Arduino

Messen, Steuern, Basteln

Informatik zum Anfassen

Ivo Blöchliger

Programm

- Einführungsvortrag
- Hard- und Software vorbereiten, erste Tests
- Pause
- Grundlegende Programme und Schaltungen
- Mittagessen
- Weitere Projekte, Tips und Tricks
- Ausblick, Schluss

Zu meiner Wenigkeit

- Dr. dipl. ing. Math. EPFL
- Elektronik seit 2 Jahren, Arduino seit 1 Jahr
- Vielen Dank für's Kommen!

Zum heutigen Kurs

- “Ivo, mach doch was!”
  - Viel Spass mit Arduino im letzten Jahr ;-)
- Einsatz im GLF Informatik?
- Ziele
  - Spass am Gerät: For Fun and Profit
  - Vertrauen bilden: “Ich kann das!”
  - Ideen für coole Projekte austauschen

Any sufficiently advanced technology is indistinguishable from magic.

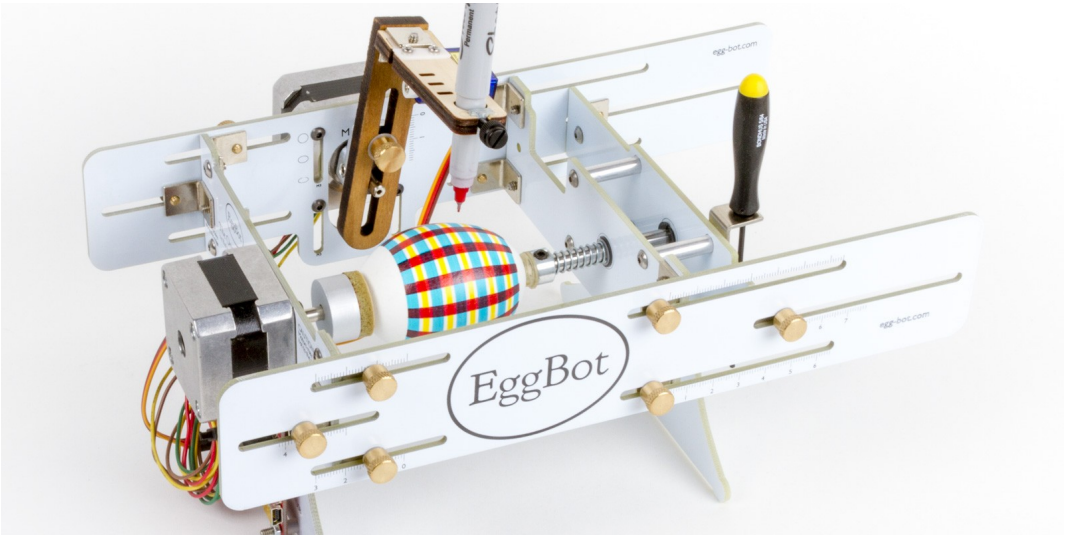
Arthur C. Clarke

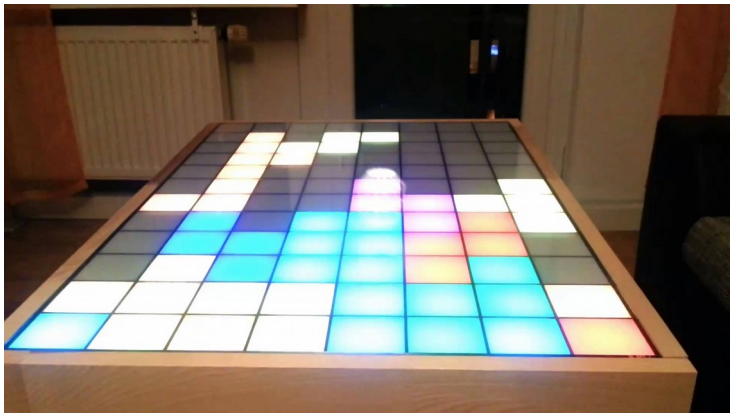
Realisierte Projekte

- Interaktion mit Präsentationsbildschirm
- Sanduhr
- Schlaf-Ampel für meine Kinder
- Flackernde Laterne
- Sprechendes Buchstabenpuzzle
- Anwendung Potenzfunktionen: Gammakorrektur

Projektideen

- Eggbot
- Bügelperlensortierer
- Roboterarm / -kran
- CNC-Maschinen
- Selbstbalancierender Roboter
- Tetris-Tisch (WS2812B)





Suchtpotential

- Aliexpress macht süchtig ;-)
  - Post startet Entzugsprogramm ab 2018?

