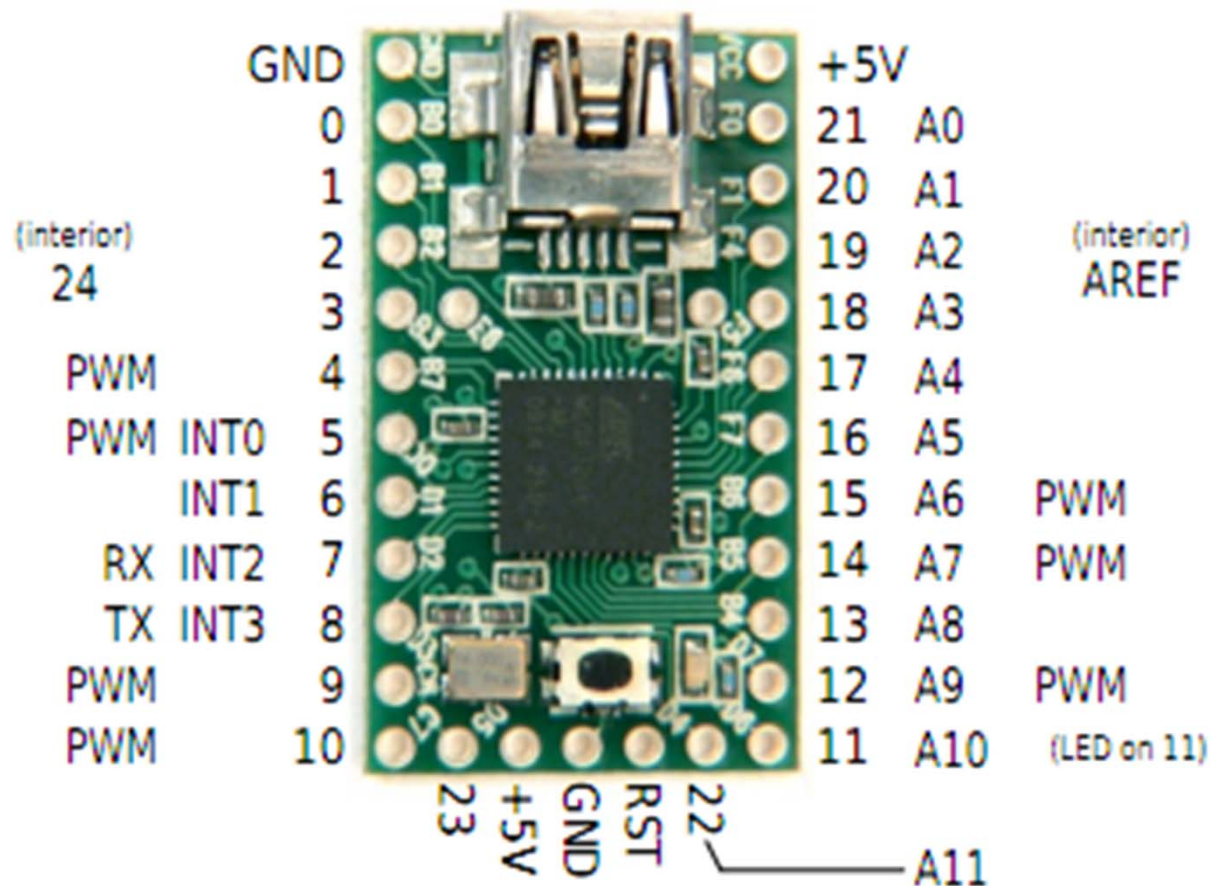


Introduction to Programming with the Arduino 2013

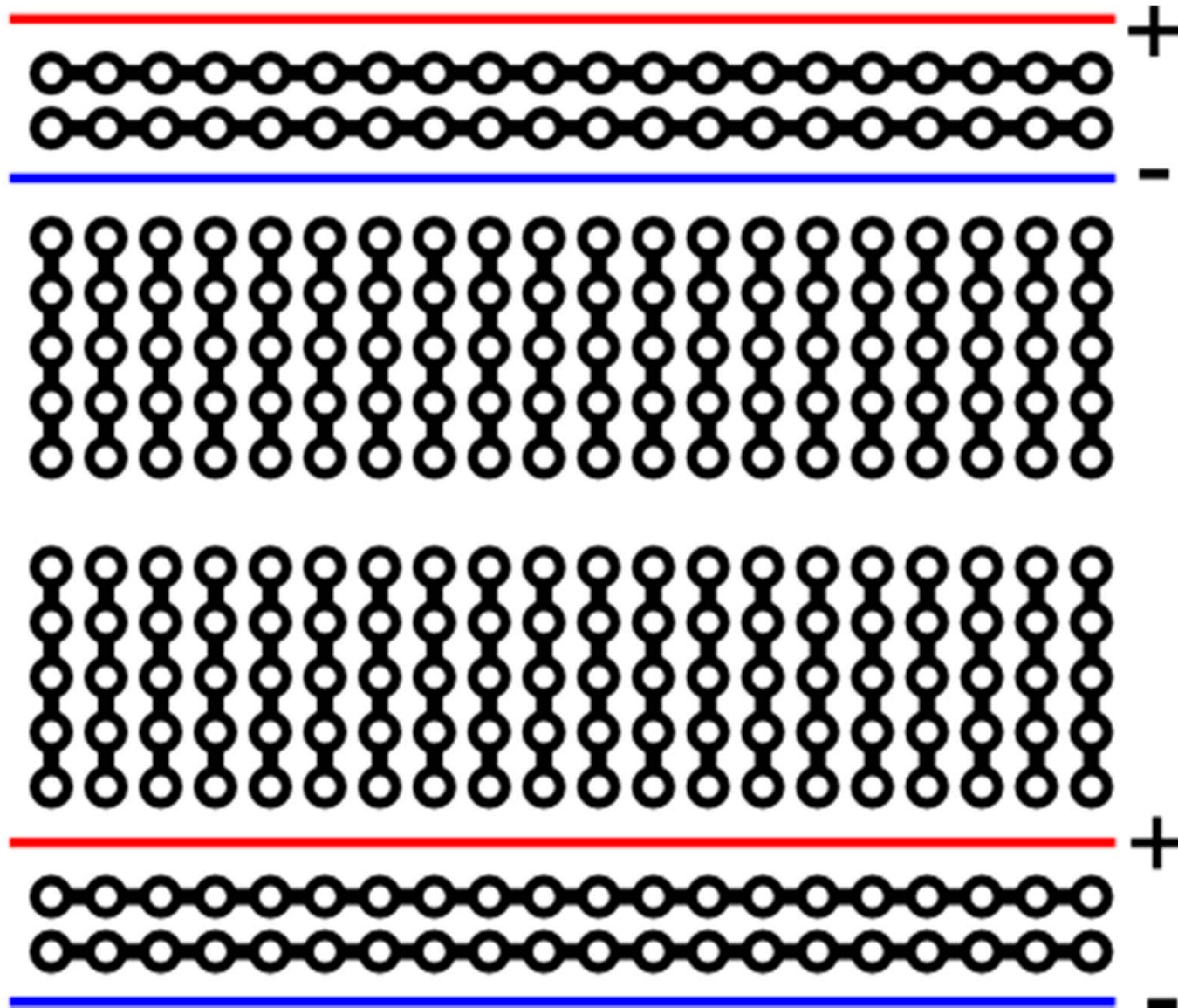
What Arduino are we using today?

