

# Instrumented Environments

Andreas Butz, [butz@ifi.lmu.de](mailto:butz@ifi.lmu.de), [www.mimuc.de](http://www.mimuc.de)



# Active artifacts (i.e. instrumented objects)

- Concept:
  - Determine activity where it occurs
  - Add “self perception” to everyday things
  - Communicate their own state
  - The artifact digitally “supports” its own applications
  - Example: MediaCup
    - <http://mediacup.teco.edu/>

# Mediacup (Teco, Univ. Karlsruhe)

- First experimental „active artifact“, 1999
- Technical Info:
  - PIC-Microcontroller, 15k/384Byte, low-energy
  - IrDA physical level communication
  - 3 acceleration, 1 weight, 1 temperature sensor
  - 2 condensers as power supply
- „Self perception“:
  - Reading out sensors periodically
  - Compute important events : in the shelf, full, empty, currently in use, etc...

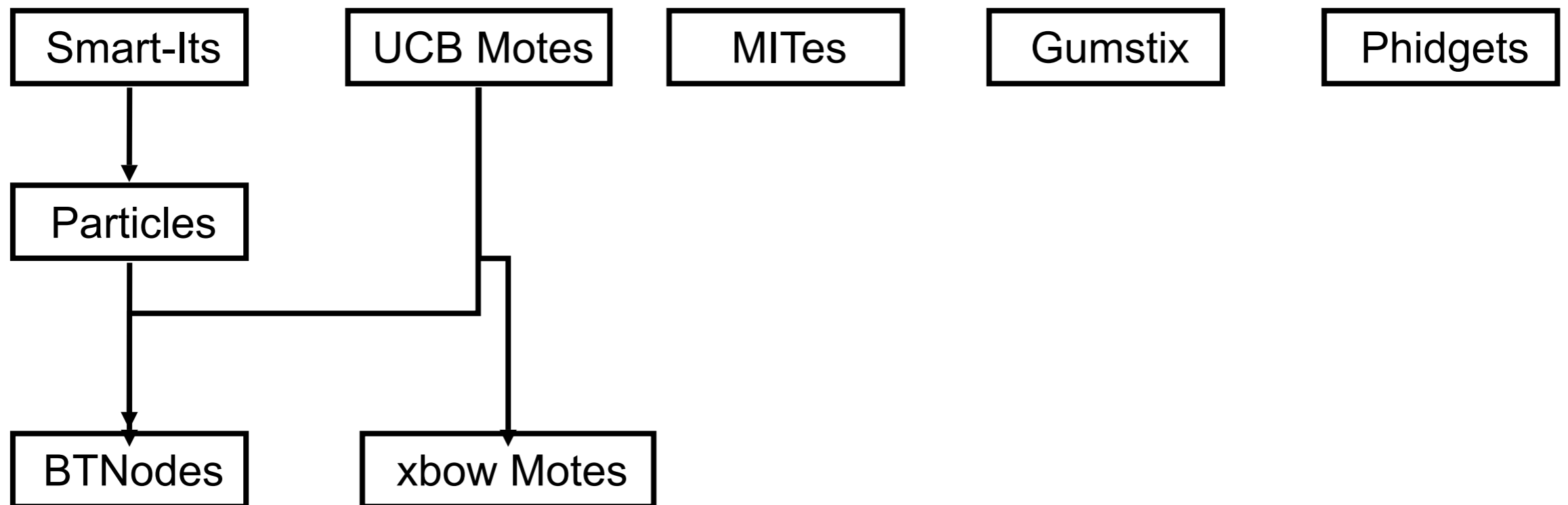
# Mediacup

- Small number of cups
- In use since 1999
- 95% correct recognition of **Multi sensor events**
- Important design criterion: **Energy consumption**, heavy influence on outward appearance



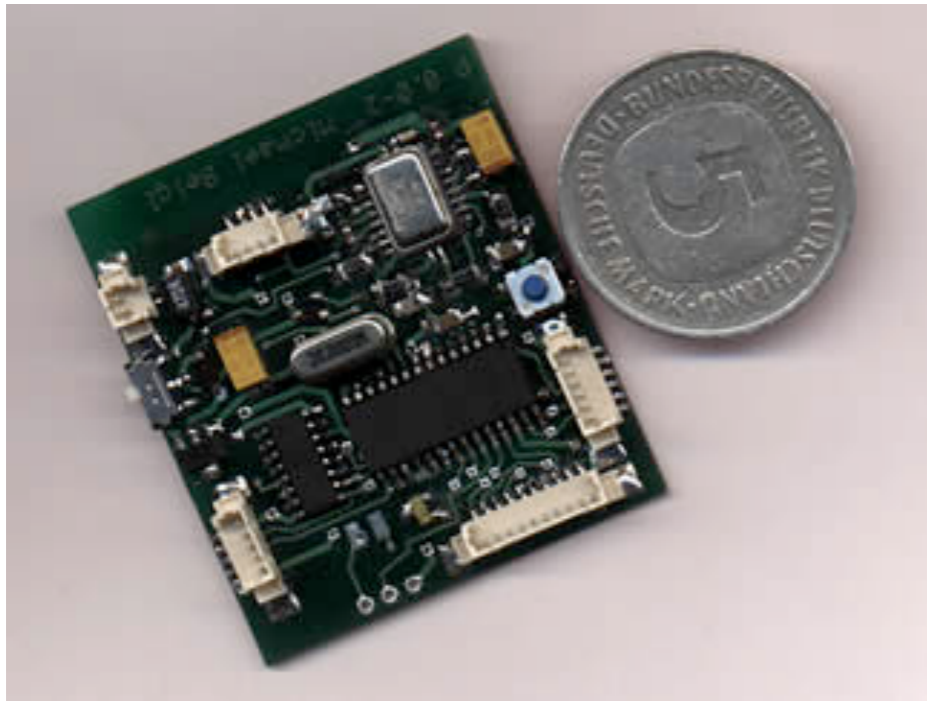
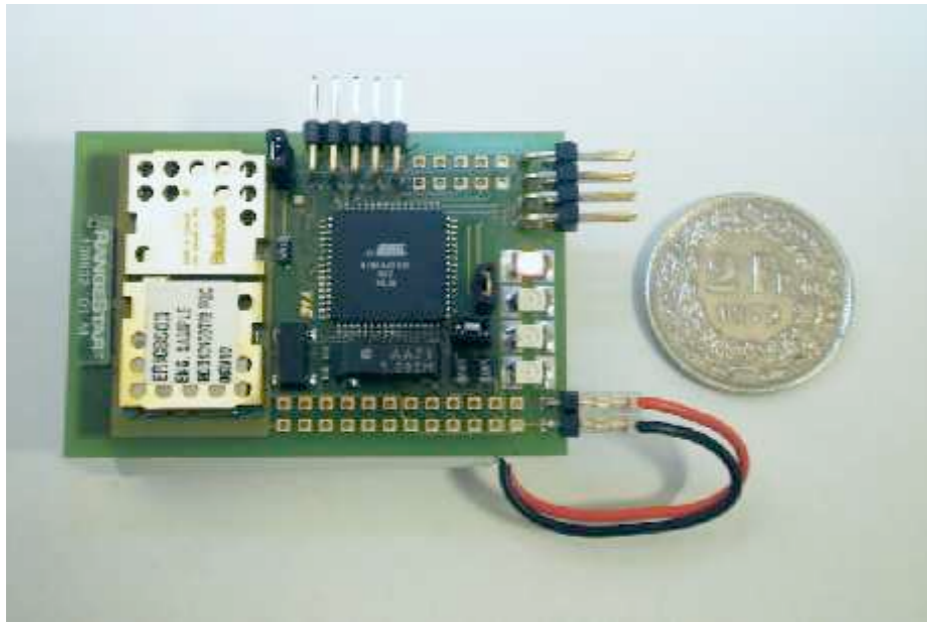
# HW Toolkits