- Das Netz ist voll von spannenden Projektideen
  - Der Alltag auch!

Challenges

- Information
  - Tutorials, Datasheets, Kollegen
- Elektronik
  - Gibt ICs für (fast) alles
- Informatik / Mathematik
  - Gibt Libraries für (fast) alles
- Mechanik
  - Löten, 3D-Drucker, Lasercutter → Sandkasten

Arduino, Geschichte

- Platform für Design-Studenten (2004)
  - Doktorarbeit, IDE von Processing (Doktorväter)
- 2005: Erste Version von Arduino
- OpenSource (Hard- und Software)
  - heute Massenware

Open Source

- Jeder darf
- Jeder kann
  - Hersteller wie Nutzer!
- Viele machen
  - Nachfrage von Bastler → Angebot in China!
- Viele erzählen davon (YouTube, Blogs, etc.)

Was kann ein Arduino?

- Kein OS → Echtzeit, nur (m)ein Programm.
- 17 GPIO, 5V, 6 davon auch analog (input).
  - 2 pin für seriellen port (hardware, bis 2Mbaud).
  - 20mA pro Input/Output
  - 16MHz (1-4 ticks pro Operation)
  - 32kB Flash, 2k RAM, 1k EEPROM
  - Stromversorgung
    - IN: 7-12V extern oder USB 5V
    - OUT: 5V (max 500 mA) und 3.3V (max 50mA).

Arduino vs. Raspberry Pi

- Kein OS → keine SD
- 5V GPIO, 20mA
- Bootzeit: 10ms??
- Echtzeitfähig
- USB-Anschluss
- CHF 5.- (China)
- Netz als Modul
- Kleine Displays

- Linux, SD-Karte
- 3.3V GPIO, ??mA
- Bootzeit: 10s??
- Kernelmodule?
- Schirm/Tastatur/Maus
- CHF 50+10+10+...
- Komplexe Server
- Multimedia

Boards

- Uno, Nano (Einsteiger)
- Mega (mehr Ports und Power)
- Yun (Ethernet, Linux), für IOT

- Nicht Arduino
  - ESP8266 (10 fache Leistung, integriertes WLAN)

    Das nächste grosse Ding?

Anwendung im Unterricht

- Macht Spass
- Einsicht in die Digitaltechnik
- Low-Level Robotics
  - Wie schwierig ist es, 10cm nach vorne zu fahren?
- Was heisst “messen”?
  - Sensoren, Messfehler, Daten aufzeichnen, verarbeiten

