Lecture 1
Introduction

Pervasive Computing
MIT 6.883 & SMA 5508

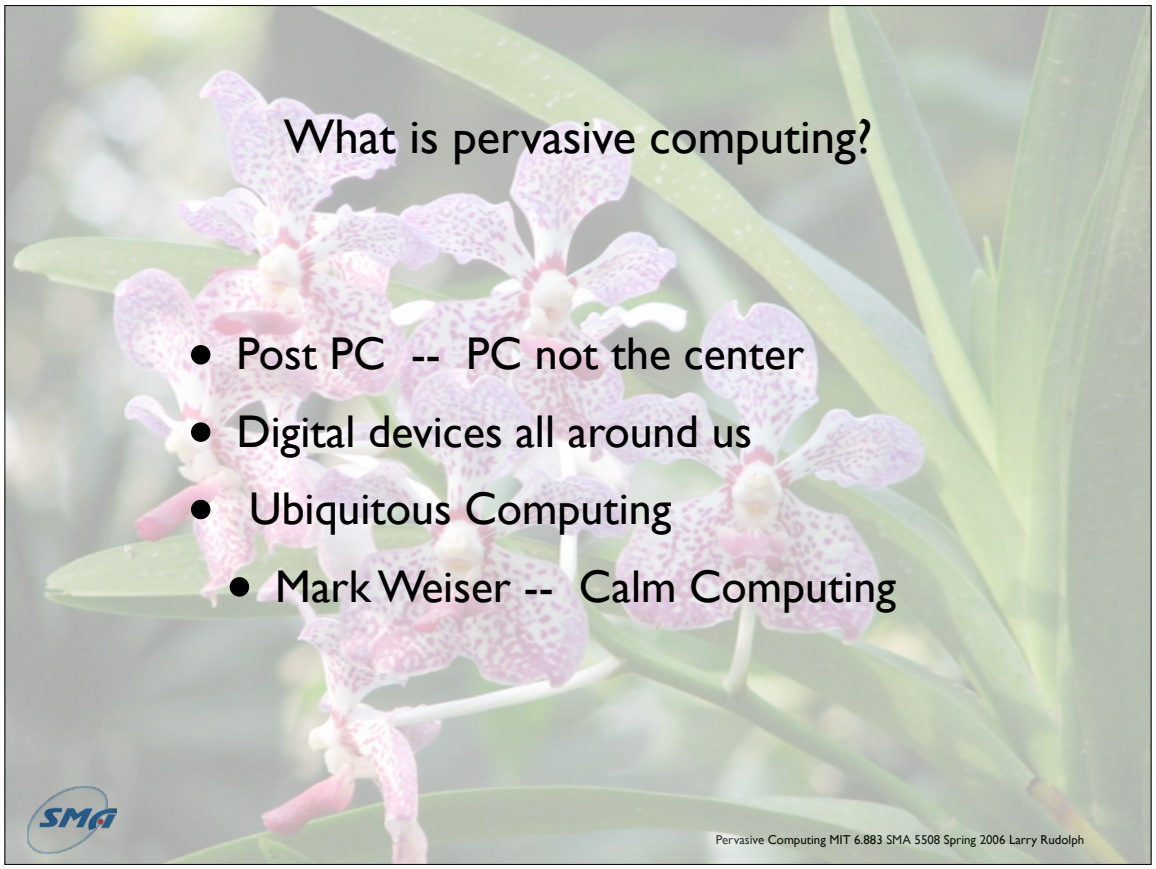
Larry Rudolph (MIT)
Tat Jen Cham (NUT)

Course Structure Overview

- Two sets of students -- MIT and SMA
- Two parts to class
  - Individual technologies (PS & Quiz)
  - Group projects
- Materials
  - iPaq & backpaq, Cricket, Nokia Series 60 Phone
  - Slides, handouts, notes (raw)
  - Readings
  - TA's -- filter for your questions
    - Ning Song (nsong@mit.edu), ???@ SMA

Administration

- Official Web Site and Wiki
  - http://people.csail.mit.edu/rudolph/Teaching/home883.html
  - http://org.csail.mit.edu/mode



What is pervasive computing?

