



Tema del módulo | Tecnología e ingeniería

Autor | Dennis (DJ) Pevey, Becario de Comunicación Científica (*Science Communication Fellow*)

Prerrequisitos | Los estudiantes deben tener conocimiento básico de herramientas tecnológicas.

Evaluación | Rúbrica de proyectos y tareas *STEM*¹ (provista al final del módulo)

Ritmo | 1-3 períodos de clase (de 45 minutos cada uno)

¹ *STEM* es un acrónimo que significa (en inglés): ciencias (*science*), tecnología (*technology*), ingeniería (*engineering*) y matemáticas (*math*). También en español, CTIM.

Materiales

- Cada grupo necesitará:
 - Una computadora que no sea modelo Chromebook
 - 1 mini cable USB (<http://nautl.us/2v1zbTH>)
 - 1 Arduino Uno/Genuino (<http://nautl.us/2uBQw4x>)
 - 1 placa de pruebas (también conocida como tablero de circuitos y *breadboard* en inglés) (<http://nautl.us/2u2uWml>)
 - Varios cables de puente 20AWG (<https://nautl.us/2ukbiYc>) (5-10 por grupo)
 - Un surtido de bombillas LED (<http://nautl.us/2eRHcEI>) (diodo emisor de luz)
 - Unos pocos resistores Ohm (<http://nautl.us/2wpb9lk>) (1-5 por grupo)
- Opcional (por grupo):
 - Placa de montaje de plástico para el Arduino y la placa de pruebas (<http://nautl.us/2vQO3Aw>)
 - Tijeras pequeñas para cortar alambre
 - lupas
- Para información adicional favor de evaluar nuestro [presupuesto sugerido](http://nautl.us/2KxaN6i) (<http://nautl.us/2KxaN6i>) para todos los módulos de esta serie.

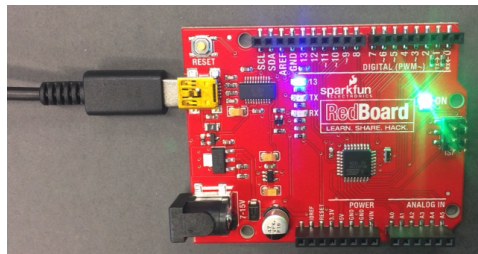
Resumen

Este módulo presentará a los estudiantes los conceptos básicos de codificación y microcontroladores utilizando el *hardware* de Arduino Uno. Los estudiantes aprenderán los componentes básicos del *hardware*, *software* y de las placas de pruebas para construir circuitos LED simples y múltiples. Analizarán lenguaje de codificación básica, manipularán códigos provistos para cambiar las variables de un dispositivo de salida, organizarán un circuito básico LED utilizando una placa de pruebas y finalmente, escribirán su propio código de socorro SOS. Los estudiantes trabajarán con sus compañeros de clase para comparar códigos y observar las distintas maneras de crear códigos para obtener los mismos resultados.

Objetivos y resultados del aprendizaje

Los y las estudiantes:

- identificarán aplicaciones prácticas para los microcontroladores.
- identificarán aplicaciones prácticas para las placas de prueba.
- aprenderán y aplicarán tecnología básica de programación.



Preguntas guía

1. Menciona algunos ejemplos de dispositivos de entrada y salida utilizados en la comunicación con una computadora.

Respuestas sugeridas: (ambos) teléfonos celulares, microcontroladores; (entrada) tomógrafos, ratón de computadora, sensor de temperatura, sonar multihaz, cámaras; (salida) bocinas, impresora 3D, monitor, brazo manipulador de ROV, luces, motores.

2. ¿Cuáles son los beneficios de utilizar un microcontrolador?

Respuestas sugeridas: económico, fácil de usar, puede ser utilizado al programar sensores para una variedad de usos investigativos y prácticos; probar circuitos antes de finalizar productos o proyectos.

3. ¿Por qué se utilizan resistores en los circuitos?

Respuestas sugeridas: para limitar la corriente que fluye a través de varios componentes como por ejemplo, las bombillas LED, para mantener funcionamiento apropiado.

Actividad/Tareas

Los estudiantes:

- determinarán los roles y la organización grupal para la distribución del trabajo (sugerencias: manejador del equipo, manejador de materiales, lector, ingeniero, asistente del equipo si fuese necesario, etc.).
- leerán la introducción y completarán las preguntas guía.
- recolectarán todos los materiales.
- trabajarán a través de todas las partes del procedimiento y completarán todas las tareas asociadas.

Este módulo está alineado a los *Next Generation Science Standards* y los *Common Core Standards* para la enseñanza en los Estados Unidos. Puede compararlos con los estándares de su localidad para determinar la alineación correspondiente.

Enlace a los *Next Generation Science Standards*

MS-ETS 1-2: Evaluar soluciones de diseño que compiten entre sí utilizando un proceso sistemático para determinar cuán bien estas pueden satisfacer los criterios y las limitaciones de un problema.

MS-ETS1-3: Analizar datos de pruebas para determinar semejanzas y diferencias entre varias soluciones de diseño para identificar las mejores características de cada una que se puedan combinar en una nueva solución que cumpla mejor con los criterios para alcanzar el éxito.

MS-ETS 1-4: Desarrollar un modelo para generar datos para pruebas iterativas y la modificación de un objeto, herramienta o proceso propuesto, de manera que se pueda alcanzar un diseño óptimo.