# Genealogy of HW toolkits



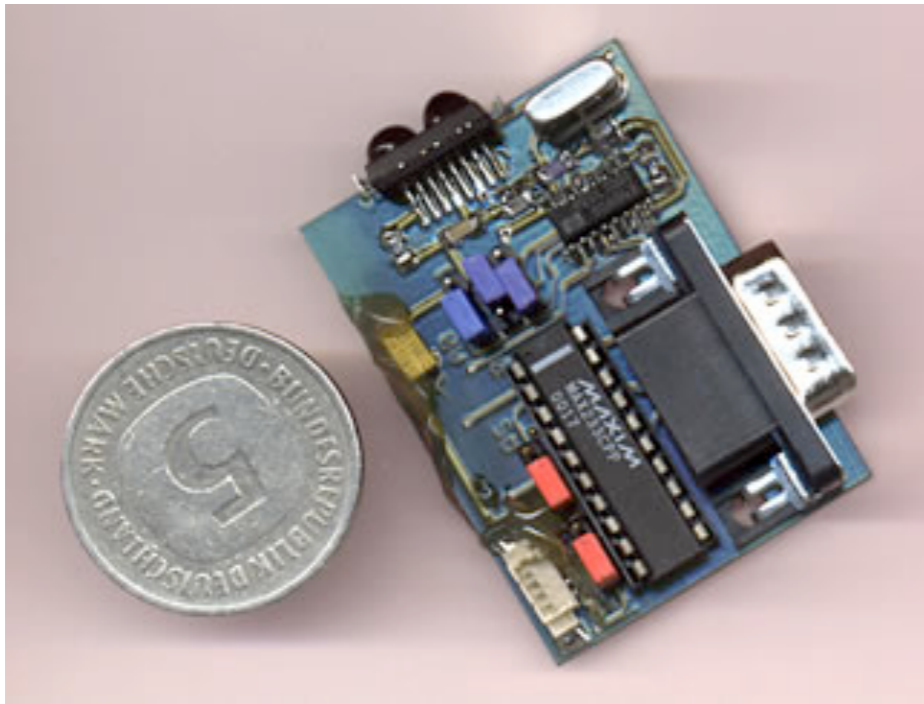
# Smart-Its (TeCo, Univ. Karlsruhe)

<http://smart-its.teco.edu/>

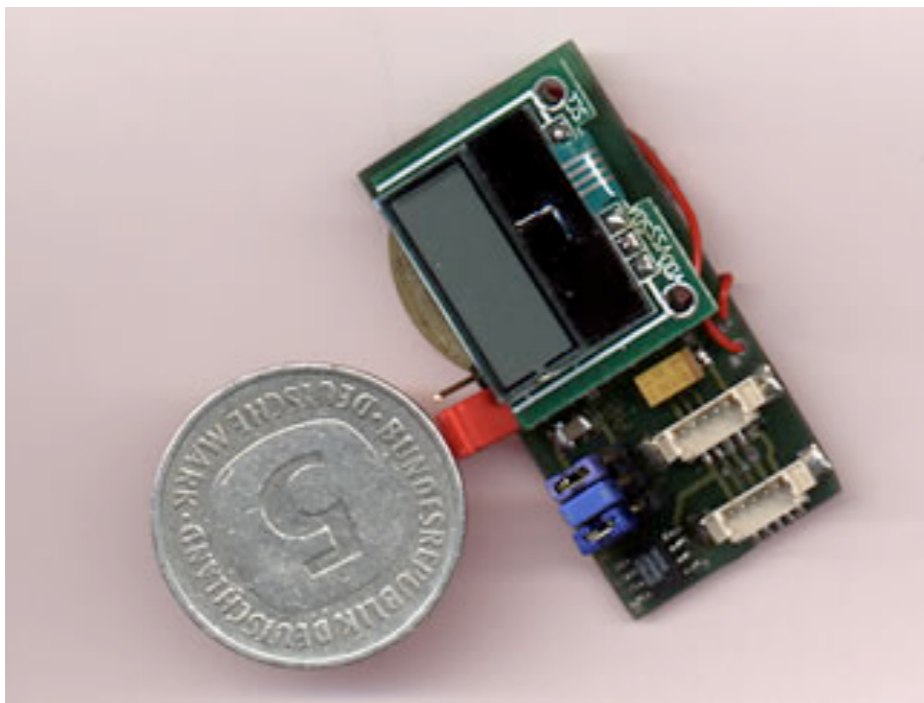


- Development under the EU project Smart-Its (2001-2003)
- 2 base module configurations:
  - Atmel's ATmega103L microcontroller with 128 kB flash memory and 4 kB of SRAM. Ericsson Bluetooth modules
  - PIC 16F876 (20 MHz) for processing, RFM 868 MHz for communication (128kbit/s), on board sensors and an I2C interface for sensor/actor boards.

# Smart-Its sensor modules



- RS232 Add-On (AR 0.0-0)
  - Interfaces to Smart-Its
  - RS232 level
  - IrDA physical layer
  - Power supply through main board



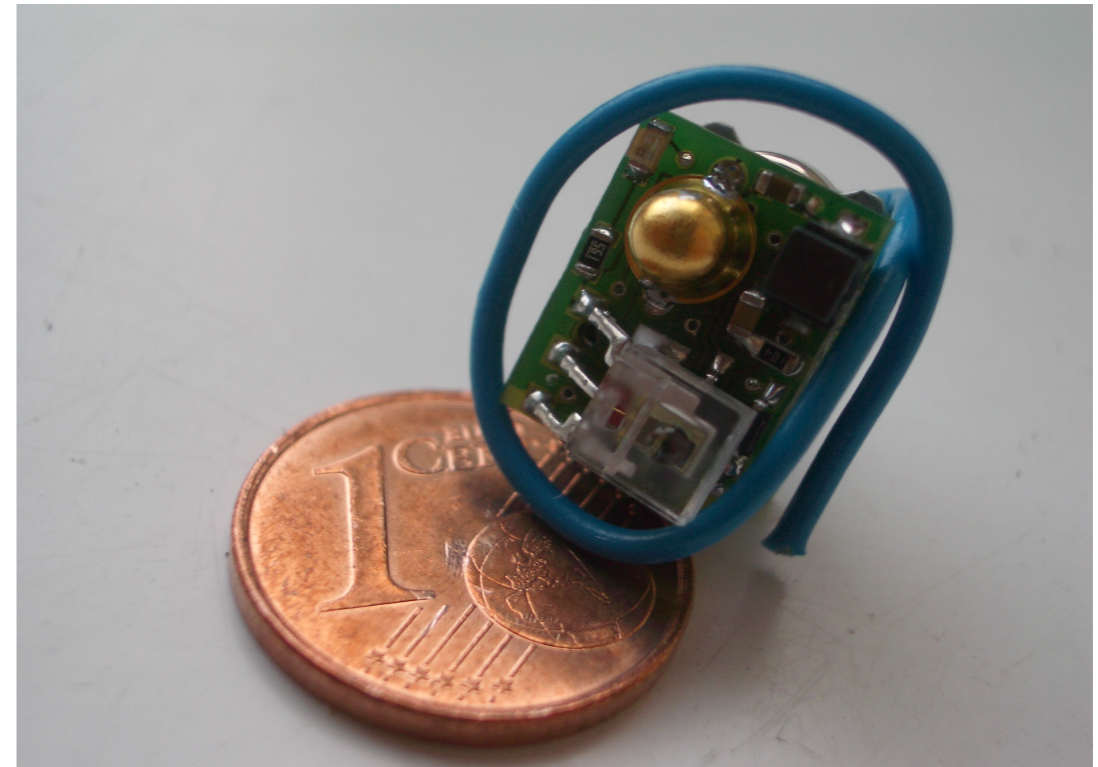
- I/O Add-on for Temperature, Display, Sound (TDS 0.0-0)
  - Interface: I2C
  - 8 char x 2 line display
  - High-Resolution temperature sensor
  - Piezo sound
  - Power supply through main board



