

## Project 1: Introduction to Arduino and LED Blink

### Description:

The Arduino is a microprocessor. A Microprocessor is a compact programmable computing device with memory, processing, input, and output pins. In this lesson we will:

1. Build and wire a device with one LED, resistor, and Arduino board.
2. Program the device to “Blink the Light”
3. Identify the flow of electricity / signal in this computing unit.
4. Use the Blockly Programming system to control speed of Blink.

### Build and Wiring:

You will need:

1. LED Light Bulb (Any Color)



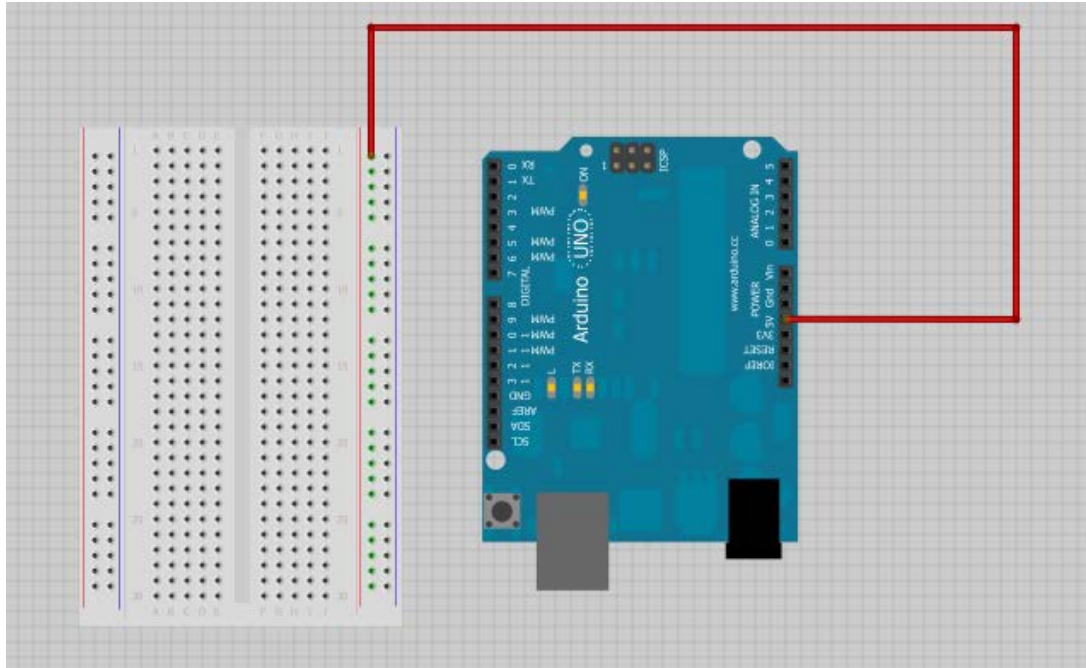
2. 330 Ohm Resistor



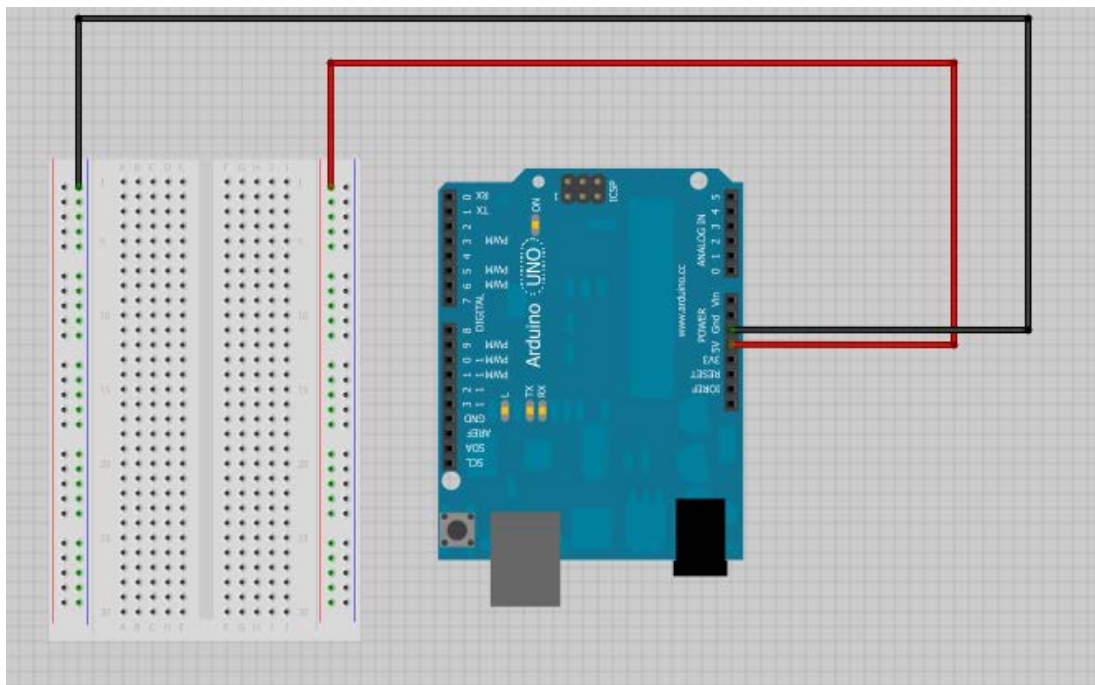
3. Red Wire
4. White Wire
5. Black (Or dark colored Wire)