Enlace a los *Common Core Standards*

CCSS.ELA-LITERACY.RST.6-8.3: Seguir un procedimiento de múltiples pasos precisamente al conducir experimentos, al tomar medidas o al desempeñar tareas técnicas.

CCSS.ELA-LITERACY.RST.6-8.4: Determinar el significado de símbolos, términos clave y otras palabras y frases específicas del dominio tal como se usan en un contexto científico o técnico específico.

Extensiones y adaptaciones

Nivel introductorio | Utilice esta [presentación](http://nautl.us/34RISox) (<http://nautl.us/34RISox>) para guiar a todos los estudiantes a través de la lección al mismo ritmo.

Nivel avanzado | Motive a los estudiantes a crear su propia pantalla LED u otro proyecto de circuitos escribiendo su propio código o buscando guías de otros proyectos en línea. Los estudiantes pueden compartir sus ejemplos con la clase. Las páginas electrónicas y tutoriales pueden ser compilados en una lista de recursos para la clase.



Solucionario para la actividad de calentamiento:

```
1 D int LED_BuiltIn = 13;

2 E void setup( ) {

3 C pinMode(LED_BuiltIn, OUTPUT);
    }

4 G void loop( ) {

5 A digitalWrite(LED_BuiltIn,HIGH);

6 F delay(1000);

7 B digitalWrite(LED_BuiltIn, LOW);

8 H delay(1000);
```

Banco de respuestas

- A. Enciende la bombilla LED
- B. Apaga la bombilla LED
- C. Declara la bombilla LED como un pin de salida
- D. Asigna un nombre a la bombilla LED conectada al pin 13 en el Arduino que será utilizado a lo largo del bosquejo
- E. Designa cuáles funciones serán efectuadas en la parte organizativa del bosquejo
- F. Le dice a la bombilla LED que se mantenga encendida por un segundo
- G. Designa cuáles funciones serán repetidas ("looped")
- H. Le dice a la bombilla LED que se mantenga apagada por 1 segundo

¡Prepare el escenario!

Tenga un modelo de iluminación Arduino/LED listo y encendido (observe un ejemplo en este enlace: <http://nautil.us/2WOaExN>). Pida a los estudiantes que discutan lo que piensan que está pasando para hacer que las luces se enciendan con ese patrón. Motíuelos a tener una discusión de clase para evaluar su conocimiento y experiencia previa con relación a este tema.

Educador: Procedimiento

Preparación previa | Se sugiere que los estudiantes trabajen en grupos pequeños para completar este módulo.

☐ Recomendaciones:

- Organice los materiales de los estudiantes con anticipación y colóquelos en cada una de las estaciones de trabajo. Establecer roles para los estudiantes dentro de cada grupo pudiese ser beneficioso (por ejemplo: manejador de materiales, anotador de tiempo, ingeniero, etc.) para asegurar eficiencia en la colaboración.
- Guía de resolución de problemas técnicos para Arduino: <https://www.arduino.cc/en/Guide/Troubleshooting>*
- En la cuarta parte del procedimiento, si las bombillas LED no parpadean al conectarlas, pida a los estudiantes que desconecten el USB y verifiquen la orientación de los alambres de las bombillas LED o resistores y que finalmente vuelvan a conectar el USB.

Educador: Procedimiento (continuación)

- ☐ Contestaciones para sección “Adentrándonos más”:
- En inglés, las placas de pruebas son conocidas como *breadboards* o tablas de pan. ¿Por qué los llaman de esa manera? **Respuesta sugerida:** En el pasado, las tablas de pan eran comúnmente utilizadas como plataforma para experimentar con circuitos encendidos, cuando los artículos electrónicos eran de gran tamaño.
- Menciona dos beneficios de utilizar la placa de pruebas: **Respuesta sugerida:** Le permite a las personas probar circuitos para ver cómo reaccionan bajo una combinación específica de parámetros. La placa puede ser utilizada para probar nuevas partes, como en los circuitos integrados (ICs por sus siglas en inglés).
- ¿Cuál es otro nombre para las líneas conductoras y para qué se utilizan? **Respuesta sugerida:** También conocidas como regletas terminales, son utilizadas para la conexión de cables u otros componentes y para la conducción de electricidad a través de los artículos en la misma fila.
- ¿Cuál es el propósito de tener las líneas de alimentación de corriente que corren verticalmente a lo largo de los dos lados de la placa de pruebas? **Respuesta sugerida:** El propósito es proveer acceso fácil a la energía donde sea necesaria en un circuito.



*En inglés.