# Particles (TeCo, Univ. Karlsruhe)

<http://particle.teco.edu/>

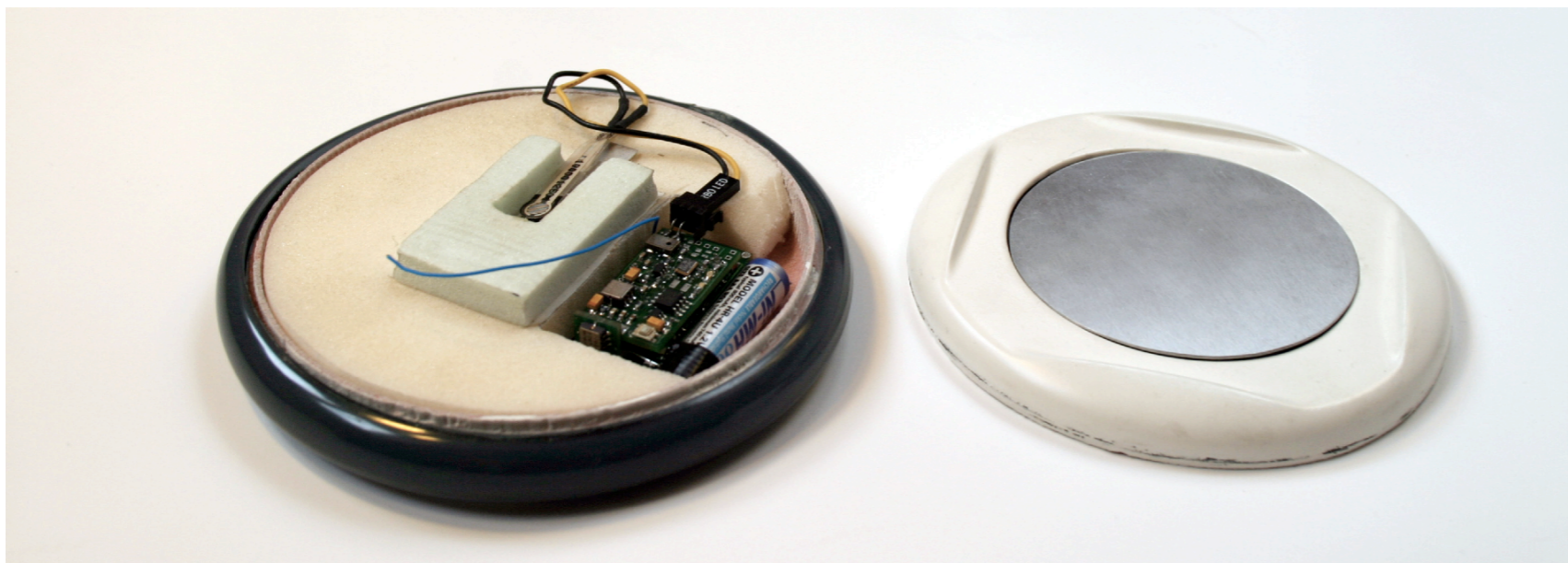
- Currently used in the Embedded Interaction Group (Albrecht Schmidt)
- ...and of course many others...
- Example: Display Cube (see video)



# A beer mat for pub interaction

[Butz, Schmitz, Ubicomp 2005]

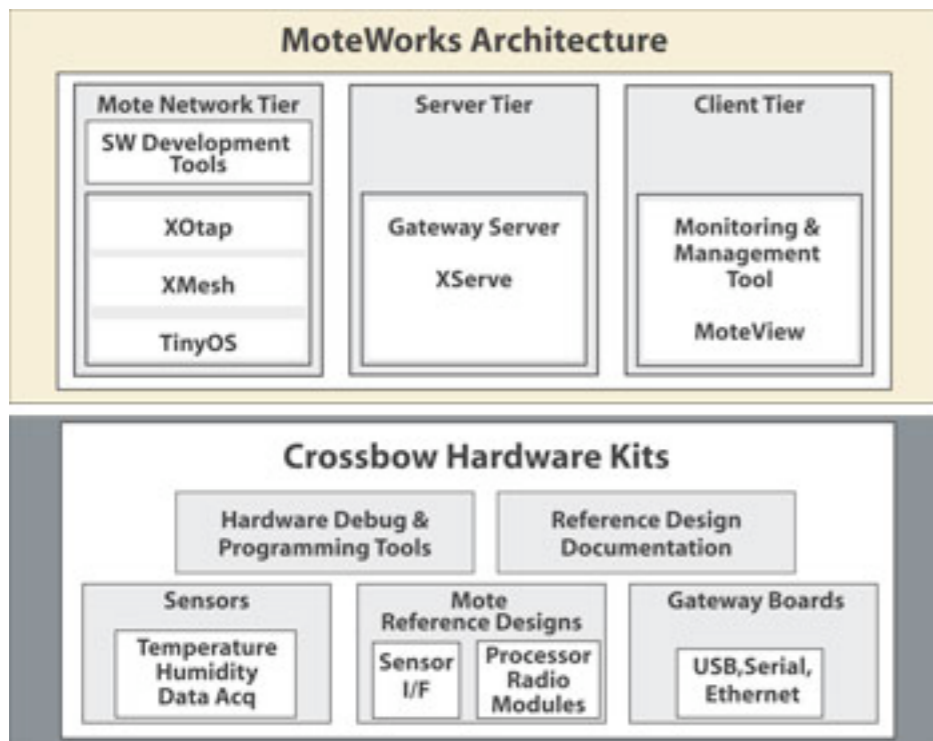
- Device to be put under an actual cardboard beer mat
- Senses weight and gravity
- Shake to order a new drink
- Raise glass (and flip mat) for voting



# Motes (UC Berkeley, crossbow)

<http://www.xbow.com/de/Funksensor-Netzwerk.htm>

[http://www.xbow.com/wireless\\_home.aspx](http://www.xbow.com/wireless_home.aspx)

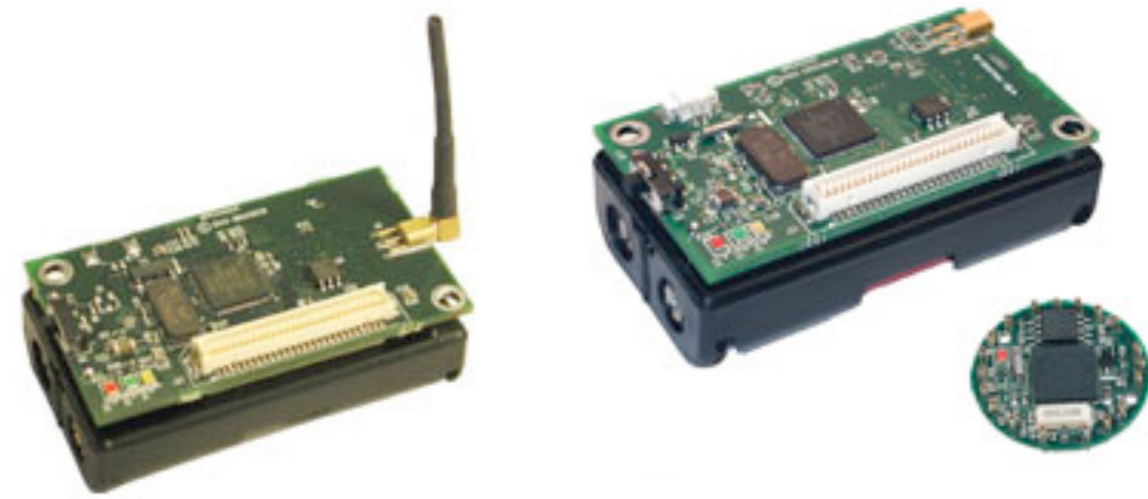


- Developed at UC Berkeley, now spin-off

- See Video

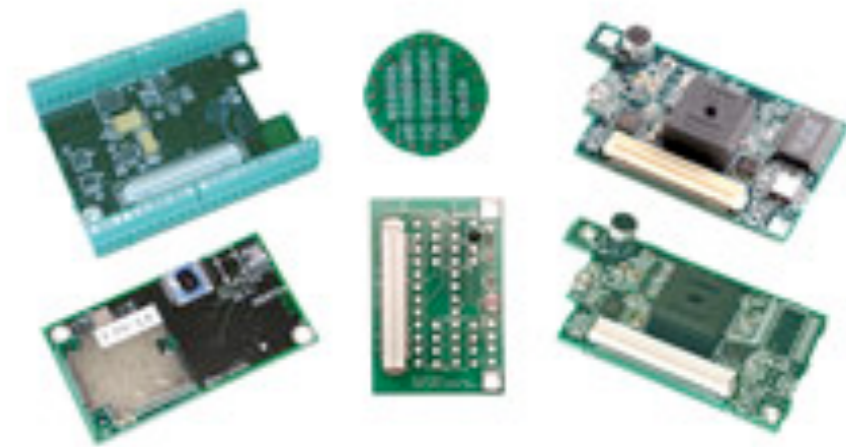


# Motes base units



- 3 Mote Processor/Radio module families:
  - MICAz (MPR2400)
    - radio works on the global 2.4GHz ISM band and supports IEEE802.15.4 and ZigBee.
  - MICA2 (MPR400)
    - available in 315,433,868/900MHz configurations
  - MICA2DOT (MPR500)
    - available in 315,433,868/900MHz configurations
- TinyOS operating system

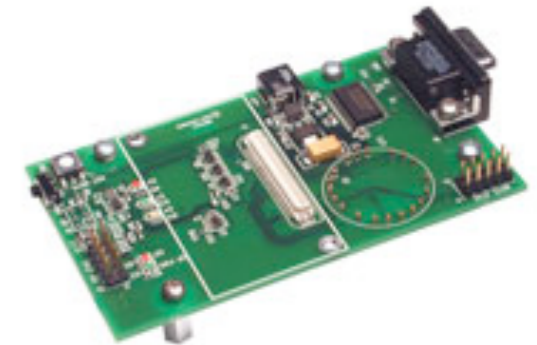
# Motes Sensor Boards



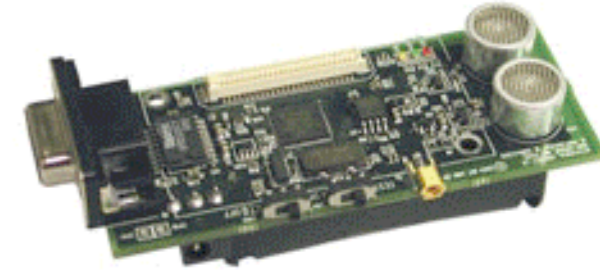
- MDA100 - precision thermistor, light sensor, and general prototyping area.
- MTS300/MTS310 - supports a variety of sensor modalities for the MICA, MICA2 and MICAz
- MDA500 - sensor and data acquisition board for external signals to the MICA2DOT mote
- MTS400/420 - environmental monitoring for the MICA2 and MICAz with built-in sensors and an optional GPS
- MDA300 - supports data acquisition and environmental monitoring for the MICA2 and MICAz
- MTS510 - Light/Accel/Microphone Sensor Board for MICA2DOT

# Mote network base stations

- MIB510 - RS-232 serial interface to the MICA family of Motes.
- MIB520 - USB for the MICA2/MICAz Motes for both communication and in-system programming
- MIB600 - Ethernet Interface board and Mote programming
- Stargate - Linux single board computer with Compact Flash, PCMCIA, Ethernet, USB Host, ...