Teensy 2.0 Arduino variation

Pin Assignments Using Arduino Software



Breadboards



Exercise 1: Blink the LED

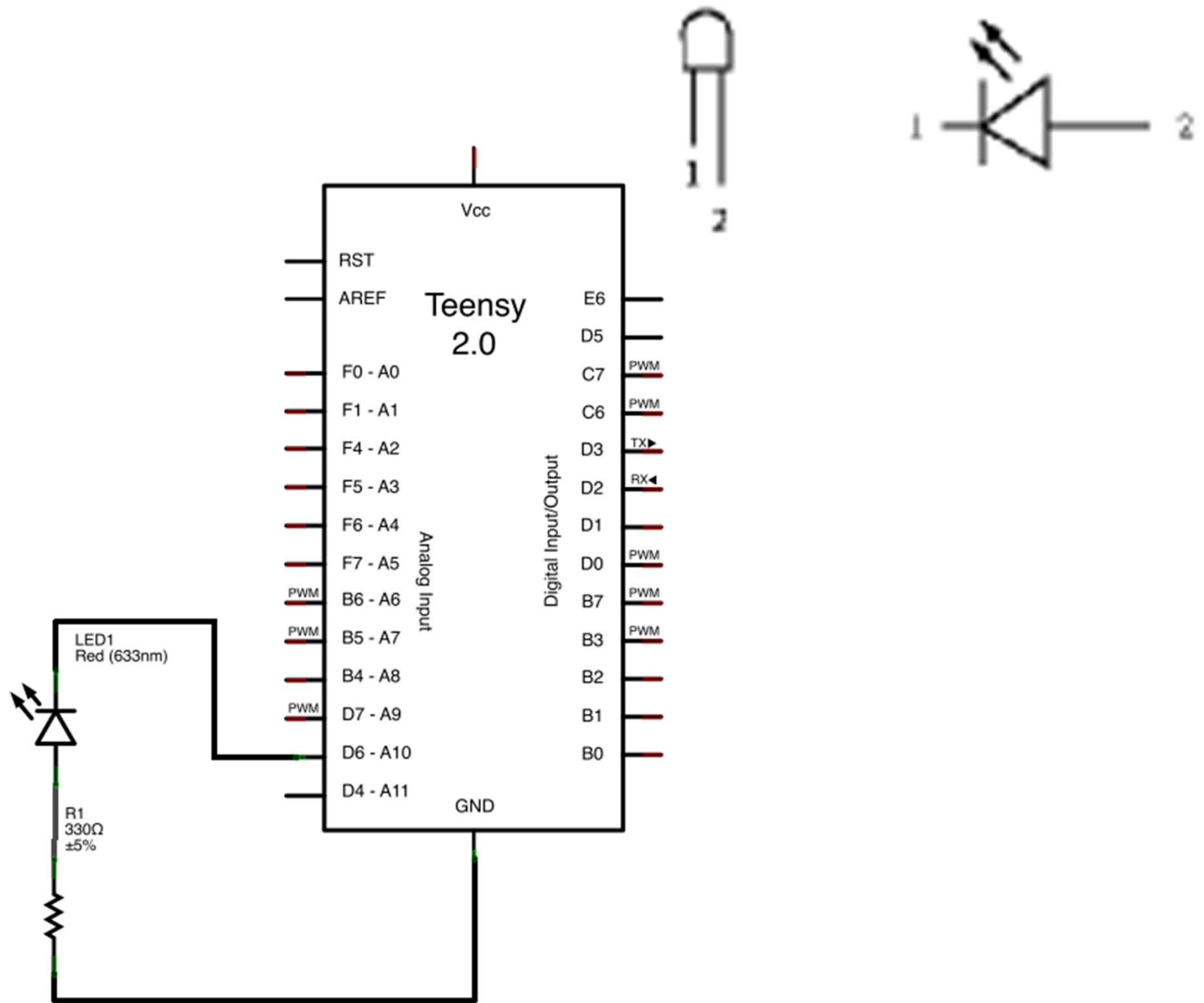
Blink 'Hello World' of microcontrollers.

