Prototyping Physical UIs

(Physical & Tangible User Interfaces)

Tutorial @ Mensch&Computer 2003

Stuttgart, September 2003

Albrecht Schmidt, LMU München

Christian Decker, Uni Karlsruhe

Why Prototyping?

Prototypes are essential to learn and understand and experience interaction in ubiquitous computing

From the idea to knowledge

- Prototyping has been central to hallmark research in the area (e.g. ParcTab, ActiveBadge)
- Learning occurs when along the prototyping process as well as in use

Towards a Methodology

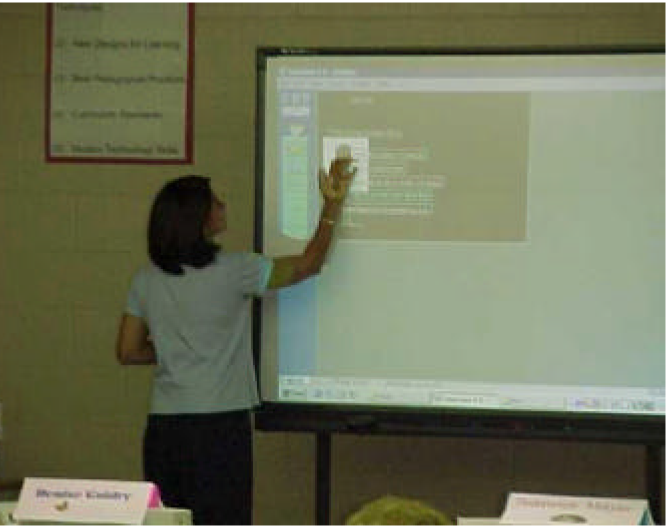
- Analysing artefacts and how they are used
- Prototyping context-aware artefacts (recording issues in the process)
- “Confronting” **real** people with these enhanced artefacts (version 0.001)
- Deployment in a living lab environment
- Facilitating everyday environments with real users

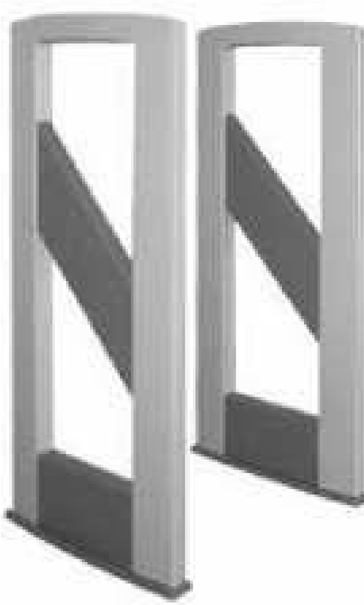
Evaluation

- Prototypes are means for evaluation
  - if you don’t have it you can’t tell whether or not it works!

Platforms for physical UIs

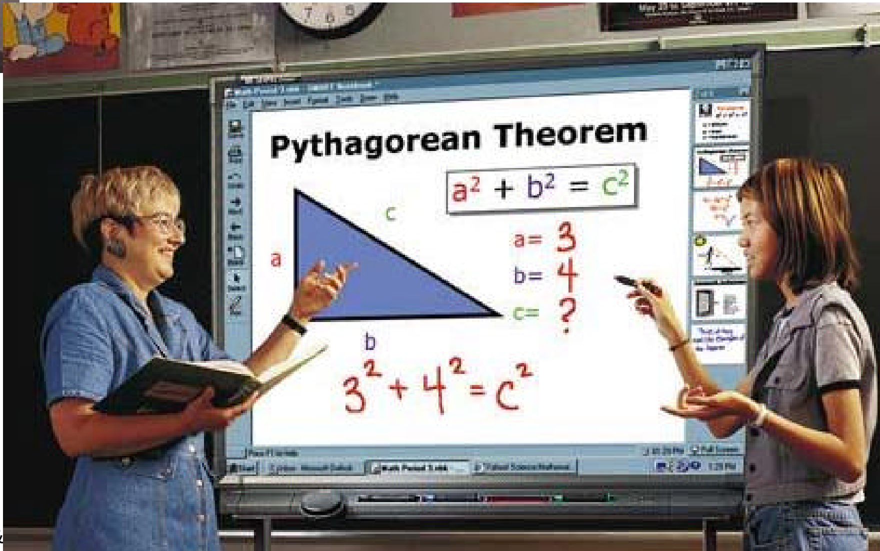
- PC with additional interface devices



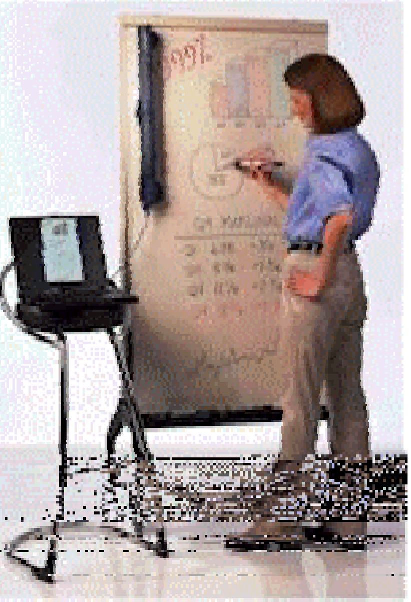


Smart-Board

- Large touch sensitive surface
- Front or back projection
- Interactive screen



Capture Interaction



- Mimio
  - Tracking of flip chart makers
  - Capture writing and drawaing on a large scale

- PC Notes Taker
  - Capture drawing and handwriting on small scale

- Basic Technology for other applications???

Platforms for physical UIs

PDAs & Mobile Phones

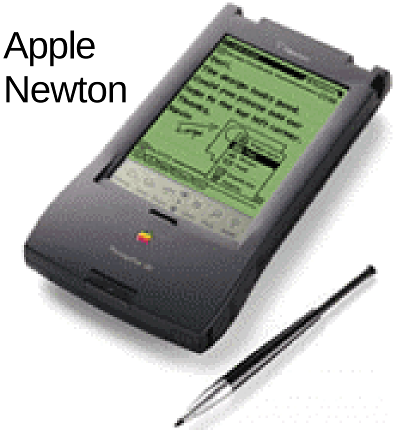
- (Touch) Screen
  - Varies in size
  - Color or gray
- Keypad/Stylus input
  - Reduced keyboard or soft-keyboard
  - Handwriting recognition, letter recognition, abstract writing
- Built-in communication capabilities
  - Serial cable connection
  - IrDA, Bluetooth, GSM...

- Add further hardware in slots
  - GPS
  - Compass
  - RFID-Reader

HP I Paq

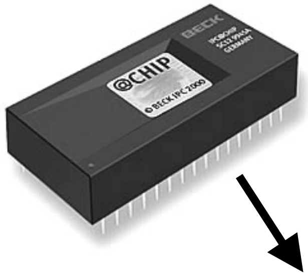
Xerox PARC Tab

Apple Newton



Platforms for physical UIs Embedded Systems

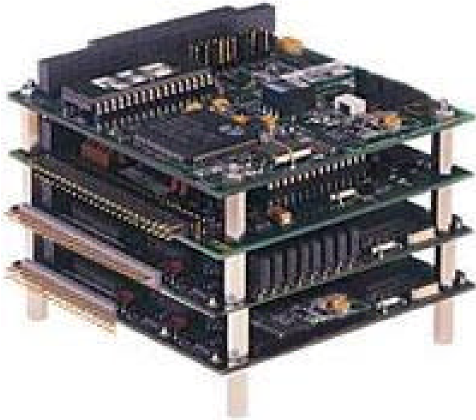
- Beck IPC@Chip
  - Single chip webserver with http, ftp, telnet, PPP server, 2 serial ports, 1 Ethernet port, 8 digital I/O pins
  - 68 mm x 61 mm
  - AMD186 core, programmable in C (or Pascal) with standard Borland C/C++ compilers
  - Applications:
    - Home automation (temperature, light sensing, motor control...)
    - Webcam controller


SC12 (single chip webserver) by beck-ipc.com

DK40 (evaluation modul) by beck-ipc.com

Platforms for physical UIs Embedded Systems

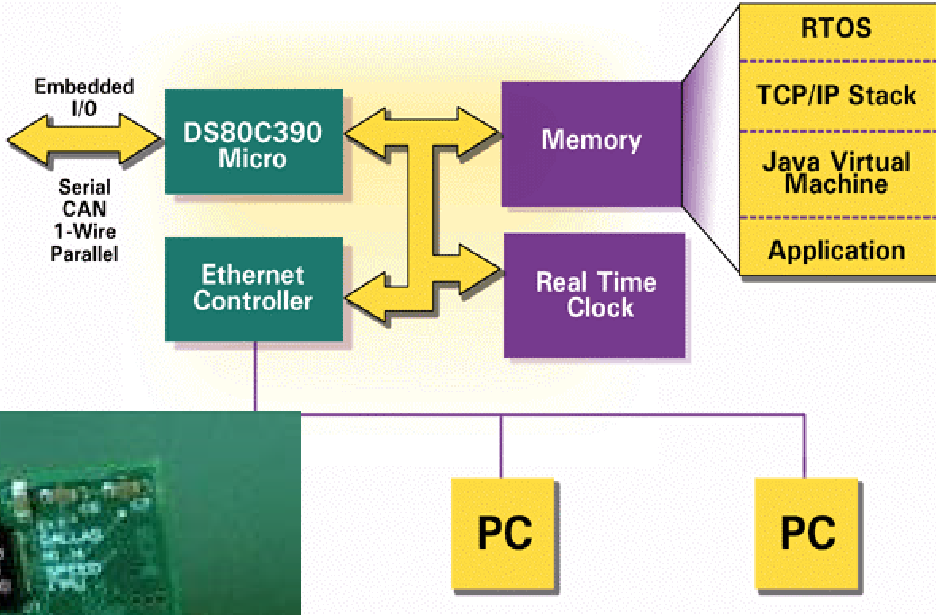
- PC104
  - Stackable PC-compatible modules creating an embedded computing system (4“x4“ per module)
  - Various modules, e.g. CPUs including everything from 8088 up to Pentium, Modems, Sound and speech I/O, Motion control(servos), Video frame grabbers, DSPs, GPS, Touch screen interfaces, etc.
  - Applications:
    - Controller in automation and industry