# Cricket Mote

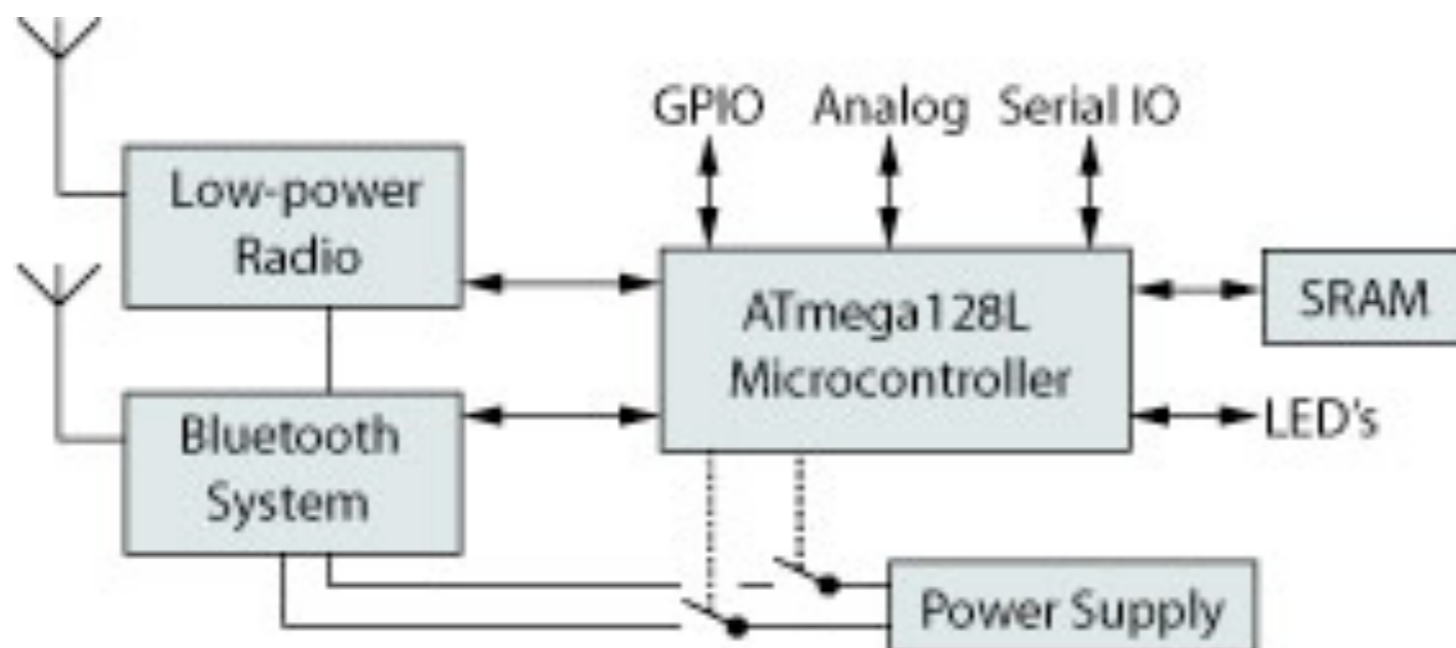


- The MCS410CA or "Cricket Mote" is a location-aware MICA2 Processor/Radio module. The Cricket Mote includes all of the standard MICA2 hardware, as well as an Ultrasound transmitter and receiver. It uses the combination of RF and Ultrasound technologies to establish differential time of arrival and hence linear range estimates.
- → see „Cricket“ in the tracking chapter ;-)

# BTnodes (ETH Zürich)

<http://www.btnode.ethz.ch/>

- Microcontroller: Atmel ATmega 128L (8 MHz @ 8 MIPS)
- Memory: 64+180K RAM, 128K FLASH ROM, 4K EEPROM
- Bluetooth subsystem: Zeevo ZV4002, supporting AFH/SFH
- Scatternets with max. 4 Piconets/7 Slaves, BT v1.2 compatible
- Low-power radio: Chipcon CC1000 in ISM band 433-915 MHz
- Ext. Interfaces: ISP, UART, SPI, I2C, GPIO, ADC, Timer, 4 LEDs
- Standard C Programming, TinyOS compatible
- <http://sourceforge.net/projects/btnode>





# BTnode Developer kit

## ■ Contents:

- 2 BTnode rev3
- 1 usbprog rev2
- 1 Atmel ATAVRISP MK2 programmer
- 2 USB cable
- 1 BTnode CDRROM

## ■ Pricing:

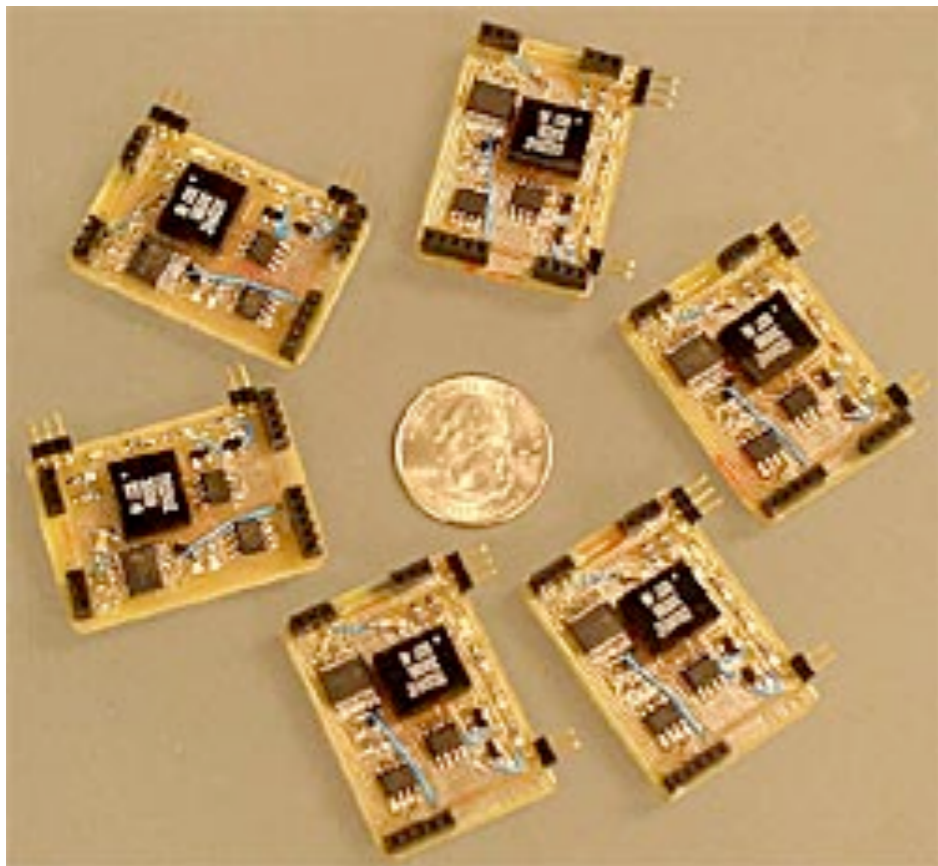
- EUR520/CHF780



# MITes (MIT environmental sensors)

<http://ubicomp.org/ubicomp2004/adjunct/demos/tapia.pdf>

[http://architecture.mit.edu/house\\_n/projects.html#mites](http://architecture.mit.edu/house_n/projects.html#mites)




- No technical web page found, only papers
- Apparently used in several MIT projects
- Senses at least movements, light, temperature
- See Video about the PlaceLab (Pervasive 2005)