**Process:**

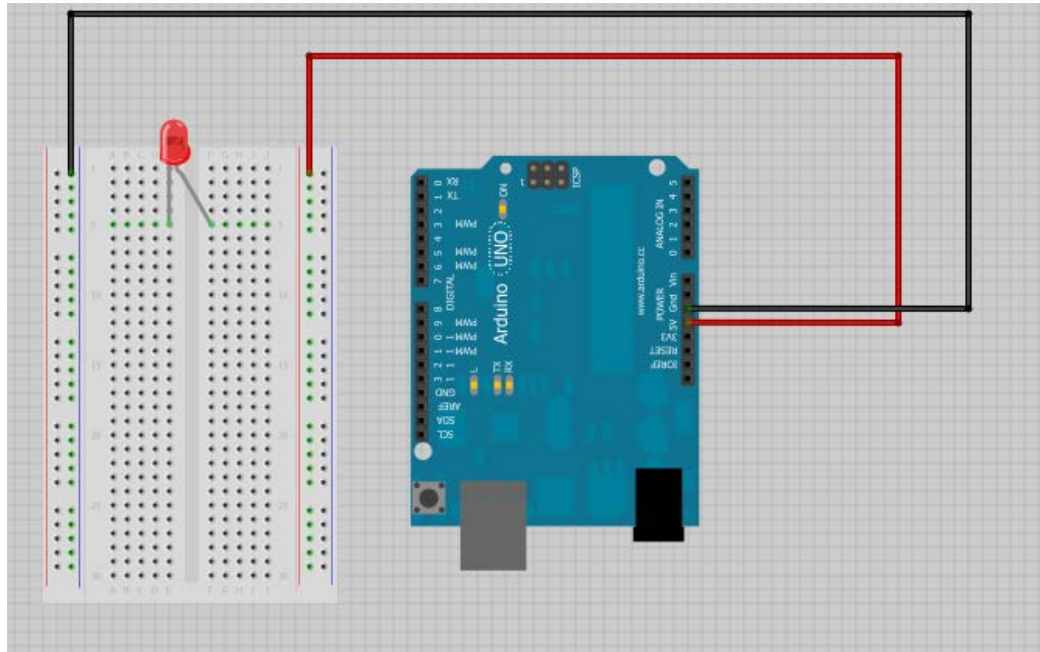
1. Run a red Wire from the 5V Pin to the Red Rail on the Breadboard. This will connect the current side of the circuit.



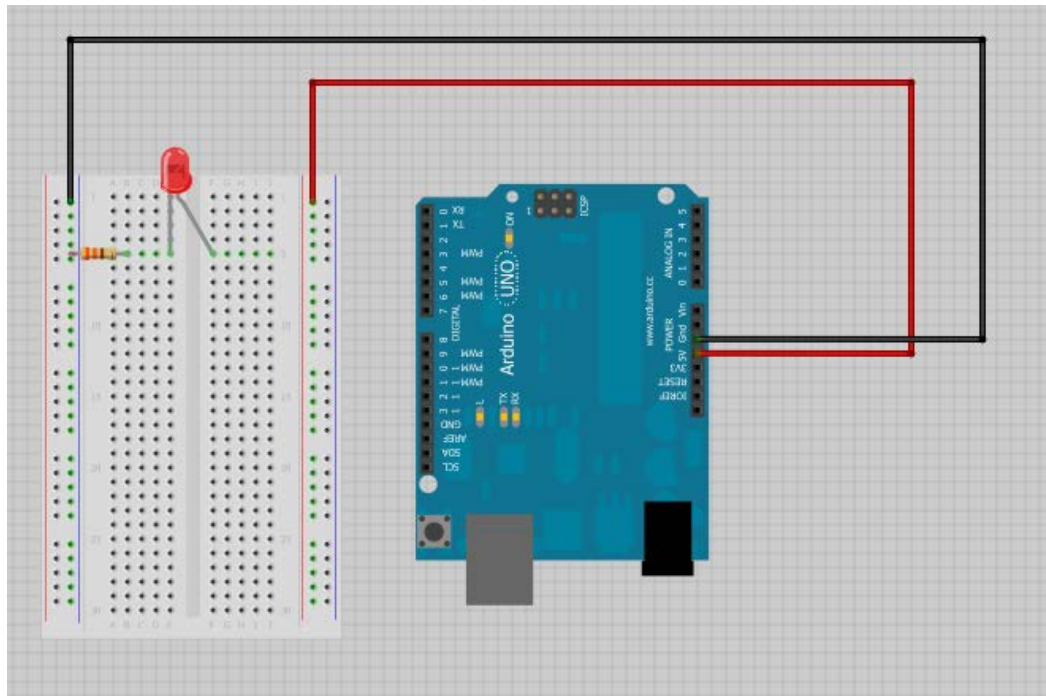
2. Run a black (or dark colored wire) from the Gnd Pin of the Arduino to the Blue Rail on the far side of the Breadboard.