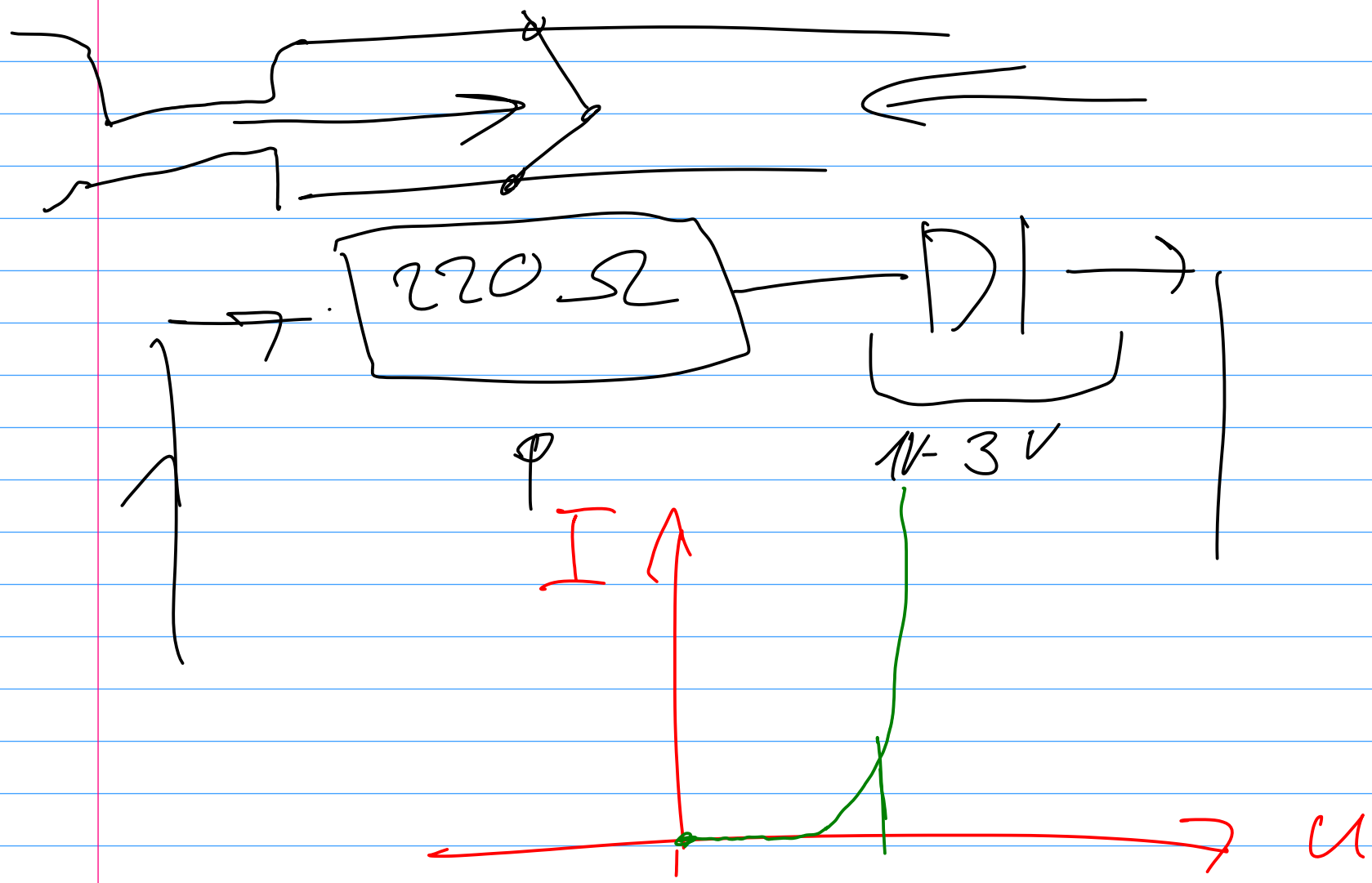
Lets Fetz!

- Installation der Entwicklungsumgebung
  https://www.arduino.cc/en/Main/Software
- Windows 7,8: Installation der CH340 Treiber?
- Windows 10: Evtl. Deaktivierung AV?
- Erster Sketch: In der IDE

