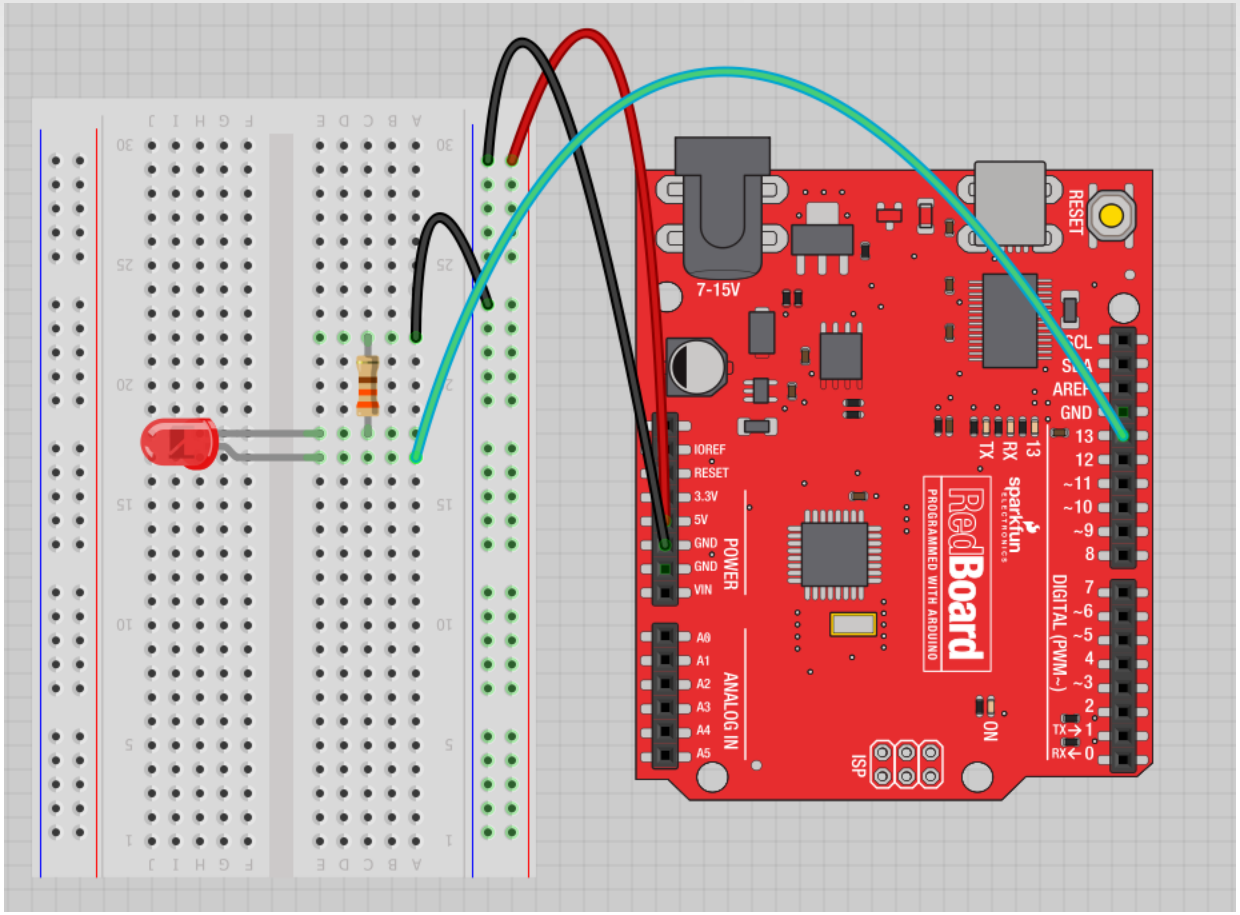
Project #1: Wiring Diagram



Move the green wire from the power bus to pin 13 (or any other Digital I/O pin on the Arduino board.

Image created in Fritzing

A few simple challenges
Let's make LED#13 blink!

Challenge 1a – blink with a 200 ms second interval.

Challenge 1b – blink to mimic a heartbeat

Challenge 1c – find the fastest blink that the human eye can still detect…

1 ms delay? 2 ms delay? 3 ms delay???

Try adding other LEDs

Can you blink two, three, or four LEDs?

(Hint: Each LED will need it's own 330Ω resistor.)

Generate your own morse code flashing

How about → Knight Rider? Disco? Police Light?

Programming Concepts: Variables

ProtosnapProMiniExample2 §

```
// Comments go here
// Written by:  Joesephine Jones
// Date:  April 12, 2013

int sensorValue;
int ledPin;

void setup()
{
  // put your setup code here, to run once:
  int setupVariable;

}

void loop()
{
  // put your main code here, to run repeatedly:
  int loopScopeVariable
}
```

Variable Scope

Global

Function-level

Programming Concepts: Variable Types

Variable Types:

8 bits	16 bits	32 bits
byte char	int unsigned int	long unsigned long float

Fading in and Fading Out (Analog or Digital?)

A few pins on the Arduino allow for us to modify the output to mimic an analog signal.

This is done by a technique called:

Pulse Width Modulation (PWM)

Concepts: Analog vs. Digital

To create an analog signal, the microcontroller uses a technique called PWM. By varying the duty cycle, we can mimic an “average” analog voltage.