# Gumstix

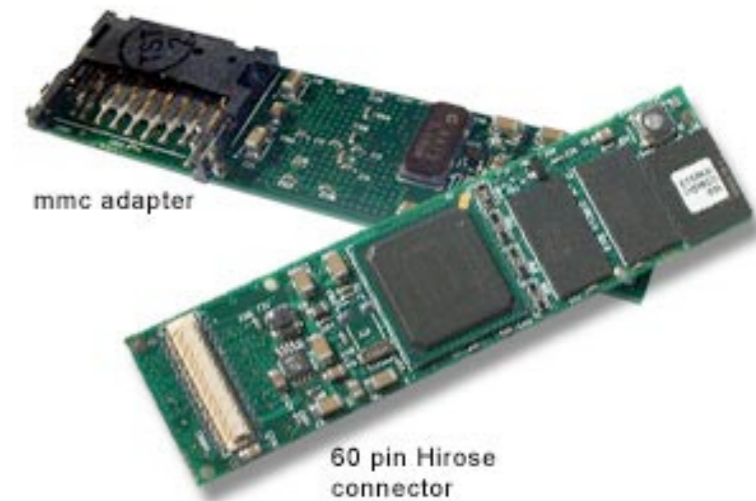
<http://sourceforge.net/projects/gumstix/>

- 80mm x 20mm x 6.3mm
- Intel XScale® PXA255 **400MHz**
- Linux kernel 2.6
- sshd, apache, bluetooth utilities, ...
- Network: Bluetooth, usbnet, PPP
- 3.4V - 5.2V takes Li-Ion, Li-Polymer, 3-NiMH, standard 4.5V or 5.0V inputs
- Draws <250 mA at 400MHz without Bluetooth

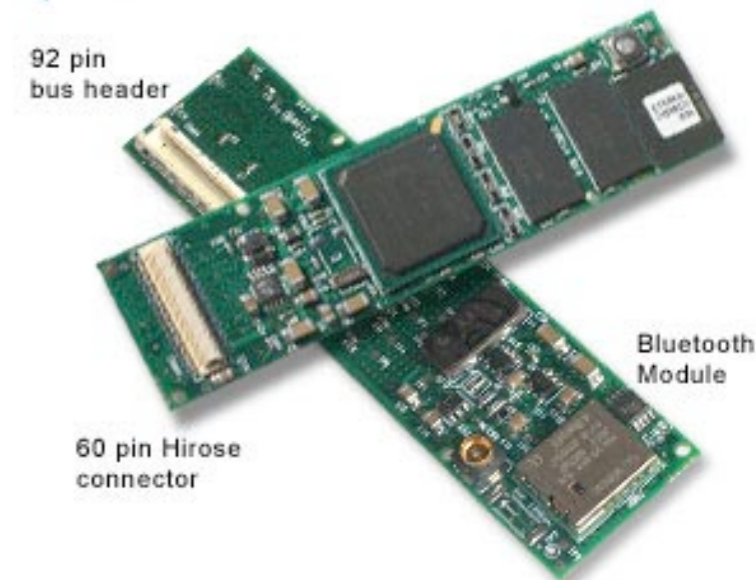
 basix 200



 basix 400 xm



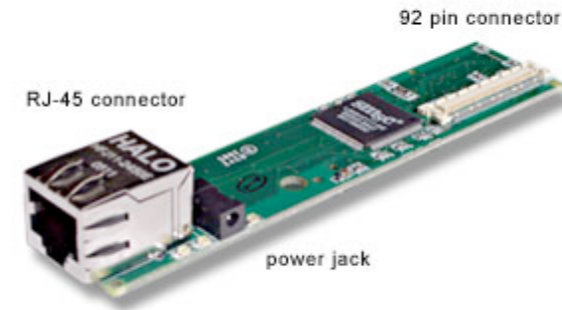
 connex 400xm-bt



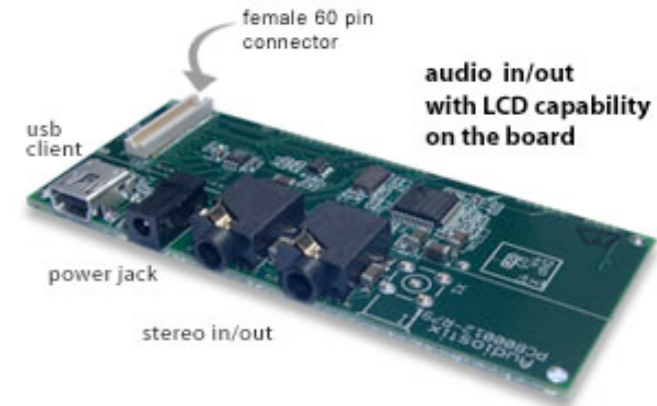
# Gumstix add-ons

- [audiostix 2](#) + [audiostix AC97](#)
- [Breakout gs](#)
- [Cfstix](#)
- [Etherstix](#)
- [GPSstix](#)
- [netCF](#)
- [netDUO](#)
- [netDUO-mmc](#)
- [netMMC](#)
- [roboaudio-th](#)
- [Robostix](#)
- [robostix-TH](#)
- [thumbstix-gs](#)
- [Tweener](#)
- [waysmall – STUART](#)
- [waysmall board](#)