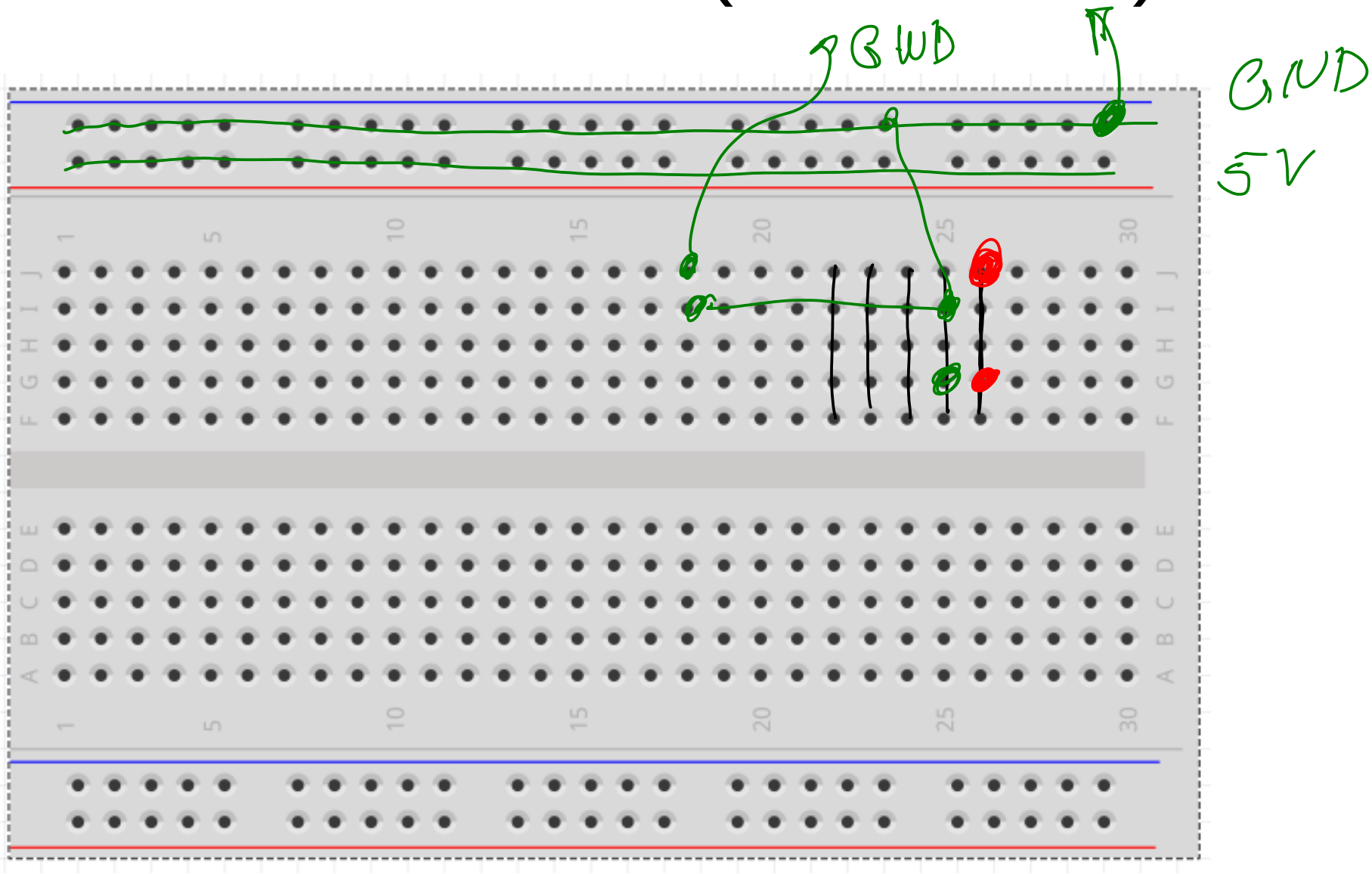
  File → Examples → 01. Basics → Blink

http://fginfo.ksbg.ch/arduino

http://dokuwiki.blöchliger.com/


220 Ω
1V - 3V
I
U

Breadboard (Steckbrett)



roboboter | Arduino 1.6.13

File Edit Sketch Tools Help

sketch_feb25a §

```
#define MYLED 3

void setup() {
  pinMode(MYLED, OUTPUT);
}
int d = 10;
void loop() {
  digitalWrite(MYLED, HIGH);
  delay(50); d
  digitalWrite(MYLED, LOW);
  delay(50); 100-d
}
```

→ meinef();

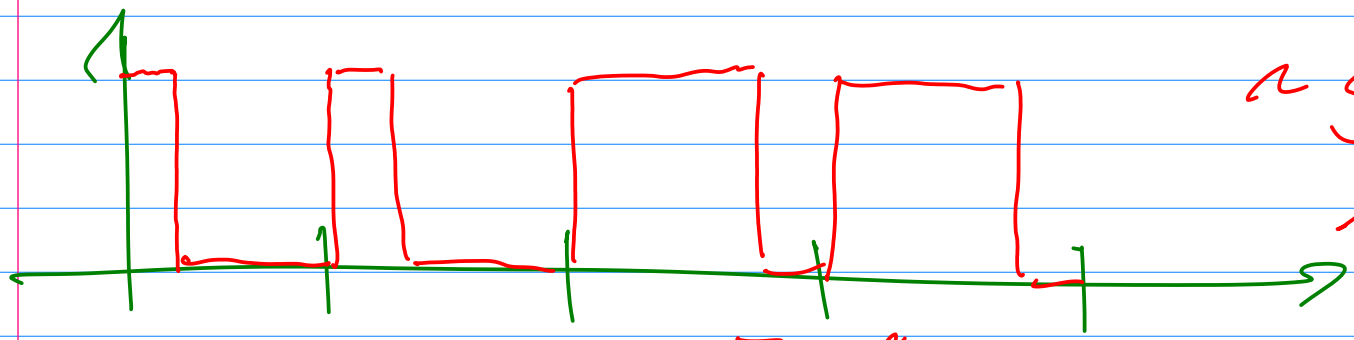
```
void meinef() {
  ⋮
}
```

delayMicroseconds(...)

