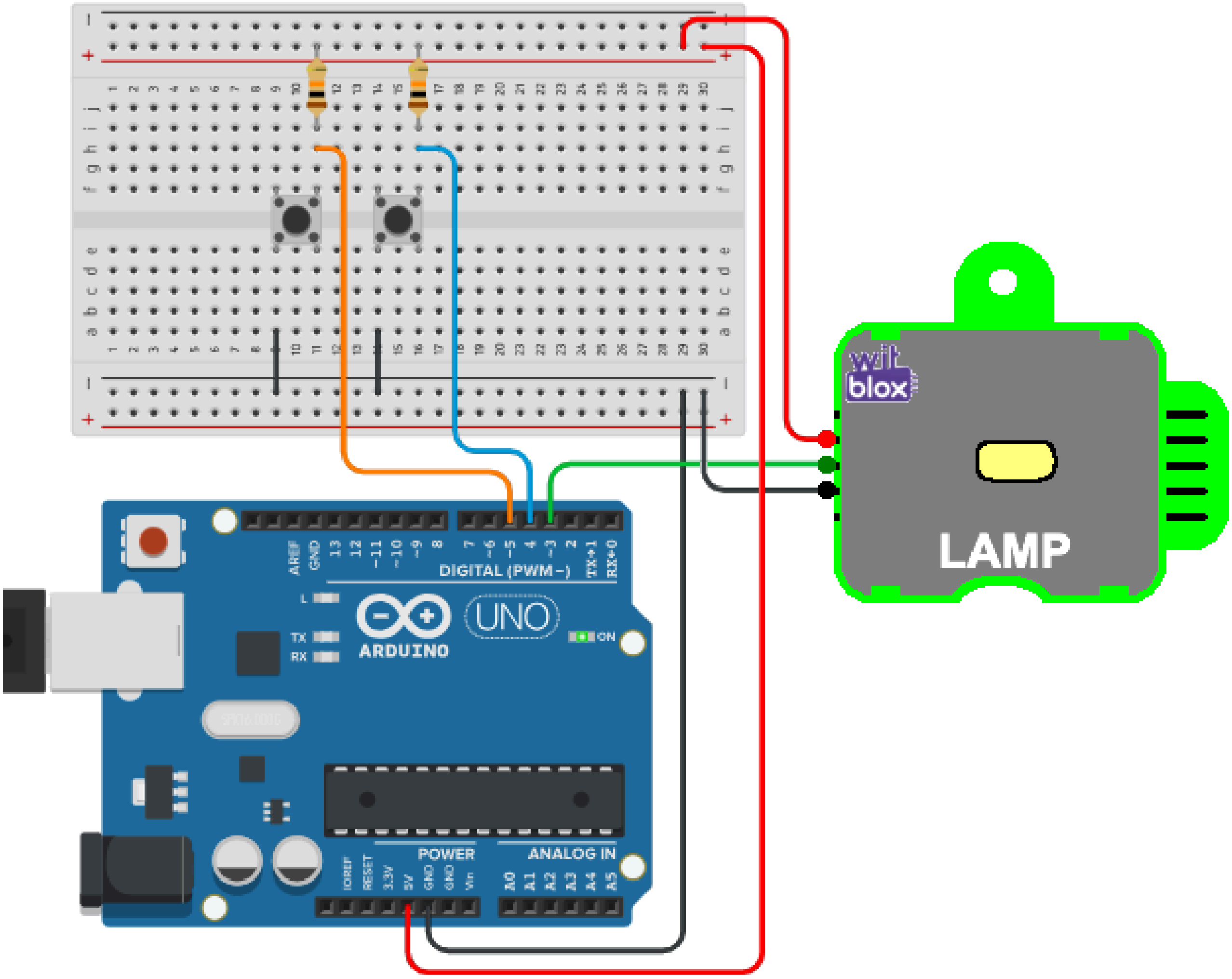
Activity: Make a Dimmable LED Lamp Using Digital Inputs

Objective:

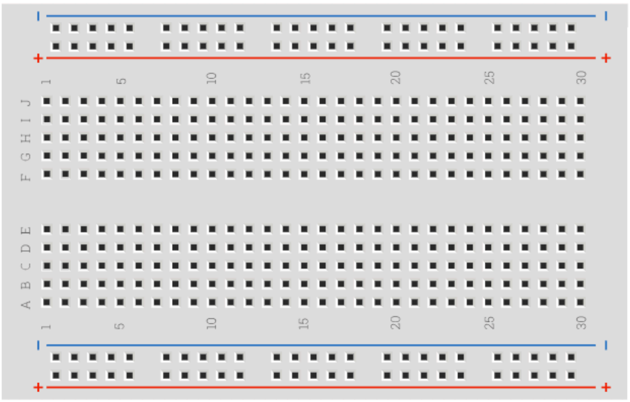


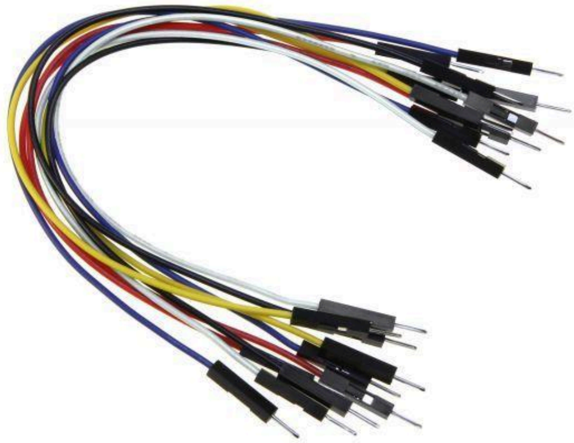
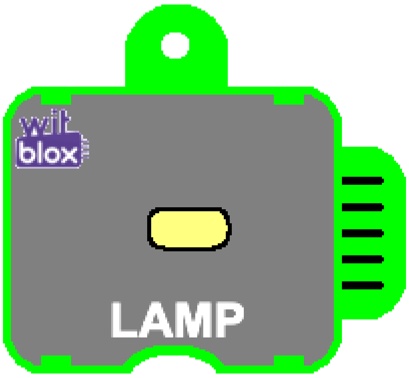
So far we have seen and learnt how to turn LED lamps On and Off using Arduino code, using Push Buttons as Momentary and Toggle type inputs. Then we learnt how to control the brightness of an LED lamp using delays as well as PWM or the analogWrite(); function and with analog inputs such as the LDR.

But the same LED lamp can also be controlled with digital inputs coming from a sensor or Push Buttons. We can have dedicated buttons connected to the Arduino for increasing and decreasing the brightness of the LED Lamp. Pressing the buttons will increase or decrease the brightness level and it will be reflected on the Lamp.

So, in today's activity we will learn how to make a Dimmable LED lamp which will be controlled by two Push Buttons, one for incrementing and the other for decrementing the Brightness level and an Arduino. The brightness of the Lamp will depend on the present value set by the buttons.

