Для минимальной файловой системы создаем файл со следующим содержанием:

```
  GNU nano 7.2                                                        init.c
#include <stdio.h>
#include <unistd.h>

int main(void){

    printf("\n----------------HELLO----------------\n");
    sleep(20);
    return 0;
}
```

Компилируем его с помощью кросс-компилятора arm-linux-gnueabihf-gcc:

```
nastya@linux:~/arm-compile$ arm-linux-gnueabihf-gcc -static init.c -o init
nastya@linux:~/arm-compile$
```

Формируем файловую систему:

```
nastya@linux:~/arm-compile/FS$ echo init | cpio -o -H newc | gzip > initramfs.cpio.gz
829 blocks
nastya@linux:~/arm-compile/FS$
```

Запускаем систему в эмуляторе QEMU:

```
nastya@linux:~/arm-compile$ QEMU_AUDIO_DRV=none qemu-system-arm -M vexpress-a9 -kernel zImage -dtb vexpress-v2p-ca9.dtb
-initrd initramfs.cpio.gz -append "console=ttyAMA0" -nographic
[    0.000000] Booting Linux on physical CPU 0x0
[    0.000000] Linux version 6.14.0my-g38fec10eb60d-dirty (nastya@linux) (arm-linux-gnueabihf-gcc (Ubuntu 10.5.0-4ubuntu
2) 10.5.0, GNU ld (GNU Binutils for Ubuntu) 2.42) #1 SMP Sun Sep  7 19:55:53 UTC 2025
[    0.000000] CPU: ARMv7 Processor [410fc090] revision 0 (ARMv7), cr=10c5387d
[    0.000000] CPU: PIPT / VIPT nonaliasing data cache, VIPT nonaliasing instruction cache
[    0.000000] OF: fdt: Machine model: V2P-CA9
[    0.000000] Memory policy: Data cache writeback
[    0.000000] efi: UEFI not found.
[    0.000000] cma: Failed to reserve 64 MiB on node -1
[    0.000000] Zone ranges:
[    0.000000]   DMA      [mem 0x0000000060000000-0x0000000067ffffff]
[    0.000000]   Normal   empty
[    0.000000]   HighMem  empty
[    0.000000] Movable zone start for each node
```

Видим вывод строки и ошибку, возникшую из-за завершения процесса init:

```
----------------HELLO----------------
[   12.733612] amba 1000f000.wdt: deferred probe pending: (reason unknown)
[   12.733861] amba 100e0000.memory-controller: deferred probe pending: (reason unknown)
[   12.734021] amba 100e1000.memory-controller: deferred probe pending: (reason unknown)
[   12.734174] amba 100e5000.watchdog: deferred probe pending: (reason unknown)
[   22.487324] Kernel panic - not syncing: Attempted to kill init! exitcode=0x00000000
[   22.488426] CPU: 0 UID: 0 PID: 1 Comm: init Not tainted 6.14.0my-g38fec10eb60d-dirty #1
[   22.489159] Hardware name: ARM-Versatile Express
[   22.489760] Call trace:
[   22.490335]  unwind_backtrace from show_stack+0x10/0x14
[   22.491839]  show_stack from dump_stack_lvl+0x50/0x64
[   22.492060]  dump_stack_lvl from panic+0x10c/0x384
[   22.492274]  panic from do_exit+0x980/0x984
[   22.492402]  do_exit from do_group_exit+0x34/0x84
[   22.492528]  do_group_exit from pid_child_should_wake+0x0/0x60
[   22.493728] ---[ end Kernel panic - not syncing: Attempted to kill init! exitcode=0x00000000 ]---
```
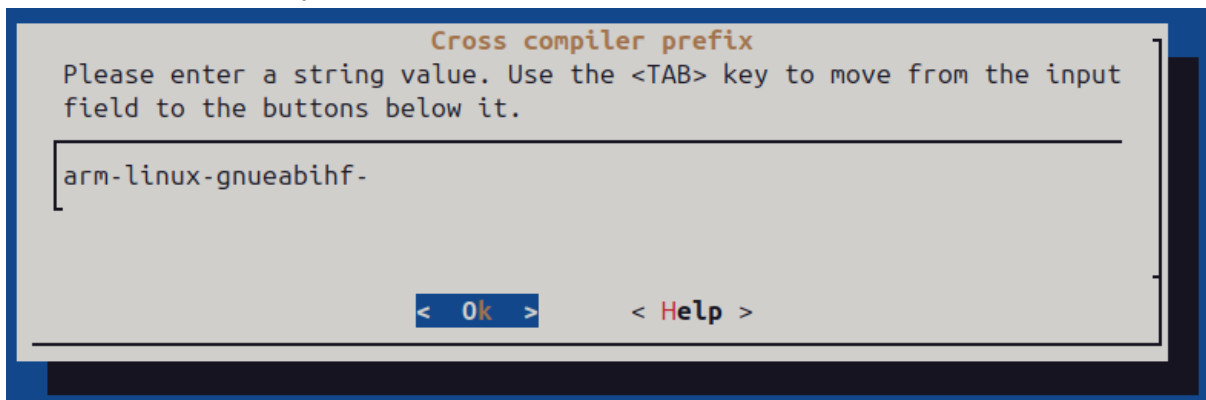
Для файловой системы на базе busybox выполняем команду make defconfig:

```
nastya@linux:~/kernel/busybox/busybox-1.36.1$ ARCH=arm make defconfig
  HOSTCC   scripts/basic/fixdep
  HOSTCC   scripts/basic/split-include
  HOSTCC   scripts/basic/docproc
  GEN      include/applets.h
  GEN      include/usage.h
  GEN      modutils/Kbuild
  GEN      modutils/Config.in
  GEN      findutils/Kbuild
  GEN      findutils/Config.in
  GEN      editors/Kbuild
  GEN      editors/Config.in
```

