

# Anatomy of an Arduino Program

compile (use to check for errors before uploading)

upload to microcontroller

open Serial Monitor

(use to send and receive data through the USB cable connected to the microcontroller)

cite your sources!

include any libraries first

global variables declared here

runs once, when the program starts

repeats over and over

Check here for any upload errors

```

Blink | Arduino 1.0.1

Blink 5
/*
  Blink
  Turns on an LED on for one second, then off for one second, repeatedly.

  This example code is in the public domain.
  */

#include <SomeLibrary.h>

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

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

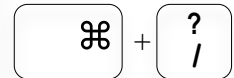
// the loop routine runs over and over again forever:
void loop()
{
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}

Done uploading.
Binary sketch size: 1,084 bytes (of a 30,720 byte maximum)
Binary sketch size: 1,084 bytes (of a 30,720 byte maximum)
avrdude: stk500_recv(): programmer is not responding

1 Arduino Duemilanove w/ ATmega328 on /dev/tty.usbmodemfa131
    
```

//Good commenting (descriptive, detailed, and plentiful) is your friend!

You can select a block of code and use command + / to "comment out" the entire block. Select it and command + / again to restore it.



are you using the correct board and serial port?

Pay attention to blocks of code within curly braces and corresponding levels of indentation!

Keeping your code neat will help you avoid syntax errors such as missing curly braces { }.

## Indentation

```

void loop()
{
  if(brightness > threshold)
  {
    for(int i=0;i<10;i++)
    {
      digitalWrite(led, LOW);
      delay(1000);
      digitalWrite(led, HIGH);
      delay(1000);
    }
  }
  else
  {
    digitalWrite(led, LOW);
  }
}
    
```

## Troubleshooting:

Upload errors?

- check board (for Botboarduino, should be "Duemilanove with ATmega 328")
- check serial port (name should contain "usbserial")

Compile errors?

- check for misplaced semicolons, curly braces, parentheses
- search the web for the error message + keyword "Arduino"

## Online Resources and Tutorials:

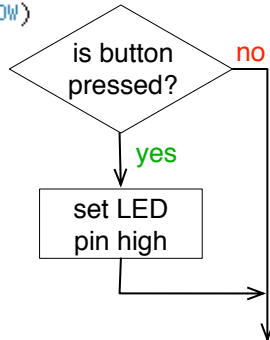
- arduino.cc
- lynxmotion.com
- learn.adafruit.com
- instructables.com
- learn.sparkfun.com
- makezine.com

# Control Structures

## conditions

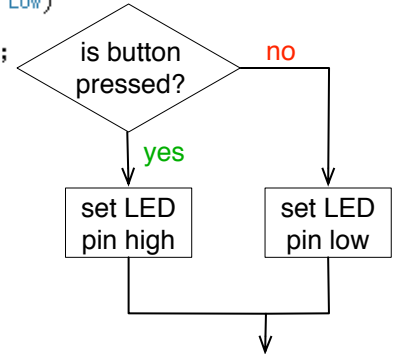
### if

```
if(digitalRead(buttonPin) == LOW)
{
  digitalWrite(ledPin, HIGH);
}
```



### if/else

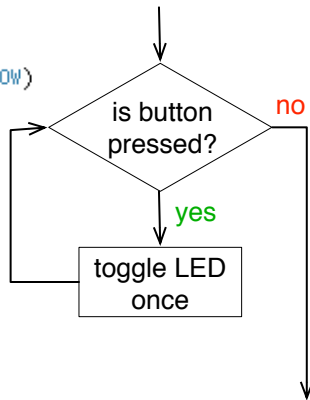
```
if(digitalRead(buttonPin) == LOW)
{
  digitalWrite(ledPin, HIGH);
}
else
{
  digitalWrite(ledPin, LOW);
}
```



## loops

### while

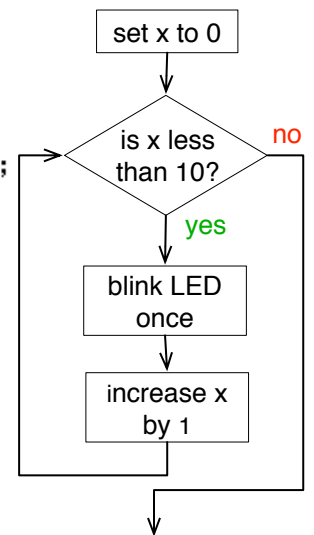
```
while(digitalRead(buttonPin) == LOW)
{
  digitalWrite(ledPin, HIGH);
  delay(500);
  digitalWrite(ledPin, LOW);
  delay(500);
}
```



### for

```
for(int x=0;x<10;x+=1)
{
  digitalWrite(ledPin, HIGH);
  delay(500);
  digitalWrite(ledPin, LOW);
  delay(500);
}
```

use when you know how many times the loop should run



## comparison operators

(from Arduino reference)

`x == y` (x is equal to y)  
`x != y` (x is not equal to y)  
`x < y` (x is less than y)  
`x > y` (x is greater than y)  
`x <= y` (x is less than or equal to y)  
`x >= y` (x is greater than or equal to y)

note! == is for comparison, = is for assignment!

## boolean operators

`&&` and  
`||` or  
`!` not

Use to combine comparisons

```
if(brightness > 0 || button == LOW)
```

"if value of brightness variable is greater than 0 OR value of button variable is LOW"