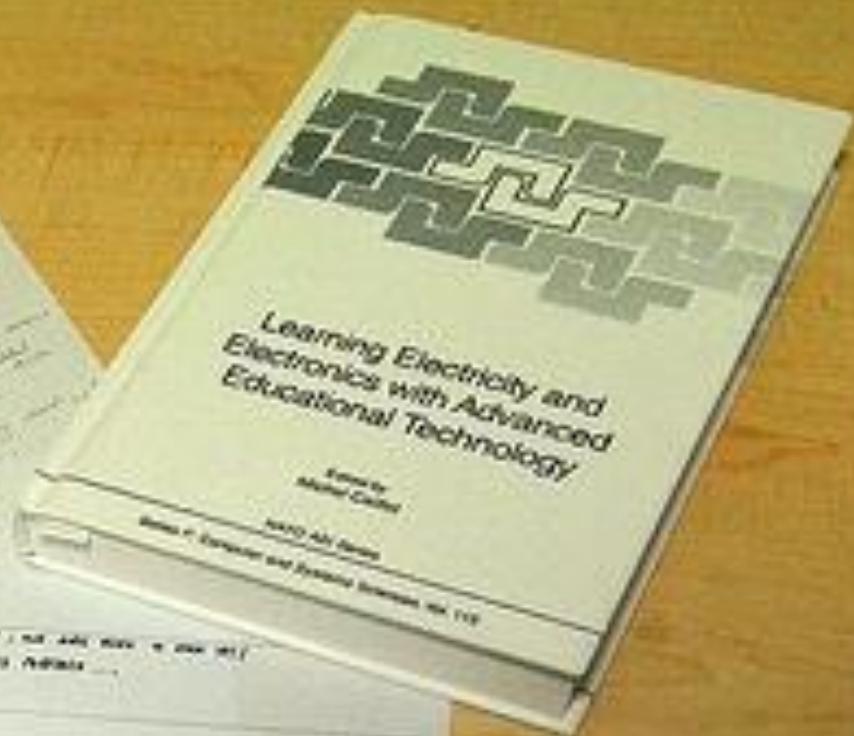


Who are we teaching to...



<http://bit.ly/audiencebvsd2013/>

Arduino for hipsters





<http://bit.ly/bvsdsparkfun2013>

<http://bit.ly/bvsdsparkfun2013pdf>

<http://bit.ly/1d9igxs>



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.



Sharing Ingenuity

WWW.SPARKFUN.COM

This work is licensed under a [Creative Commons License](#).



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.





KICKSTARTER

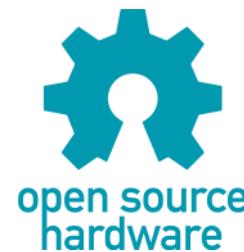
Motivation



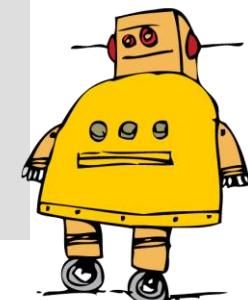
Make:
technology on your time



the
MAKER
MOVEMENT



indiegogo



instructables.com
THE WORLD'S BIGGEST SHOW & TELL

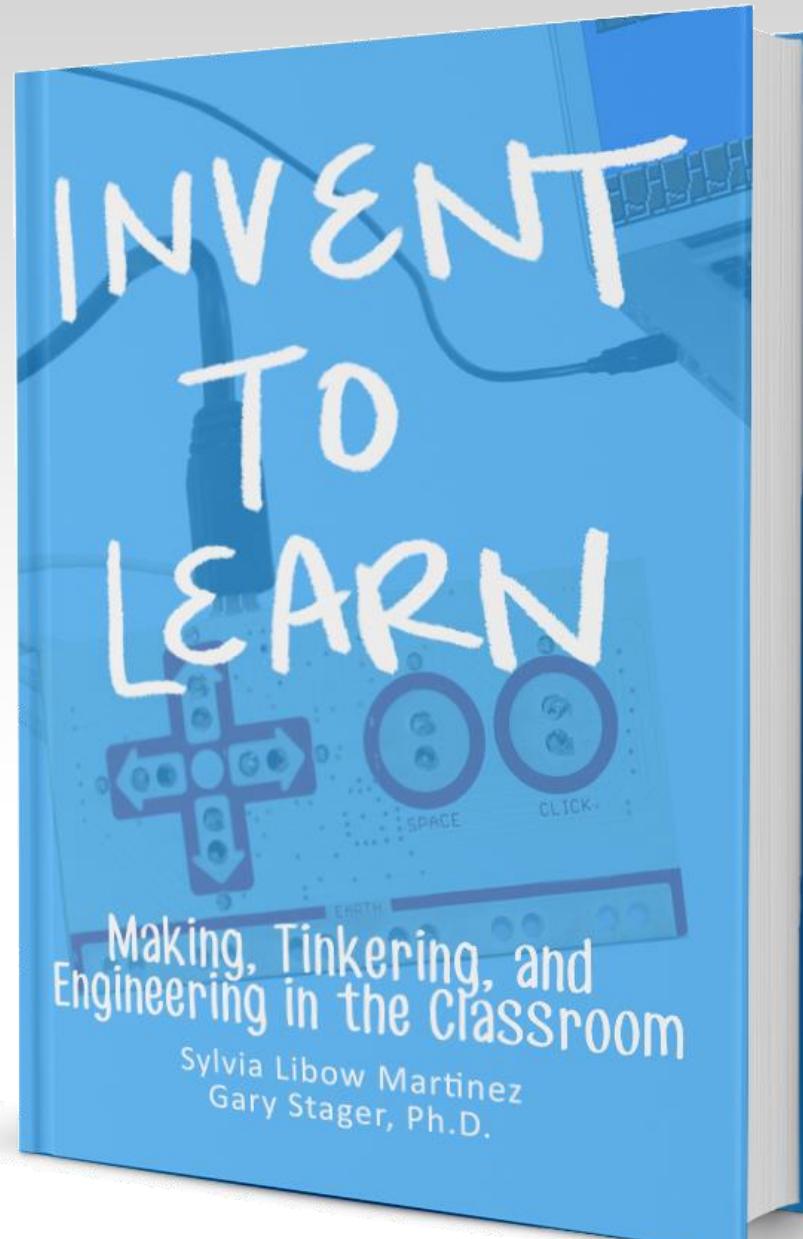
H A R V A R D | B U S I N E S S | S C H O O L



The “IKEA Effect”: When Labor Leads to Love

**Michael I. Norton
Daniel Mochon
Dan Ariely**





Tinkering -- A Mindset For Learning

“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.”

Martinez, Sylvia Libow., and Gary Stager. *Invent to Learn: Making, Tinkering, and Engineering in the Classroom*. Torrance, CA: Constructing Modern Knowledge, 2013. N. p.31-32.



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.”

Martinez, Sylvia Libow., and Gary Stager. *Invent to Learn: Making, Tinkering, and Engineering in the Classroom*. Torrance, CA: Constructing Modern Knowledge, 2013. N. p.31-32.



Engineering as Solving Problems...

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

Martinez, Sylvia Libow., and Gary Stager. *Invent to Learn: Making, Tinkering, and Engineering in the Classroom*. Torrance, CA: Constructing Modern Knowledge, 2013. N. p.31-32.



I want _____ to _____.

What is *it* all about?

**Stop Following
Directions**

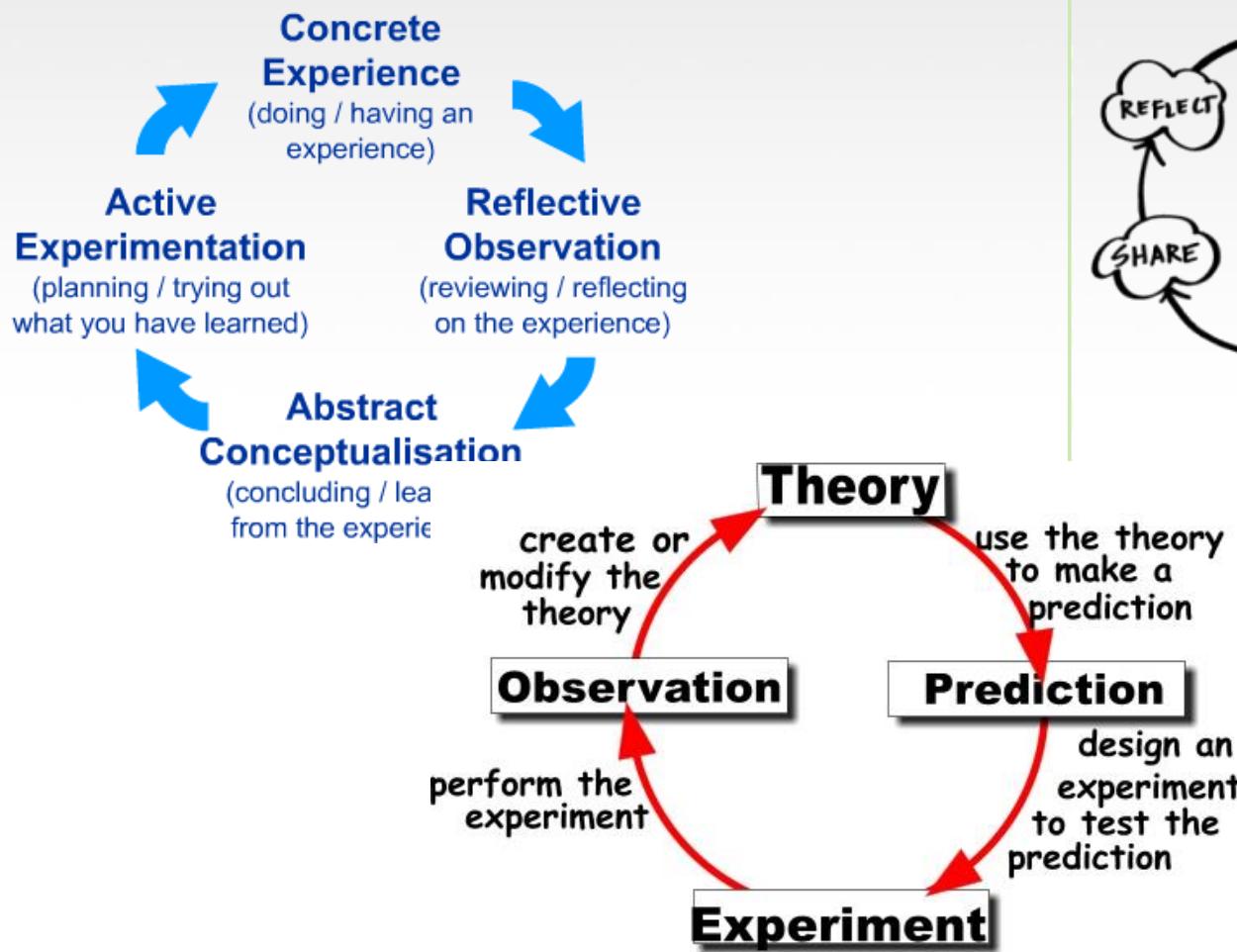
&

**Start
Engineering**



The Learning Cycle

Time to play, reflect, question and redesign



Computer Science Education Act (CSEA)



Search the site... Find us on social networks Email Updates

**COMPUTING
IN THE CORE**

Advocating for K-12 Computer Science Education

ABOUT ISSUES & SOLUTIONS FACTS & RESOURCES NEWSROOM EVENTS GET INVOLVED

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](#).

SHARE THIS



1k



Sharing this with your friends helps further awareness of computer science education. Thank you!

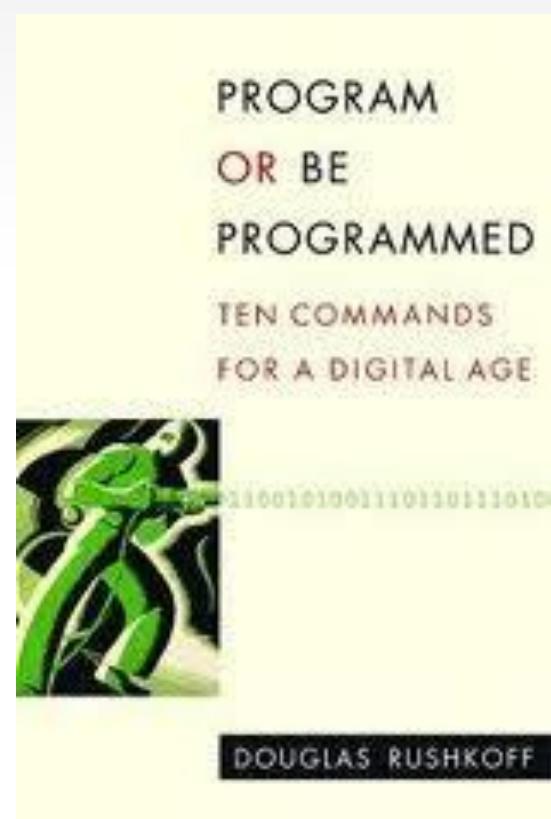
Tell Your Members of Congress to Support Computer Science Education

[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...



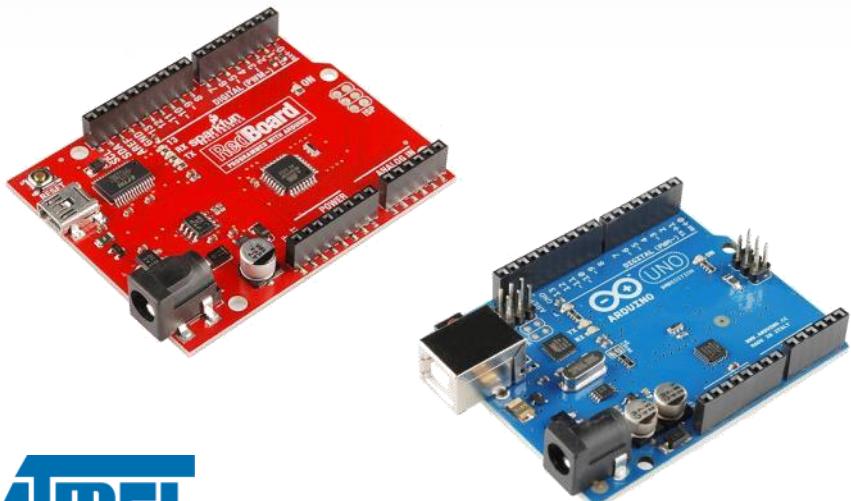
What is Arduino...