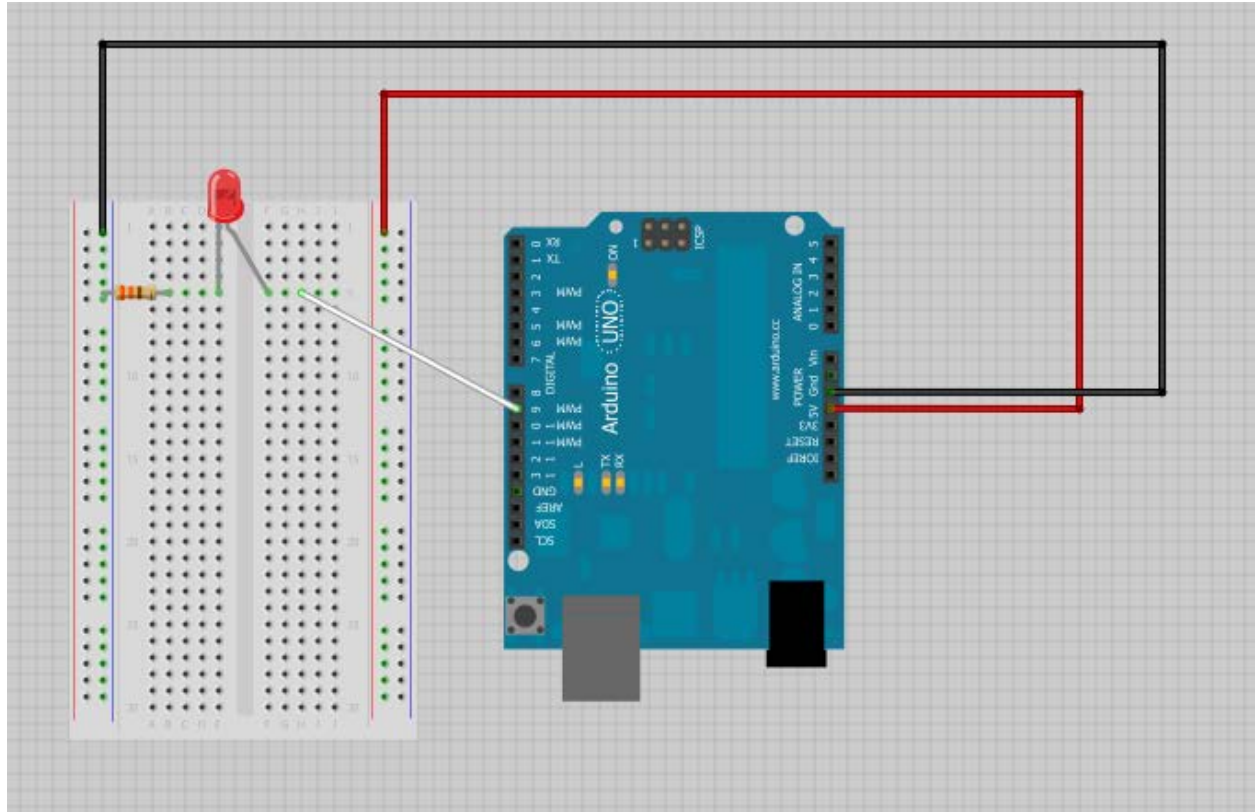
3. Plug an LED bulb into the circuit. Note that the longer pin will face the Arduino Board and the pins should cross the “gap” in the breadboard. (Plug the Long Pin into F5 and the Short Pin into E5).



4. This step is VERY IMPORTANT!!! Plug a 330 OHM resistor from Port B5 to Ground (The blue rail). An LED bulb must ALWAYS have a Resistor in the circuit. If we do not use a Resistor, we will burn out the bulb or the Arduino Board.



5. Now we will run the signal wire. Wire Pin 9 on the Arduino to H5 on the Breadboard. This will carry the current from the Arduino Pin to the LED and allow the Arduino to switch on and off the LED.



6. You are finished! Go on to the next section to program your Arduino.

## Blockly Programming:

1. Go to the website: <http://nebomusic.net/blockly/apps/blocklyduino/index.html>
2. We want to define a procedure (write directions) for our Arduino Blinky to switch the light. Click on "Procedures" and drag a procedure block to the screen.

The screenshot shows a web browser window with the address bar displaying `nebomusic.net/blockly/apps/blocklyduino/index.html`. The page title is "BlocklyDuino > web-based visual programming editor for arduino". The interface is divided into two main sections: a left sidebar and a main workspace.

The sidebar, titled "Blocks", lists various categories: In/Out, Servo, Grove Analog, Grove, Grove LCD, Grove Motor, Control, Math, Text, Logic, Tone, Variables, and Procedures. The "Procedures" category is currently selected and highlighted in blue.

Under the "Procedures" category, there are four purple blocks visible in the "Arduino" tab: a "procedure" block with a "+" icon and a "do" slot; a "procedure" block with a "+" icon, a "do" slot, and a "return" slot; an "if" block with a "return" slot; and a "do procedure" block. A blue arrow points from the first "procedure" block in the sidebar to a similar "procedure" block in the main workspace, indicating the action of dragging the block from the sidebar to the workspace.

3. We need to name our 'procedure'. Click on the title block and name the block 'blink'. (Note that we will always start procedures with a lower case letter).

The screenshot shows a web browser window with the title "Blockly Demo: Arduino" and the URL "nebomusic.net/blockly/apps/blocklyduino/index.html". The page header reads "BlocklyDuino > web-based visual programming editor for arduino". Below the header, there are two tabs: "Arduino" (selected) and "XML". On the left, a sidebar lists various block categories: "Blocks", "In/Out", "Servo", "Grove Analog", "Grove", "Grove LCD", "Grove Motor", "Control", "Math", "Text", "Logic", "Tone", "Variables", and "Procedures". The main workspace contains a single purple block with a yellow border, labeled "do" and "blink", representing a procedure block.

