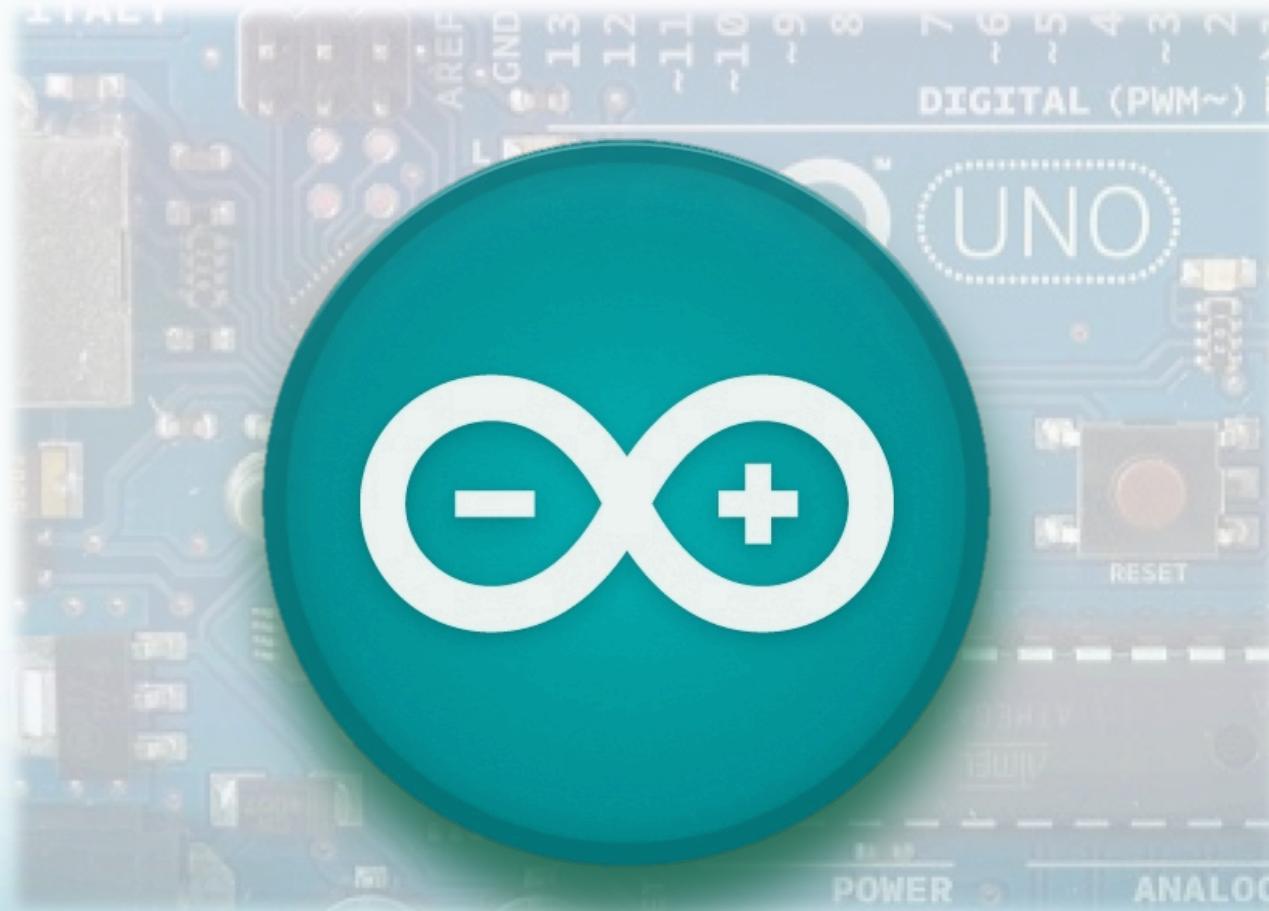