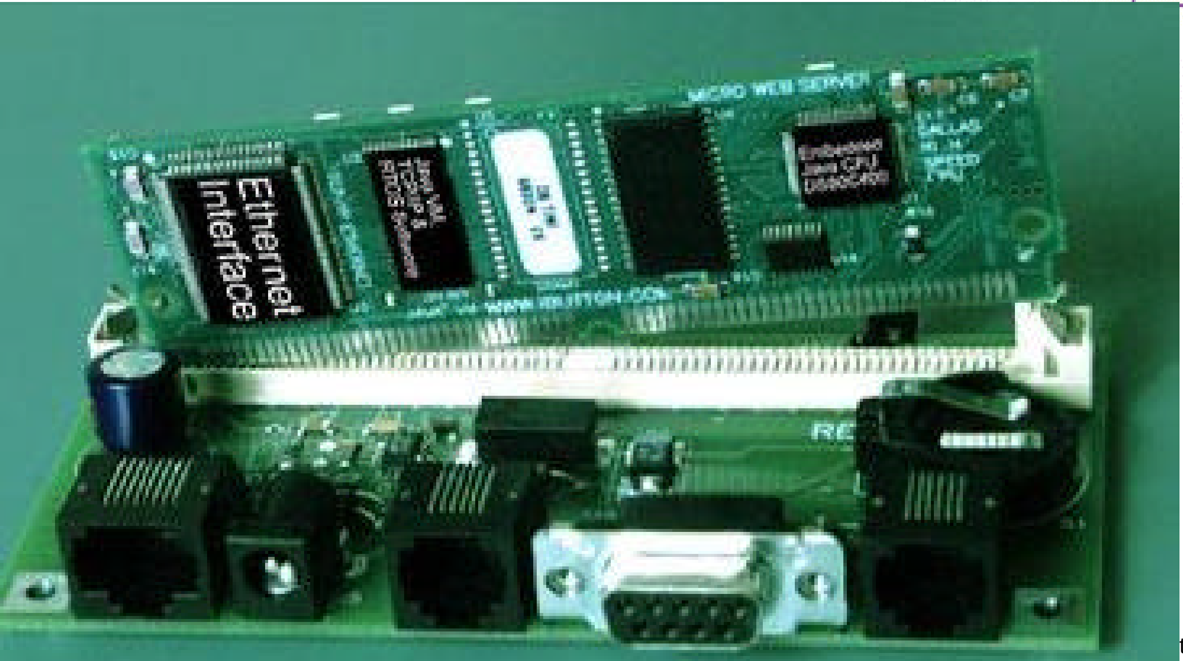


PC104 stack by pc104.com

Platforms for physical UIs Embedded Systems

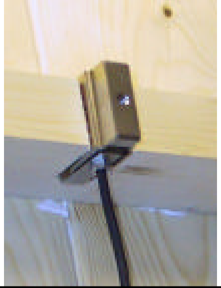
- Tini Board (http://www.ibutton.com/TINI/)

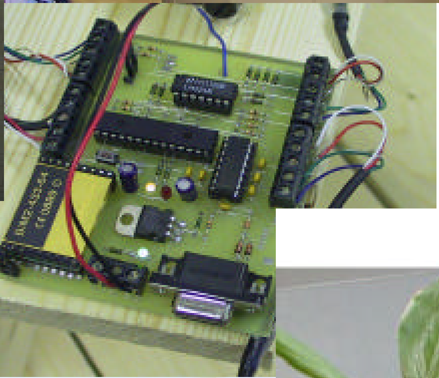


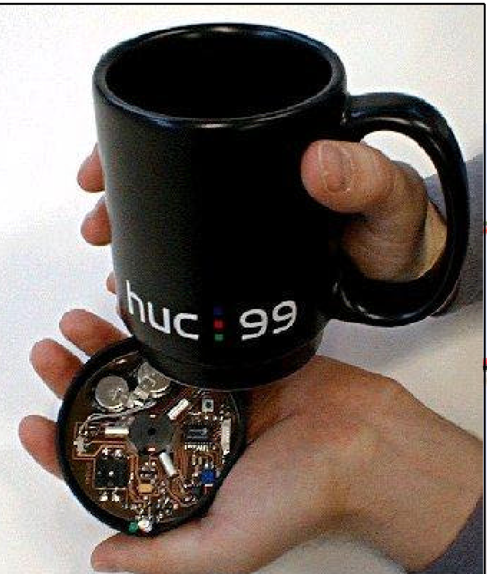


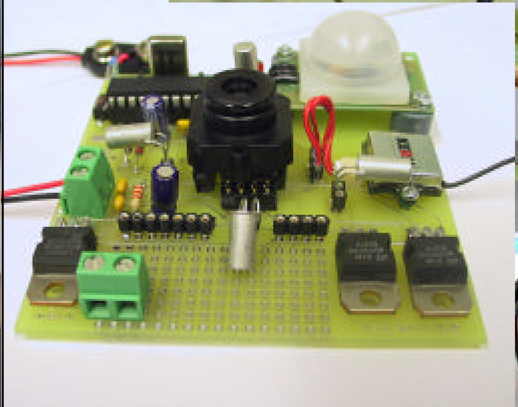
t

Prototypes…







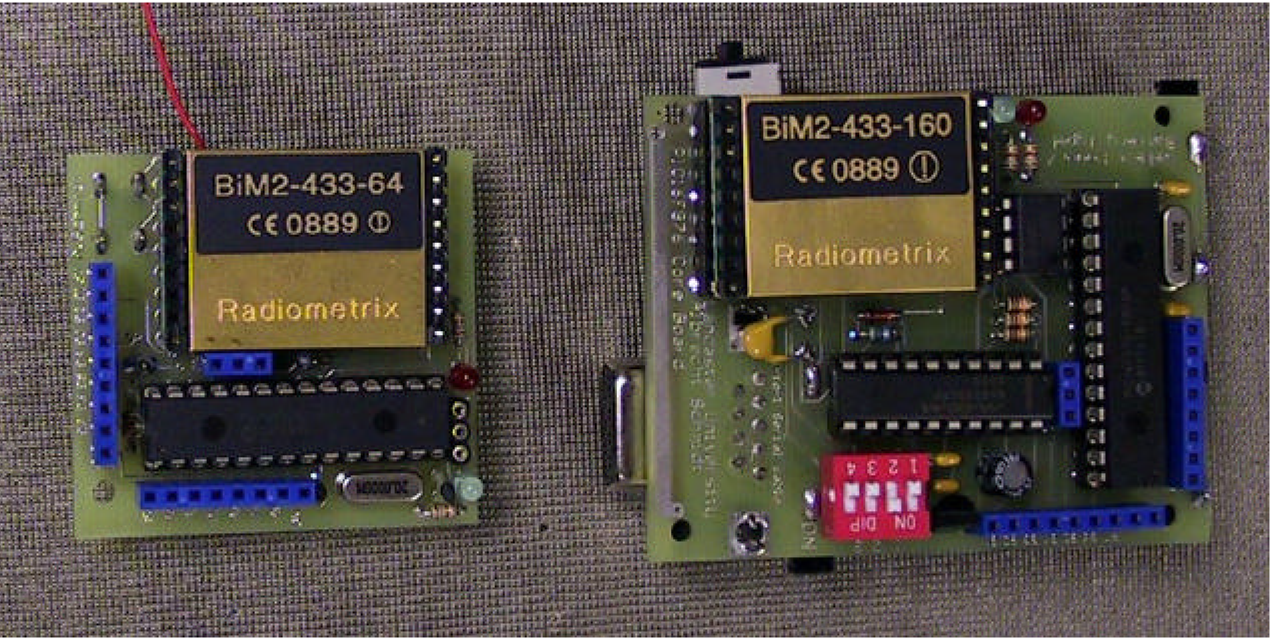


Smart-Its – A new Computing Platform YAP - Yet another platform?

