Bare Metal Introduction
Runtime Environment of Bare Metal

Linux

- User application using C Library
- Linux Kernel
- Device Drivers

Bare Metal

- Hardware
- Hardware

User Mode

Privileged Mode
Bare Metal Intro

• There are no OS services
  - ..
  - ..
  - ..
  - drivers (LEDs, WatchDog, Network)
  - apps (node.js, native GCC, X11)

• Embedded OS's:
  - Full Feature:
    Linux, Windows Embedded, Android, QNX
  - Low level:
    Custom RTOS (Real Time OS); many others!
Bare Metal Advantages

• Advantages of running bare metal
  – ..
  – no extra code running (certification issues)
  – no "wasted" space (if small device)
  – ..
    (no context switches, pre-emption, page-faults....)
  – some hardware not powerful enough to run full OS
    (no MMU for protected memory, <2 meg ram...)

19-11-14
Practical Differences

- ..
  - no stdin/stdout
  - use UART: Universal Asynchronous Receiver-Transmitter

- ..
  - main() uses a while(1) {...}

- ..
  vs via OS drivers
  - Try to use library to abstract the access a little.

- StarterWare UART modules
  - UART: low-level reads and writes to serial port.
    - must initialize first.
  - ConsoleUtils: high-level printf and scanf
Compile & Boot Process
Compile & Boot: UBoot Prompt

Host

- Compile app: bm_uart.bin
  - bm_uart.c + StarterWare
  - Use Linaro GCC (arm-none-eabi)
- Deploy
  - copy bm_uart.bin to~/cmpt433/public/baremetal
  - ..

- Easy to change on host
  - TFTP server of download.bin

Target

- In UBoot:
  - press a key (to get prompt)
  - => tftp ... download.bin
  - => go ...
- Bare metal App:
  - runs from address 0x8000 0000
Compile & Boot: uEnv.txt

- UBoot loads uEnv.txt for boot commands
  - boot Linux with /boot/uEnv.txt on eMMC
  - can change uEnv.txt to boot bare metal.
  - before edits, create backup copies.

- UBoot can copy files on eMMC
  - to change ‘default’ boot commands, copy the correct file into uEnv.txt

- Technical note
  - Our UBoot actually looks for different uEnv.txt files and handles them differently.
  - For bare metal, put commands in /uEnv.txt. Boot to Linux by wiping /uEnv.txt’s contents.
UART Tx Demo

- Compile `bm_uart.c` on host
- Link it in public folder as `download.bin`
- Load on target and see printing out.
- When running, board reboots!
  - WD started (by UBoot?); reboots in ~45s
  - Hit the watch by adding:
    ```
    #include "watchdog.h"
    in main():
    WatchdogTimerSet(SOC_WDT_1_REGS, ...);
    ```
Demo Setup Configuration (1/2)

Show the folders for the following:

- **Host:**
  - General
    - StarterWare
      (~/cmpt433/AM335X_StarterWare_02_00_01_01)
    - Linaro GCC (~/cmpt433/linaro-gcc)
  - App
    - bm_uart.c, load-script, Makefile
    - Deploy (~/cmpt433/public/baremetal)
      ln -s xyz.bin download.bin
Demo Setup Configuration (2/2)

- **Target**
  - uEnv.txt
    - Linux: /boot/uEnv.txt
    - Bare metal: /boot/uEnv-BareMetal.txt
      copy to /uEnv.txt to activate

- **Setting Default Boot Choice**
  - Boot to Bare Metal:
    => ext4load mmc 1:1 0x82000000 /boot/uEnv-BareMetal.txt
    => ext4write mmc 1:1 0x82000000 /uEnv.txt ${filesize}
    => boot
  - Boot to Linux:
    => ext4write mmc 1:1 0x82000000 /uEnv.txt 0
    => boot
Summary

- Bare metal apps give you full control of the hardware, but you lose the benefits of a full OS.
  - No terminal: use a UART
  - main() never exits
  - direct access to hardware registers
- Configure uEnv.txt for booting
  - host links download.bin to the actual application file to allow us to easily change the link on the host and change the app we download.