(circa 2005) Small, low-cost micro-controller (mini-computer).

Describes both the hardware (board) and the programming language.

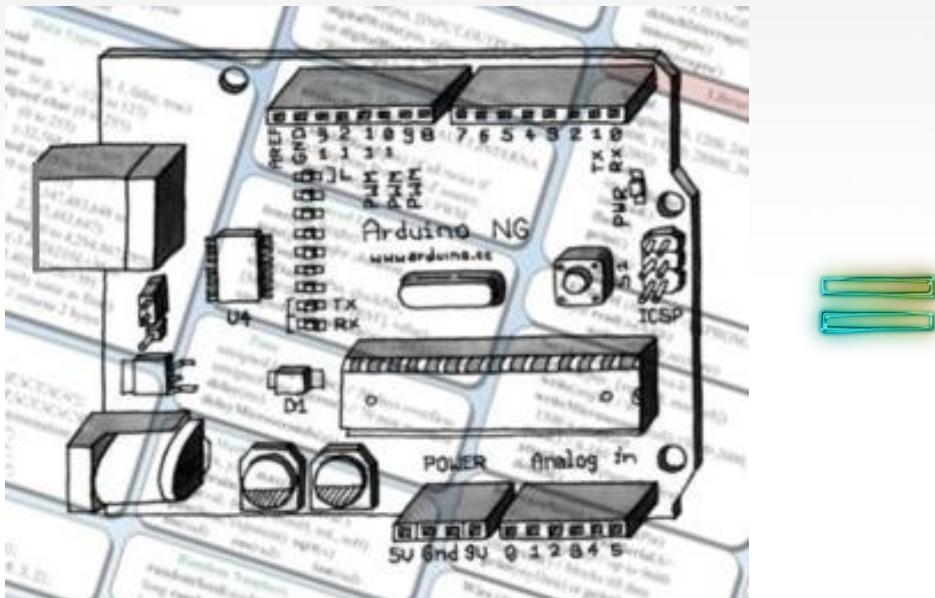
Developed off the Atmel ATmega328 chip.

Designed with Artists & Non-Engineers in mind...

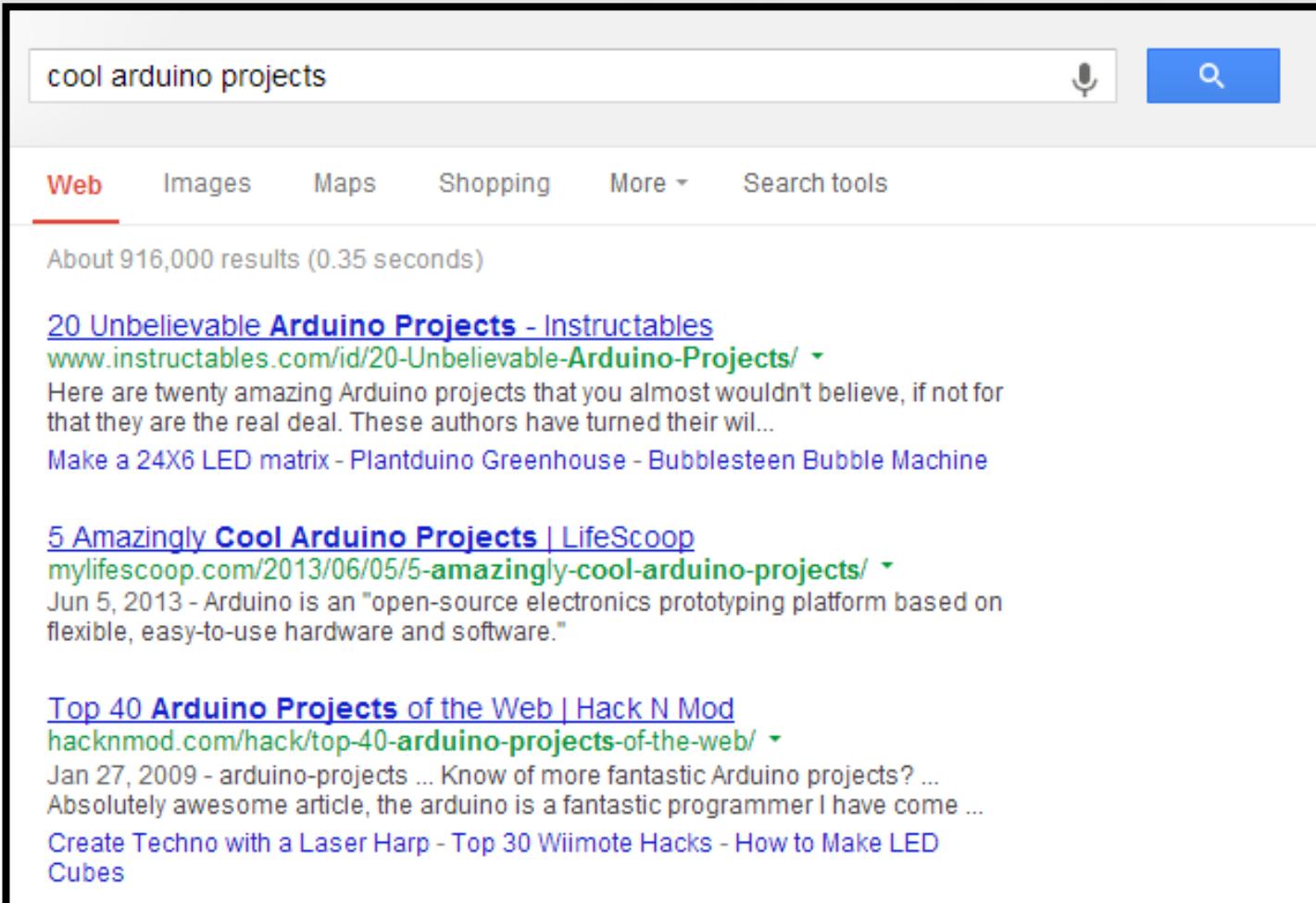


But... what is it???

Think of this as a mini-brain...



Nearly a million ideas...



cool arduino projects

Web Images Maps Shopping More Search tools

About 916,000 results (0.35 seconds)

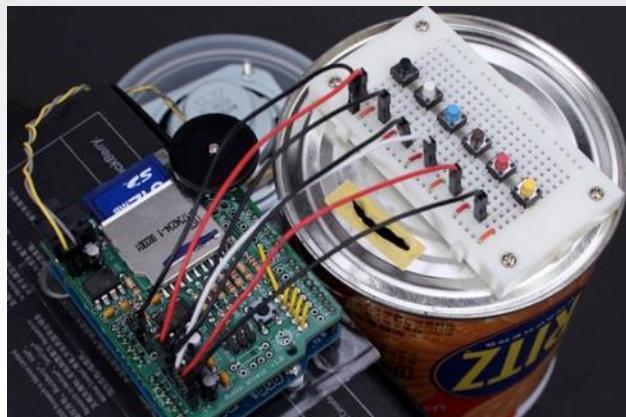
[20 Unbelievable Arduino Projects - Instructables](#)
www.instructables.com/id/20-Unbelievable-Arduino-Projects/ ▾
Here are twenty amazing Arduino projects that you almost wouldn't believe, if not for that they are the real deal. These authors have turned their wil...
Make a 24X6 LED matrix - Plantduino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

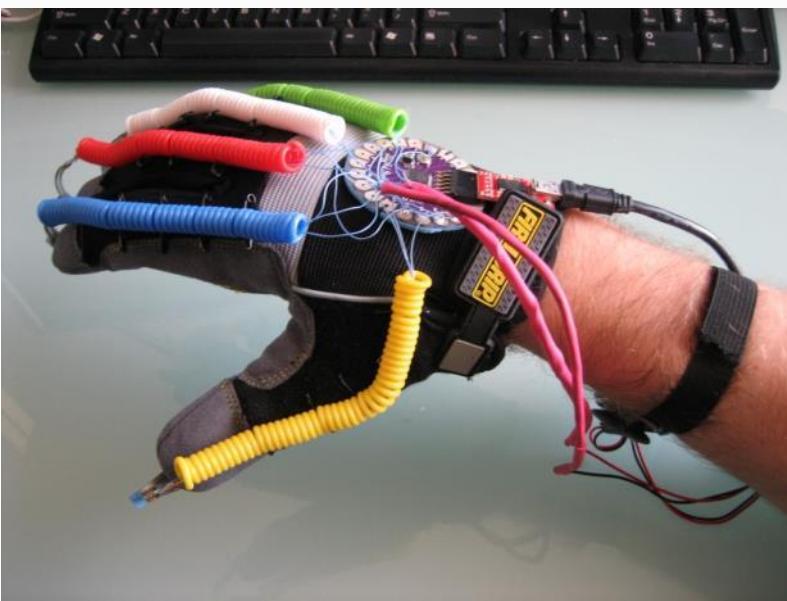
[5 Amazingly Cool Arduino Projects | LifeScoop](#)
mylifescoop.com/2013/06/05/5-amazingly-cool-arduino-projects/ ▾
Jun 5, 2013 - Arduino is an "open-source electronics prototyping platform based on flexible, easy-to-use hardware and software."

[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 Wiimote Hacks - How to Make LED Cubes



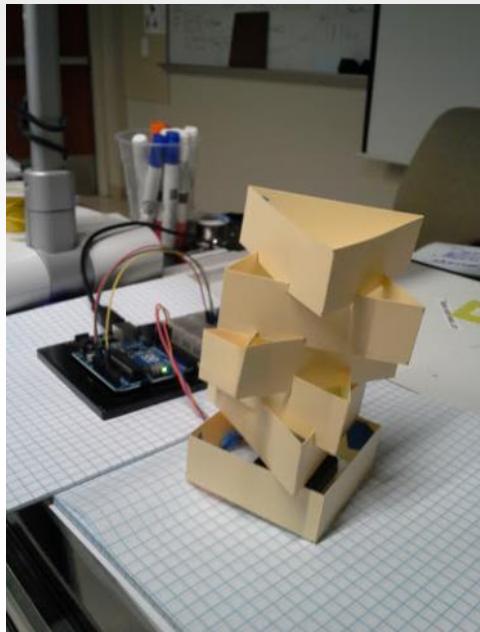
Some cool projects





Project: Mood Lamp / Light Sculpture

sparkfun
ELECTRONICS



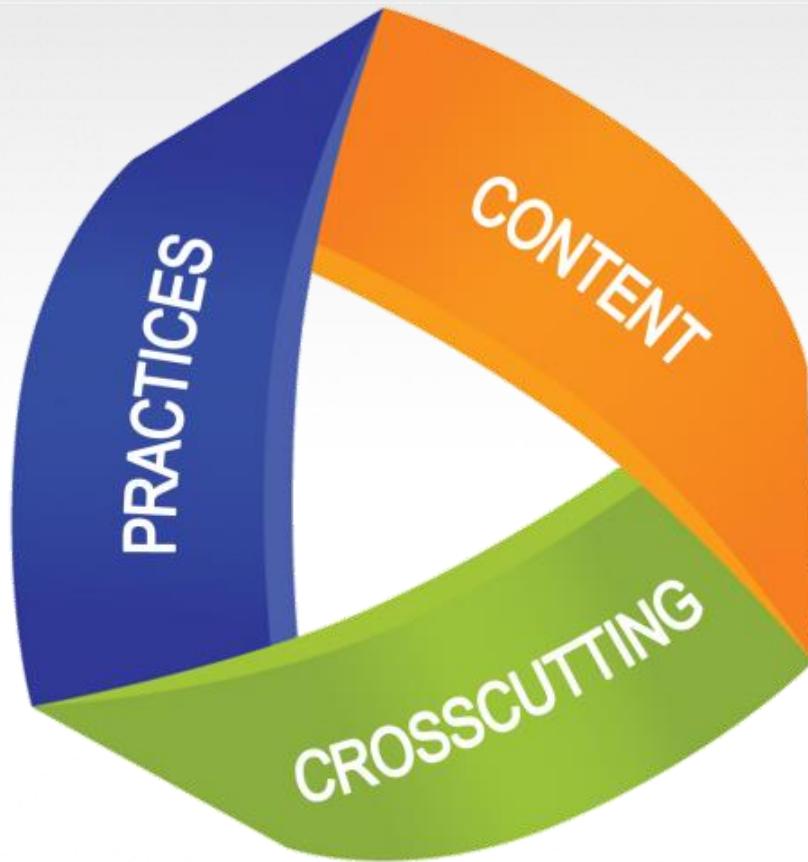
This work is licensed under a Creative Commons Attribution-ShareAlike 3.0 United States License.

Massimo Banzi TED Talk

2012 Edinborough



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.