Done Saving.

12 Arduino/Genuino Uno on /dev/ttyUSB5

PWM ~ 3, 5, 6, 9, 10, 11



~ 500 Hz
1 kHz

25% 75%

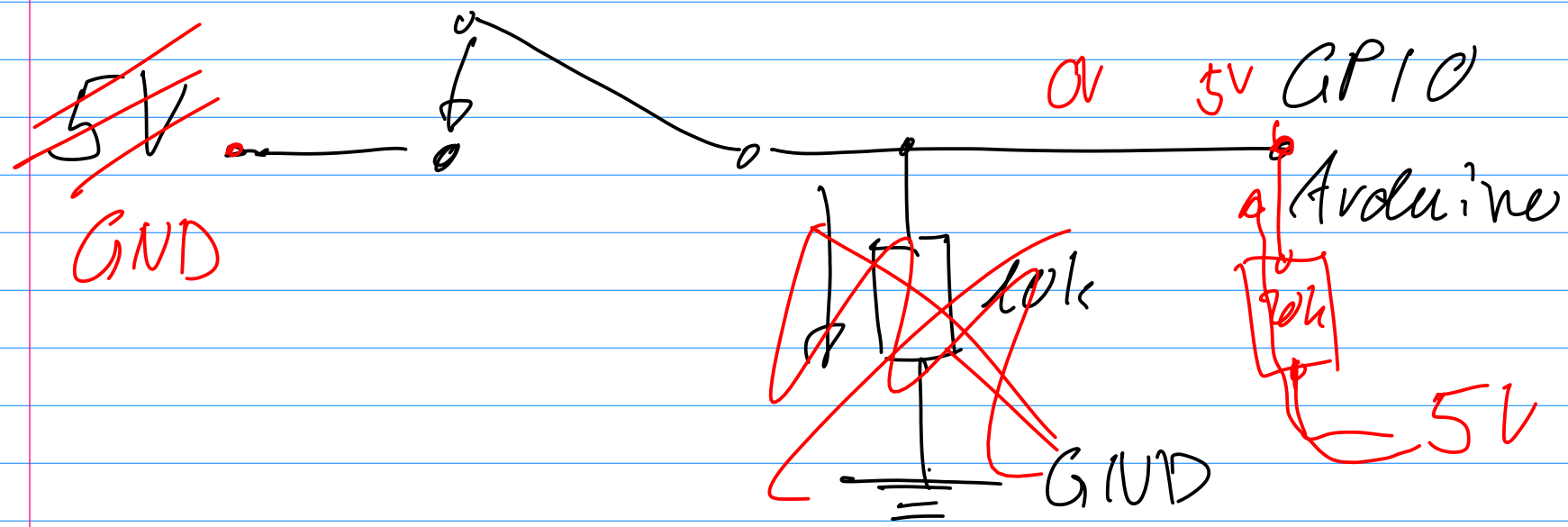
Duty-Cycle

analogWrite

0-255 25% → 63

Pins A0 - A5

analogRead(A0) → Wert 0-1023


5V
GND
10k
GND
0V
5V
GPIO
Arduino
10k
5V

Xournal - Vortrag.pdf.xoj

lehrkrae | New Tab | Arduino | I bought | CH341SE | Windows | lehrkrae | Gammal

Secure | https://fginfo.ksbg.ch/dokuwiki/doku.php?id=lehrkraefte:blc:math:formi:button-breadboard

```
void setup() {
  pinMode(2, INPUT_PULLUP);
  Serial.begin(115200);
  // Funktion aufrufen, wenn immer der Zustand ändert.
  attachInterrupt(digitalPinToInterrupt(2), click, CHANGE);
}

#define NUM 20
// volatile, damit im Interrupt verfügbar
volatile int pos = 0;
volatile long us[NUM];

// Interrupt Methode
void click() {
  if (pos<NUM) {
    us[pos++]=micros();
  }
}

void loop() {
  if (pos==NUM) {
     for (int i=0; i<NUM-1; i++) {
        Serial.print(us[i+1]-us[i]);
        Serial.print(" ");
     }
     Serial.println();
     pos = 0;
  }
}
```

