Designing the Next-Generation of Handheld Devices

Class 445
Thursday, September 18th, 2003

Andrew Girson

240-558-2014
agirson@inhandelectronics.com
www.inhandelectronics.com

Presentation Focus

A survey of the significant issues in the development of system hardware and system software for programmable handheld devices, based on 32-bit CPUs and operating systems

Presentation Overview

- Market Opportunity & Design Challenges
- How Handheld Design is Different
- Designing a Handheld

Handhelds - A Growing Market

- Over 400M ARM CPUs shipped in 2001
  - majority of these CPUs in cellular telephones, PDAs, and other wireless devices

- Yet, cellular telephone penetration was just 6% in China in 2000
  - US penetration 41%
  - Worldwide penetration 12%

Opportunity

- Mass Markets
  - Merging multiple handhelds into one
  - Bringing handhelds to everybody
- Vertical markets
  - Bringing smart handhelds to new segments
  - Upgrading “fixed-function” handhelds
- All Markets
  - Leveraging new technologies

What This Means for Engineers

Opportunity

↓

Volatility & Innovation

↓

Design Challenges

"make it smaller" "ruggedize it" "make it run for weeks on a single charge"

"make it hands-free" "add location-based services" "use this new wireless technology"

How Handheld Design is Different

- Designers of handheld devices must overcome all of the challenges of traditional embedded design, yet they must also...

- ...make it fit
- ...make it last

How Handheld Design is Different

It all comes down to…

“Form Factor” & “Battery Life”

Designing a Handheld

- Define user requirements
- Select a Form Factor
- Select a Display
- Select Batteries
- Select a CPU & OS
- Design system hardware platform
- Develop system software platform

Define User Requirements

- Will it be used outside?
- What types of data will be displayed?
- How will the user input data?
- Is it likely to be dropped on a hard surface?
- Could it get wet?
- Will the user have ready access to AC power?
- How will the user talk to the network?

Select a Form Factor

- PDA
- Cell phone
- Smartphone
- Web Tablet
- Wearable Computer
- Wireless Terminal

Select a Form Factor

- Develop mechanical specifications
  - Package form factor and materials
  - Display, battery, electronics, and input integration
  - Manufacturability and assembly issues
  - Environmental and ruggedization issues

Select a Display

- Based on form factor
- Based on user interface requirements
- Based on ambient environment
  - indoor, outdoor, readable in sunlight, readable in low light, viewing angle, orientation

Select a Display

- Different Technologies
  - STN, TFT, OLED
  - reflective, transmissive, transflective
  - Backlights - CCFT, EL, LED (watch out for power consumption and noise)
- Always get:
  - a demo of the working display in the environment in which it will be used
  - a power measurement with and without backlighting

Select Batteries

- Two Types of Batteries
  - Primary
  - Rechargeable
- Each has advantages and disadvantages
- Make sure you estimate battery life for "likely" user scenarios

Characteristics of Primary Batteries

- Primary batteries have higher energy density than rechargeables
- Carbon-Zinc and Alkaline are commodity “drugstore” batteries
- Lithium batteries are best all-around performers with very flat discharge curves, high energy density, and excellent shelf life and low-temperature capacity

Comparison of Primary Battery Types

Chemistry	Cell Voltage (V)	Cell Capacity (mAh)	Gravimetric Energy Density (WHr/kg)	Volumetric Energy Density (WHr/l)	Capacity loss per YEAR (@ 20C)
Carbon-Zinc	1.5	10-5000	105-195	100-180	5%
Alkaline	1.5	10-5000	125-225	150-440	4%
Silver Oxide	1.5	5-200	155-285	250-500	3%
Zinc Air	1.4	30-1000	245-455	470-1450	5% (sealed)
Lithium	1.5, 3.0	10-3000	32-260	340-500	1%

Characteristics of Rechargeable Batteries

- Rechargeable batteries are reusable, reducing cost and environmental issues
- Li-Ion batteries are best performers, but are costly and require tight control of charging algorithms
- NiMH and Li-Ion can be dangerous if overcharged
- NiCad is simpler with better overcharging tolerance, but lower performing

Comparison of Rechargeable Battery Types

Chemistry	Cell Voltage (V)	Cell Capacity (mAh)	Gravimetric Energy Density (WHr/kg)	Volumetric Energy Density (WHr/l)	Capacity loss per MONTH (@ 20C)
NiCad	1.2	50-5000	55	170	10%
NiMH	1.2	10-5000	70	250	15%
Li-Ion	3.6	25-1600	120	350	3%

