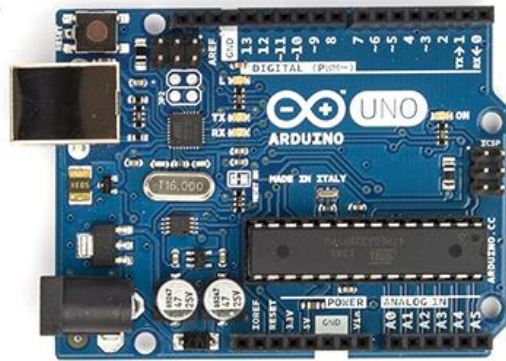


HTL | MÖSSINGERSTRASSE



# Arduino moves Spacecar

by Burkhard Grabner

Kapitel 1: Einführung



## 1.1 Beschreibung

Dieses Tutorial soll einen kurzen Einblick in die Arduino Programmierung liefern: Aufbau des Arduino sowie die Funktion der Benutzeroberfläche der Entwicklungsumgebung, Die folgenden Übungen sind soweit wie möglich selbstständig zu erledigen. Bei Problemen sollen die zuständigen Betreuer gerufen werden.

<b>Themenbereich</b>	Arduino moves Spacecar
<b>Zeitraumen:</b>	4-8 UE / Fachschule
<b>Zielgruppe:</b>	NMS & Fachschule HTL
<b>Schulstufe:</b>	ab der 3. Klasse
<b>Fach</b>	Informatik
<b>Lehr- und Lernziele:</b>	Kennenlernen des Arduino-Boards und dessen Programmierung
<b>Lehrplanbezug:</b>	Programmierung
<b>Materialien:</b>	Elektronische Bauteile, Tutorials, IDE
<b>Anmerkungen:</b>	Alle Bauteile können selbst besorgt werden – siehe Einkaufliste am Ende des Kapitels

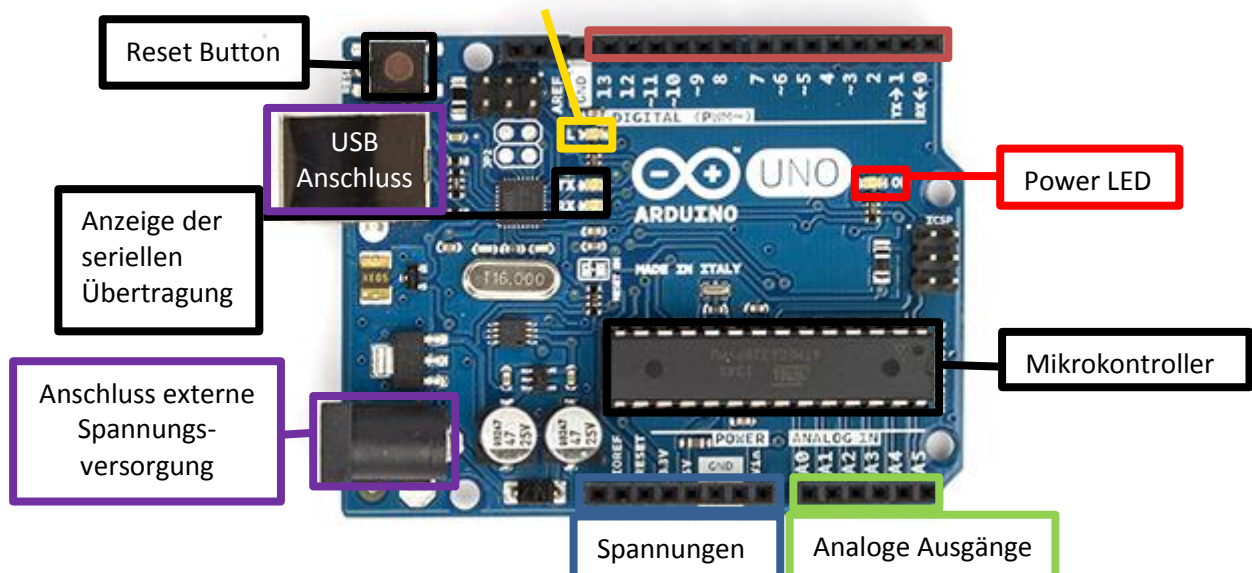
## Hardware

### 1.2 Aufbau des Arduino Boards

Die Arduino-Plattform ist eine aus Soft- und Hardware bestehende Physikalische-Computer-Plattform. Beide Komponenten sind im Sinne von Open Source quelloffen.

Die Hardware besteht aus einem einfachen I/O-Board mit einem Mikrocontroller und analogen und digitalen Ein- und Ausgängen. Die Programmierung selbst erfolgt in C bzw. C++ wobei technische Details vor den Anwendern weitgehend verborgen werden und umfangreiche Libraries und Beispiele die Programmierung vereinfachen.

Arduino kann verwendet werden, um eigenständige interaktive Objekte zu steuern oder um mit Softwareanwendungen auf Computern zu interagieren.



## Hardware

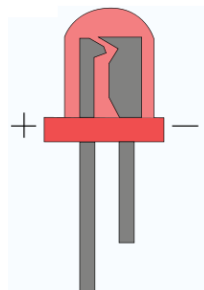
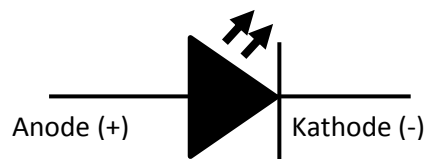
### 1.3 LED

LED ist die Abkürzung für „Licht Emittierende Diode“ und bezeichnet ein elektronisches Halbleiter-Bauelement.

Auf dem Arduino befindet sich an Pin 13 bereits eine eingebaute LED. Wenn man aber eine andere LED anschließt, muss diese einen Vorwiderstand besitzen um das Board nicht durch die entstehenden Spannungen zu schädigen.

**Schaltzeichen:**

**Bauteilrealansicht:**



### Widerstand

Ein Widerstand ist ein Bauelement, das in diesem Fall zur Reduzierung der Spannung verwendet wird.

**Berechnung:**

Berechne den Widerstand selbstständig!

**Schaltzeichen:**



**Bauteilrealansicht:**

**Farbcode:**

braun grün braun gold



Übung: Bestimmen Sie den Wert des Widerstandes:

## Hardware

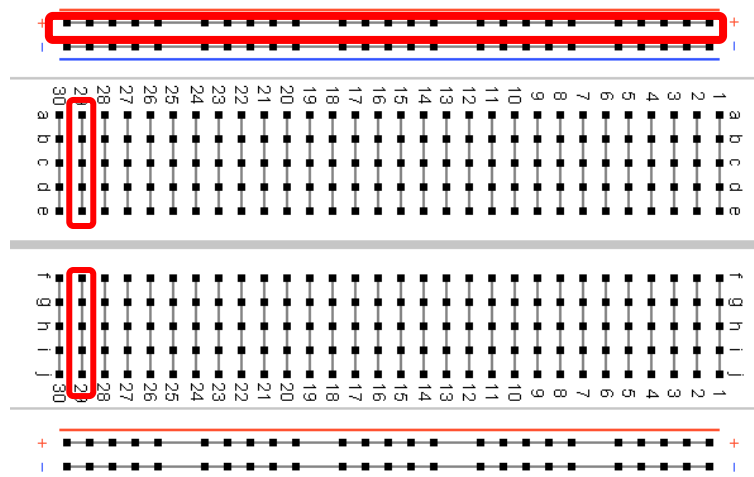
### 1.4 Steckboard

Ein Steckboard wird verwendet, um schnell Schaltungen aufzubauen und Messungen durchzuführen. Generell sind die Löcher eines Steckboards immer im gleichen Schema verbunden.