CDM_v2.12.0....exe

Show all

Ausblick, Schluss

ICs, Boards und Libraries

- Für fast alles!
  - Sensoren
  - 7-Segment Displays
  - Port-Expander
  - Motorensteuerung
  - Funkverbindung

Tutorials, Videos, Instructables

- Für jedes Niveau
  - Und in jeder Qualität :-/

- Im Zweifelsfall: Datasheet lesen!

Standalone

- Stromversorgung via USB
  - z.B. Ladegerät
- Batteriebetrieb: Board schluckt ca. 40 mA
  - Entweder 5 V reguliert
  - Oder 7 V bis 12 V unreguliert
- Alternative: ATmega328P-PU alleine
  - 1mA (oder einige µA im Schlafmodus)
- ATTiny (8-Beiner), 5 IO-Ports

Ausblick

- Folgekurs in einem Jahr?
- Gegenseitige Vorstellung von Projekten
  - Realisiert, geplant oder einfach Ideen
- Unterrichtsideen und -erfahrungen

Zusatz-Slides

GPIO Pins (inkl. Analog-pins!)

- General Purpose Input/Output
- 4 Zustände:
  - OUTPUT
    - HIGH (5V, liefert bis zu 40mA)
    - LOW (0V, schluckt bis zu 40mA)
  - INPUT (nicht auf pin 13)
    - Es fliesst “kein” Strom, high impedance (100MΩ)
    - INPUT_PULLUP. Via ca. 20KΩ auf 5V verbunden

Memory

- Flash-Speicher: 32kB, nicht-flüchtig
  - Für Daten und Programmcode
- RAM: 2kB, flüchtig
  - Für temporäre Daten
- EEPROM: 1kB, nicht-flüchtig
  - Für sich ändernde Daten, die RESET überleben
  - **Achtung**: Begrenzte Anzahl Schreibzugriffe (10^5)
    - Keine Schreibzugriffe in einem Loop!

analogRead()

- 6 Pins (A0-A5)
  - A4,A5 für I2C (Wire) Protokoll reserviert.
- analogRead dauert ca 0.2ms
- Nur 1 pin kann auf einmal konvertieren
- Werte von 0 (GND) bis 1023 (Vref).
  - Vref ist normal 5V, kann tiefer gesetzt werden

Stromversorgung

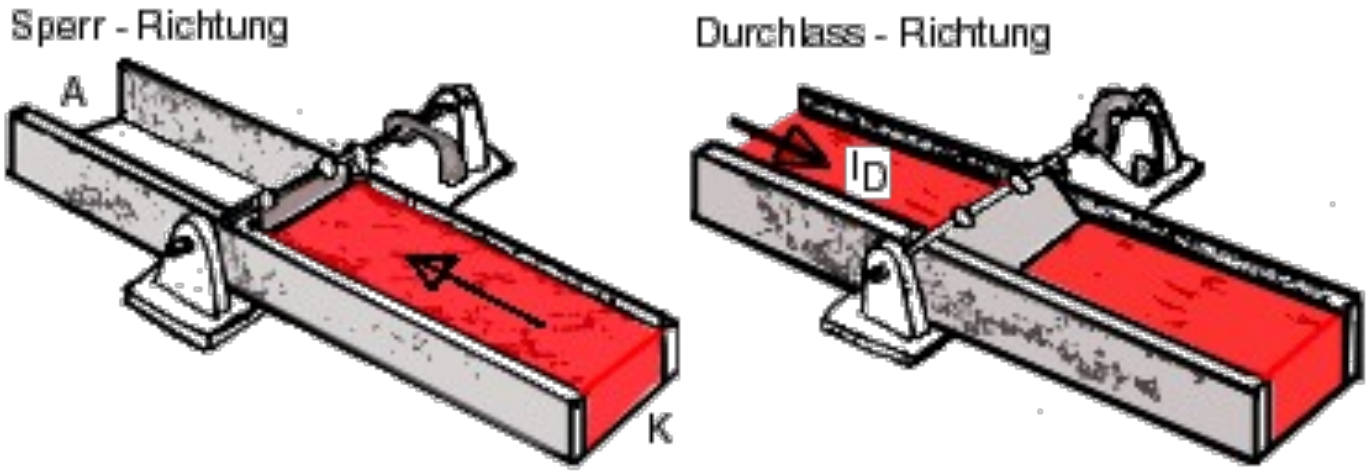
- Arduino liefert 3.3V (max 50mA).
- Versorgung durch USB (max. 500mA)
- Externe Versorgung (7V bis 12V)
  - Onboard Konverter nach 5V (max 500mA)
- Gute 5V Versorgung kann bei 5V pin angeschlossen werden (passiert bei USB)
- Batterie: 7V oder mehr, oder Konverter → 5V

