Project Pengachu: Cheap Wireless Linux for Everyone

Wendy Ju
Rehmi Post
Matt Reynolds
Motivation

- Educational applications (e-books, CS lessons, etc)
- People need quick access to news, weather, Web, or digital books
- People need peer-to-peer voice or data connectivity, especially in infrastructure-poor places
Design Goals

- Small, cheap
- Long battery life
- Peer to peer ad-hoc wireless TCP/IP networking
- Open software, open hardware
- Complete software development environment on board
- Reusable design for embedded computing (to enable TTT!)
Hardware Detail

- Open hardware and software (no big SW license fees!)
- Simplified backplane bus using 1Mbps SPI for low cost
- On-board power management for long battery life (50mW while decoding MP3s)
- 128x64 pixel onboard LCD, possibly VGA or NTSC output in next version
- Projected system cost in volume O($50), easy to integrate into a cheaper few-chip solution
Built-In Networking

• Standards-based (TCP/IP)
• Everything peer-to-peer:
  – Wireless LAN: 900MHz, 1mW, 200Kbps peering or hub-and-spoke internet gateways
  – RS-485 wired LAN: 1Mbps multidrop. Ethernet consumes too much power, Cat-5 cable is too expensive
  – Easy transition between “connected” or “detached” modes
Software Applications

• Software Footprint: < 1MB!!!!
• Web browser, full TCP/IP software suite, NFS, web server, etc.
• Native or cross-compile -- either way a full software development environment: vi, emacs, C and Scheme
• MP3 audio and IP telephony support in hardware
• Browse or navigate with onboard buttons and wheel; write code with external keyboard/mouse
Hardware Reuse

- Cheap embedded controller for tasks bigger than a PIC can handle
- Embedded routers, wireless gateways, data loggers, process controls, info displays, toasters, thermostats, robotic pets, …
- Everybody owns all software IP through GNU Public License (hardware is ML licensed)
- Design is scalable through Motorola ColdFire 50MIPS solutions or Verilog 68K models
PDA System Architecture

• Four reusable main modules:
  – Processor core containing Motorola Dragonball, 8MB flash, 8MB DRAM, running Linux
  – Mothercard containing audio DSP, 200Kbit/sec TDMA/FDMA data radio, backplane bus controller, and up to 64MB removable flash
  – Removable user interface board
  – Docking station containing battery charger, serial port, keyboard port, RS-485 multidrop network
Pengachu PDA block diagram

- LCD 128x64 or 320x240
- Dragonball CPU
- 8M Flash
- 8M DRAM
- 900 MHz Radio (100Kbit/s)
- MultiMedia Card
- Docking Connector
- Power Supplies +3.3, -15 from 2 AAA NiMH
- System MCU (PIC16LF877)
- Input Hardware
- MPEG Audio DSP
- SPI Bus
- 3
- 16
- 8
Pengachu PDA Unit

One week before Spring TTT

As seen at Spring TTT
Pengachu Server Block Diagram

- **LCD 320x240**
- **Dragonball CPU**
- **8M Flash**
- **8M DRAM**
- **900 MHz Radio (100Kbit/s)**
- **Xilinx XCS30 30K-gate FPGA**
- **10-BT Ethernet Controller**
- **Power Supplies +3.3, -15 from 5V-12V In**
- **IDE/Compact Flash**
- **Arbitrarily Complicated Custom Stuff**
- **Reconfigurable User Port (40 I/O Pins)**
Pengachu Server Board
Pengachu Inside

Quantum Computer Module with Pengachu Interface
(Yael and JasonT)
Free IP resources for reconfigurable systems

http://www.media.mit.edu/~rehmi/freeip.html
Our Little Skunkworks
(or how to do this without Neil knowing)

- 2 Month Crash Hardware Development
- Trilateral Org Chart with Equal Contributors
- Top-down mechanical/electronics integration
- Use of rapid prototyping as appropriate