WaveHit - Entwicklung eines elektronisch steuerbaren Modalhammers

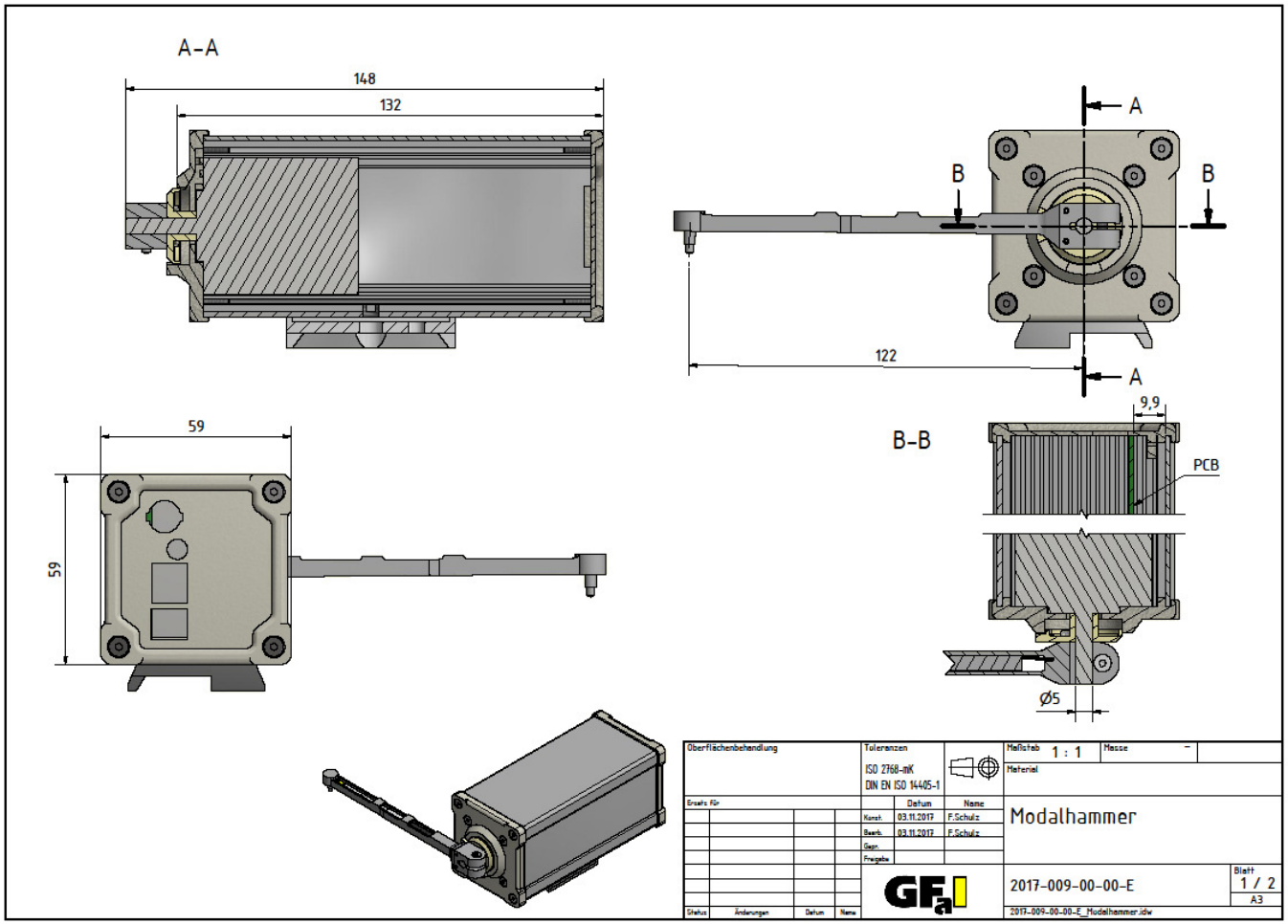
Dr. Gerd Heinz

Großer Saal der GFaI

19.04.2015, 15.30 Uhr

Inhalt

- Grundlagen
- Entwicklungsziele
- Experimentelle Basis
- Auf der Suche nach dem "Single Hit"
- Weg der Erkenntnis
- Physik hilft: $F = m \cdot a$
- Hardware: Es brummt!
- Software
  - Kernalgorithmus
  - zwei Interfaces
- Installation
- Hitparade
- Ausblick

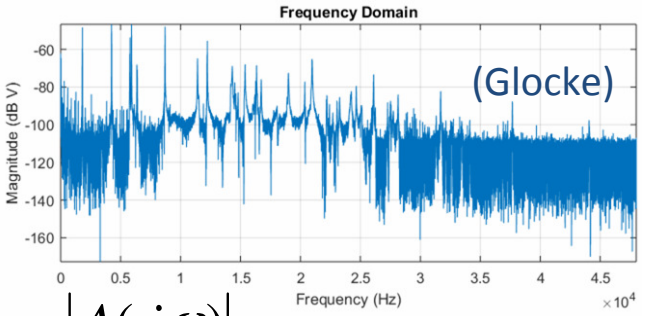


Grundlage: Frequency Response Function (FRF)

- FRF: *H(jω)* als Verhältnis von Antwort *A(jω)* zu Erregung *E(jω)* :

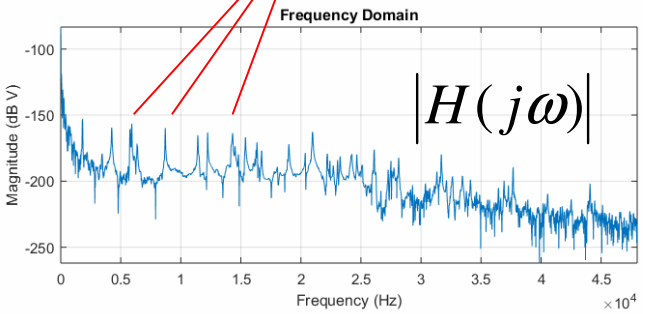
$$H(j\omega) = \frac{A(j\omega)}{E(j\omega)}$$

- die Zeitfunktion der Erregung *E(jω)* wird vom Sensor im Hammer abgenommen
- die Zeitfunktion der Antwort *A(jω)* am Meßobjekt
- *H(jω)* hat eine sog. SDOF-Struktur (Single Degree Of Freedom), jeder Peak ist einer Schwingungsmode zuordenbar ("Resonanzfrequenzen")
- Ziel: Erregung *E(jω)* muß breitbandig sein, im Frequenzbereich konstant,
  → Rauschen oder Dirac-Stoß