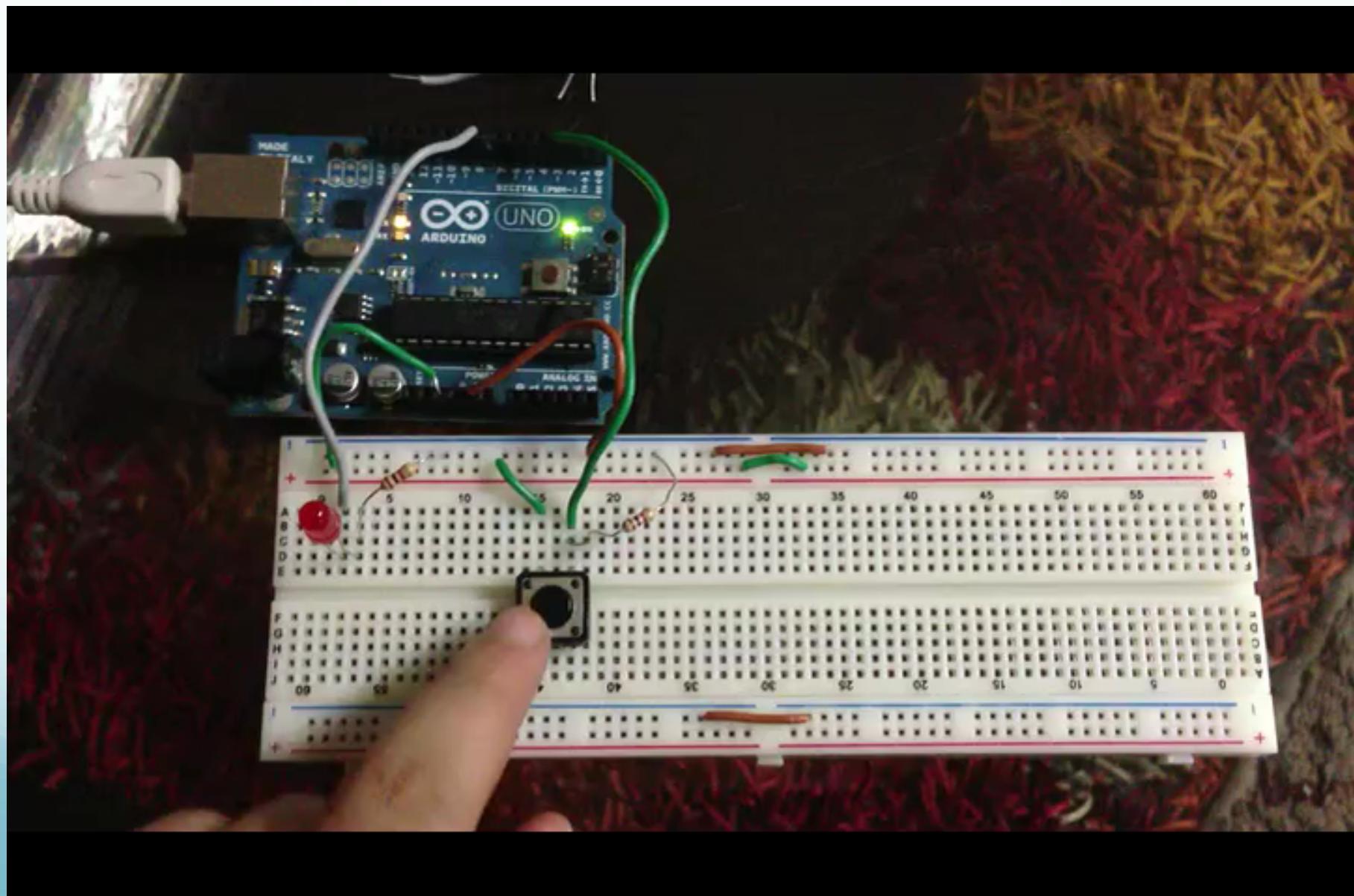


