Who are we teaching to...

Arduino for hipsters
Introduction to Arduino
With Protosnap Pro Mini

BRIAN HUANG
Education Engineer

JIM LIMBLOM
Design Engineer

BEN LEDUC-MILLS
Education Outreach Coordinator
Ignite Format

Speakers are given 20 slides, each shown for 15 seconds, giving each speaker 5 minutes of fame.

Why do it??

• Keeps the slides moving.
• Gives students a well defined format to rehearse and practice.
• It’s fast, fun, and crazy…
In 5 minutes...

• About SparkFun
• Learning Outcomes
• Motivation behind all of this
• What is Arduino
• Examples of projects and things people have done...
About Us

SparkFun Electronics is an Open Source Hardware Company. We are all about creation, innovation and sharing information. We want to get you excited about cutting edge electronics technology.
Learning Outcomes

You will understand basic electrical and computational principles used in making, building, and inventing with Arduino.

You will be able to design and write programs to interface sensors and control LEDs, buzzers, and other devices.
Motivation
The “IKEA Effect”: When Labor Leads to Love

Michael I. Norton
Daniel Mochon
Dan Ariely
“When we allow children to experiment, take risks, and play with their own ideas, we give them permission to trust themselves. They begin to see themselves as learners who have good ideas and can transform their own ideas into reality.”

From Tinkering to Engineering...

“Engineering is the application of scientific principles to design, build, and invent.”

It is “…any engagement in a systematic practice of design to achieve solutions to particular human problems.”

“We teach science and math so that they can make the world a better place, not so that they can pass tests.”

I want ________ to ________.
What is *it* all about?

Stop Following Directions & Start Engineering
The Learning Cycle

Time to play, reflect, question and redesign

Concrete Experience
(doing / having an experience)

Active Experimentation
(planning / trying out what you have learned)

Reflective Observation
(reviewing / reflecting on the experience)

Abstract Conceptualisation
(concluding / learning from the experience)

Theory
create or modify the theory

Observation
perform the experiment

Prediction
use the theory to make a prediction
design an experiment to test the prediction

Experiment
About the Computer Science Education Legislation

Members of Congress have introduced legislation that will remove barriers to computer science in K-12 classrooms nationwide. The Computer Science Education Act (CSEA) has been introduced in the US House of Representatives (HR 2536) and the Computer Science Education and Jobs Act was introduced in the US Senate (S 1407). Read a summary of the issues the bills address and what they do [here](#).

Tell Your Members of Congress to Support Computer Science Education

[Code.org](#), Computing in the Core and its members, and K-12 computer science education advocates support this bipartisan legislation. Representative Susan Brooks (R-IN) and Representative Jared Polis (D-CO) were joined by Representatives Susan Davis (D-CA), Suzan DelBene (D-WA), Mike Hanna (R-NY), Mike Honda (D-CA), Duncan Hunter (R-CA), Bill Johnson (R-OH), Jim Langevin (D-RI), Cathy McMorris Rogers (R-WA) and Representative Luke Messer (R-IN) to introduce the Computer Science Education Act (CSEA) in the US House of Representatives. Senator Robert Casey (D-PA) and Senator Marco Rubio (R-FL) introduced the Computer Science Education and Jobs Act in the Senate. You can help build support for this commonsense, no-cost legislation. How? Write to your Representative and Senators and ask them to cosponsor the legislation.

- Is this your first time writing your Members of Congress? It is easy! This [website](#) can help you write.
Program or be programmed

If nothing else – we must help to developing a computationally literate society…
What is Arduino...

Describes both the hardware (board) and the programming language.
Developed off the Atmel ATMega328 chip.
Designed with Artists & Non-Engineers in mind...
But... what is it???

Think of this as a mini-brain...
Nearly a million ideas...

