

Arduino

Eine Einführung

Uwe Ziegenhagen

16. August 2011

- ▶ lebe und arbeite seit Oktober 2008 in Köln
- ▶ habe BWL & Statistik studiert, wenngleich mit IT-Fokus
- ▶ ⇒ „Elektronik funktioniert mit Rauch!“
- ▶ Blog unter www.uweziegenhagen.de, dort liegen auch die \LaTeX -Folien

Was ist „Arduino“?

Allgemeines

Hardware

Software

Mein erstes Programm

Mehr Spaß mit LEDs

Vernetztes. . .

LCD Display anschließen – DOGM

Mehr Beispiele

Mehr Informationen

Das Arduino Projekt

- ▶ Arduino = Physical-Computing-Plattform
- ▶ besteht aus Hard- und Software (beides Open Source)
- ▶ Hardware = einfache I/O-Boards mit Mikrocontroller und analogen/digitalen Ein- und Ausgängen
- ▶ Entwicklungsumgebung beruht auf Processing (Java-Dialekt) und Wiring (C-Dialekt)

Geschichte beginnt 2005 in Ivrea, Italien

- ▶ Massimo Banzi & David Cuartielles vom IDII (Interaction Design Institute Ivrea)
- ▶ Projekt, um Studenten einfaches Prototyping mit Mikrocontrollern zu ermöglichen
- ▶ Arduino = italienischer König, Name einer Bar

Es gibt verschiedene Arten von Arduinos:

- ▶ [Duemilanove/Uno](#), der Standard-Arduino
- ▶ [LilyPad](#), zum Auf- und Einnähen in Kleidung
- ▶ [ArduinoBT](#), mit eingebautem Bluetooth-Stack
- ▶ [Mega/Mega 2560](#), mit mehr I/O Ports
- ▶ [Mini](#), kleinste Version des Arduino
- ▶ diverse Clone: Boarduino, Nano, Arduino Pro, etc.
- ▶ Netduino, Gadgeteer \Rightarrow Microsoft .Net

Wir konzentrieren uns auf den Duemilanove/Uno!

Arduino Duemilanove/Uno

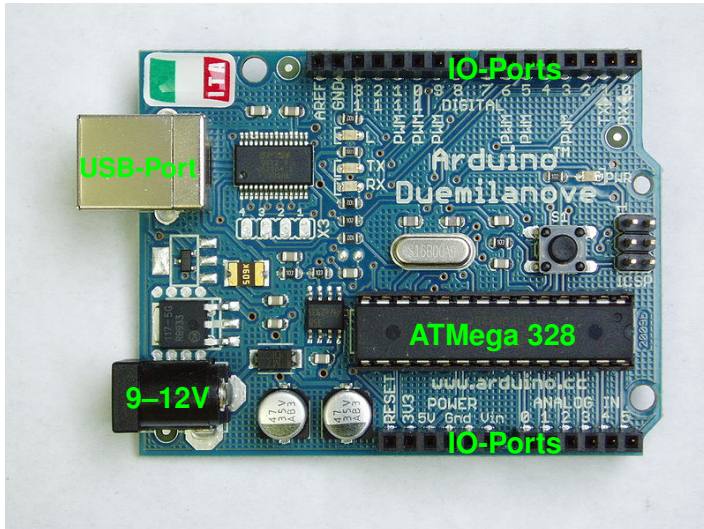


Abbildung: Arduino Duemilanove Quelle: Wikipedia

Arduino Duemilanove/Uno

- ▶ ATmega328 Mikrocontroller
- ▶ 32KB Flash (davon 0,5KB für Bootloader)
- ▶ 2KB RAM, 1 KB EEPROM
- ▶ 16 MHz Takt
- ▶ 14 digitale I/O-Pins
- ▶ davon 6 als PWM nutzbar
- ▶ 6 analoge Eingänge (10 Bit)
- ▶ On-Board USB-Schnittstelle (mit Atmega8U2 als Seriell-Wandler)
- ▶ 5 V Betriebsspannung, Speisung über USB oder Spannungsregler (7–12V)

- ▶ Shields = Platinen, die direkt auf den Arduino gesteckt werden können
- ▶ erweitern Arduino um viele interessante Möglichkeiten
- ▶ Liste der Shield sehr lang: shields.com
- ▶ für Einsteiger gut geeignet: Danger Shield, Proto Shield, Ethernet Shield (Netzwerk!)

