THE ENGINEERING GAMES
THE UNIVERSITY OF ILLINOIS

G.A.M.E.S. Camp 2016

Computer Science

Welcome:

Welcome to GAMES camp! We are excited that you are here and are looking forward to a fun week! During this camp, you will...

- Learn about circuits
- Solder
- Code with Scratch
- Program in Arduino
- Make a wearable electronics project
- Develop your own app

What is Computer Science:

Computer Science *IS:*

- Practiced by mathematicians, scientists, and engineers
- A discipline that spans theory and practice
- Fun and creative
- The study of problem solving
- Diverse, and often requires both computer science expertise AND knowledge of a particular application domain
- Focused on processes for handling and manipulating information

Career Resources

http://www.jason.org/live/stem-career-qa-beth-mccabe-google-technologist
Google Technologist

http://www.computerscienceonline.org/careers/
Computer Science Online

http://www.businessnewsdaily.com/7863-women-stem-career-resources.html
Business News Daily

http://www.onlinedegrees.com/computer-science/
Choosing the Right Degree

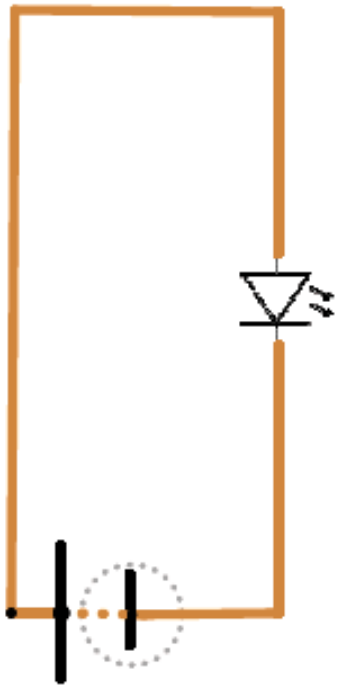
http://www.jason.org/live/stem-career-qa-james-bryant-cybersecurity-consultant
Cyber Security Consultant

Inspiration:

Made with Code https://www.madewithcode.com/

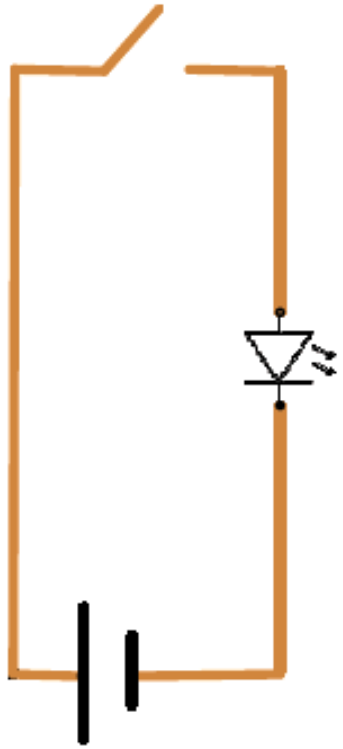
SIMPLE CIRCUIT

Trace the path of the circuit with copper tape, leaving breaks in the tape for the battery and LED. Both sides of the battery need to be a part of the conducting path. Place the battery on the copper tape right before the break. Extend the copper tape from the other side of the break so that it touches the top of the battery.
Tape the LED and battery into the circuit.



SIMPLE CIRCUIT
with a switch

Leave a small gap in the circuit for the switch. One side of the copper tape near the gap should be long enough to touch the other side plus an extra inch. Fold the long piece of tape under itself, covering up the adhesive for a half inch.

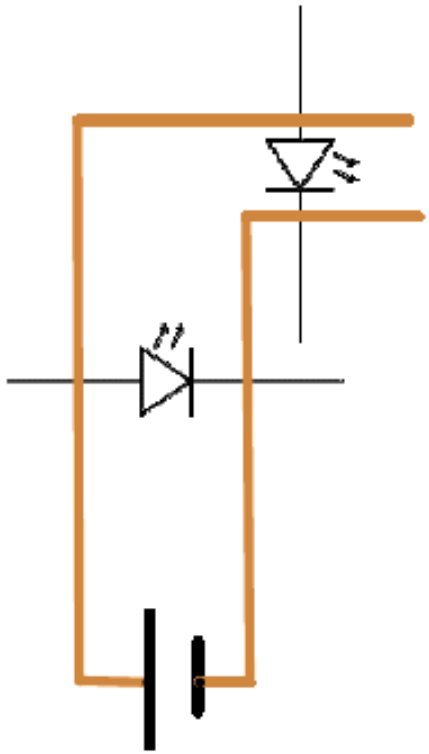


Materials -

copper tape
clear tape
LEDs
3V coin battery
scissors

PARALLEL CIRCUIT

Use a parallel circuitto power two or more LEDs, .
How can you add more LEDs?

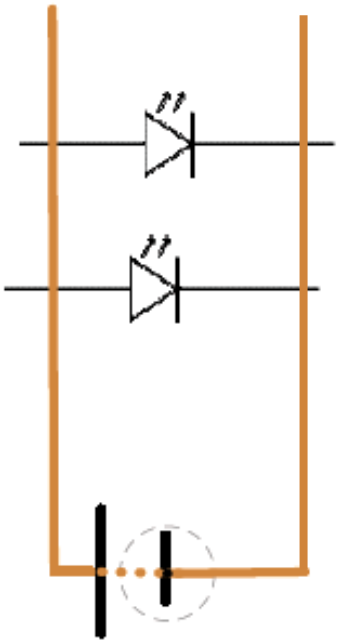


TCIPG

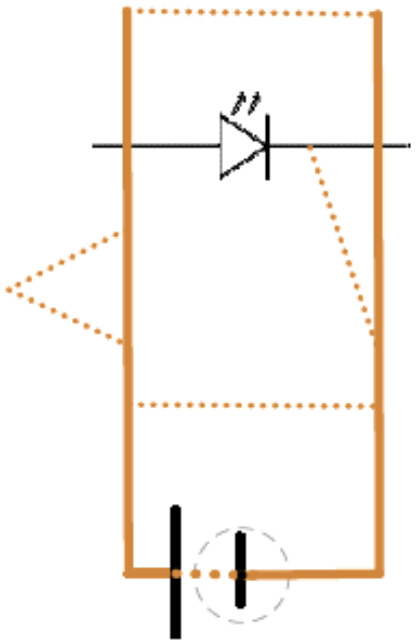
Short Circuits

Materials -
copper tape
clear tape
LEDs
3V coin battery
scissors

CIRCUITS
Trace the solid line paths of the circuits with copper tape, and tape the battery into the circuit.



Challenge I
1. Place two red LEDs into the circuit as shown.
 Do both light?
 Is this a parallel circuit?
2. Add a third and fourth LED.
3. Try LEDs of different colors.
4. How many LEDs can you light in this circuit?
5. Record what you observe in your handbook.



Challenge II
1. Place an LED into the circuit as shown. Does it light?
2. Cut a 3 inch strip of copper tape and leave the backing on. Place the tape into the circuit with the shiny side down in the various places shown by the dotted lines.
3. Add more LEDs.
4. Record what you observe in your handbook.

Learn more about LEDs.
University of Illinois Celebrates LED 50 years
http://www.led50years.illinois.edu/
LED's: The Basics
https://learn.sparkfun.com/tutorials/light-emitting-diodes-leds

Light emitting diodes of LEDs create light when electrons are pushed through two different semiconductor materials. The two materials are layered together so that electrons can flow in only one direction. The moving electrons release photons that we see as colored light. The color of the LED light depends on the type of semiconductor.

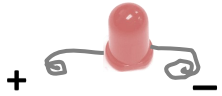
LEDs need only small amounts of current and voltage to create light, but some colors require more power than others. LEDs don't create heat so they are more efficient light sources and they last longer than traditional light bulbs.

LED lighting is becoming more common. They are used to light supermarket displays and freezer sections, streetlights and traffic lights, automobile taillights, and have recently been installed to light the giant signs in Times Square in New York City.

Circuit Puzzles

Predict whether each circuit will light all the LEDs . Then build the circuits. If the circuit doesn't work make changes! Add or remove copper tape. Add more or different LEDs.

LEDs have a positive and negative lead (wire). Typically the longer lead is positive and the shorter lead is negative, check each LED before placing it in your circuit. In this activity the picture shows the tightly wound lead as positive, and the loosely wound lead as negative.

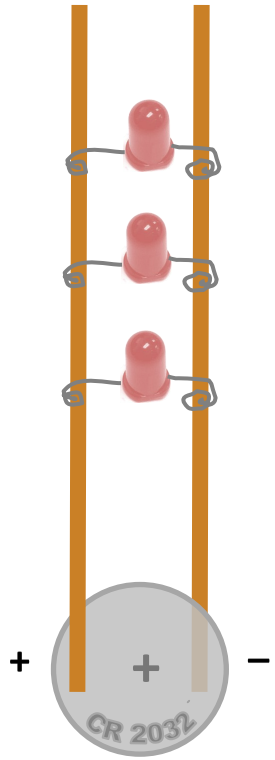


1.) Will all three LEDs light?

Predict:

ALL? TWO? ONE? NONE?

What actually happens?

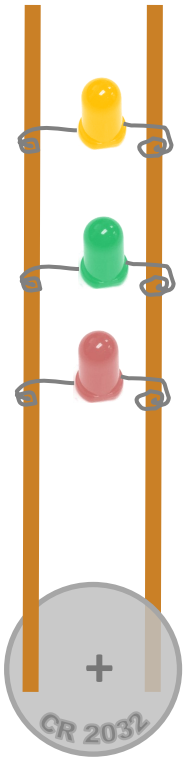


2.) Will all three LEDs light?

Predict:

ALL? TWO? ONE? NONE?

What actually happens?

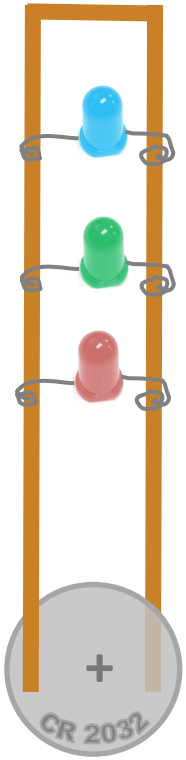


3.) Will all three LEDs light?

Predict:

ALL? TWO? ONE? NONE?

What actually happens?

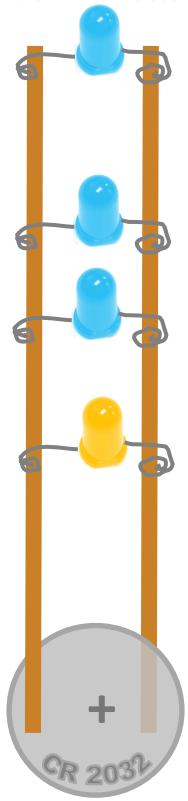


4.) Will all four LEDs light?

Predict:

ALL? THREE? TWO? ONE? NONE?

What actually happens?

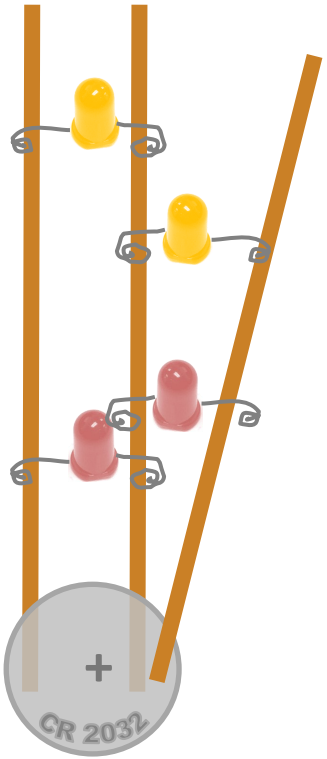


Will all four LEDs light?

Predict:

ALL? THREE? TWO? ONE? NONE?

What actually happens? Add two more LEDs so all six light.