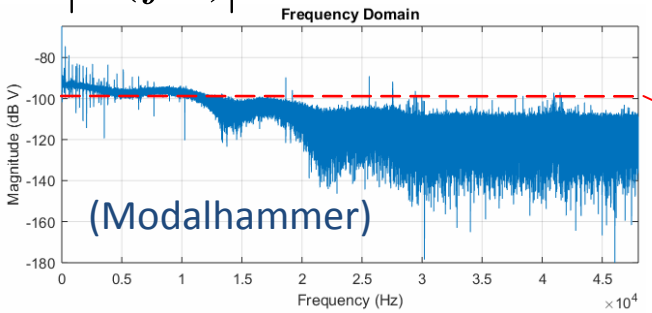
$|A(j\omega)|$

Peaks: Eigenfrequenzen des Systems



$|H(j\omega)|$

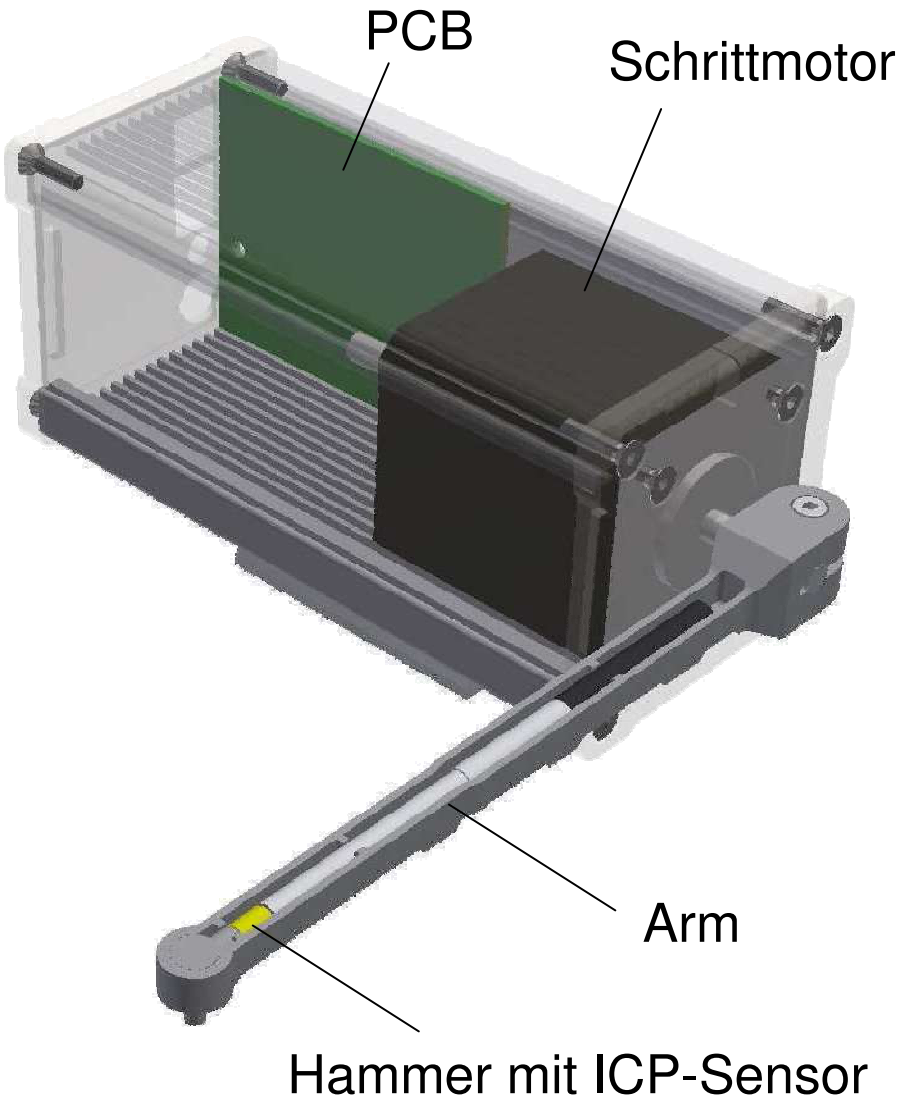
$|E(j\omega)|$



Ideale Anregung

Entwicklungsziele

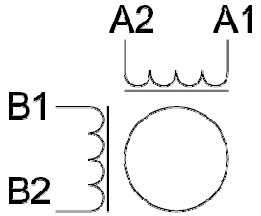
- Modalanalyse bedingt reproduzierbare Erregungs-Stöße
  - Serientests in der Produktion
  - Langzeit-Tests
  - Vergleichbarkeitstests
- Historie: Viele Methoden und Verfahren zur Stoßanregung wurden entwickelt, setzten sich aber nicht durch
- Schrittmotoren brachten den Durchbruch, z.B.
  - Maul-Theet Berlin
  - Prof. Blaschke Wildau
- Unser Ziel: Eigenentwicklung ohne die Nachteile marktüblicher Produkte
  - uneingeschränkte Beweglichkeit des Arms
  - keine proprietären Softwarepakete (Trinamic: 1,5 GB)
  - effiziente Fertigung
  - geringe Selbstkosten
  - externe Triggermöglichkeit

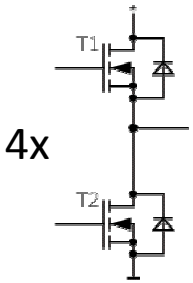


Schrittmotor?

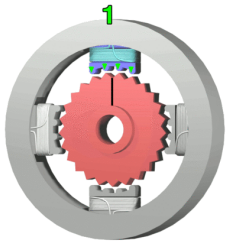
- Typ NEMA17 (BxHxT 42x42x50 mm)
- 200 / 400 Schritte pro Umdrehung