etherstix



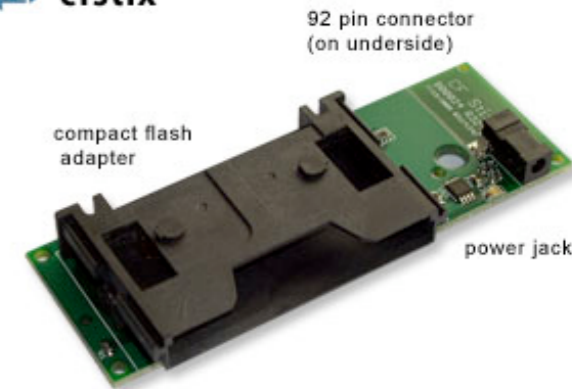
audiostix 2



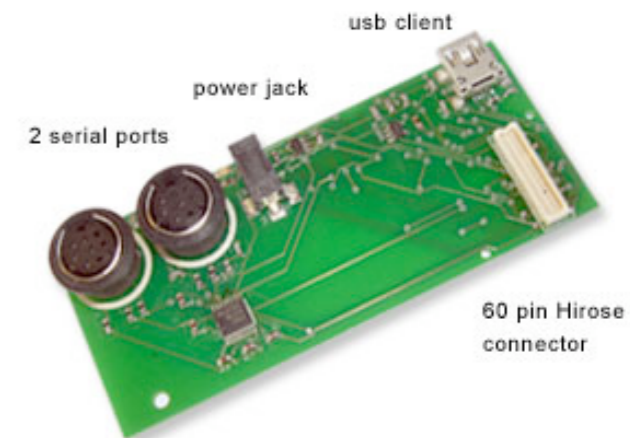
GPSstix



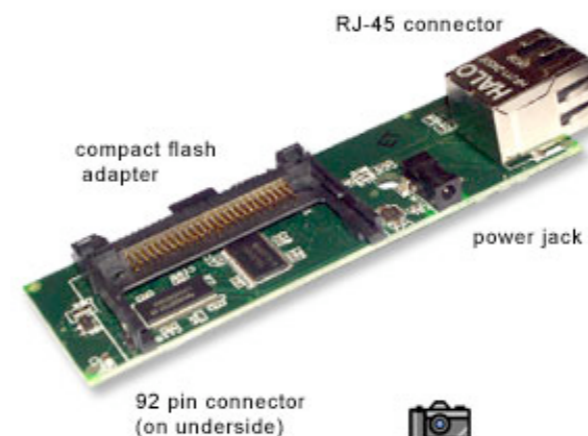
cfstix



waysmall-HWUART

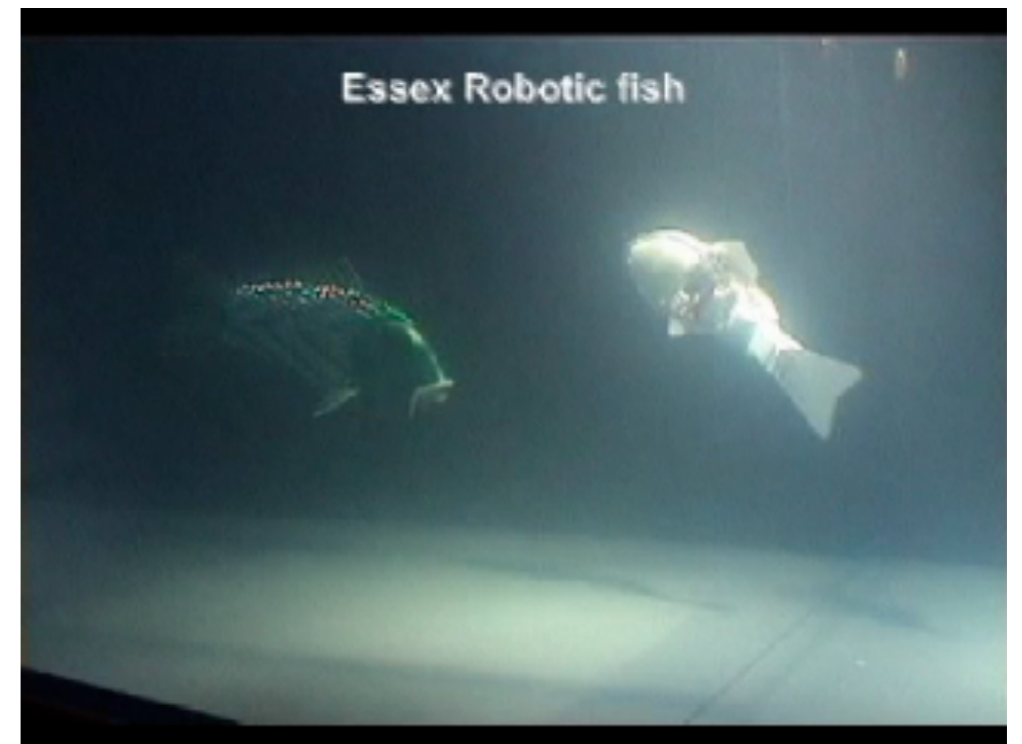
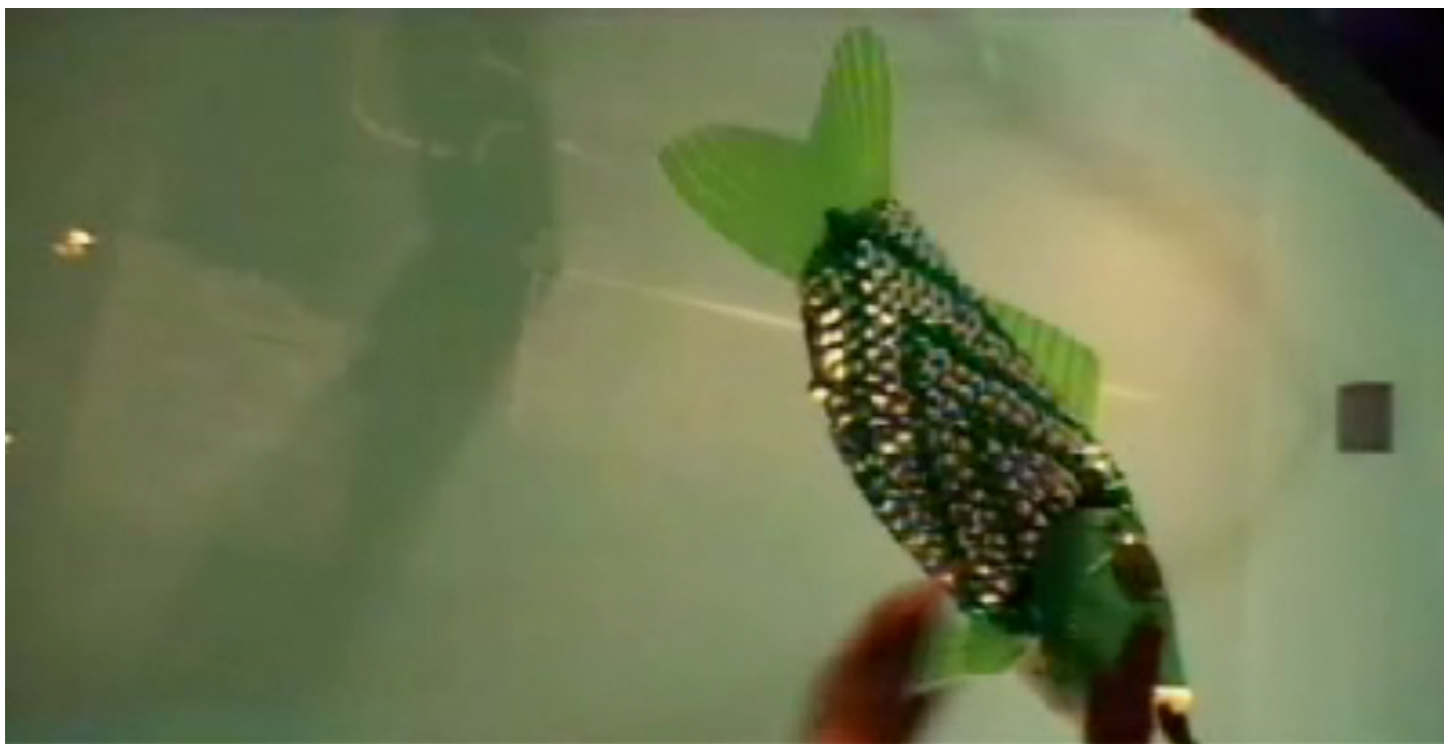


netCF™



# Gumstix app: robotic fish (Essex University)

- **The world's first autonomous robotic fish are the latest attraction at the London Aquarium.** Biologically inspired by the common carp, the new designs can avoid objects and swim around a specially designed tank entirely of their own accord.