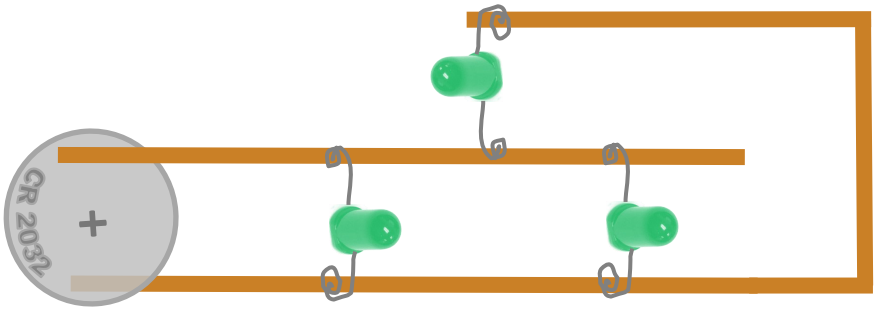
6.) Will all three LEDs light?

Predict:

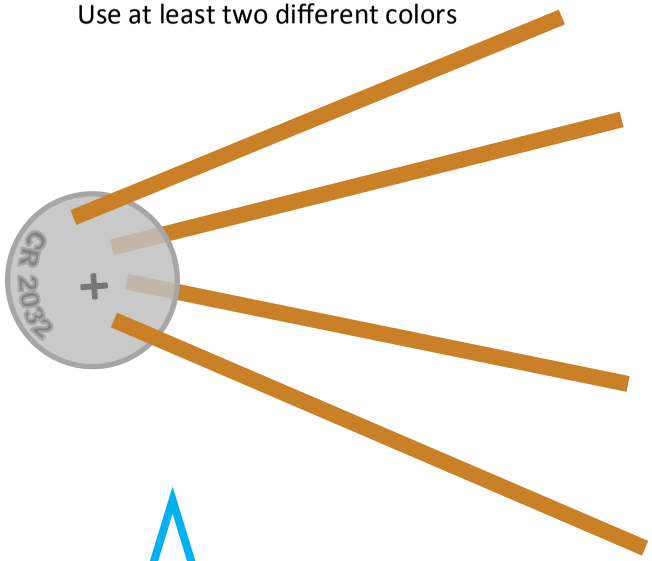
ALL? TWO? ONE? NONE?

What actually happens?

Add three more LEDs so all six light.



7.) Place four LEDs so that they all light. Use at least two different colors



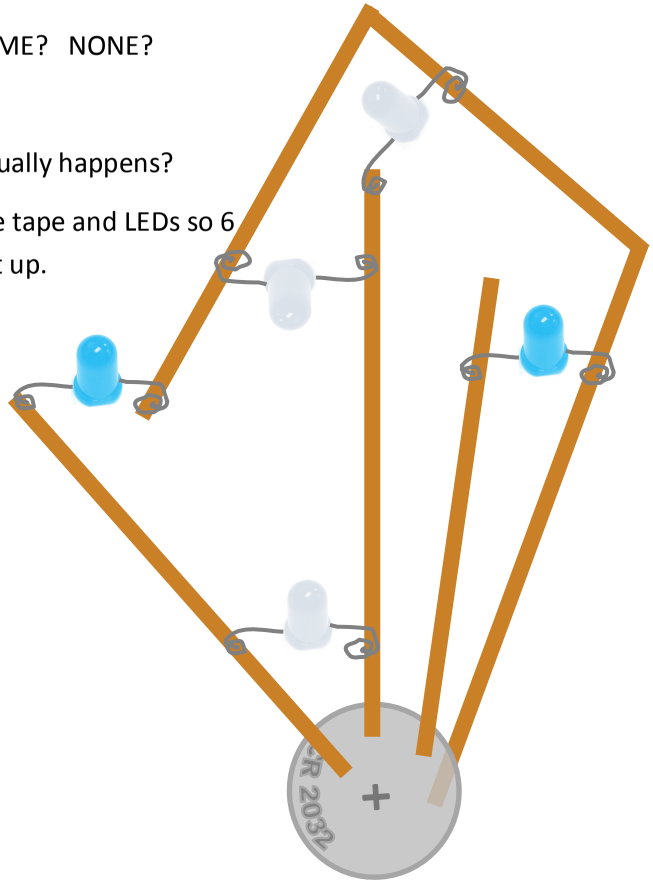
8.) Will all five LEDs light?

Predict:

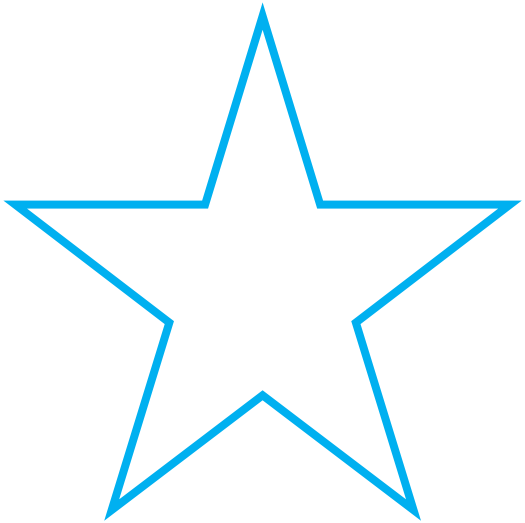
ALL? SOME? NONE?

What actually happens?

Add more tape and LEDs so 6 LEDs light up.



9.) Design a circuit that will light the five corners of the star and use LEDs of at least two different colors.



Other Resources for Copper Tape Projects

NexMap: http://www.nexmap.org/

Education initiative to provide toold to stimulate creative thinking and learning with paper circuitry.

21st Century Notebooking: http://www.nexmap.org/21c-notebooking-io/

Hack a notebook! Use copper tape, LEDs and batteries to light up a notebook. Program your pages using microcontrollers.

Jie Qi: http://technolojie.com/

Paper electronics, and DIY technology design. Add amazing circuit stickers to your projects.

Chibitronics: http://chibitronics.com/

Tutorials and resources to get started and expand on paper circuit projects.

NOTES

Intro to Soldering

MATERIALS:

Copper Tape

LEDs

1 coin cell battery

Soldering Iron

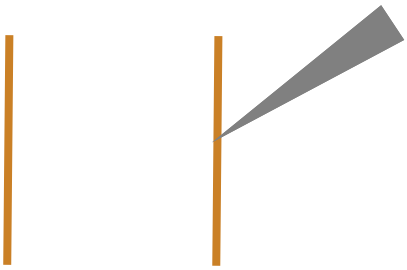
Solder

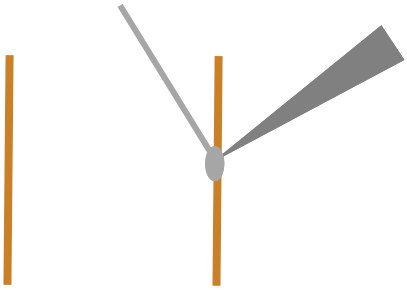
SET UP:

Tape Circuit 1 on the next page. Stretch the leads of the LEDs to touch both strips of copper tape, but **do not** tape them down. (Remember: the long leg of the LED is positive).

STEPS FOR SOLDERING ON COPPER TAPE

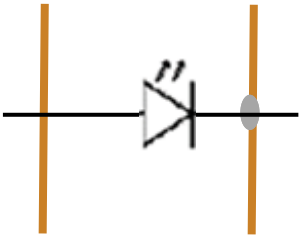
1.) HEAT IT UP — touch the hot soldering iron to the copper tape where an LED lead will be connected for 2-3 seconds.

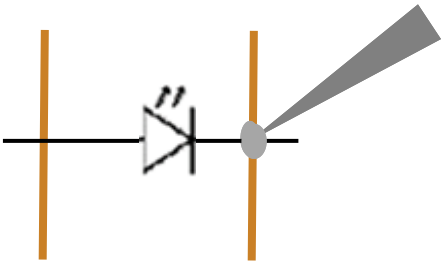




2.) ADD A DAB — touch the end of a piece of solder to the hot copper tape. When the solder starts to melt on the copper tape, remove the soldering iron. Let the solder cool and harden on the copper tape.

3.) POSITION YOUR LED — lay the LED lead across the dab of solder.

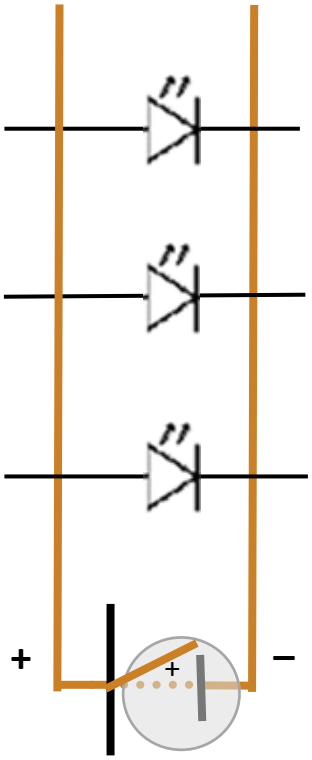




4.) HEAT AND FLOW — apply the hot soldering iron to the LED leg *and* the solder it is touching, so that both heat up. Add some more solder to ensure that the LED is securely connected to the copper tape.

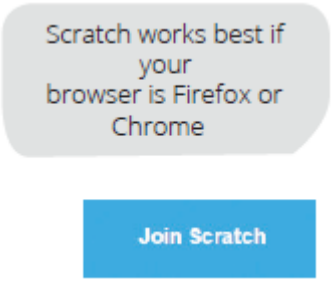
5.) REPEAT — solder the other leg of the LED to the copper tape and repeat for remaining LEDs.

Practice Circuit



On a piece of cardboard, recreate this circuit and solder 3 LEDs to this circuit

Creating with Scratch



Scratch is a visual programming language that lets you create and share projects with others. It encourages you to problem solve, think logically, test and evaluate outcomes, and collaborate with others.

To try out Scratch, see examples of projects, and join Scratch, visit:

https://scratch.mit.edu/

Create your own account on the Scratch website. Click Join Scratch and follow the instructions. Then click **Try it Out** to get started.

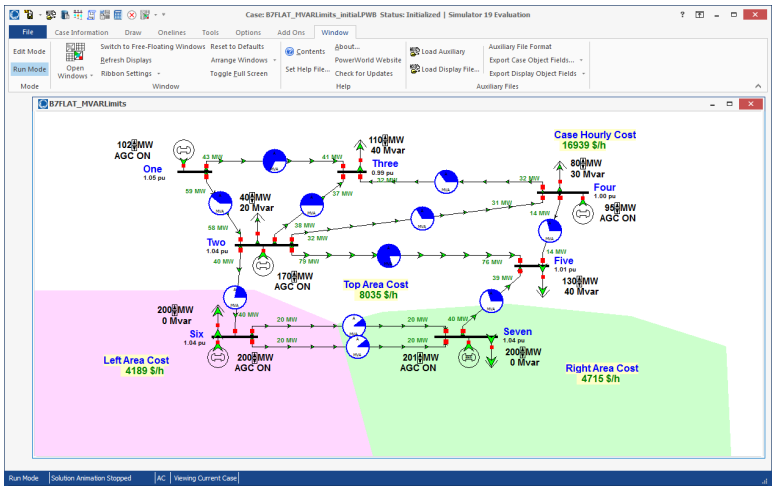
Discovering Computer Science and Programming through Scratch will get

This guide is an introduction to the basic elements of programming within the Scratch environment. It covers the most fundamental principles of programming in any programming language: sequence, iteration, conditionals, variables, and modularization. After completing the activities in this guide you should be able to write simple programs for a variety of purposes. Learning to program is much like learning to play a musical instrument. Only with lots of practice can you improve your skills and create beautiful things. This guide should give you some fundamentals on which to build, but you will want to spend lots of time on your own practicing, experimenting, exploring, and creating. Luckily, doing this is easy and fun with Scratch.

Introducing PowerWorld

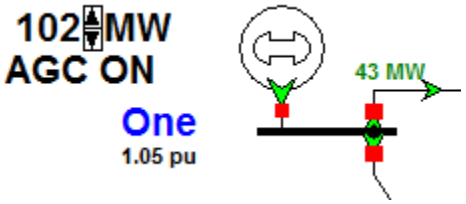
B7FLAT_MVARLimits_initial.PWB

- Open the PowerWorld Simulation.
- Go to the File Menu and open the B7FLAT simulation case .
- You should see the opening screen.
- Click on **Tools** in the ribbon tab. Then click on the **green arrow** to play the simulation. You should see the generator symbols turning and the green power flow arrows moving.

