
Power Management of iPAQ

Sukjae Cho

(sjcho@east.isi.edu)

(sjcho@redwood.snu.ac.kr)

Agenda

- **About iPAQ**
 - What's iPAQ
 - Linux on iPAQ
- **Power management of iPAQ**
- **Sleep mode of iPAQ**
- **SA-1100 register monitor**

What's iPAQ

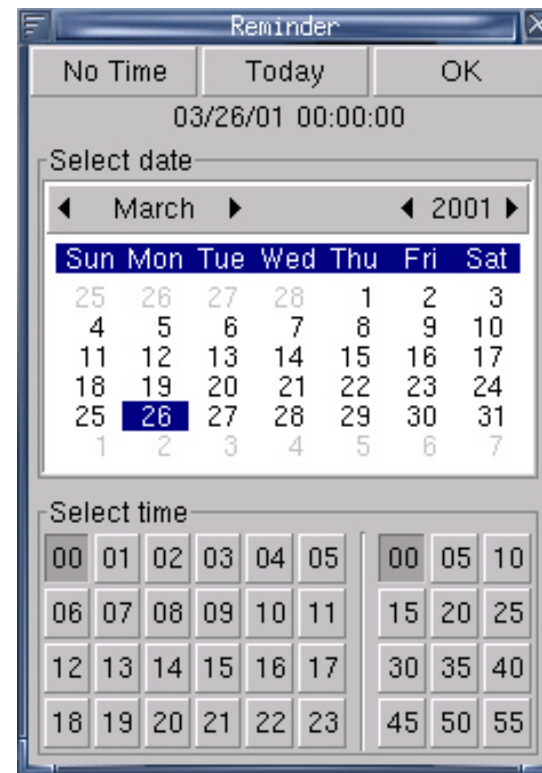
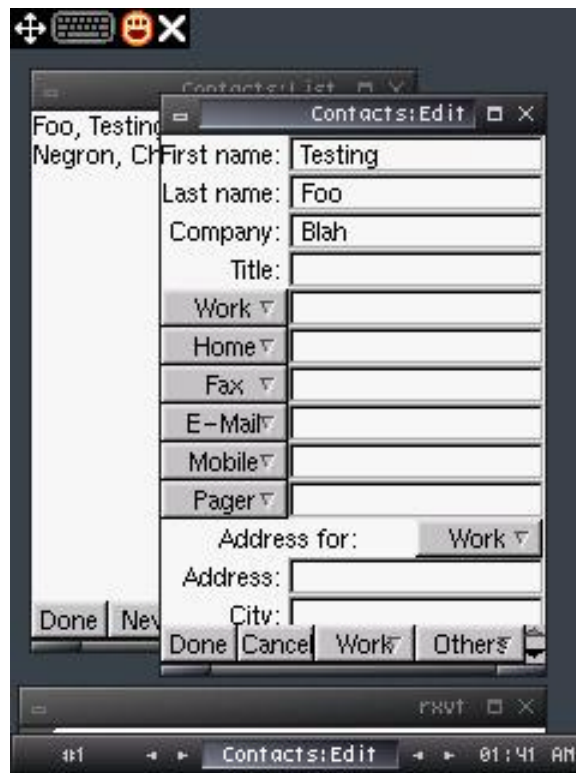
- **Powerful handheld computer**

- **206MHz StrongArm SA-1110 processor**
- **320x240 resolution color TFT LCD**
- **Touch screen**
- **32MB SDRAM / 16MB Flash memory**
- **USB/RS-232/IrDA connection**
- **Speaker/Microphone**
- **Lithium Polymer battery**
- **PCMCIA card expansion pack & CF card expansion pack**