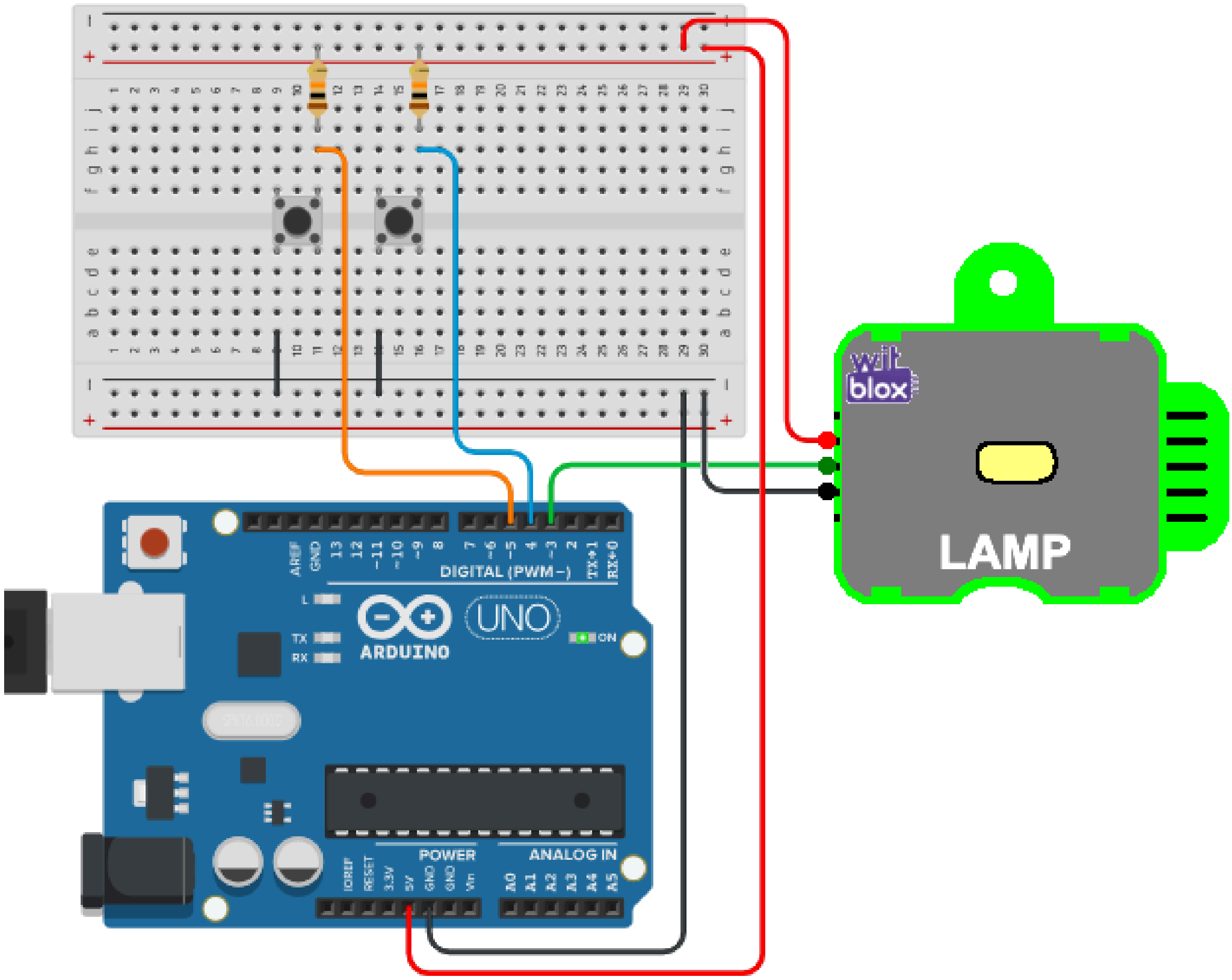
Materials Required:

S.no.	Part	Qty	Image
1	Arduino (Nano / UNO)	1	
2	Breadboard	1	
3	USB cable for Arduino	1	

4	Connection Wires (M - M)	10	
5	4-pin Tact Switch (Push Button)	2	
6	10k resistor (Brown Black Orange Golden)	2	
7	Lamp Blox	1	

Connection Diagram:

The connection diagrams shown in the two images below are the two versions of the same circuit.



Explanation:

The connection diagrams shown in both the images show the connections between the Arduino, the Input and the Output. The LDR, which is the analog light sensing component, is the Input for the Arduino while the Lamp blox is the Output for the Arduino.

When we closely observe the connection diagrams, it becomes quite clear that:

1. The Increment Push Button is connected such that its top-right pin is connected to the lower pin of the 10k pull-up resistor and its bottom-left pin is connected to the -ve of the breadboard. Additionally, its top-right pin is also

connected to the Arduino digital pin 4 (D4) through a connection wire that is shown as the Blue line in the diagrams above.