Die oberen zwei Reihen, die mit Plus(+) und Minus(-) gekennzeichnet sind, sind durchgehend miteinander verbunden (Achtung: die beiden Reihen sind nicht untereinander verbunden).

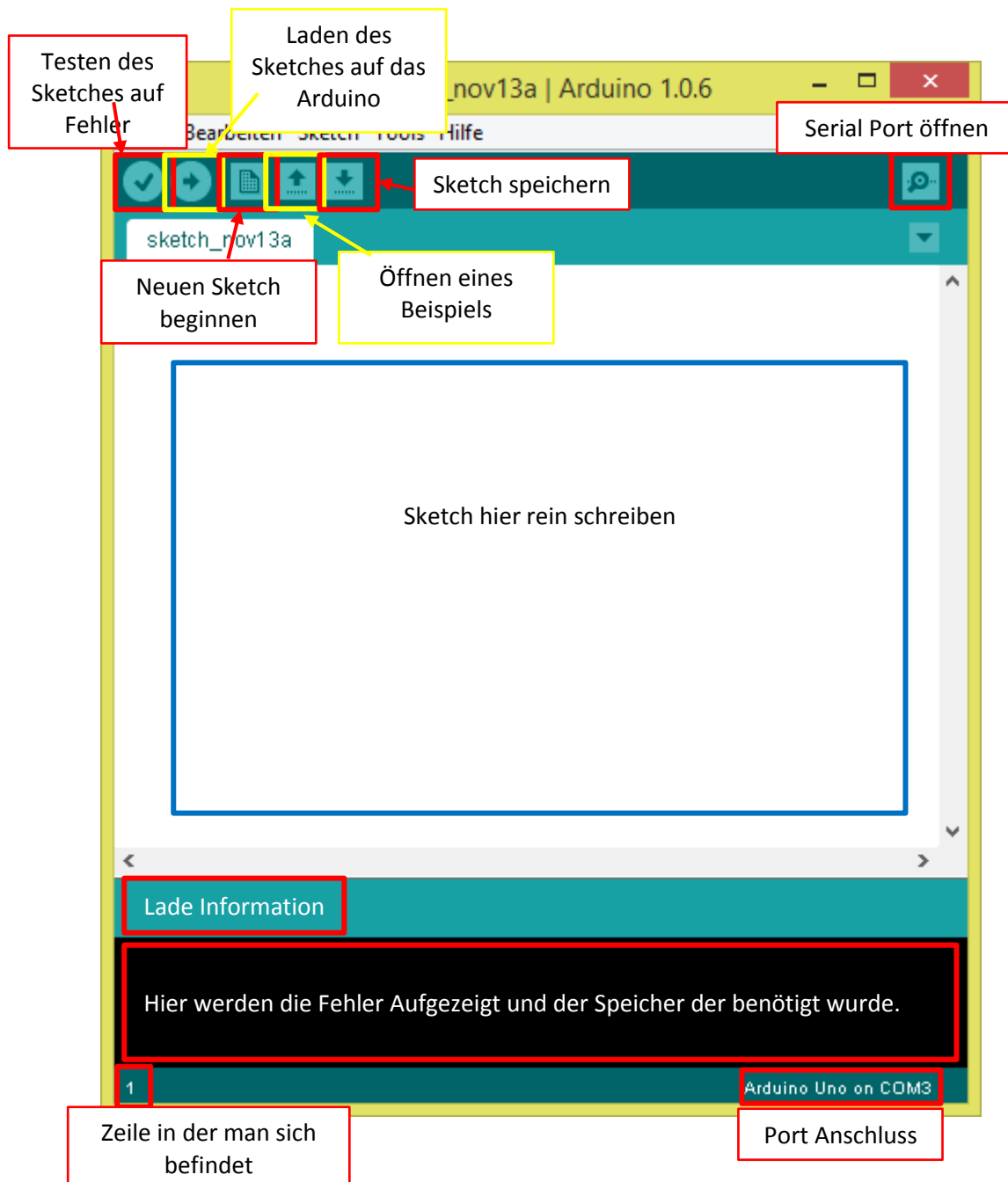
Die Reihen, die mit Buchstaben und Zahlen versehen sind, sind immer nach unten hin verbunden (Achtung: eine Verbindung von der oberen zur unteren Hälfte existiert nicht).

Die Verbindungen sind grau auf dem Steckboard angezeigt. Zur Veranschaulichung sind die Verbindungen umrahmt



## Software

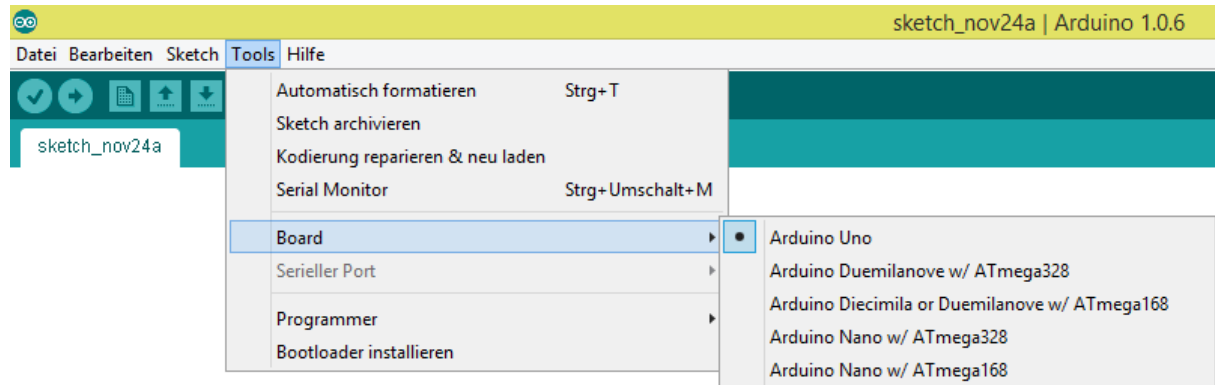
### 1.5 Erklärung der Programmieroberfläche