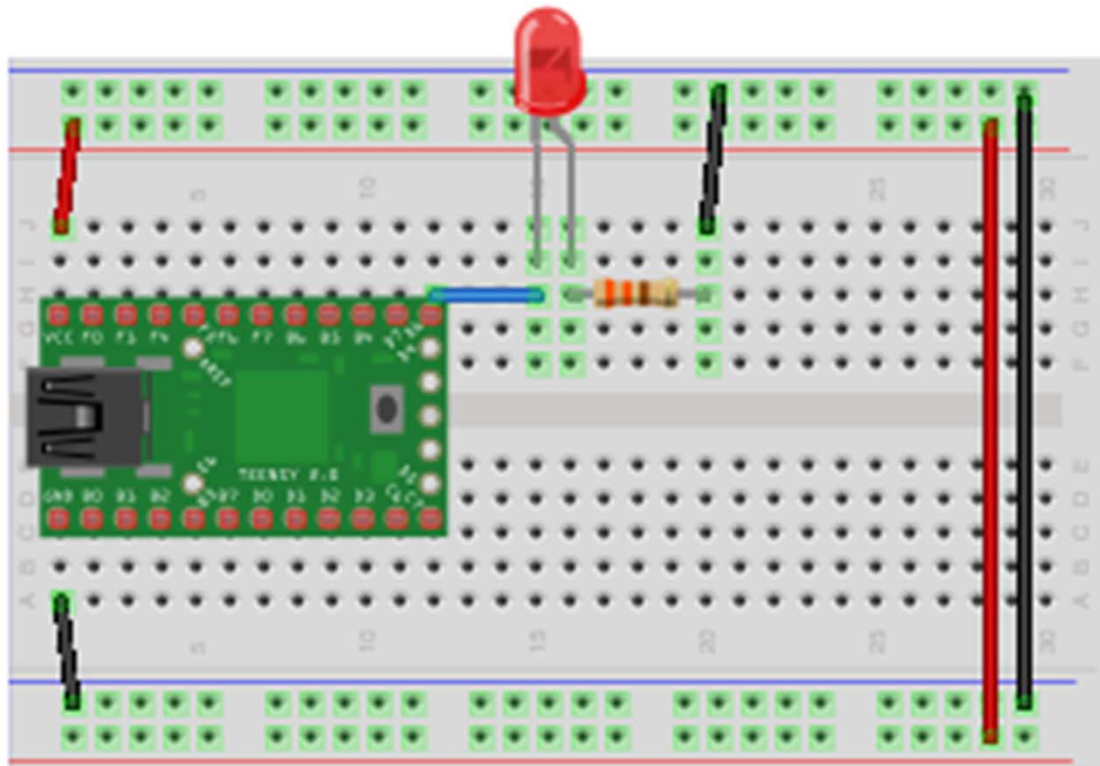
It demonstrates digital output.

It also tells you that your device is working.

The Teensy has an on board LED on PIN 11.

We will configure the Arduino to blink an external LED on pin 11. So both blink at the same time.





Made with  Fritzing.org

LED positive (long) leg connected to pin 11

Blink an LED

```
void setup()  
{ //runs once  
  
}
```

```
void loop()  
{ //executed repeatedly  
  
}
```

Blink an LED

```
int ledPin = 11; //led on pin 11  
                //defined in one place for our convenience
```

```
void setup()  
{  
  
}
```

```
void loop()  
{  
  
}
```


Blink an LED

```
int ledPin = 11; //led on pin 11
```

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

```
void loop()  
{  
  
}
```

Blink an LED

```
int ledPin = 11; //led on pin 11
```

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

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


Blink an LED

```
int ledPin = 11; //led on pin 11
```

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

```
void loop()  
{  
    digitalWrite(ledPin, HIGH);    digitalWrite(ledPin, LOW);  
}
```