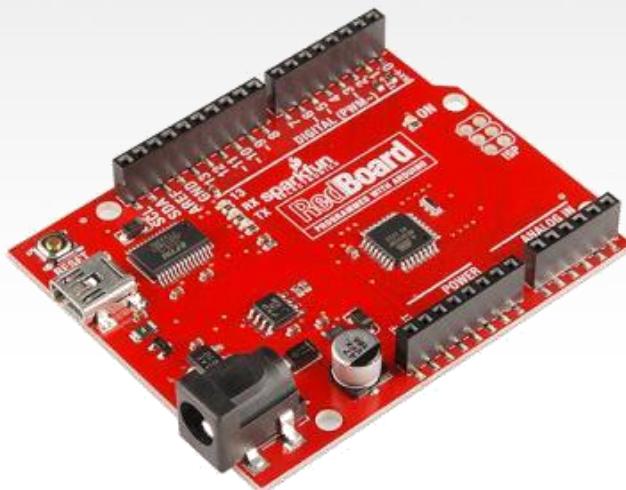
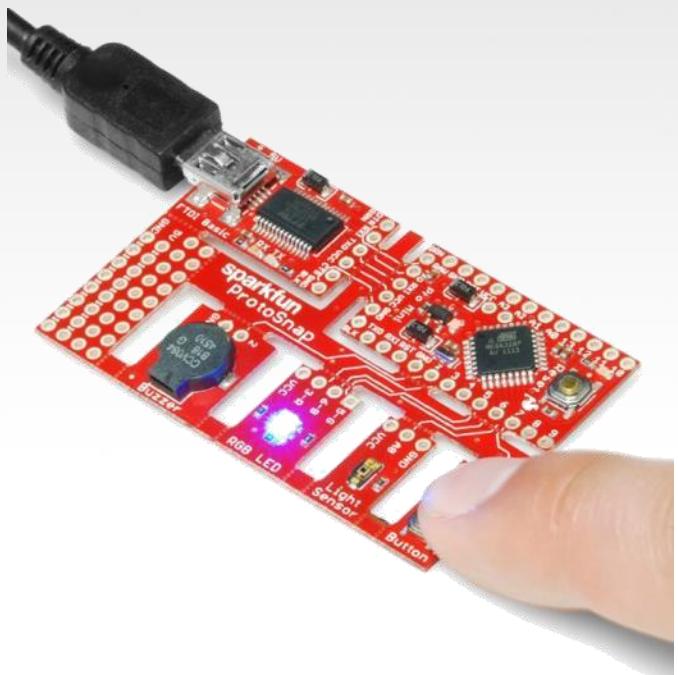


Agenda

- 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



A screenshot of a web browser window displaying the ArduBlock homepage. The browser has a blue circuit board-style header bar with various icons. The main content area has a yellow header with the title "ArduBlock" and subtitle "A Graphical Programming Language for Arduino". Below the header is a large blue wavy graphic. A navigation bar at the bottom includes "Links", "About", and "Getting Started with ArduBlock", with the "Getting Started with ArduBlock" button highlighted by a red rectangle. A news banner below the navigation bar reads "[20130712 Release] Update with Li... by david | 07-12-2013 >". To the left of the banner is a screenshot of the ArduBlock software interface showing a vertical stack of blocks. The footer contains a Creative Commons Attribution-ShareAlike 3.0 United States License logo and a link to the license page.

ArduBlock | A Graphical P...

blog.ardublock.com

SFE Sparkle Bugzilla Main Page Event Form Bookmarklet Toodledo Processing & Interact... Tour Other bookmarks

ArduBlock

A Graphical Programming Language for Arduino

Links About Getting Started with ArduBlock

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

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

blog.ardublock.com/getting-started-ardublock-hardublock/

Creative Commons Attribution-ShareAlike 3.0 United States License

A screenshot of a web browser window displaying the ArduBlock homepage. The browser has a blue circuit board background. The address bar shows the URL blog.ardublock.com/en/getting-started-ardublockhardublock/. The page title is "ArduBlock" with the subtitle "A Graphical Programming Language for Arduino". Below the title is a large blue wavy graphic. A navigation bar at the bottom includes "Links", "About", and "Getting Started with ArduBlock". The main content area features a purple header with the text "Getting Started with ArduBlock" and a "Page" link. Below this is a section titled "DOWNLOAD Ardublock" with three numbered steps: 1. Download ardublock-all.jar ArduBlock, 2. In Arduino IDE, open menu "Arduino" -> "Preferences", 3. Find "Sketchbook location.". An inset image shows the "Sketchbook location:" preference in the Arduino IDE Preferences dialog, set to "/Users/toyhouse/Documents/Arduino". A Creative Commons Attribution-ShareAlike 3.0 United States License logo is at the bottom.

Getting Started with ArduBlock

Page

DOWNLOAD Ardublock

1. Download ardublock-all.jar ArduBlock
2. In Arduino IDE, open menu "Arduino" -> "Preferences"
3. Find "Sketchbook location."

Sketchbook location:
/Users/toyhouse/Documents/Arduino

Browse

BY SA

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

Installing ArduBlock

Create a folder structure:

**Note: the folder names
are case-sensitive!!!**

tools\ArduBlockTool\tool

tool → ArduBlockTool → tool

under the **\My Documents\Arduino** folder

Save \ Move the **ArduBlock.jar** file here.



General Rules

Raise your hand if you have a question.

Be respectful.

When I raise my hand...

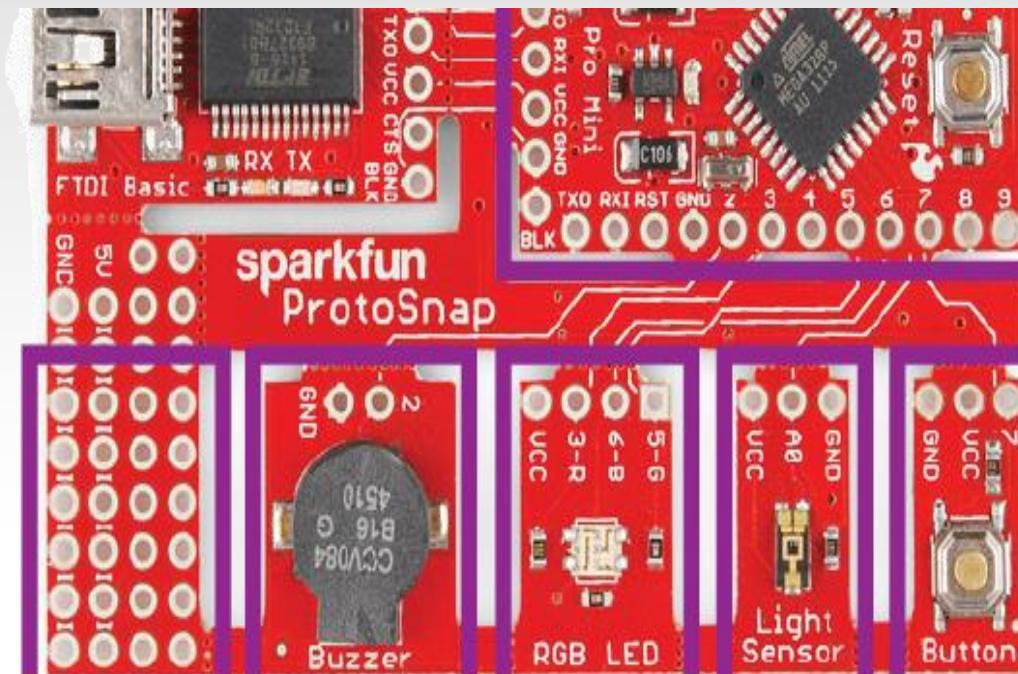
Share your findings with the people near you.

Poke around, experiment, and have fun...

Imagine, create, and play!!!



The ProtoSnap ProMini Board



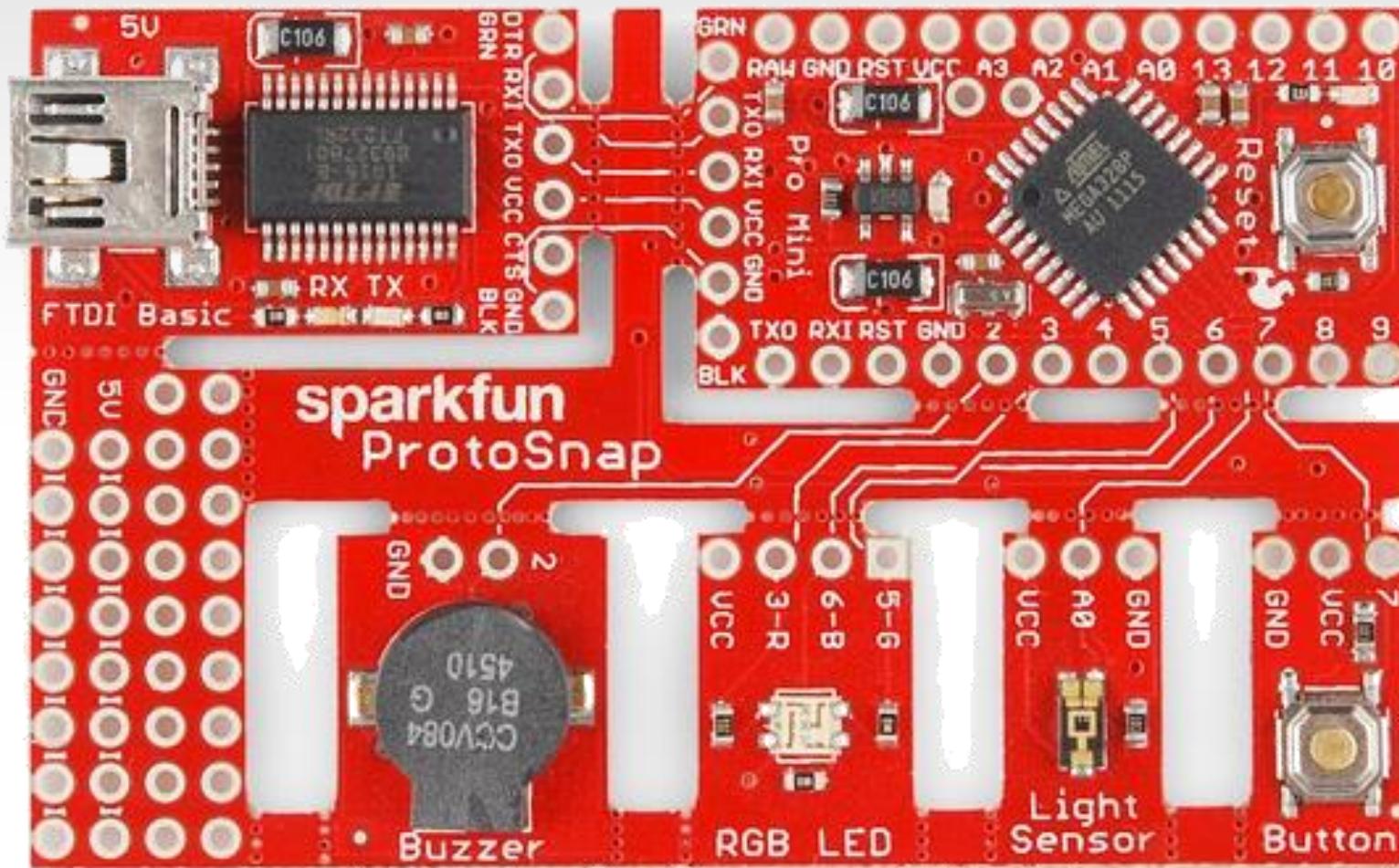
|
|
Buzzer
Prototyping space

|
|
Button
Light Sensor
Red, Green, Blue LED



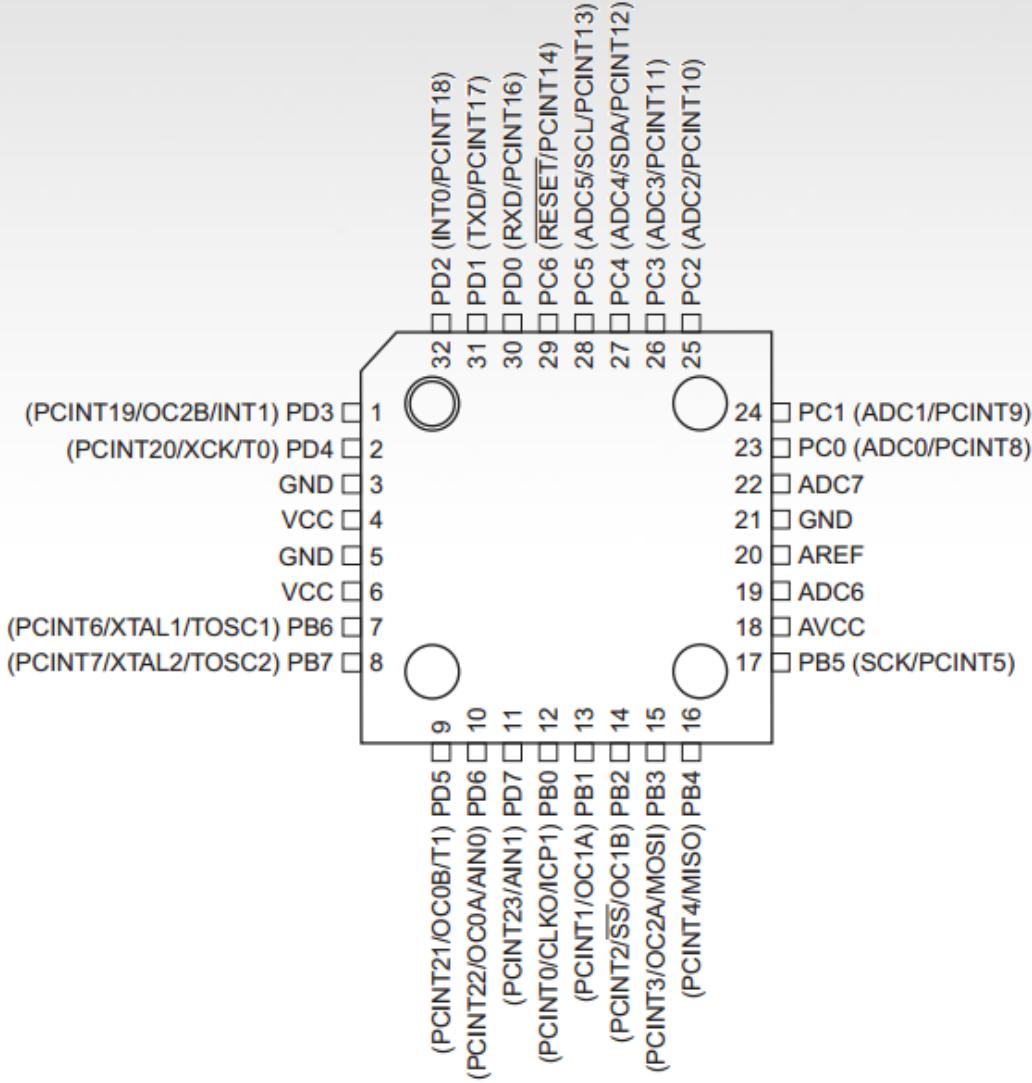
Pin Connections

Where you attach sensors, LEDs, button switches, etc...



