# Booting linux on ESP32

Sources:

- https://github.com/jcmvbkbc/esp-idf/tree/linux
- https://github.com/jcmvbkbc/linux-xtensa/tree/xtensa-5.19-esp32
- buildroot mainline

```
$ cd examples/get-started/linux_boot
$ idf.py build
$ idf.py -p /dev/ttyUSB0 flash
$ parttool.py -p /dev/ttyUSB0 write_partition --partition-name linux --input
xip-esp32-esp32/arch/xtensa/boot/xipImage
$ parttool.py -p /dev/ttyUSB0 write_partition --partition-name rootfs --
input build-xtensa-nommu-de108/images/rootfs.cramfs
```

Gives the following:

```
ets Jul 29 2019 12:21:46

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0030,len:4660
load:0x40078000,len:14272
ho 0 tail 12 room 4
load:0x40080400,len:3192
entry 0x4008060c
I (92) psram: This chip is ESP32-D0WD
I (93) spiram: Found 64MBit SPI RAM device
I (93) spiram: SPI RAM mode: flash 80m sram 80m
I (95) spiram: PSRAM initialized, cache is in low/high (2-core) mode.
I (103) cpu_start: Pro cpu up.
I (106) cpu_start: Starting app cpu, entry point is 0x400810bc
I (0) cpu_start: App cpu up.
I (609) spiram: SPI SRAM memory test OK
I (617) cpu_start: Pro cpu start user code
I (617) cpu_start: cpu freq: 160000000
I (617) cpu_start: Application information:
I (620) cpu_start: Project name:     linux_boot
I (625) cpu_start: App version:      v4.4.1-1-g2a25757abfa0
I (631) cpu_start: Compile time:     Jun  3 2022 06:49:46
I (637) cpu_start: ELF file SHA256:  a2a563b5646d5ebb...
I (643) cpu_start: ESP-IDF:          v4.4.1-1-g2a25757abfa0
I (650) heap_init: Initializing. RAM available for dynamic allocation:
I (657) heap_init: At 3FFAE6E0 len 00001920 (6 KiB): DRAM
I (663) heap_init: At 3FFB2BA0 len 0002D460 (181 KiB): DRAM
I (669) heap_init: At 3FFE0440 len 00003AE0 (14 KiB): D/IRAM
```

```
I (675) heap_init: At 3FFE4350 len 0001BCB0 (111 KiB): D/IRAM
I (682) heap_init: At 4008C1A0 len 00013E60 (79 KiB): IRAM
I (688) spiram: Adding pool of 4095K of external SPI memory to heap
allocator
I (696) spi_flash: detected chip: generic
I (700) spi_flash: flash io: dio
I (705) cpu_start: Starting scheduler on PRO CPU.
I (0) cpu_start: Starting scheduler on APP CPU.
I (715) spiram: Reserving pool of 32K of internal memory for DMA/internal
allocations
This is esp32 chip with 2 CPU core(s), WiFi/BT/BLE, silicon revision 3, 4MB
external flash
Minimum free heap size: 4452739 bytes
[    0.000000] Ignoring boot parameters at (ptrval)
[    0.000000] Linux version 5.18.0-00045-g137adeae9008 (jcmvbkbc@octofox)
(xtensa-esp32-elf-gcc (GCC) 12.1.0, GNU ld (GNU Binutils) 2.37) #55 PREEMPT
Fri Jun 3 11:33:34 PDT 2022
[    0.000000] config ID: c2bcfffe:1cc5fe96
[    0.000000] Zone ranges:
[    0.000000]   Normal   [mem 0x000000003f800000-0x000000003fbfffff]
[    0.000000] Movable zone start for each node
[    0.000000] Early memory node ranges
[    0.000000]   node   0: [mem 0x000000003f800000-0x000000003fbfffff]
[    0.000000] Initmem setup node 0 [mem
0x000000003f800000-0x000000003fbfffff]
[    0.000000] pcpu-alloc: s0 r0 d32768 u32768 alloc=1*32768
[    0.000000] pcpu-alloc: [0] 0
[    0.000000] Built 1 zonelists, mobility grouping off.  Total pages: 1016
[    0.000000] Kernel command line: earlycon=esp,esp32-
uart,mmio32,0x3ff40000,115200n8 console=ttyS0,115200n8 debug rw
root=mtd:data init=/bin/sh ip=dhcp
[    0.000000] Unknown kernel command line parameters "ip=dhcp", will be
passed to user space.
[    0.000000] Dentry cache hash table entries: 1024 (order: 0, 4096 bytes,
linear)
[    0.000000] Inode-cache hash table entries: 1024 (order: 0, 4096 bytes,
linear)
[    0.000000] mem auto-init: stack:off, heap alloc:off, heap free:off