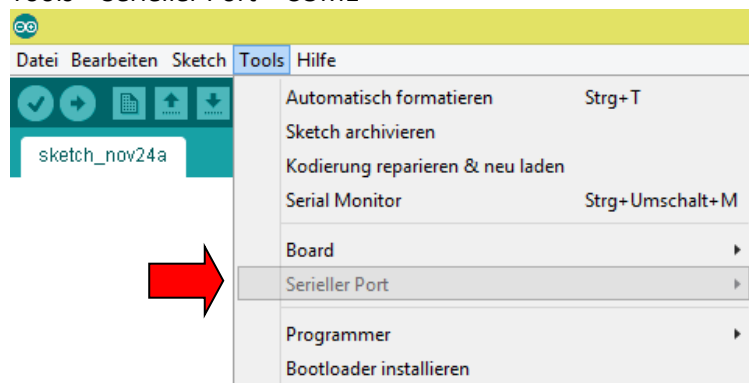
## Software

### 1.6 Konfiguration des Arduino-Boards und der COM Schnittstelle

Um festzustellen, ob das richtige Arduino-Board eingestellt ist, muss man unter:  
Tools > Board > Arduino Uno



Die COM Einstellungen befinden sich auch unter Tools:  
Tools > Serieller Port > COM1



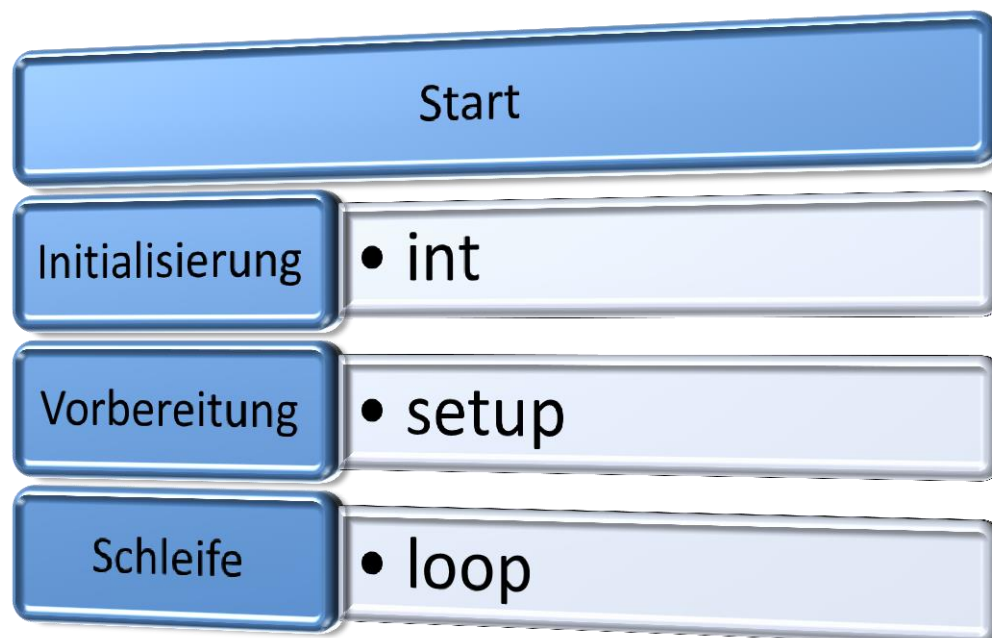
Die COM Einstellungen hängen davon ab, auf welcher Schnittstelle man den Arduino angesteckt hat.  
Wenn man sich nicht sicher ist, welcher es ist, sollte man einfach die verschiedenen ausprobieren.

## 1.7 Programmierung

Hier sind die wichtigsten Grundbefehle für die Programmierung aufgeführt.

Befehl	Erklärung
int	Legt den Wert einer Variable fest. Eine Variable kann eine LED sein, welche auf Pin 13 ihren Pluspol hat.
setup	Ist die Vorbereitung für die Programmierung. In dieser legt man z.B. fest, welcher Pin angesteuert wird. Der Setup Teil des Programmes läuft nur einmal durch. Wenn Setup noch einmal durchlaufen werden soll, muss man den Reset Knopf drücken, um von neuem das Programm zu starten.
loop	Ist die Ausführung der Vorbereitung und wird in einer Dauerschleife abgerufen.

Blockschaltbild





## Software

### LED soll im Sekundentakt blinken

#### 1.8 Übung 1a:

Seite für jeden Schüler und jede Schülerin ausdrucken

Zeichne den Schaltplan mit den nötigen Bauteilen in das folgende Bild ein und baue die Schaltung auf dem Übungsboard auf.

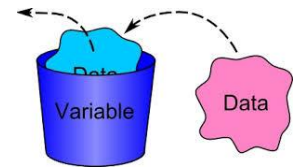


## Software - Übungen

### 1.9 Übung 1b:

Erstelle einen Programmablauf mit den Kärtchen in den grünen Kuverts

# Variable setzen



## Definition der Ports



## Digitaler PORT = Ausgang

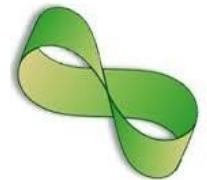


## Start des Programmes Endlosschleife



## Software - Übungen

# LED - Einschalten



# Verzögerung



# Endlosschleife Ende



## • Software – Übungen

### 1.10 Übung 1c:

Vergleichen Sie den Programmablauf mit Hilfe der Kärtchen in den orangenen Kuverts.

Hinweis: Da dies ein Memory-Spiel ist, sind die Kärtchen ohne Symbole nach oben vom Lehrer aufzulegen!

Schreiben Sie das nun erstellte Programm in die Entwicklungsumgebung am PC:

- Öffne einen neuen Sketch
- Schreibe den Code wo die LED im Sekundentakt blinken in die Entwicklungsumgebung  
normale LED im halben Sekundentakt blinkt
- Lade den Sketch auf den Arduino

### 1.11 Übung 1d:

Öffne in der IDE-Entwicklungsumgebung über **Datei – Beispiele – 01 Basic** das Beispiel **Blink** und vergleiche dieses Programm mit Deinem selbst geschriebenen LED-BLINKEN Programm.

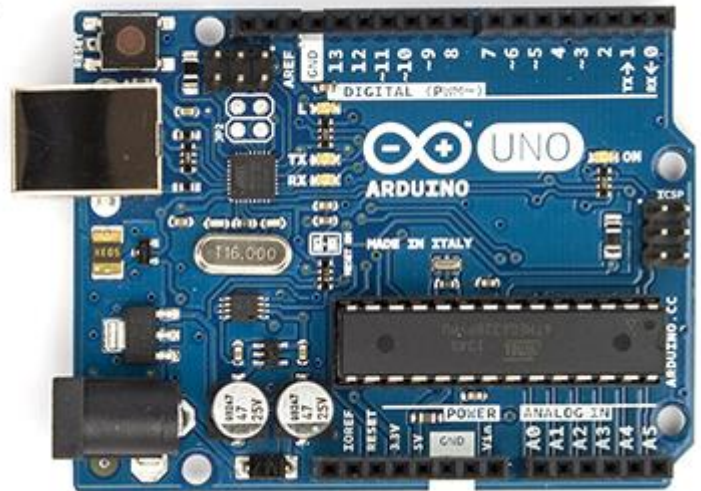
Beschreibe jede einzelne Programmzeile in Deinem Programm um genau zu dokumentieren was in jeder Zeile gemacht wird.

### 1.12 Übung 1e:

Eine Leuchtdiode ist über einen Taster Ein- und Auszuschalten.