Danger Shield

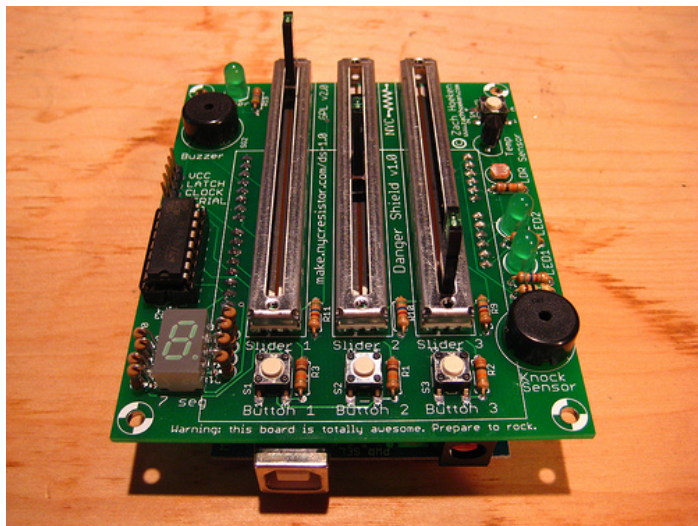


Abbildung: Das Danger Shield Quelle: www.zachhoeken.com

Proto Shield

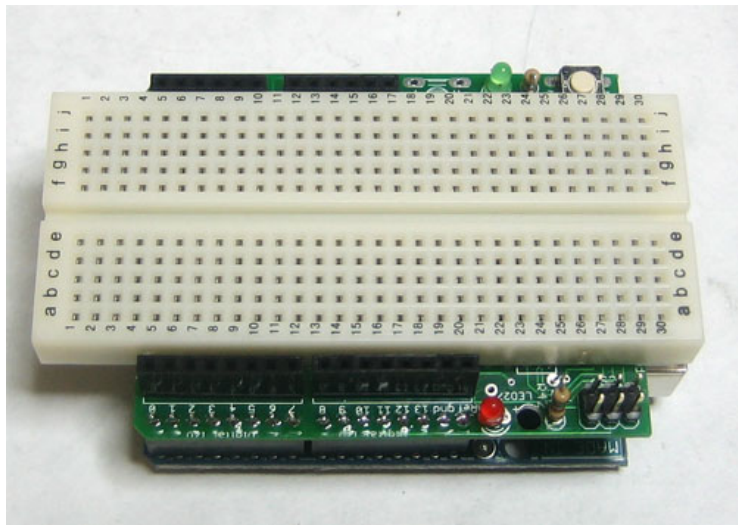


Abbildung: Das Proto Shield Quelle: www.ladyada.net

Ethernet Shield



Abbildung: Das Ethernet Shield mit MicroSD Quelle: www.watterott.com

Arduino Software & Fritzing

- ▶ erhältlich für Windows, Linux, Mac
- ▶ aktuell momentan Version 0022
- ▶ benötigt installierte JRE
- ▶ Mac und Windows Version bringen Compiler mit
- ▶ Linux benötigt GCC mit einigen AVR Bibliotheken

am besten auch gleich „Fritzing“ installieren

- ▶ Projekt der FH Potsdam, auf Spenden angewiesen
- ▶ grafische Design-Umgebung
- ▶ kann auch Schaltplan und Leiterplatte erzeugen

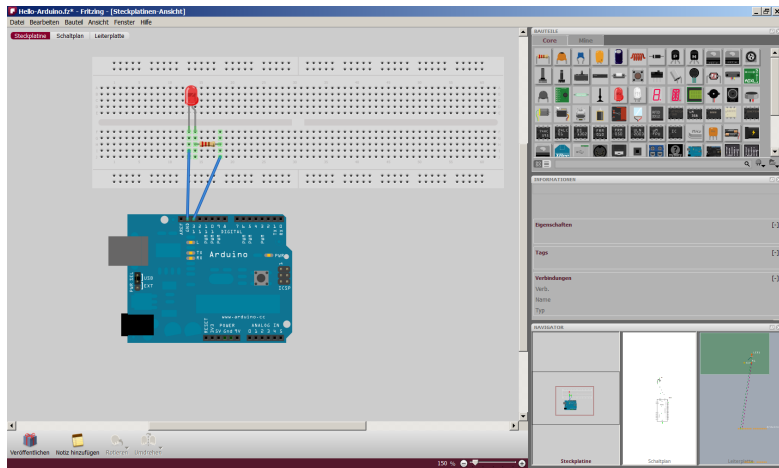


Abbildung: Screenshot Fritzing

Aufgabe: LED blinken lassen

- ▶ LEDs haben Kathode (+) und Anode (-, längeres Beinchen)
- ▶ nicht direkt anschließen, Vorwiderstand nutzen
- ▶ Widerstandsrechner im Internet
- ▶ 20 mA bei $5 \text{ V} = 220 \Omega$