:Blink Blink:

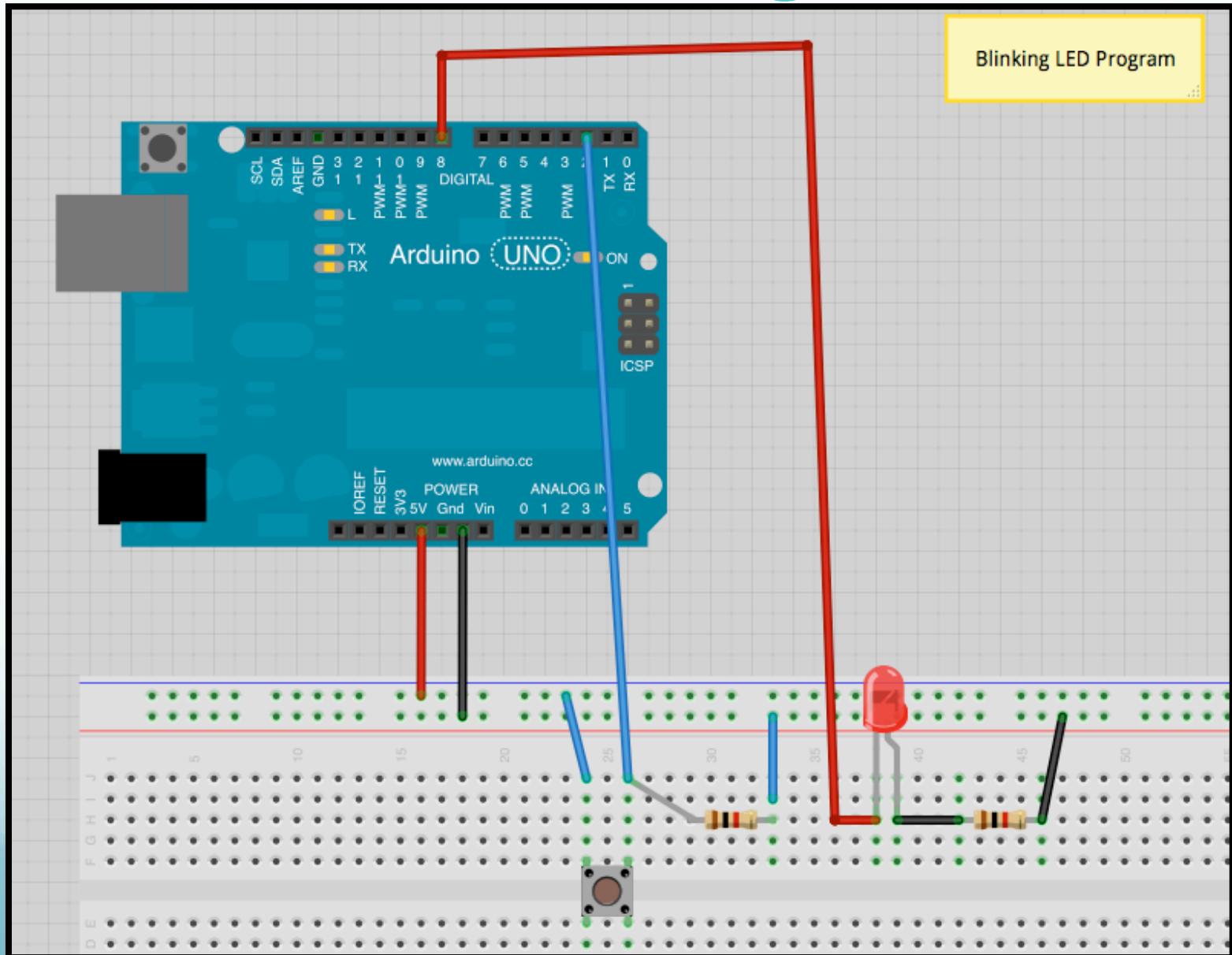


Er. Sahil Khanna
www.SahilKhanna.org

It's VIDEO time!