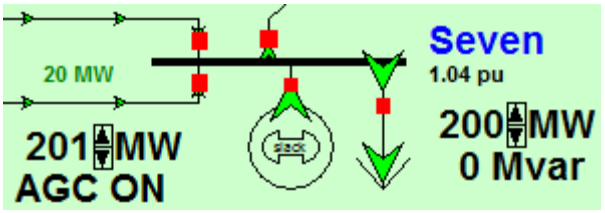


RESET THE SYSTEM AT ANY TIME BY CHOOSING THE B7FLAT FILE FROM THE FILE MENU.

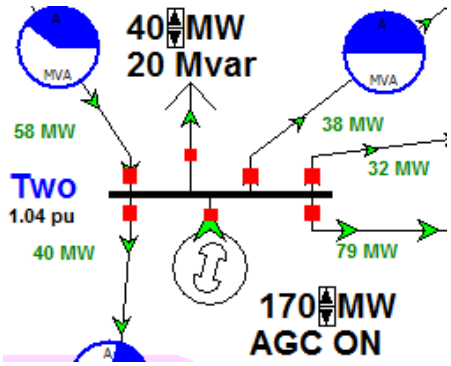
1. Each of the five generators has automated generation control on and has adjustable output. At bus One the generator is producing 102 MW of power.



2. How much power is being produced by the whole system? _____
3. The arrows that look like they're going nowhere are showing power going to the loads (demand). At bus Seven the demand is 200 MW and the generator is producing 201 MW.



4. How much power is being used by the whole system? _____
5. What happens to the other generators when you disconnect the generator near Bus Six by clicking on the red square just above it? Does the demand change? _____
6. A closer look at Bus Two shows power flowing in and out. How much power is leaving Bus Two? _____ How much is entering? _____



7. Blue circles show transmission line power flow as a fraction of capacity. Find the line between Bus Two and Bus Three. Estimate its capacity.

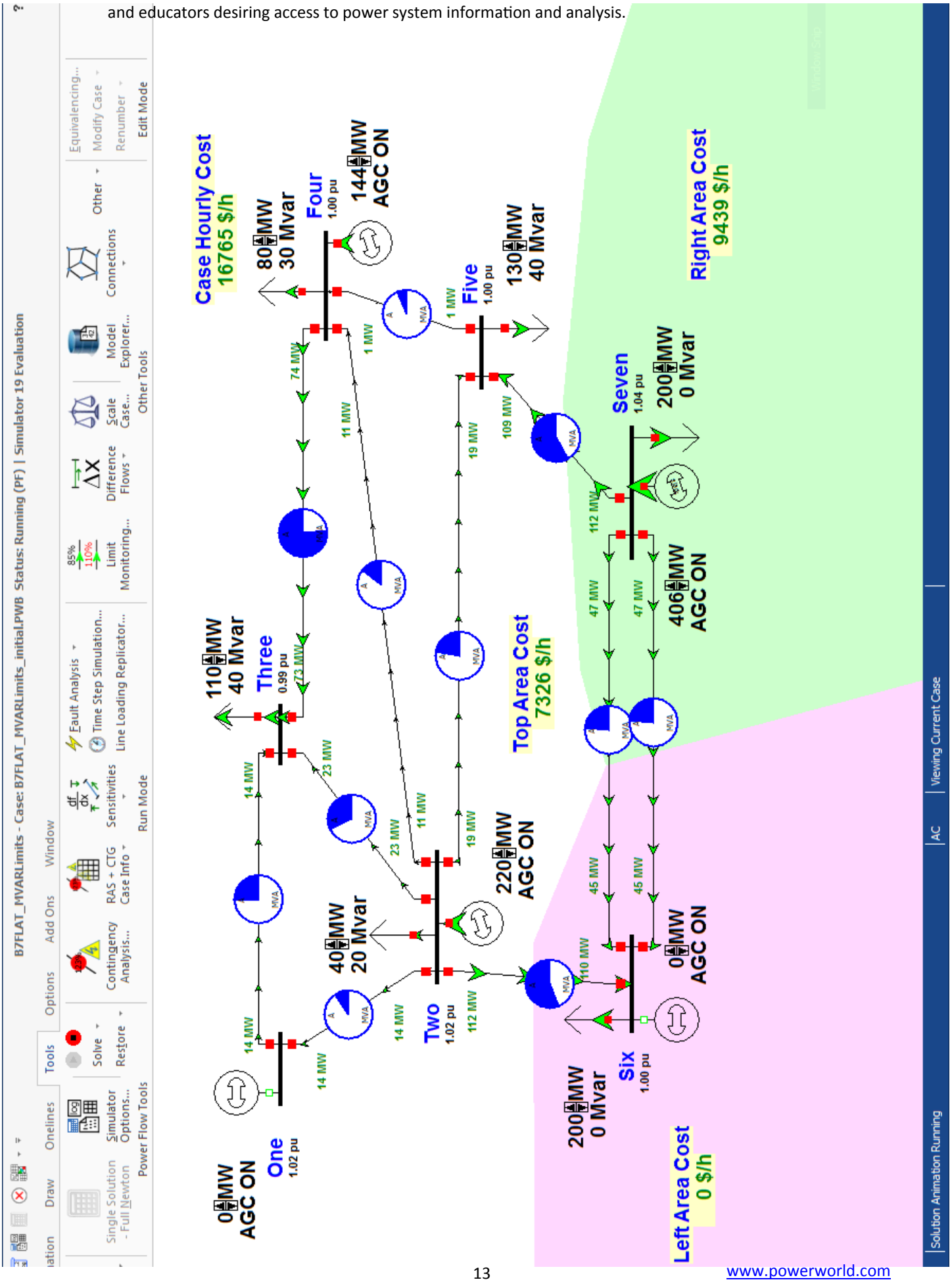
_____

8. What happens when you click on a red square?

_____

9. What happens if you adjust the output or disconnect the generator near bus Two? Near Bus Seven _____
10. When the system is operating well the per unit (pu) voltage at each bus should be close to 1.00.

The PowerWorld Simulator is a tool used by transmission planners, power marketers, system operators and trainers, and educators desiring access to power system information and analysis.



CIPHERS

Ciphers

A **cipher** is a set of rules for converting between **plaintext** and **ciphertext**. These rules often use a secret **key**, and are used to encrypt and decrypt messages. It's a way of protecting secrets!

Shift ciphers are one common type of cipher. These simple ciphers substitute one letter for another letter that is some fixed number of positions further down the alphabet. The cipher key determines the number of letters for the shift. You can use a cipher wheel to encrypt and decrypt your secret message.

Today computers use much, much more complicated rules to encrypt passwords and other private information.

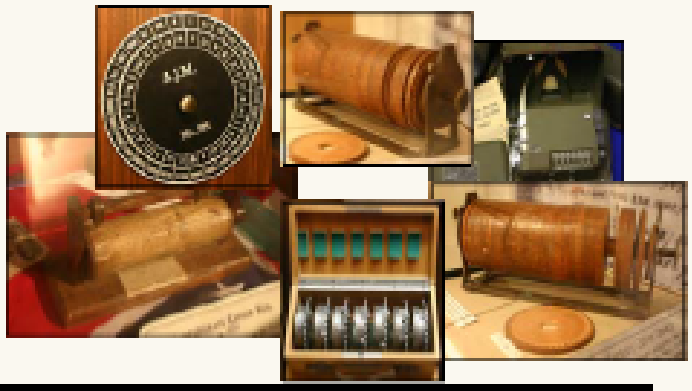
Encryption

Encryption consists of a few parts:

-Plaintext: The original text.

-Ciphertext: The encoded version of plaintext.

-Cipher: The algorithms or rules used to encrypt and decrypt ciphertext.



Examples

1. **Key=3**

   Ciphertext: *E-B-I-I-I*

   Plaintext: *hello*

2. **Key=6**

   Ciphertext: *I-G-K-Y-G-X*

   Plaintext: *caesar*

Ciphers in History

The Caesar cipher was used by Julius Caesar two thousand years ago. This is one of the first known uses of a shift cipher.

Ciphers were also used during the American Civil War, where Union and Confederate generals and civilians used codes and ciphers to transmit secret messages. Both sides attempted to break each other's code and cipher systems with varying degrees of success!

CRYPTOGRAPHY & INFORMATION THEORY

Journey into cryptography

How have humans protected their secret messages through history? What has changed today?

ALL CONTENT IN "JOURNEY INTO CRYPTOGRAPHY"

Ancient cryptography

Explore how we have hidden secret messages through history.

- What is cryptography?
- The Caesar cipher
- Caesar Cipher Exploration
- Frequency Fingerprint Exploration
- Polyalphabetic cipher
- Polyalphabetic Exploration
- The one-time pad
- Perfect Secrecy Exploration
- Frequency stability

Modern cryptography

A new problem emerges in the 20th century. What happens if Alice and Bob can never meet to share a key in the first place?

- The fundamental theorem of arithmetic
- Public key cryptography: What is it?
- The discrete logarithm problem
- Diffie-hellman key exchange
- RSA encryption: Step 1
- RSA encryption: Step 2
- RSA encryption: Step 3
- Time Complexity (Exploration)
- Euler's totient function
- Euler Totient Exploration
- RSA encryption: Step 4
- What should we learn next?

Investigate Computer Science at the Khan Academy. Visit *Cryptography & Information Theory* to learn select topics from computer science - cryptography (how we protect secret information) and information theory (how we encode and compress information). Start with Ancient Cryptography www.khanacademy.org/computing/computer-science/cryptography

GUARDIANS OF THE GRID

CONGRATULATIONS. You have accepted the invitation to join TEAM BLACKOUT, the world's preeminent hacker brigade targeting power systems around the world. Our goal is to plunge the modern world into darkness … for no reason other than we can.

Today we will be targeting the buzzing, bustling metropolis of Springpatch. Springpatch consists of seven large neighborhoods. Your goal is to disrupt the electrical supply to the entire city. If successful, you will cause a blackout that will cripple Springpatch's ability to conduct modern life. Yowza!

We have already simulated the system and have determined that a sequence of eight control actions will do the trick. To keep our plans secret, we will be sending encrypted instructions to you for each action. All you have to do is decrypt each message and then perform the control actions. It's just that simple.

For each successful operation that you perform, the members of your team will be awarded prizes. You will be competing with your fellow team blackout new recruits. We hope you will be a member of the first team to black out Springpatch

Good luck. Go Team Blackout!

Hiding data

There are two basic techniques for hiding data:

1. encryption (two-way: what is encrypted can be decrypted)
2. hashing (one-way: you can hash a value, but you can't unhash it)

Encryption

When you encrypt data, you take a plaintext message and a key, and you use the key along with an algorithm to make the plaintext unintelligible. In other words, you produce cipher text.

When you decrypt data, you go in reverse: you take a ciphertext message and a key, and you use an algorithm to recover the original plaintext.

There are several ways to do this. Here are just a few.

1. Caesar
2. Monoalphabetic Substitution Cipher
3. Affine
4. Keyword
5. Playfair
6. Viganere

Learn More -

- Use the **Cypher Handbook** that begins on page 26.
- Visit these websites:

  NovaLabs: Cybersecurity Lab

  http://www.pbs.org/wgbh/nova/labs/lab/cyber/

  Khan Academy

  https://www.khanacademy.org/computing/computer-science/cryptography

  CrpytoClub

  http://cryptoclub.org/

Springpatch Blackout

Now that you're a master hacker, complete the following challenges with your teammates to send Springpatch into a Blackout!

Challenge 1

WKHUH LV D OLQH WKDW

VWUHWFKHV EHWZHHQ

EXV WZR DQG EXV WKUHH

RSHQ LW QRZ L SOHD

Clue: This was encrypted using the Caesar Cipher with a key that is not divisible by 2.

Question: Which two lines are now over 80% loaded?

Challenge 2

W CHGHMWOIM WO W XPN

SKINH GWFH MKUFHN

SEOK ZIIM ZHHGHMCEVH EO

GIS E EFJDIMH

Clue: This is a Keyword Cipher. The keyword is a 6-letter noun for someone who breaks into computers. The key letter is the letter that usually comes after I unless C is involved.

Question: How many lines are now loaded beyond their limits?

Challenge 3

K F J R L V S K E V M V U K A R M V F O H

K D R O F K I N H K A E R R E F D H R

K A R P K V F K V K F M Y V T R E V U

V S R A N S O E R O F S O U D U K Z

Clue: This was encrypted with the Affine Cipher. The line that describes it passes through pointe (0, 5) and (1, 8)

Question: What is the voltage (measured in pu) at Bus 3?