Blink an LED

 `int ledPin = 11; //led on pin 11`

Declaration of an integer

 Single line comment

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

Declaration of a function with VOID return type

 Calling a predefined function

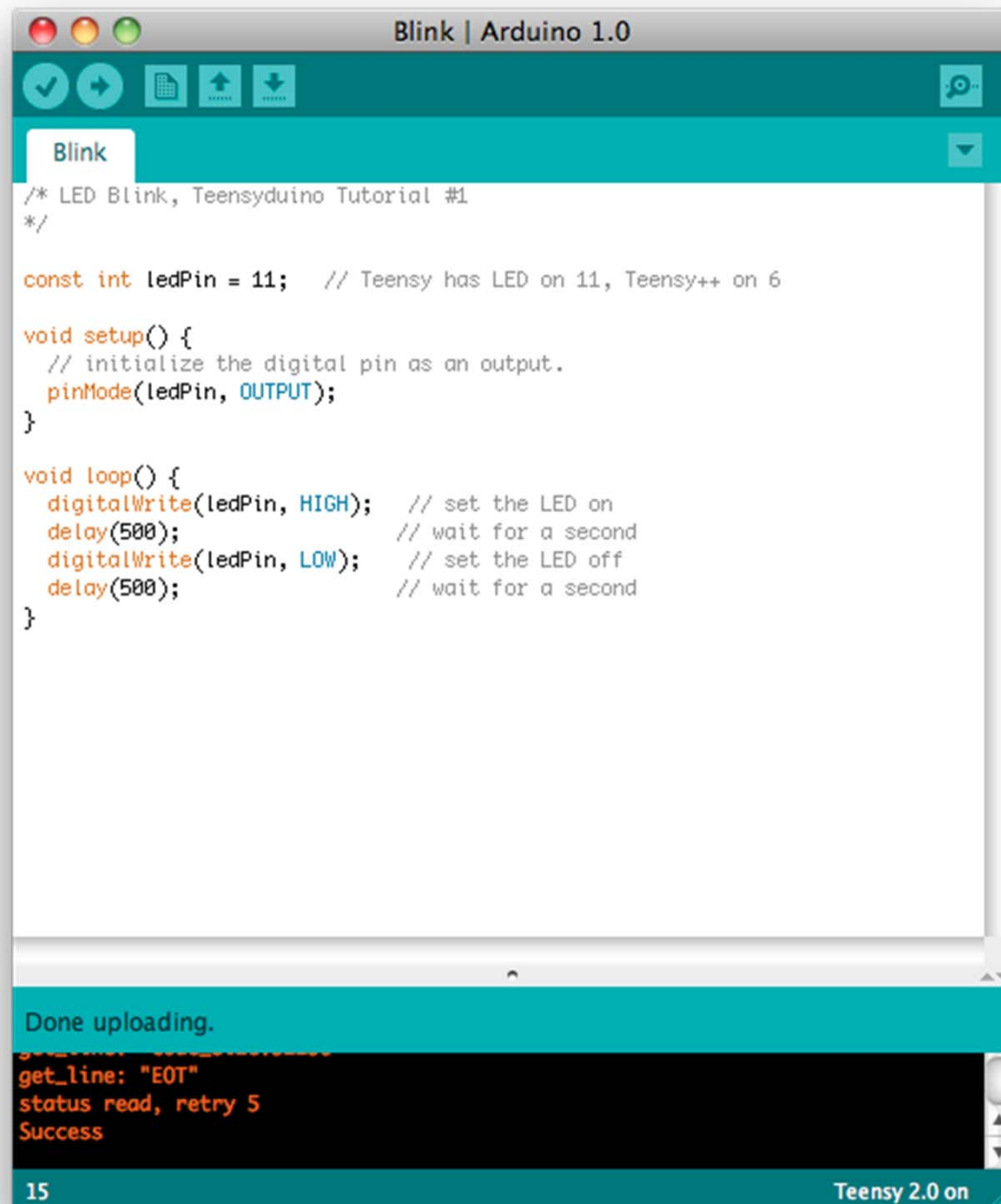
`void loop()`
`{`
 `digitalWrite(ledPin, HIGH);`
 `digitalWrite(ledPin, LOW);`
`}`

Function arguments (constants)

`delay(1000);`
`delay(1000);`

 Reserved words

Blink an LED



```
Blink | Arduino 1.0

/* LED Blink, Teensyduino Tutorial #1
*/

const int ledPin = 11;  // Teensy has LED on 11, Teensy++ on 6

void setup() {
  // initialize the digital pin as an output.
  pinMode(ledPin, OUTPUT);
}

void loop() {
  digitalWrite(ledPin, HIGH);  // set the LED on
  delay(500);                  // wait for a second
  digitalWrite(ledPin, LOW);   // set the LED off
  delay(500);                  // wait for a second
}

Done uploading.
get_line: "EOT"
status read, retry 5
Success

15 Teensy 2.0 on
```

Blink an LED

Verify = Compile and debug



The screenshot shows the Arduino IDE window titled "Blink | Arduino 1.0". The interface includes a toolbar with icons for checking, uploading, and downloading, along with a "Verify" button. Below the toolbar, a tab labeled "Blink" is active. The main text area contains the following C++ code:

```
/* LED Blink, Teensyduino Tutorial #1
*/

const int ledPin = 11;  // Teensy has LED on 11, Teensy++ on 6

void setup() {
  // initialize the digital pin as an output.
  pinMode(ledPin, OUTPUT);
}

void loop() {
  digitalWrite(ledPin, HIGH);  // set the LED on
  delay(500);                  // wait for a second
  digitalWrite(ledPin, LOW);   // set the LED off
  delay(500);                  // wait for a second
}
```

Blink an LED

Upload = Compile and Upload to the board



The screenshot shows the Arduino IDE window titled "Blink | Arduino 1.0". The interface includes a toolbar with icons for checking, uploading, and downloading, along with the text "Upload Using Programmer". A tab labeled "Blink §" is active. The code editor displays the following C++ code:

```
/* LED Blink, Teensyduino Tutorial #1
*/

const int ledPin = 11;    // Teensy has LED on 11, Teensy++ on 6

void setup() {
  // initialize the digital pin as an output.
  pinMode(ledPin, OUTPUT);
}

void loop() {
  digitalWrite(ledPin, HIGH);  // set the LED on
```

Other uses for digital output

- Switching other devices on/off
 - Relays - Can open/close a switch in a separate circuit for controlling high-power loads and avoiding electrical noise
 - Transistors - Can be used to amplify a signal to much higher power than the microcontroller alone
- Communication
 - Software Serial