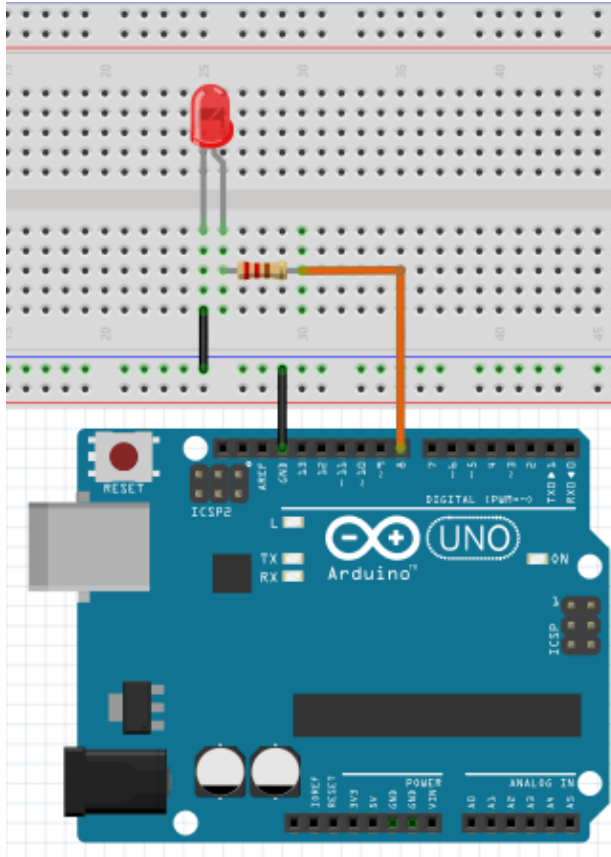
Öffne danach die Übungsdatei Datei/Beispiele/02.Digital/Button und erkläre in schriftlicher Form die einzelnen Programmschritte.

Jeder Schüler und jede Schülerin sollte diese Seite ausdrucken, da der Schaltplan mit Led, Taster und den nötigen Widerständen lt. den Programmanweisungen eingezeichnet werden muss.



- Software – Übungen - Lösungen

### 1.13 Übung 1a:



### 1.14 Übung 1b und 1c

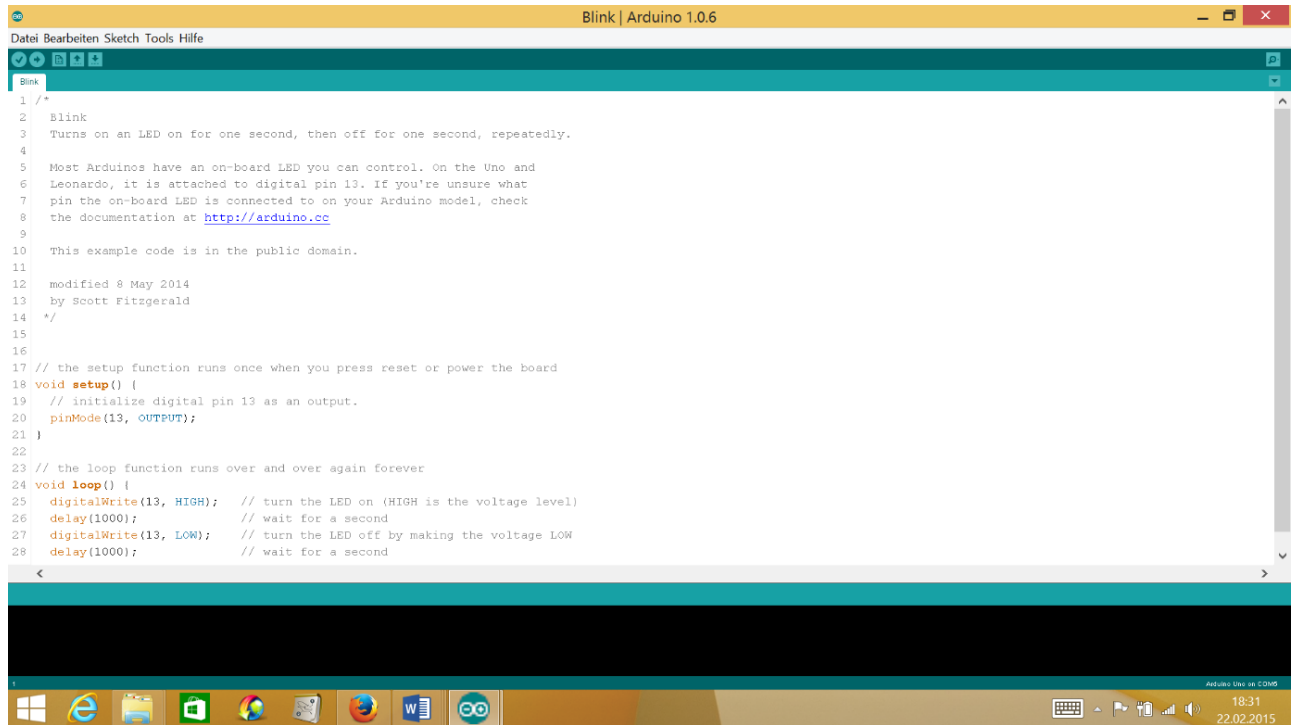
```
int led = 13;

void setup() {
  pinMode(led, OUTPUT);    // initialisiert den Pin 13 als Ausgang
}

void loop() {
  digitalWrite(led, HIGH); // schaltet die LED ein
  delay(1000);             // warten für eine Sekunde
  digitalWrite(led, LOW);  // schaltet die LED aus
  delay(1000);             // warten für eine Sekunde
}
```

## Software – Übungen - Lösungen

### 1.15 Lösung Übung 1d



```
1 /*
2  Blink
3  Turns on an LED on for one second, then off for one second, repeatedly.
4
5  Most Arduinos have an on-board LED you can control. On the Uno and
6  Leonardo, it is attached to digital pin 13. If you're unsure what
7  pin the on-board LED is connected to on your Arduino model, check
8  the documentation at http://arduino.cc
9
10 This example code is in the public domain.
11
12 modified 8 May 2014
13 by Scott Fitzgerald
14 */
15
16 // the setup function runs once when you press reset or power the board
17 void setup() {
18   // initialize digital pin 13 as an output.
19   pinMode(13, OUTPUT);
20 }
21
22 // the loop function runs over and over again forever
23 void loop() {
24   digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)
25   delay(1000); // wait for a second
26   digitalWrite(13, LOW); // turn the LED off by making the voltage LOW
27   delay(1000); // wait for a second
28 }
```

Welche Zeile fehlt in der obigen Programmstruktur?