Challenge 4

19 07 00 19 18 14 20 19 07 01 17 00 13 02 07 19 07 00 19 02 17 14 18 18 04 18

19 07 04 19 14 15 00 17 04 00 13 04 04 03 18 19 14 14 15 04 13

20 15 13 14 22 08 18 22 04 00 17 19 14 24 00

18 14 03 14 08 19 13 14 22 22 14 13 19 24 00

Clue: Numbers have been used to replace letters in the encryption above.

Question: How many buses in the top area are now below 0.85pu voltage?

Challenge 5

K F S V P C L D J N K U G V S W P Y T

A Y I I I F Y U R D V Y Z F Y E T K H I K

U S A Z V C Q X S D B W C G C Q

R F I V O O Z N O E R K M Y

Clue: This was encrypted by the Vigenere cipher. The keyword is the 8-letter name of the kind of event that happened in New York in 1967 and in the Northeast United States on August 14, 2003.

Question: What is the percentage loading on the line connecting Bus 2 and Bus 6?

Challenge 6

s b h h m n m x v c h s m m x v m c n m x

f d h k g c u h g p c q b x h i c d v m u

d b q x g p p d v c d k g c u h g d v c d

h g g x d v m w k k s m u f h e c w m u

Clue: This was encrypted using the mono-alphabetic substitution cipher. Perform an SQL injection attack against the website below to discover the key:

http://cs.lewisu.edu/~klumpra/gotg/challenge6.html

Question: How much power (MW) is being generated by the generator at Bus 1?

Challenge 7

I D R T R X C L X A K J M Q H Z Y L L A R R

W X M U F S B B Y U N S M L R A N W H

P N Y F I D N X O J L Y J X C U X Q P U

Q G C P L S X Q J Y S H A

Clue: Go to the website below to guess the password:
http://cs.lewisu.edu/~klumpra/gotg/challenge7.html

The password was created using a simple hashing algorithm that has 20 possible values. Once you log in with the correct guessed password, you'll learn the key and technique used for this one.

Question: What is the voltage at Bus 4?

Challenge 8

frdqtsekcfbssltobcze

gkztfrfrdqcstueaaibt

uadawlkcvfkcozaecsdm

scaeslgbmtukutetrt

Clue: Use the Playfair Cipher with key “ALMOST.”

Question: At what MW load at Bus 5 does the system experience a blackout?

Cypher Handbook

The Caesar Cipher

With the Caesar cipher, every character in the plain text message is advanced by the same amount in the alphabet to produce the cipher text. The key is an integer, and it indicates how much to advance each letter.

Example: HAPPY becomes KDSSB when the key is 3. Notice how we had to wrap back around to the beginning of the alphabet.

To decrypt, we subtract the key from each character in the ciphertext to figure out the plaintext was.

Example: KDSSB becomes HAPPY again when the key is 3 just by stepping back 3 places.

Monoalphabetic substitution cipher

Rather than step ahead by the same amount for each character, we instead map each character in the plaintext alphabet to a new character in the ciphertext alphabet.

Here's an example.

For the alphabet: a b c d e f g h I j k l m n o p q r s t u v w x y z
Let this be the key: l f b o z s m x g I h q y t r e p k c d w n j v a u

Encrypt the word: encyclopedia

Result: ztbabqrezogl

Keyword Cipher

The Keyword Cipher identifies both a keyword and a key letter. The keyword is written underneath the plaintext alphabet starting at the key letter. The resulting alignment of characters identifies the substation patterns.

For example, let the keyword be GUITAR and the key letter be C.

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Y Z G U I T A R B C D E F H J K L M N O P Q S V W X

The word "HAPPY" then becomes "RYKKW".

Playfair – another substitution cipher

The best-known multiple-letter encryption cipher is the Playfair, which treats digrams in the plaintext as single units and translates these units into ciphertext digrams. The Playfair algorithm is based on the use of a 5x5 matrix of letters constructed using a keyword. The rules for filling in this 5x5 matrix are: L to R, top to bottom, first with keyword after duplicate letters have been removed, and then with the remain letters, with I/J used as a single letter.

- Using the keyword "MONARCHY"

M	O	N	A	R
C	H	Y	B	D
E	F	G	I/J	K
L	P	Q	S	T
U	V	W	X	Z

1. a 5X5 matrix of letters based on a keyword
2. fill in letters of keyword (sans duplicates)
3. fill the rest of matrix with the rest of the alphabet.

Plaintext is encrypted two letters at a time, according to the rules as shown. Note how you wrap from right side back to left, or from bottom back to top.

if a pair is a repeated letter, insert a filler like 'X', eg. "balloon" encodes as "ba lx lo on"

if both letters fall in the same row, replace each with letter to right (wrapping back to start from end), eg. "ar" encrypts as "RM"

if both letters fall in the same column, replace each with the letter below it (again wrapping to top from bottom), eg. "mu" encrypts to "CM"

otherwise each letter is replaced by the one in its row in the column of the other letter of the pair, eg. "hs" encrypts to "BP", and "ea" to "IM" or "JM" (as desired)

Decrypting follows mostly the same rules, except that you have to choose the character immediately above when the two letters are in the same column and immediately to the left when they are in the same row.

Affine Cipher

With the Affine Cipher, we plug the number of each character into a linear equation y = (mx + b) mod 26, and we figure out what the resulting character is. This creates the substation alphabet.

For example, with m = 3 and b = 3, we get

ABCDEFGHIJKLMNOPQRSTUVWXYZ

DGJMPSVYBEHKNQTWZCFILORUXA

So, the word “HAPPY” would become “YDWWX”

Viganere – another substitution cipher

This one uses the Viganere Tableau:

Key	Plaintext																								
	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
a	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
b	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
c	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
d	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
e	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
f	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
g	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
h	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
i	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
j	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
k	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
l	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
m	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
n	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
o	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
p	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
r	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
s	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
t	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
u	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
v	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
w	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
x	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Example: Encrypt “Happy” with key “Dog”

key (i.e. the row) DOGDO

plain (i.e. the col) HAPPY

ciphertext (look at cell) KOVSM

Hashing

Remember that there are two ways to hide something:

1. encrypt it, which is reversible

hash it, which is not

Encryption is applied, for example, to personal information in database that the owner of the database will want to get back. For example, Amazon has to keep your charge card number on file in an encrypted way, but it also must be able to recover the charge card number by decrypting it so that it can charge your bank.

Unlike encryption, a hash is one-way. It is a mathematical mapping between text and a value that can't be undone. Hashes are typically used to store passwords in databases. There are many different kinds of hashes, including md5 and sha1.

Hash functions are many-to-one. Many strings may yield the same hash function, but finding the original text given the hash is virtually impossible.

Here's an example of a hash function: add up the ascii codes, and then mod by 100. That will map anything you apply the hash function to to one of 100 values (0 through 99).

Here's some python code to do that:

```
sum = 0
phrase = input("Enter phrase: ")
for ch in phrase:
    sum = sum + ord(ch)
hash = sum % 100
print(hash)
```

This hash function will produce 100 different values. If you used this to protect a website, a person would have a pretty easy time logging in even without knowing the password, because only 100 different values are possible. So after a person tries any 100 different passwords in succession s/he is in.

How could the code be used to make better protection?

Modern hashing in a database

A database is an organized way to store data. The data are stored in tables that are connected to each other. A table consists of rows and columns. Each table stores data about a particular kind of thing.

Databases are hosted on database servers. One of the most popular kinds of database server is called MySQL. Many websites use MySQL to host data behind the scenes.

The problem, of course, is, if anybody gets a hold of the database that powers a website, all that personal and sensitive information behind that website would be revealed.

For example, all that data could be revealed to a hacker who employs an **sql injection attack**. An sql injection attack is one in which the attacker plugs in some maliciously crafted sql command to retrieve more data than what they are supposed to get.

For example, consider a website with a simple login form. If the site is poorly coded, then, the user could type the code below into the password box, and be able to get the information for all users in the table. NOT GOOD!

```
‘  o r  1 = 1 ;  - -
```

(Apostrophe, space, o, r, space, 1, =, 1 ;, space, -, -, space)

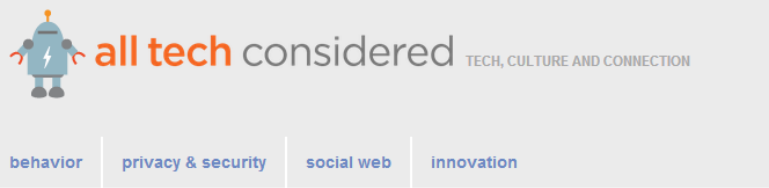
This is an SQL Injection attack, and it is the most popular way to hack into a database and steal personal information.

Digital Forensics, Privacy, and Security

What information are you sharing?

What does your online activity reveal about you?

Who could be listening to your phone activity?



privacy & security

Project Eavesdrop: An Experiment At Monitoring My Home Office

by STEVE HENN

June 10, 2014 3:47 AM ET

Your Digital Footprint

http://www.internetsociety.org/your-digital-footprint

Illinois Cyber Security Scholars Program

Explore the fascinating field of cybersecurity through the Illinois Cyber Security Scholars Program, open to undergraduate and graduate students in computer science and computer engineering as well as to law students.

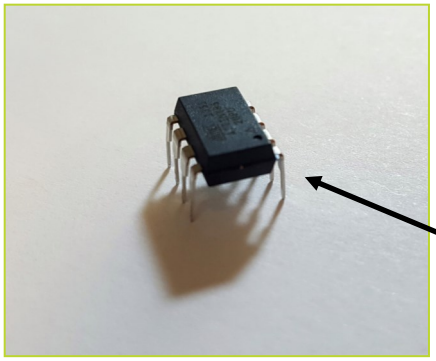
Through the program, funded by the National Science Foundation, students can explore cutting edge profession that is expected to increase by 53 percent over the next 5 years. Graduates will be prepared for careers that may take them to research labs or government agencies where they will defend against the growing threat of cyber crime.

Students will study under some of the nation's top academic experts in cyber security and cyber law, learning how to protect the nation's cyber infrastructure by designing more secure systems and methodologies, as well as better policy. Graduates of the program will receive a certificate in cyber security.

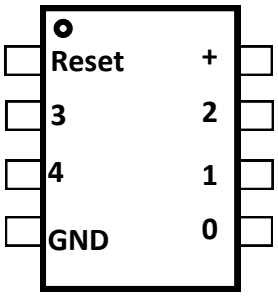
https://publish.illinois.edu/cybersecurityscholars/

Programming in Arduino

Arduino is a fun and powerful tool you can use to program a multitude of exciting projects, ranging from exciting light projects to personalizing clothing and other accessories with beautiful blinking lights! Not only is Arduino powerful, it is easy to learn with a little practice and curiosity!