- Post PC -- PC not the center
- Digital devices all around us
- Ubiquitous Computing
  - Mark Weiser -- Calm Computing

The origin of the course: Project Oxygen

To bring an abundance of computation & communication
within easy reach of humans
through natural perceptual interfaces of speech and vision
so computation blends into peoples' lives
enabling them to easily do tasks they want to do:
- ***collaborate,***
- ***access knowledge,***
- ***automate routine tasks***

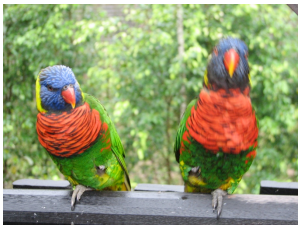
MIT PROJECT OXYGEN
PERVASIVE, HUMAN-CENTERED COMPUTING

CSAIL

Pervasive, Human-Centric Computing

What do these words mean?

- Computers are already pervasive
  - even in Boston and Singapore
- Computers are already human-centric
  - are they for the birds?
- It's not really about computing
  - we already know how to do that



Massachusetts Institute of Technology

So, what do we mean?

- Pervasive
  - Should be where we need them
    - not have to go to them or set them up
- Human-centric
  - Computers should adapt to humans
    - computation enters our world/environment
- Computing
  - Computer-mediated function
    - digital media

Look back to see ahead

- Monolithic Programs & Hardware
- Decompose into interactive pieces
  - Compose to build large thing
- Continue decomposing into autonomous, interacting components

Finding and naming stuff

- Few items
  - Use list
- Many items
  - Use heirarchy
- Very many items
  - Use multi-index

Organization of material

- Top-down
  - would be nice to start writing apps
  - but we are not there yet
- Bottom-up
  - Build on what is known
    - Keyboard, mouse, pen
    - Location, Speech, Multimodal
    - Integrative Technologies

H21 components

- Hardware
  - iPAQ
  - Backpaq
  - Wireless Communication

Software

– Linux
– Landcam
– Galaxy Audio Interface
– Cricket Location Reporting

iPAQ 3870

3870 iPAQ

– 206 MHz Strong Arm
– 64 Mbytes SDRAM
– 32 Mbytes flash storage
– Bluetooth
– SD/MMC card slot
– 16 bit color display

5500 iPAQ

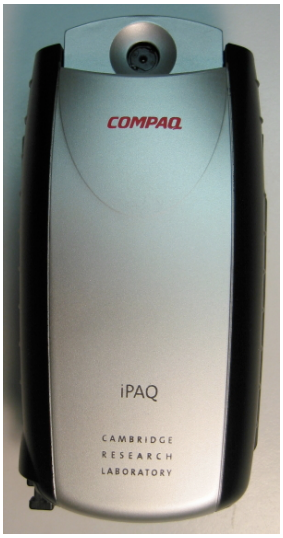
– 400 MHz Xscale
– 128 Mbytes SDRAM
–48 Mbytes flash storage
– Bluetooth & WiFi
– SD/MMC card slot
– 16 bit color display

