

ECE 471 – Embedded Systems

Lecture 15

Vince Weaver

`http://web.eece.maine.edu/~vweaver`

`vincent.weaver@maine.edu`

4 October 2021

Announcements

- HW#5 was posted
- Midterm is a week from Friday, the 15th



Operating Systems Types

- Monolithic kernel – everything in one big address space. Something goes wrong, lose it all. Faster
- Microkernel – separate parts that communicate by message passing. can restart independently. Slower.
- Microkernels were supposed to take over the world. Didn't happen. (GNU Hurd?)
- Famous Torvalds (Linux) vs Tannenbaum (Minix) flamewar



Common Desktop/Server Operating Systems

- Windows
- OSX
- Linux
- FreeBSD / NetBSD / OpenBSD
- UNIX (Irix/Solaris/AIX/etc.)
- BeOS/Haiku



Embedded Operating Systems

- Microsoft WinCE, Windows Mobile
- Linux / Android
- VXworks – realtime OS, used on many space probes
- Apple iOS
- QNX – realtime microkernel UNIX-like OS, owned by Blackberry now
- Cisco iOS
- ThreadX – found in Pi GPU



Embedded Linux Distributions

- linaro – consortium that work on ARM software
- openwrt – small distro initially designed for wireless routers
- yocto – Linux Foundation sponsored embedded distro
- maemo – embedded distro originally by Nokia (obsolete)
- MeeGo – continuation of maemo, also obsolete
- Tizen – Follow up on MeeGo, by Samsung and Intel
- Ångstrom – Merger of various projects
- And many others. It's very easy to put together a Linux



distribution



Firmware

- What is firmware?



Device Firmware

- Devices are their own embedded systems these days. May even have full CPUs, etc.
- Need to run code. Firmware.
- In ROM? Or upgradable? Why might you want to upgrade? (bug fixes, economy, etc.)
- Talk about recent USB firmware malware



Firmware

Provides booting, configuration/setup, sometimes provides rudimentary hardware access routines.

Kernel developers like to complain about firmware authors. Often mysterious bugs, only tested under Windows, etc.

- BIOS – legacy 16-bit interface on x86 machines
- UEFI – Unified Extensible Firmware Interface
ia64, x86, ARM. From Intel. Replaces BIOS
- OpenFirmware – old macs, SPARC
- LinuxBIOS



Bootloaders

- Firmware doesn't usually directly load Operating System
- Bootloader (relatively simple code, just smart enough to load OS and jump to it) is loaded first
- Bootloader is often on a very simple filesystem (such as FAT) as the code has to be simple (possibly even written in assembly language)
- Bootloader is often just complex enough to load OS kernel from disk/network/etc and jump to it



Raspberry Pi Booting (pre pi4)

- Unusual
- Small amount of firmware on SoC
- ARM 1176 brought up inactive (in reset)
- Videocore loads first stage from ROM
- This reads `bootcode.bin` from FAT partition on SD card into L2 cache. It's actually a RTOS (real time OS in own right "ThreadX") (50k)
- This runs on videocard, enables SDRAM, then loads `start.elf` (3M)



- This initializes things, the loads and boots Linux kernel.img. (also reads some config files there first)
(4M)



Pi4 booting

- <https://www.raspberrypi.org/documentation/hardware/raspberrypi/booteprom.md>
- SPI EEPROM holds equivalent of `bootcode.bin`, no longer read from partition
- Why? SDRAM, PCIe USB, etc are more complex
- No network/USB booting yet, coming soon



More booting

- Most other ARM devices, ARM chip runs first-stage boot loader (often MLO) and second-stage (uboot)
- FAT partition
Why FAT? (Simple, Low-memory, Works on most machines, In theory no patents despite MS's best attempts (see exfat))
The boot firmware (burned into the CPU) is smart enough to mount a FAT partition