Solucionario para las aplicaciones de los estudiantes

Vea [la traducción](#) al español de los comentarios incluidos en el código:

Parte 1: Código de revisión:

```
int led = 9;           // the PWM pin the LED is attached to
int brightness = 0;    // how bright the LED is
int fadeAmount = 5;    // how many points to fade the LED by

// the setup routine runs once when you press reset:
void setup() {
  // declare pin 9 to be an output:
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  // set the brightness of pin 9:
  analogWrite(led, brightness);

  // change the brightness for next time through the loop:
  brightness = brightness + fadeAmount;

  // reverse the direction of the fading at the ends of the fade
  if (brightness <= 0 || brightness >= 255) {
    fadeAmount = -fadeAmount;
  }
  // wait for 30 milliseconds to see the dimming effect
  delay(30);
}
```

Parte 2: Escribe un Código SOS: Crea tu propio código

para hacer que una bombilla LED parpadee en un patrón SOS (3 pulsos cortos, seguidos de 3 pulsos largos, seguidos de 3 pulsos cortos). Toma una captura de la pantalla incluyendo tu código o anótalo a continuación.

```
int ledPin = 9; // LED connected to digital pin 9

int s = 300; // 0,3 second delay for 'S' letter
int o = 800; // 0,8 second delay for 'O' letter
int pause = 100; // 0,1 second delay between letters

void setup() {
  // put your setup code here, to run once:
  pinMode(ledPin, OUTPUT);
}

void character(int speed, int pause) {
  for (int i = 1; i <= 3; i++) {
    digitalWrite(ledPin, HIGH);
    delay(speed);
    digitalWrite(ledPin, LOW);
    delay(300);
  }
  delay(pause); //delay between letters
}

void loop() {
  character(s,pause);
  character(o,pause);
  character(s,pause);
  delay(2000); //2 seconds delay for after the word SOS
}
```

Pregunta de discusión: Compara el código de tu grupo al de otros. ¿Pueden distintos códigos producir los mismos resultados?

Respuesta: Sí, los códigos pueden ser escritos de forma distinta para producir los mismos resultados.

Introducción

¿Te gustaría programar una computadora para que haga tu trabajo? Los microcontroladores, como Arduino, proveen la llave para la creación de proyectos personalizados en electrónica. En esta serie de módulos, aprenderás cómo programar un microcontrolador para obtener datos científicos con un sensor. Podrás aplicar tus nuevas destrezas para diseñar tu propio proyecto en tu escuela o comunidad.

Las actividades en esta lección te permitirán familiarizarte con los conceptos básicos para la configuración de un microcontrolador y con la escritura de código, para hacer que una bombilla LED parpadee en un patrón determinado.

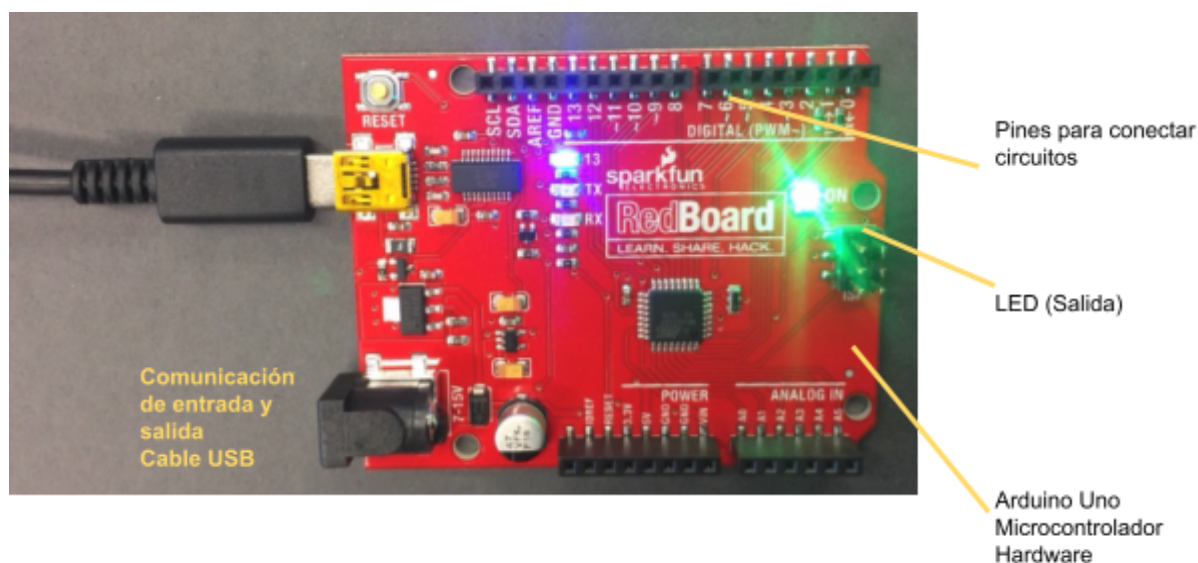
Objetivos del aprendizaje

- Identificar las aplicaciones prácticas de los microcontroladores en la vida cotidiana.
- Aprender y aplicar términos y destrezas básicas en programación.
- Utilizar destrezas de codificación para programar un microcontrolador.