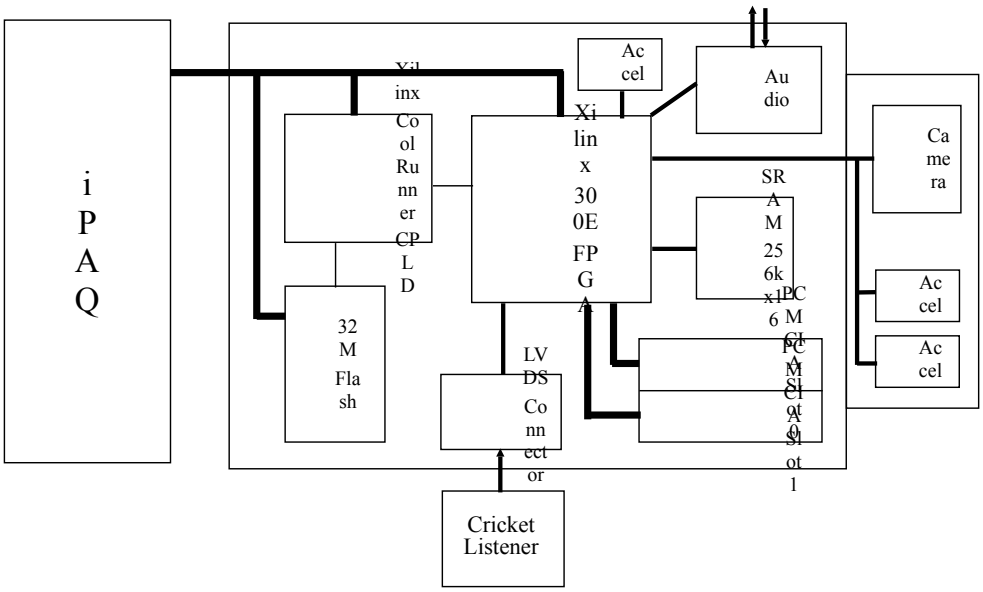
Linux on H21

- Why Linux?
  - Linux allows full access to all software
  - Common development with desktop
  - Can use open source code from many sources
- Porting Linux to a handheld device
  - More difficult than standard PC or Laptop
    - Non-standard interfaces (screen, control FPGAs, touch screen, …)
    - Requires rewritable Flash ROMs
    - For iPaq, port done by HP's Cambridge Research Lab

H21 Backpaq

- Redesigned BackPaq (Version 3)
  - Philips imager (640x480 CMOS color imager UPA1022)
  - Larger FPGA (Xilinx Virtex 300E)
  - 256k x 16 SRAM
  - Lower power
  - 3-axis accelerometer in camera housing
  - 2-axis accelerometer in Backpaq
  - Dual PCMCIA slots
  - Audio input/output codec and headset jack
  - 32 MBytes Flash in Backpaq
  - LVDS Connector from FPGA pins
  - Lion Battery





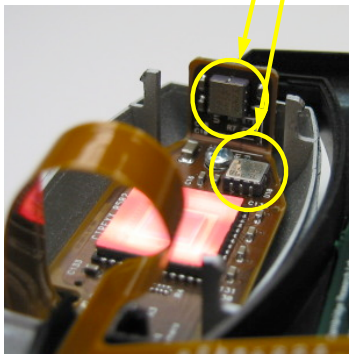
- Philips UPA1022 Imager
  - 640x480 CMOS
  - Improved image processing

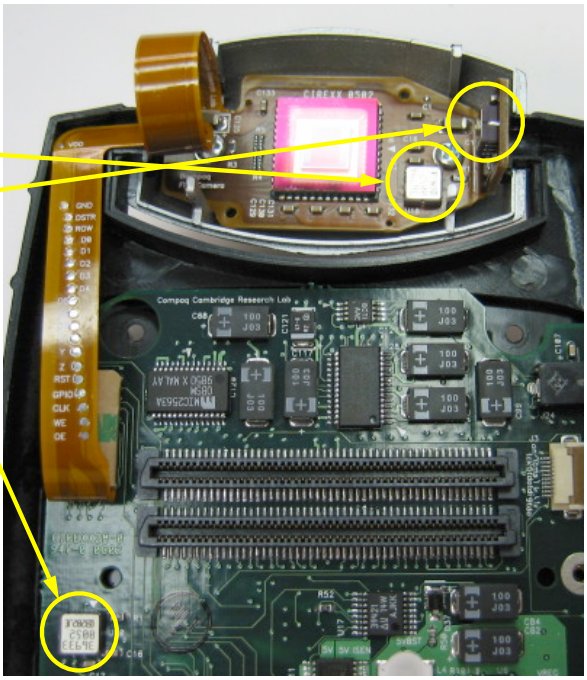
Three Accelerometers:

One in camera plane

One perpendicular

One on Backpaq PCB





Accelerometer Linux Devices

- 2-Axis accelerometer (on main PCB)
  - `/dev/backpaq/accel`
  - Each read returns X and Y acceleration values
- 2-Axis accelerometer (on camera PCB)
  - `/dev/backpaq/cam_accel`
  - Each read returns X and Y acceleration values
- 3-Axis accelerometer (in camera housing)
  - `/dev/backpaq/cam_accel_xyz`
  - Created from 2 perpendicular 2-axis accelerometers
  - Each read returns X,Y and Z acceleration values