Linux on iPAQ

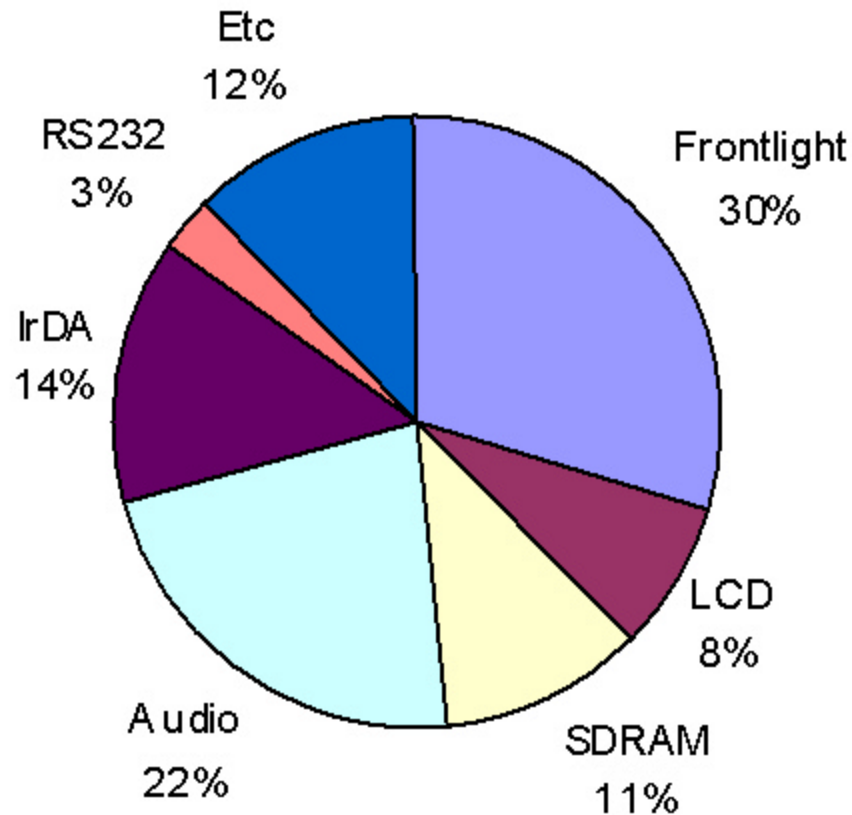
- Modification based on arm linux kernel 2.4
- ARMV binaries can run without modification
- Many projects for iPAQ linux are on the work



Agenda

- **About iPAQ**
- **Power management of iPAQ**
 - **Power usage of iPAQ components**
 - **Previous work**
 - **Tweaks for reducing power consumption of iPAQ**
 - **Improvements**
- **Sleep mode of iPAQ**
- **SA-1100 register monitor**

Power Usage of Components in iPAQ



*** Note**

SDRAM power consumption is guessed by setting it self-refresh mode

CPU is idle state of most of its time

Audio, IrDA, RS232 power is measured when each part is idling

Etc includes CPU, flash memory, touch screen and all other devices

Frontlight brightness was 16

Previous Work (1/2)

- **Idle loop**
 - Idle process uses Strong Arm's idle mode
 - In idle mode, CPU stops its clocking and wait for interrupt
- **Turning off sound chip**
 - Turning off power to sound chip when we don't need it
 - Maintain all CPU's output to sound chip low
 - Prevents the sound chip from draining power from CPU's output pins
- **Screen saver**
 - Turns off LCD and backlight after some time of no user-input
- **Scaling down the CPU clock**
 - SA1110 core clock can be switched from 57.3MHz to 214.8MHz by software
 - No voltage change is supported
 - Time to change clock takes about 150ms for calibrating delay loop

Previous Work (2/2)

- **iPAQ consumes 470mW if you turn on LCD (no frontlight) and turn off other unused chips**
 - **iPAQ that runs linux consumes 1.9 times more power than iPAQ that runs Windows CE does in this condition**
 - **iPAQ lasts about 8 hours in this condition**
 - **Actual usable time with front light is about 4 hours**
- **iPAQ linux can sleep but can't wake up**
 - **WinCE can last a long time in the sleep mode**
 - **Linux should be turned off when you don't use it**

Tweaks for reducing power consumption of iPAQ (1/2)

- **Enable SDRAM auto power down mode**
 - SA1110 supports SDRAM auto power down mode
 - In SDRAM auto power down mode, CPU disables clock input to memory when the memory is not in use
 - Saves about 190mW when the memory activity is low

- **Tweak in Display Driver**
 - Reduce refresh rate of LCD
 - Current LCD driver refreshes rate is about 60Hz
 - Refresh rate can be adjusted by various LCD timing
 - We can reduce SDRAM usage, bus usage and power consumption by lowering refresh rate
 - Disabling LCD controller when there is no change in screen
 - Both method require changes in X server or user applications

Tweaks for reducing power consumption of iPAQ (2/2)

■ CPU-scaling experiment result

- Saves about 100mW ~ 200mW if CPU runs at 57.3MHz rather than 206MHz
- Lowering CPU clock reduce static power consumption of CPU
- Lowering CPU clock reduce other components whose clocks are derived from CPU clock (memory, LCD controller and etc.)

