Teaching Next Generation Computing Skills;
The Challenge of Embedded Computing

Minjuan Wang

http://FortiTo.com

ic11@FortiTo.com

Buzz Boards

Minjuan Wang

- Associate Professor of Educational Technology at San Diego State University
- One of a few experts on education for 'Internet of Things' and 'Intelligent Environments'.
- Expert in Mobile learning

2

Overview of Talk

"Exploring how to teach the skills needed to create the internet of things"

- Section 1 – The Challenge of Learning Embedded Computing
- Section 2 – A Modularised Approach to Engineering Design
- Section 3 – ***Buzz Board*** Hardware Platforms
- Section 4 – Participation
- Section 4 – Summary

Buzz Boards

The Challenge of Learning Embedded Computing

- *"Microcontrollers are getting cheaper, more powerful and more flexible, but there remains a barrier to a host of new applications; someone has to build the first prototype! With mbed, we've focused on getting you there as quickly as possible"*
- For most people, even software developers being confronted with such a "raw" computer would be somewhat of a shock
- Simple questions are where is the keyboard, where is the screen, what sort development software does it contain and how can you power it up?

http://mbed.org/

The mbed is a tool for Rapid Prototyping with Microcontrollers

The **mBed** is based on the ARM Cortex-M3 Core running at 96MHz, with 512KB FLASH, 64KB RAM and various interfaces including Ethernet, USB Device, CAN, SPI, I2C

A Computer Science Student's Viewpoint

- Embedded-computers, as supplied from manufactures, are incomplete systems, and require extra hardware and software to make them do anything useful or interesting. Building such hardware and software takes more time than typical university laboratory sessions allow.
- Computer science students have little or no electronic design expertise and, to undertake any meaningful functional design of bare embedded systems would require them to design and build various kinds of hardware input-output schemes and peripherals.
- Once students have constructed hardware, the system has relatively fixed functionality that is difficult to alter, thus working against students getting experience of programming a wide variety of systems.

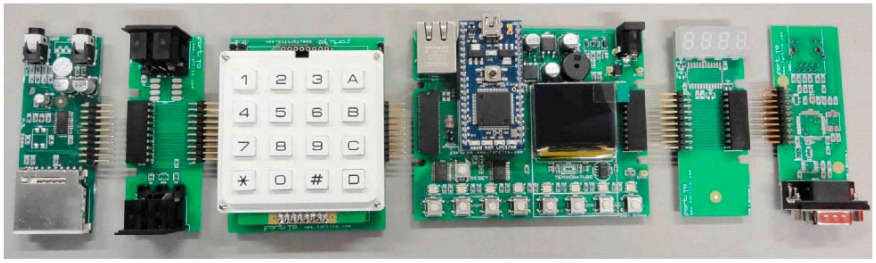
5

An Instructor's Viewpoint

- Doing things from the bottom up is time consuming and, within the limits of typical lab sessions, limits the complexity of the systems that students can build.
- Much of the focus of the computer science curriculum is on the software aspects of embedded computing whereas existing embedded computing offerings revolve around the hardware level, which can distort the focus of the computer science curriculum.
- System level solutions for embedded-computing education tend to either be single appliance oriented (eg a robot), or too simple to give realistic product development experience.
- The software tools are sometimes overly complex, taking a lot of learning and distorting the focus of the underlying computing principles being taught.

6

A Modularised Approach to Engineering Design



- "*divide & conquer*" = modularisation
- Modularisation widely used in computing
- examples include:
  - object-oriented programming,
  - bus-based computer hardware,
  - hardware/software libraries

Buzz Boards

7

Buzz Board Hardware Platforms

1. Mezzanine ARM
2. Processor Base *Buzz Board*
3. Audio-SD *Buzz Board*
4. Manual Control *Buzz Board*
5. Environmental Sensing *Buzz Board*
6. Navigation *Buzz Board*.
7. Inter-board Extension *Buzz Board*
8. Inter-board Right Angled *Buzz Board*
9. 3 Way Inter-board *Buzz Board*
10. Development *Buzz Board*
11. Prototyping *Buzz Board*
12. Keypad *Buzz Board*
13. LED Display *Buzz Board*
14. Medical *Buzz Board*
1. MIDI *Buzz Board*
2. Navigation *Buzz*
3. Network/232 *Buzz Board*
4. Quantum *Buzz Board*
5. RFID *Buzz Board*
6. Robot *Buzz Board*
7. Robot-Lite *Buzz Board*
8. Bluetooth *Buzz Board*
9. GPRS *Buzz Board*
10. WiFi *Buzz Board*
11. Range Finder *Buzz Board*
12. Supplementary Range Finder *Buzz Board*
13. Infrared Beacon *Buzz*
14. Battery *Buzz Board*
15. Test Point *Buzz Board*

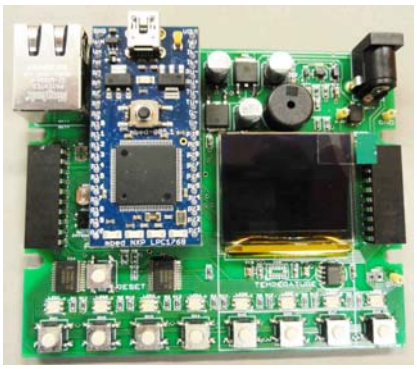
Buzz Boards

8

Software Develpment

- *Buzz Boards* work with standard C and C++
- Development software is based on a simple ‘drag & drop’
- Processor Base Board connected to a PC via USB which behaves like a USB pen drive allows drag and drop of compiled program device – press the ‘reset’ button to execute it.
- Variety of software demos and assignment templates provided (including software source code and assignment text)
- Web-based graphical programming environment (*Buzz Blocks*) under development for less experienced people.