4. Click on 'In/Out' and drag a 'DigitalWrite PIN#' block and place it inside the 'blink' block.



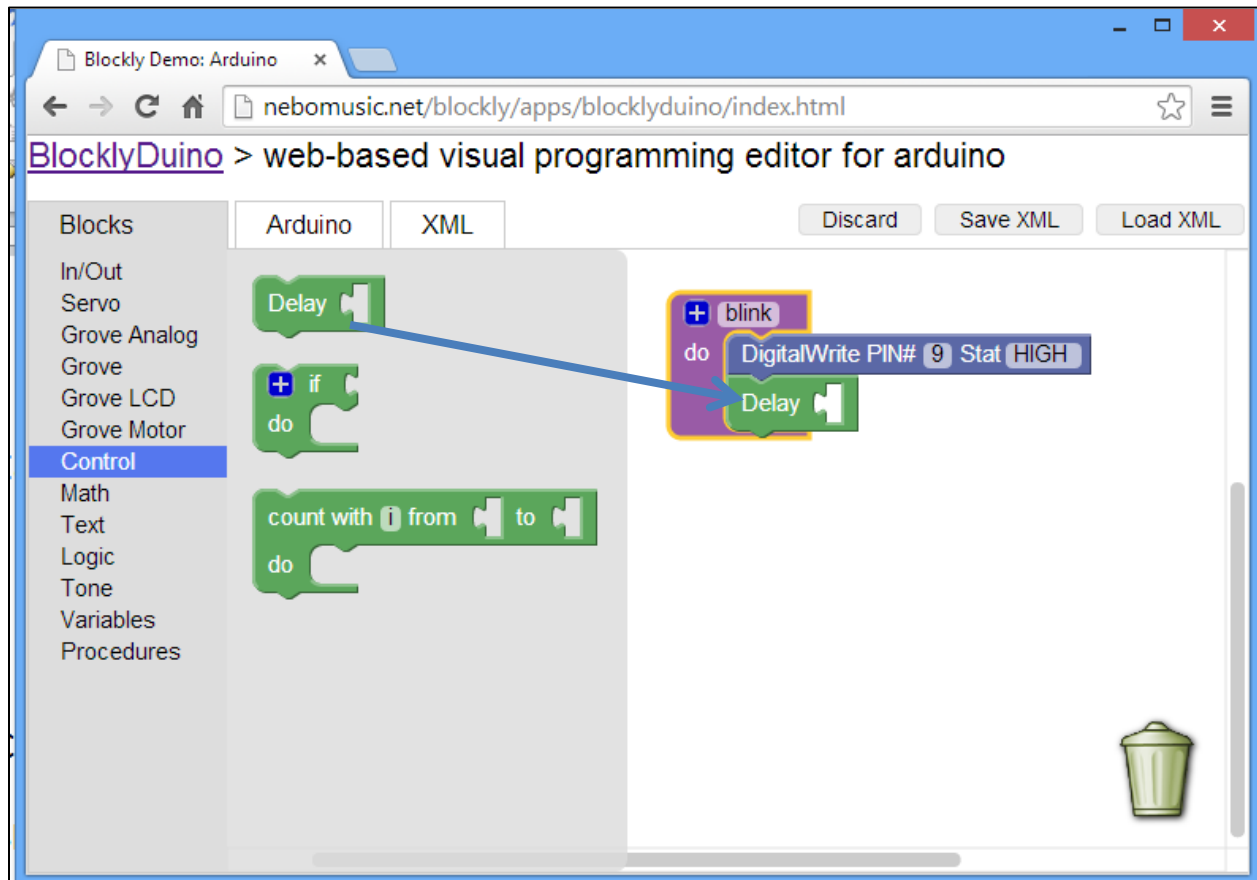
5. Our Arduino device has the LED plugged in to Pin 9. So we will change the PIN# to 9:

The screenshot shows a web browser window with the following elements:

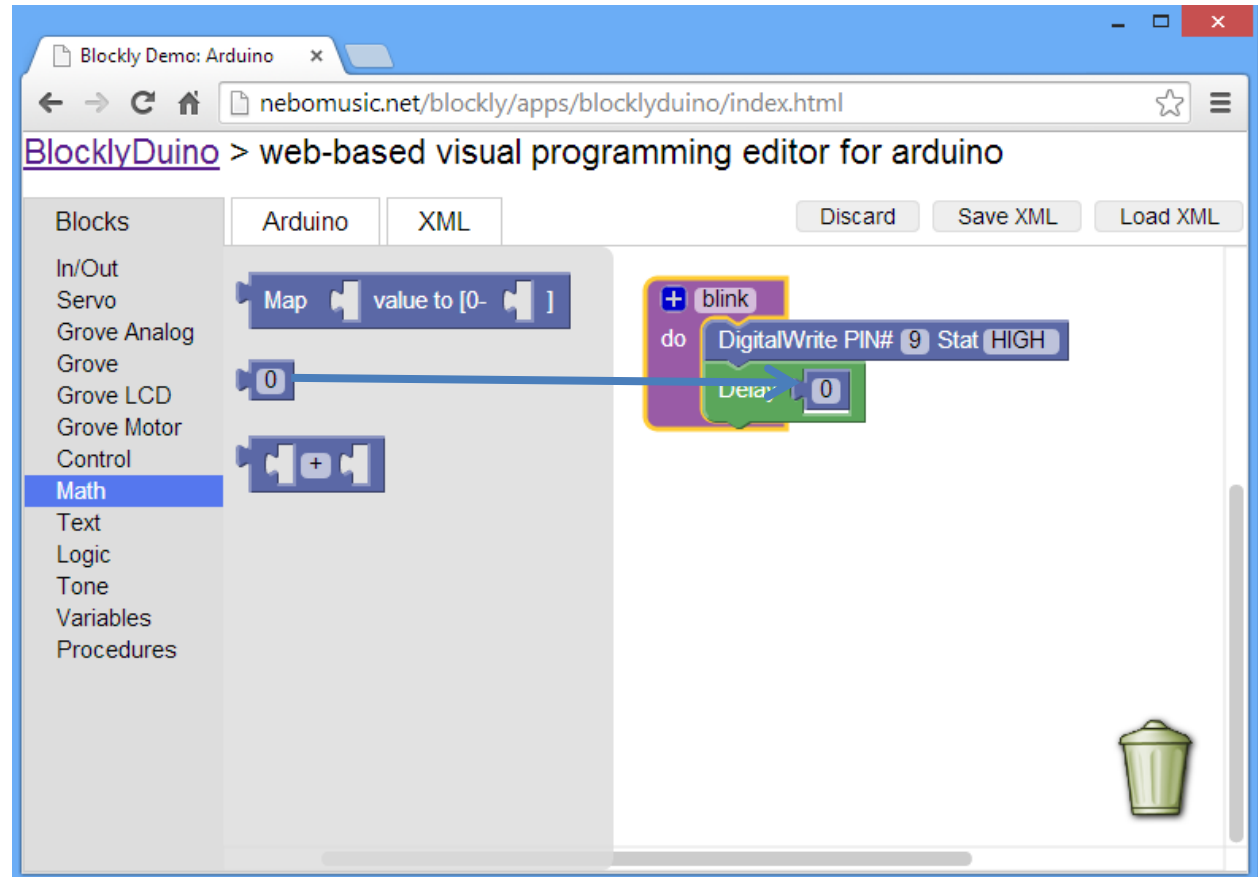
- Browser Tab:** "Blockly Demo: Arduino"
- Address Bar:** "nebomusic.net/blockly/apps/blocklyduino/index.html"
- Page Title:** "BlocklyDuino > web-based visual programming editor for arduino"
- Navigation:** "Discard", "Save XML", and "Load XML" buttons.
- Left Panel (Blocks):** A list of categories including In/Out, Servo, Grove Analog, Grove, Grove LCD, Grove Motor, Control, Math, Text, Logic, Tone, Variables, and Procedures.
- Main Workspace:** Contains a "do" block with a "DigitalWrite PIN# 9 Stat HIGH" block nested inside it. A "blink" block is also visible above the "do" block.
- Bottom Right:** A trash can icon for deleting blocks.



6. 'HIGH' means to turn the current on for the Pin. We will leave this set to HIGH. Next, click on 'Control' and drag a 'Delay' block into the 'blink' function.



7. We want the LED to blink on for ½ second and off for ½ second. The Delay takes in numbers measuring in milliseconds. That means 1000 milliseconds equals 1 second. For ½ a second, we will place the number 500 in the delay block. Click on 'Math' and drag a number block inside the 'Delay' block.



8. Now type '500' in the Delay block.

The screenshot displays the BlocklyDuino web-based visual programming editor. The browser window title is "Blockly Demo: Arduino" and the address bar shows "nebomusic.net/blockly/apps/blocklyduino/index.html". The page title is "BlocklyDuino > web-based visual programming editor for arduino".

The interface features a left sidebar with a "Blocks" category and a list of sub-categories: In/Out, Servo, Grove Analog, Grove, Grove LCD, Grove Motor, Control, Math, Text, Logic, Tone, Variables, and Procedures. The main workspace is divided into two tabs: "Arduino" (selected) and "XML". Above the workspace are three buttons: "Discard", "Save XML", and "Load XML".

The workspace contains a "do" loop block. Inside the loop, there are two blocks: a "DigitalWrite PIN# 9 Stat HIGH" block and a "Delay" block. The "Delay" block has the value "500" entered in its input field. A trash can icon is located in the bottom right corner of the workspace.

9. This will turn the LED on. Add the following blocks to have the LED turn off. Note that we set the value of 'Stat' to 'LOW' and place another Delay 500 block.

The screenshot displays the BlocklyDuino web-based visual programming editor. The browser address bar shows the URL `nebomusic.net/blockly/apps/blocklyduino/index.html`. The page title is "BlocklyDuino > web-based visual programming editor for arduino". The interface includes a "Blocks" sidebar on the left with categories like In/Out, Servo, Grove Analog, Grove, Grove LCD, Grove Motor Control, Math, Text, Logic, Tone, Variables, and Procedures. The main workspace shows a "do" loop containing the following blocks: a "DigitalWrite PIN# 9 Stat HIGH" block, a "Delay 500" block, a "DigitalWrite PIN# 9 Stat LOW" block, and another "Delay 500" block. At the top right of the workspace, there are buttons for "Discard", "Save XML", and "Load XML". A trash can icon is located in the bottom right corner of the workspace.

10. We are almost done! We have defined the procedure 'blink' and told the Arduino what to do. Now we need to tell the Arduino when to run this function. Drag a 'do blink' block from the Procedures menu and place it on the screen.