20 Unbelievable Arduino Projects - Instructables
www.instructables.com/id/20-Unbelievable-Arduino-Projects/
Here are twenty amazing Arduino projects that you almost wouldn't believe, if not for that they are the real deal. These authors have turned their wil...
Make a 24X6 LED matrix - Plantduino Greenhouse - Bubblesteen Bubble Machine

5 Amazingly Cool Arduino Projects | LifeScoop
mylifescoop.com/2013/06/05/5-amazingly-cool-arduino-projects/
Jun 5, 2013 - Arduino is an "open-source electronics prototyping platform based on flexible, easy-to-use hardware and software."

Top 40 Arduino Projects of the Web | Hack N Mod
hacknmod.com/hack/top-40-arduino-projects-of-the-web/
Jan 27, 2009 - arduino-projects ... Know of more fantastic Arduino projects? ...
Absolutely awesome article, the arduino is a fantastic programmer I have come ...
Create Techno with a Laser Harp - Top 30 Wilmote Hacks - How to Make LED Cubes

This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 United States License.
Some cool projects
Project: Mood Lamp / Light Sculpture
Further Motivation
From the NGSS

The practices describe behaviors that scientists engage in as they investigate and build models and theories about the natural world and the key set of engineering practices that engineers use as they design and build models and systems.
Agenda

- Ignite Introduction
- Protosnap ProMini vs. Arduino
- ArduBlock – Graphical Programming for Beginners…
- Adult - Open Swim
- Arduino IDE Programming Environment
- Data Collection Demo
- Resources
Protosnap ProMini vs. Arduino
ArduBlock
A Graphical Programming Language for Arduino

Getting Started with ArduBlock

[20130712 Release] Update with L... by david | 07-12-2013

We are very happy to receive LinkerKit update from the LinkSprite team. For more information about LinkerKit, click here.

This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 United States License.
ArduBlock
A Graphical Programming Language for Arduino

Getting Started with ArduBlock

DOWNLOAD Ardublock
1. Download arduino-1.jar ArduBlock
2. In Arduino IDE, open menu "Arduino" -> "Preferences"
3. Find "Sketchbook location:"

Preferences

Sketchbook location:
/Users/toyhouse/Documents/Arduino

This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 United States License.
Installing ArduBlock

Create a folder structure:

```
tools\ArduBlockTool\tool\tool
```

under the \My Documents\Arduino folder

Save \ Move the ArduBlock.jar file here.

Note: the folder names are case-sensitive!!!
General Rules

Raise your hand if you have a question.
Be respectful.
When I raise my hand…
Share your findings with the people near you.
Poke around, experiment, and have fun…
Imagine, create, and play!!!
The ProtoSnap ProMini Board

Prototyping space

Buzzer

RGB LED

Light Sensor

Red, Green, Blue LED

Button
Pin Connections

Where you attach sensors, LEDs, button switches, etc…
Actual Pin-outs

TQFP Top View

1. (PCINT19/OC2B/INT1) PD3
2. (PCINT20/XCK/T0) PD4
3. GND
4. VCC
5. GND
6. VCC
7. (PCINT6/XTAL1/TOSC1) PB6
8. (PCINT7/XTAL2/TOSC2) PB7
9. PD5
10. PD6
11. PD7
12. PD8
13. PD9
14. PC2
15. PC3
16. PC4
17. PB5 (SCK/PCINT5)
18. AVCC
19. ADC6
20. AREF
21. GND
22. ADC7
23. PC0 (ADC0/PCINT8)
24. PC1 (ADC1/PCINT9)
Flip the Board Over

Do you see the wires that are running to the sensors, LEDs, and buttons?
The microcontroller is pre-wired to the inputs and outputs on this board.
The ProtoSnap Pro Mini Board

Prototyping space

Buzzer

RGB LED

Light Sensor

Red, Green, Blue LED

Button
Notes on Vocabulary

Vcc = Common Collector Voltage (+5 V or the positive terminal of the power source)
GND = Ground (also referred to as the negative terminal of the power)