Analog out with PWM

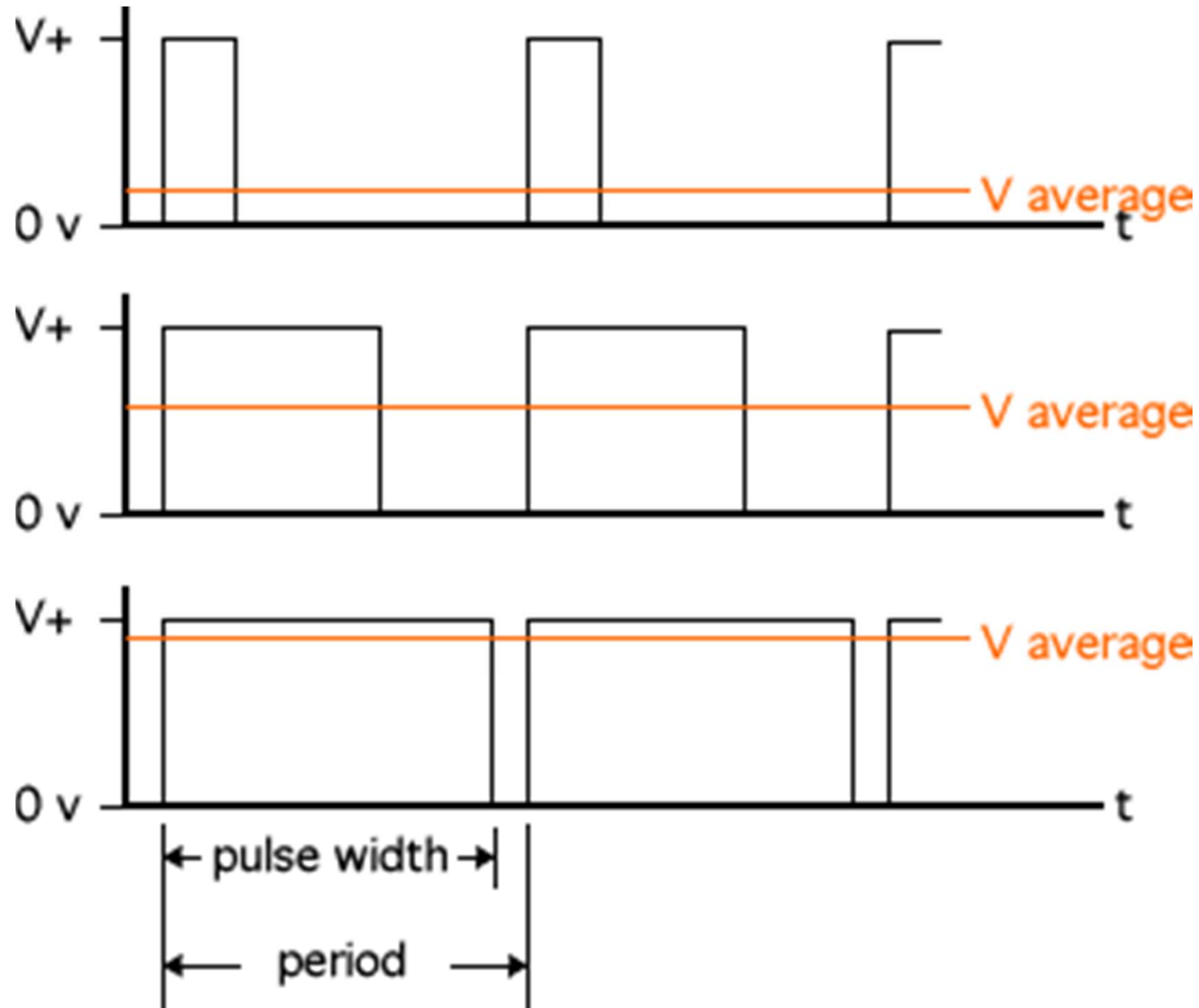
PWM - Stands for Pulse Width Modulation

So far we know how to turn an output OFF or ON (LOW and HIGH)

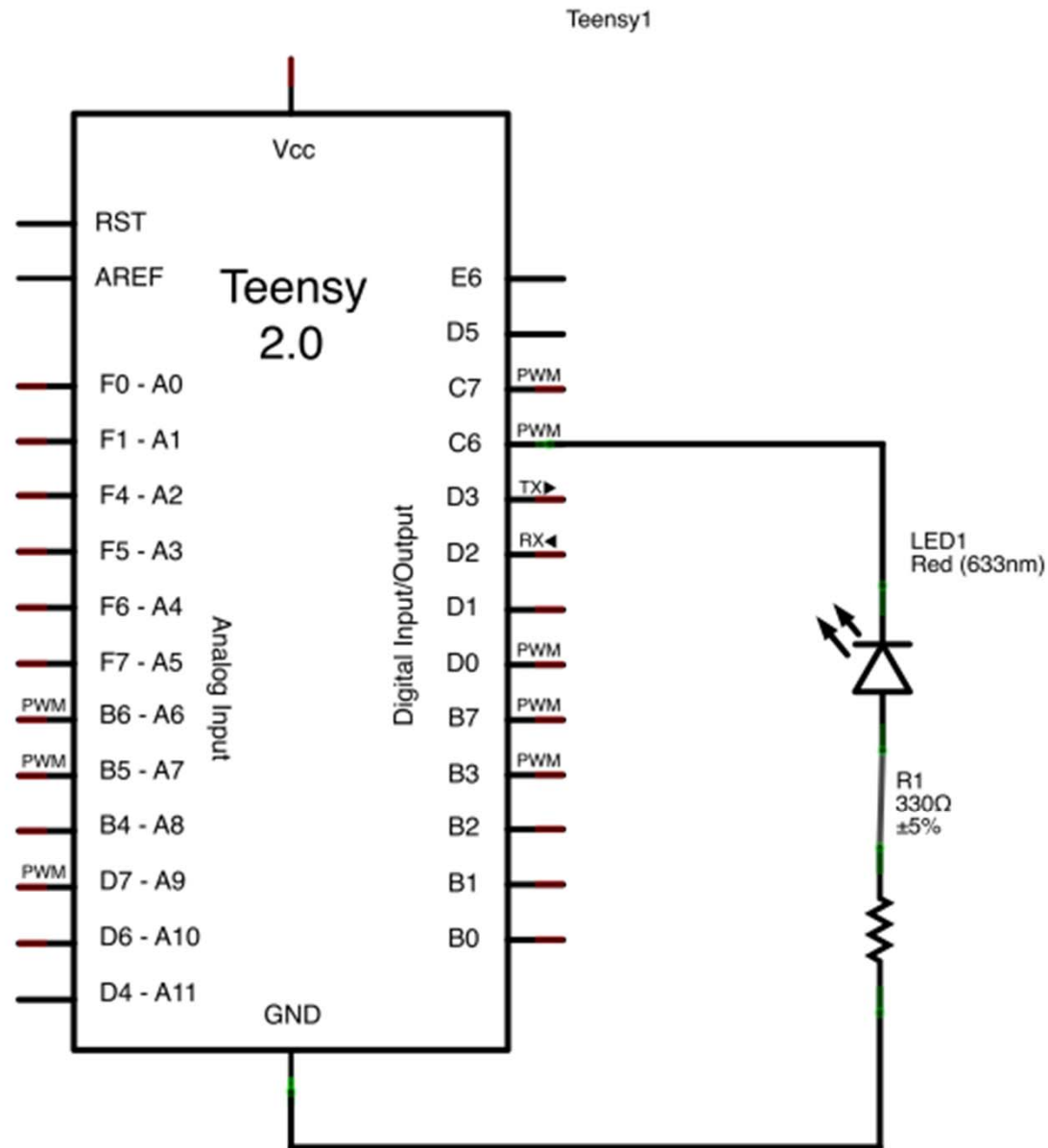
What if we want to change the speed of a motor, or the intensity of a light?