Сборку будем делать без динамических библиотек:

```
--- Build Options
[*] Build static binary (no shared libs)
[ ] Force NOMMU build
```

Указываем префикс нужного кросс-компилятора:

```
                    Cross compiler prefix
   Please enter a string value. Use the <TAB> key to move from the input
   field to the buttons below it.

   arm-linux-gnueabihf-



                    <  Ok  >          < Help >
```

Выполняем make:

```
nastya@linux:~/kernel/busybox/busybox-1.36.1$ make -j 4
scripts/kconfig/conf -s Config.in
#
# using defaults found in .config
#
  SPLIT    include/autoconf.h -> include/config/*
  GEN      include/bbconfigopts.h
  GEN      include/common_bufsiz.h
  GEN      include/embedded_scripts.h
  HOSTCC   applets/usage
  HOSTCC   applets/applet_tables
```

Выполняем make install:

```
nastya@linux:~/kernel/busybox/busybox-1.36.1$ make install
  ./_install//bin/arch -> busybox
  ./_install//bin/ash -> busybox
  ./_install//bin/base32 -> busybox
  ./_install//bin/base64 -> busybox
  ./_install//bin/cat -> busybox
  ./_install//bin/chattr -> busybox
  ./_install//bin/chgrp -> busybox
  ./_install//bin/chmod -> busybox
  ./_install//bin/chown -> busybox
  ./_install//bin/conspy -> busybox
  ./_install//bin/cp -> busybox
```

Формируем файловую систему из полученных файлов:

```
nastya@linux:~/kernel/busybox/busybox-1.36.1/_install$ find . | cpio -o -H newc | gzip > initramfs.cpio.gz
cpio: File ./initramfs.cpio.gz grew, 786432 new bytes not copied
5157 blocks
nastya@linux:~/kernel/busybox/busybox-1.36.1/_install$
```

Запускаем нашу систему в эмуляторе QEMU:

```
nastya@linux:~/arm-compile$ QEMU_AUDIO_DRV=none qemu-system-arm -M vexpress-a9 -kernel zImage -dtb vexpress-v2p-ca9.dtb
-initrd initramfs.cpio.gz -append "console=ttyAMA0 rdinit=/bin/ash" -nographic
[    0.000000] Booting Linux on physical CPU 0x0
[    0.000000] Linux version 6.14.0my-g38fec10eb60d-dirty (nastya@linux) (arm-linux-gnueabihf-gcc (Ubuntu 10.5.0-4ubuntu
2) 10.5.0, GNU ld (GNU Binutils for Ubuntu) 2.42) #1 SMP Sun Sep  7 19:55:53 UTC 2025
[    0.000000] CPU: ARMv7 Processor [410fc090] revision 0 (ARMv7), cr=10c5387d
[    0.000000] CPU: PIPT / VIPT nonaliasing data cache, VIPT nonaliasing instruction cache
[    0.000000] OF: fdt: Machine model: V2P-CA9
[    0.000000] Memory policy: Data cache writeback
[    0.000000] efi: UEFI not found.
[    0.000000] cma: Failed to reserve 64 MiB on node -1
[    0.000000] Zone ranges:
[    0.000000]   DMA      [mem 0x0000000060000000-0x0000000067ffffff]
[    0.000000]   Normal   empty
[    0.000000]   HighMem  empty
```

Выполняем команду ls:

```
~ # ls
bin                    initramfs.cpio.gz  sbin
dev                    root               usr
```

Монтируем директорию proc и выполняем команду ps:

```
~ # mount -t proc proc proc
~ # ps
PID   USER      TIME  COMMAND
    1 0         0:00 /bin/ash
    2 0         0:00 [kthreadd]
    3 0         0:00 [pool_workqueue_]
    4 0         0:00 [kworker/R-rcu_g]
    5 0         0:00 [kworker/R-sync_]
    6 0         0:00 [kworker/R-kvfre]
    7 0         0:00 [kworker/R-slub_]
    8 0         0:00 [kworker/0:0-rcu]
```