Circuit = Circle

Plated Through Hole →

This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 United States License.
No Circuit building, no fuss
Double-click on either
the Arduino Icon or
wherever you
installed (saved) the
Arduino program.
Plug it in
Lesson #0 – Configuring Arduino

1. Setup Board
   (LilyPad Arduino w/ ATmega328)

2. Setup COM Port
   PC – Highest **COM #**
   Mac – `/dev/tty.usbserial-A#####xXx`
Settings: Tools → Board

- ATtiny45 (internal 1 MHz clock)
- ATtiny45 (internal 8 MHz clock)
- ATtiny45 (external 20 MHz clock)
- ATtiny85 (internal 1 MHz clock)
- ATtiny85 (internal 8 MHz clock)
- ATtiny85 (external 20 MHz clock)
- ATtiny44 (internal 1 MHz clock)
- ATtiny44 (internal 8 MHz clock)
- ATtiny44 (external 20 MHz clock)
- ATtiny84 (internal 1 MHz clock)
- ATtiny84 (internal 8 MHz clock)
- ATtiny84 (external 20 MHz clock)
- Arduino Uno
- Arduino Duemilanove w/ ATmega328
- Arduino Diecimila or Duemilanove w/ ATmega168
- Arduino Nano w/ ATmega328
- Arduino Nano w/ ATmega168
- Arduino Mega 2560 or Mega ADK
- Arduino Mega (ATmega1280)
- Arduino Leonardo
- Arduino Esplora
- Arduino Micro
- Arduino Mini w/ ATmega328
- Arduino Mini w/ ATmega168
- Arduino Ethernet
- Arduino Fio
- Arduino BT w/ ATmega328
- Arduino BT w/ ATmega168
- LilyPad Arduino USB

This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 United States License.
Settings: Tools → Serial Port

![Arduino IDE Tools Menu with Serial Port Setting](image-url)
Other Board Types
Start ArduBlock
Tools  ArduBlock
ArduBlock

Introduction to Programming
Scratch-like interface
Clear transition to text-based coding
Lesson #0 – Test Upload
Check for Status Messages

```c
void loop()
{
  digitalWrite( 13, HIGH );
}
```

Done uploading.

Binary sketch size: 1,076 bytes
(of a 32,256 byte maximum)
Success.
Block Organization in ArduBlock
Block Organization Control

- loop do
- if/else
  - test
  - then
  - else
- if
  - then
- while
- condition
- repeat
- subroutine
- subroutine

This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 United States License.
Block Organization Pins

digital pin #

analog pin #

set digital pin #

set analog pin #

Servo pin#

angle

tone pin#

frequency

noTone pin#
Block Organization

Numbers/Constants

1
HIGH
LOW
TRUE
FALSE

set number variable var name
value 0

set digital var variable name
value HIGH

message

number var name
glue
digital var name
glue
Let’s get to hacking…

Lesson #1 – Blinking an LED

“Hello World” of Physical Computing

Psuedo-code – how should this work?

Turn LED ON → Wait → Turn LED OFF → Wait → Rinse & Repeat
Let’s get to hacking...

Lesson #1 – Blinking an LED

“Hello World” of Physical Computing
Alright? Let’s blink!

Challenge 1a – Sweet Heart. Blink to mimic a heartbeat…

Challenge 1b – Too Fast! Experiment with the blink rate. What’s the fastest blink rate you can see? 1 ms? 2 ms??

Challenge 1c – Three in One! Use the RGB LED to create different combinations and patterns of light. Anything peculiar?
Save your Project

- Save early, Save often!
- Enumerate your iterations – (e.g. blink_10.adp, blink_11.adp, blink_12.adp...)
New Challenge...

Create a blink that starts off with a 10 ms delay and gets progressively slower and slower...