Means for exploring applications

- Building scenarios
  - Rapid-prototyping of context-aware computing applications
  - Assessing the potential as an enabling technology for ubiquitous computing in various application domains
- Why a new computing platform?
  - Investigating the difference between Smart-Its and an iPAQs with Bluetooth and a sensor board.
  - Price, size and power consumption matters now – even if the future brings it anyway!
- Understanding and refining the requirements

Smart-Its Platform



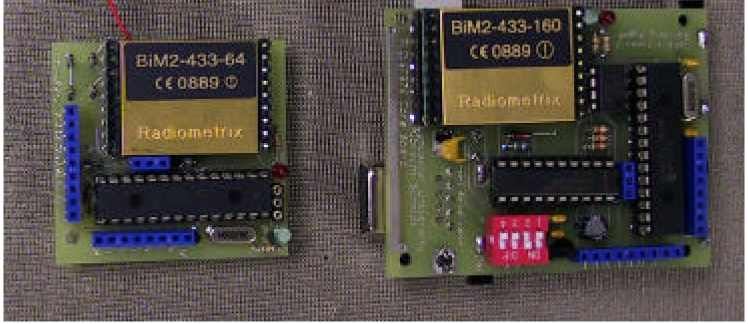
All base boards

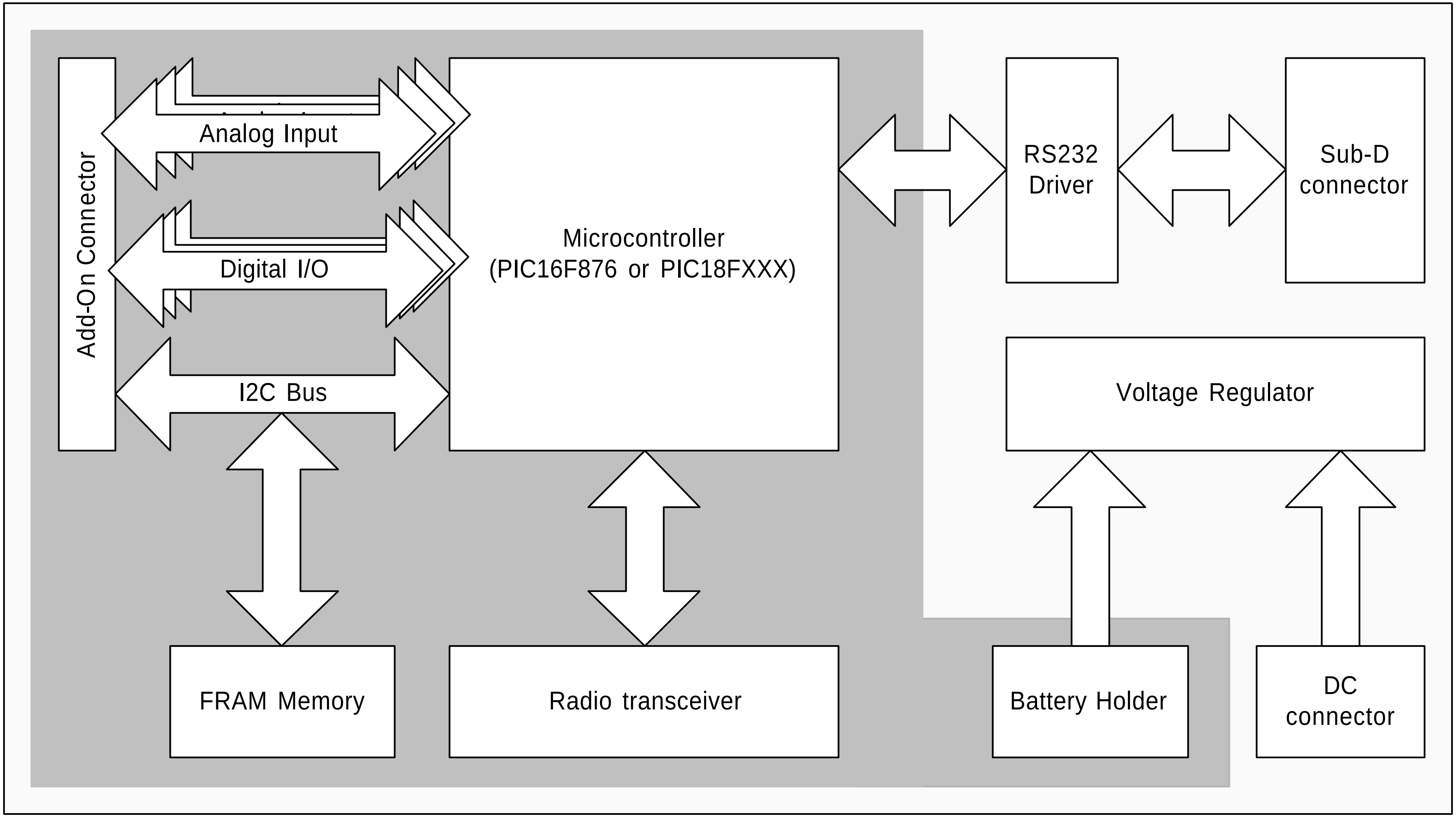
- Microcontroller
- RAM
- Analog Inputs
- Digital I/O
- Wireless communication
- All boards are software and hardware compatible

- Small portable unit
  - 45mm x 50mm x 19mm
  - 29g with battery

- Base station and debug unit
  - 55mm x 70mm x 29mm
  - 110g with 4x AAA
  - RS232 connector
  - DC Power Connector

Hardware
Base Board Basics





Add-On Connector

Analog Input

Digital I/O

I2C Bus

Microcontroller
(PIC16F876 or PIC18FXXX)

RS232
Driver

Sub-D
connector

Voltage Regulator

FRAM Memory

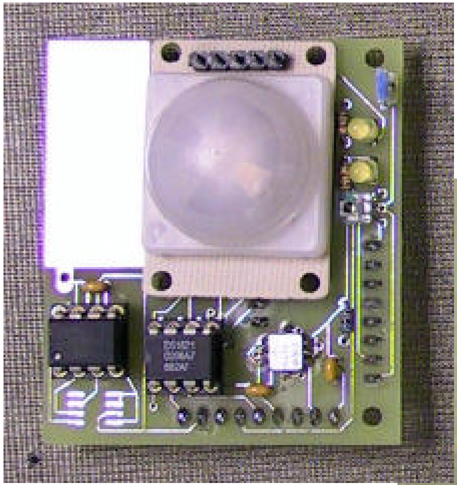
Radio transceiver

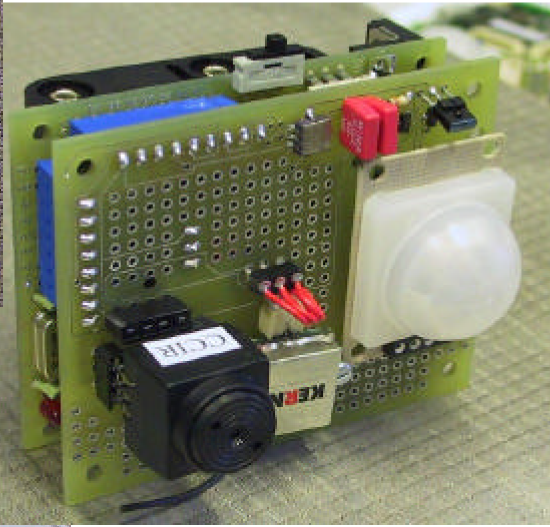
Battery Holder

DC
connector

New Sensor boards
Add-Ons to the core smart-It

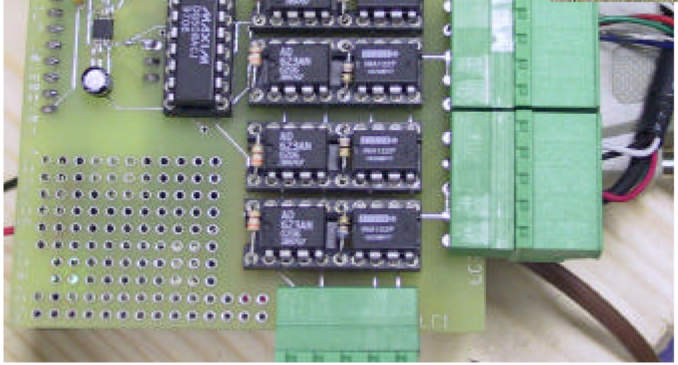
- Hardware
  - Much simpler
- Software
  - Build upon frameworks
- Communication
  - Basic functions available