Verkabelung

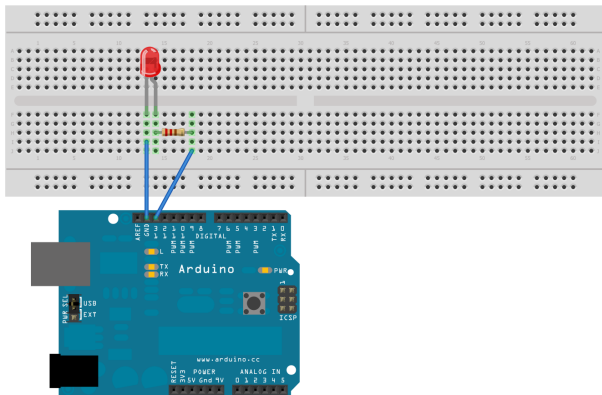


Abbildung: Arduino Duemilanove Quelle: Wikipedia

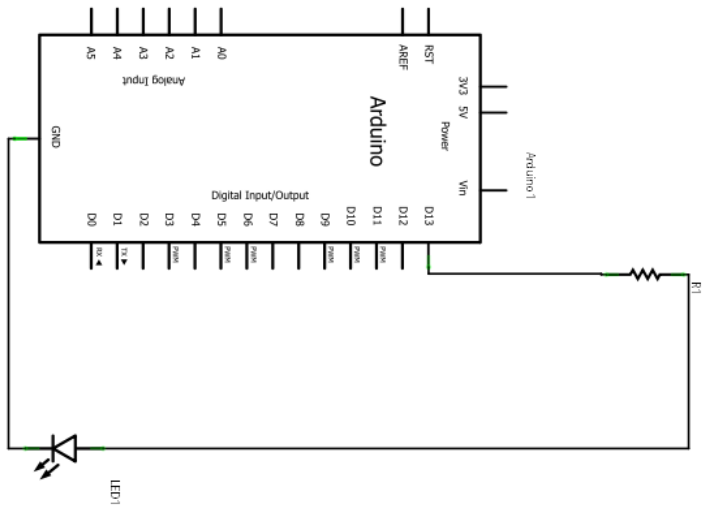


Abbildung: Screenshot Fritzing

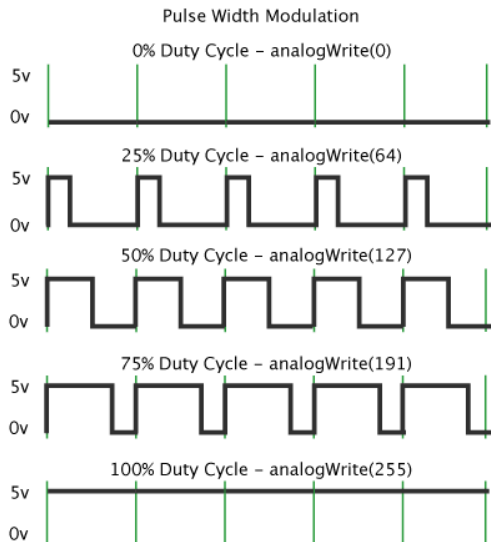
Programmcode

Siehe Examples ⇒ Basic ⇒ Blink.pde

```
/*  
  Blink  
  Turns on an LED on for one second, then off for one second,  
  repeatedly.  
  This example code is in the public domain.  
*/  
  
void setup() {  
  // initialize the digital pin as an output.  
  // Pin 13 has an LED connected on most Arduino boards:  
  pinMode(13, OUTPUT);  
}  
  
void loop() {  
  digitalWrite(13, HIGH); // set the LED on  
  delay(1000);           // wait for a second  
  digitalWrite(13, LOW); // set the LED off  
  delay(1000);           // wait for a second  
}
```

LEDs dimmen mit PWM

- ▶ PWM = Pulsweiten-Modulation



LEDs dimmen mit PWM

Siehe Examples ⇒ Analogue ⇒ Fading.pde

```
/*
  This example shows how to fade an LED on pin 9
  using the analogWrite() function.
  */
int brightness = 0;    // how bright the LED is
int fadeAmount = 5;    // how many points to fade the LED by

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

void loop() {
  analogWrite(9, brightness);
  // change the brightness for next time through the loop:
  brightness = brightness + fadeAmount;
  // reverse the direction of the fading at the ends of the fade:
  if (brightness == 0 || brightness == 255) {
    fadeAmount = -fadeAmount ;
  }
  // wait for 30 milliseconds to see the dimming effect
  delay(30);
}
```

