Overview

• Motivation
  – Real time systems
  – non PC software development

• New Course
  – Goals:
    • Small scale microcontroller system development
    • Focus = hardware control and interaction
    • Advanced computer science students (graduate / undergrad)
    • Limit need for hardware development (and cost)

• Results – Fun, successful course
  – Limited hardware knowledge not a barrier
  – Changed student perceptions
  – Economical
Course Tools

- **Hardware**
  - Arduino open source microcontroller board
    - Atmel Atmega 328 processor
    - 32K bytes of memory
    - Power options, clock, timers
    - inputs / outputs
    - Physical reset button
  - Wireless breadboards
  - Electronics parts

- **Software** - Programming in C
  - Standard C, interfaces to hardware, helper methods
  - Boot loader, USB interface
  - Very basic main loop “operating system”

Software Structure

```c
/*Blink
Turns on an LED on for one second, then off for one second, repeatedly.
The circuit: LED connected from digital pin 13 to ground.
By David Cuartielles; based on an original by H. Barragan for the Wiring i/o board*/
int ledPin = 13; // LED connected to digital pin 13

void setup() {
  pinMode(ledPin, OUTPUT);
}

void loop() {
  digitalWrite(ledPin, HIGH);  // set the LED on
  delay(1000); // wait for a second
  digitalWrite(ledPin, LOW);  // set the LED off
  delay(1000); // wait for a second
}
```

Hardware Tinkering

• Project Construction
  – Arduino board a piece part
  – Add
    • Inputs – buttons, sensors (temp, sound, touch)
    • Outputs – LEDs, motors / servos, speakers
  – Need to learn basic electronics
    • What’s a multi meter?
    • Electrostatic Discharge (ESD) precautions
    • Only 9 volts and milliamps, but…

Course Structure

• Introduction to Arduino and Hardware
  – Series of simple projects
  – Experimentation with sensors
  – Build simple circuits from diagrams
  – Access to the Arduino community

• Custom Project(s)
  – One or two people
  – Student defined project
  – In class demonstrations and help
  – Student self assessment of others
Findings & Results

Do you understand the difference between programming microcontrollers and higher level programming (i.e., Java)?

Structured Analysis and Design Technique (SADT)
William L. Honig, whonig@luc.edu

Reasonable Economics (USD):
- $150 per student kit
- $500 in hand tools
- $45 damaged parts

Advanced Projects

Arduino: capable platform; enables creativity

Arduino Glider Auto Pilot

William L. Honig, whonig@luc.edu
Next Steps / Improvements

- Speed up the basics
  - Expanded standard set of labs
  - Circuit experiments (no Arduino)
- Option to BYO Arduino
  - Build Arduino from parts
  - Students enjoyed soldering
- Option for assembly code
  - When is timing really critical?
- Expand in robotics
  - Interest is keen
  - Real time input and response

Summary

- Small scale intelligent systems / embedded computing
  - Important student learning
- Arduino open source hardware / software
  - Accessible for advanced computer science students
    - Even those without electronics background
  - Strong and capable platform
- Economics – possible for non engineering schools

• Acknowledge
  - Reviewers, Thanks!
  - Arduino Community:
    • Massimo Banzi, Tom Igoe, …
Backup Materials

• Rover Videos:
  • see http://technospino.com/micro/

• Robert’s YouTube:
  • http://www.youtube.com/watch?v=moFl8xIHG9c