scripts/kconfig/util.o
  HOSTLD  scripts/kconfig/conf
*** Default configuration is based on 'multi_v7_defconfig'
#
# configuration written to .config
#
```

2. Сборка ядра из .config

```
timofei@Acer-Aspire-E5-575G:~/Coding/linux$ ARCH=arm CROSS_COMPILE=arm-linux-gnueabihf- make zImage
  SYNC    include/config/auto.conf
  SYSHDR  arch/arm/include/generated/uapi/asm/unistd-oabi.h
  SYSHDR  arch/arm/include/generated/uapi/asm/unistd-eabi.h
```

```
  GZIP    arch/arm/boot/compressed/piggy_data
  AS      arch/arm/boot/compressed/piggy.o
  CC      arch/arm/boot/compressed/misc.o
  CC      arch/arm/boot/compressed/decompress.o
  CC      arch/arm/boot/compressed/string.o
  AS      arch/arm/boot/compressed/hyp-stub.o
  CC      arch/arm/boot/compressed/fdt_rw.o
  CC      arch/arm/boot/compressed/fdt_ro.o
  CC      arch/arm/boot/compressed/fdt_wip.o
  CC      arch/arm/boot/compressed/fdt.o
  CC      arch/arm/boot/compressed/atags_to_fdt.o
  CC      arch/arm/boot/compressed/fdt_check_mem_start.o
  AS      arch/arm/boot/compressed/lib1funcs.o
  AS      arch/arm/boot/compressed/ashldi3.o
  AS      arch/arm/boot/compressed/bswapsdi2.o
  LD      arch/arm/boot/compressed/vmlinux
  OBJCOPY arch/arm/boot/zImage
  Kernel: arch/arm/boot/zImage is ready
```

3. Создал отдельную директорию и проверил что собралось верно

```
timofei@Acer-Aspire-E5-575G:~$ cd Coding/test_kernel/arm/
timofei@Acer-Aspire-E5-575G:~/Coding/test_kernel/arm$ file zImage
zImage: Linux kernel ARM boot executable zImage (little-endian)
```

4. Собираем dtbs

```
timofei@Acer-Aspire-E5-575G:~/Coding/linux$ ARCH=arm make dtbs
```

5. Загрузка

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

Выключили звук, в качестве платы выбрали vexpress-a9, передали ядро zImage, передали dtb файл, отключили графику и сказали, чтобы использовал терминал.

6. Результат

```
[    3.789784] Kernel panic - not syncing: VFS: Unable to mount root fs on unknown-block(0,0)
[    3.801417] CPU: 0 UID: 0 PID: 1 Comm: swapper/0 Not tainted 6.15.0test-g119b1e61a769-dirty #1 NONE
[    3.802229] Hardware name: ARM-Versatile Express
[    3.802845] Call trace:
[    3.803688]  unwind_backtrace from show_stack+0x10/0x14
[    3.805195]  show_stack from dump_stack_lvl+0x54/0x68
[    3.805579]  dump_stack_lvl from panic+0x10c/0x36c
[    3.806119]  panic from mount_root_generic+0x1f8/0x2a8
[    3.806676]  mount_root_generic from prepare_namespace+0x1f8/0x250
[    3.807133]  prepare_namespace from kernel_init+0x1c/0x12c
[    3.807561]  kernel_init from ret_from_fork+0x14/0x28
[    3.808064] Exception stack(0xc8825fb0 to 0xc8825ff8)
[    3.808740] 5fa0:                                     00000000 00000000 00000000 00000000
[    3.809329] 5fc0: 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
[    3.809889] 5fe0: 00000000 00000000 00000000 00000000 00000013 00000000
[    3.811718] ---[ end Kernel panic - not syncing: VFS: Unable to mount root fs on unknown-block(0,0) ]---
```

Ошибка: отсутствие корневой файловой системы. Это появилось, так как мы запустили голое ядро в эмуляторе без корневой файловой системы.