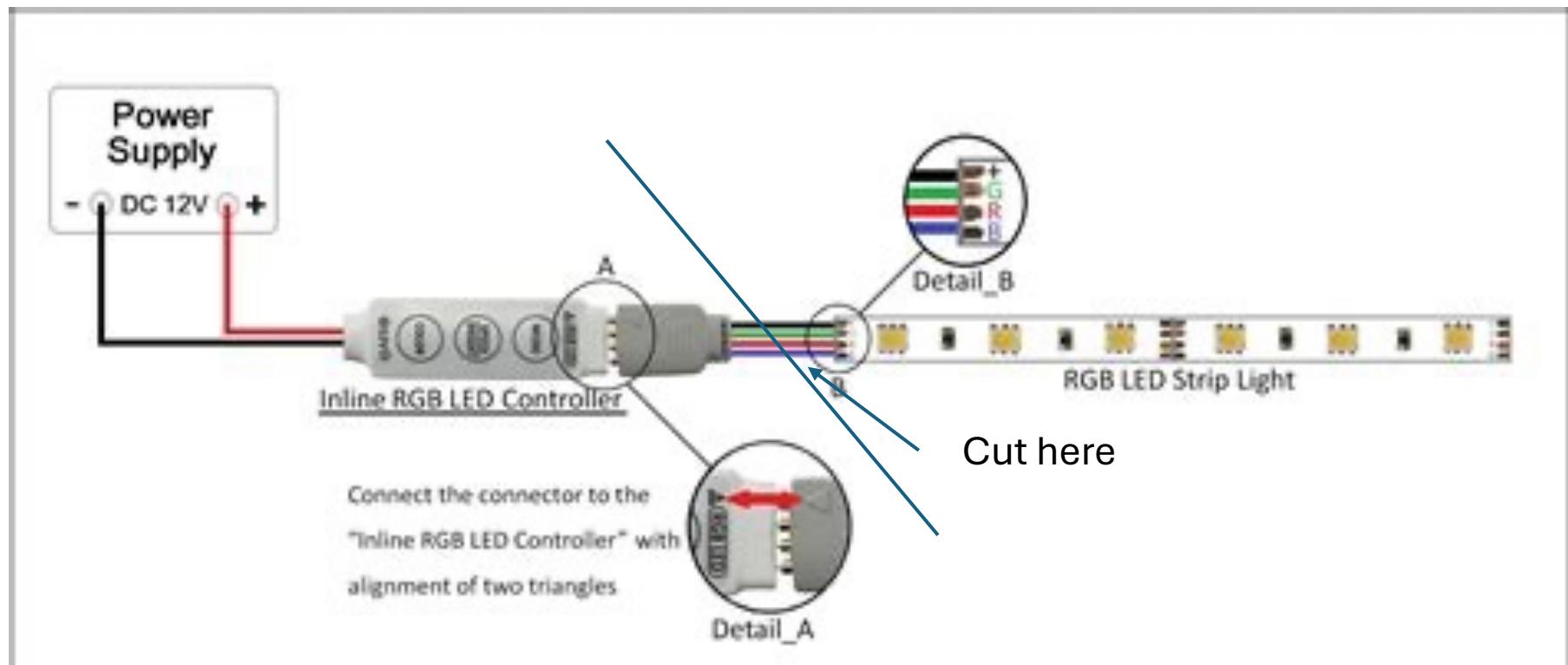


Let's Operate a 12V LED Light Strip using our ATmega328P

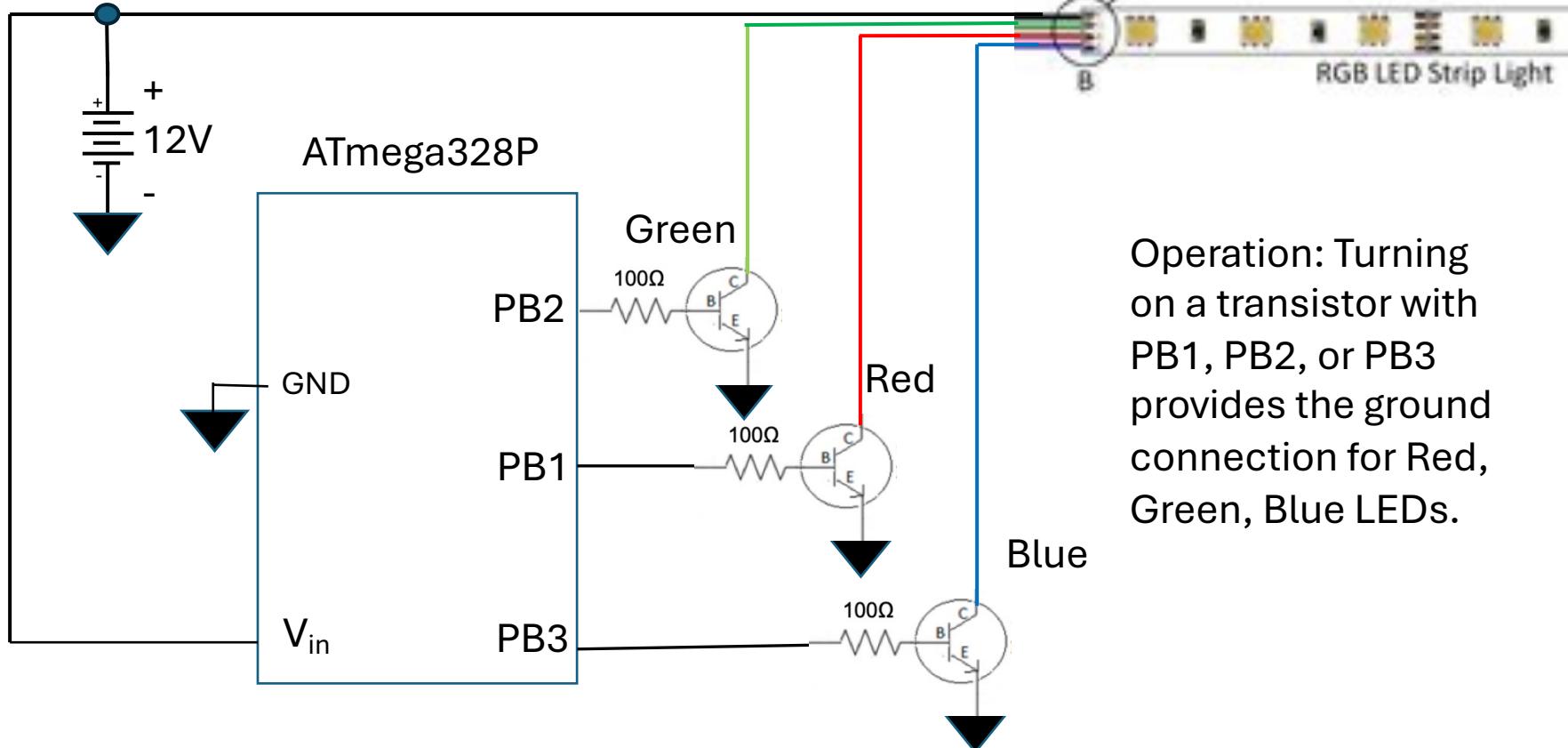
LED Strip Operation:

- Strip is a string of RGB LEDs wired in parallel, each with current-limiting resistors
- Black Wire (+) is 12V and connects to the anodes (+) of the RGB LEDs going all down the line.
- Grounding (-) the R, G, or B wires will ground the cathodes (-) of the respective red, green, or blue LEDs all down the line, causing them to glow.



8 AA
batteries in
series

+12V



disconnect when the
Arduino Uno dev board
connects to USB

We actually use 1kOhm instead of
100 Ohm resistors

```
*****  
* bruceforce.c    Blinks the RGB LED Light Strip  
* This code uses "brute force" to turn all bits of portB on and  
* off in a blinking pattern. Result is white LED's only.  
* Version  Date      Author      Revision  
* 1.0       4/8/25     D. McLaughlin initial code creation  
* *****
```

```
// _____ preamble _____  
#include <avr/io.h>           // Defines port pins  
#include <util/delay.h>         // Declares _delay_ms  
#define ONDELAY 3000             // LEDs ON in msec  
#define OFFDELAY 1000            // LEDs OFF in msec  
  
int main(void){  
  
    // _____ inits _____  
    DDRB = 0xFF;                  // Make all of PortD output pins  
  
    // _____ event loop _____  
    while(1){                      // Loop forever  
        PORTB = 0xFF;                // Make all pins of Port D high  
        _delay_ms(ONDELAY);        // Wait  
        PORTB = 0x00;                // Make all pins of Port D low  
        _delay_ms(OFFDELAY);       // Wait  
    } // end event loop  
  
    return 0;                    // Code never gets here.  
}  
  
***** End of File *****
```

```
/* **** RGB_sequencer.c ... ****
 * RGB_sequencer.c Operates the LED's of the RGB LED Light Strip
 * in sequence. This code uses bit-level manipulations to turn on
 * and off individual R, G, B bits to achieve desired colors.
 * Assume R = PB1; G = PB2; B = PB3.
 * Version Date Author Revision
 * 1.0 4/8/25 D. McLaughlin initial code creation
 * **** */

// _____ preamble _____
#include <avr/io.h> // Defines port pins
#include <util/delay.h> // Declares _delay_ms
#define ONDELAY 2000 // LEDs ON in msec
#define RED PB1
#define GREEN PB2
#define BLUE PB3

int main(void){

    // _____ inits _____
    DDRB = 1<<DDB1|1<<DDB2|1<<DDB3; // Make PB1, PB2, PB3 outputs; PB0 & PB4-PB7 inputs

    // _____ event loop _____
    while(1){ // Loop forever

        // Make the light strip Red
        PORTB = 1<<RED; // Red ON
        PORTB &= ~(1<<GREEN); // Green off
        PORTB &= ~(1<<BLUE); // Blue off
        _delay_ms(ONDELAY); // Wait

        // Make the light strip green
        PORTB &= ~(1<<RED); // Red off
        PORTB |= (1<<GREEN); // Green ON
        PORTB &= ~(1<<BLUE); // Blue off
        _delay_ms(ONDELAY); // Wait

        // Make the light strip blue
        PORTB &= ~(1<<RED); // Red off
        PORTB &= ~(1<<GREEN); // Green off
        PORTB |= (1<<BLUE); // Blue ON
        _delay_ms(ONDELAY); // Wait

        // Make the light strip magenta (red + blue)
        PORTB |= (1<<RED); // Red ON
        PORTB &= ~(1<<GREEN); // Green off
        PORTB |= (1<<BLUE); // Blue ON
        _delay_ms(ONDELAY); // Wait

    } // end event loop

    return 0; // Code never gets here.
}
```