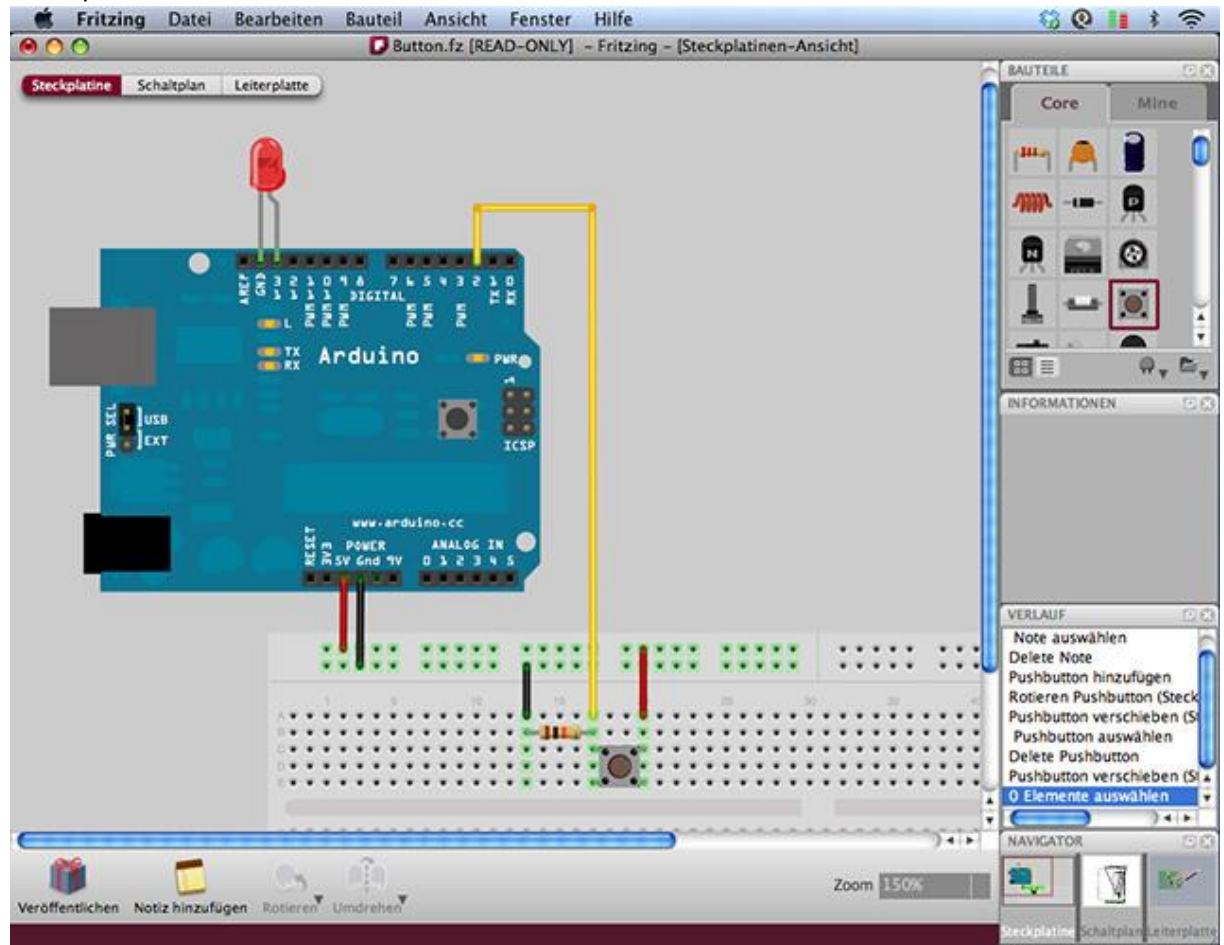
## • Software – Übungen - Lösungen

### 1.16 Lösung Übung 1e

Beschreibung der Programmzeilen:

<http://www.arduino-tutorial.de/2010/06/digital-in/>

Schaltplan:



#### Erklärung LED ohne Vorwiderstand an Pin 13:

Interner Vorwiderstand ist am Arduino bereits vorhanden, da es eine LED zur Kontrolle bereits gibt.

#### LED nie ohne Vorwiderstand betreiben!

#### Erklärung warum der Widerstand am Taster nötig ist:

Der Widerstand (1 – 10 kΩ) wird benötigt, um Spannungsschwankungen und Störsignale herauszufiltern. Man bezeichnet so einen Widerstand als Drop-Down-Widerstand. Er wird mit dem Digitalen Pin 2 und dem GND verbunden. Alle störenden Ströme werden somit über den Widerstand in den GND geführt. Ist der Taster nicht gedrückt, liegt am Pin 2 jetzt ein GND-Signal an (LOW), drückt man den Taster, so liegt ein 5V+ Signal an (HIGH).

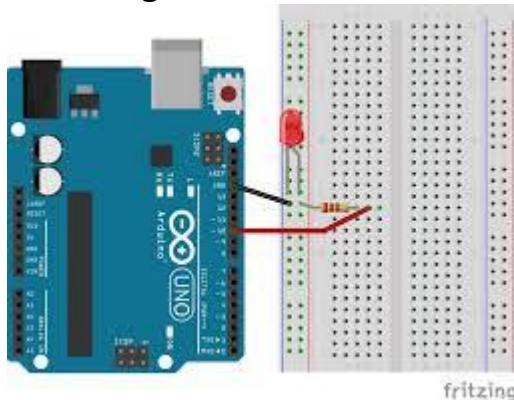


## • Software – Übungen - Lösungen

### 1.17 Lösung Übung 1f 1– Ampelschaltung - Vorbereitung

Wiederholung:  
LED blinkt im Sekundentakt

Schaltung:



Programmcode mit Erklärung:

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

// 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
}
```

Da es sich hier um eine LED handelt, wird der Pin als ein Ausgang bestimmt.

Die LED wird mit "HIGH" beschrieben. Sie wird eingeschaltet.

Dieser Befehl ist nur eine Zeitverzögerung. Für eine Sekunde passiert hier gar nichts mehr.

Die LED wird neu beschrieben mit "LOW". Sie wird ausgeschaltet.

Dieser Part wird im Programm nur einmal am Anfang durchlaufen.

Alles, was im "void loop" steht wird in einer Dauerschleife ausgeführt. Was heißt das? Eine LED wird angemacht, leuchtet eine Sekunde. Dann wird eine LED ausgemacht und wartet eine Sekunde. Das Programm ist hier fertig, jedoch durch die Dauerschleife fängt das Programm wieder an die LED einzuschalten. Das Blinken fängt von vorne an.

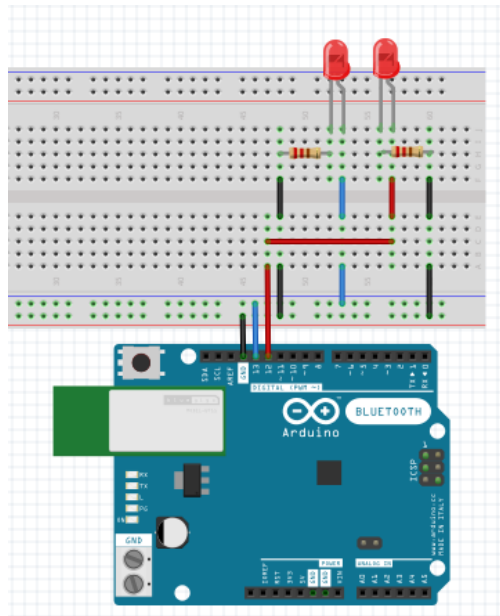
Nachdem der Code in das Arduino-Programm eingefügt wurde, auf überprüfen kicken und hochladen (upload).

## • Software – Übungen - Lösungen

### 1.18 Lösung Übung 1f 2– Ampelschaltung - Vorbereitung

Auf das Arduino Steckboard sind zwei Leds mit Vorwiderständen aufgebaut. Die Leds blinken abwechselnd!

#### Schaltung:



#### Programmcode:

```
#define LEDr 13
#define LEDg 12

void setup()
{
  pinMode(LEDr, OUTPUT);
  pinMode(LEDg, OUTPUT);
}

void loop()
{
  digitalWrite(LEDr,HIGH);
  delay(1000);
  digitalWrite(LEDr,LOW);
  delay(1000);
  digitalWrite(LEDg,HIGH);
  delay(1000);
  digitalWrite(LEDg,LOW);
  delay(1000);
}
```

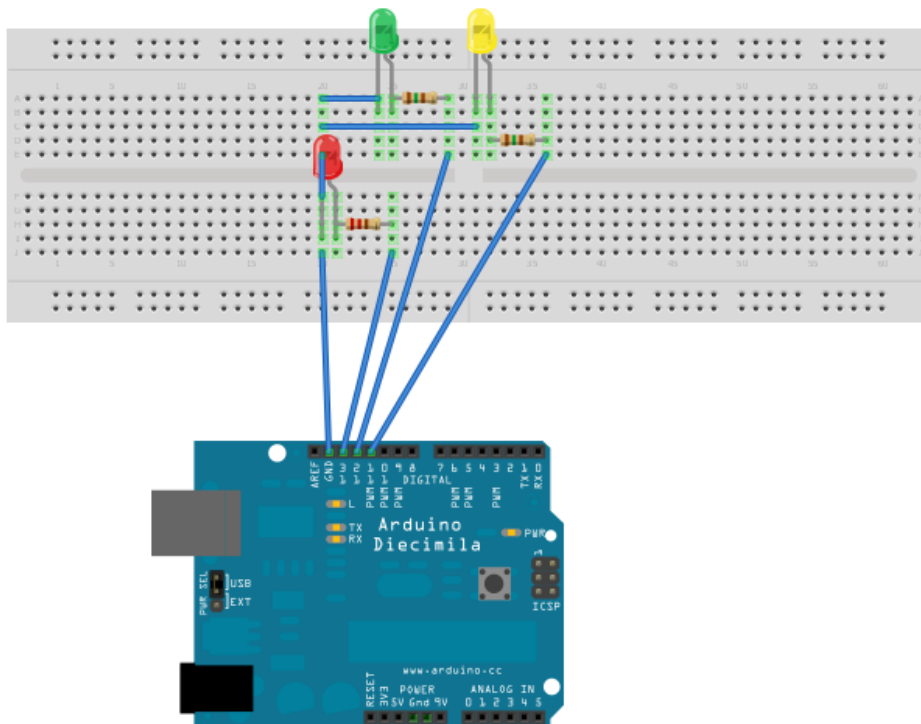
Nachdem der Code in das Arduino-Programm eingefügt wurde, auf überprüfen klicken, danach hochladen (upload).

- **Software – Übungen - Lösungen**

### 1.19 Lösung Übung 1f 3– Ampelschaltung - Vorbereitung

Erweiterung auf drei LEDs – Grundschiung für die Ampelsteuerung.  
Die LEDs blinken abwechselnd.

**Schaltung:**



### Programmcode:

```
#define LEDr 13
#define LEDg 12
#define LEDy 11

void setup()
{
  pinMode(LEDr, OUTPUT);
  pinMode(LEDg, OUTPUT);
  pinMode(LEDy, OUTPUT);
}

void loop()
{
  digitalWrite(LEDr,HIGH);
  delay(1000);
  digitalWrite(LEDr,LOW);
```

## Kapitel 1: Einführung

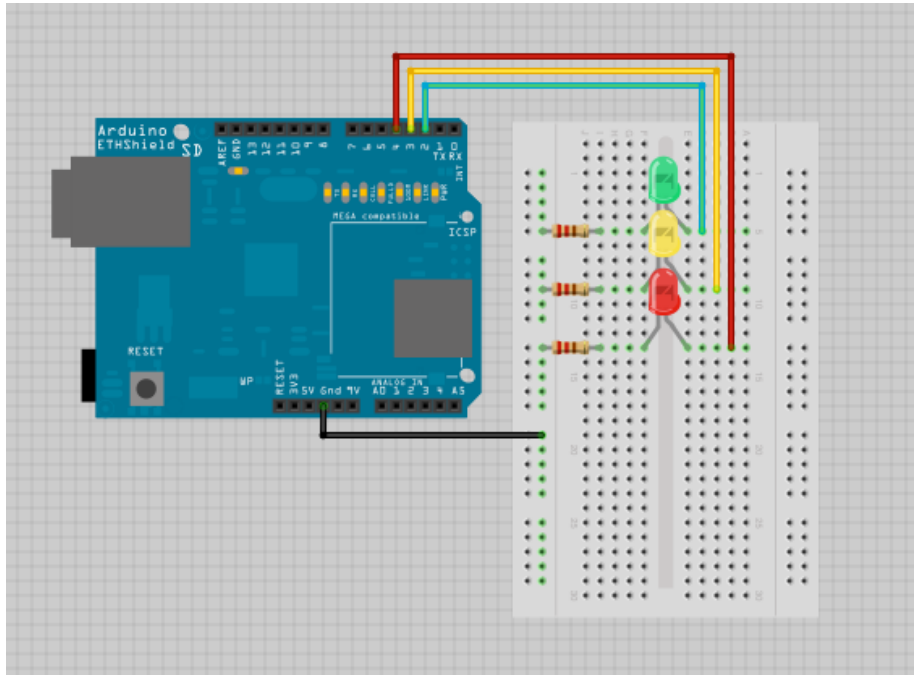


```
delay(1000);  
digitalWrite(LEDg,HIGH);  
delay(1000);  
digitalWrite(LEDg,LOW);  
delay(1000);  
digitalWrite(LEDy,HIGH);  
delay(1000);  
digitalWrite(LEDy,LOW);  
delay(1000);  
}
```

## • Software – Übungen - Lösungen

### 1.20 Lösung Übung 1f 4 – Ampelschaltung

Die LEDS blinken wie bei einer einfachen Ampelschaltung.



//Led auf die Steckplätze verteilen

