Announcements

Outline

- Quantifying the Energy Consumption of a Pocket Computer and a Java Virtual Machine
  - Keith Farkas (DEC WRL), Jason Flinn (CMU), Godmar Back (Univ. of Utah), Dirk Grunwald (Univ of Colo. Boulder), Jennifer Anderson (Vmware)

Energy consumed (Joules) = time * power consumed (Watts)

Goals

1. Energy usage characteristics of the Itsy
   - Itsy is a prototype PDA built at DEC (Compaq) SRC
   - 200 MHz StrongARM SA-1100 microprocessor
   - 320x200, 0.18mm, 15 level gray scale display
   - Touchscreen, microphone, speaker, serial and irda
   - 64MB RAM, 32MB flash
   - 2 AAA battery
   - Precursor to the iPAQ

Itsy battery power tricks

- Can change the CPU clock to 206 MHz, 133 MHz and 59 MHz)
- It can scale the voltage to 1.5V and 1.23V

  – 30 minutes in high power mode
  – 2 hours in high power “Idle” mode
  – 18 hours in low power (59 MHz) Idle mode

Comparison between Itsy and laptop

- Compared Thinkpad (233 Mhz, 64 MB) and Itsy
  - Laptops consume a lot more overall power
  - Itsy allows more power states
    - Certain subsystems have lot higher relative power costs (because other systems consume less power)
      - E.g. on a laptop
        - » 68% - display, 20% - disk, 12% CPU and memory
    - Itsy consumes less for display, but adding a backlight has a much higher relative cost
    - Memory subsystem has impact on Itsy

JVM

- Java Virtual machine
- Interesting to look at Java on a PDA because it is simpler to expect apps to be downloaded to the PDA than expect them to be installed all the time
- Looked at the power characteristics of:
  - Single JVM vs multiple JVM
  - Compressed vs Uncompressed class files
  - Class loading vs JIT compilation
  - Cache flushing after code generation

Results

- Single JVM less power than multiple JVM
  - Reliability issues
- Compressed vs uncompressed class files: not much difference
- Class loading vs JIT Compilation
  - Preloading works, but have to be careful on what to preload
- Cache flushing after code generation
  - Little impact; only 16 KB I and 8KB D cache
- Interpreting and JIT
  - JIT has dramatic gain; maybe because of KAFFE for JVM
- AWT polling frequency – slight difference

Discussion

- Power consumption has to be investigated closely for each underlying architecture
- There are few “generic” tricks for every hardware