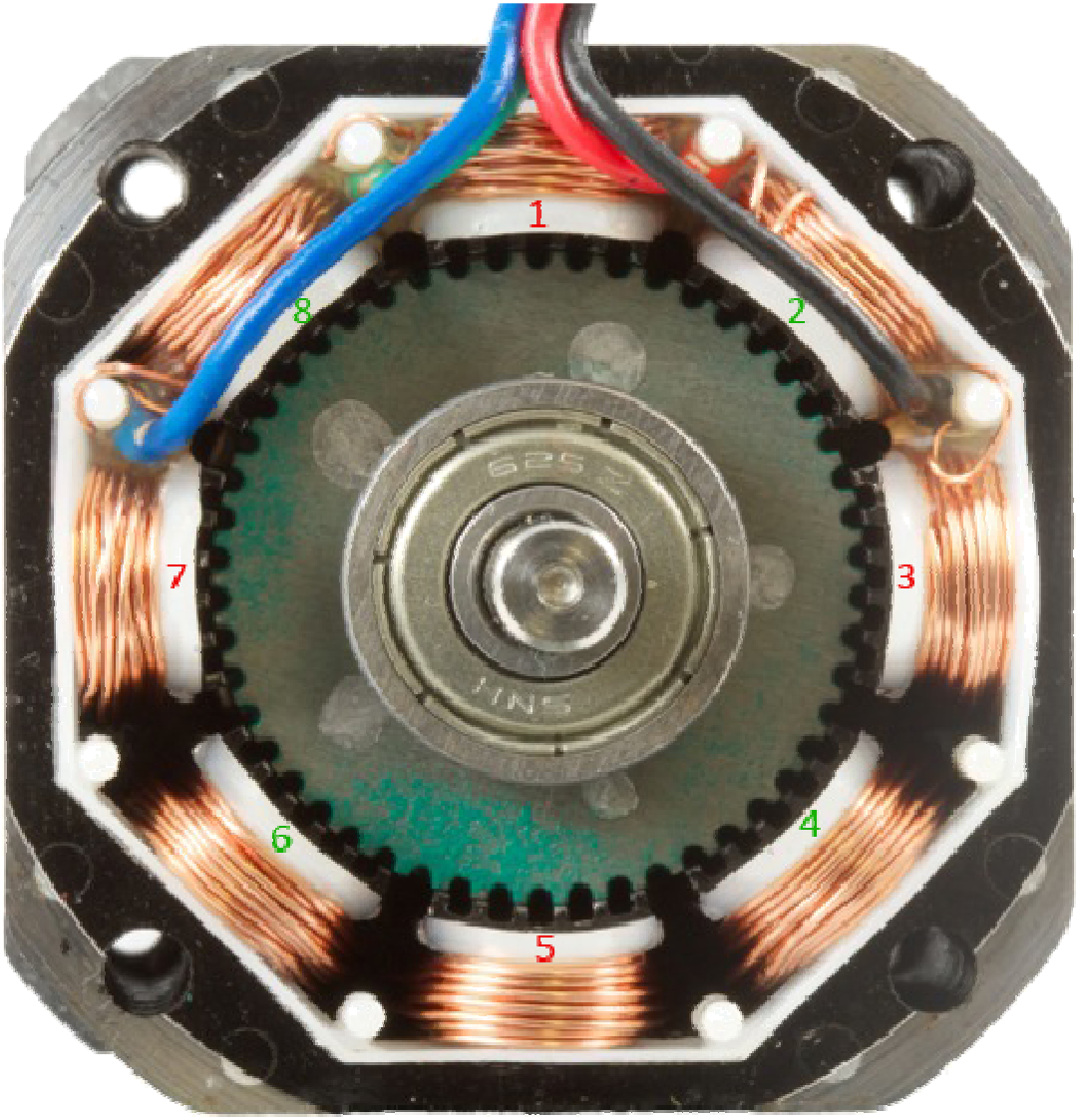
- bipolare Ansteuerung der zwei Wicklungen (1,3,5,7) und (2,4,6,8) mit vier Halbbrücken





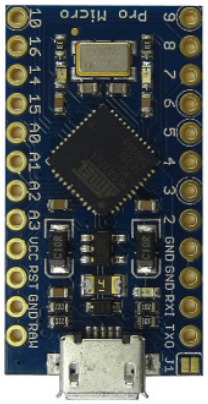
- gezahnte Polschuhe z.B. Stator = 6x8 = 48 Rotor = 50 für 200 steps/rev, Wirkprinzip etwa (Quelle: Wikipedia)





Experimentelle Basis für erste Versuche

µC — **Motor-treiber** — **Schritt-motor** — **Hammer mit Sensor**



Arduino proMicro

A4988, TMC2100, TMC2208, 2130…

NEMA17

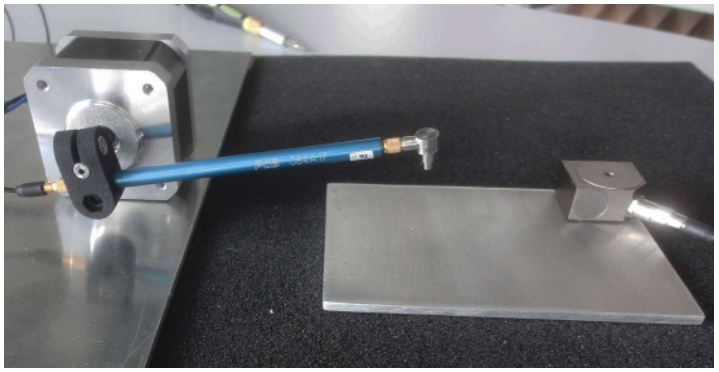
PCB 086E80

Arduino IDE (C++)

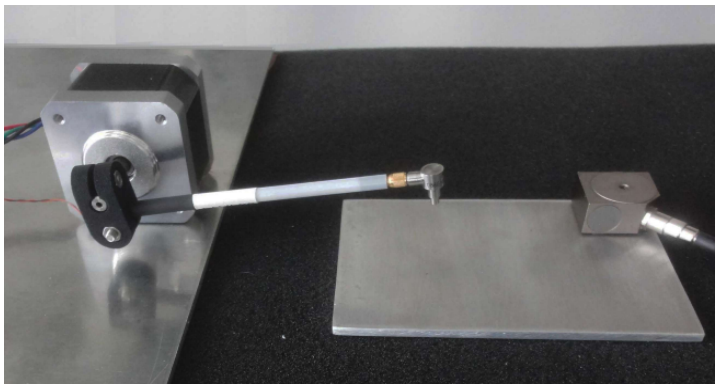
Weg der Erkenntnis

- Experimente mit verschiedenen Armen und Beschleunigungskurven
- Variation der Drehrichtungsumkehr vor/nach Aufschlagszeitpunkt
- beste Ergebnisse brachte das "Trinkröhrchen"
- Wie ist das System zu abstrahieren?
- Aufkommende Vermutung:
  Je steifer der Arm,
  desto mehr prellt es;
  je weicher der Arm,
  desto weniger?