Información de trasfondo

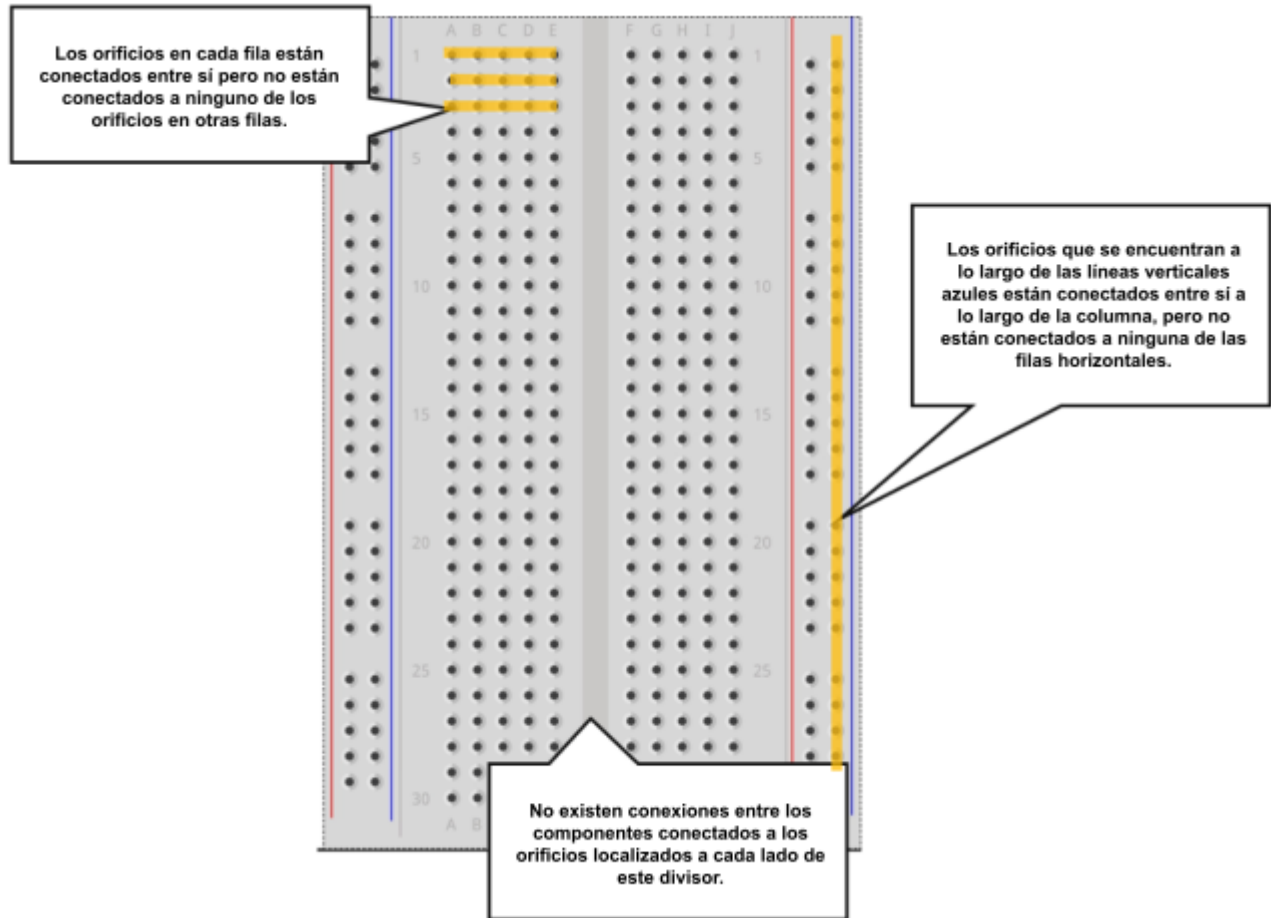
En su forma más básica, los **microcontroladores** contienen un pequeño núcleo procesador, una memoria y porciones programables de entrada y salida que le dan a la computadora el poder para detectar información proveniente del ambiente que le rodea. Los microcontroladores se usan en muchos productos electrónicos comunes, desde aparatos de cocina, herramientas eléctricas y juguetes, hasta dispositivos médicos y equipo científico.

Desarrollado en el inicio de la década del 2000 por investigadores italianos, Arduino se ha convertido en uno de los microcontroladores más comúnmente utilizados en el mercado. Su pequeño perfil, bajo costo y facilidad de uso tanto para ingenieros como no ingenieros, presentan posibilidades para innovación en una variedad de campos STEM.



Normalmente, los proyectos de electrónica requieren **soldadura**. Mediante este proceso, se fusionan los componentes en sus puntos de conexión añadiendo un metal de relleno derretido (estaño). Sin embargo, una vez las piezas están soldadas, es muy difícil separarlas. Las **placas de pruebas** nos proveen una

alternativa a este problema. Las partes internas de la placa son conductoras. Esto significa que permiten un flujo de electrones entre todos los componentes que se encuentren conectados a los múltiples orificios sin tener que estar soldados. Las placas de pruebas se pueden usar una y otra vez, y son una excelente opción para la construcción de prototipos de circuitos.



Para estas actividades, necesitarás una placa de pruebas, un Arduino y otros pocos componentes para construir un circuito simple de LED.

El código Arduino es conocido como un bosquejo. Los **bosquejos** se escriben en un lenguaje de programación llamado C. Este lenguaje reconoce mayúsculas y minúsculas (siempre debes verificar la función de *caps lock* o todo en mayúsculas) y requiere un punto y coma (;) al final de cada línea de código. Los bosquejos son cargados desde la computadora a un Arduino para dirigir una tarea específica. Aquí presentamos unos enlaces para que puedas aprender los conceptos básicos de la escritura de código Arduino:

Programa Blink: <http://nautl.us/2ile93V> * (alternativa en español: <https://nautl.us/2WNGnz2> **)

Introducción a Arduino: <https://nautl.us/2KYkudQ> **, <https://nautl.us/2KG2mFH> **



* En inglés



* En español

Preguntas guías

1. Menciona algunos ejemplos de dispositivos de entrada y salida utilizados en la comunicación con una computadora.
2. ¿Cuáles son algunos beneficios de utilizar un microcontrolador?
3. ¿Por qué se utilizan resistores en los circuitos?