2. The Decrement Push Button is connected such that its top-right pin is connected to the lower pin of the 10k pull-up resistor and its bottom-left pin is connected to the -ve of the breadboard. Additionally, its top-right pin is also connected to the Arduino digital pin 5 (D5) through a connection wire that is shown as the Orange line in the diagrams above.

3. The 10k pull-up resistor for the Increment Push button is connected such that its one side is connected to the +ve of the breadboard while the other side is connected to the top-right pin of the Push button.

4. The 10k pull-up resistor for the Decrement Push button is connected such that its one side is connected to the +ve of the breadboard while the other side is connected to the top-right pin of the Push button.

5. The Lamp blox is connected such that its +ve power pin (2nd pin from the top) is connected to the +ve of the breadboard and its -ve power pin (4th pin from the top) is connected to the -ve of the breadboard. The Data input pin of the Lamp blox (3rd pin from the top) is connected to Arduino digital pin 3 (D3).

6. The 5v pin of the Arduino is connected to the +ve of the breadboard and the Gnd pin of the Arduino is connected to the -ve of the breadboard.

7. Finally, the USB cable from the Arduino is connected to the Computer’s USB. This cable serves for providing power, program uploading, as well as bi-directional serial data communication.

Arduino Code:

Here is the complete Arduino code for making a Dimmable LED Lamp using an Arduino and Digital Inputs.

```
#define Lamp      3
#define Increment 4
#define Decrement 5

int increment_state;
int decrement_state;
int brightness;

void setup()
{
 pinMode(Lamp, OUTPUT);
 pinMode(Increment, INPUT);
 pinMode(Decrement, INPUT);
}


void loop()
{
  increment_state = digitalRead(Increment);
  decrement_state = digitalRead(Decrement);

  if(increment_state == 0)
  {
     if(brightness<255)
      {
          brightness = brightness + 5;
          delay(500);
      }
  }

  if(decrement_state == 0)
  {
     if(brightness>0)
      {
          brightness = brightness - 5;
          delay(500);
      }
  }

  analogWrite(Lamp,brightness);
}
```

Explanation:

```
#define Lamp      3
#define Increment 4
#define Decrement 5
```

```
int increment_state;
int decrement_state;
int brightness;
```

Here, we are declaring that:

1. Digital pin 3 (D3) will be referred to as “Lamp” because it is used for connecting the Lamp blox.

2. Digital pin 4 (D4) will be referred to as “Increment” because it is used for connecting the increment Push Button.

3. Digital pin 5 (D5) will be referred to as “Decrement” because it is used for connecting the decrement Push Button.

4. Then we take 3 different integer type variables for storing the state of the increment button, the decrement button and brightness level of the LED Lamp and call them “increment_state”, “decrement_state” and “brightness” respectively.

```
void setup()
{
 pinMode(Lamp, OUTPUT);
 pinMode(Increment, INPUT);
 pinMode(Decrement, INPUT);
}
```

Here, inside the setup function, we are declaring that the “Increment” pin and the “Decrement” pin will be used as Input pins and the Lamp pin will be used as an Output pin. These need to be done just once.

```
void loop()
{
  increment_state = digitalRead(Increment);
  decrement_state = digitalRead(Decrement);

  if(increment_state == 0)
  {
     if(brightness<255)
```

```
      {
          brightness = brightness + 5;
          delay(500);
      }
  }

  if(decrement_state == 0)
  {
     if(brightness>0)
      {
          brightness = brightness - 5;
          delay(500);
      }
  }

  analogWrite(Lamp,brightness);
}
```