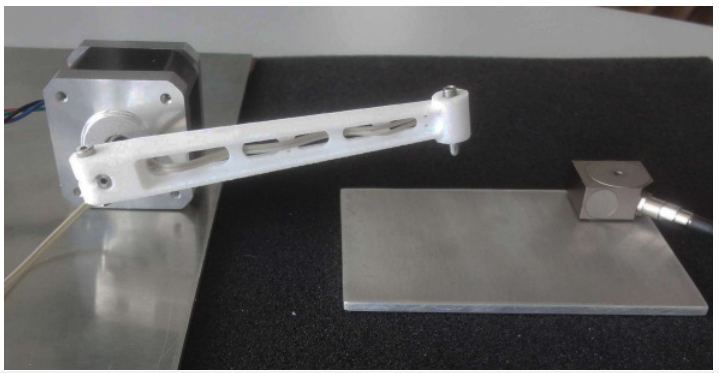
~~Alurohr~~



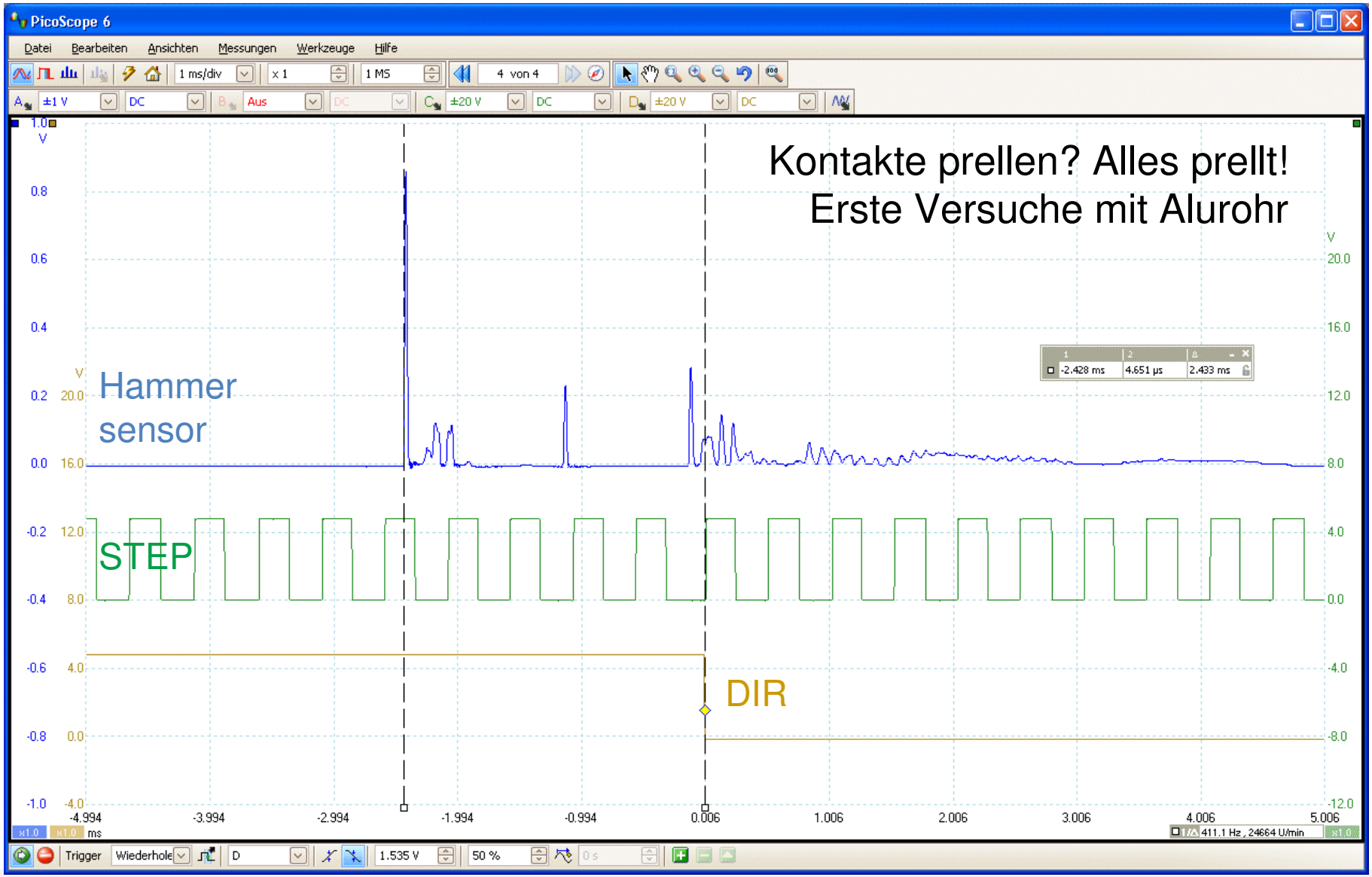
Trinkröhrchen



~~leicht und supersteif~~
~~mit Eigenbau-Sensor~~



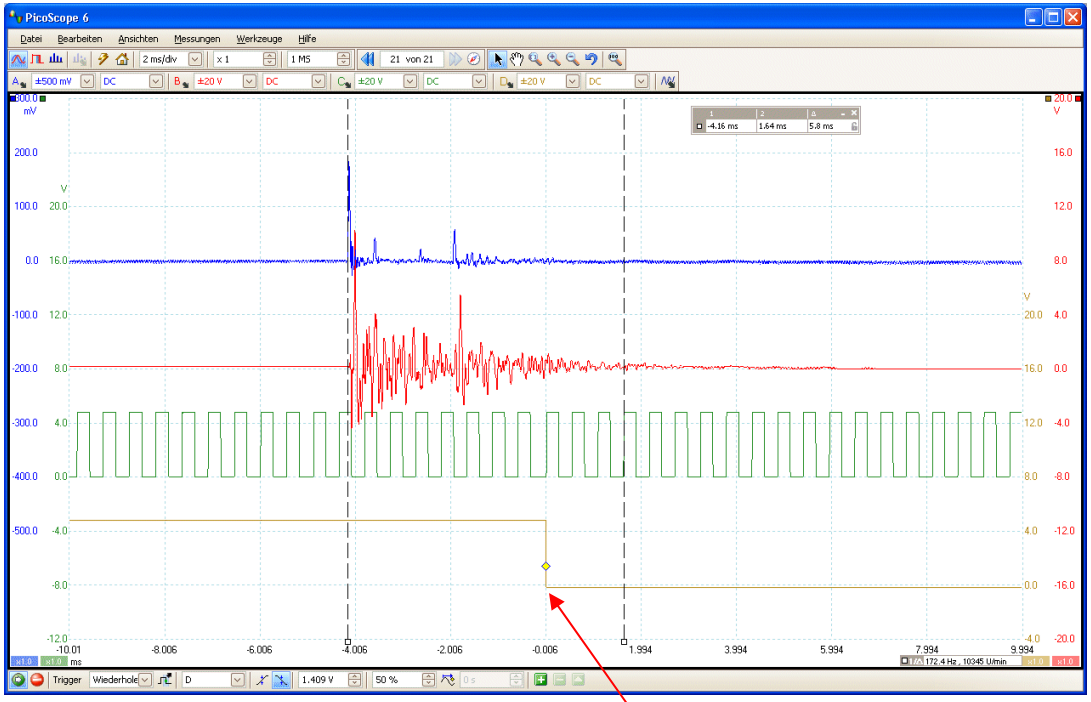
Auf der Suche nach dem Single Hit



Es prellt

- und prellt
- und prellt
- …

- Man vergleiche die Breite des Prellens mit der Lage zur Richtungsumkehr, siehe Bild
- Es scheint, daß die Richtungsumkehr erfolgen muß, *bevor* der Hammer aufschlägt
- Wird die Richtung aber vor dem Aufschlag gewechselt, schlägt der Hammer nicht auf!