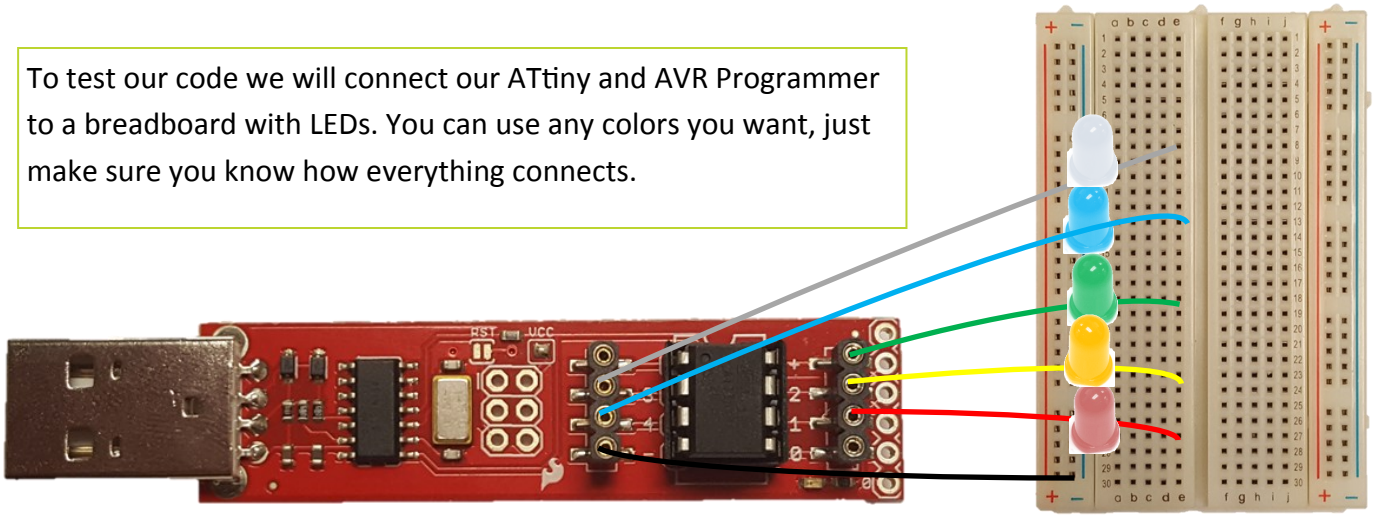


ATTiny85

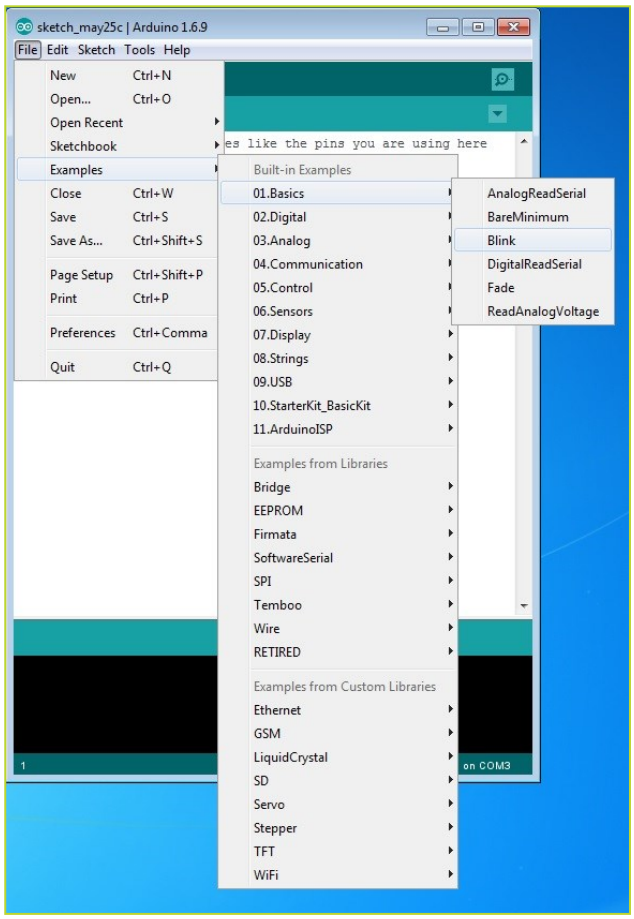


The ATtiny85 is a small microcontroller that is convenient for running small, simple programs. We will connect the ATtiny to the computer with an AVR Programmer. To create a program we will use Arduino. You can base your program off of an example program or you can create your own. When you are done programming you can take your ATtiny out of the Programmer, flatten the pins and add it to the copper tape circuit earlier in this packet, or a circuit of your own design!

To test our code we will connect our ATtiny and AVR Programmer to a breadboard with LEDs. You can use any colors you want, just make sure you know how everything connects.

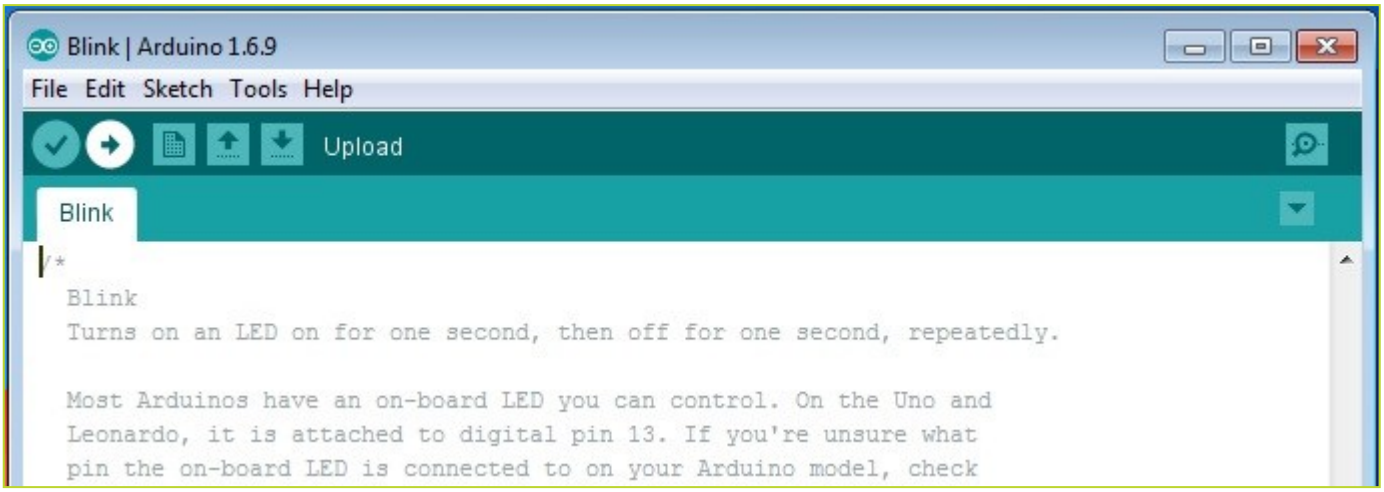


Example—Blink!



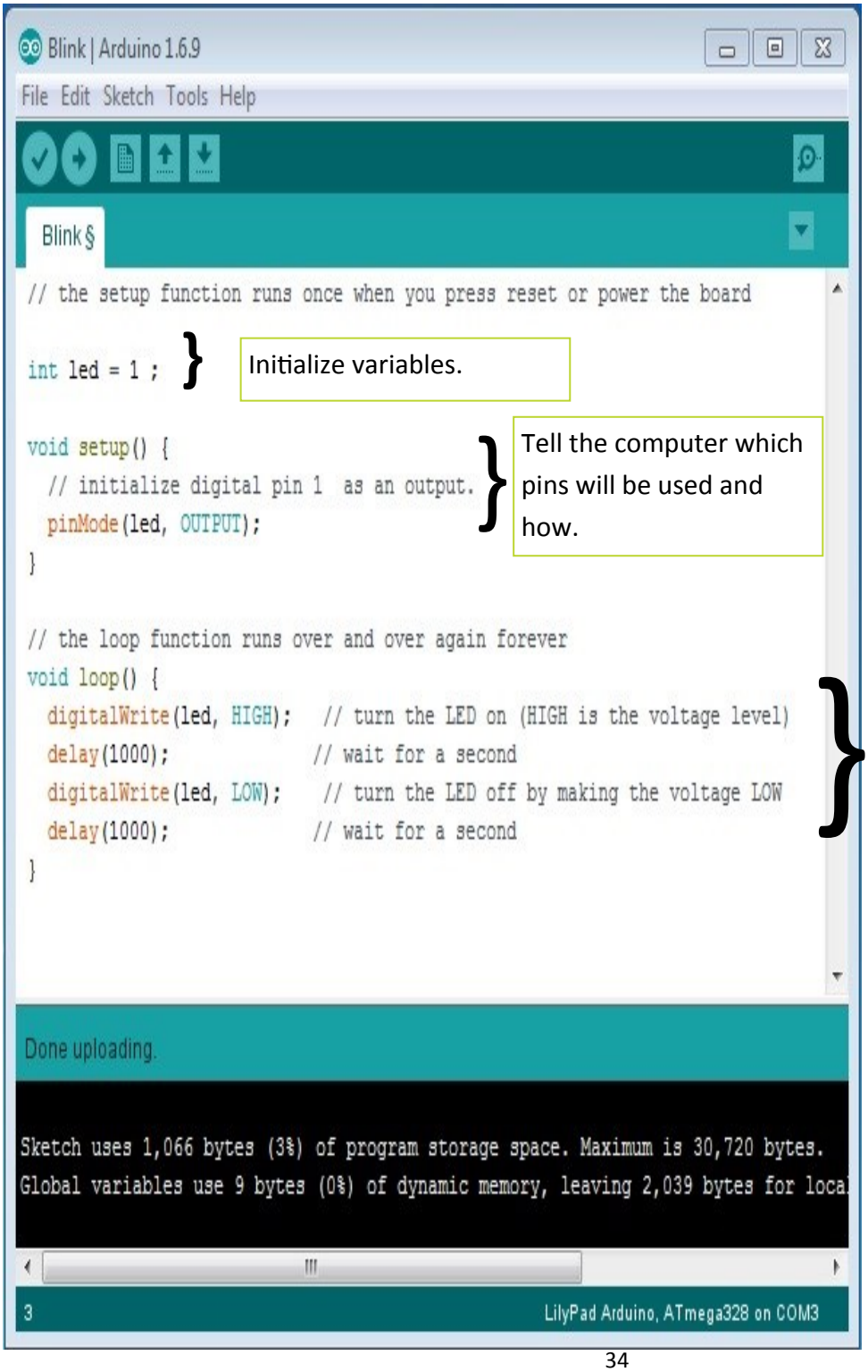
In Arduino, go to File ➞ Examples ➞ Basics ➞ Blink. Open it!

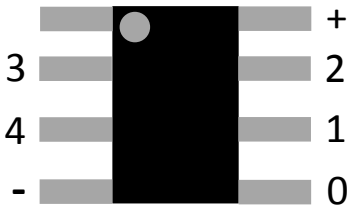
You will now see code for a simple program you can run! We'll talk about it more on the next page, but for now, click the "upload" arrow in the top left corner of the IDE and watch the program work! How would you describe what is happening to a friend? Could you do the same thing with a light switch in your home? If you wanted a friend to execute this code like your computer, what instructions would you give them?



How does Blink work?

There are three basic chunks to every Arduino program, in this order: Variable Declaration section, Setup Section, and Loop Section. They are all very important to the creation of your program! **Note that anytime you see "//" everything that follows is a comment, and won't be read by the computer.**





Now You Try!

Write some code that uses two LEDs and contains a different pattern depending on the LED. You will need to use two legs on the ATTiny85. Fill in the empty lines below.

```
int ledPin0 = 0; //First LED assigned to Pin 0
__________________     //Write code to assign another LED to pin 1

void setup() {
  // put your setup code here, to run once:
  pinMode(ledPin0, OUTPUT); //Sets pin 0 to output
  ____________________________   //Write code to set pin 1 to output

}

void loop() {
  // put your main code here, to run repeatedly:
digitalWrite(LedPin0, HIGH); //Turns on LED
delay(1000);                 //Stays on for 1 second
_____________________________    //write code to turn off the LED
_____________________________    //and wait a second.
// Write code to turn your other LED on and off

_____________________________

_____________________________

_____________________________

_____________________________
}
```

Compiling & Uploading Your Code

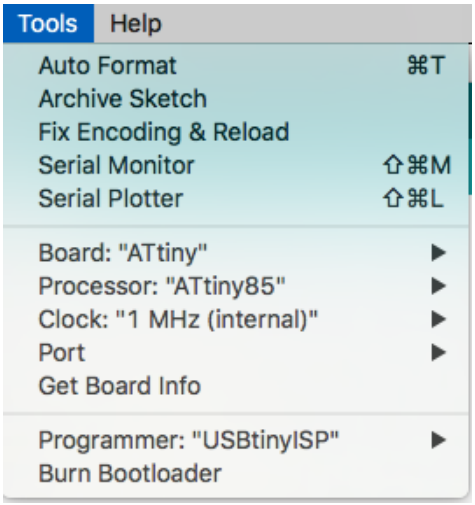
Now that you have written some of your own code, you need to compile and upload it to the ATtiny85.