Noticias Nautilus de última hora

A bordo del buque de exploración Nautilus, los ingenieros de datos utilizan microcontroladores llamados *Raspberry Pi* (imagina un Arduino con un puerto conector de ethernet) para correr quioscos de datos para herramientas científicas, registrar temperatura y humedad alrededor de las computadoras del sonar y para seguir la trayectoria de las condiciones del clima en el cuarto donde se encuentra el servidor. Este cuarto es donde también está el equipo que dirige la transmisión de video en vivo desde la embarcación a los satélites y a la audiencia alrededor de todo el mundo. Los microcontroladores son utilizados en la industria de vehículos submarinos para recopilar y procesar imágenes del fondo marino, además de otros datos ambientales.

Durante una de las expediciones del año 2016, científicos construyeron su propio sensor de temperatura utilizando un microcontrolador y un envase de miel reciclado.





Adentrándonos más

Usa la siguiente página web <https://www.rinconingenieril.es/usar-una-protoboard/> ** para aprender más sobre las placas de pruebas (también conocidas como *breadboard* y *protoboard*). Luego, contesta las siguientes preguntas:

- En inglés, las placas de pruebas son conocidas como *breadboards* o tablas de pan. ¿Por qué los llaman de esa manera?
- Menciona dos beneficios de utilizar la placa de pruebas:
- ¿Cuál es otro nombre para las líneas conductoras y para qué se utilizan?
- ¿Cuál es el propósito de tener las líneas de alimentación de corriente que corren verticalmente a lo largo de los dos lados de la placa de pruebas?

Piensa, pare, comparte

Dibuja un diagrama mostrando la conexión entre los siguientes términos: **microcontrolador**, **la placa de prueba**, **el sistema de procesamiento**, **entrada** y **salida**. Comparte tu diagrama con un compañero o compañera de clase.



Actividad de calentamiento

Mira el siguiente bosquejo (listado en el orden correcto). Trata de parear cada línea de código con su función en el banco de respuestas:

```
1 ____  int LED_Builtin = 13;  
2 ____  void setup( ) {  
3 ____  pinMode(LED_Builtin, OUTPUT);  
        }  
4 ____  void loop( ) {  
5 ____  digitalWrite(LED_Builtin,HIGH);  
6 ____  delay(1000);  
7 ____  digitalWrite(LED_Builtin, LOW);  
8 ____  delay(1000);
```

Banco de respuestas

- A. Enciende la bombilla LED
- B. Apaga la bombilla LED
- C. Declara la bombilla LED como un pin de salida
- D. Asigna un nombre a la bombilla LED conectada al pin 13 en el Arduino que será utilizado a lo largo del bosquejo
- E. Designa cuáles funciones serán efectuadas en la parte organizativa del bosquejo
- F. Le dice a la bombilla LED que se mantenga encendida por un segundo
- G. Designa cuáles funciones serán repetidas ("looped")
- H. Le dice a la bombilla LED que se mantenga apagada por 1 segundo

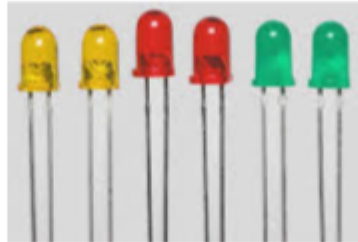
Materiales

Cada grupo necesitará lo siguiente:

- 1 computadora (que no sea una Google Chromebook)



Un puñado de cables 20AWG



Un surtido de bombillas LED
(diodo emisor de luz)



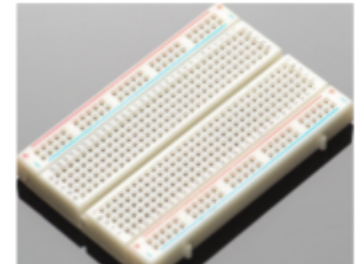
Arduino Uno/Genuino



cable mini USB



Resistores 10 Ohm



placa de pruebas

Procedimiento

Parte 1: Configuración del Arduino

1. Descarga Arduino (*software*) de www.arduino.cc/download.
2. Instala el FTDI Driver.
3. Sigue las direcciones en pantalla para completar la instalación.

Parte 2: Conectando el *hardware*

1. Haz doble clic con el cursor sobre el ícono de Arduino para abrir el programa donde está grabado.
2. Usa el mini cable USB para conectar el Arduino a tu computadora. Una vez esté conectado, la bombilla LED verde debe encenderse y la otra bombilla LED debe comenzar a parpadear.
3. En el menú, ve a *Tools* (herramientas) y selecciona *Board* (tablero). Luego, escoge *Arduino Uno*.
4. Bajo *Tools*, selecciona el puerto serie (cualquiera que esté rotulado Arduino/Genuino Uno).
Si no estás seguro(a) del puerto de tu dispositivo, desconecta tu Arduino y verifica el menú. Luego, vuelve a conectarlo y verifica el menú nuevamente. En las computadoras portátiles marca Windows, el puerto normalmente está rotulado como "COM3" o números más altos. En OSX, aparece bajo algo similar a "/dev/cu.usbmodem1461". El proceso de instalación varía dependiendo del sistema operativo. En Windows, corre el instalador, que también instalará los *drivers* necesarios. En OSX, arrastra la aplicación de Arduino hacia la carpeta de aplicaciones. No necesitarás instalar ningún

driver. Para problemas técnicos en Windows puedes referirte a <https://www.arduino.cc/en/Guide/ArduinoUno>*. Para problemas técnicos en Mac <https://learn.sparkfun.com/tutorials/how-to-install-ftdi-drivers/mac>*.