CSAIL

Reading Accelerometers

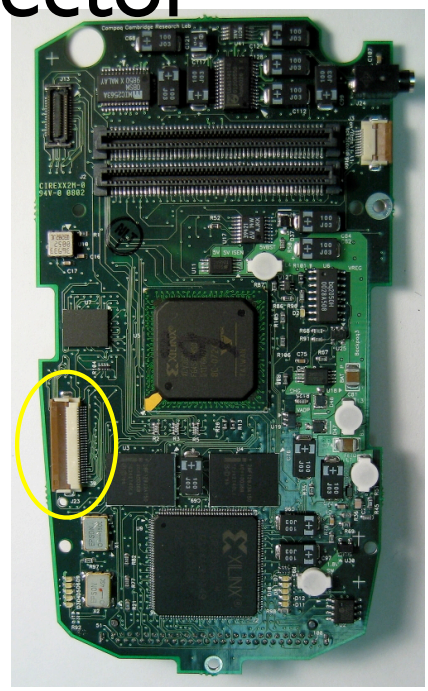
- Linux character device
- Open the device:
  - `fd = open("/dev/backpaq/accel",O_RDONLY | O_NOCTTY);`
- Read from the device
  - `struct h3600_backpaq_accel_data accel_buffer;`
  - `read(fd,&accel_buffer,sizeof(accel_buffer));`
- The structure returned:
  - `struct h3600_backpaq_accel_data {`
    - `short x_acceleration;`
    - `short y_acceleration;`
- From:
  - http://cvs.handhelds.org/cgi-bin/viewcvs.cgi/apps/backpaq/oneko/

PCMCIA Slots

- Dual PCMCIA Slots
- 16-bit cards (Not CardBus) supported
- Pins driven directly from the FPGA
  - Hardware supports CardBus cards
  - Could implement CardBus controller in FPGA

LVDS Connector

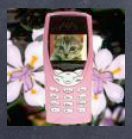
- Low Voltage Differential Signaling
  - 10 differential signal pairs
    - **or 20 single signals**
  - Driven from Xilinx 300E FPGA
  - 5 Power and 5 ground pins
- 30 pin flex cable ZIF connector



Massachusetts Institute of Technology

Mobile Phones
What's the big deal







- < 200 Million PC's sold last year
- > 200 Million Phones sold last quarter
- .5 Billion PC's in 2003
- 1.5 Billion consumers own mobile phones worldwide -- Economist, Jan 2006
- 3 Billion subscribers by 2008

CSAIL

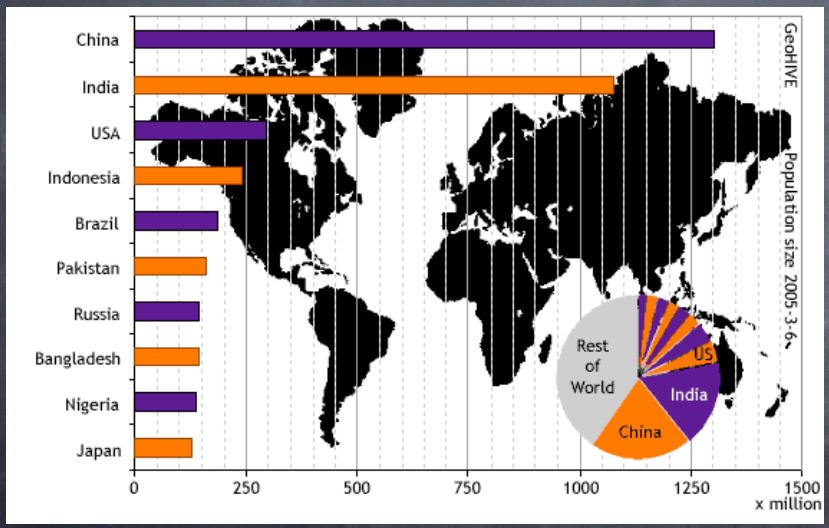




CSAIL

- September 18, 2005 -- 2 Billion connections.

Perspective



CSAIL

- 6.4 Billion people
- 2 Billion mobile phones sold

OK, so lots of phones ....

- But there are lots of digital watches as well
  - they have chips inside, but who cares?
- Today, there are
  - Basic phones (modem chip)
  - Regular phones (modem + microprocessor)
  - Smart phones (modem + micro + ...)
- Tomorrow, will all be smart, difference in
  - extra features
  - extra fashion

CSAIL

Smartphones == 1996 PC?

- Smartphones (and PDA's) are like old PC's
- If they are the same, then
  "been there, done that"
- If they are different, then in what ways?