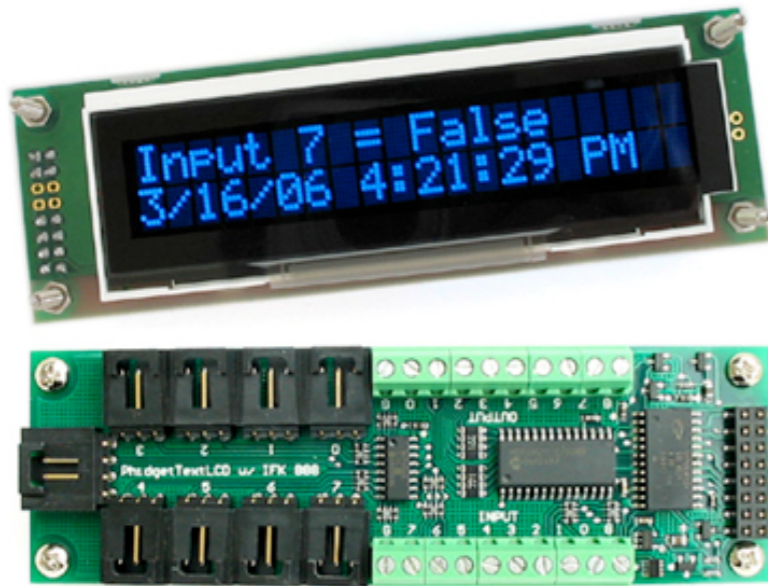
# Gumstix controls flying robot

<http://cswww.essex.ac.uk/staff/owen/research.htm>



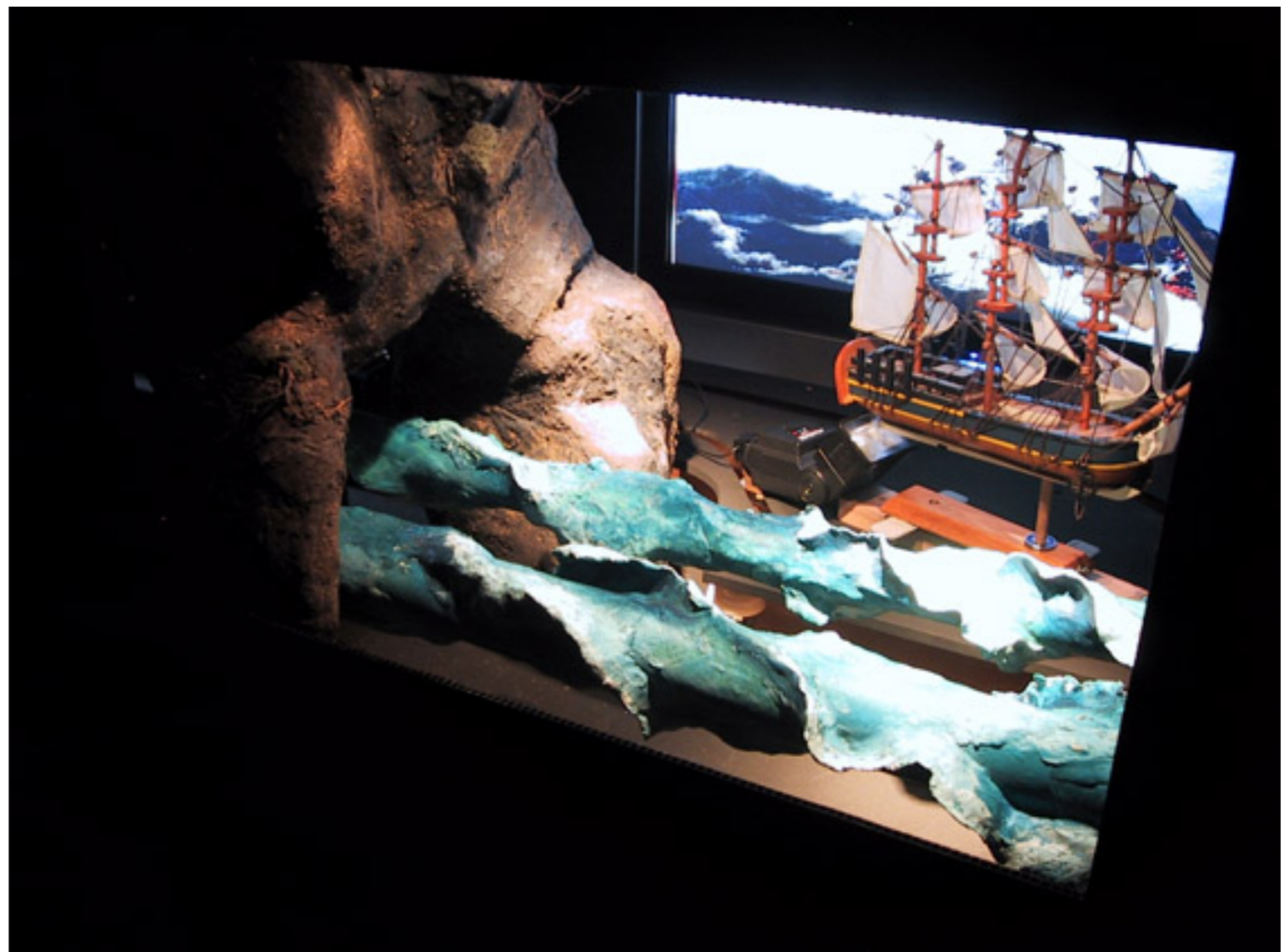
# Phidgets (U Calgary, Phidgets, Inc.)

<http://phidgets.com/>



- Connected to host via USB
- Input modules
  - Analog + digital inputs
  - Sensors, touch sensors,
  - RFID
- Output modules
  - LEDs, Displays
  - Motors, Servos
  - Relay
- Driver support for Windows, MacOS X, Linux
- Java API available

# Phidget app: museum installation



- Various Phidgets used in this museum installation:
  - quadservo controller connected to marine
  - electronic speed controller powering halogen lighting
  - interfacekit 0/0/4 for triggering 2 flashes for lightning
  - motor controller for wave rollers and ship movement



# iStuff (Stanford)

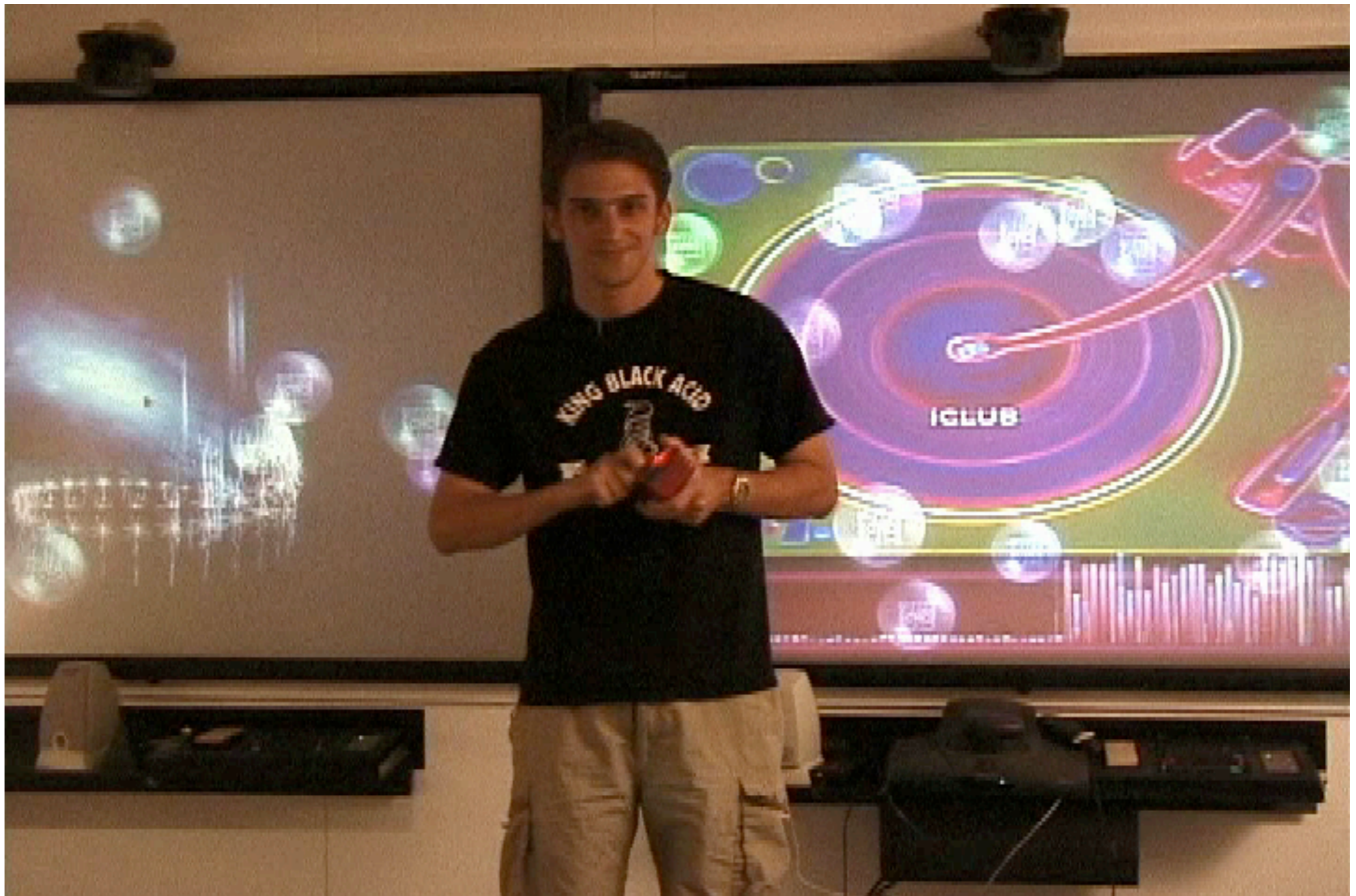
<http://hci.stanford.edu/research/istuff.html>



- Set of input devices for prototyping interaction in instrumented environments
- Also includes software infrastructure (→ event heap)

# iStuff app: iClub

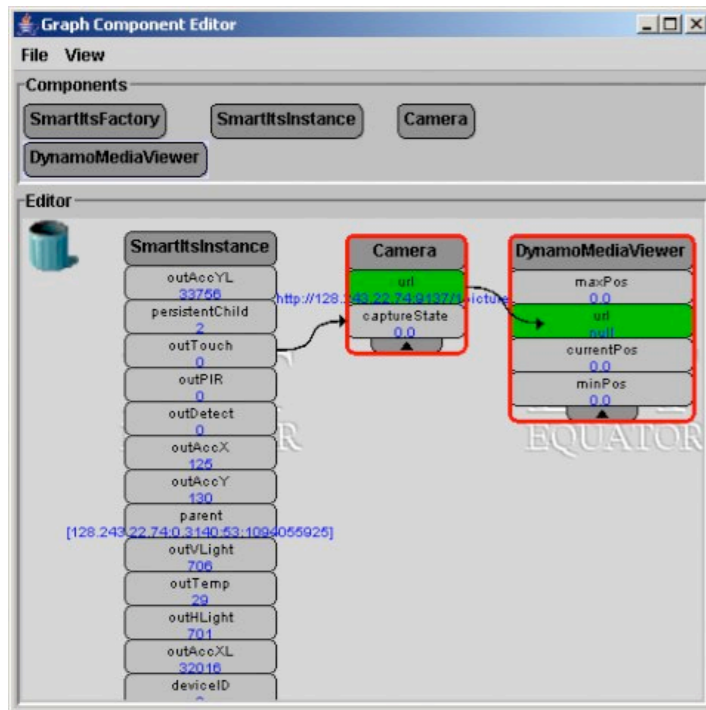
<http://iwork.stanford.edu/photos.shtml#iclub>