The screenshot shows the BlocklyDuino web-based visual programming editor. The browser address bar displays `nebomusic.net/blockly/apps/blocklyduino/index.html`. The page title is "BlocklyDuino > web-based visual programming editor for arduino".

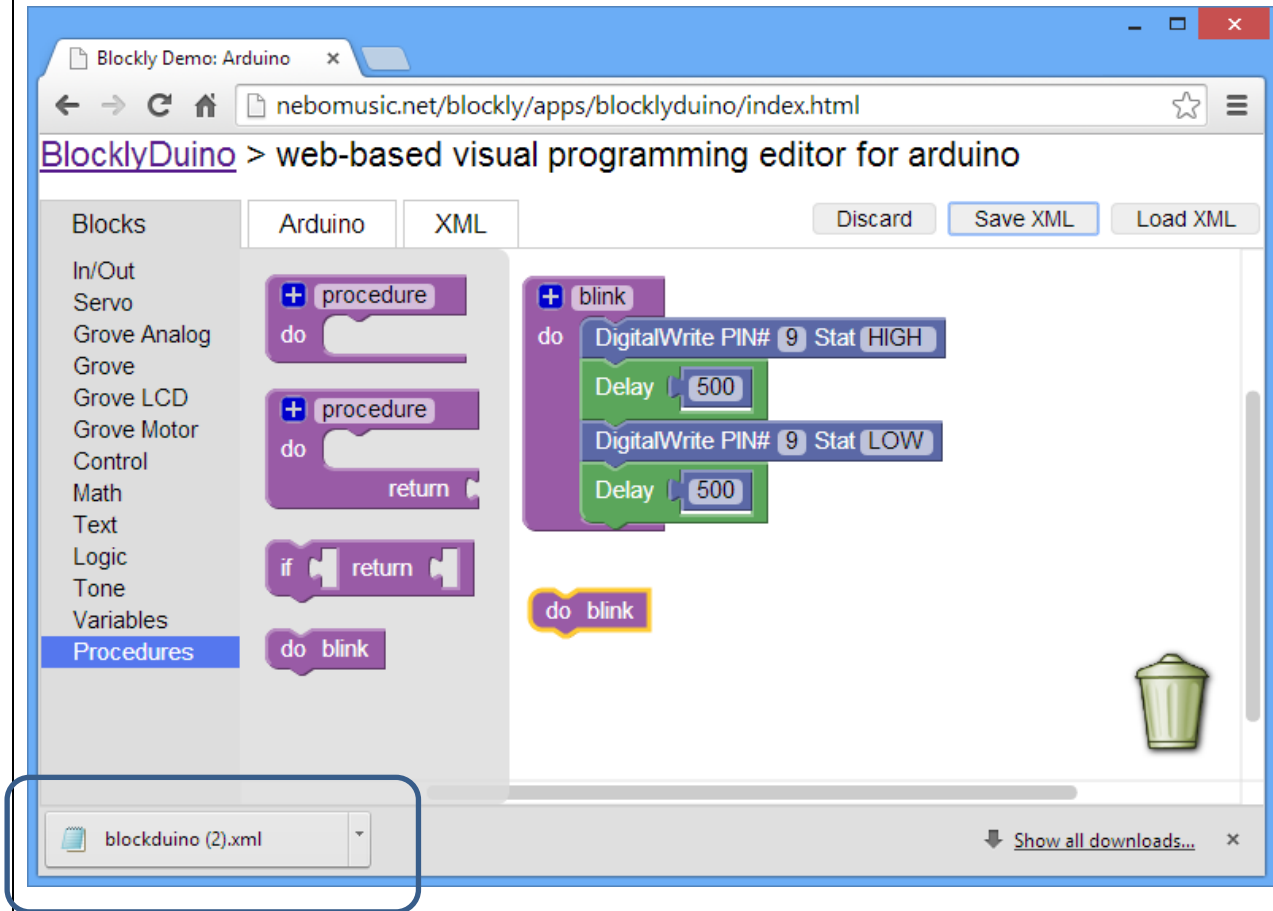
The interface is divided into a left sidebar and a main workspace. The sidebar, titled "Blocks", lists various categories: In/Out, Servo, Grove Analog, Grove, Grove LCD, Grove Motor, Control, Math, Text, Logic, Tone, Variables, and Procedures. The "Procedures" category is selected, showing a "do blink" block.

The main workspace is divided into two tabs: "Arduino" and "XML". The "Arduino" tab is active, showing a "do blink" procedure block. This block contains the following code blocks in sequence:

- DigitalWrite PIN# 9 Stat HIGH
- Delay 500
- DigitalWrite PIN# 9 Stat LOW
- Delay 500

Below the "do blink" procedure block, there is a separate "do blink" block. A trash can icon is visible in the bottom right corner of the workspace.