CSAIL

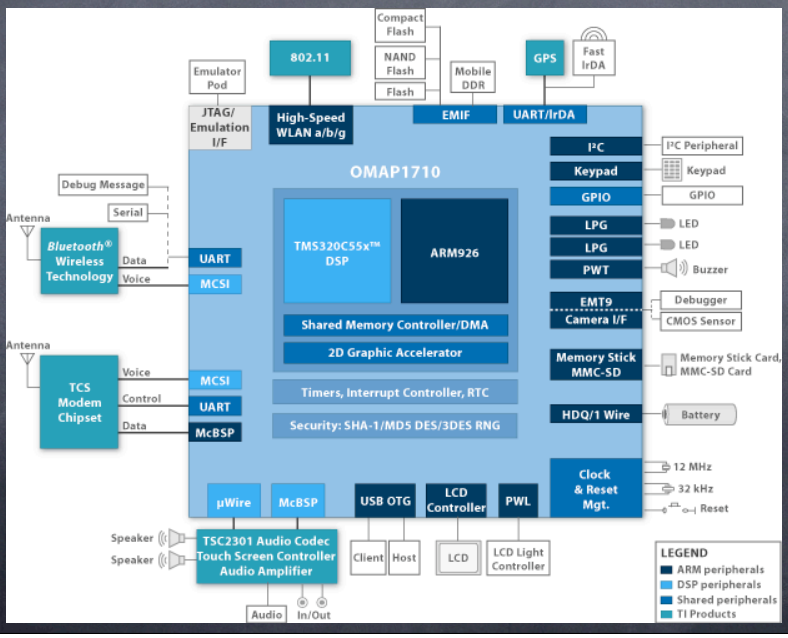
1996 Pentinum

- 200 MHz CPU; 60 MHz memory bus
- Floating point; expansion bus for
  - graphics, sound, other accelerators
- 3 million transistors; Voltage 3.3
- Primary Cache: 8 KB; Level 2: 512 KB
- Memory: usual ??? MB; Max 4 GB
- Disk capacity: ??? find out 160 MB ???

Phone's two major cores

- DSP Core
  - 220 MHz
  - 64 KB on-chip Ram; 24 KB Instr. Cache
  - 1/2 instructions per cycle
- ARM Core
  - 229 MHz
  - 32 KB Data Cache; 16 KB Instr. Cache

Phone == Lots of Integration



CSAIL

Not really the same

- More connectivity
- More parallelism
- More advanced in
  - Hardware features
  - Software features & necessities
- More sophisticated expectations
  - cannot turn back time; people have evolved

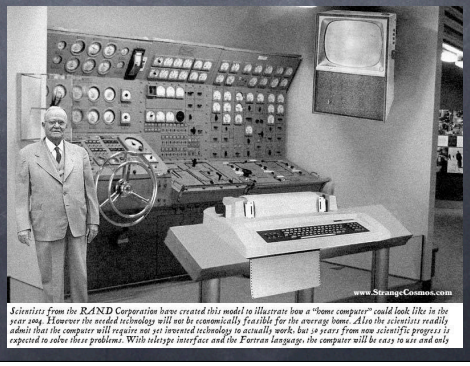
CSAIL

Phones are different

- They are mobile
- They will always be bounded by power
- They will follow a different Mores' law
- The economics are different
  - different producer-consumer relationship
  - hw --> operators --> end users
  - ISP, independent software vendors, role?

The Point?

- Phones are different from PC's
  - Claim: people want PC functionality
  - They do not want the PC's overhead
- There will be billions of smart phones
- Time to start taking up the challenge!



Research Areas I

- User Interface (Huge)
  - Configuration
  - Syntax-free
  - Accessibility: physical & mental disabilities
- Security, Reliability, Fault Tolerance
  - Naive users; harsh physical world
- Synchronization & Sharing
  - Interoperability (no platform)

Research Areas II

- Architecture:
  - Phone chips as building blocks
    - wireless expansion bus (no other board)
  - Power & heat management
    - e.g. streaming video via DSP or ARM
    - local vs remote compute & store
  - No H/W upgrades

Research Areas III

- Applications
  - Services not applications; easier on user
  - Finding features (e.g. 287 menu items)
  - Platform independence (?)
    - same app for server; pc; phone
    - too many models (binary rewrite?)
- (location, user, env)-aware computing
- Phone as Sensor+Actuator Server
- Phone as (out-of-band) debugger

Conclusion

- Whatever your expertise, phones offer
  - different set of constraints
  - different levels of abstractions

If you think technology is frustrating today, just wait...