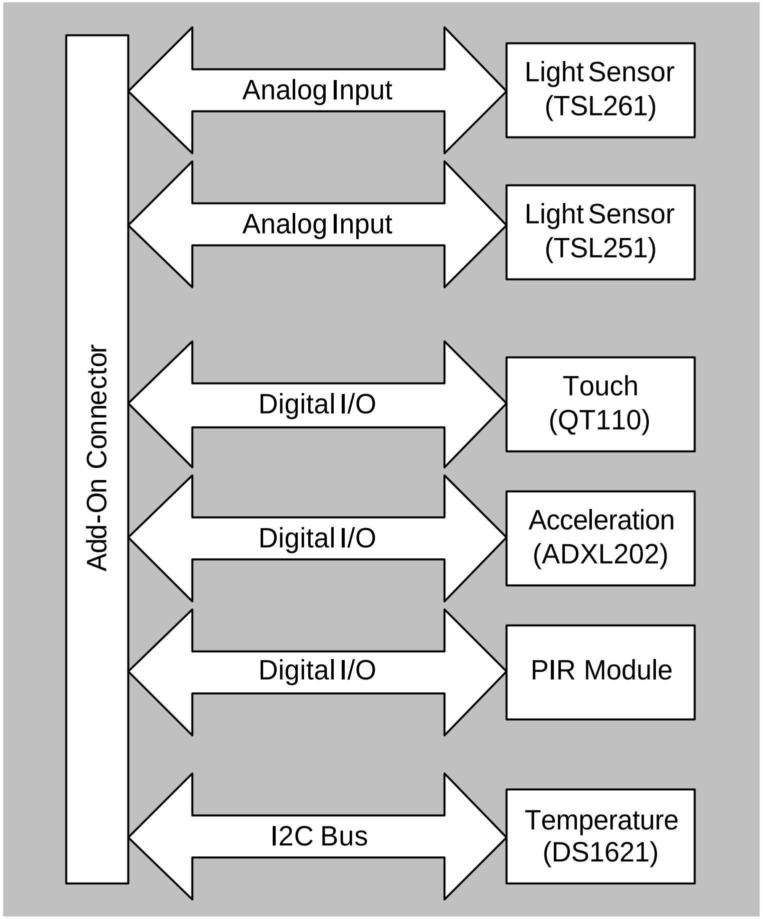


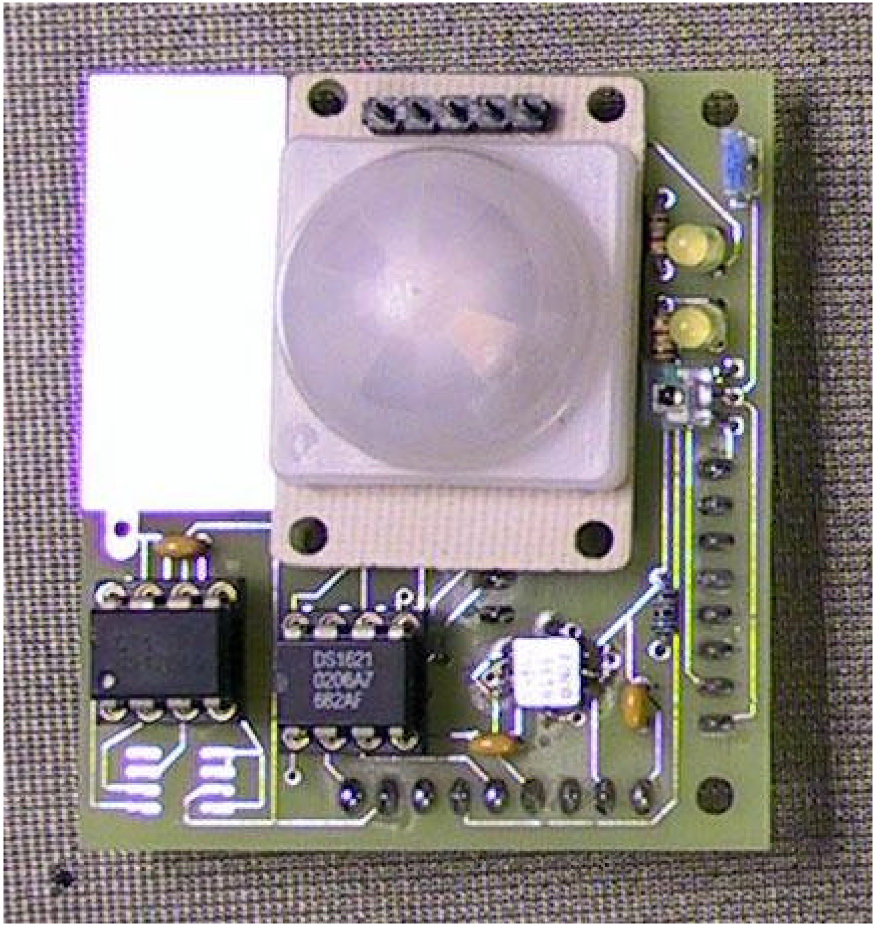
Examples

- General sensors
- Vision / Camera
- Load sensing
- Weather board
- Motion sensing
- Actuator boards
- …