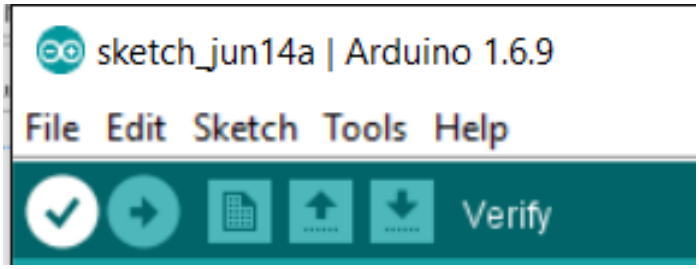
Before you load your code, in the "Tools" menu, you need to confirm that all of the settings are correct for the ATtiny. If these settings are not correct your code won't load and your ATtiny won't work.

Board: "ATtiny"

Processor: "ATtiny85"

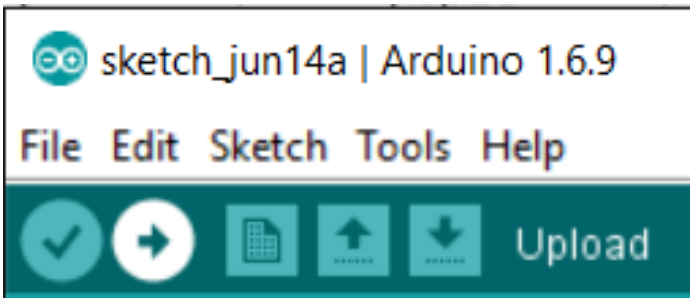
Clock: "1MHz (internal)

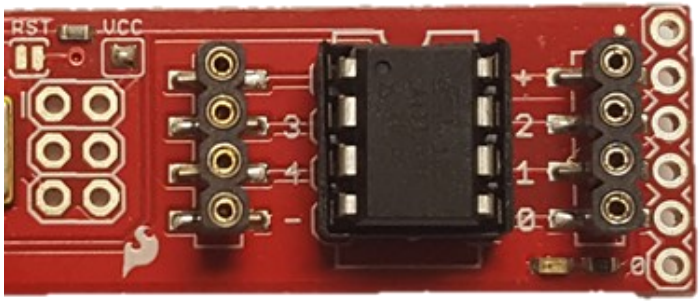




In the top left corner of the Arduino IDE, you'll see the checkmark and arrow buttons. The checkmark, known as "Verify", is kind of like "spell check" for your code. You can use this to check if there are any mistakes in your code that will prevent it from working.

The Arrow button, known as "Upload", will compile the code and upload it to the device you have plugged into your machine. This is how we will use the ATtiny85! You first need to upload the code onto the microcontroller in order to run your program.

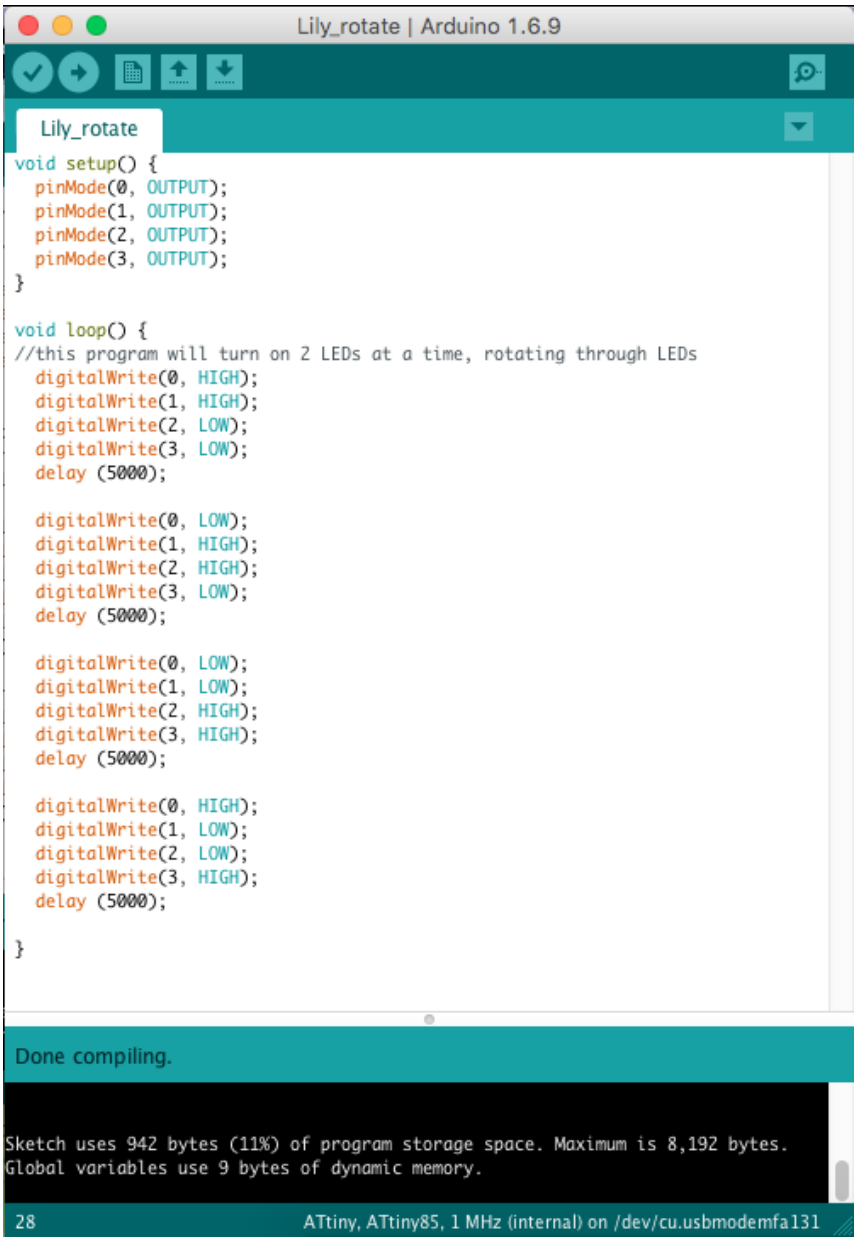




Plug the ATtiny85 into the AVR programmer as shown in the picture. The small "dot" in the top left corner of the ATtiny85 needs to be on the same side as the "notch" in the white outline. Plug the AVR into any USB port on your computer and you are ready to upload your code!

Example Code

Now that you know the basics about Arduino you can start to write your own code. By using the Arduino commands listed on this page you can make your own amazing program for your project. There are a few example codes on the next few pages that you can use.



Arduino Commands

pinMode()

digitalWrite()

digitalRead()

analogWrite()

analogRead()

shiftOut()

pulseIn()

millis()

micros()

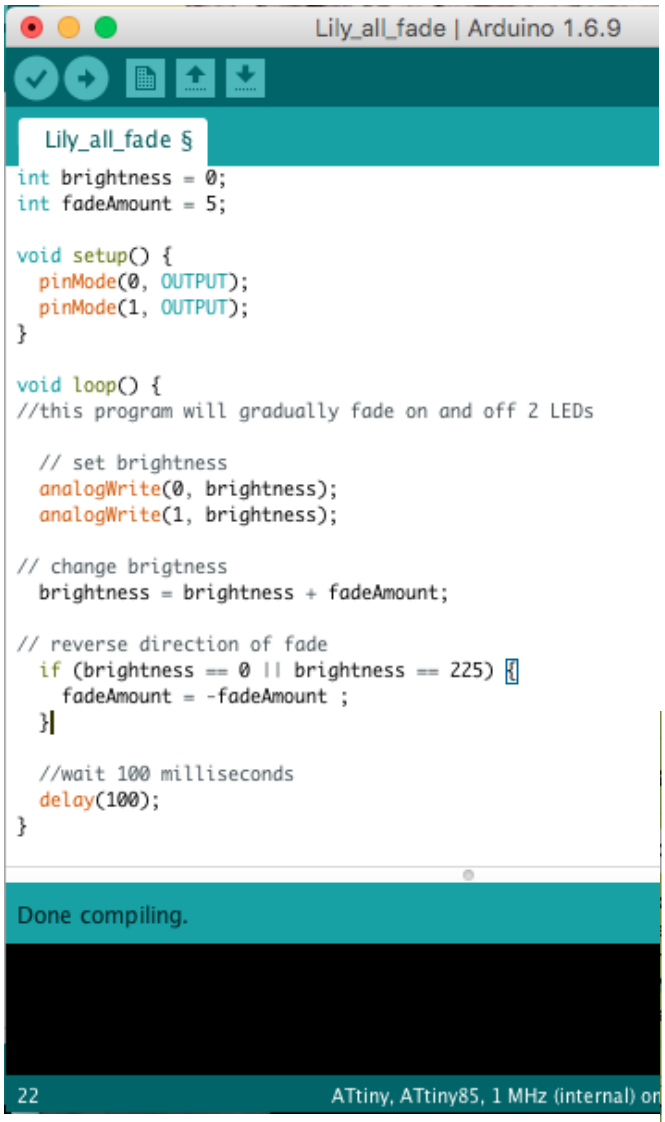
delay()

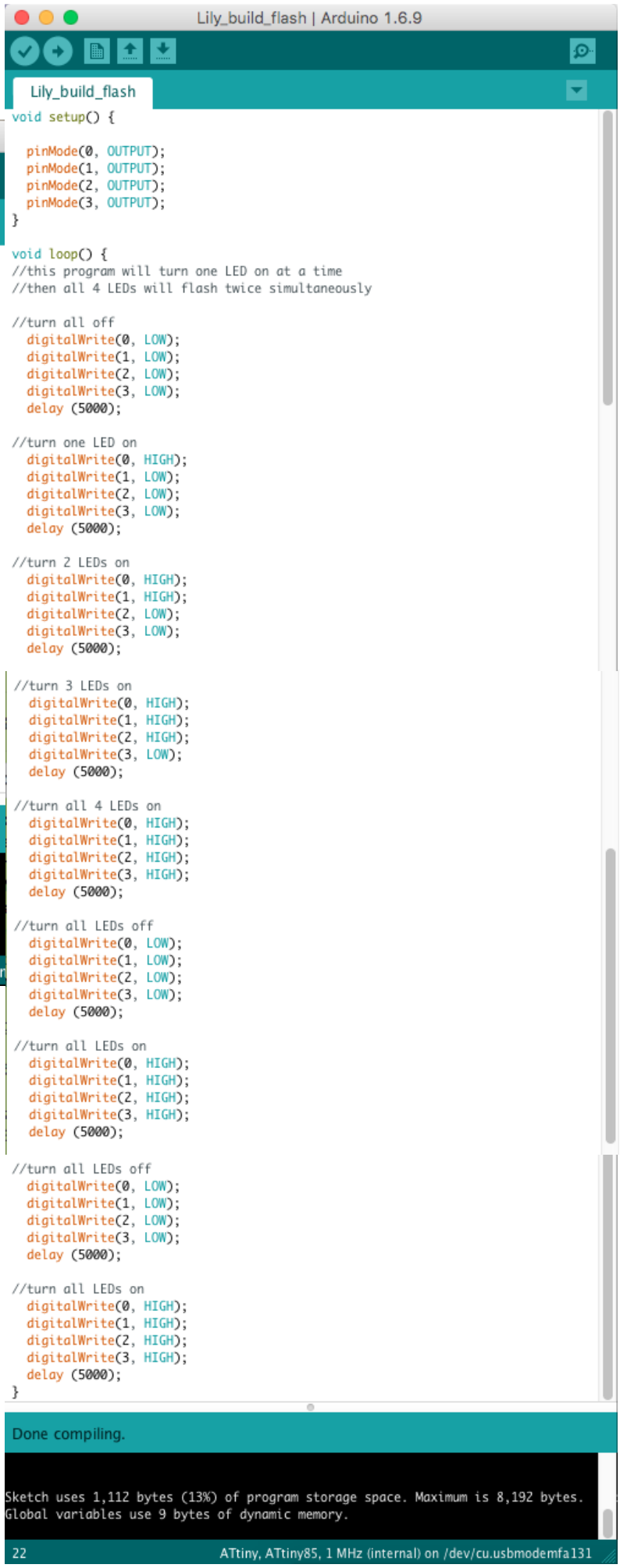
delayMicroseconds().

What does the code do?