blau Erregung Hammer
rot Antwort auf der Platte
grün Motor-Steps
gelb Drehrichtung

Richtungsumkehr

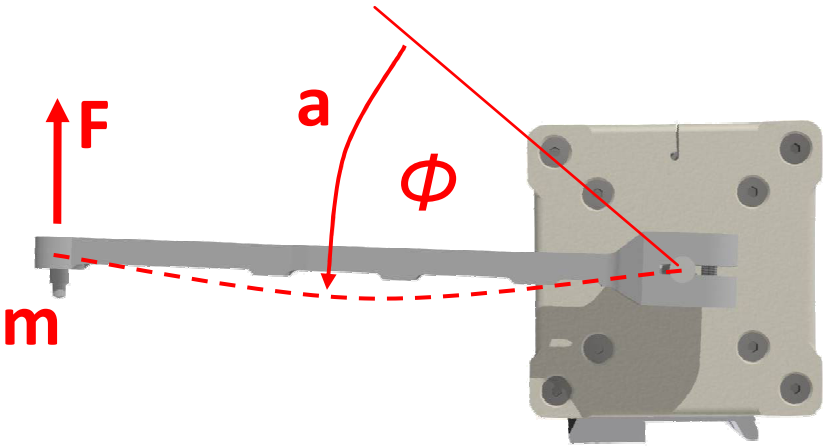
Physik hilft

Der elastische Arm

- Was bedeutet $F = m \cdot a$? Bei konstanter Hammermasse *m* ist die auf den Arm wirkende Biegekraft *F* proportional zur Beschleunigung *a*

$$a(t) = \frac{d\omega}{dt} = \frac{d^2\phi}{dt^2} = F(t)/m$$

$$F \sim a$$



- Der Arm wird entsprechend seiner Biegesteife durchgebogen und speichert dabei Energie
- Es wird möglich, die Drehrichtung *umzukehren, bevor* der Hammer aufschlägt
- Je elastischer der Arm, je größer wird die Durchbiegung, je reproduzierbarer wird ein Single Hit
- Dies erklärt, warum steife Arme keinen Single Hit liefern können
- logische Abfolge: *a* const → *F* const. → Durchbiegung const. → Offset const. für alle Schlagenergien
- Novum: Unser Hammer arbeitet mit *konstanter* Beschleunigung bei *variablem* Weg

Hardware: es brummt!

Beginn der Experimente:

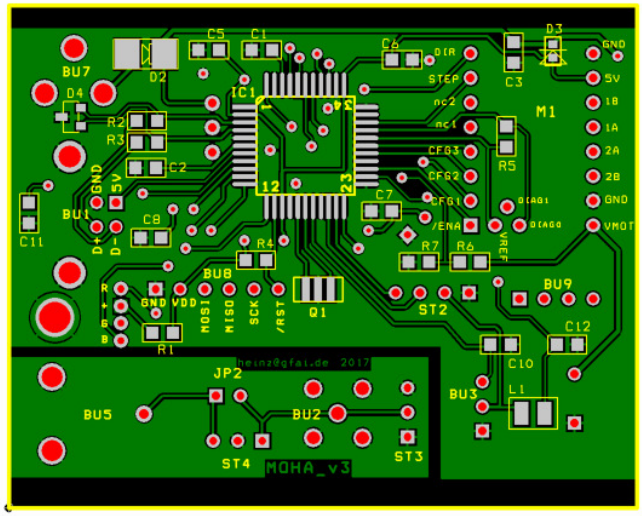
- Vollschritt mit 2 Vollbrücken – laut, Vibration wirkt auf Hammersensor

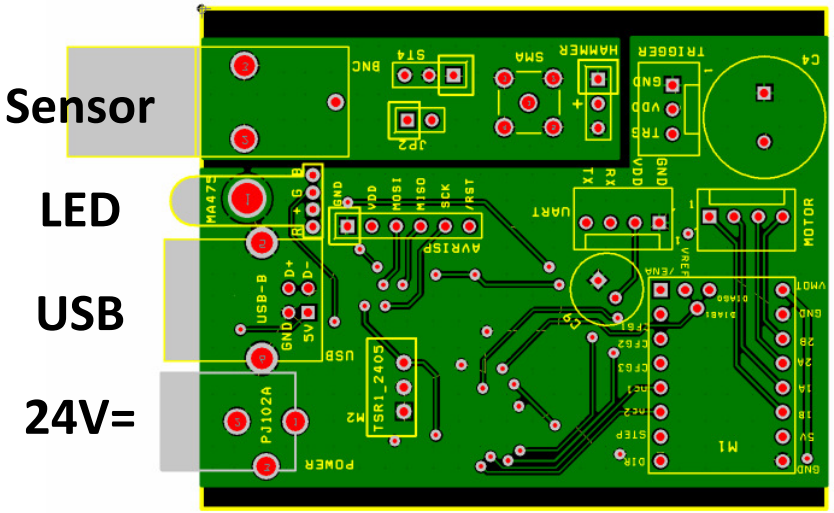
Fortsetzung:

- 16 Microsteps pro Schritt mit Allegro A4988 Modul – besser, dafür instabiler

Finale:

- 16x16 Microstep-Technologie von Trinamic TMC2100 Modul – fast unhörbar, aber noch instabiler
- zweilagige PCB
- µC: ATmega32U4
- Motor NEMA17 mit 400 Schritten/Umdrehung