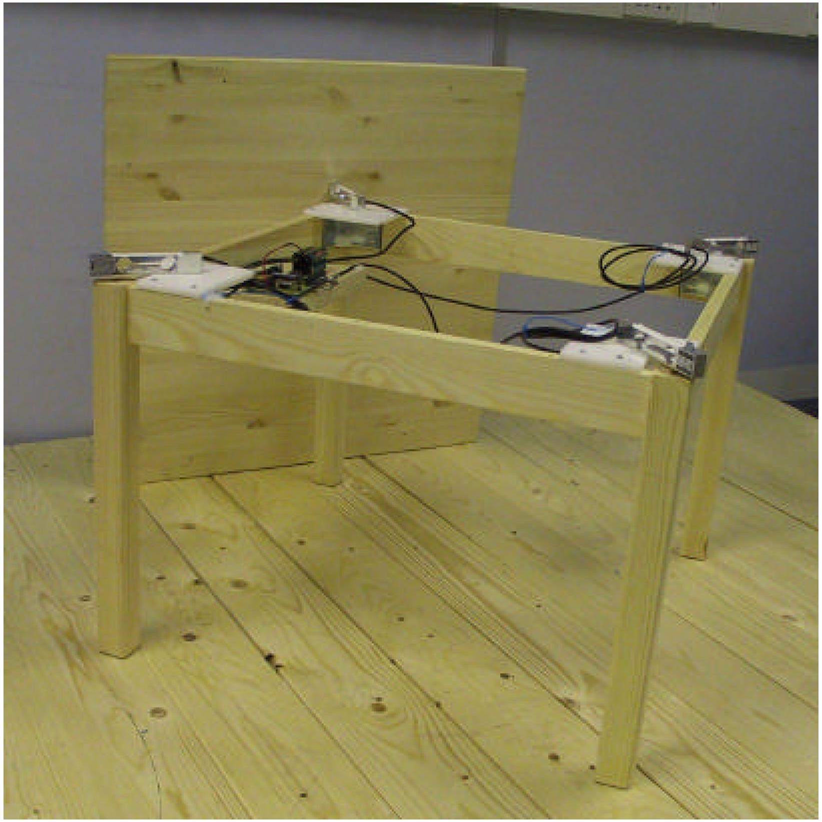


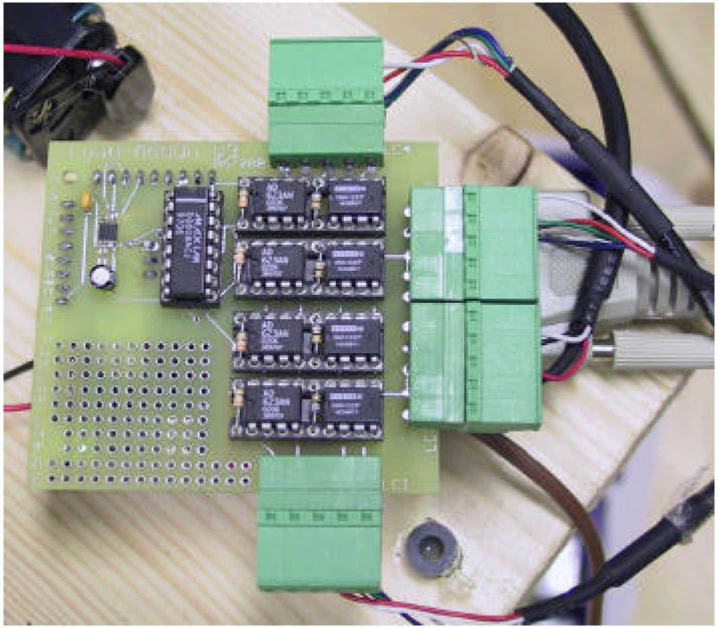
Hardware
Sensor Board Basics



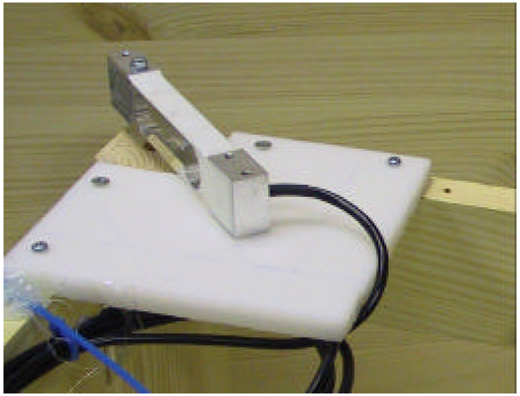


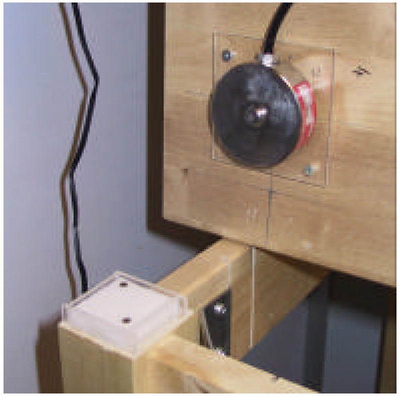
Example I: Table as a Sensors



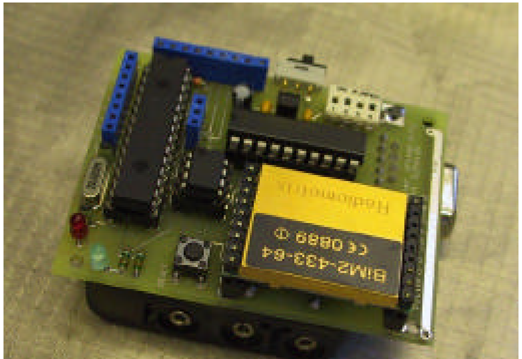


- Smart-Its sensor AddOn board
- 16 Bit DA
- Instrumentation Amps

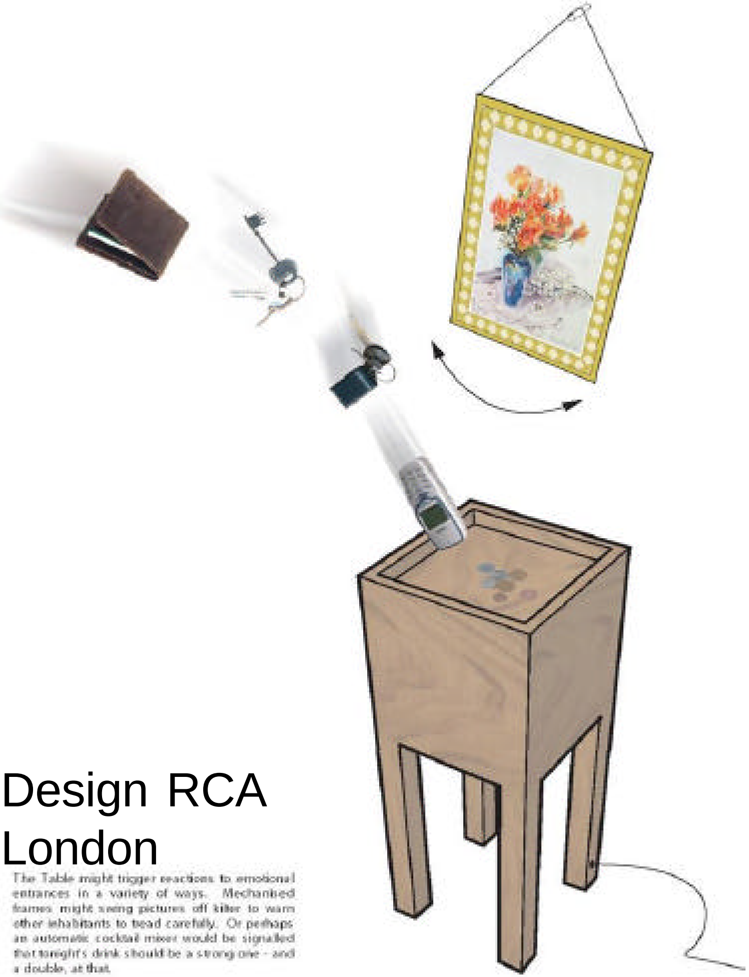




Computer 2003, Stuttgart

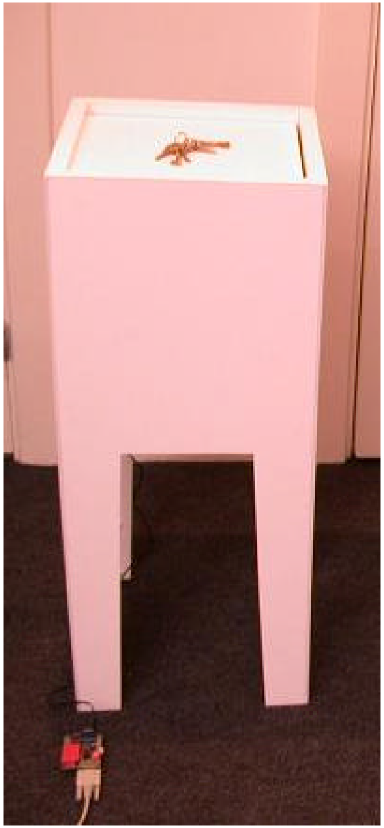


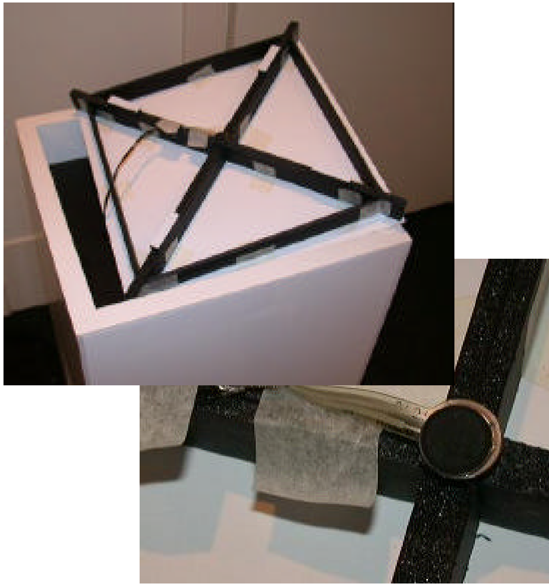
Example II: another table as a sensor and a picture frame as actuator

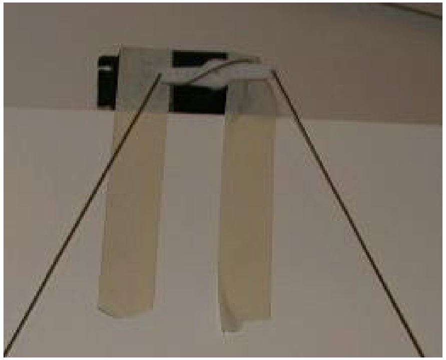


Design RCA London

The Table might trigger reactions to emotional entrances in a variety of ways. Mechanised frames might swing pictures off kilter to warn other inhabitants to tread carefully. Or perhaps an automatic cocktail mixer would be signalled that tonight's drink should be a strong one - and a double, at that.







Implementation

- Smart-Its sensor AddOn board
- simple pressure sensor

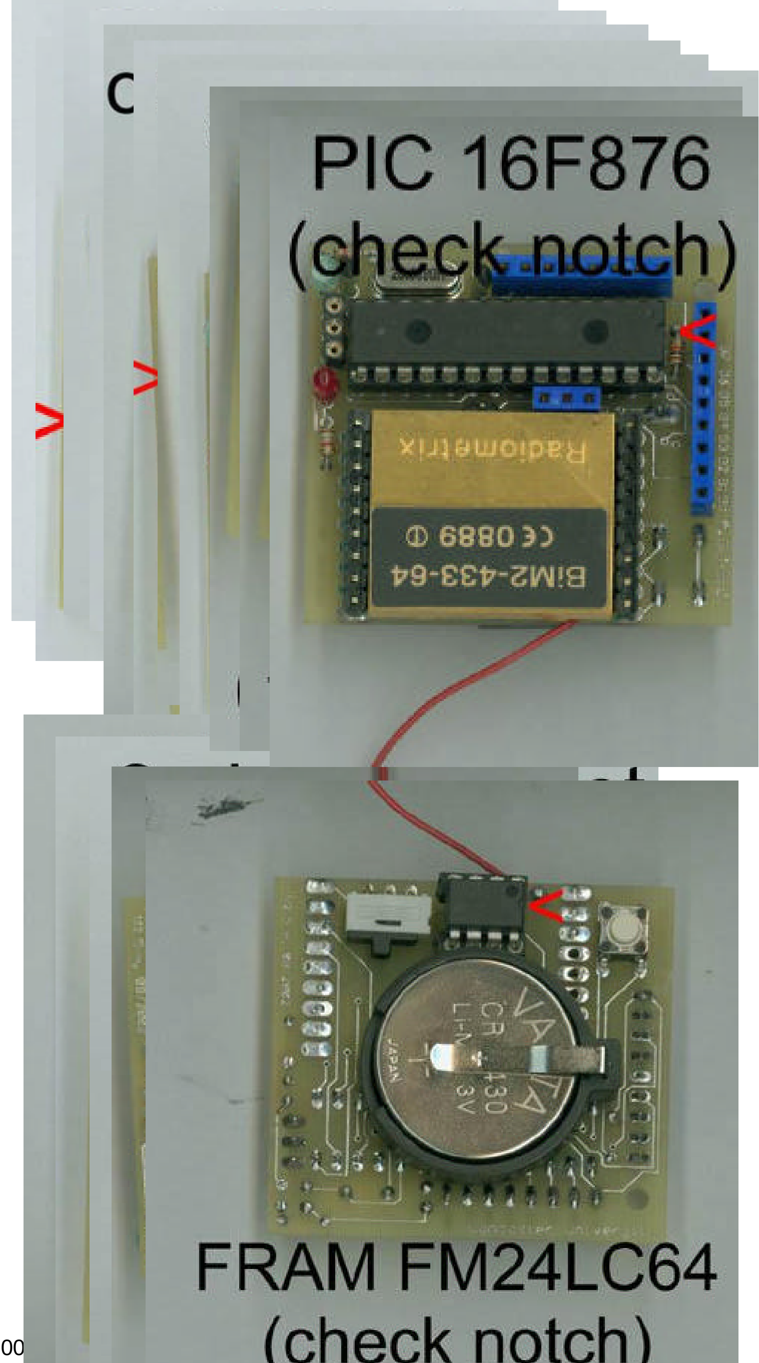
Hardware DIY Approach

Selectedrequirements

- Understandable with a CS background
- Minimal electronics skill
- It is a tool
- Similar to electronic kits
- Easing embedding of sensors and actuators
- Reusable
- Basic hardware and software should run within a week for most scenarios