11. You are done! Click “Save XML” to save your work. (It will download on your computer.)



12. Follow the steps in the next section to download and run your code.

## Downloading the Blockly Program to the Arduino

1. Click on the Arduino Sketch Icon to start Sketch.



2. Plug the Arduino board into the computer's USB port.
3. Go back to the website and click on the 'Arduino' tab.

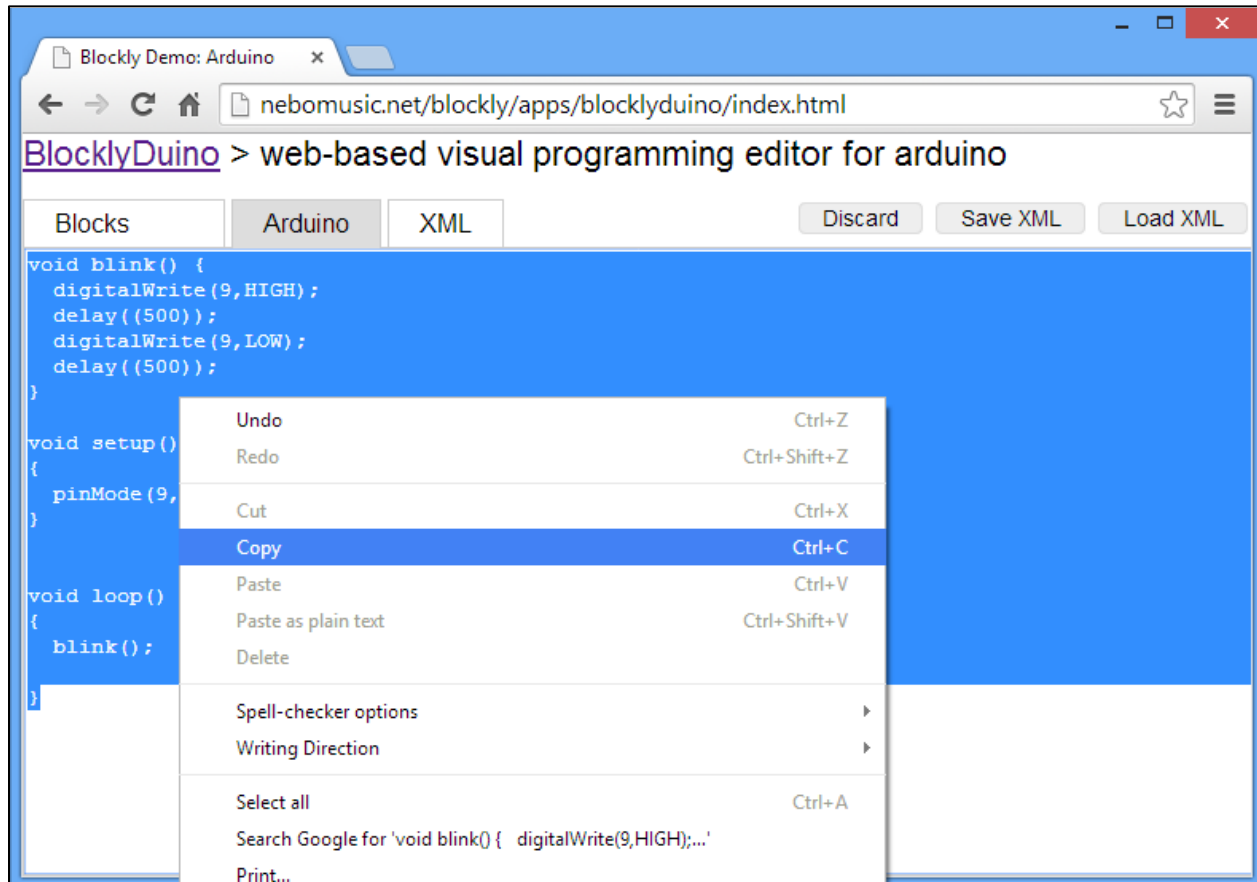
A screenshot of a web browser window showing the BlocklyDuino interface. The browser tab is titled "Blockly Demo: Arduino" and the address bar shows the URL "nebomusic.net/blockly/apps/blocklyduino/index.html". The page title is "BlocklyDuino > web-based visual programming editor for arduino". There are three tabs: "Blocks", "Arduino", and "XML", with "Arduino" selected. To the right of the tabs are three buttons: "Discard", "Save XML", and "Load XML". The main area contains a code editor with the following C++ code:

```
void blink() {
  digitalWrite(9,HIGH);
  delay((500));
  digitalWrite(9,LOW);
  delay((500));
}

void setup()
{
  pinMode(9, OUTPUT);
}

void loop()
{
  blink();
}
```

4. When we drag the blocks in the Block view, the website automatically creates the text code to load into the Arduino through Sketch. Select all the code in the Arduino tab and right click and select 'copy'.



