Embedded Linux and Emulation Overview

Advanced Embedded Linux Development

with **Dan Walkes**

University of Colorado **Boulder**

Learning objectives:

Understand elements of Embedded Linux.

Understand Linux Hardware Support.

Introduce QEMU for Emulation.

Elements of Embedded Linux

- Toolchain
  - Compiler and other tools needed to create code for your device

Elements of Embedded Linux

- Bootloader
  - Initializes the board and loads the Linux kernel.
- Kernel
  - Manages system resources, interfaces with hardware
- Root filesystem
  - Libraries and programs that run after kernel has completed initialization.

Linux Hardware Support

- Linux supports 30+ architectures
  - All are at least 32 bit architectures.
  - Most hardware includes a Memory Management Unit (MMU).
    - Exception is uClinux.

Linux Hardware Support

- Requires larger amount of RAM than simple microcontroller minimums (16MB good minimum.)
- Requires some type of non-volatile storage (8MB + is a suggestion.)

Linux Emulation - QEMU

- Emulates a processor architecture and a board built using that architecture.
  - qemu-system-aarch64 simulates 64 bit ARM architecture devices.

Mastering Embedded Linux Programming Chapter 1
https://www.qemu.org/

Linux Emulation - QEMU

- Example starting an ARM aarch64 QEMU system:

```
qemu-system-aarch -m 256M -nographic -M
virt -kernel /path/to/Image -append
"rdinit=/bin/sh" -initrd
/path/to/initramfs.cpio.gz
```

Linux Emulation - QEMU

```
qemu-system-aarch -m 256M -nographic -M
virt -kernel /path/to/Image -append
"rdinit=/bin/sh" -initrd
/path/to/initramfs.cpio.gz
```

- Allocates/emulates 256 MB of RAM

Mastering Embedded Linux Programming Chapter 1
https://www.qemu.org/
https://wiki.qemu.org/Documentation/Platforms/ARM

Linux Emulation - QEMU

```
qemu-system-aarch -m 256M -nographic -M
virt -kernel /path/to/Image -append
"rdinit=/bin/sh" -initrd
/path/to/initramfs.cpio.gz
```

- Emulates a “virt” target

Mastering Embedded Linux Programming Chapter 1
https://www.qemu.org/
https://wiki.qemu.org/Documentation/Platforms/ARM

Linux Emulation - QEMU

```
qemu-system-aarch -m 256M -nographic -M
virt -kernel /path/to/Image -append
"rdinit=/bin/sh" -initrd
/path/to/initramfs.cpio.gz
```

- Uses kernel image in /path/to/Image to boot the kernel (bypassing bootloader).
- Passes arguments in “append” to the kernel.
  - Start a shell interpreter

Mastering Embedded Linux Programming Chapter 1
https://www.qemu.org/
https://wiki.qemu.org/Documentation/Platforms/ARM

Linux Emulation - QEMU

```
qemu-system-aarch -m 256M -nographic -M
virt -kernel /path/to/Image -append
"rdinit=/bin/sh" -initrd
/path/to/initramfs.cpio.gz
```

- Uses a root filesystem in a ramdisk
  - Contents are described in the initramfs.cpio.gz file

Mastering Embedded Linux Programming Chapter 1
https://www.qemu.org/
https://wiki.qemu.org/Documentation/Platforms/ARM