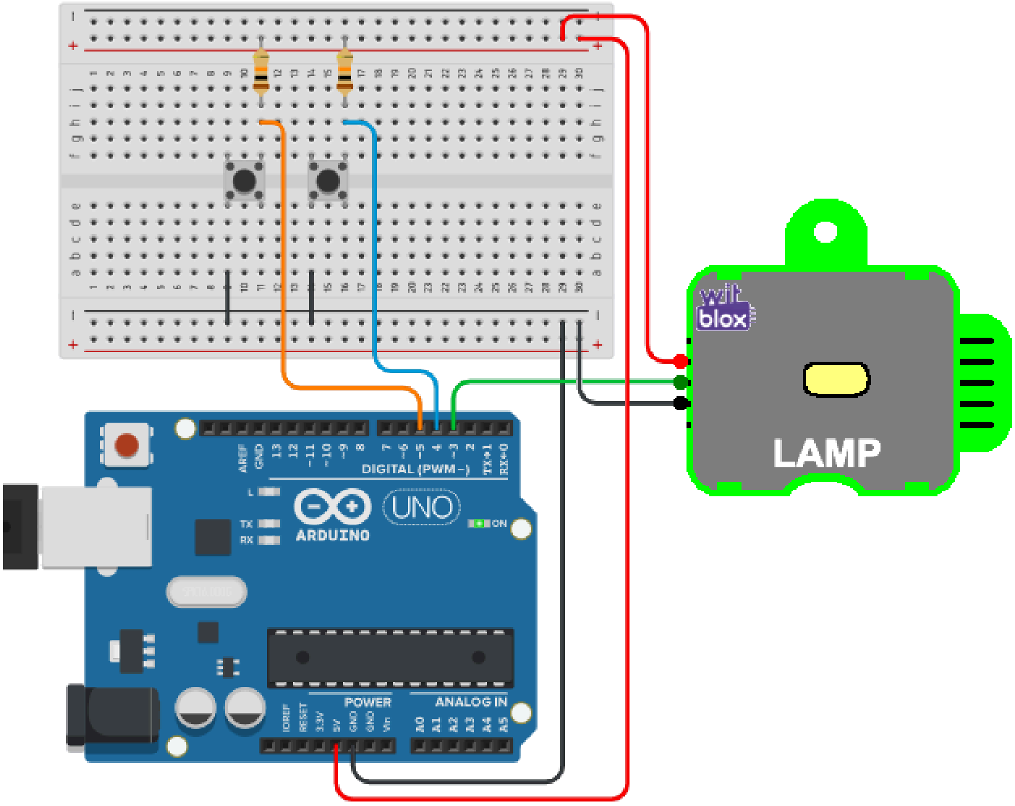
Here, in the loop function, first the Increment and Decrement button states are stored in the “increment_state” and “decrement_state” variables using two different “digitalRead()” functions.

If the Increment button is pressed and the present brightness level is less than 255 (highest possible value), the brightness is incremented by 5 and then wait for half a second.

If the Decrement button is pressed and the present brightness level is more than 0 (lowest possible value), the brightness is decremented by 5 and then wait for half a second.

Finally, once both the buttons have been checked and the value of the brightness variable has been set by pressing the increment or decrement buttons, the set value is implemented as brightness level on the LED Lamp by using an “analogWrite()” function on the Lamp pin of Arduino.

Outcome and Observations:



1. Once the Arduino code is compiled and uploaded, the LED on the Lamp blox is Off initially.

2. When the Increment Push Button is pressed and released for the first time, the Lamp is turned On with extremely low brightness and it might not even be visible if there is sufficient light in the surrounding area.

   As we keep pressing the Increment Button again and again, we see that the brightness of the LED Lamp keeps getting higher and higher. At some point we will reach the highest brightness level which is 255, and beyond that the Lamp brightness does not change even if we keep pressing the Button.

3. When the Decrement Push Button is pressed and released for the first time, the Lamp is turned On with very slightly decreased brightness and the change might not even be visible if there is sufficient light in the surrounding area.

   But, as we keep pressing the Decrement Button again and again, we see that the brightness of the LED Lamp keeps getting lower and lower. At some point we will reach the lowest brightness level which is 0, and beyond that the Lamp brightness does not change even if we keep pressing the Button.

4. It is not mandatory to press the Decrement button after reaching full brightness. Anywhere before reaching the full brightness, if the Decrement button is pressed, the brightness will go down and similarly if the Increment Button is pressed, the brightness will go up.