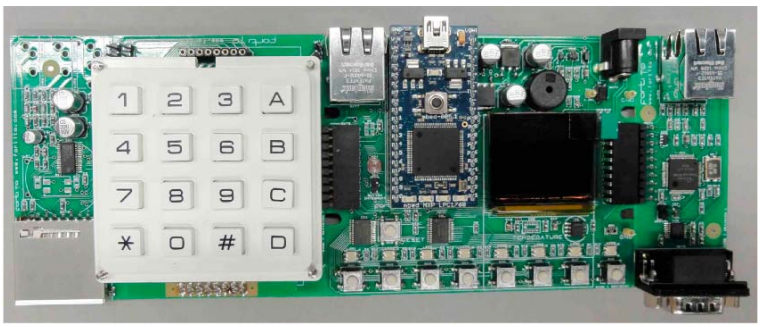
Processor Base *Buzz Board*



- Baseboard accepts mezzanine based processor
- Contains:
  - 8 General purpose push buttons with interrupt output
  - 8 tri-colour LED’s
  - temperature sensor
  - light sensor (with a spectral response that matches the human eye)
  - audio sounder (that can also be used as a microphone),
  - high-resolution full colour OLED display
  - Both external DC and USB power operation
  - 2 bus ports that have I2C, SPI, and general purpose IO
  - 3-Axis accelerometer (optional)

Buzz Boards

A *Buzz Board* Internet Radio

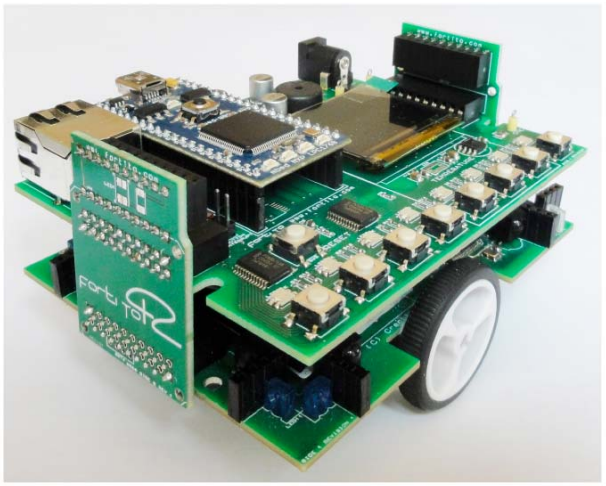


(from left to right) an audio, keypad, base & network *Buzz Boards*

- Internet radio assembled by plugging together
  - ARM-Cortex mBed mezzanine,
  - processor base board, network
  - keypad (optional)
  - audio *Buzz Boards*

Buzz Boards

A *Buzz Board* Desk Top Robot Vehicle



- Desktop robot assembled using
- ARM-Cortex mBed mezzanine,
- Processor base board
- Robot chassis (with IR proximity sensors and batteries)
- Two three-way inter board connectors

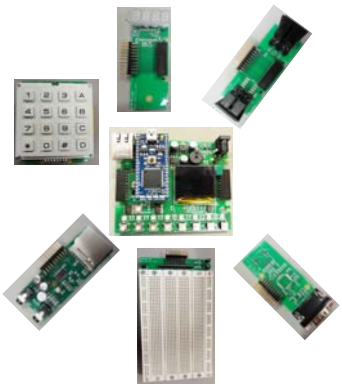
Buzz Boards

Contemporary Approaches

- LEGO MindStorms – ARM based educational toy
  - Mix of Lego bricks to build simple machines such as robots
  - Computer hardware restricted to the use of proprietary modules. **Fortito connects using I²C, SPI or GPIO.**
- Arduino Electronics Prototyping Kit – widely used at school and university level.
  - Based on AVR processor
  - Based on somewhat dated 5v devices (difficult to use with modern 3.3v computer hardware)
  - Expansion system based on modules called 'Shields' (some incompatibilities do to diverse developers). Fortito provide an AVR Mezzanine carrier to allow Arduino based software development.
- mbed – Rapid prototyping system
  - partnership between ARM and Philips (uses Philips NXP LPC1768 Cortex-M3 MCU)
  - takes form of mezzanine carrier for the processor
  - online development tools online
  - Processor memory appears as a USB disk, facilitating 'drag and drop' executables etc.
  - FortiTo provides a carrier for the mbed

13

Academic Participation



Buzz Boards

- **Faculty Cooperatives**
  - Enabling academics to have a stronger stake-holding in the companies that provide educational technology
- **Competition**
  - **Embedded Computing Assignment Innovation**
  - two categories of entry
    - ideas for assignments (based on existing or proposed *Buzz Boards*)
    - actual assignments (code and assignment documents) built from existing *Buzz Board*

14

Summary

- Emerging technologies such as pervasive / ubiquitous computing, and Ambient Intelligence, are creating new opportunities for companies
- Challenges educators as to how to teach the skills involved (differ significantly from desktop computing).
- Requires different types of computational infrastructure and laboratory assignment support.
- Applications can be rapidly created by the teacher or students plugging together various combinations of *Buzz Boards* and *Buzz Blocks*
  - *Currently some 30 modules*
  - Support for differing processor families and differing applications
  - development of hardware (via prototyping boards
  - development of software (via using C/C++ or visual *Buzz Blocks*).
- Faculty Cooperative scheme to involve the academic community at large (see website)
- Competition (see website)

forti tot 42

15

Buzz Boards

forti tot 42

SAN DIEGO STATE UNIVERSITY

http://FortiTo.com

ic11@fortito.com

Buzz Boards

forti tot 42

16