Actual Pin-outs

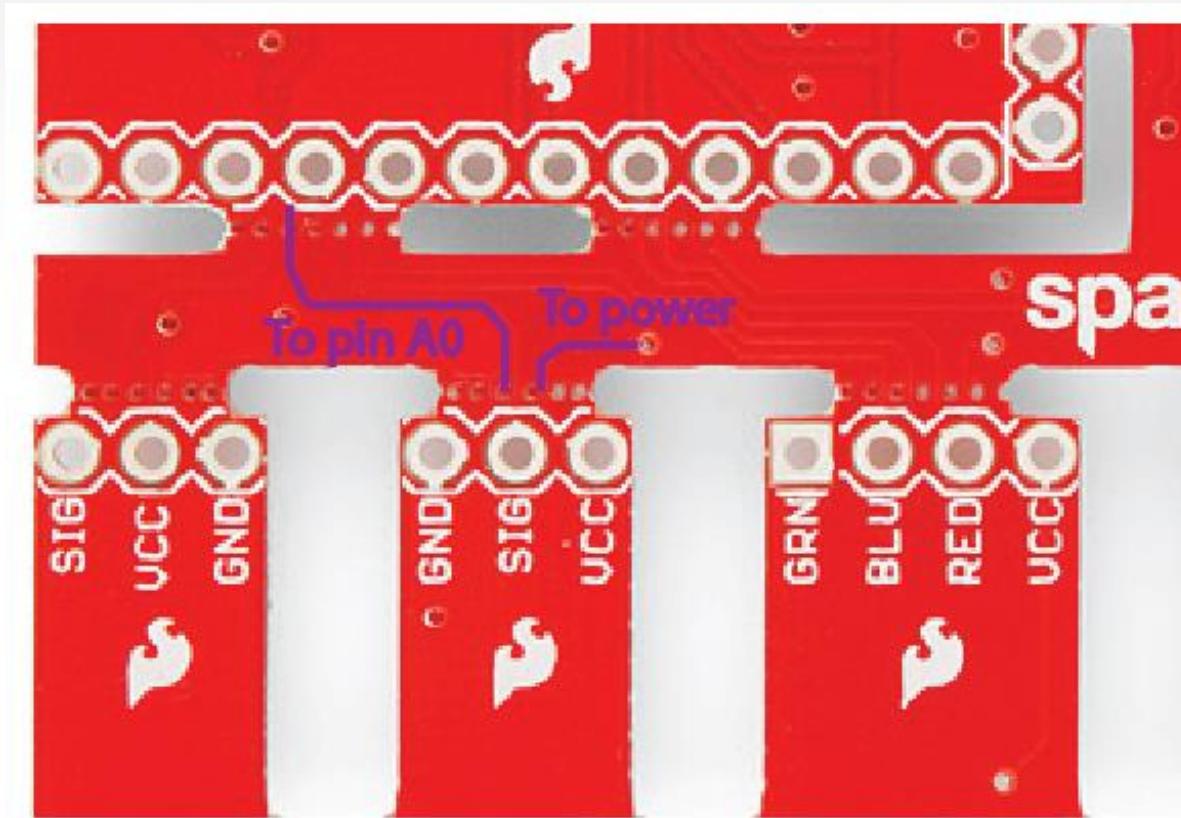
TQFP Top View



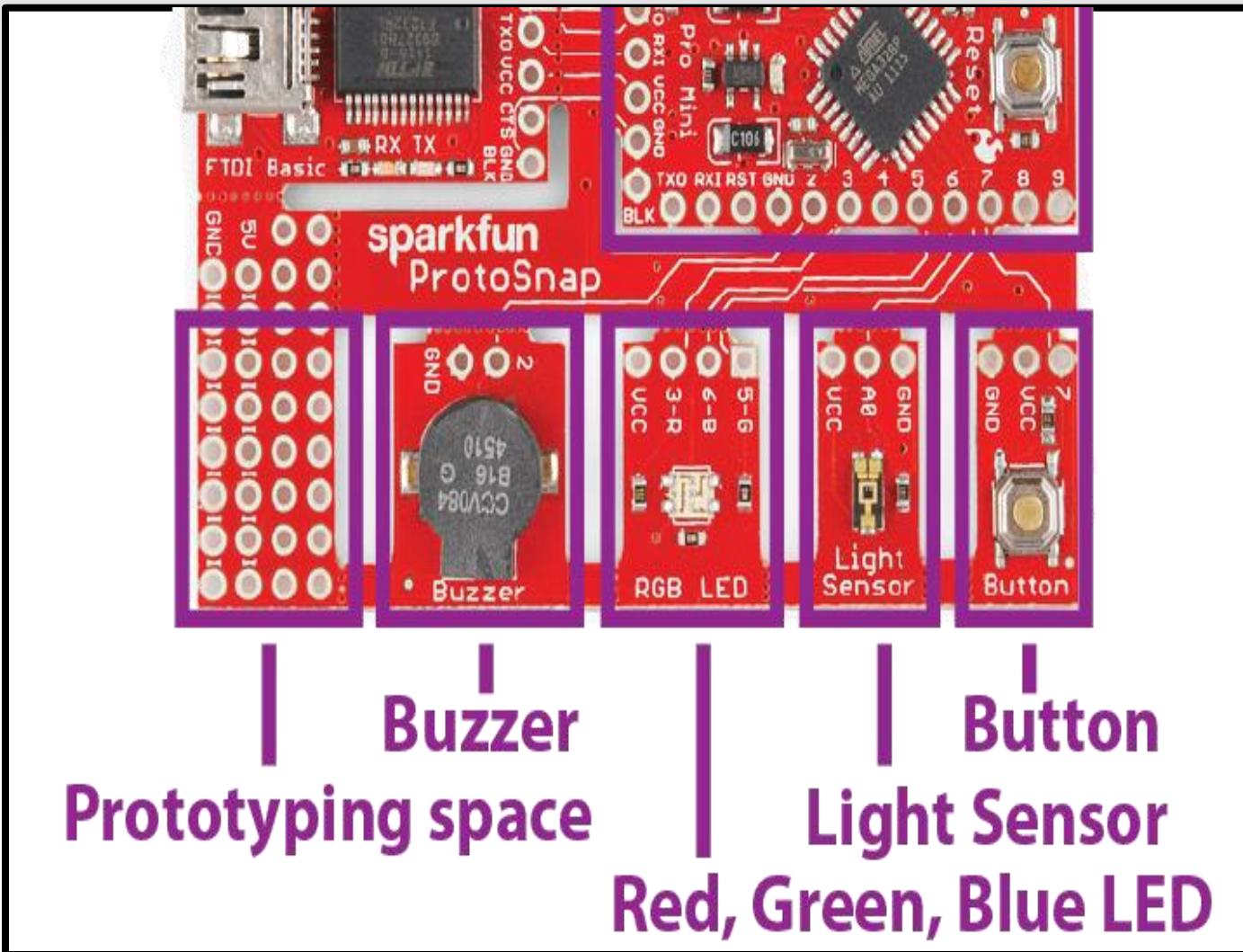
Flip the Board Over

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



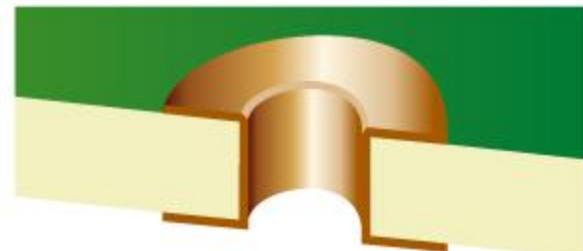
Notes on Vocabulary

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 →



No Circuit building, no fuss



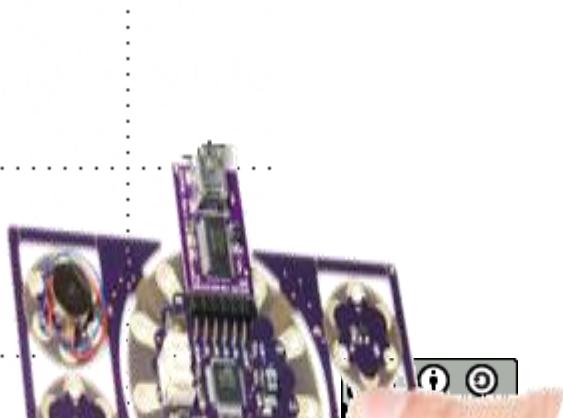


Startup Arduino IDE

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)