```
int ledpinRot=13;
```

```
int ledpinGelb=12;
```

```
int ledpinGruen=11;
```

```
void setup() {
```

```
//Outputs definieren
```

```
pinMode(ledpinRot, OUTPUT);
```

```
pinMode(ledpinGelb, OUTPUT);
```

```
pinMode(ledpinGruen, OUTPUT);
```

```
}
```

```
//Ampelschaltung
```

```
void loop()
```

```
{
```

```
digitalWrite(ledpinRot,HIGH);//Rot und Gelb
```

```
delay(5000); //leuchten
```

```
digitalWrite(ledpinGelb,HIGH);//zusammen auf!
```

```
delay(2000);
```

```
digitalWrite(ledpinRot,LOW);
```

```
digitalWrite(ledpinGelb,LOW);
```

## Kapitel 1: Einführung



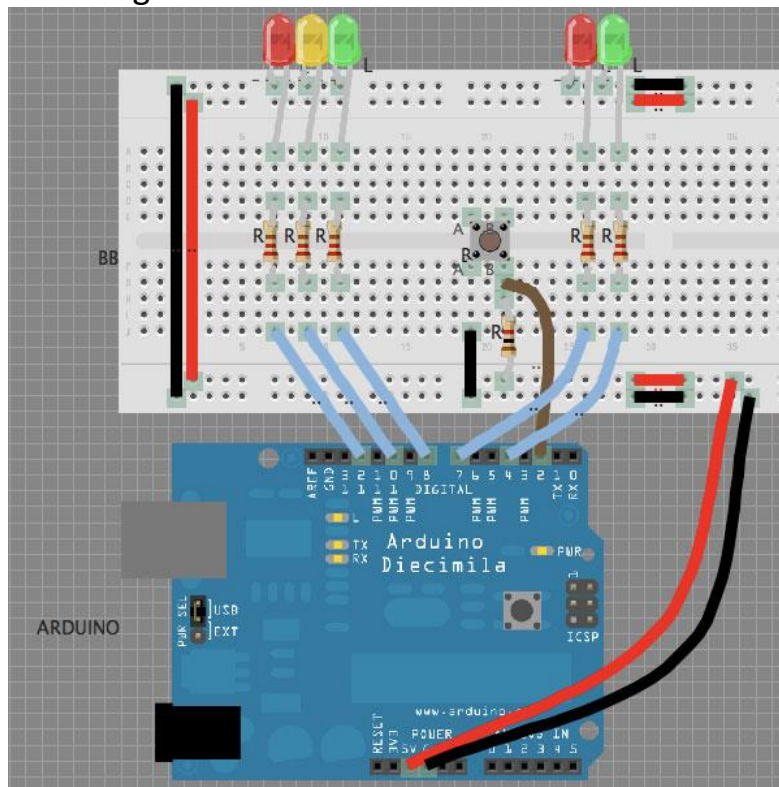
```
digitalWrite(ledpinGruen,HIGH);  
delay(5000);  
digitalWrite(ledpinGruen,LOW);  
digitalWrite(ledpinGelb,HIGH);  
delay(2000);  
digitalWrite(ledpinGelb,LOW);  
}
```

- **Software – Übungen - Lösungen**

### 1.21 Lösung Übung 1f 4– Ampelschaltung mit Fußgängerübergang, der mit einem Taster aktiviert wird

Am Arduino Steckboard gibt es nun fünf LEDs mit Vorwiderständen.

Schaltung:



Programmcode:

```
int autoRot = 12;    // Pin für die Autoampel Rot
int autoGelb = 10;   // Pin für die Autoampel Gelb
int autoGruen = 8;   // Pin für die Autoampel Grün
int fussRot = 7;     // Pin für die Fußgängerampel Rot
int fussGruen = 4;   // Pin für die Fußgängerampel Grün
int fussBtn = 2;     // Button für die Fußgänger
int val = 0; // Variable um den Zustand des Button zu speichern
```

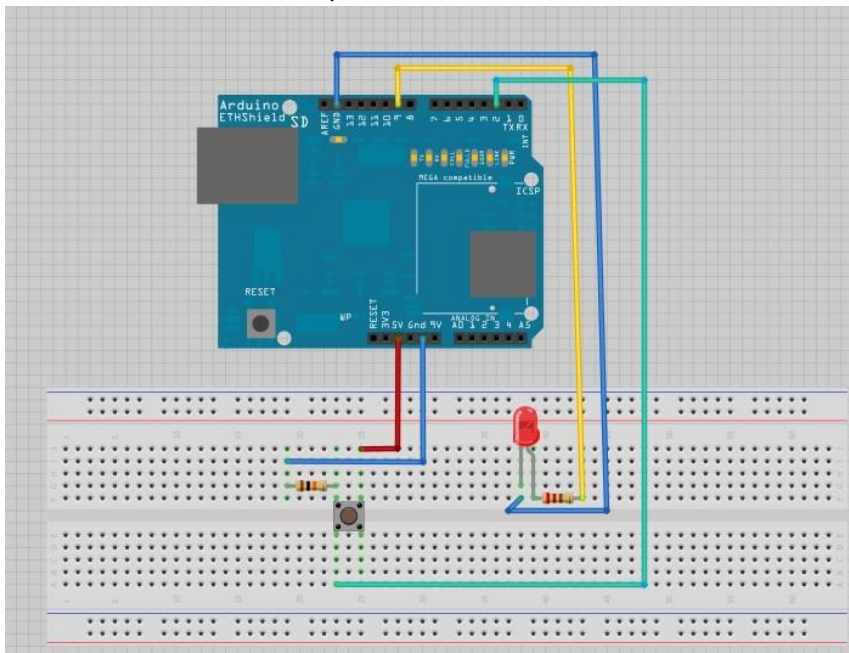
```
void setup() { // Funktion, die nur einmal durchlaufen wird
  pinMode(fussRot, OUTPUT); // Ampelpin ist ein Ausgang
  pinMode(fussGruen, OUTPUT); // Ampelpin ist ein Ausgang
  pinMode(autoRot, OUTPUT); // Ampelpin ist ein Ausgang
  pinMode(autoGelb, OUTPUT); // Ampelpin ist ein Ausgang
  pinMode(autoGruen, OUTPUT); // Ampelpin ist ein Ausgang
  pinMode(fussBtn, INPUT); // Buttonpin ist ein Eingang
}
```



```
void loop(){           // Methode – Endlosschleife
    digitalWrite(fussRot, HIGH); // Fußgängerampel auf rot
    digitalWrite(autoGruen, HIGH); // Autoampel auf grün
    val = digitalRead(fussBtn); //speichern des Buttonwertes
    // Der Button sendet wenn er nicht gedrückt ist ein HIGH-Signal. Wenn man ihn drückt,
    // wird ein LOW-Signal ausgegeben.
    if (val == LOW) {   // wenn der Buttonwert LOW ist dann:
        delay(2000);    // zwei Sekunden warten
        digitalWrite(autoGruen, LOW); // Autoampel Grün ausschalten
        digitalWrite(autoGelb, HIGH); // Autoampel Gelb einschalten
        delay(3000);    // drei Sekunden warten
        digitalWrite(autoGelb, LOW); // Autoampel Gelb abschalten
        digitalWrite(autoRot, HIGH); // Autoampel Rot einschalten
        delay(3000);    // drei Sekunden warten
        digitalWrite(fussGruen, HIGH); // Fußgängerampel Grün einschalten
        digitalWrite(fussRot, LOW); // Fußgängerampel Rot abschalten
        delay(10000);    // zehn Sekunden warten (OK, das müssen schnelle Fussgänger
        // sein.)
        digitalWrite(fussGruen, LOW); // Fußgängerampel Grün abschalten
        digitalWrite(fussRot, HIGH); // Fußgängerampel Rot einschalten
        delay(3000);    // drei Sekunden warten
        digitalWrite(autoGelb, HIGH); // Autoampel Gelb einschalten
        delay(3000);    // drei Sekunden warten
        digitalWrite(autoGelb, LOW); // Autoampel Gelb abschalten
        digitalWrite(autoRot, LOW); // Autoampel Rot abschalten
    }                  // Ende der if-Abfrage
}
```



Aufgabenstellung 6: Led mit einen Taster schalten:  
Tastendruck-> Led an, Tastendruck-> Led aus



Der Programmiercode:

```
const int buttonPin = 3;
const int ledPin = 13;
int merker=0;

int buttonState = 0;

void setup()
{
  pinMode(ledPin, OUTPUT);
  pinMode(buttonPin, INPUT);
  Serial.begin(9600);
}

void loop()
{
  buttonState = digitalRead(buttonPin);

  if ((buttonState == HIGH)&&(merker==0))
  {
    digitalWrite(ledPin, HIGH);
    merker=1;
  }

  if ((buttonState == LOW)&&(merker==1))
  {
    merker=2;
  }

  if ((buttonState == HIGH)&&(merker==2))
  {
  }
```


## Kapitel 1: Einführung



```
    digitalWrite(ledPin, LOW);  
    merker=3;  
}  
if ((buttonState == LOW)&&(merker==3))  
{  
    merker=0;  
}  
  
Serial.print("Merker= ");  
Serial.println(merker);  
}
```

## 2 Einkaufsliste

Komponente	Firma	Bestellung	Bestellnummer
<b>Arduino Uno</b>	Conrad Electronic GmbH & Co KG Durisolstraße 2 4600 Wels	Telefonisch bestellen: 050 - 20 40 40 <a href="http://www.conrad.at">http://www.conrad.at</a>	<b>191789 - 62</b>
<b>Arduino Motor Shield R3</b>	Conrad Electronic GmbH & Co KG Durisolstraße 2 4600 Wels	Telefonisch bestellen: 050 - 20 40 40 <a href="http://www.conrad.at">http://www.conrad.at</a>	<b>323477 - 62</b>
<b>Steckplatine EIC-801</b> Breadboard	Conrad Electronic GmbH & Co KG Durisolstraße 2 4600 Wels	Telefonisch bestellen: 050 - 20 40 40 <a href="http://www.conrad.at">http://www.conrad.at</a>	<b>526819</b>
<b>Arexx Roboterfahrgestell</b>	ELV Elektronik AG Maiburger Straße 29 - 36 26789 Leer	Telefonisch bestellen: 0662-624-084 ONLINE <a href="http://www.elv.at/">http://www.elv.at/</a>	<b>68-10 18 30</b>
<b>DAGU Roboterfahrgestell</b> <i>4WD Chassis kit</i>	Génération Robots 1 rue Théodore Blanc, Bâtiment L 33049 Bordeaux - France	☎ Tel: +33 556 393 705 <a href="http://www.generationrobots.com">http://www.generationrobots.com</a>	<b>A-000000-01017</b>
<b>Sharp GP2D12 Distanzsensor</b>	reichelt elektronik GmbH & Co. KG Elektronikring 1 26452 Sande Germany	<b>+49 (0)4422 955-333</b> <a href="http://www.reichelt.at/">http://www.reichelt.at/</a>	<b>GP2-1080K</b>
<b>Servomotor</b> Servo MEX55 HD Digital Micro modelleXXpert	Modellsport Schweighofer GmbH Wirtschaftspark 9, A-8530 Deutschlandsberg	03462/2541-100 <a href="http://www.der-schweighofer.at">http://www.der-schweighofer.at</a>	<b>100601</b>
<b>Motraxx Motor 3V-DC</b>	Conrad Electronic GmbH & Co KG Durisolstraße 2 4600 Wels	Telefonisch bestellen: 050 - 20 40 40 <a href="http://www.conrad.at">http://www.conrad.at</a>	<b>244422 - 62</b>
<b>Flexible Verbinder</b>	Neuhold Elektronik Griesplatz 1 8020 Graz	<b>0316-711245</b> <a href="mailto:office@neuhold-elektronik.at">office@neuhold-elektronik.at</a>  <a href="http://www.neuhold-elektronik.at">www.neuhold-elektronik.at</a>	<b>N8720</b> <b>N8721</b> <b>N8722</b> <b>N8723</b> <b>N4700</b>
<b>RGB-SMD-LED</b>	Conrad Electronic GmbH & Co KG Durisolstraße 2 4600 Wels	Telefonisch bestellen: 050 - 20 40 40 <a href="http://www.conrad.at">http://www.conrad.at</a>	<b>182985 - 62</b>

<b>Widerstand (150 <math>\Omega</math>)</b>	Conrad Electronic GmbH & Co KG Durisolstraße 2 4600 Wels	Telefonisch bestellen: 050 - 20 40 40 <a href="http://www.conrad.at">http://www.conrad.at</a>	<b>400157 - 62</b>
<b>RGB-LED</b>	Conrad Electronic GmbH & Co KG Durisolstraße 2 4600 Wels	Telefonisch bestellen: 050 - 20 40 40 <a href="http://www.conrad.at">http://www.conrad.at</a>	<b>1050465 - 62</b>
<b>LED (gelb)</b>	Conrad Electronic GmbH & Co KG Durisolstraße 2 4600 Wels	Telefonisch bestellen: 050 - 20 40 40 <a href="http://www.conrad.at">http://www.conrad.at</a>	<b>1050454 - 62</b>
<b>LED (grün)</b>	Conrad Electronic GmbH & Co KG Durisolstraße 2 4600 Wels	Telefonisch bestellen: 050 - 20 40 40 <a href="http://www.conrad.at">http://www.conrad.at</a>	<b>1050456 - 62</b>
<b>LED (rot)</b>	Conrad Electronic GmbH & Co KG Durisolstraße 2 4600 Wels	Telefonisch bestellen: 050 - 20 40 40 <a href="http://www.conrad.at">http://www.conrad.at</a>	<b>1050453 - 62</b>
<b>Chassis-Aufbau</b> Plattform für die Auflage der Komponenten	HTL-Mössingerstraße	<b>f@ctory</b>	
<b>LUX Sortierkästchen</b> Zwillingbox	OBI Markt Klagenfurt St. Jakoberstraße 2, 9029 Klagenfurt	Tel.: 0463/336 94 <b>www.ob.at</b>	<b>6181218</b>
<b>Bauteilkästchen</b> Sortierkästchen	OBI Markt Klagenfurt St. Jakoberstraße 2, 9029 Klagenfurt	Tel.: 0463/336 94 <b>www.ob.at</b>	<b>6181143</b>
<b>USB A-B Kabel</b>	Media Markt Völkermarkter Str. 140, 9020 Klagenfurt am Wörthersee	<a href="http://www.mediamarkt.at">www.mediamarkt.at</a> Telefon: 0463 382110	<b>3350</b>