PWM output is a signal where the output switches very rapidly between fully on and fully off. For a lower value, the output is on a smaller percentage of the time, for a higher value, the output is on for a larger percentage of the time.

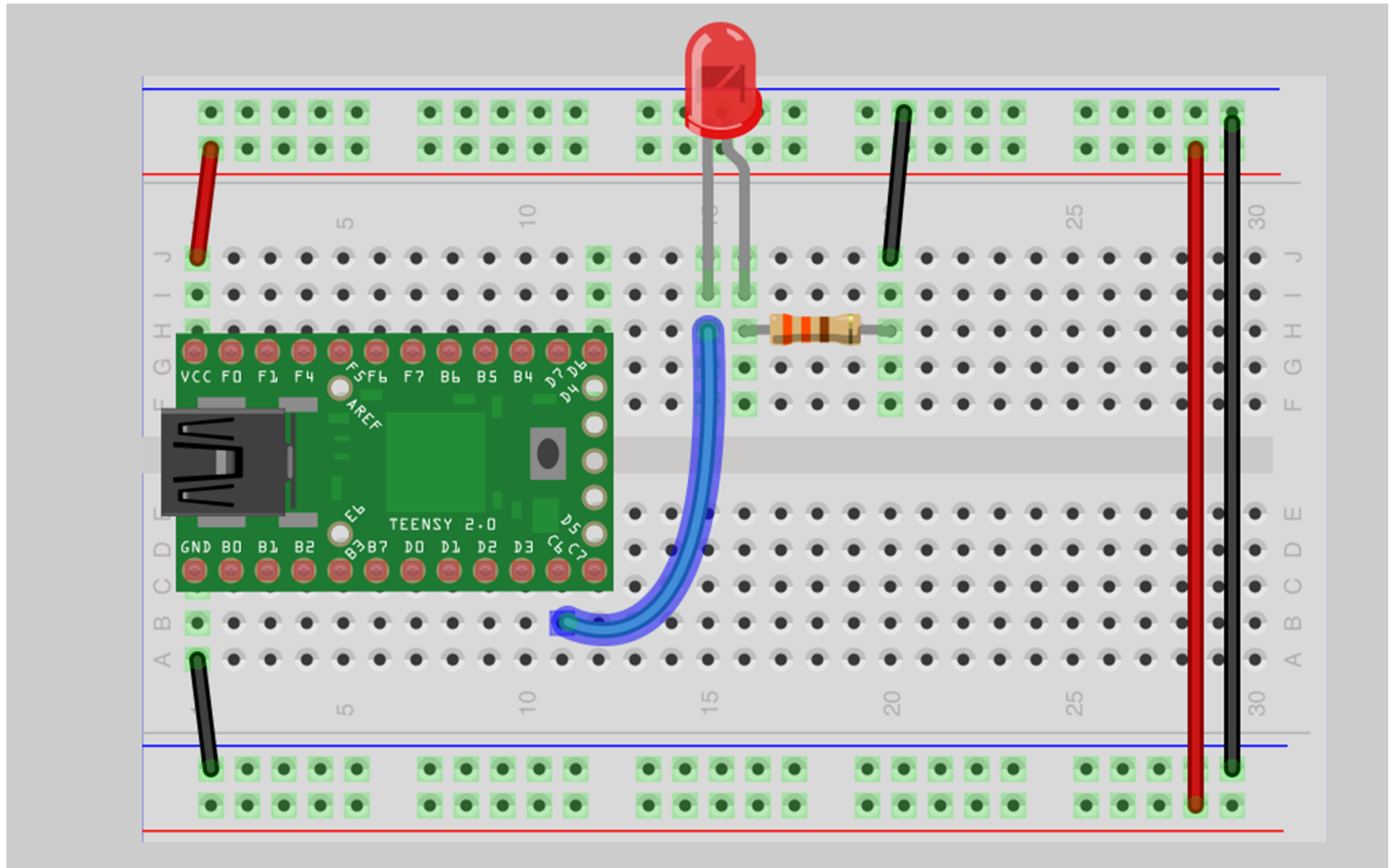
PWM



Exercise 3- Fade an LED on/off



Exercise 3- Fade an LED on/off



Two ways to fade an LED

First way is to use void loop()

Increment and decrement the values every time through the loop. Same setup will be used in both examples:

```
/*=====*/
```

```
const int ledPin = 9;  
int brightness = 0;    // how bright the LED is  
int fadeAmount = 5;    // how many points to fade the LED by
```

```
void setup() {  
    // declare pin 9 to be an output:  
    pinMode(ledPin, OUTPUT);  
}
```

Fade inside loop()

```
const int ledPin = 9;
int brightness = 0;  // how bright the LED is
int fadeAmount = 5;  // how many points to fade the LED by
```

```
void setup() {
  // declare pin 9 to be an output:
  pinMode(ledPin, OUTPUT);
}
```

```
void loop() {
  // set the brightness of pin 9:
  analogWrite(ledPin, 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);
}
```