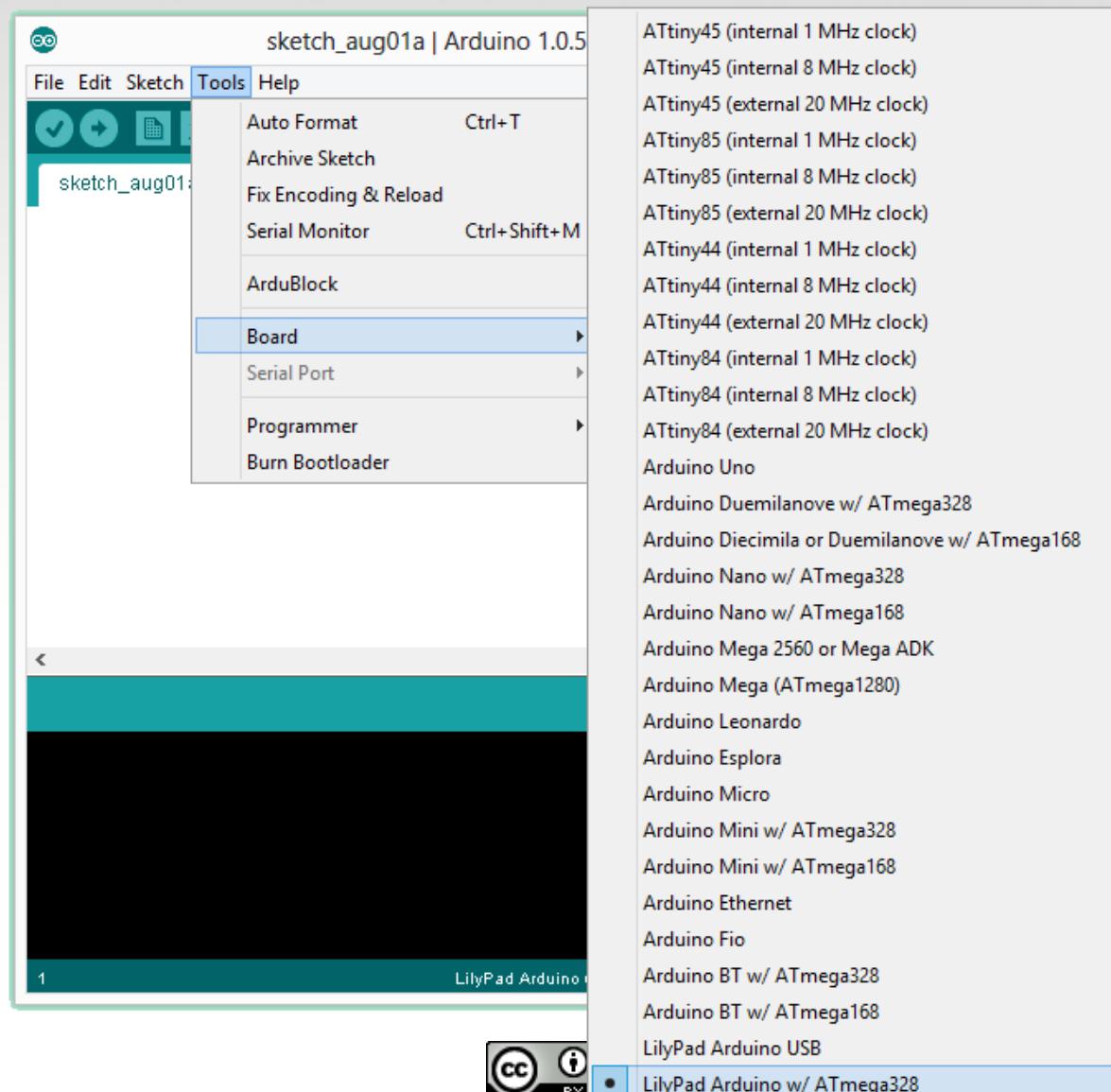
2. Setup COM Port

PC – Highest **COM #**

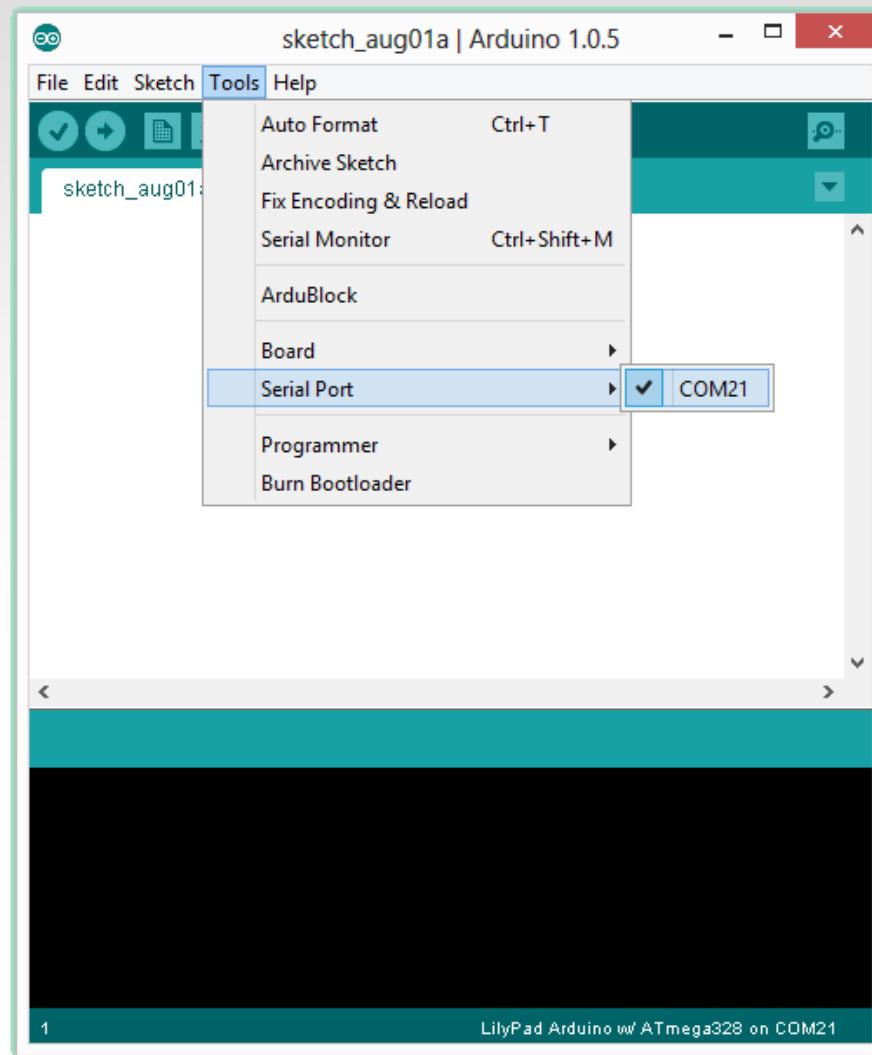
Mac – **/dev/tty.usbserial-A#####xXx**



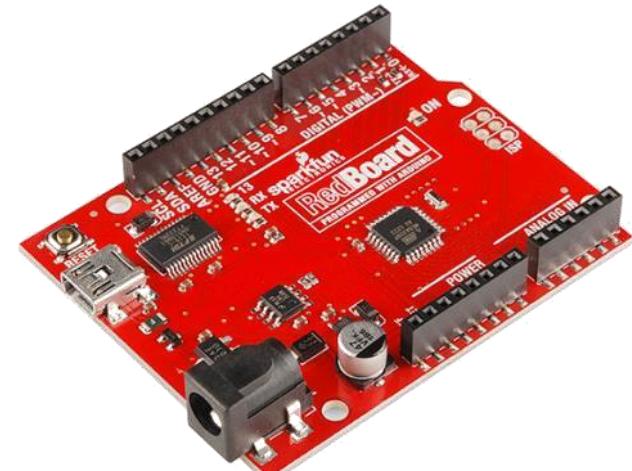
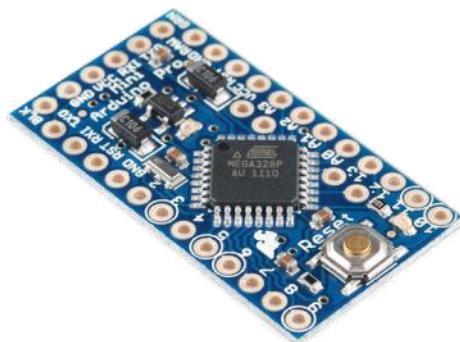
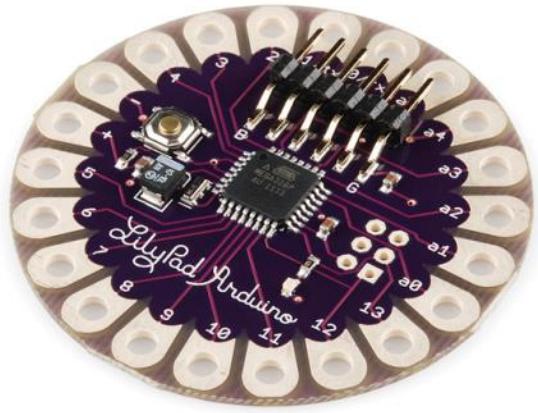
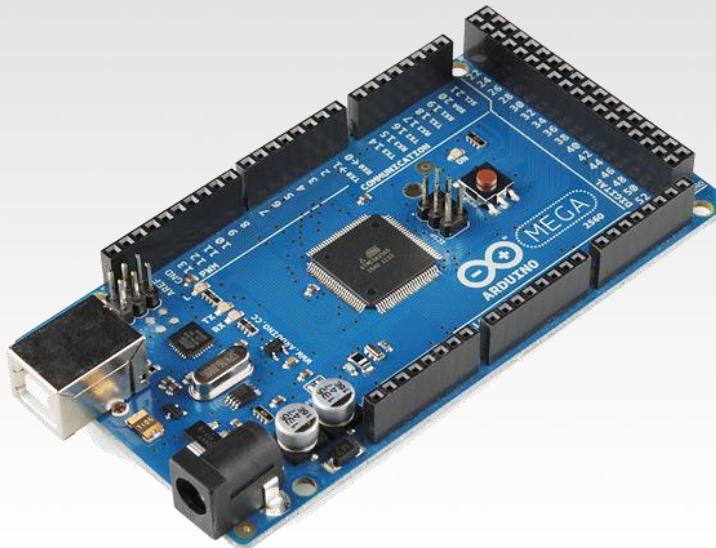
Settings: Tools → Board



Settings: Tools → Serial Port

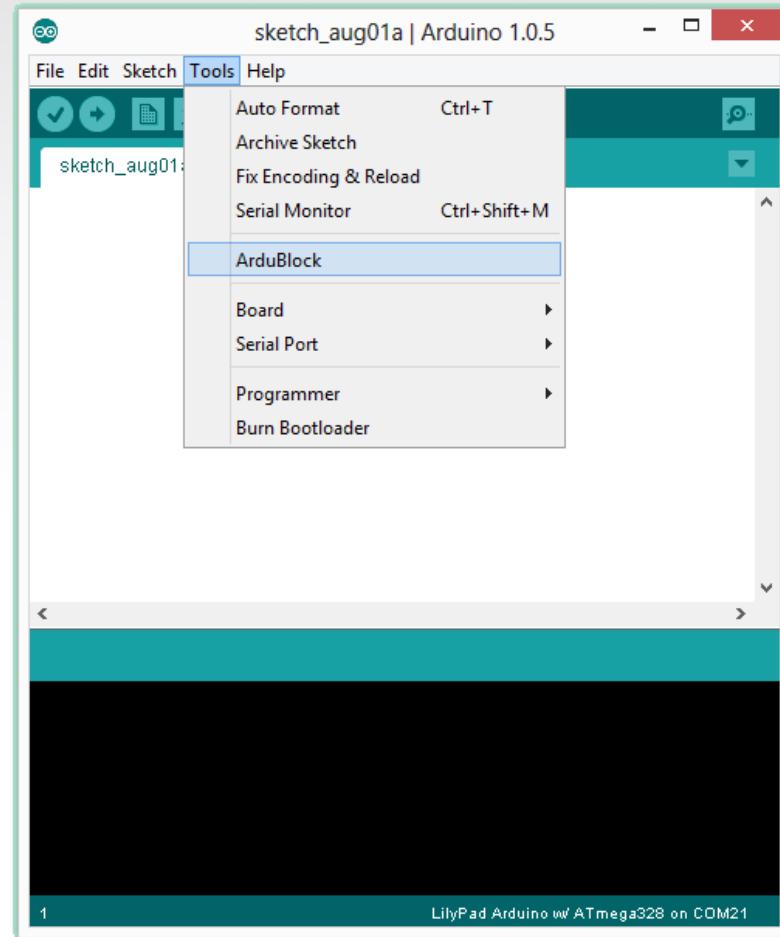


Other Board Types



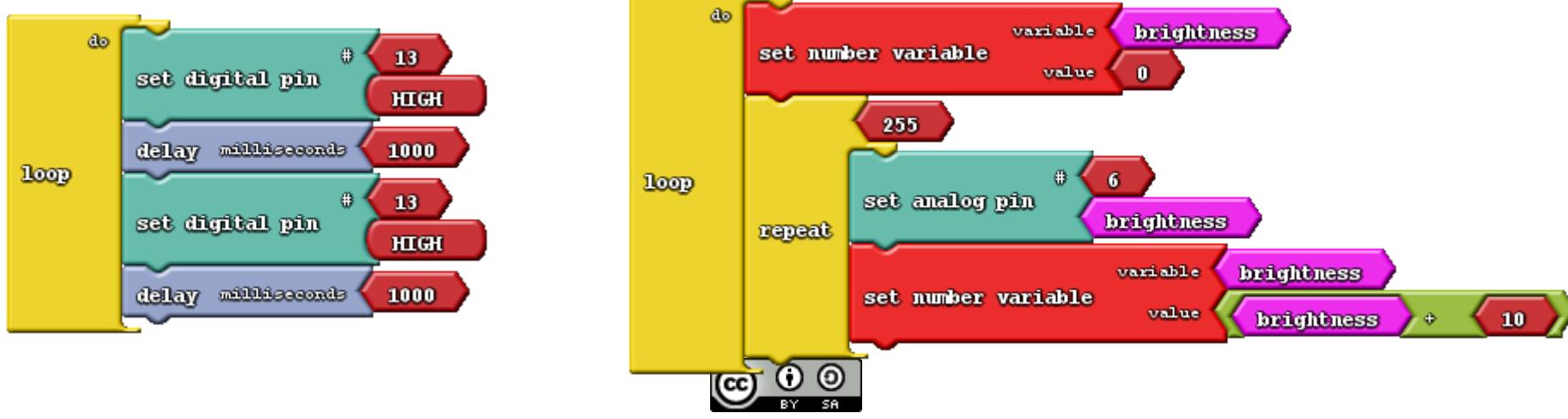
Start ArduBlock

Tools → ArduBlock



ArduBlock

Introduction to Programming
 Scratch-like interface
 Clear transition to text-based coding

The image shows two Scratch scripts side-by-side. The left script is a direct translation of the ArduBlock code into Scratch blocks. The right script shows how the ArduBlock blocks are converted into standard Scratch text-based coding.

Left Script (ArduBlock blocks):

```

loop
  do
    set digital pin #13 to [HIGH v]
    delay milliseconds [1000 v]
    set digital pin #13 to [HIGH v]
    delay milliseconds [1000 v]
  end
end

```

Right Script (Scratch text-based coding):

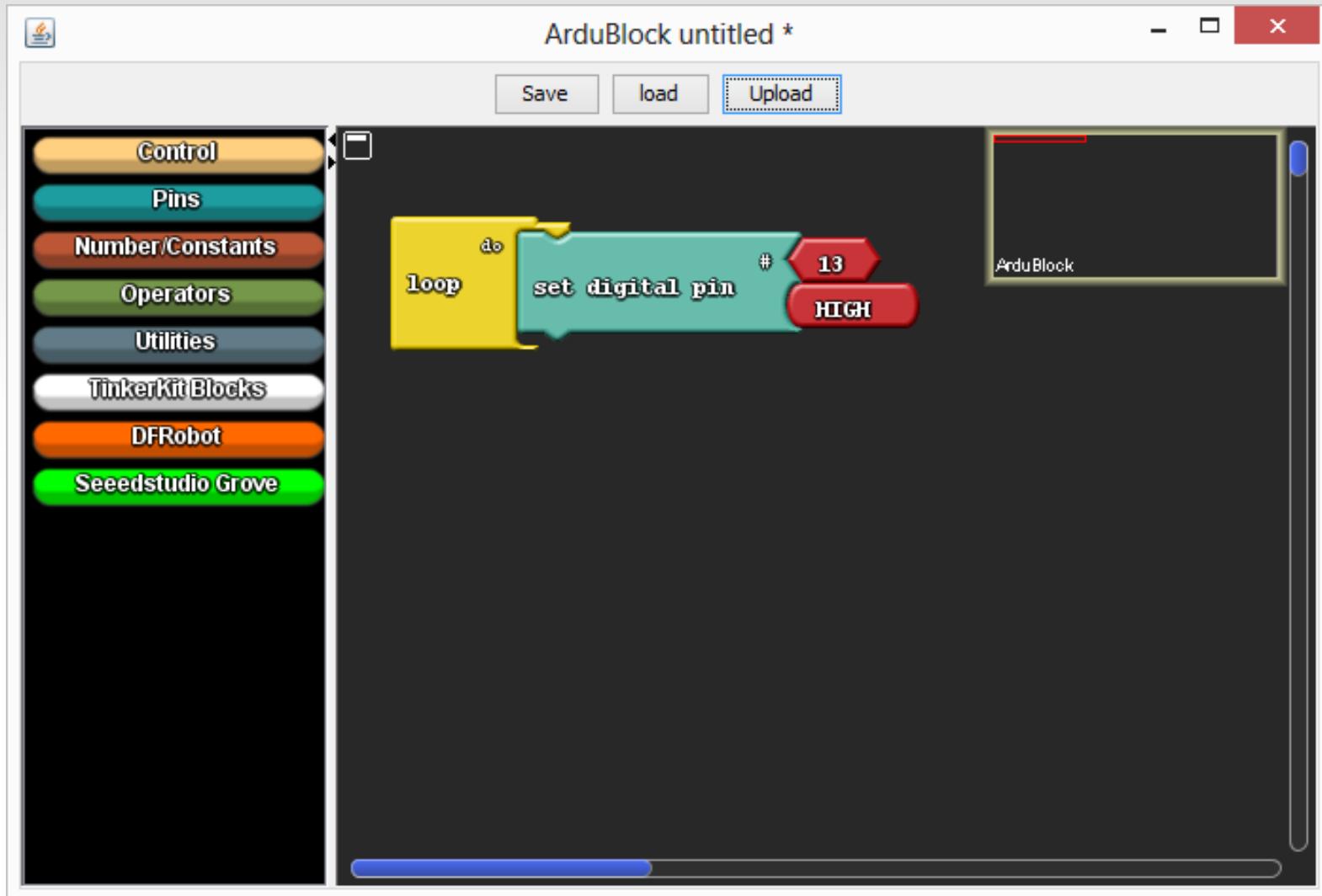
```

when green flag clicked
repeat (6)
  set brightness variable to (0 v)
  do
    set brightness variable to (brightness + (10 v))
    set analog pin #6 to [brightness v]
    set brightness variable to (brightness v)
    delay (255 ms)
  end
end