Battery Life Estimates for Common Scenarios

- PDA - 900mW on; 200mW idle; 5mW sleep; 1% on time; 3% idle time
  - 3 AAA Alkaline cells (Energizer): ~200 hours
  - 1 Li-Ion cell (iPAQ): ~200 hours
- Smart sensor - 50mW on; 0.5mW sleep; 2% on time
  - 3 AAA Alkaline cells (Energizer): ~2700 hours
  - 1 Li-Ion cell (iPAQ): ~2900 hours

Select a CPU

- ARM architecture is entrenched
  - Pocket PC now is only on ARM
  - PalmOS is now based on ARM
  - Intel, TI, Samsung, Sharp, Atmel, Philips, and Motorola all have ARM CPUs specifically targeted at handhelds
- MIPs and x86 have intriguing options
  - AMD (Alchemy and Geode)

Select a CPU

- Important features
  - Power modes (run, idle, sleep, and more...)
  - Voltage and clock scaling
  - Lots of peripheral integration and I/O
  - On-chip memory
  - “Stacked” memory

Select an Operating System

- Palm OS is market leader
  - Will licensing restrictions loosen?
- Microsoft's Pocket PC is coming on strong
  - Will it overtake Palm?
- Symbian
  - Strong in cellular (Java)
- Linux
  - Will embedded growth translate to handhelds?

Hardware Platform

- Power Supply
- Test & Debug
- Memory
- User Input
- LCD
- Audio
- Communications
- Expansion

Hardware Platform - Power Supply

- Regulators
  - Linear
    - Cheap & simple, low noise
  - Switched-Mode
    - High efficiency
- Boost Supplies
  - Allow operation from fewer/smaller batteries
- Watch out for current transients
- Segment the power system

Hardware Platform - Test & Debug

- Small boards limit test points
- Small boards limit debug connectors

- Options
  - Single debug connector with all debug hardware on daughterboard
  - Large board for development; respin in production

Hardware Platform - Memory

- Non-volatile
  - Linear NOR Flash
    - allows XIP, can reduce DRAM requirements
  - Linear NAND Flash
    - reduces cost, scalable to large densities
  - Peripheral Flash Storage Devices
    - can be less expensive, straightforward parameter storage
- Volatile/SDRAM
  - Must have battery backup
  - New CPUs supporting 2.5V (and lower) parts

Hardware Platform - User Input

- Very dependent on user environment
  - Keypads, buttons, and touchpads imply larger device
  - Touchscreens allow smaller device, but can be fragile and noisy
  - Speech input allows small device, but ambient and audio subsystem noise are issues

Hardware Platform - LCD

- Very dependent on user environment
  - Affects device size
  - Affects device battery life
  - Different "lighting" techniques
  - Watch out for touchscreen pointing accuracy (noise) with large displays
  - Be careful with connector placement

Hardware Platform - Audio

- Can greatly impact battery life
- Audio component placement affects sound input and output quality

Hardware Platform - Comms

- Wireless Data and Voice
  - 802.11 is power-hungry
  - Bluetooth is not ubiquitous
  - New technologies (e.g., UWB, Zigbee) are intriguing
- Location (GPS)
- Component placement affects quality
- Many manufacturers are opting for off-the-shelf, plug-in solutions

Hardware Platform - Expansion

- Standard connectors are flexible, but drive device size
- Daughtercard connectors provide optional follow-on development
- New peripherals requiring higher speed busses (USB 2.0, Firewire, GB Ethernet)

- Make sure peripheral is off or in “sleep” when not in use

System Software

- Power Modes
- Frequency/Voltage Adjustment
- Interrupt Reduction
- Intelligent Waiting

System Software
Power Modes

- CPU Power Modes
  - Run, Idle, Doze, Sleep, Off
  - Prepare memory for battery backup
  - Allow instant-on
  - Set I/O pins properly in Sleep
- Peripheral Power Modes
  - Many peripherals have them
  - Usage can double or triple battery life

System Software
Frequency/Voltage Adjustment

- Power consumption linearly proportional to CPU core frequency
- Power consumption proportional to square of CPU core voltage

- Analyze your software for periods when performance is independent of CPU core frequency and adjust accordingly

System Software
Interrupt Reduction

- Make interrupt buffers larger
- Use DMA whenever possible
  - Allows CPU to remain in Idle
  - Reduces interrupt count
  - Reduces computational bandwidth requirements

System Software

Intelligent Waiting

- Avoid “spinning” - try to go to Idle
- If “spinning” is unavoidable, can clock frequency and voltage be reduced?

Summary

- Hardware, software, & mechanical design teams must interact
  - mechanical, electrical, and software engineering matters are highly intertwined
- Design challenges
  - Form factor
  - Battery life