Turn on two LEDs at a time, cycling through LEDs

More Example Code



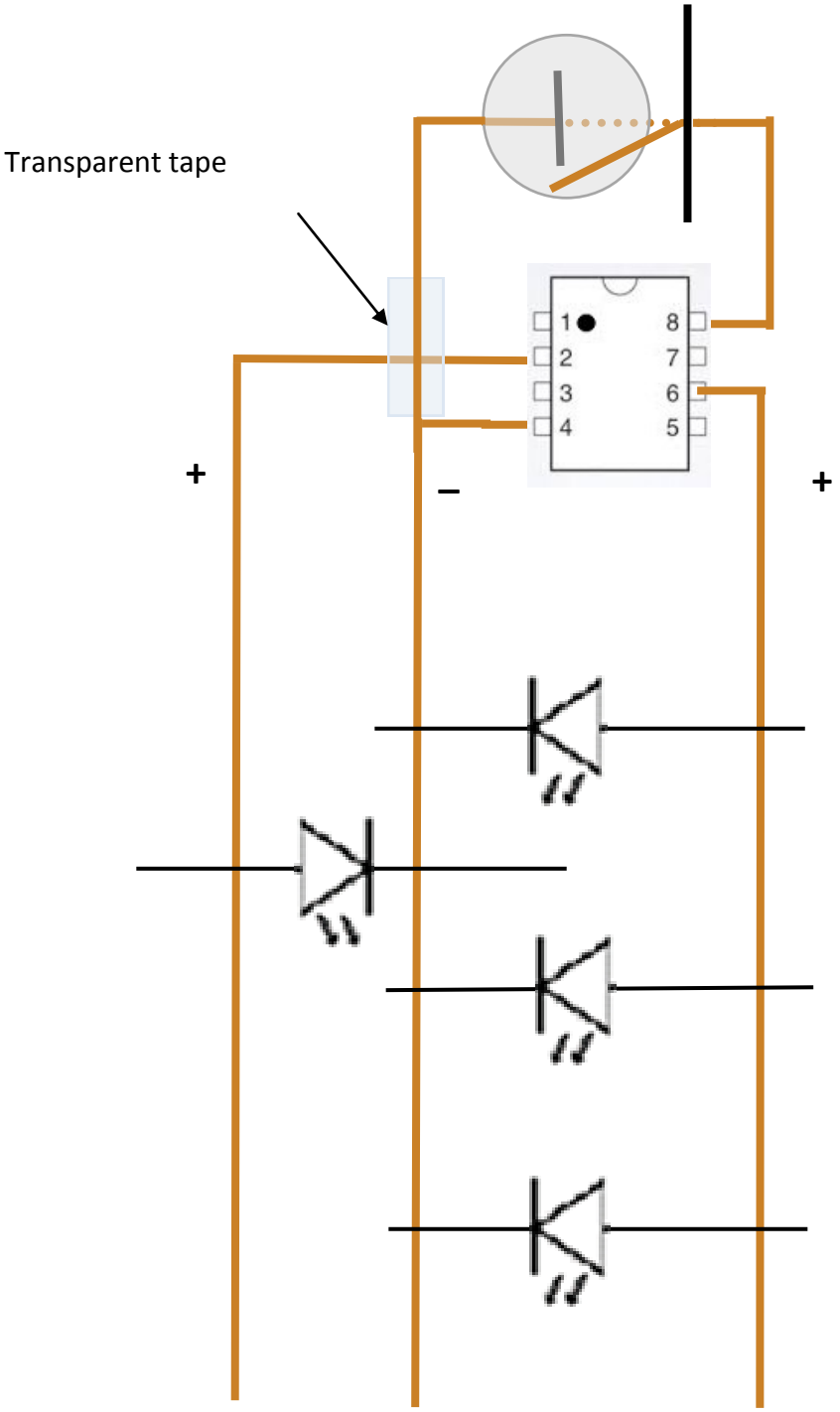


What does the code do?

Above: Lights attached to pin 0 & 1 will fade from dim to bright to dim, in a constant pattern. Note: In the code above, only pin 0 & 1 fade. Pins 0 & 1 are analog outputs, so they can fade. Pins 2, 3, & 4 are digital outputs and can only turn on and off.

Right: Turn on one LED at a time, then flash all LEDs on and off.

Now that you've coded your ATtiny you can attach it to a project. Be careful though, once you solder down the ATtiny you won't be able to change the code any more. You can add your ATtiny to the circuit shown below or you can design your own.



Program two pins on your ATTiny so that it will light up the lights in this circuit.

Add another side to your circuit, and at least on more LED

Solder the ATTiny and LEDs to the copper tape

LilyTwinkle Hack

The LilyTwinkle is a microcontroller board designed for wearables and e-textile projects. It can be sewn to fabric (or taped into notebooks or to hard surfaces) to add programmable fun to any project! The LilyPad is programmed with an ATtiny and comes with 6 pins: power (+), ground (-), and 4 pre-programmed pins. Once you have the Lily connected to the Programmer you can open Arduino and start programming. You can check out the sparkfun website for more information about the Lily. https://www.sparkfun.com/products/11590

You can add the Lily to a variety of projects. You can make a custom bracelet, you can also sew it onto a purse or an item of your own.

Regardless of what you decide to do—do NOT snip apart your Development Board until you are done programming and ready to sew!

Materials

AVR Programming Stick

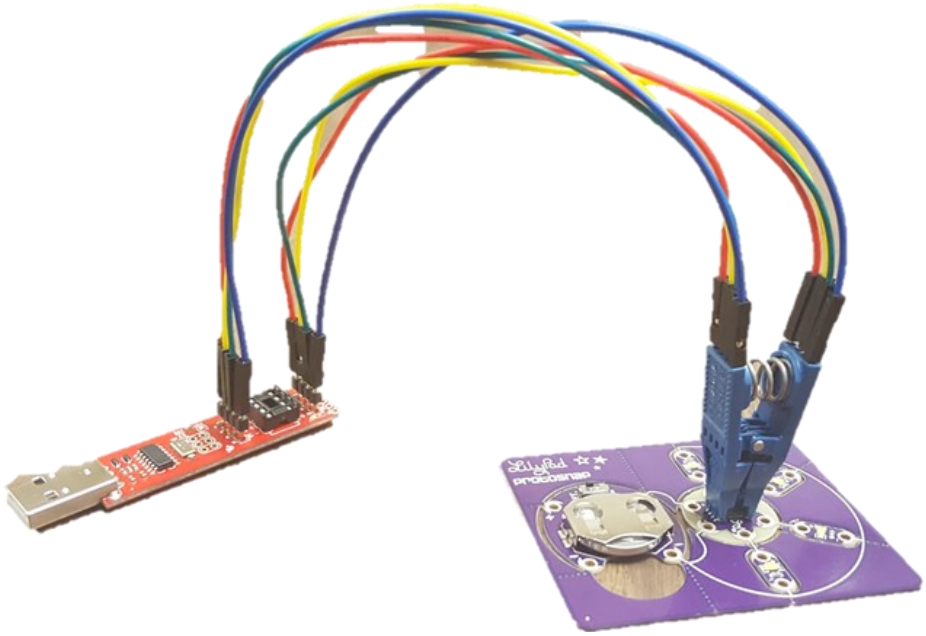
LilyTwinkle Protosnap

8-Pin SOIC IC Test Clip

Male to Female Jumper Wires

Hooking up the Twinkle to the AVR Programmer

We can reprogram the Lily by connecting it to the Programming Stick with the Test Clip and Jumper Wires. When connecting the Lily to the Programming stick make sure the correct side of the pins are connected to the correct side of the programmer. If we keep the Programming Stick so the numbers are right side up, the Lily can be arranged so that the numbers are also right side up, you can also look for a small circle indent on the upper left hand corner of the Lily.

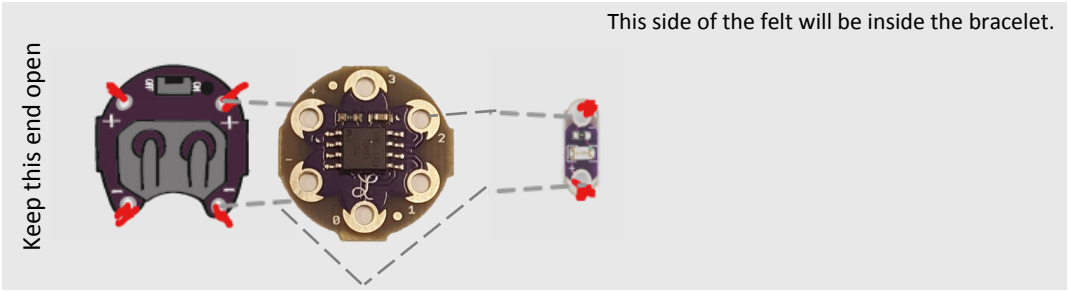


NOTES

LED Bracelet

Two strips of felt will be layered to form the bracelet.

1. Use regular sewing thread to attach the battery holder, LilyTwinkle, and LED to one side of the layer of felt that will be next to your skin.
2. Use conductive thread to sew traces from the positive tab of the battery holder to the positive tab of the Twinkle and from the negative tab of the battery holder to the negative tab of the Twinkle.
3. Use conductive thread to sew traces from the coded pad of the Twinkle to the positive tab of the LED and from the negative tab of the Twinkle to the negative tab of the LED.



4. Flip the bottom layer over and use regular thread to sew the stud parts of the two snaps to the end that does not have the battery.

5. Place the top layer over the bottom layer and mark where the LED will shine through.
6. Incorporate the LED as you embroider and stitch beads and other decorations to the outside of the top layer.
7. Use regular sewing thread to attach the socket parts of the two snaps to the end of the outside of the top layer.

8. Connect the two layers with regular sewing thread. Sew around both long edges and the short edge that is not near the battery holder.
9. Leave the short end near the batter holder open.
10. Insert the battery and turn on the switch. Then snap the bracelet onto your wrist!!!

MATERIALS

1 Coin battery

1 Battery holder

1 LilyTwinkle

2 Metal snaps

LEDs

Needle

Conductive thread (shown as grey stitches in the drawings)

Regular sewing thread (shown as red stitches in the drawings)

2 strips of felt material, 8-9 inches long and 1.5-2 inches wide

Troubleshooting Your Circuit:

LEDs don't light, blink, or don't stay lit

Double check that your battery is secured and that the positive tab of the battery is connected to the positive tab of the LED

Be sure the conductive thread traces don't cross each

Light Up Bag

Program your Twinkle to make a pattern using 2 or more LEDs.

1. Decide where you want to place your Twinkle and your LEDs.
2. Trace your circuit so that each LED is connected to a pin and the negative pin of the Twinkle, make sure none of the paths intersect (if they have to, add a piece of fabric between the lines so they don't touch).
3. Using the conductive thread, sew the Twinkle onto the purse and follow your traced lines to the LEDs. Make sure to use small stiches.
4. Using embroidery thread, sew your own design onto your purse.

MATERIALS

1 Coin battery

1 Battery holder

LilyTwinkle

LEDs

Needle

Conductive thread

Embroidery thread

Purse

Troubleshooting:

Be sure to check every connection! Are the LEDs connected in the right direction (positive is where it is supposed to be and negative is where it's supposed to be)? Make sure there are no short circuits!

NOTES

Other Resources for Electronic Projcts

Sparkfun: https://www.sparkfun.com/

Sparkfun is an online store that sells the parts to make electronics projects possible. You can also access tutorials and information about products and projects.

Arduino: https://www.arduino.cc/

Arduino is an open-source hardware and software electronics platform. It is geared towards anyone making interactive electronics projects.

Sew Electric: http://sewelectric.org/

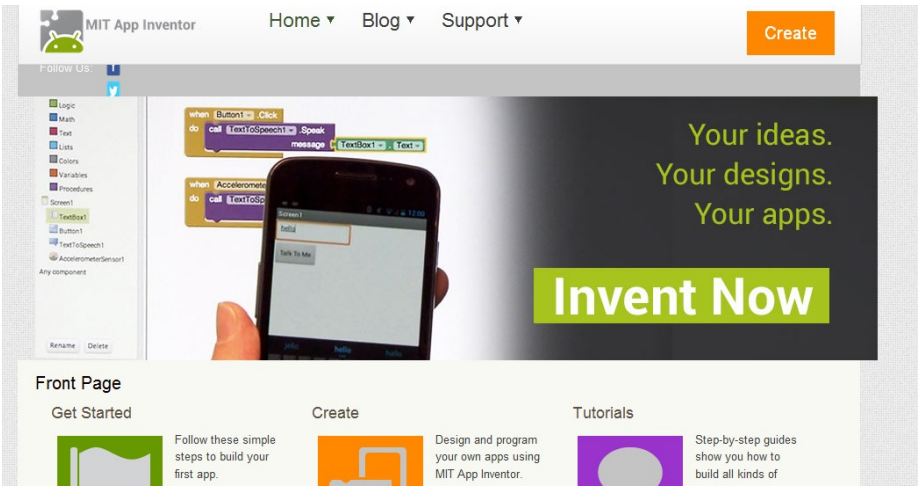
DIY projects for people interested in sewing, crafting, electronics, and programming.