```

CC BY SA

Lesson #0 – Test Upload



Check for Status Messages



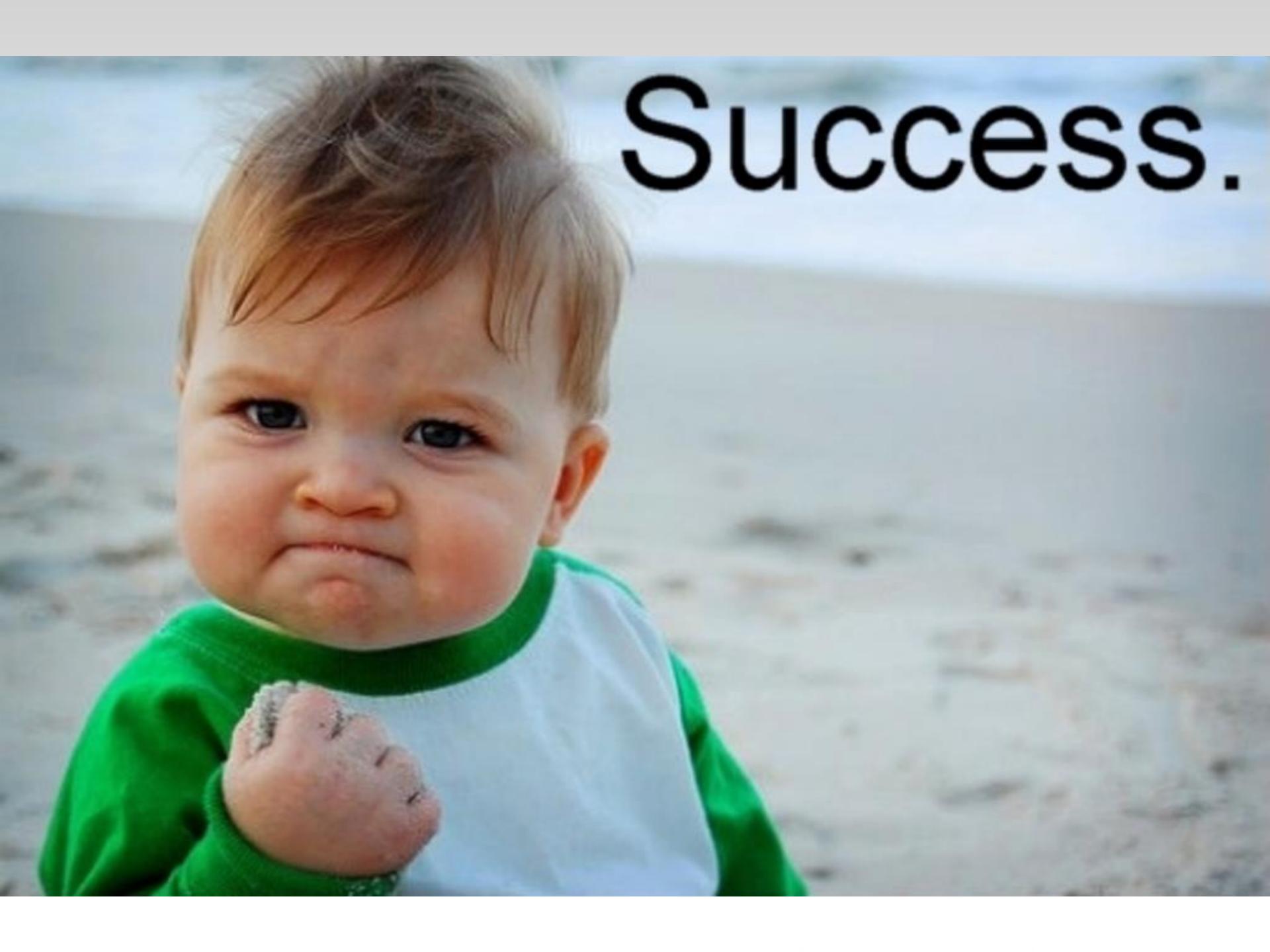
The image shows the Arduino IDE interface. The top bar displays "sketch_aug01a | Arduino 1.0.5". The menu bar includes File, Edit, Sketch, Tools, and Help. Below the menu is a toolbar with icons for upload, refresh, and other functions. The main code editor window contains the following sketch:

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

In the bottom right corner of the IDE, there is a terminal window displaying the results of a recent upload. A red box highlights this terminal window. The text in the terminal is:

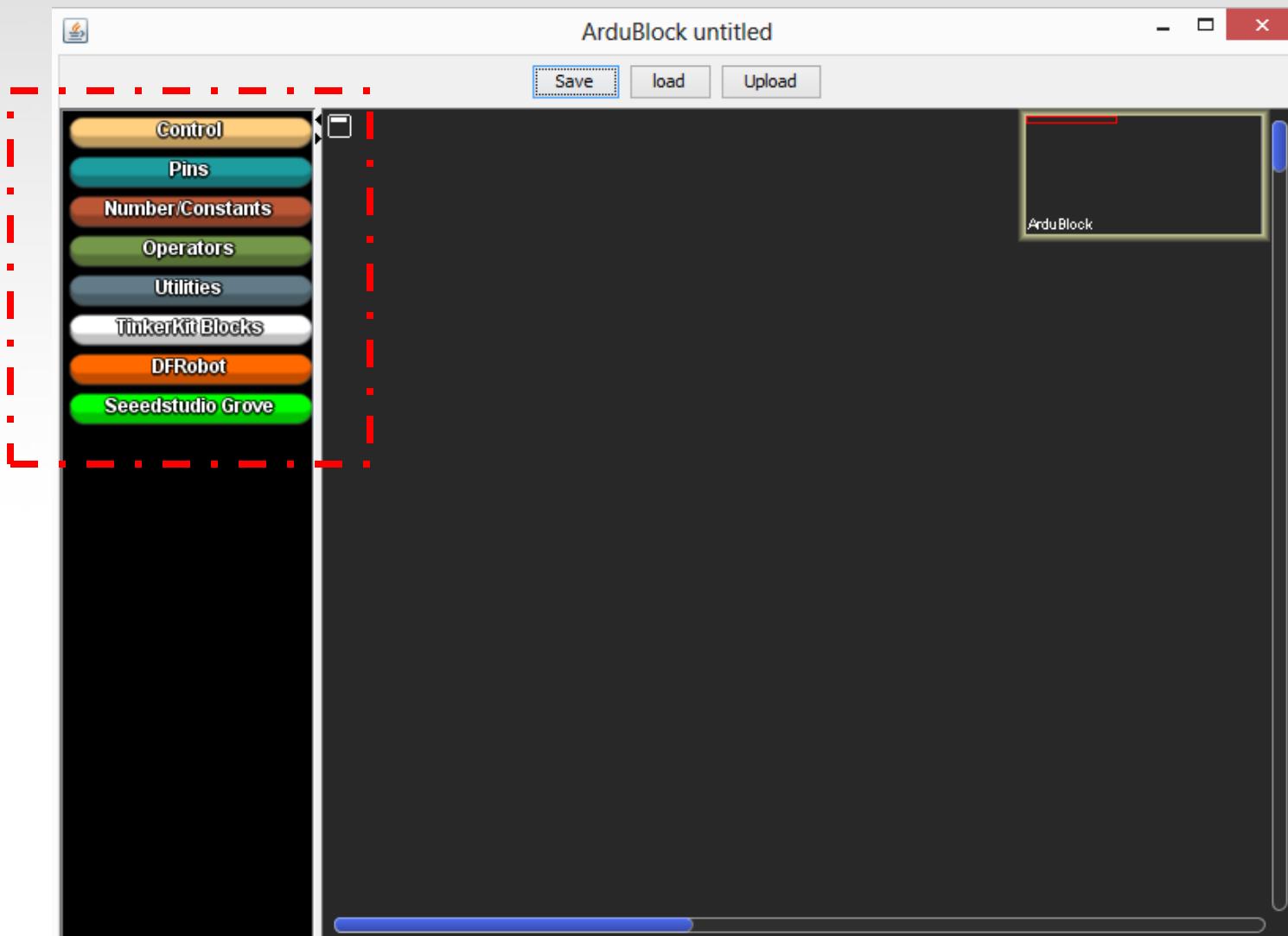
```
Done uploading.  
Binary sketch size: 1,076 bytes  
(of a 32,256 byte maximum)
```

At the very bottom of the screen, the status bar shows "Arduino Uno - COM31".

A close-up photograph of a baby with light brown hair, wearing a green and white long-sleeved shirt. The baby is looking directly at the camera with a neutral expression. The background is a blurred beach scene with sand and water. Overlaid on the upper right side of the image is the word "Success." in a large, bold, black sans-serif font.

Success.

Block Organization in ArduBlock



Block Organization Control



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

number var name

digital var name

set number variable

variable var name

value 0

set digital var

variable variable name

value HIGH

message

glue

glue



Let's get to hacking...

Lesson #1 – Blinking an LED

“Hello World” of Physical Computing

Pseudo-code – how should this work?



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



Using Variables

Numbers/Constants

Variable is just a placeholder for a value.

Notice the different shapes?

number var name

digital var name

set number variable

variable

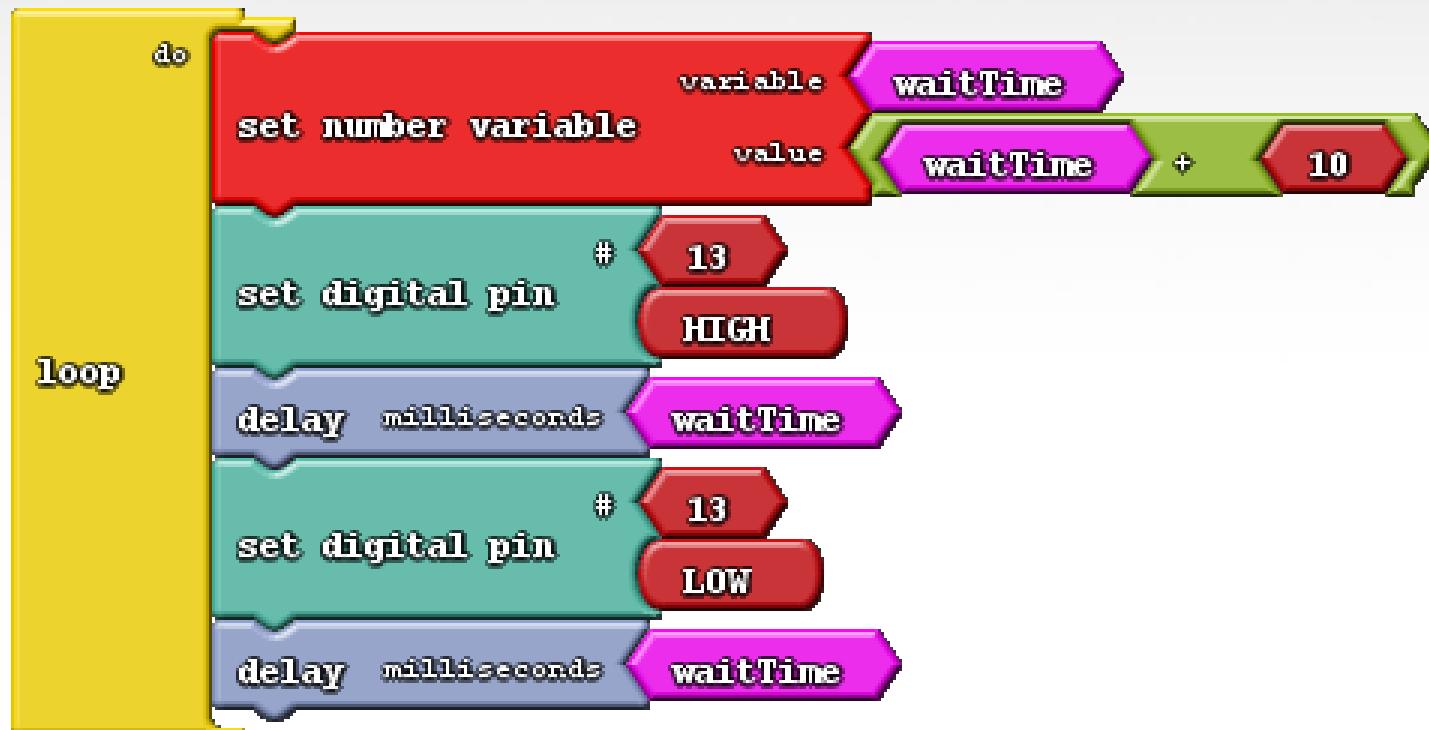
value

set digital var

variable

value

Example: Did you come up with something like this?