Das LoL-Shield

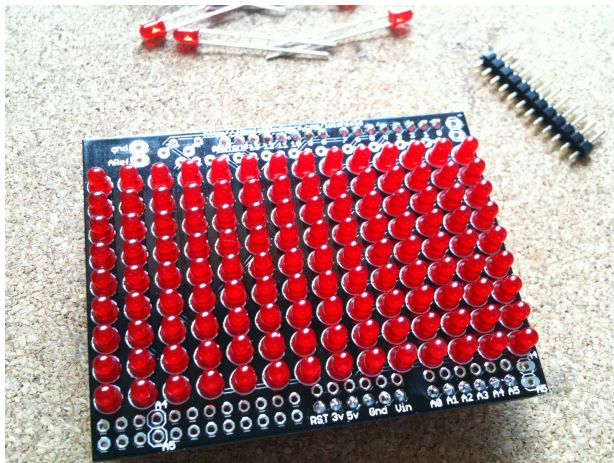


Abbildung: LoLshield, Quelle www.makethisdothat.com

Arduino Webserver I

Siehe Examples ⇒ Ethernet ⇒ Webserver.pde

```
/*  
A simple web server that shows the value of the analog input pins.  
using an Arduino Wiznet Ethernet shield.  
* Ethernet shield attached to pins 10, 11, 12, 13  
* Analog inputs attached to pins A0 through A5 (optional)  
*/  
#include <SPI.h>  
#include <Ethernet.h>  
  
// Enter a MAC address and IP address for your controller below.  

```

Arduino Webserver II

```
void loop()
{
  // listen for incoming clients
  Client client = server.available();
  if (client) {
    // an http request ends with a blank line
    boolean currentLineIsBlank = true;
    while (client.connected()) {
      if (client.available()) {
        char c = client.read();
        // if you've gotten to the end of the line
        // (received a newline character)
        // and the line is blank, the http request
        // has ended, so you can send a reply
        if (c == '\n' && currentLineIsBlank) {
          // send a standard http response header
          client.println("HTTP/1.1_200_OK");
          client.println("Content-Type:_text/html");
          client.println();
        }
      }
    }
  }
}
```

Arduino Webserver III

```
// output the value of each analog input pin
for (int analogChannel = 0; analogChannel < 6;
     analogChannel++) {
  client.print("analog_input_");
  client.print(analogChannel);
  client.print("_is_");
  client.print(analogRead(analogChannel));
  client.println("<br_>");
}
break;
}
if (c == '\n') {
  // you're starting a new line
  currentLineIsBlank = true;
}
else if (c != '\r') {
  // you've gotten a character on the current line
  currentLineIsBlank = false;
}
}
}
delay(1); // give the web browser time to receive the data
client.stop(); // close the connection
}}
```


LEDs über das Internet steuern

- ▶ benötigt Ethernet Shield (vorzugsweise mit MicroSD Slot)
- ▶ Code von [aus dem Internet](#)¹
- ▶ kann zwei digitale Ausgänge steuern

¹URL darf ich nicht angeben

Arduino Webserver 1.0 by Poldi

Ausgänge schalten:

Ausgang 3	<input type="button" value="einschalten"/>	<input type="button" value="ausschalten"/>	OFF
Ausgang 4	<input type="button" value="einschalten"/>	<input type="button" value="ausschalten"/>	OFF

Abbildung: Screenshot „Poldis Webserver“

per LC-Display lassen sich einfach und schnell Informationen ausgeben

- ▶ ideal: Displays kompatibel mit HD44780
- ▶ ich nutze DOGM Displays für knapp 11 Euro
- ▶ Bibliothek: <http://code.google.com/p/doglcd/>