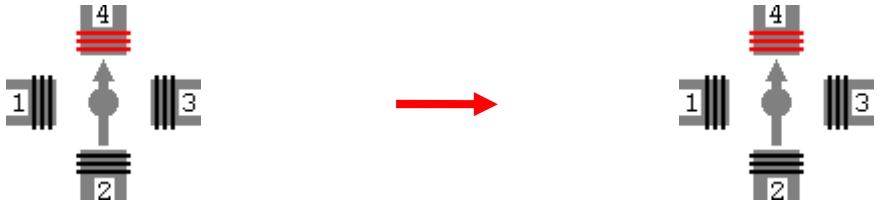


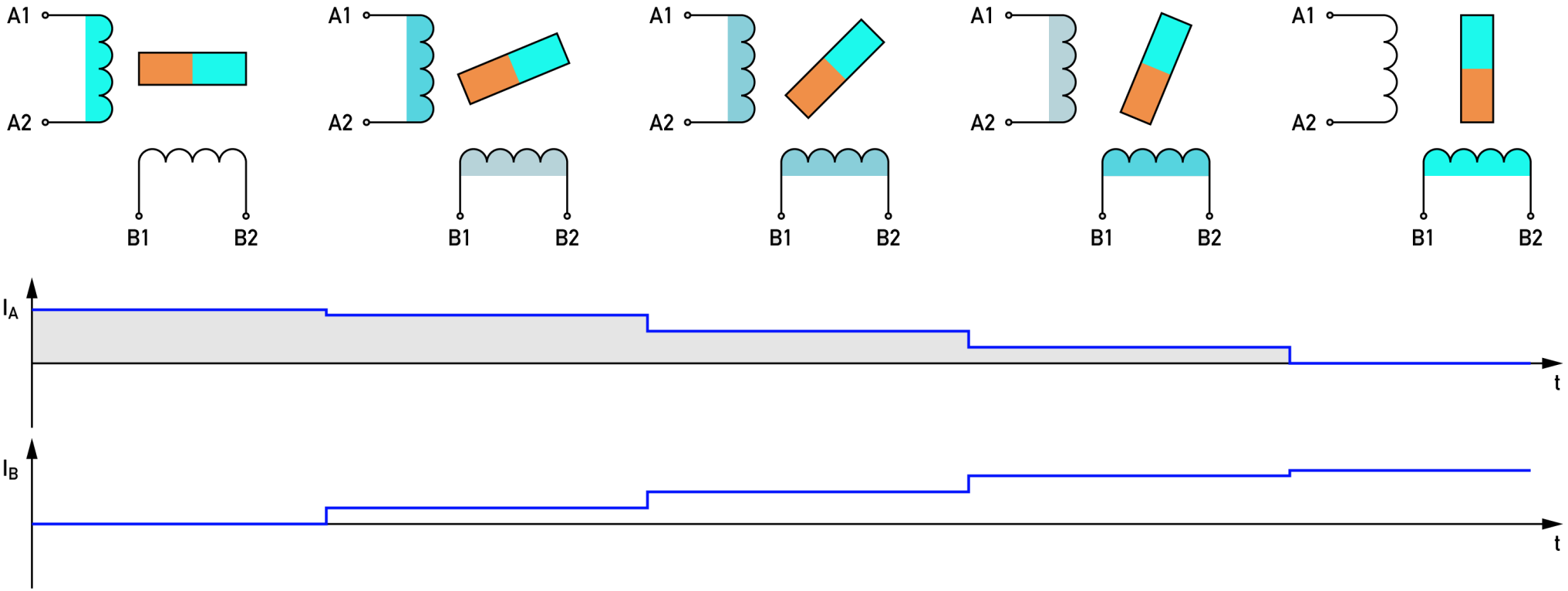


Microstepping

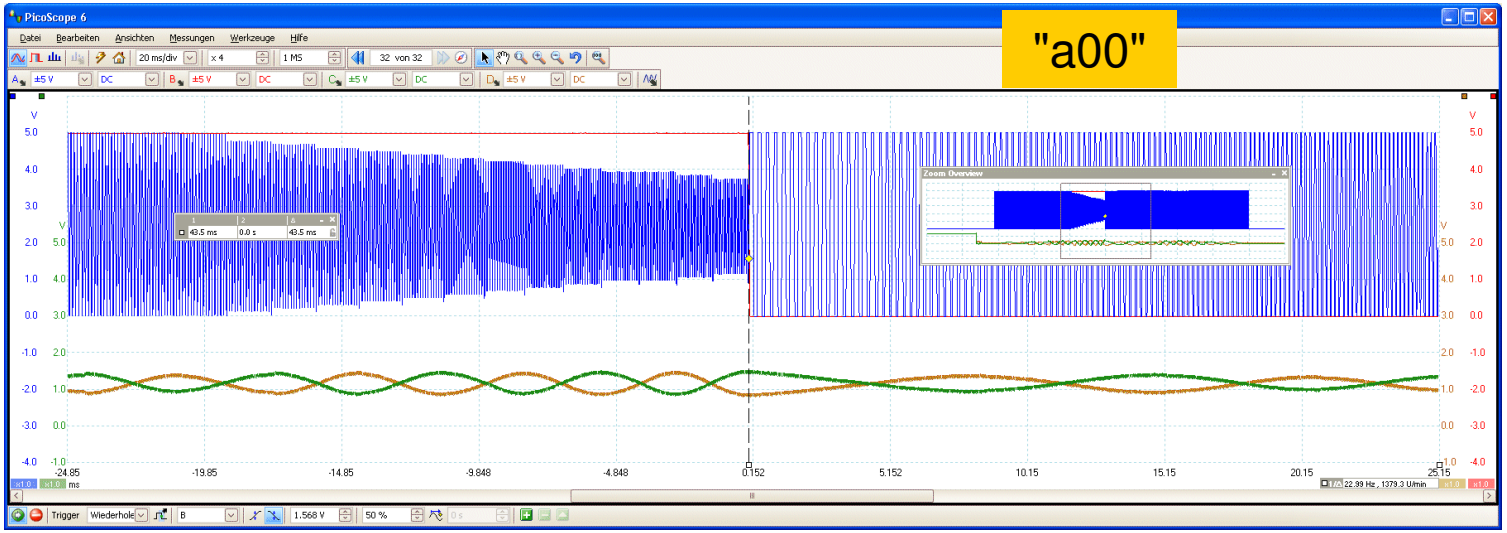
durch Variation des Motorstroms

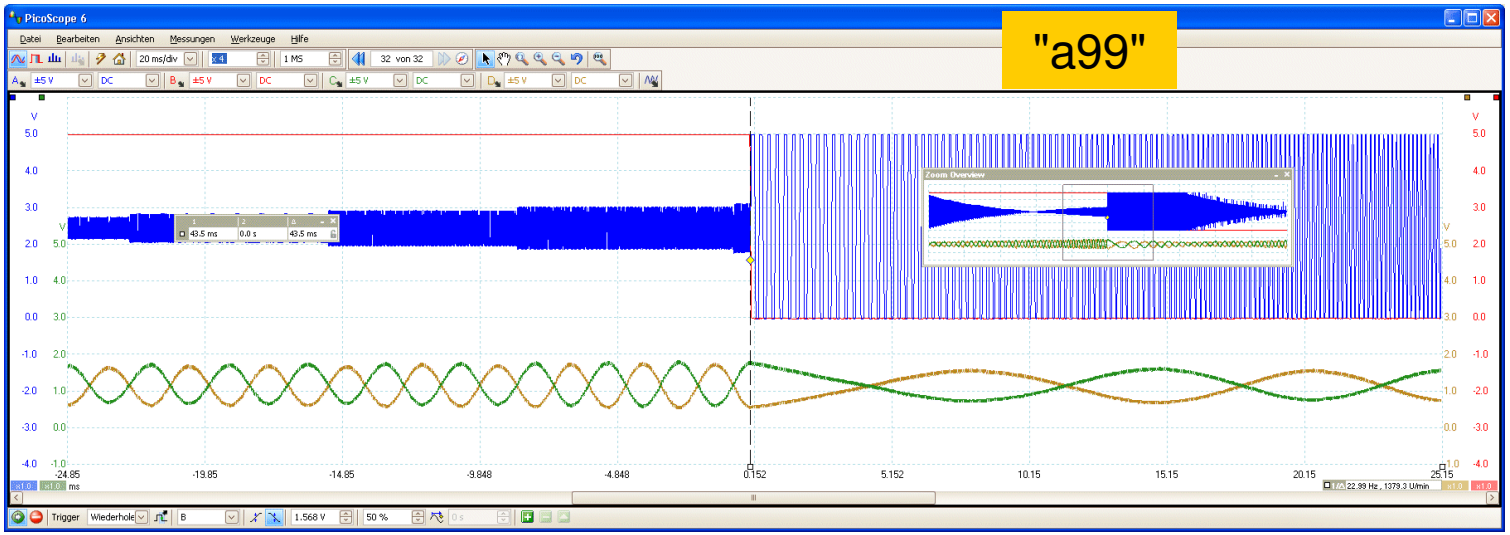
- Vorreiter: Allegro, Trinamic





Nanosteps: PWM-Modulation der Coils





blau STEP:

Microsteps vom µC

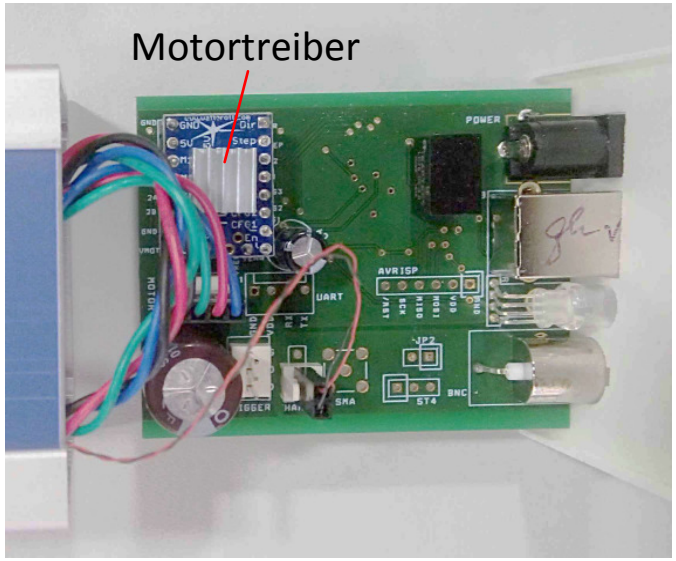
rot DIR:

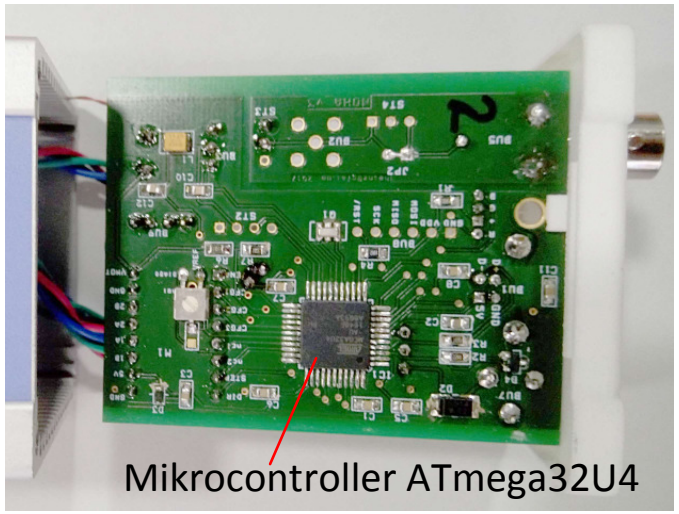
Drehrichtung

grün + **gelb**:

Motor-Coil A (PWM)

Hardware Aufbau





Kernalgorithmus: Beschleunigungstabelle

```
uint8_t delaytab(int index) {  // Beschleunigungskurve mit Step-Delays in Mikrosekunden*2
        // input: index, output: delay in µs aus Beschleunigungstabelle a_const_neu.sce
        // Step-Delays in Mikrosekunden mal 2, (symm. Puls) für 400er Motor
        const static uint8_t dellist[130] = {
        145,122,107,97,89,83,78,74,70,67,\
        64,62,59,57,56,54,53,51,50,49,\
        48,47,46,45,44,43,42,42,41,40,\
        40,39,38,38,37,37,36,36,35,35,\
        35,34,34,33,33,33,32,32,32,31,\
        31,31,30,30,30,30,29,29,29,29,\
        28,28,28,28,27,27,27,27,27,26,\
        26,26,26,26,26,25,25,25,25,25,\
        25,24,24,24,24,24,24,24,23,23,\
        23,23,23,23,23,23,22,22,22,22,\
        22,22,22,22,22,22,21,21,21,21,\
        21,21,21,21,21,21,20,20,20,20,\
        20,20,20,20,20,20,20,20,19,19\
        };
return( dellist[index] );  } // wird aus dem RAM gelesen (schneller)
```

```
…
   while (stepi < a_value) {   // Schlag
    deli = delaytab(stepi);    // Delay i holen
     while (n < 16) {          // 16 Microsteps
       puls(deli);    // Puls erzeugen (plus 2 µs)
       n++;
     }
     n=0;
     stepi++;              // nächster Schritt
   }
…
```