Parte 3: Práctica con código

1. Desde la aplicación de Arduino, carga el bosquejo de ejemplo seleccionando *File* ➔ *Examples* ➔ *01.Basics* ➔ *Blink* (*Archivo* ➔ *Ejemplos* ➔ *Básicos 01* ➔ *Parpadeo*).
2. Examina el código en el ejemplo titulado *Blink* (parpadeo). Observa que hay notas luego de los //. Este texto incluye comentarios y no será considerado por el programa ni afectará su tamaño. Añadir comentarios al código es una buena práctica para tener referencias y solucionar problemas futuros.
3. Selecciona el botón de *Upload* (mostrado a continuación) para transferir el bosquejo de *Blink* al Arduino. Las bombillas LED rotuladas RX y TX van a parpadear y el programa mostrará el mensaje *Done uploading* (subida finalizada) en la parte inferior de la ventana. Algunos segundos después la bombilla LED en el tablero comenzará a parpadear de acuerdo al patrón del bosquejo.



4. ¡Felicidades! Acabas de enviar tu primer código a un Arduino. Aplicando estas mismas destrezas, puedes crear una gran variedad de proyectos electrónicos.

¡Verificación de destrezas!

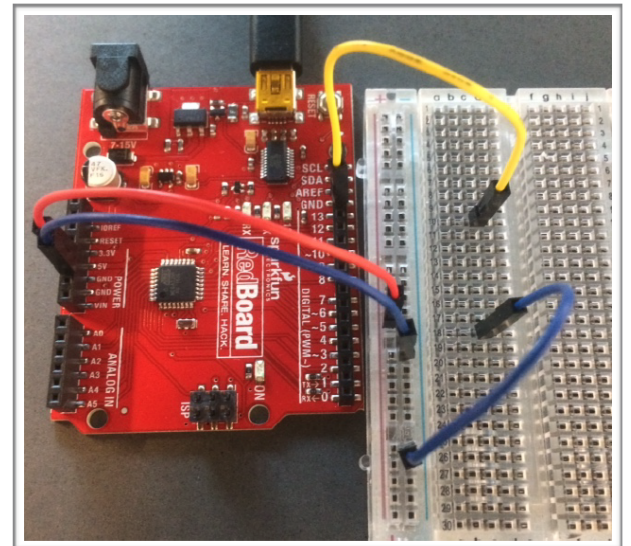
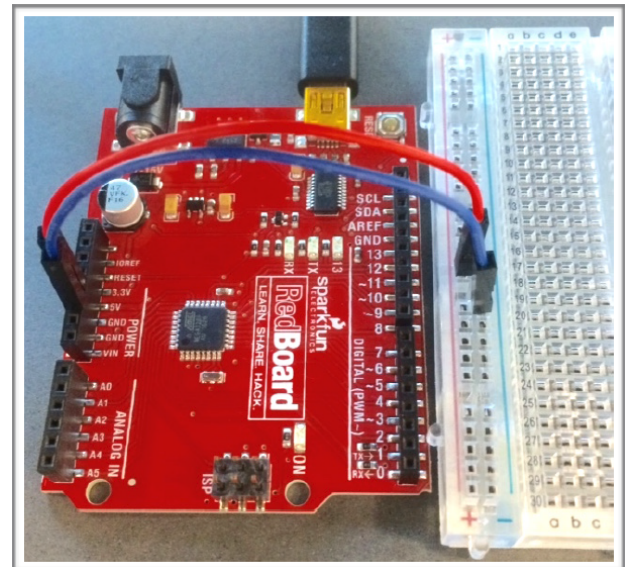
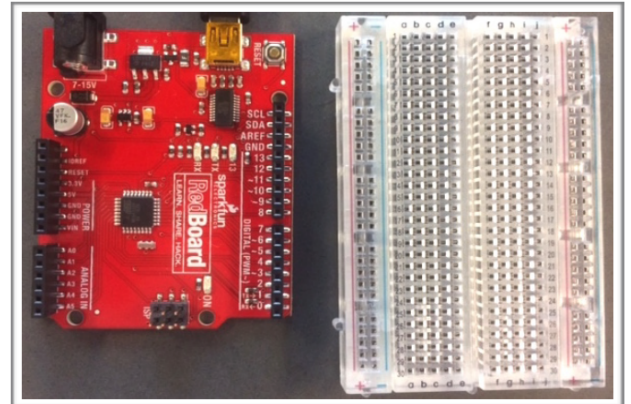
Manipula el código en el ejemplo de *Blink* para hacer que la bombilla LED parpadee a mayor y menor velocidad. Recuerda seleccionar el botón de *upload* (subir) para transferir el nuevo código al tablero Arduino. ¿Completaste la tarea? Si pudiste hacerlo, anota aquí lo que hiciste para cambiar el ritmo de parpadeo:

¡FELICIDADES!