What would the pseudo-code look like for this:
Programming Concepts: Variables

Variable Types:

- **8 bits**
  - byte
  - char

- **16 bits**
  - int
  - unsigned int

- **32 bits**
  - long
  - unsigned long
  - float

This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 United States License.
Using Variables

Numbers/Constants

Variable is just a placeholder for a value.

Notice the different shapes?
Example: Did you come up with something like this?
Serial Monitor

Serial print command

Any message

“glue”

Data
Add this block... so that we can watch the variable value
Upload & Click back to Arduino

Open the Serial Monitor

```cpp
Serial.println(""");
digitalWrite( 13 , HIGH );
delay( _ABVAR_1_waitTime );
digitalWrite( 13 , LOW );
delay( _ABVAR_1_waitTime );
```

Done uploading.

Binary sketch size: 3,056 bytes (of a 32,256 byte maximum)
Conditional Statements
If()... else...

test conditions – aka Boolean

Greater Than
Less Than
Equal To
Greater or Equal
Less or Equal
Not Equal to
Aside from just blinking (on and off), what else might we want to do??
Understanding: Analog vs. Digital

Microcontrollers are **digital** devices – ON or OFF.

An Analog signal is anything that can be a full range of values. What are some examples? Think of like a ramp or a hill.
Understanding: Analog vs. Digital

To create an analog signal, the microcontroller uses a technique called PWM.

Pins 3, 5, 6, 9, 10, 11 are capable of producing an Analog Output.

[Diagram of Pulse Width Modulation (PWM)]
In ArduBlock

**Digital OUTPUT**
- HIGH or LOW
- PIN# -- 0 to 13

**Analog OUTPUT**
- 0 to 255
- PIN# -- 3, 5, 6, 9, 10, 11
Challenge 2a:

1) How do you turn on the RED LED?

2) Experiment with different values – what do you observe happens? Anything peculiar??

3) How do you turn on the GREEN or the BLUE LED?
Blinking in 3 colors
Tri-color LED

Only to throw you for a loop…
RGB LED is a Common-Anode
(means the positive side of the LED is all tied to $V_{CC}$)
We’ve controlled stuff... what’s next?
What about Getting INPUT?

Digital INPUT
Range: ??
PIN# -- 0 to 13

Analog INPUT
Range: ??
PIN# -- 0 to 5 (A0 to A5)
Digital Input & Feedback – Serial Monitor

Serial print command

Any message

“glue”

Data
Example

or

---

This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 United States License.
Digital Input & Conditional Logic
Challenge --