Beyond rapid prototypes

- All components as CAD unit
- Different physical shapes (of the same HW) are easy to do
- Software compatible to modules

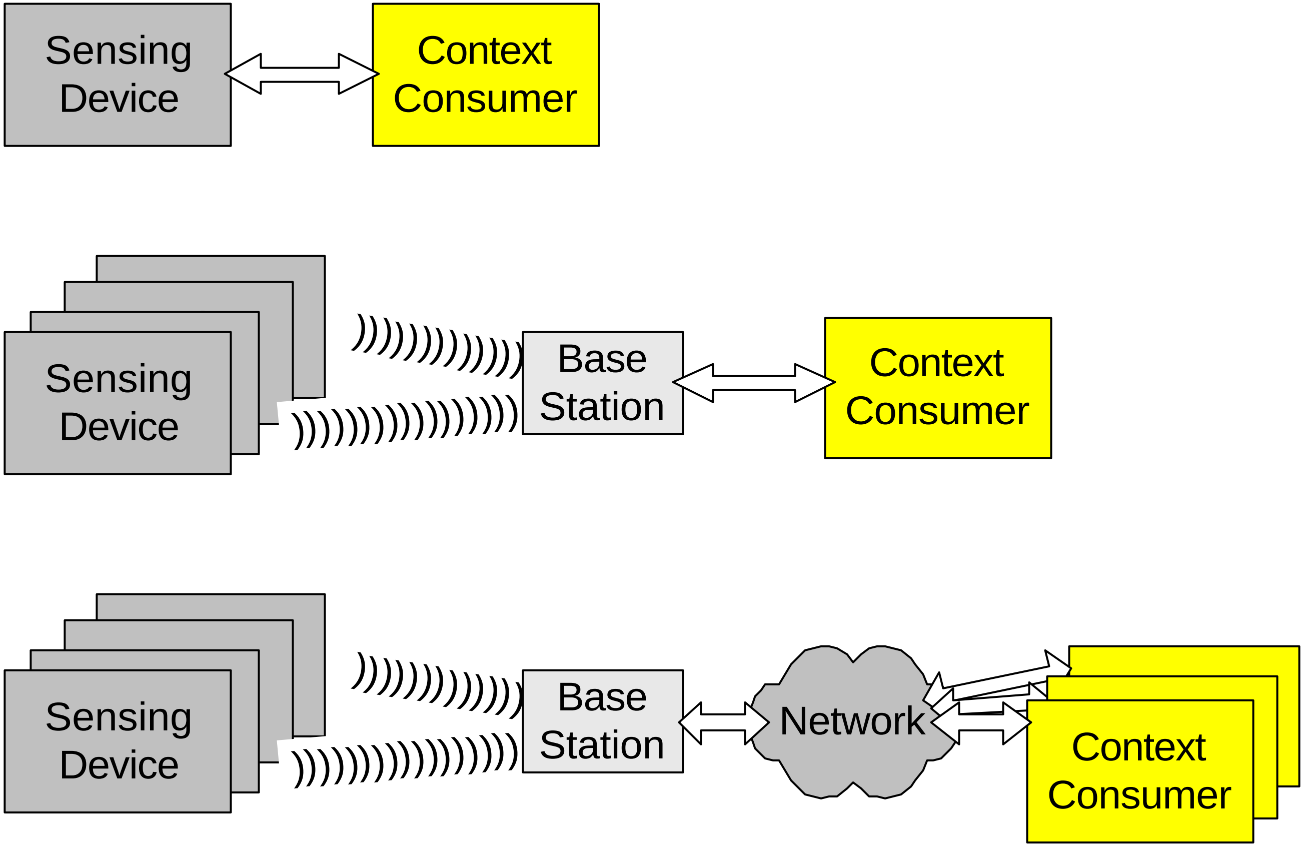


PIC 16F876 (check notch)

FRAM FM24LC64 (check notch)

Software
Context Acquisition Systems

Architectures & Software Frameworks



Context Acquisition Library Structure

Category	***Sub Categories***	***Implementation***
Architectural Frameworks	▪Attached sensing architecture. ▪Wireless single consumer architecture. ▪General wireless sensing architecture	System architectures
Hardware Library	▪Processing cores and memory units. ▪Sensor blocks ▪Communication blocks ▪Power supply blocks.	EAGLE CAD files
Software Library	▪Program Templates	Program skeletons in C and function in PIC-C
	▪Sensor drivers ▪Communication drivers ▪Timer	Drivers implemented in functions (PIC-C)
Perception Library	▪Statistical functions ▪Time domain analyis	Function in PIC-C
Backend Library	▪Serial line access ▪Network access	Variety of skeletons and functions/classes in Java, C/C++, and Visual Basic for Linux and Win32.

Platform Evaluation
Prototyping Exercise I

Evaluation Method

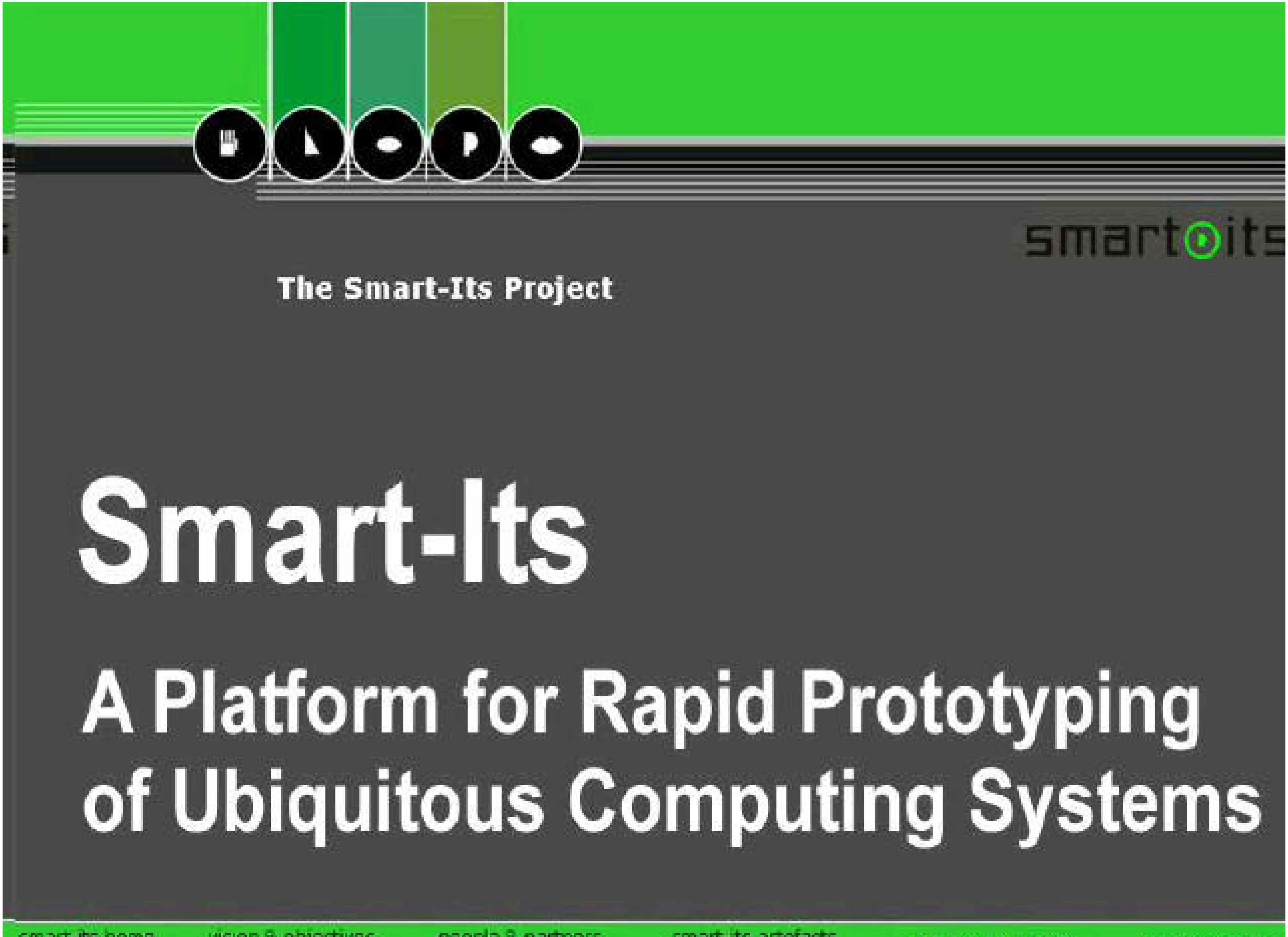
- Developers Workshop (DC Atelier)
- 2,5 days hands-on

Results

- Prototypes & Demonstrators
  - Smart Ball
  - Wireless Gesture Remote Control
  - Singing Smart-It
  - Wireless RFID Sensor
  - Wireless Gesture Joystick

- Value of implementation

Prototyping Exercise - Impressions



The Smart-Its Project

Smart-Its

A Platform for Rapid Prototyping of Ubiquitous Computing Systems

Hardware Tutorial – very brief :)

Microcontroller PIC16F876

- only page 1 of the datasheet…

Serial line driver MAX233

- Serial line on the PIC is TTL (0 to 5V) on the PC it is -12V to 12V. This chip does the translation.