Serial Monitor



**Serial print
command**

**Any
message**

“glue”

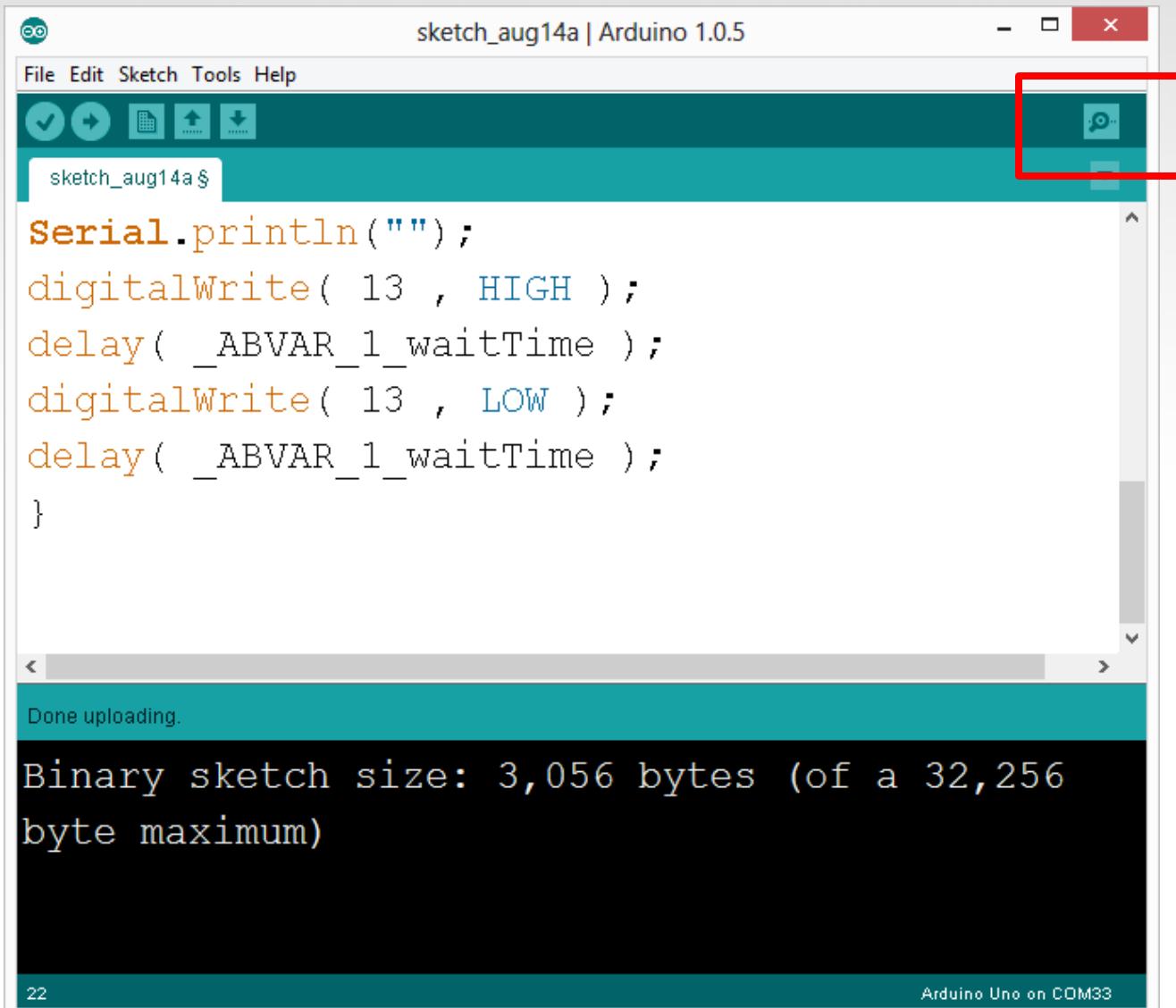
Data

Add this block... so that we can watch the variable value



Upload & Click back to Arduino

Open the Serial Monitor



The screenshot shows the Arduino IDE interface with the following details:

- Title Bar:** sketch_aug14a | Arduino 1.0.5
- Toolbar:** File, Edit, Sketch, Tools, Help
- Sketch Area:** Contains the following code:

```
Serial.println("");
digitalWrite( 13 , HIGH );
delay( _ABVAR_1_waitTime );
digitalWrite( 13 , LOW );
delay( _ABVAR_1_waitTime );
}
```
- Upload Button:** Located in the toolbar, highlighted with a red box.
- Status Bar:** Done uploading.
Binary sketch size: 3,056 bytes (of a 32,256 byte maximum)
Arduino Uno on COM33

Conditional Statements

If()... else...



test conditions – aka
Boolean



Greater Than



Less Than



Equal To



Greater or Equal



Less or Equal



Not Equal to

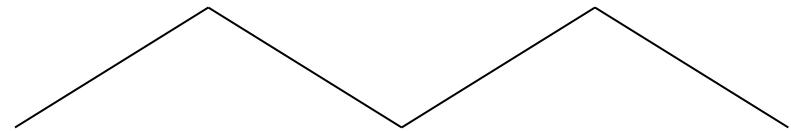
**Aside from just blinking
(on and off), what else
might we want to do??**



Understanding: Analog vs. Digital

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

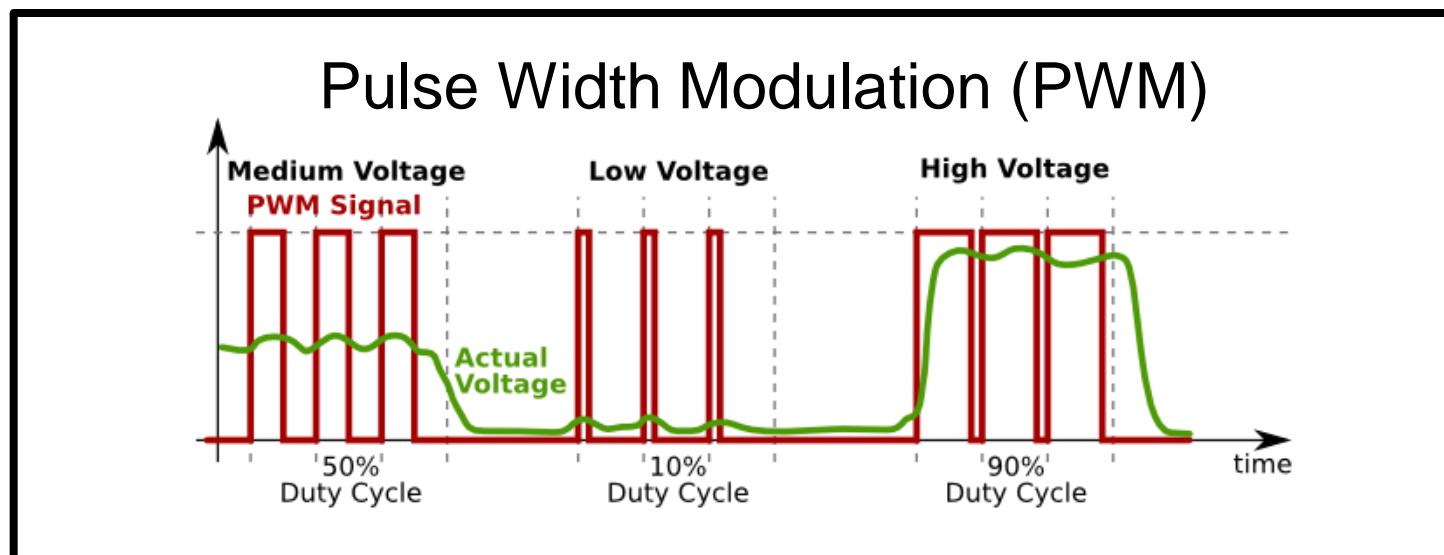
An Analog signal is anything that can be a full range of values. What are some examples? Think of like a ramp or a hill.



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



In ArduBlock

Digital OUTPUT

HIGH or LOW

PIN# -- 0 to 13

Analog OUTPUT

0 to 255

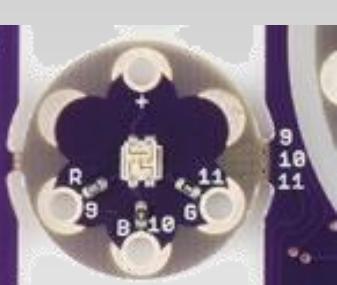
PIN# -- 3, 5, 6, 9, 10, 11

SHAPE



SHAPE





Color Mixing with 3 LEDs analogWrite()

sparkfun
ELECTRONICS



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?

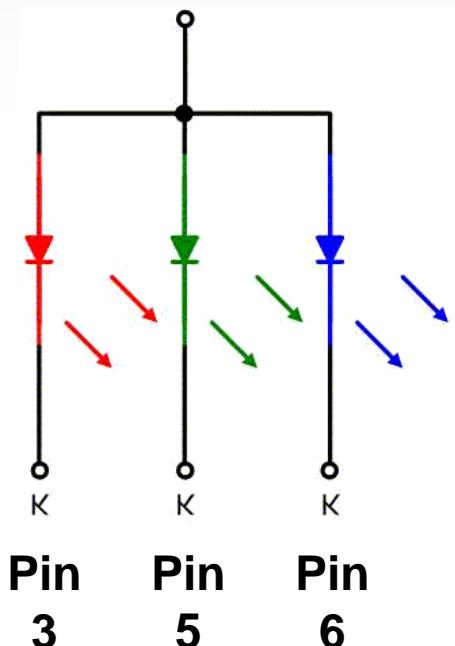


Blinking in 3 colors Tri-color LED



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

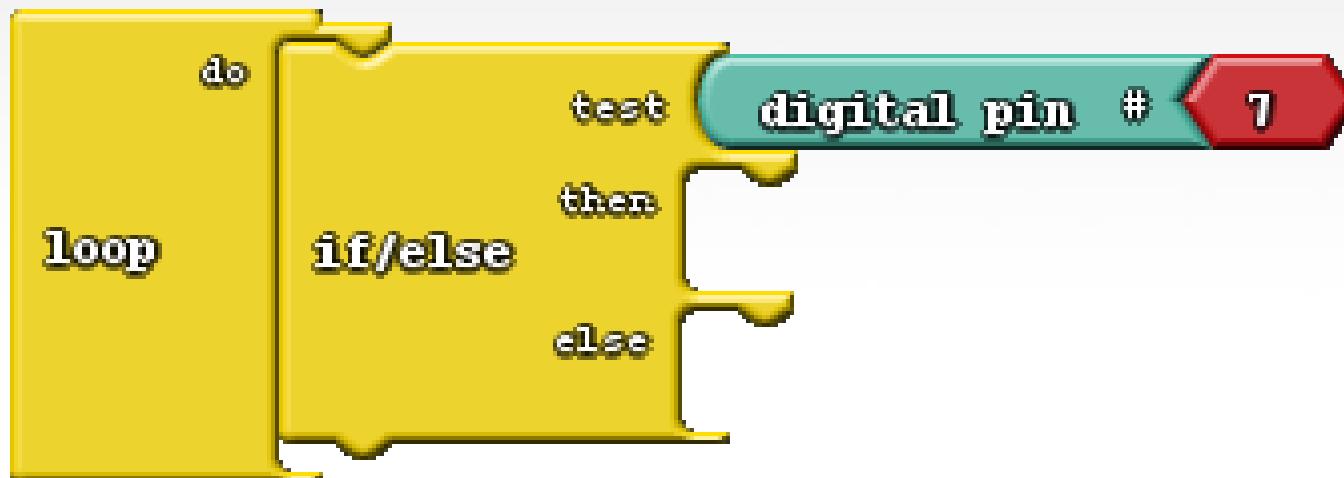
Example



or



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

Note	Frequency (Hz)
C ₄	261
C [#] ₄ /D ^b ₄	277
D ₄	293
D [#] ₄ /E ^b ₄	311
E ₄	329
F ₄	349
F [#] ₄ /G ^b ₄	369
G ₄	392
G [#] ₄ /A ^b ₄	415
A ₄	440
A [#] ₄ /B ^b ₄	466
B ₄	493

Note	Frequency (Hz)
C ₅	523
C [#] ₅ /D ^b ₅	554
D ₅	587
D [#] ₅ /E ^b ₅	622
E ₅	659
F ₅	698
F [#] ₅ /G ^b ₅	739
G ₅	783
G [#] ₅ /A ^b ₅	830
A ₅	880
A [#] ₅ /B ^b ₅	932
B ₅	987