NOTES

App Inventor 2

Now that we've explored some tile-based programming, you're ready for more of a challenge! App Inventor is another cloud-based tool developed by MIT that runs through a Web browser. You will be using App Inventor 2 to design and create your own app! To get started, you need a Google account. Login with your Gmail (or school email account if it is tied to Google) to get inventing.

After learning the basics of App Inventor 2, start brainstorming an app of your own design. What will it do? What will it look like? Do you have the skills necessary to create your design? Do you need additional resources (time, tutorials, etc) to program your app?



Instructions for set-up and connecting an Android device:

http://appinventor.mit.edu/explore/ai2/setup-device-wifi.html

Instructions for setting up the Emulator:

http://appinventor.mit.edu/explore/ai2/setup-emulator.html

Tour of App Inventor 2:

http://appinventor.mit.edu/explore/designer-blocks.html

Beginner Tutorials:

http://appinventor.mit.edu/explore/ai2/beginner-videos.html

App Inventor 2 Challenge

Let's put your knowledge of App Inventor 2 to use! Your challenge, is to create an app that satisfies one of the two options below:

1.) Apps for a Better World

The MIT App Inventor team is looking for innovative apps from creative programmers like you that illustrate App Inventor's versatility and functionality. Winning apps will be featured on the App Inventor homepage and need to be from one of the four areas below:

- Apps for Communities (apps that help organizations, companies, governments)
- Apps for Individuals (apps that help or entertain individuals)
- Apps for Research (apps that support research)
- Apps for Education (apps that support education)

To enter your app into this program you must email aiwebreview@mit.edu with "Application for App of the Month" as the subject line. Your email must contain the following information:

- Your App's title
- Which category(ies) you're submitting it for
- What the app does
- Why did you build it?
- What is your name, age, profession?
- What is the current status of the app? (e.g., is it currently in the Play Store?)
- 1-3 screenshots of the app
- App's .apk file OR a link to the app on the play store, if applicable

2.) Create a game (this is G.A.M.E.S. camp, after all!)

- A well-designed user interface with at least one button
- Media (sound, image, etc) played in response to an event
- Some decision-making (an if-block)
- A timer or clock component
- A procedure (a set of sequence statements that you refer to as a single command)

Try out the MoleMash game tutorial for inspiration—you'll also gain a deeper understanding of what App Inventor 2 can do!

The Google Play Store—Get your App Out There!

You love the app that you've made and feel like it has potential to improve the world or entertain the masses—so now what? You can add your app to the Google Play Store and the world can start downloading and using it!

To start, you'll need to register (and pay the $25 registration fee) at the Google Play Store's Developer Console:

https://play.google.com/apps/publish/signup/

Once registered, follow these instructions for uploading your app into the store:

https://support.google.com/googleplay/android-developer/answer/113469?hl=en

Other Resources for Coding Games

Stencyl: www.stencyl.com

Quick and easy way to create games for iPhone, iPad, Android, Mac, Windows, and Flash without code.

Kodu: http://www.kodugamelab.com

A visual programming language, create games and stories for your PC or XBox.

AgentCubes: https://www.agentcubesonline.com/

A new 3D web-based programming & modeling tool that allows you to turn 2D images into 3D shapes.

Pencil Code: https://pencilcode.net/

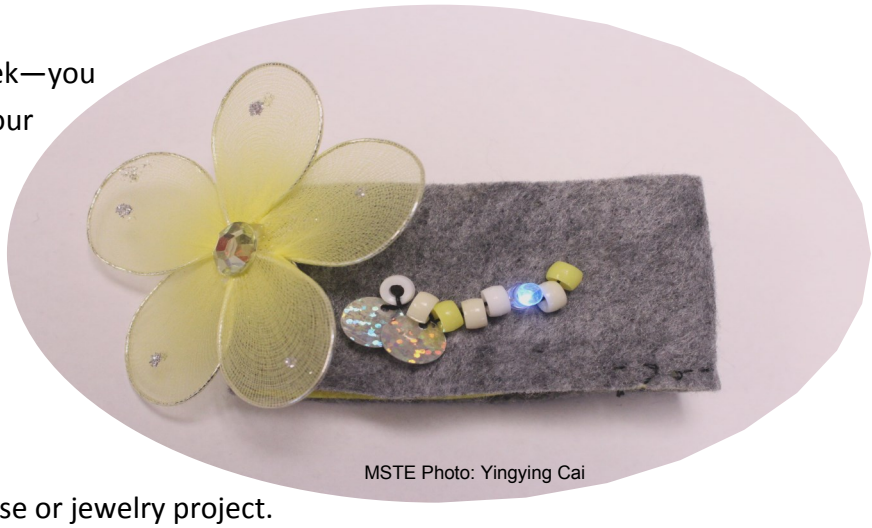
Create games, music, art, and stories while learning web development skills and professional programming languages.

NSB/AppStudio: https://www.nsbasic.com/

NSB/AppStudio is a JavaScript programming environment positioned above graphic tools like Scratch and AppInventor. User interfaces are designed using drag and drop, while code is written in real world languages like JavaScript or BASIC.

NOTES

Final Projects



MSTE Photo: Yingying Cai

We've covered a lot of ground so far this week—you can code, you can stitch, and you can code your stitching! You've seen a variety of projects and techniques as inspiration and now it is time to choose your final project. You will be sharing your project with your parents/ guardians during the Closing Ceremony on Saturday.

Join a specialty group and learn more about incorporating your Lilypad Arduino into a purse or jewelry project.
Use copper tape and LEDs to create art. Design and develop an App and submit it to the Google Play Store.
Imbed a secret message into a photo by altering a small number of pixels.

What you decide to do is ultimately up to you, but it *must* involve programming. (This is Computer Science camp, after all!)

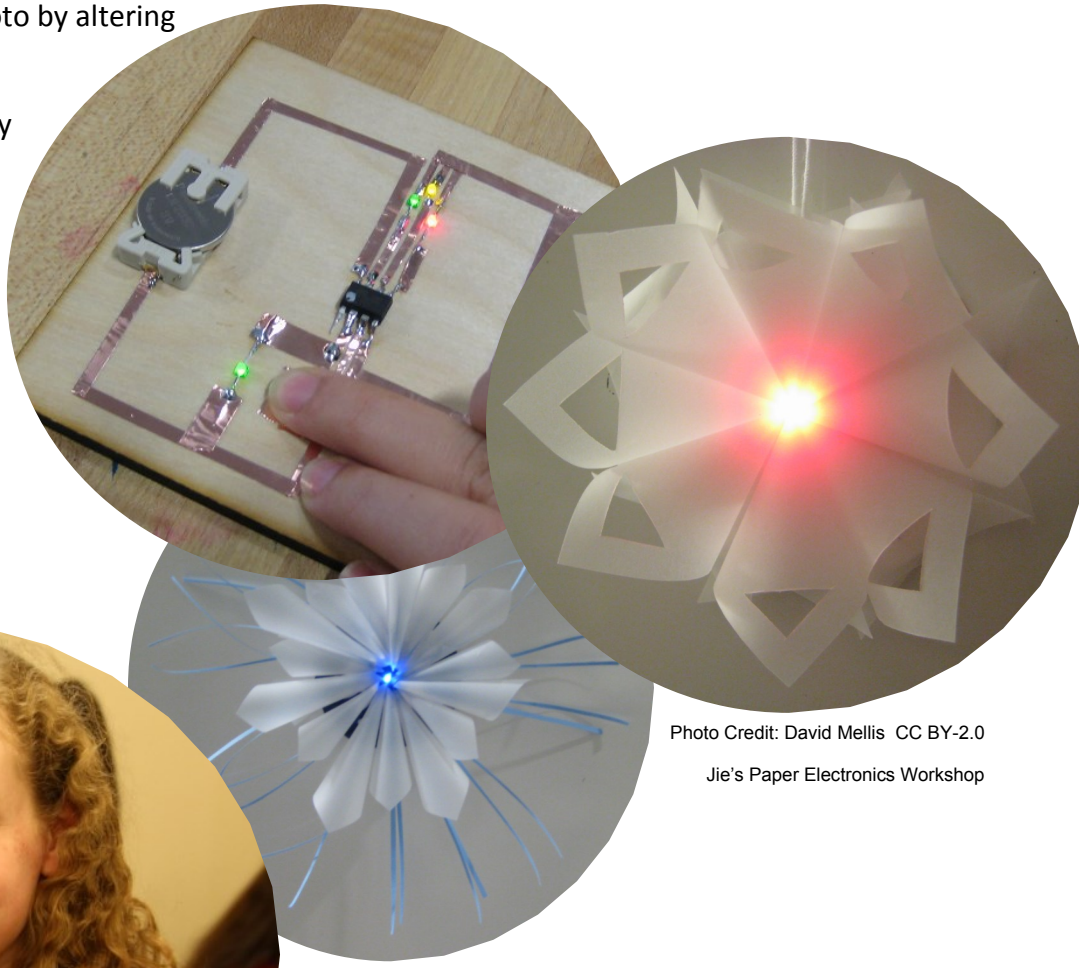


Photo Credit: David Mellis CC BY-2.0

Jie's Paper Electronics Workshop

Final Project

Project

Ideas & Inspiration

CS Component

Materials Needed

NOTES

NOTES

Women in Engineering UIUC

Computer Science G.A.M.E.S. Camp 2016

Samantha Lindgren

Jana Sebestik

Lizzie Diamond

Computer Science GAMES Camp receives support from all of these projects at the University of Illinois.

MSTE | **EDUCATION** AT ILLINOIS

Office for Mathematics, Science, and Technology Education enhances student achievement and teaching performance in the fields of mathematics, science, and technology. http://mste.illinois.edu/

CYBER RESILIENT ENERGY DELIVERY CONSORTIUM

The Cyber Resilient Energy Delivery Consortium (CREDC) Education team continues the work of the TCIPG Education project. The team develops interactive lessons and activities designed to link researchers, educators, consumers, and students. The materials illustrate challenges, trade-offs, and decisions required for secure and economical power delivery. The project seeks to involve families learning together while creating interest in STEM disciplines and careers. The project website offers a variety of hands-on and virtual energy related activities and challenges. credc.mste.illinois.edu

This material is based upon work supported by the Department of Energy under Award Number DE-OE0000780.

Hazards SEES: GIC Hazard Prediction: From the Solar Wind to Power System Impacts project is intended to improve the scientific understanding of the processes governing the impacts on our power distribution system of severe solar storms. The team is studying the relationship between solar wind drivers and magnetic field perturbations on the ground, developing improved models of induced electric fields, and enhancing prediction capabilities for GIC hazards.

This work is supported by the National Science Foundation (NSF) under Award Number NSF 15-20864.

4-H Computing Connections develops informal computer science activities for elementary and middle school youth. Working with Extension, faculty and staff are creating a curriculum that addresses computer programming as well as fundamental computer science concepts. The project team trains 4-H volunteers and 4-H teens as teachers to run the activities, with the potential for statewide, and even nationwide 4-H club adoption of the curriculum.

This project is funded by the University of Illinois Extension and Outreach Initiative. **http://web.extension.illinois.edu/initiative/**

Illinois Cyber Security Scholars Program is supported by a grant from the National Science Foundation. It capitalizes on the particular strengths of the University of Illinois in computer science, engineering, and intellectual property law, the Illinois Cyber Security Scholars Program (ICSSP) trains the next generation of specialists capable of protecting the nation's cyber infrastructure.

The Program in Digital Forensics team is developing a new undergraduate educational curriculum in digital forensics in order to address a national shortage of trained cyber-security professionals. *Forensics* is the use of scientific methods to obtain information with legal significance, and *digital forensics* is a type of forensics that deals with recovery and investigation of data in digital devices . This work is supported by the National Science Foundation under Grant No. 1241773.