FRAM FM24C64

- Ferro-electric memory, fast, keeps its content when power is switched off, connected via I2C, “Harddrive”

BIM2 Radio Transceiver

- Up to 160KBits/s in half duplex mode, either transmitter or receiver or off, serial data stream should have equal numbers of 0 and 1, packet done in software

78L05 Voltage regulator

- Regulate input voltage (6V…18V) to 5V

Software Basics

Templates

- Base station (e.g. receiv1.c)
  - Basic receiver
  - Gateways to the PC
  - Foundation for actuator add-ons
- Sensor node (e.g. node1.c)
  - Basic sender
  - Foundation for sensor baords

Drivers

- For modules or add-ons
- Implement access to sensors/actuators

Backend

- Basically reading from serial line
- Examples in java, C/C++, and Visual Basic
- For Linux and Win

Compiler for MCU

Compiler CCS

- You have to know the compiler – can be tricky :)
- See www.ccsinfo.com for the newsgroups
- Programming is often dirty (e.g. global variables, …)

No in-circuit programming

- Change configuration
  (by physically moving processor and FRAM)
- Debugging is easier

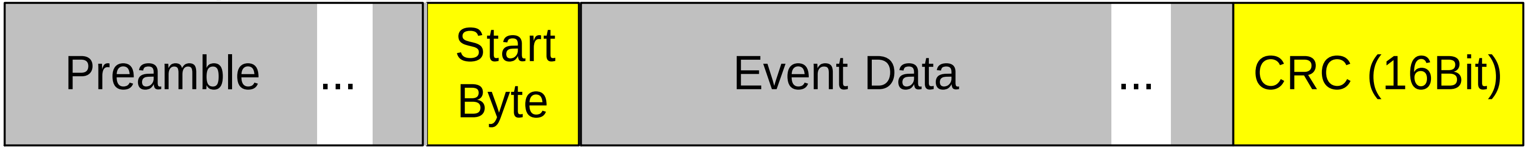
Software - Files

core.c

- Defines according to schematic,
- I2C initialization
- funtion to control LED

bim2rf.c

- Simple RF protocol implementation, e.g.
- void reset_rf_buffer() // clear buffer, use before printf
- void to_rf_buffer(char c) // as first argument in printf
- void RF_printf() // print the buffer over RF



fr24c64.c

- functions to use the FRAM chip
- Read and write to memory

16F876.H

- Standard h-file for the PIC used in the core board
- Provided with the compiler

Next Steps

Software

- More examples & bugfixes
- Port to PIC18F252 (double ram & double flash ram)
- chip configurable via terminal or wireless (standard tasks without programming)

Hardware

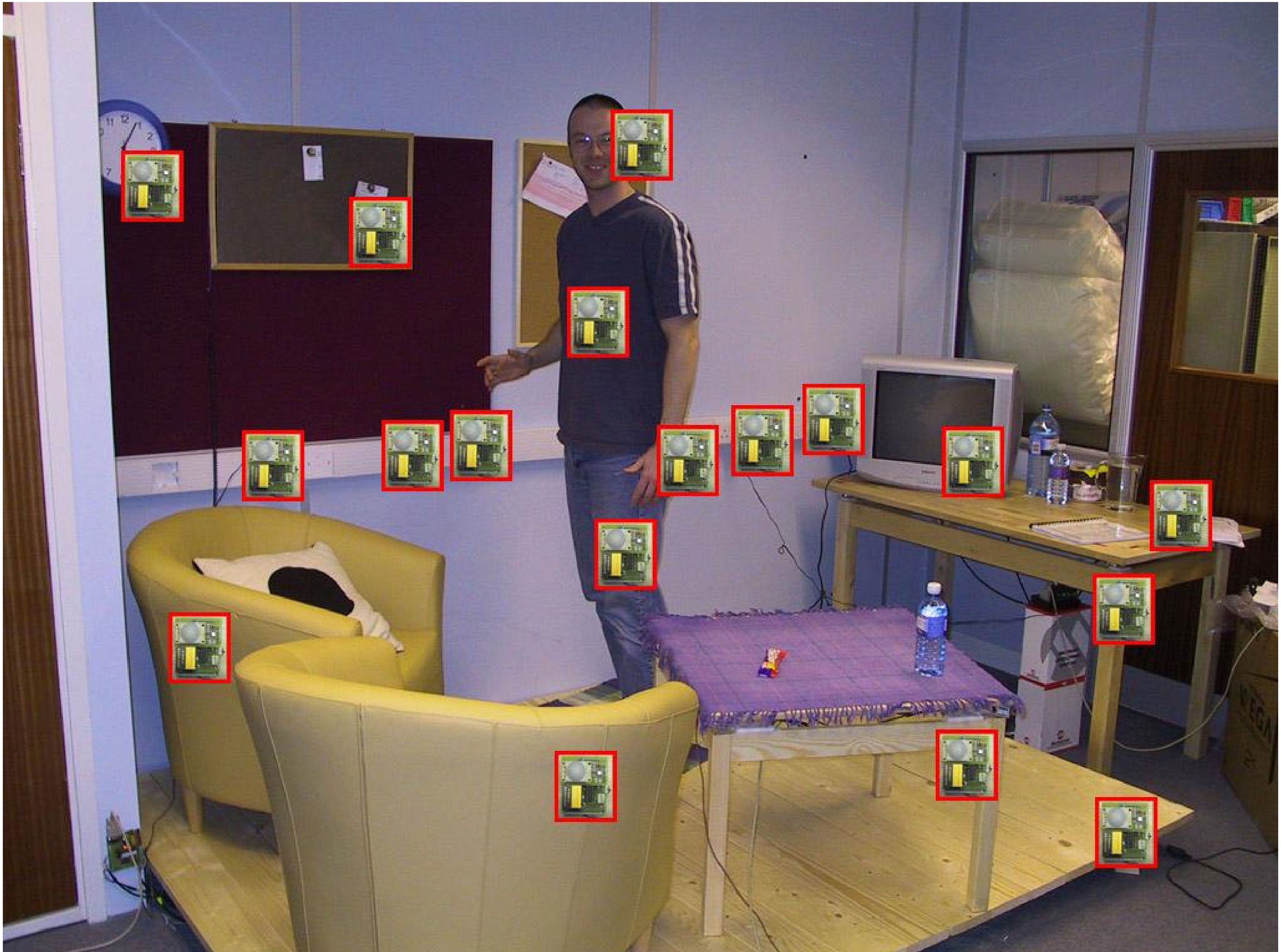
- Resisting to make it more complex
- Additional form factors
- More & new add-on boards
  - sensing
  - actuators
  - communication bridges
- CAD copy&paste library

Community

- Revise website (make it possible that people can comment on each page)
- Mailing list, tracking bugs, maintaining a public wish list
- Providing teaching materials
- Platform workshop

Prototyping in the lab

Prototyping in the lab



Evaluation of physical interfaces and new interaction methods

Evaluation = Assuring validity/quality of results

Evaluation Methods

- proof of concept (you can do it and its reproducible)
- User workshops and user feedback (formal and informal)
- Living lab and monitoring of usage
- Controlled studies

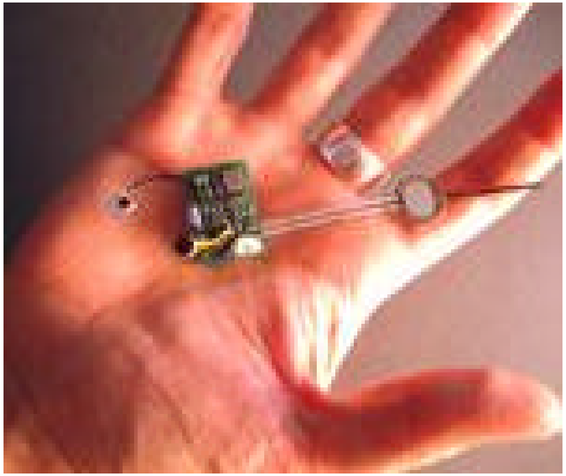
Difficulties

- Evaluation in context – in a real environment (no lab condition)
- Stability of prototypes
- Causality – many things are changed at once
- Goal is often beyond “being faster” but still relevant for productivity. What are we evaluating?
  - Pleasure?
  - Creativity & Inspiration?
  - Experience?