World's cheapest dubstep instrument

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

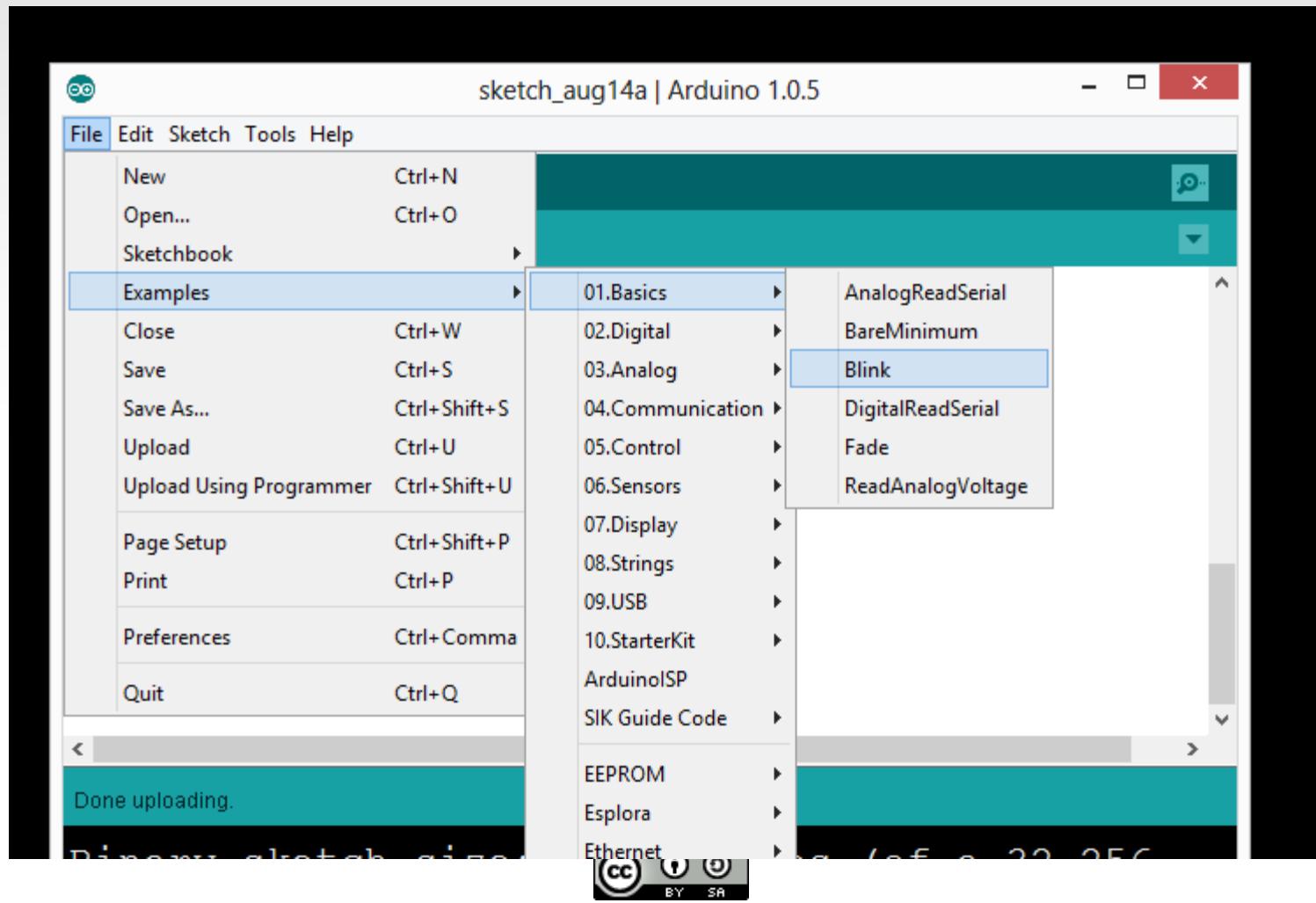
Play with the frequency – scale it by applying multiplier factors or offsets.



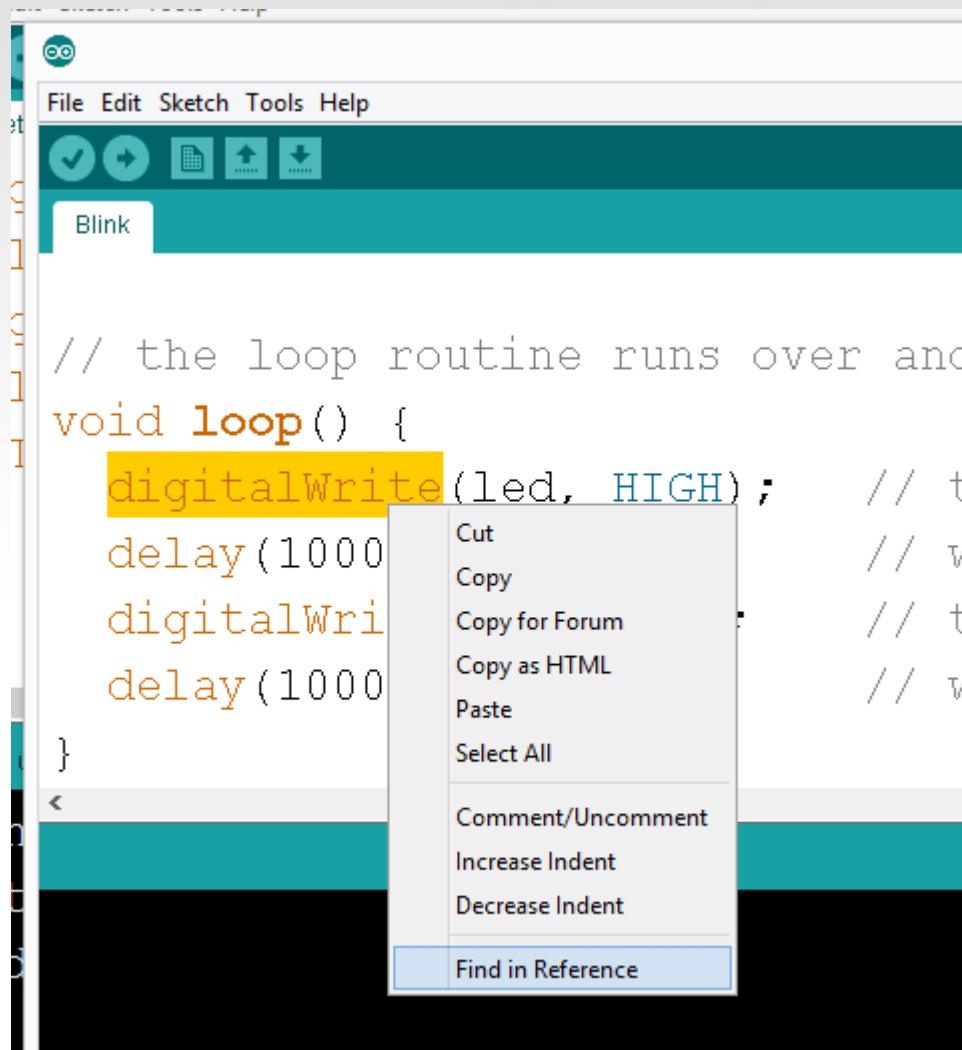
OK? What's going on under the hood?



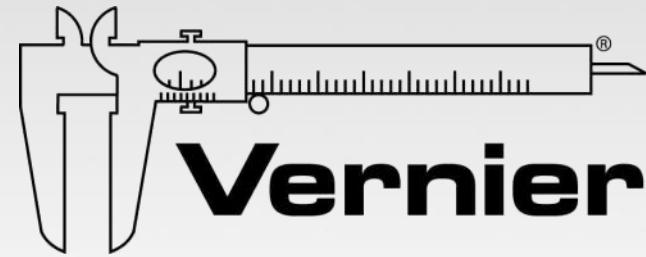
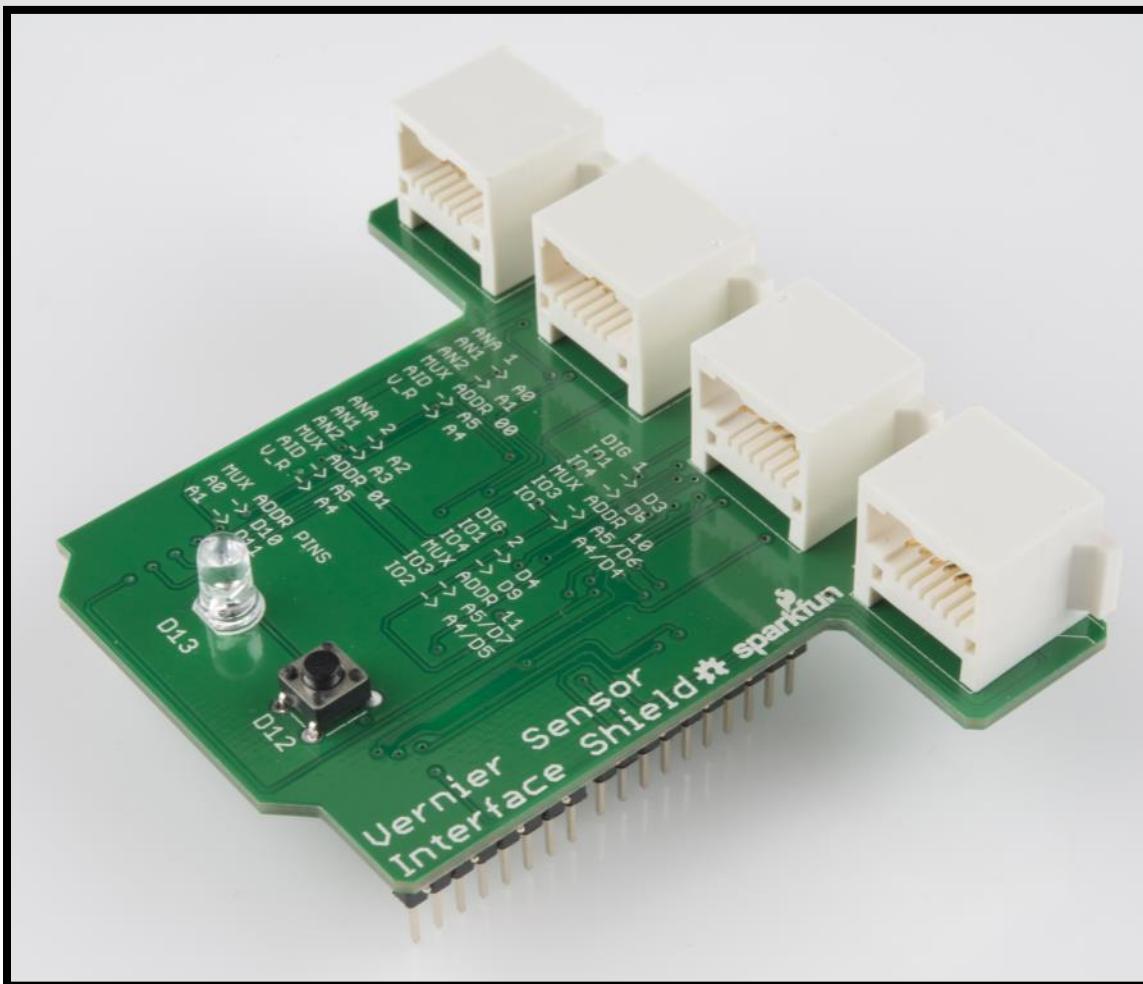
Arduino Code, Examples, References, etc...



Find in Reference



A partnership...



Resources

Programming Cheat Sheet →
<http://bit.ly/ArduinoCheatSheet>

Download Arduino / Reference →
<http://arduino.cc/en/>



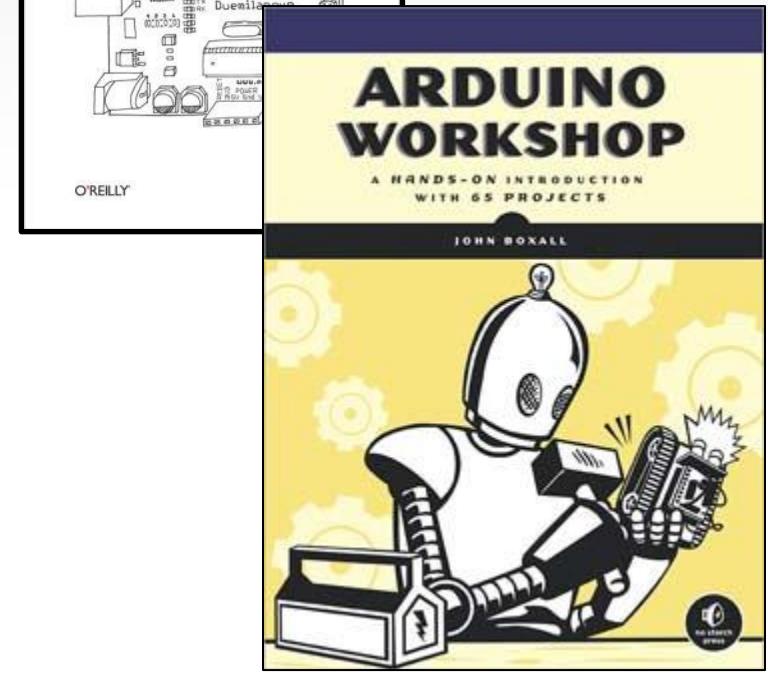
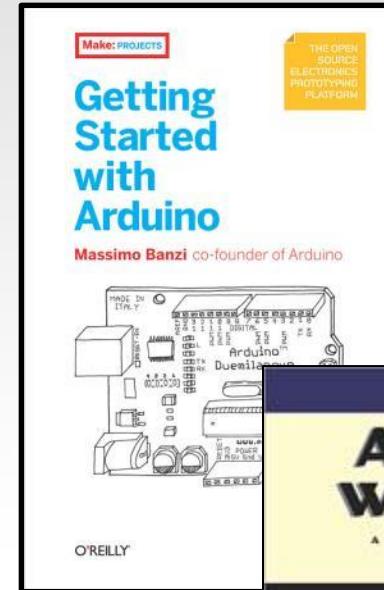
Additional Resources

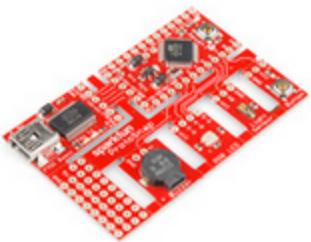
<http://www.sparkfun.com>

<http://learn.sparkfun.com>

<http://www.arduino.cc>

<http://www.bildr.org>





images are CC BY-NC-SA 3.0



ProtoSnap - Pro Mini

DEV-10889 RoHS✓

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](#)

\$29.95

[Add to Cart](#)

1

quantity



11 in stock

\$29.95

price

\$26.96

10+ units

\$23.96

100+ units

[Add to Wish List](#)

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...

<http://www.plasma2002.com/blenderdefender/>

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>



www.sparkfun.com
6175 Longbow Drive, Suite 200
Boulder, Colorado 80301