⇒ <http://www.reichelt.de/?ACTION=3;ARTICLE=77826;PROVID=2402>

Schaltplan

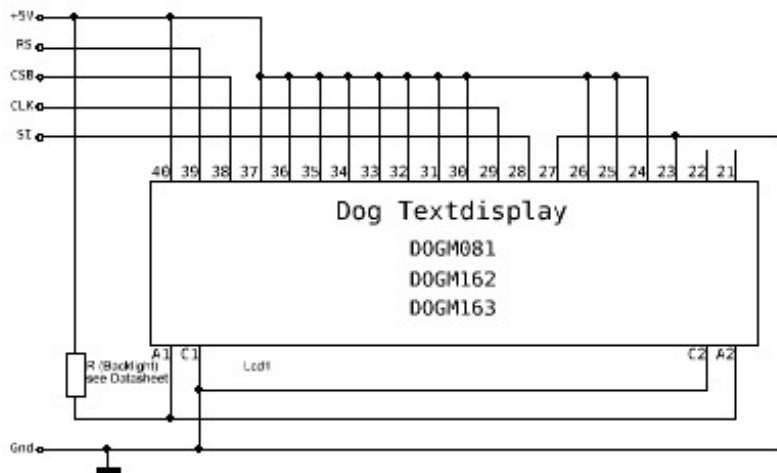


Abbildung: Schaltplan

DOGM Programmcode

```
/* basiert auf DogLcd Library - Hello World
We assume the following pins are connected:
* LCD SI pin to digital pin 2
* LCD CLK pin to digital pin 3
* LCD RS pin to digital pin 4
* LCD CSB pin to digital pin 5
http://www.arduino.cc/en/Tutorial/LiquidCrystal
*/
```

```
#include <DogLcd.h>
```

```
DogLcd lcd(2, 3, 4, 5);
```

```
void setup() {
  lcd.begin(DOG_LCD_M081, 0x20);
  lcd.print("Hallo!");
  delay(500);
  lcd.clear(); lcd.noCursor();
}
```

```
void loop() {
  lcd.setCursor(0, 0);
  lcd.print(millis()/1000);
}
```

„Ein 14-jähriger Schüler zeigt der Chilenischen Regierung, wie man mit einfachsten Mitteln ein funktionierendes Erdbeben-Warnsystem errichten kann. Sebastian Alegria beschaffte sich einen Erdbeben-Detektor für Haushalte, tauschte dessen integrierte Elektronik mit einem Arduino-Mikrocontroller aus und verband das Ganze mit seinem Webserver. Sekunden, bevor ein Erdbeben droht, wird eine Warnung per Twitter verschickt.“

Quelle: <http://kurier.at/techno/3924051.php>

„Die AEG-Waschmaschine meldet über eine LED, ob die Wäsche fertig gewaschen wurde. Wenn die Maschine jedoch im Keller steht, ist das schwer zu überwachen. Ein Bastler hat nun dieses Signal abgegriffen und kann in seinem LAN nachsehen, ob die Waschmaschine ihre Arbeit beendet hat.“

Quelle: <http://www.golem.de/1108/85541.html>

„Das ‚Bluetooth Morse Code Keyboard‘ von Zunkworks ist mit der freien Prototyping-Plattform Arduino entwickelt worden. Es übermittelt per Bluetooth die Tastenkommandos an den Rechner, an den es gekoppelt ist. Vorher werden die eingegebenen Morsesignale noch in normale Tastaturbefehle zurückübersetzt, damit jeder Rechner, der ein Bluetooth-HID-Interface unterstützt, damit funktioniert.“

Quelle: <http://www.golem.de/1108/85465.html>

„Die PLOTS-Forscher nutzen Teile, die regulär im Handel erhältlich sind, wie die Teammitglieder Jae-Ok Lee und Byeongwon Ha erklären. Der Chemikaliensensor MQ 135 und eine farbige LED werden von einem Arduino-Microcontroller-Board gesteuert. All das hängt wiederum samt Akkusatz an einem iRobot Roomba Staubsauger, den die Wissenschaftler gebraucht für 50 Dollar erworben.“

Quelle: <http://www.heise.de/newsticker/meldung/Roomba-als-Raumluftsensor-1282241.html>

WASP: Drohne zum Entern von WiFi und GSM-Netzen

„Die Flugzeugelektronik basiert auf dem Ardupilot, einem auf einem Arduino Mega basierendem Autopilotensystem. Eine HD-Kamera, die an der Drohne befestigt ist, kann zudem Bilder vom überflogenen Gebiet aufnehmen.“

Quelle: <http://www.pro-linux.de/news/1/17328/wasp-drohne-zum-entern-von-wifi-und-gsm-netzen.html>

Bezugsquellen

- ▶ <http://www.arduino.cc>
- ▶ <http://www.watterott.com>
- ▶ <http://www.elmicro.com>
- ▶ <http://www.segor.de>

- ▶ *Arduino: Praxiseinstieg* von Thomas Brühlmann
- ▶ *Praxisbuch Arduino - Mikrocontroller-Programmierung mit Arduino und Freeduino* von Ulli Sommer
- ▶ *Arduino - Physical Computing für Bastler, Designer und Geeks* von Odendahl, Finn, Wenger
- ▶ *Getting Started with Arduino* von Massimo Banzi
- ▶ *Practical Arduino: Cool Projects for Open Source Hardware* von Ozer und Blemings