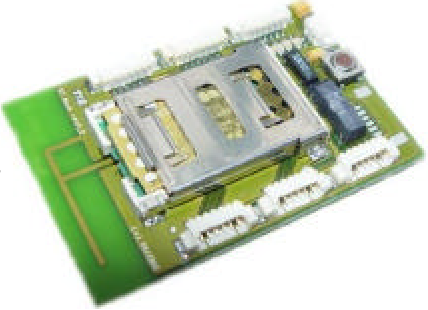
Platforms for Prototyping – Further routes in Smart-Its

- Smart-Its
  - Generic computing, sensing and communication platform
  - 2-folded: communication board, sensor board
  - Available sensors:
    - Acceleration
    - Light sensor (visible/infrared)
    - Force sensor
    - Temperature sensor
    - Audio sensor (microphone)
    - Spare space for additional sensors
  - Actuators:
    - Light (LED)
    - Audio (Piezo Speaker)

Smart-Its (TecO)

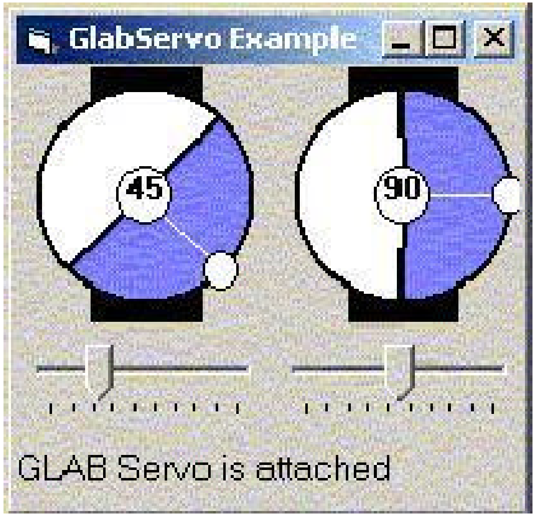


BTNode (ETH)

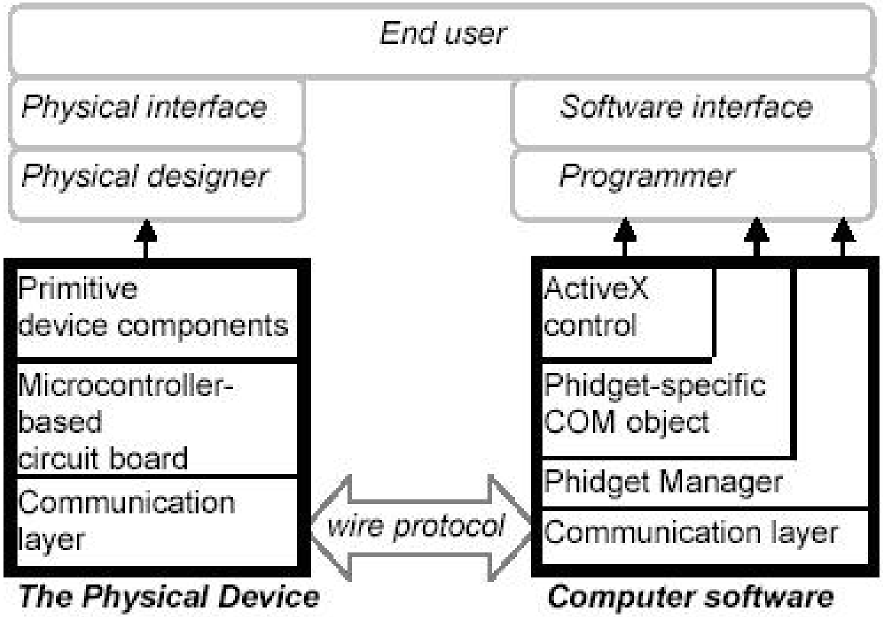


Phidgets

- Physical widgets (Greenberg, – a direct analog to graphical widget
- Aim to simplify and to speed up the development process for physical interface
- Component based approach in hardware and software
- ActiveX controls and COM objects for rapid development in VisualBasic



GLAB Servo is attached



End user

Physical interface

Physical designer

Software interface

Programmer

Primitive device components

Microcontroller-based circuit board

Communication layer

wire protocol

ActiveX control

Phidget-specific COM object

Phidget Manager

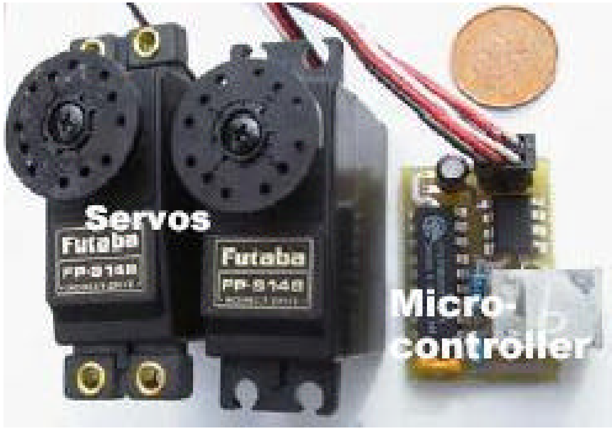
Communication layer

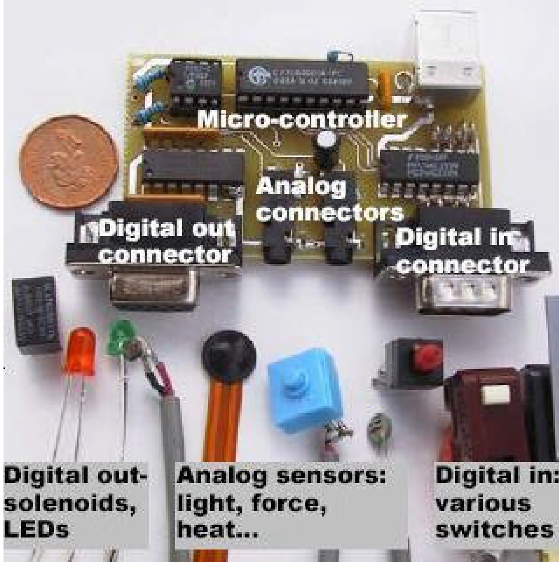
The Physical Device

Computer software

Phidgets – Devices

- GlabServo: lets a programmer control a device containing several servo motors. The position of each motor can be set programmatically
- GlabPowerBar: 120-volt power bar; a programmer can programmatically and rapidly turn on/off plugged devices
- GlabInterfaceKit: general-purpose ‘construction’ kit,
  - switches, LEDs, sensors, 8 digital input/output devices (e.g. various types of switches), heat, force and light sensors



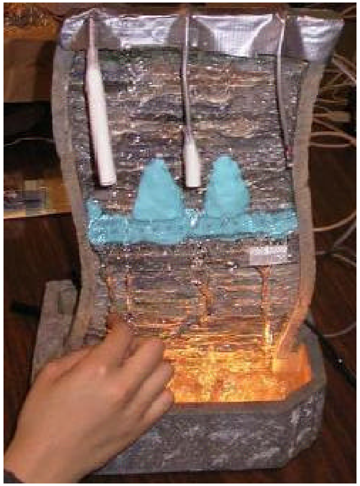

Micro-controller
Analog connectors
Digital out connector
Digital in connector
Digital out- solenoids, LEDs
Analog sensors: light, force, heat...
Digital in: various switches

Phidgets – Examples

Flower in Bloom

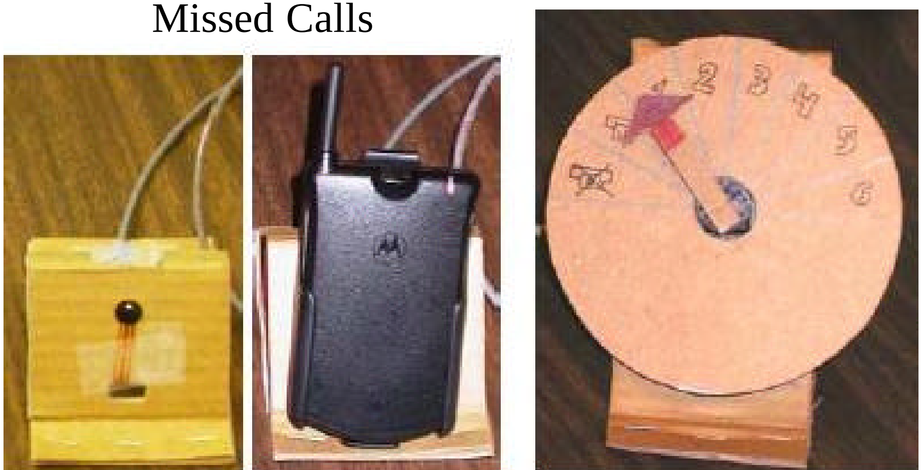
Power Lamp

Phidgets Eyes



Waterfall Harp

Missed Calls



Platforms for Prototyping – References

- PDAs
  - http://www.palm.com
  - http://www.dte.de (CompactFlash RFID Reader)
  - http://www.magellangps.com (GPS Receiver)
- Beck IPC@Chip
  - http://www.beck-ipc.com
- PC104
  - http://www.pc104.com
  - http://www.pc104.org (standardization group)
- Smart-Its
  - http://smart-its.teco.edu
  - http://www.inf.ethz.ch/vs/res/proj/smart-its/btnode.html
  - http://www.comp.lancs.ac.uk/~albrecht/smart-its/platform/
- Phidgets
  - Greenberg. S. and Fitchett, C. *Phidgets: Easy development of physical interfaces through physical widgets.* Proceedings of the ACM UIST 2001 Symposium on User Interface Software and Technology, November 11-14, Orlando, Florida. ACM Press. (http://www.cpsc.ucalgary.ca/grouplab/papers/)
  - Video: http://www.cpsc.ucalgary.ca/grouplab/phidgets/gallery/phidgets.UIST01.wmv