

Создаём init.c и собираем под ARM:

```
arm-linux-gnueabi-gcc -static init.c -o init
```

Архивируем его:

```
echo init | cpio -o -H newc | gzip > initramfs.cpio.gz
```

Собираем busybox:

```
ARCH=arm make qemu_arm_defconfig ARCH=arm make menuconfig
```

```
ARCH=arm CROSS_COMPILE=arm-linux-gnueabi- make -j6
```

```
QEMU_AUDIO_DRV=none qemu-system-arm -M virt -bios u-boot.bin -nographic
```

Собираем корневую файловую систему:

```
QEMU_AUDIO_DRV=none qemu-system-arm -M vexpress-a9 -kernel zImage -initrd initramfs.cpio.gz -dtb  
vexpress-v2p-ca9.dtb -append "console=ttyAMA0 rdinit=/bin/ash" -nographic
```

```
35 0      0:01 [kworker/u16:1-e]
36 0      0:00 [oom_reaper]
37 0      0:00 [kworker/R-write]
38 0      0:00 [kcompactd0]
39 0      0:00 [kworker/R-kbloc]
40 0      0:00 [kworker/u16:2-e]
41 0      0:00 [kworker/R-ata_s]
42 0      0:00 [kworker/R-edac-]
43 0      0:00 [kworker/R-devfr]
44 0      0:00 [watchdogd]
45 0      0:00 [kworker/R-rpcio]
46 0      0:00 [kworker/0:1H]
47 0      0:00 [kworker/R-xprti]
49 0      0:00 [kswapd0]
50 0      0:00 [kworker/R-nfsio]
53 0      0:00 [kworker/u16:6-e]
54 0      0:00 [kworker/u16:7-a]
59 0      0:00 [irq/35-mmci-pl1]
65 0      0:00 [kworker/0:3-eve]
69 0      0:00 [kworker/R-mld]
70 0      0:00 [kworker/R-ipv6_]
83 0      0:00 ps

~ # arp
~ # route
Kernel IP routing table
Destination      Gateway          Genmask         Flags Metric Ref    Use Iface
~ # ls
bin              linuxrc          sbin
dev              proc             sys
initramfs.cpio.gz root             usr
~ # who
~ #
~ #
~ # whoami
whoami: unknown uid 0
~ # QEMU: Terminated
```