# Equip Component Toolkit (Nottingham, RCA)

<http://sourceforge.net/projects/equip>

<http://www.crg.cs.nott.ac.uk/%7Ejym/ect/ect.php>

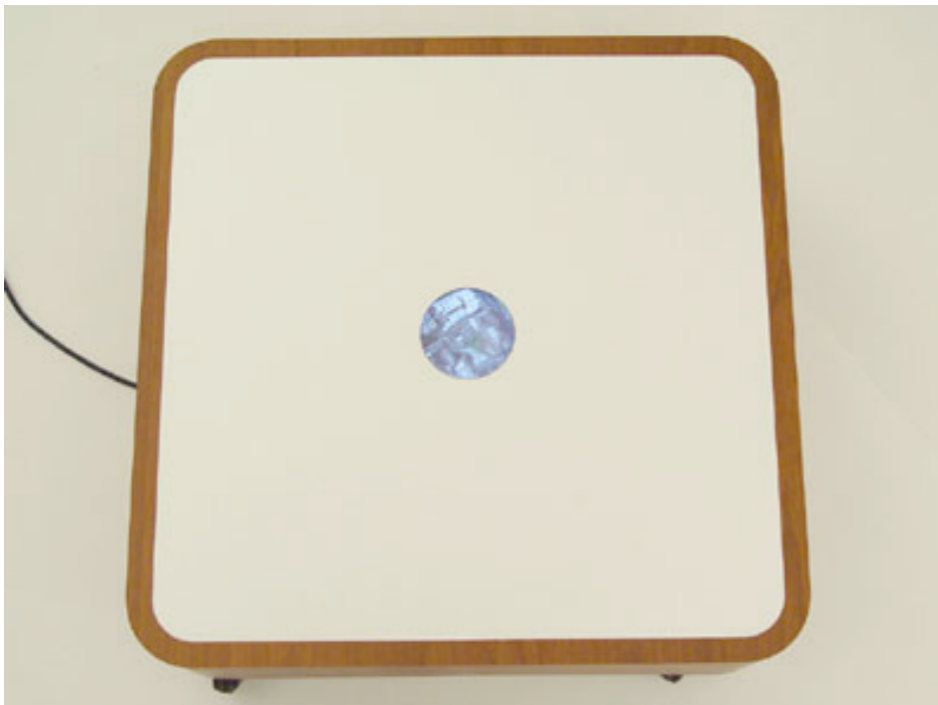


- Collection of I/O hardware components with corresponding software components:
  - [TagIt tag readers](#), using the [TagIt](#) component
  - [Smart-Its](#), using the [SmartItsFactory](#) and [SmartItsInstance](#) components
  - [EZIO serial interface](#), using the [EZIO](#) component
  - Video Input ([JMF](#)), using the [Camera](#) component
  - Audio Input (via [JMF](#)), using the [AudioCaptureHandler](#) and [AudioCaptureDevice](#) component
  - Multimedia Display (via [Dynamo](#)), using the [DynamoMediaViewer](#) and [DynamoSurface](#) components
  - [X10 power-line control system](#), using the [X10](#) component
  - [Processing](#) electronic arts language, using the [ProcessingHandler](#) component





# Example app from RCA: Drift Table



- The Drift Table enables people to slowly float over the British countryside from their own sitting room. The centre of gravity of objects left on this table control the slow scroll of aerial photographs displayed in the table surface. Adding weight to the table causes it to 'descend' zooming in on the landscape below, stacking books on the same side of the table may allow for faster motion. This table suggests a 'hole' in the home connecting physical and virtual space. A display on the side of the table shows the location of the aerial image.

# Calder Toolkit (MERL)

<http://www.merl.com/publications/TR2004-076/>

- Collection of wired and wireless HW components for fast prototyping of physical interfaces
- SW layer:
  - Surrogate objects (SW) represent state of the corresponding HW object
  - Each wired component is a USB HID

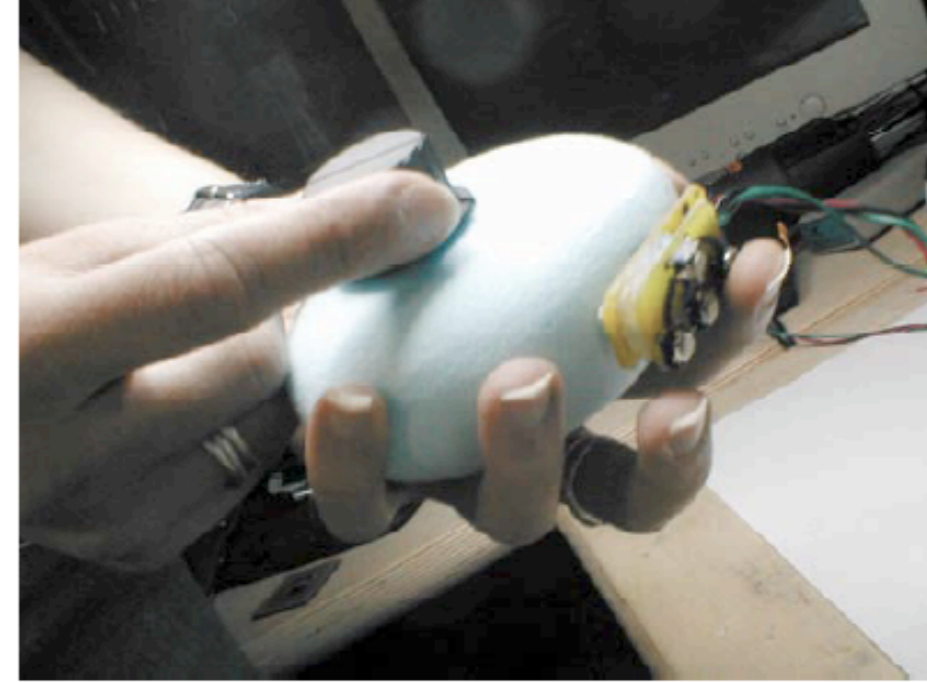


Figure 2. Wireless knob and buttons used within a foam model of a concept game controller.

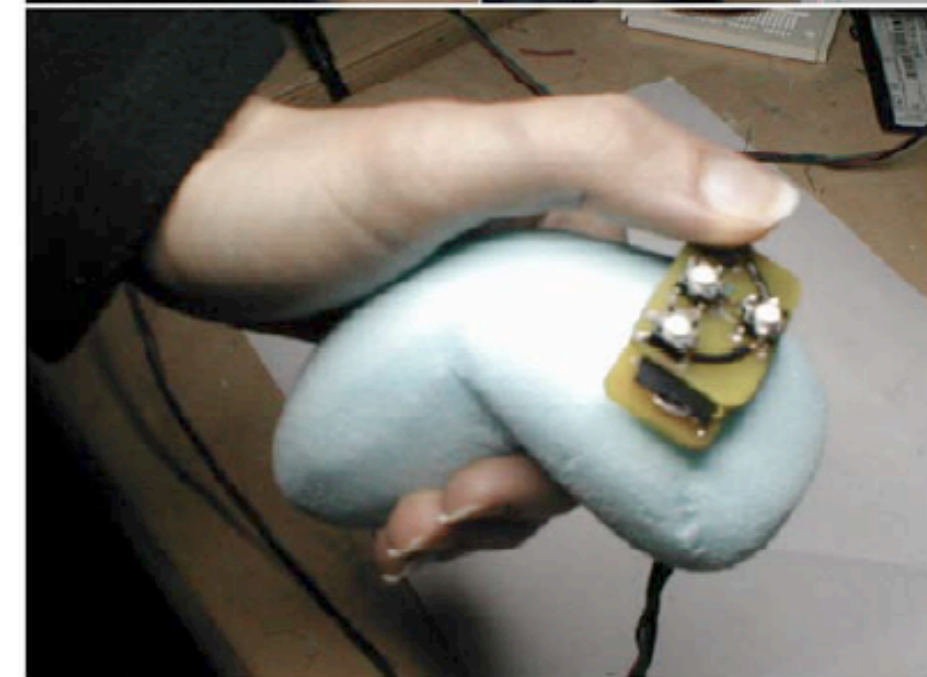


Figure 3. Navigation controller with the wireless buttons placed in a variety of locations around the form.



Figure 6. Wired components. (left to right) I/O Breadboard component, 4-button array, analog joystick, D/A Input Component, single button, and analog knob.



Figure 7. Wireless components: (left to right) analog knob, 4-button array, master uplink transceiver, 2-axis tilt sensor, and 3-LED array.

# Papier Mache (UC Berkeley)

<http://guir.berkeley.edu/projects/papier-mache/>

- Small library combining:
  - Marker recognition
  - Camera-based object recognition
  - RFID recognition
- Easy association between recognized objects and actions through events
  - Sound
  - graphics