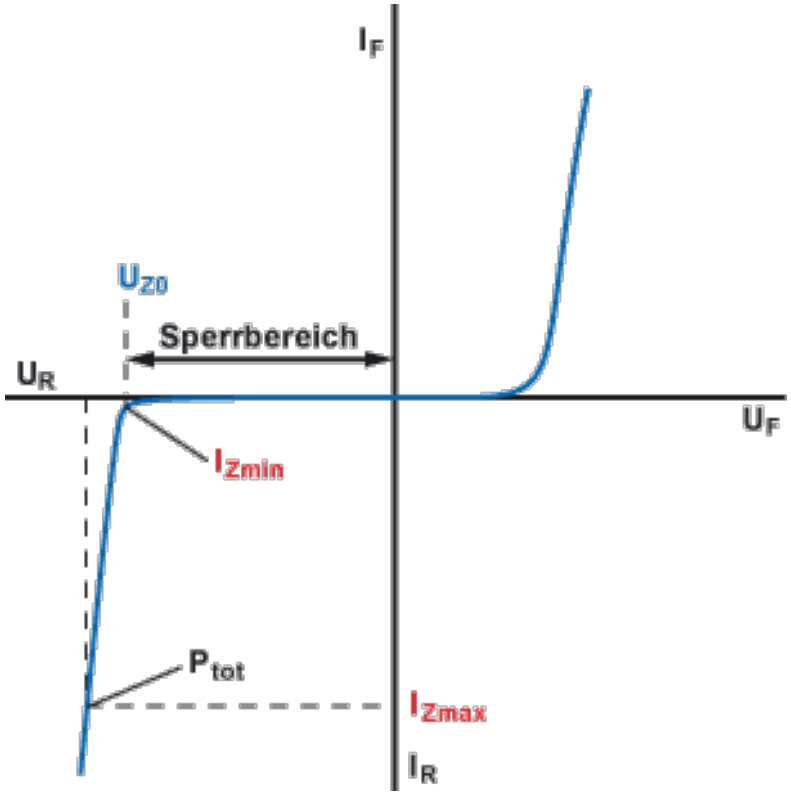
$U = R\ I$

- Spannung: Wasserdruck
- Strom: Liter pro Sekunde
- Widerstand: Dünne Leitung, Wasserrad.

- Viel Druck heisst nicht automatisch viel Wasser!
  - Druck fällt zusammen, sobald Wasser ausfliesst!

Diode


Sperr - Richtung
A
K
Durchlass - Richtung
I_D


I_F
U_{Z0}
Sperrbereich
U_R
U_F
I_{Zmin}
P_{tot}
I_{Zmax}
I_R

Pin-Bezeichnungen

- 5V, Vcc, VD, V
  - Spannungsversorgung, oft 5V, manchmal 3.3V
- GND, G, 0, - (Masse, Ground, 0V)
- S (Signal, digital), A (Analog)
- TX, RX (Transmit, Receive)
- CLK, SDC (Clock, Taktsignal)
- EN (Enable)
- RST (Reset)

Protokolle

- Seriell (Pins 0/1, oder Software sonstwo)
  - https://www.arduino.cc/en/Reference/Serial
  - https://www.arduino.cc/en/Reference/softwareSerial
- I2C (Wire-Protokoll, Pin A4 (SDA) ,A5 (SCL))
  - https://www.arduino.cc/en/Reference/Wire