When the button is pressed, blink the LED 10 times.
In ArduBlock

**Digital OUTPUT**
- HIGH or LOW
- PIN# -- 0 to 13

**Analog OUTPUT**
- 0 to 255
- PIN# -- 3, 5, 6, 9, 10, 11

SHAPE
- set digital pin #13 HIGH

SHAPE
- set analog pin #5 255
Last bit... the buzzer

Final command to know:

```
tone(pin, freq, duration);
```

**pin** – the OUTPUT pin the buzzer is connected to.

**freq** – unsigned int (0 ... 65,535)

**duration** – unsigned long (0 ... 2^32 - 1)
## Musical Notes / Frequencies

<table>
<thead>
<tr>
<th>Note</th>
<th>Frequency (Hz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>C₄</td>
<td>261</td>
</tr>
<tr>
<td>C#₄/Dᵇ₄</td>
<td>277</td>
</tr>
<tr>
<td>D₄</td>
<td>293</td>
</tr>
<tr>
<td>D#₄/Eᵇ₄</td>
<td>311</td>
</tr>
<tr>
<td>E₄</td>
<td>329</td>
</tr>
<tr>
<td>F₄</td>
<td>349</td>
</tr>
<tr>
<td>F#₄/Gᵇ₄</td>
<td>369</td>
</tr>
<tr>
<td>G₄</td>
<td>392</td>
</tr>
<tr>
<td>G#₄/Aᵇ₄</td>
<td>415</td>
</tr>
<tr>
<td>A₄</td>
<td>440</td>
</tr>
<tr>
<td>A#₄/Bᵇ₄</td>
<td>466</td>
</tr>
<tr>
<td>B₄</td>
<td>493</td>
</tr>
<tr>
<td>C₅</td>
<td>523</td>
</tr>
<tr>
<td>C#₅/Dᵇ₅</td>
<td>554</td>
</tr>
<tr>
<td>D₅</td>
<td>587</td>
</tr>
<tr>
<td>D#₅/Eᵇ₅</td>
<td>622</td>
</tr>
<tr>
<td>E₅</td>
<td>659</td>
</tr>
<tr>
<td>F₅</td>
<td>698</td>
</tr>
<tr>
<td>F#₅/Gᵇ₅</td>
<td>739</td>
</tr>
<tr>
<td>G₅</td>
<td>783</td>
</tr>
<tr>
<td>G#₅/Aᵇ₅</td>
<td>830</td>
</tr>
<tr>
<td>A₅</td>
<td>880</td>
</tr>
<tr>
<td>A#₅/Bᵇ₅</td>
<td>932</td>
</tr>
<tr>
<td>B₅</td>
<td>987</td>
</tr>
</tbody>
</table>
World’s cheapest dubstep instrument

Tie both the buzzer and the LED to light sensor… and, create the world’s cheapest dub step instrument.

Play with the frequency – scale it by applying multiplier factors or offsets.
OK? What’s going on under the hood?
Arduino Code, Examples, References, etc...
// the loop routine runs over and over.
void loop() {
  digitalWrite(led, HIGH);  // turn the LED on (HIGH is the voltage level)
  delay(1000);              // put pause for 1 second
  digitalWrite(led, LOW);   // turn the LED off by making the voltage level LOW
  delay(1000);              // put pause for 1 second
}

This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 United States License.
A partnership...
Resources


Additional Resources

http://www.sparkfun.com

http://learn.sparkfun.com

http://www.arduino.cc

http://www.bildr.org
ProtoSnap - Pro Mini
DEV-10889

Description: The ProtoSnap is an Arduino-compatible development platform aimed at teaching the basics of Arduino programming as efficiently as possible. It requires zero assembly, wiring, or soldering, so you can jump right into programming the ProtoSnap to control LEDs, buzzers, light sensors, and more. There’s even a small prototyping space so you can add your own parts!

Once you’ve gotten a firm grasp of the programming, you can snap off the individual components of the ProtoSnap for use in future projects. Those components include an Arduino Pro Mini microcontroller platform and an FTDI Basic Breakout to program it.

In the latest revision, we’ve added some helpful silkscreen to show you how everything is connected.

Included on the Board:

- 1 x Arduino Pro Mini 5V/16MHz
- 1 x FTDI Basic Breakout 5V
- 1 x Buzzer
- 1 x RGB LED
- 1 x Light Sensor
- 1 x Push Button
- 1 x Protoboard

Documents:

- Schematic
- Eagle Files
- Getting Started Guide
- Example Code
SparkFun Tutorials and Projects

- GardenBot:
  www.sparkfun.com/news/598
- High Altitude Balloon:
  www.sparkfun.com/tutorials/180
- GPS Clock:
  www.sparkfun.com/tutorials/47
- Earthquake Data Logger:
  www.sparkfun.com/tutorials/235
Thousands of Amazing ideas...

The Cat Defender...

List of projects at Sparkfun:
http://goo.gl/2M3AM
Questions?
Your to keep...

The little red box, the protosnap Pro Mini, and the cable are yours to keep today.

Please e-mail us with ideas and things that you end up doing at your school.

Thanks!

email: education@sparkfun.com
Feedback?

Before leaving, please fill out a quick survey. Thanks!

http://www.sparkfun.com/WorkshopFeedback