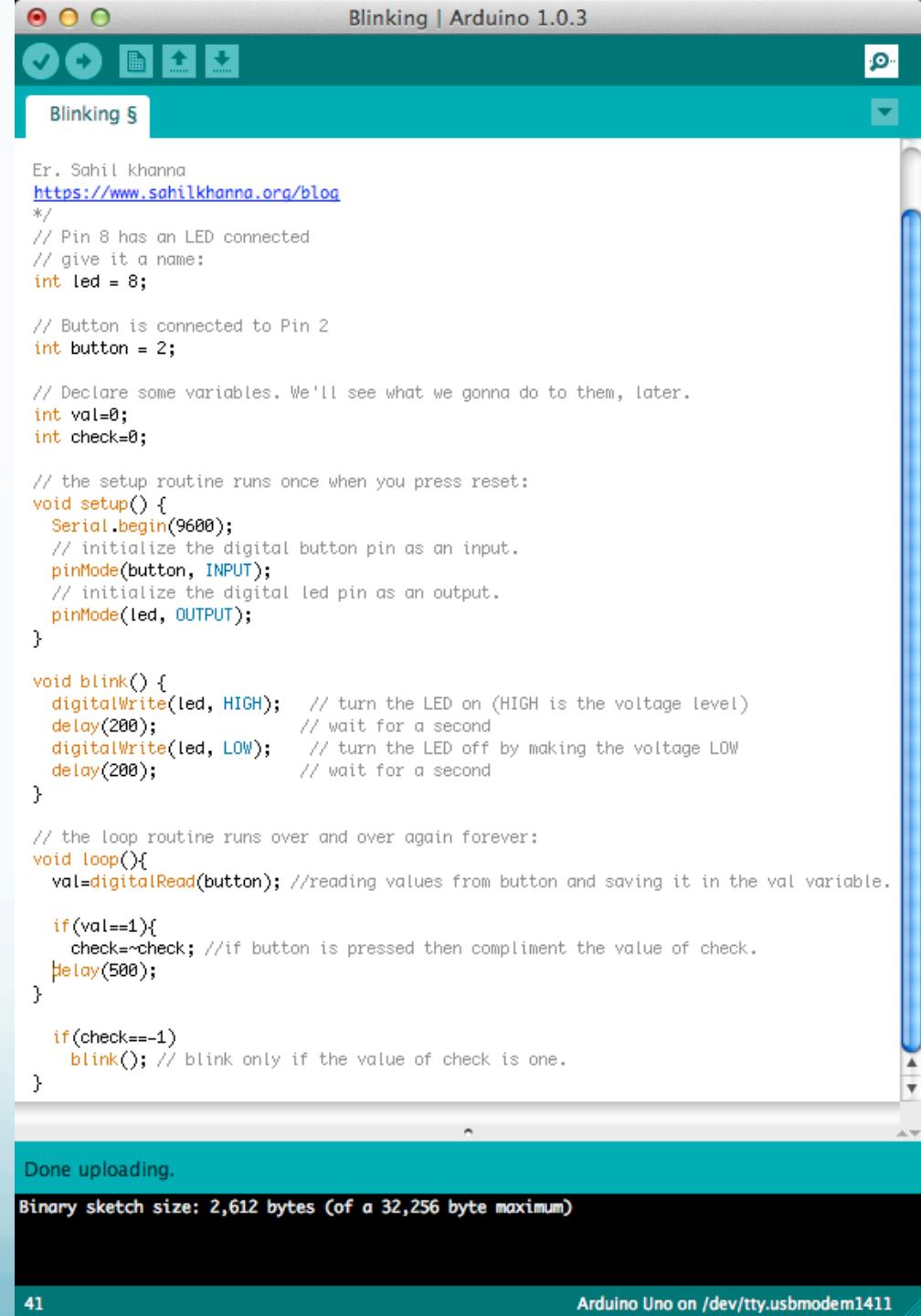


Circuit Diagram



//Code

Download Link ->
www.sahilkhanna.org/blog/arduino-blink-program



The screenshot shows the Arduino IDE interface. The title bar reads "Blinking | Arduino 1.0.3". The main window displays a C++ code for an Arduino sketch. The code includes setup and loop functions for an LED connected to pin 8 and a button connected to pin 2. It also includes logic to check if the button has been pressed and update a variable accordingly. At the bottom of the code editor, a message says "Done uploading." and "Binary sketch size: 2,612 bytes (of a 32,256 byte maximum)". The status bar at the bottom right indicates "Arduino Uno on /dev/tty.usbmodem1411".

```
Er. Sahil khanna
https://www.sahilkhanna.org/blog
*/
// Pin 8 has an LED connected
// give it a name:
int led = 8;

// Button is connected to Pin 2
int button = 2;

// Declare some variables. We'll see what we gonna do to them, later.
int val=0;
int check=0;

// the setup routine runs once when you press reset:
void setup() {
  Serial.begin(9600);
  // initialize the digital button pin as an input.
  pinMode(button, INPUT);
  // initialize the digital led pin as an output.
  pinMode(led, OUTPUT);
}

void blink() {
  digitalWrite(led, HIGH);    // turn the LED on (HIGH is the voltage level)
  delay(200);               // wait for a second
  digitalWrite(led, LOW);    // turn the LED off by making the voltage LOW
  delay(200);               // wait for a second
}

// the loop routine runs over and over again forever:
void loop(){
  val=digitalRead(button); //reading values from button and saving it in the val variable.

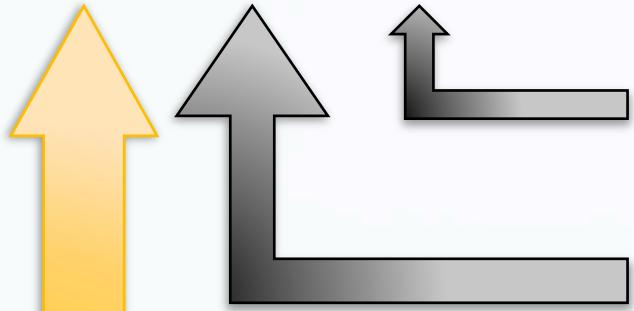
  if(val==1){
    check=-check; //if button is pressed then compliment the value of check.
    delay(500);
  }

  if(check==1)
    blink(); // blink only if the value of check is one.
}


```

Code Explantation

```
int led = 8;
```



Pin number of Arduino attached to led.