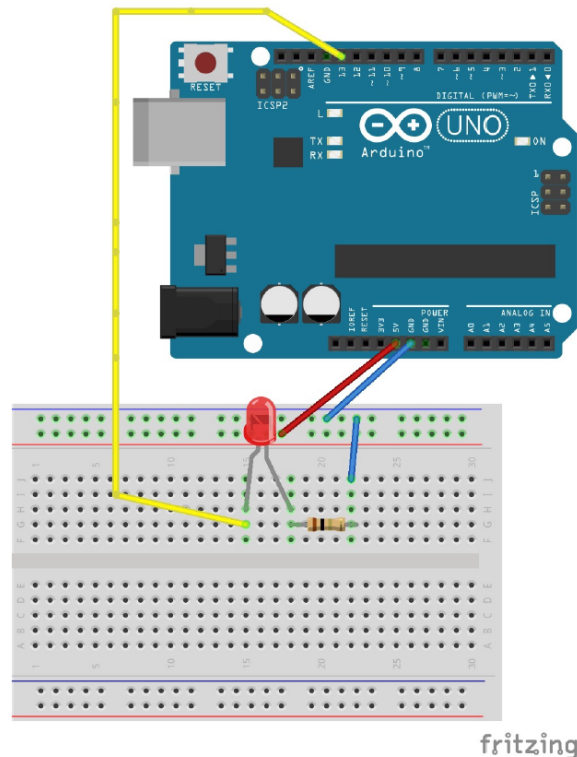
Has completado los primeros pasos en el uso y programación de Arduino. En los próximos pasos aprenderás cómo conectar un circuito compuesto de una bombilla LED y una placa de pruebas.

Parte 4: Circuito de LED y placa de pruebas

1. Desconecta el cable USB antes de conectar tu Arduino a la placa de pruebas.
 2. Coloca el Arduino Uno y la placa de pruebas uno al lado del otro, como se muestra en la imagen.
 3. Conecta un extremo del cable rojo al pin marcado 5V en el Arduino. Conecta el otro extremo a la placa de pruebas, en la columna donde se encuentra la línea roja (línea de alimentación).
 4. Conecta el cable azul al pin marcado GND. (Nota: hay 3 pines GND que hacen contacto a tierra. Todos están conectados al mismo chip de contacto a tierra, así que no importa cuál selecciones.) Conecta el otro extremo del cable azul a la línea de alimentación azul en la placa de pruebas.
- ***Nota importante: todos los cables son iguales. Los distintos colores se usan para poder diferenciar conexiones y circuitos fácilmente al leer esquemas, diagramas o instrucciones escritas.**
5. Conecta un cable amarillo al pin 13 en el Arduino. Este lo usarás para completar un circuito con la bombilla LED (se conecta al pin de contacto a tierra). Conecta el otro extremo a cualquier fila horizontal dentro de la placa de pruebas que esté cerca del Arduino.
 6. Conecta un cable azul de la línea de alimentación azul a cualquier fila de la placa de pruebas (columnas a-e).



7. Encuentra un resistor 1K entre tus materiales (tienen bandas marrones, negras y doradas). Conecta uno de los lados del resistor a la misma fila donde se encuentre uno de los cables azules. Conecta el otro extremo a la misma columna, al lado del cable amarillo. Nota: Un resistor debe seguir el camino del circuito, la parte más corta de la bombilla LED debe estar en la misma fila que un extremo del resistor y el otro extremo tiene que estar en la misma fila que el cable a tierra.
8. Agarra una bombilla LED roja y conecta el alambre más largo (ánodo positivo) a la misma fila donde se encuentra el cable amarillo. Conecta el alambre más corto (cátodo negativo) en la misma fila, al lado del resistor.



En este circuito, una carga positiva fluye desde el pin de Arduino, a través de la bombilla LED, pasa al resistor, hasta hacer contacto a tierra. El resistor ayuda a limitar la corriente eléctrica que fluye hacia la bombilla LED.

9. Conecta tu tablero Arduino a la computadora utilizando el cable USB. La bombilla LED debe comenzar a parpadear. Sube el documento de *Blink* del menú '*Arduino Examples* → *Basics*' (Ejemplos Arduino ➡ Básicos) y edita el bosquejo para que la bombilla parpadee más rápido y más despacio.
10. ¿Listo para otro reto? Trata de conectar múltiples LED siguiendo estas instrucciones (<http://nautl.us/2WOaExN>).
11. Cuando estés listo para continuar, desconecta el cable USB y remueve todas las bombillas LED menos la primera y el cable amarillo del Arduino.
12. Mueve el cable amarillo al pin 9.
13. Completa las dos aplicaciones en la siguiente página.



Aplicaciones estudiantiles

Parte 1: Corrige el código: Examina el siguiente bosquejo y corrige cualquier error. Si lo haces correctamente, una vez se cargue al Arduino, el código debe hacer que la luz de la bombilla LED se desvanezca. Escribe tu código corregido en un bosquejo en blanco para verificar si funciona.

```
int led = 9;           // the PWM pin the LED is attached to
int brightness = 0;    // how bright the LED is
int fadeAmount = 5     // how many points to fade the LED by

// the setup routine runs once when you press reset:
void setup() {
  // declare pin 5 to be an output:

}

// the loop routine runs over and over again forever:
void loop() {
  // set the brightness of pin 5:
  analogWrite( led, brightness);

  // change the brightness for next time through the loop:
  brightness = brightness + fadeAmount;

  // reverse the direction of the fading at the ends of the fade:
  if (brightness <= 255 || brightness >= 0) {
    fadeAmount = -fadeAmount;
  }
  // wait for 30 milliseconds to see the dimming effect
  delay(10);
}
```

Parte 2: Escribe un código SOS: Crea tu propio código para hacer que la bombilla LED parpadee en un patrón SOS (3 pulsos cortos, seguido de 3 pulsos largos, seguido de 3 pulsos cortos). Tome una foto de tu pantalla o anote tu código a continuación.