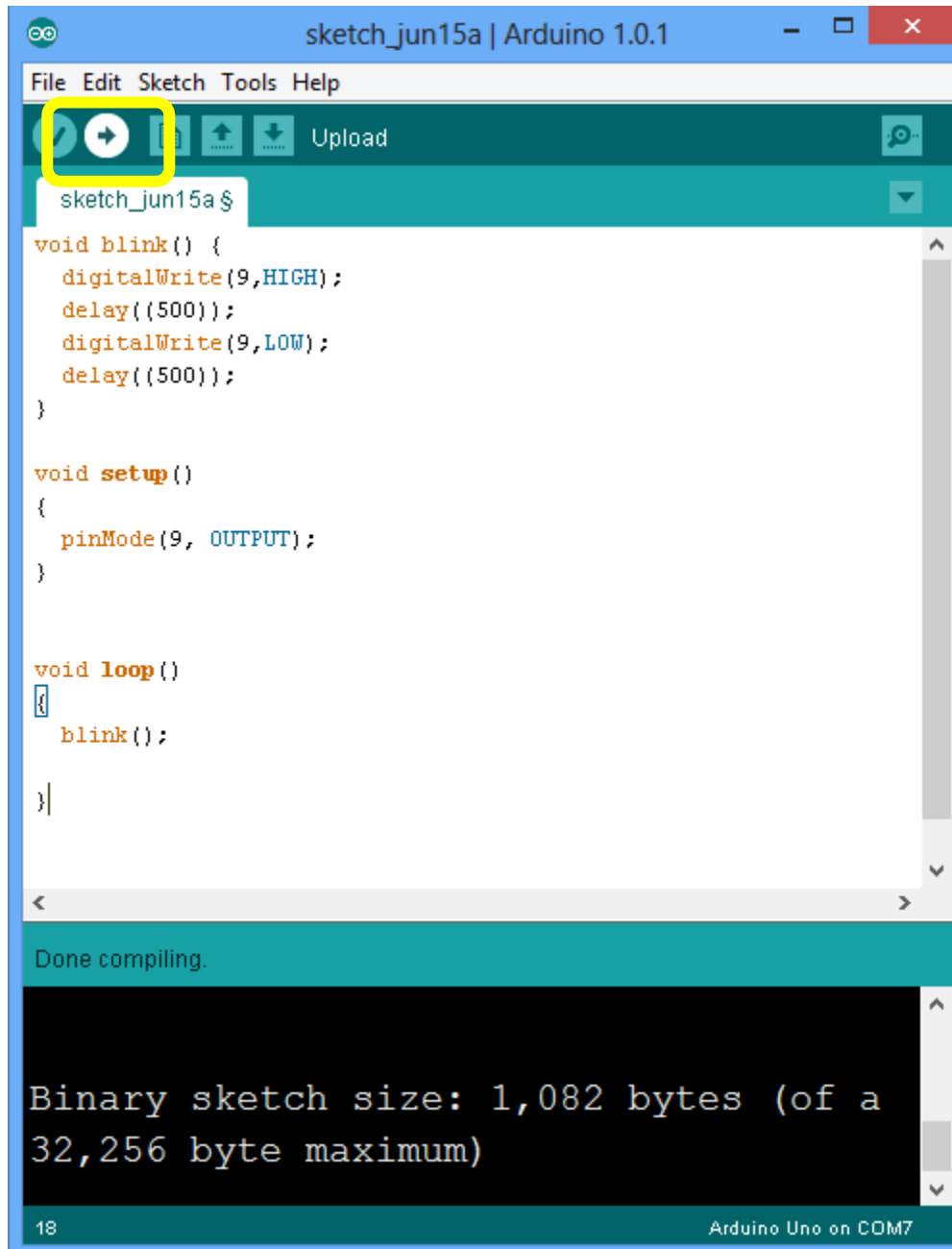


5. Go to Arduino Sketch right click. Select "Paste" to put the code into Sketch.

```
sketch_jun15a $  
void blink() {  
  digitalWrite(9,HIGH);  
  delay((500));  
  digitalWrite(9,LOW);  
  delay((500));  
}  
  
void setup()  
{  
  pinMode(9, OUTPUT);  
}  
  
void loop()  
{  
  blink();  
}
```

18 Arduino Uno on COM7

6. Click the Upload Icon to lead the code onto the Arduino. The light should blink!



7. Congratulations! Here are some other ideas to try:
- Change the value of the 'Delay' block to control the speed of the blink.
  - Create a new procedure 'blinkFast' and the following code and watch what happens!

Blockly Demo: Arduino x

nebomusic.net/blockly/apps/blocklyduino/index.html

BlocklyDuino > web-based visual programming editor for arduino

Discard Save XML Load XML

Blocks

- In/Out
- Servo
- Grove Analog
- Grove
- Grove LCD
- Grove Motor
- Control
- Math
- Text
- Logic
- Tone
- Variables
- Procedures

do blink

- DigitalWrite PIN# 9 Stat HIGH
- Delay 500
- DigitalWrite PIN# 9 Stat LOW
- Delay 500

do blinkFast

- DigitalWrite PIN# 9 Stat HIGH
- Delay 250
- DigitalWrite PIN# 9 Stat LOW
- Delay 250

count with i from 1 to 4

- do blink

count with i from 1 to 4

- do blinkFast

**Vocabulary:**

**Arduino Micro-Processor:** A mini computer with memory, processing, input, and output. We can program this computer to control electrical current and signals to control electric powered devices.

**Breadboard:** Where we can wire electrical circuits and make connections without permanently soldering the connections. A test platform for developing electronic devices.

**Signal:** A pulse or pulses of electrical current to carry information. The Arduino uses signals to control LED's, speakers, motors, and other devices.

**Current:** A steady state of charge from the positive side of the circuit

**Volts:** The measure of potential energy difference between the positive and negative side of a circuit.

**Amps:** The measure of how much current passes through the circuit in a given length of time.

**Ground:** The source of electrons or the negative side of the circuit.

**LED:** Light Emitting Diode. Acts as a one way gate for current and will emit a light when current is passed through the LED. LED's use very little power and are very common in electronic devices. Any light you see on an electronic device most likely is created by an LED.

**LED's have two pins:**

The Long Pin is wired to the Current side (+ side) of the circuit (Anode)

The Short Pin is wired to the Ground side (- side) of the circuit (Cathode)

**Resistor:** Resists the flow of current and electrons in a circuit. Measured in Ohms.