Assigning a variable for pin 8

Nature of variable i.e. integer

```
// Button is connected to Pin 2
```

```
int button = 2;
```

```
// Declare some variables. We'll see what we gonna do to them,  
later.
```

```
int val=0;
```

```
int check=0;
```

Code Explantation

```
// the setup routine runs once when you press reset:  
void setup() {  
    // initialize the digital button pin as an input.  
    pinMode(button, INPUT);
```



Setting what the pin(in this case the button) will act as i.e. **OUTPUT** or **INPUT**.

As we will be reading the signal from the button, so we will assign it as an **INPUT**.

```
// initialize the digital led pin as an output.  
pinMode(led, OUTPUT);  
}
```

As we will be turning on the LED, so we will assign it as an **OUTPUT**.

Code Explantation

// the loop routine runs over and over again forever:

```
void loop(){
```

 val=digitalRead(button); //reading values from button
and saving it in the val variable.



This function helps us to read digital values from our “button” pin
and then save it in “val” integer.

```
if(val==1){
```

 check=~check; //if button is pressed then
 compliment the value of check.

```
delay(500); //wait for 0.5 seconds
```

```
}
```

```
if(check==1)
```

 blink(); // blink only if the value of check is one.

```
}
```



Here we have used a custom made function to define the
blinking process. Lets see how it works.

Code Explantation

```
void blink() {  
    digitalWrite(led, HIGH); // turn the LED on (HIGH is  
the voltage level)
```



This function tells to make the given pin(in this case our LED) as either **HIGH(1 or 5V)** or **LOW(0 or 0V)**.

```
delay(200); // wait for a second
```



This function gives us a delay in milli seconds i.e. if we have given 1000 then the delay is 1 sec and if 200 then 0.2 seconds and so on.

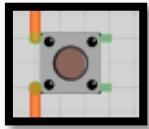
```
digitalWrite(led, LOW); // turn the LED off by making  
the voltage LOW
```

```
delay(200); // wait for a second
```

```
}
```

What's Happening?

Arduino is constantly checking the signal from the button(which is fed into **val**).



Whenever the button(**val = 1**) is pressed it compliments the value of **check** variable.

When the value of **check** becomes 1, our arduino starts the blinking process.

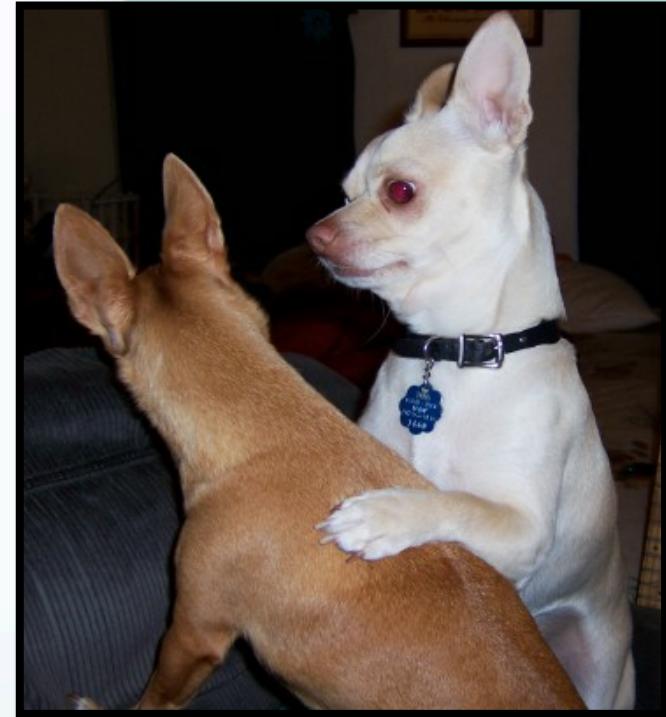


Congratulations,
You have just made your first Arduino
code.

Give yourself a pat
on the back. You
did great!

for any doubts mail me at
mail@sahilkhanna.org

or find me here ->
<https://facebook.com/sahilkhanna>



Aloha...