Pregunta de discusión: Compara el código de tu grupo con el de otros grupos. ¿Pueden distintos códigos producir los mismos resultados?

Rúbrica de tareas y proyectos STEM

Objetivo	Criterios			
	4 Sobresaliente	3 Muy bueno	2 Aceptable	1 En desarrollo
Conocimiento y comprensión	Constantemente contesta todas las preguntas de forma correcta y detallada. Utiliza vocabulario relevante regularmente y logra explicar las relaciones dentro del contenido utilizando ejemplos. Puede aplicar el contenido a otros temas o a la vida real.	Contesta la mayoría de las preguntas correctamente. Emplea una cantidad adecuada de vocabulario relevante. Puede explicar las relaciones dentro del contenido y es capaz de aplicarlas a otros temas o a la vida real.	Contesta algunas preguntas correctamente. Utiliza algún vocabulario relevante. No elabora sobre las relaciones que existen dentro del contenido, ni hace conexiones entre el contenido y la vida real.	Contesta pocas preguntas correctamente. Presenta un uso inconsistente del vocabulario. No elabora sobre las relaciones dentro del contenido, ni hace conexiones entre el contenido y la vida real.
Organización del contenido, metodología y análisis	Es capaz de organizar ideas complejas, conceptos e información para hacer conexiones y distinciones importantes. Esto incluye procedimientos profundos y detallados, además de tablas de datos, gráficas, diagramas o algún tipo de análisis.	Es capaz de organizar ideas, conceptos e información para hacer conexiones y distinciones. Esto puede incluir procedimientos profundos y detallados; además de tablas de datos, gráficas, diagramas o algún tipo de análisis.	Intenta organizar ideas, conceptos e información para hacer conexiones y distinciones. Es capaz de presentar procedimientos básicos, tablas de datos, gráficas, diagramas o algún tipo de análisis.	Presenta dificultad para organizar ideas, conceptos e información para hacer conexiones y distinciones. No es capaz de presentar procedimientos básicos, tablas de datos, gráficas, diagramas o algún tipo de análisis.
Aprendizaje autodirigido	Se involucra activamente en el proceso de aprendizaje; contribuye a la discusión en clase de manera consistente y hace preguntas para aclarar sus dudas. Busca recursos	Se involucra en el proceso de aprendizaje. Contribuye a la discusión en clase y hace preguntas para aclarar sus dudas. Aboga por sus necesidades de aprendizaje.	No se involucra consistentemente en el proceso. Contribuye a las discusiones en clase y hace preguntas ocasionalmente. No aboga consistentemente por sus	Se involucra muy poco en el proceso de aprendizaje. Raramente contribuye a las discusiones en clase o hace preguntas. Raramente aboga por sus

	adicionales de forma independiente y los comparte con la clase o maestra(o). Aboga por sus necesidades de aprendizaje.		necesidades de aprendizaje.	necesidades de aprendizaje.
Herramientas tecnológicas	Siempre usa los recursos digitales apropiadamente para la tarea. Está dispuesto(a) a aprender y a utilizar la tecnología para incluir gráficas, tablas, ilustraciones, etc. que amplifiquen el mensaje.	Usa los recursos digitales apropiadamente para la tarea. Está dispuesto(a) a aprender y utilizar la tecnología para incluir gráficas, tablas, ilustraciones, etc. que amplifiquen el mensaje.	Algunas veces usa recursos digitales apropiadamente para la tarea. Usa inconsistentemente la tecnología para incluir gráficas, tablas, ilustraciones, etc. que amplifiquen el mensaje.	Raramente usa los recursos digitales apropiadamente para la tarea. No usa la tecnología para incluir gráficos, tablas, ilustraciones, etc. que amplifiquen el mensaje.
Destrezas colaborativas	Trabaja de manera efectiva y respetuosa con un grupo diverso de estudiantes. Busca maneras de entender, conversando con otros y pregunta cómo puede ayudar. Escucha cuando otros hablan e incorpora o expande las ideas de otros.	Trabaja de manera efectiva y respetuosa con un grupo diverso de estudiantes. Busca maneras de entender conversando con otros y pregunta cómo puede ayudar. Escucha cuando otros hablan.	En ocasiones trabaja de manera efectiva y respetuosa con un grupo diverso de estudiantes. Ocasionalmente busca maneras de entender conversando con otros y pregunta cómo puede ayudar. Escucha cuando otros hablan.	Presenta dificultad para trabajar de manera efectiva y respetuosa con un grupo diverso de estudiantes. Raramente busca maneras de entender conversando con otros y no pregunta cómo puede ayudar. Habla por encima de otros o no escucha cuando otros hablan.
Puntuación total:	Comentarios:			