[    0.000000] virtual kernel memory layout:
[    0.000000]     lowmem  : 0x3f800000 - 0x3fc00000  (    4 MB)
[    0.000000]     .text   : 0x400d0000 - 0x401b3ad8  (  910 kB)
[    0.000000]     .rodata : 0x401b4000 - 0x401f7000  (  268 kB)
[    0.000000]     .data   : 0x3f800000 - 0x3f873880  (  462 kB)
[    0.000000]     .init   : 0x3f873880 - 0x3f877d00  (   17 kB)
[    0.000000]     .bss    : 0x3f877d00 - 0x3f8a024c  (  161 kB)
[    0.000000] Memory: 3372K/4096K available (910K kernel code, 462K rwdata,
268K rodata, 68K init, 161K bss, 724K reserved, 0K cma-reserved)
[    0.000000] rcu: Preemptible hierarchical RCU implementation.
[    0.000000]  Trampoline variant of Tasks RCU enabled.
[    0.000000] rcu: RCU calculated value of scheduler-enlistment delay is 10
```

```
jiffies.
[    0.000000] NR_IRQS: 33
[    0.000000] clocksource: ccount: mask: 0xffffffff max_cycles: 0xffffffff,
max_idle_ns: 11945377789 ns
[    0.000066] sched_clock: 32 bits at 160MHz, resolution 6ns, wraps every
13421772796ns
[    0.001308] Calibrating delay loop (skipped)... 160.00 BogoMIPS preset
[    0.001808] pid_max: default: 4096 minimum: 301
[    0.003279] Mount-cache hash table entries: 1024 (order: 0, 4096 bytes,
linear)
[    0.004232] Mountpoint-cache hash table entries: 1024 (order: 0, 4096
bytes, linear)
[    0.048931] cblist_init_generic: Setting adjustable number of callback
queues.
[    0.049741] cblist_init_generic: Setting shift to 0 and lim to 1.
[    0.059606] rcu: Hierarchical SRCU implementation.
[    0.076658] devtmpfs: initialized
[    0.083391] clocksource: jiffies: mask: 0xffffffff max_cycles:
0xffffffff, max_idle_ns: 19112604462750000 ns
[    0.084360] futex hash table entries: 16 (order: -5, 192 bytes, linear)
[    0.121536] clocksource: Switched to clocksource ccount
[    0.174347] workingset: timestamp_bits=30 max_order=10 bucket_order=0
[    2.082460] 3ff40000.serial: ttyS0 at MMIO 0x3ff40000 (irq = 1, base_baud
= 0) is a ESP32
[    2.357056] printk: console [ttyS0] enabled
[    2.370634] physmap-flash 3f400000.flash: physmap platform flash device:
[mem 0x3f400000-0x3f7fffff]
[    2.373536] 2 fixed-partitions partitions found on MTD device
3f400000.flash
[    2.378928] Creating 2 MTD partitions on "3f400000.flash":
[    2.385423] 0x000000040000-0x000000200000 : "linux"
[    2.403718] 0x000000200000-0x000000400000 : "data"
[    2.435035] cramfs: checking physical address 0x3f600000 for linear
cramfs image
[    2.436006] cramfs: linear cramfs image on mtd:data appears to be 596 KB
in size
[    2.443537] VFS: Mounted root (cramfs filesystem) readonly on device
31:1.
[    2.448906] devtmpfs: mounted
[    2.453437] Freeing unused kernel image (initmem) memory: 12K
[    2.455068] This architecture does not have kernel memory protection.
[    2.462924] Run /bin/sh as init process
[    2.465281]   with arguments:
[    2.468212]     /bin/sh
[    2.470645]   with environment:
[    2.474634]     HOME=/
[    2.476129]     TERM=linux
[    2.478803]     ip=dhcp
[    2.603870] random: fast init done
[    3.535085] Caught unhandled exception in 'sh' (pid = 1, pc = 0x3f900054)
- should not happen
```

```
[    3.535085]  EXCCAUSE is 2
[    3.537257] Kernel panic - not syncing: Attempted to kill init!
exitcode=0x00000004
[    3.544067] ---[ end Kernel panic - not syncing: Attempted to kill init!
exitcode=0x00000004 ]---
```

From:
http://wiki.osll.ru/ - **Open Source & Linux Lab**

Permanent link:
**http://wiki.osll.ru/doku.php/etc:users:jcmvbkbc:linux-xtensa:esp32?rev=1659815842**

Last update: **2022/08/06 22:57**