Fade an LED - using a for loop

- Use the same setup code as before
- Same circuit
- You can use save as and delete the code in loop and replace with the new code

The for loop

```
for(int t=0; t<=100 ; t+=1)
{
    //Do something 100 times
}
```


Fade an LED - the Code

```
....  
void loop()  
{  
  for( )  
  {  
    // What goes in the for loop?  
    // sets the value (range from 0 to 255. 8 bit resolution):  
    analogWrite(ledPin, fadeValue);  
  }  
}
```

Fade an LED - the Code

```
....  
void loop()  
{  
  for(int fadeValue = 0 ; fadeValue <= 255; fadeValue +=5)  
  {  
    // sets the value (range from 0 to 255):  
    analogWrite(ledPin, fadeValue);  
  }  
}
```

Fade an LED - the Code

....

```
void loop()
{
  for(int fadeValue = 0 ; fadeValue <= 255; fadeValue +=5)
  {
    // sets the value (range from 0 to 255):
    analogWrite(ledPin, fadeValue);
    // wait for 30 milliseconds to see the dimming effect
    delay(30);
  }
}
```

Fade an LED - the Code

```
....  
void loop()  
{  
  for(int fadeValue = 0 ; fadeValue <= 255; fadeValue +=5)  
  {  
    // sets the value (range from 0 to 255):  
    analogWrite(ledPin, fadeValue);  
    // wait for 30 milliseconds to see the dimming effect  
    delay(30);  
  }  
  // duplicate in reverse  
  for(int fadeValue = 255 ; fadeValue >= 0; fadeValue -=5)  
  {  
    // sets the value (range from 0 to 255):  
    analogWrite(ledPin, fadeValue);  
    // wait for 30 milliseconds to see the dimming effect  
    delay(30);  
  }  
}
```

1 circuit -- 1,000 behaviors

With one simple circuit, you can write many behaviors just by changing the software, without having to alter the circuit or add additional components.

Digital vs Analog

What things are digital?

What things are analog?

What things appear to be digital or analog?

How can we communicate exact values?

How about approximate values?

What about minimum values?

What about maximum values?

Serial Output

`Serial.begin(9600);`

Starts the serial connection - call once at start of sketch

`Serial.println(someValue);`

Sends someValue across the serial connection with a newline character.

`var = Serial.read();`

Returns incoming serial data, stores it in var

Exercise: Another Hello World

```
void setup()  
{  
  Serial.begin(9600);  
}
```

```
void loop()  
{  
  Serial.println("Hello World");  
  delay(1000);  
}
```


Analog Input

```
val = analogRead(analogPin);
```

Reads voltage on analog input pins and translates to a numerical value.

Value is based on percentage of reference voltage (in this case 5v)

The reading has a 10 bit resolution (returns a value in the range 0-1023)

Analog Read code

```
int sensorPin = A0;
```

```
int sensorVal = 0;
```

```
void setup()
```

```
{
```

```
  Serial.begin(9600);      // setup serial
```

```
  pinMode(sensorPin, INPUT);
```

```
}
```

```
void loop()
```

```
{
```

```
  sensorVal = analogRead(sensorPin);  // read the input pin
```

```
  Serial.println(sensorVal);          // debug value
```

```
}
```

Blink with out delay:

Let's Rewrite blink with out delay.

What will we need?

- A Fixed time stamp (What units?)
- A reading of the current clock value
- Something to track the current state
- Some conditional expression that checks if the amount of time passed is more than the length of time we want the led to stay on/off

A refrence:

<http://arduino.cc/en/Tutorial/BlinkWithoutDelay>

Blink with out delay

```
const int ledPin = 11;
int ledState = LOW; //LOW is a reserved word
long previousMillis = 0; // Which one is this?
long interval = 1000;

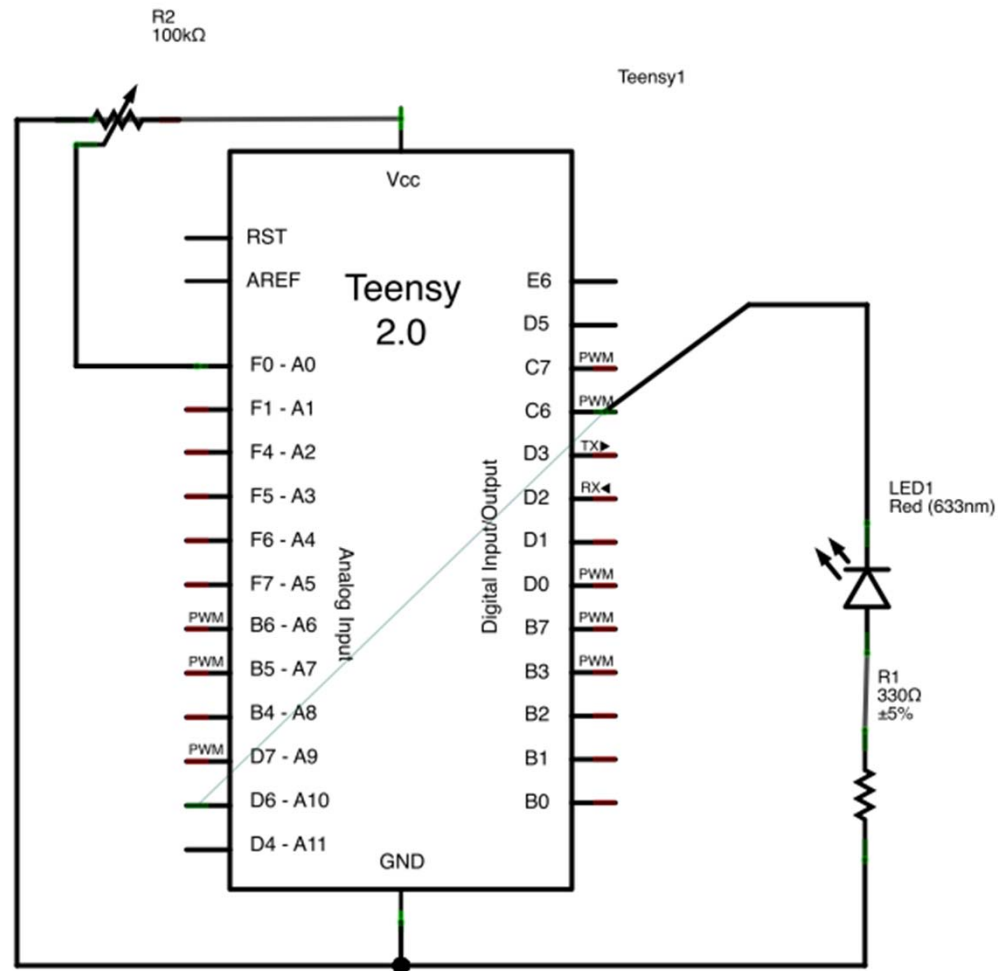
void setup() {
  pinMode(ledPin, OUTPUT);    //setup
}

void loop()
{
  unsigned long currentMillis = millis();

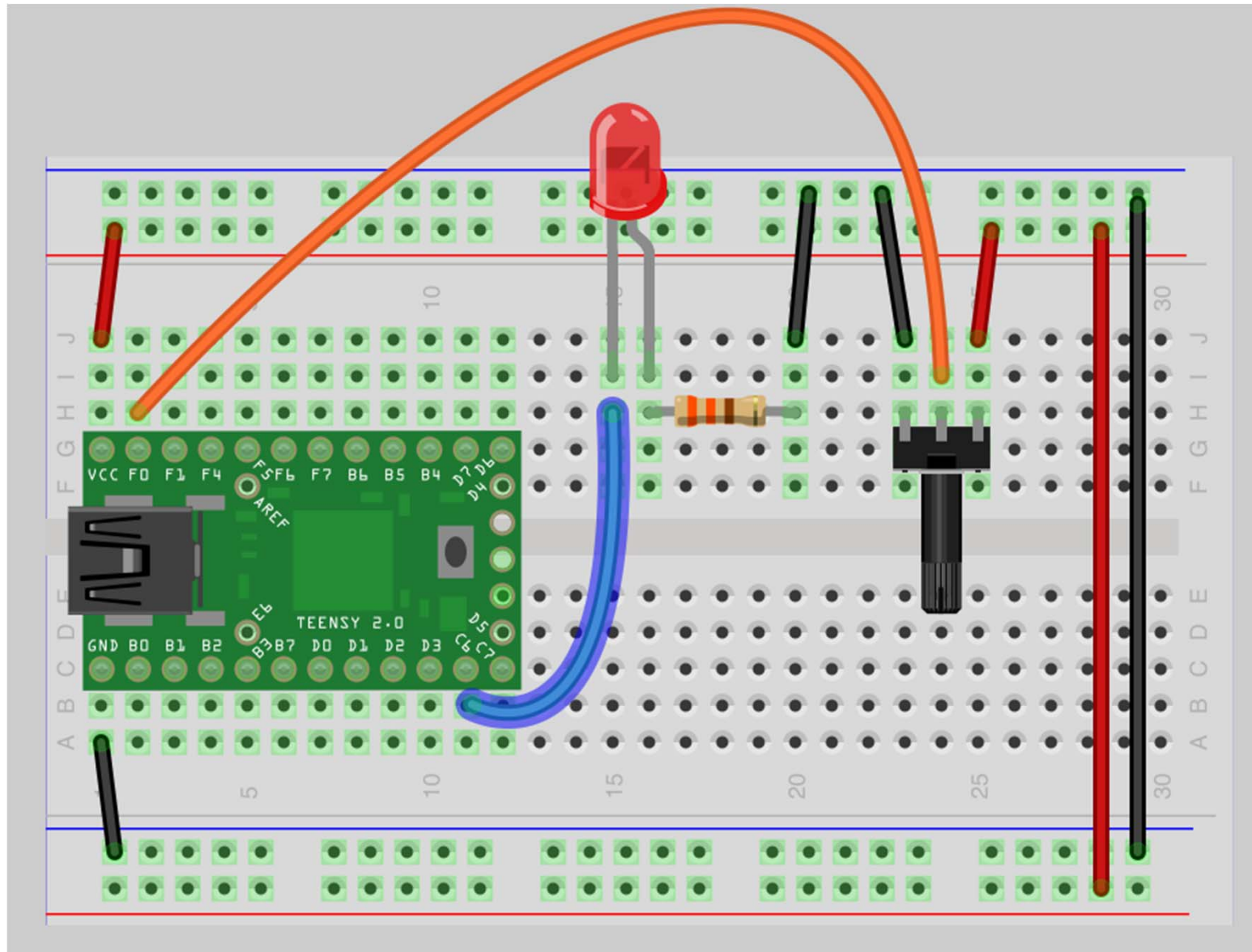
  if(currentMillis - previousMillis > interval) {
    previousMillis = currentMillis; // why is this here?
    if (ledState == LOW)
      ledState = HIGH;
    else
      ledState = LOW;
    digitalWrite(ledPin, ledState);
  }
}
```

Exercise 5 - LED dimmer control

The circuit



Starting Circuit



Dimmer Control code

```
int sensorPin = A0;
int ledPin = 9;
int sensorVal = 0;
void setup()
{
  Serial.begin(9600);      // setup serial
  pinMode(sensorPin, INPUT);
  pinMode(ledPin, OUTPUT);
}
void loop()
{
  sensorVal = analogRead(sensorPin); // read the input pin
  Serial.println(sensorVal);         // debug value
  analogWrite(ledPin, sensorVal/4);}
```