Improvements

- **When LCD is on, most other chips are off and CPU is idle,**
 - **Linux iPAQ consumes**
 - **470mW with previous linux kernel**
 - **280mW with SDRAM power down mode**
 - **238mW with SDRAM power down mode and 30Hz refresh rate**
 - **172mW with SDRAM power down mode and CPU speed 56MHz**
 - **WinCE iPAQ consumes**
 - **248mW**

*** Note**

- **Consumes 460mW more power if the front light is on**
- **Linux can consume as low as 98mW if it also turns off LCD**

Agenda

- **About iPAQ**
- **Power management of iPAQ**
- **Sleep mode of iPAQ**
 - Previous work
 - Works done for sleep/wakeup
 - More things to do
- **SA-1100 register monitor**

Previous Work

■ Itsy

- Research prototype of handhelds device of compaq
- Uses SA1100 processor
- Uses kernel based on linux kernel 2.2
- Kernel and bootloader support sleep/wakeup

■ Yopy

- Multimedia PDA. Commercial product is not released.
- Uses SA1110 processor
- Uses kernel based on linux kernel 2.2
- Kernel and bootloader support sleep/wakeup

■ Arm linux kernel 2.4.0

- Arm linux kernel did not support sleep/wakeup
- Bootloader did not support sleep/wakeup
- Some device drivers support suspend/resume
- Frameworks for APM

Works Done for Sleep/Wakeup (1/2)

■ Bootloader modification

- Bootloader tests whether the cause of reset is the sleep reset
- Bootloader wakeup SDRAM from self-refresh state to normal state
- Bootloader jumps to linux kernel wakeup entry point which is set by kernel

Works Done for Sleep/Wakeup (2/2)

■ Kernel modification

□ Enter sleep mode

- Execute each device driver's suspend code
- Save/restore the vital registers of SA1110
- Setup power management unit properly
- Give wake-up information to bootloader before sleep
- Execute work around code to avoid SA1110 sleep bugs
- Enter sleep mode

□ Wake up from sleep

- Restore CPU status
- Execute each driver's resume code

□ Device drivers' support for suspend/resume

- LCD driver
- Touch screen & backlight
- Sound driver
- DMA
- RTC

More Things To Do

- **Test more device drivers' suspend/resume**
 - IRDA
 - USB
 - CF cards
 - PCMCIA cards
- **Implementing power consumption reduction method**
 - CPU clock adjustment
 - Fix CPU clock scaling code
 - Dynamic CPU clock scaling on the load
 - Automatic refresh rate adjustment
 - Improve display driver and interface
- **Implement various power down-mode**
 - Alarm sleep mode
 - Slow-running mode
 - Screen off mode

SA-1100 Register Monitor Module

- Provide simple method to read & write 150 special registers in SA-1100
- Ex)

```
(none):~# insmod ./regmon.o
(none):~# cd /proc/cpu/registers/
(none):/proc/cpu/registers# ls
DBAR1  DBTA2  DCSR4   GPDR   LCCR0   MDCAS22  PGSR  RTSR   UDCOMP  UTDR
DBAR2  DBTA3  DCSR5   GPLR   LCCR1   MDCNFG   PMCR  RTTR   UDCSR   UTDR
DBSA0  DBTA4  DDAR0   GPSR   LCCR2   MDREFR   POSR  SMCNFG  UDCWC   UTDR
...
DBSB5  DCSR1  GPCLKR2  ICLR   MDCAS02  OSSR     RCSR  UDCD0  UTCR3
DBTA0  DCSR2  GPCLKR3  ICMR   MDCAS20  OWER     RSRR  UDCDR  UTCR3
DBTA1  DCSR3  GPCR     ICPR   MDCAS21  PCFR     RTAR  UDCIMP  UTCR4
(none):/proc/cpu/registers# cat PSPR
0x00000011
(none):/proc/cpu/registers# echo 0x12345678 >PSPR
(none):/proc/cpu/registers# cat PSPR
0x12345678
```

Special Thanks to

- Carl
- Dong-in
- Brian
- Chris