Software: Debug-Interface (USB-VCP 115200 8n1n)

```
>?
Welcome! Type h for help!
>h
 Modal Hammer WaveHit
 USB-VCP, 115200 baud, 8n1n
 moha_v33 for PCB MOHA_v3
 Nov 27 2017 16:50:23 AIDE 1.8.3
To start:
 - set the hit acceleration to a00...a99
 - press d for touch-down
 - beat with f, space, n or x
 - achieve a one-hit with o or j functions
Commands:
 h: this help
 p: show parameters
 a..: set acceleration (a00...a99)
 d: touch down and go back to start
 g: go to start position
 +: offset justage step
 -: offset justage step
 o..: set offset justage steps (o-9...o+9)
 j..: set reversion delay (j00...j99)*8 in usec
 f: (or space key): fire
 n: cyclic repetition
 x: infinite repetition demo
 t: test variables on/off
 r: read parameters from EEPROM
 s: save parameters to EEPROM
 z: read motor driver vref
login/logout:
 ?: enable communication  - answer: Wel...
 #: disable communication - answer: Bye...
```

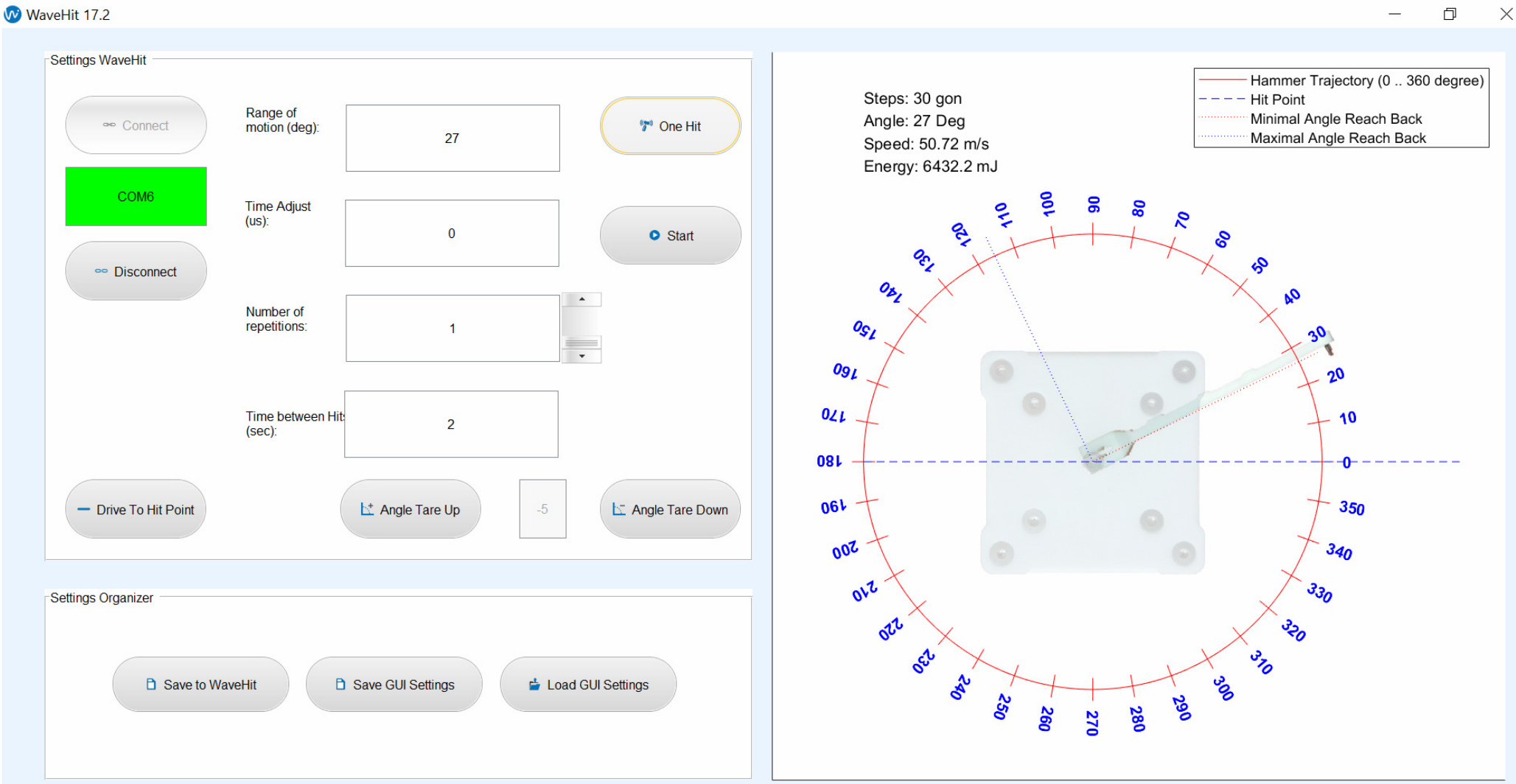
```
>r     // read EEPROM
[read_eeprom]
[steps] 30.00 {gon};
[offset] 3.00 {gon};
[sum] 33.00 {gon};
[justdel] 0 {us};
[mode] mv;
[rdy]

>f     // fire
[steps] 30 {gon};
[angle] 27.0 {deg};
[offset] 3 {gon};
[sum] 33 {gon};
[step_duration] 82.40 {us};
[hitspeed] 1.48 {m/s};
[energy] 5.5 {Nm};
[rdy]

>a99 // set new angle (0…99 ~ 30…129 GON)
[move] -99 {gon};
[steps] 129 {gon};
[angle] 116.1 {deg};
[offset] 3 {gon};
[sum] 132 {gon};
[rdy]
```

```
>p     // show parameters ( a00)
Parameters:
step angle [alpha] 0.90 {deg};
microsteps per step [msps] 16;
microsteps per rev [mspr] 6400;
steps per rev [stepsperrev] 400;
minimum accel [minsteps] 30 {gon};
maximum accel [maxsteps] 130 {gon};
hammer arm radius [radius] 124.00 {mm};
hammer mass [mass] 5.0 {gram};
arm inertial moment [m*r^2] 0.077 {g m^2};
rotor inertial moment [m/2*r^2] 0.010 {g m^2};
maximum speed [hitspeed] 1.48 {m/s};
kinetic hit energy [energy] 5.5 {mJ};
high current vref [imax] 0.51 {V};
low current vref [imin] 0.08 {V};
power supply voltage [vmot] 23.8 {V};
EEPROM parameters:
acceleration [steps] 30 {gon};
offset steps [offset] 3 {gon};
reversion delay [justdel] 0 {us};
stepper driver [mode] mv;
Stepper motor: type NEMA17-400;
[rho_Fe] 7874.00 {kg/m^3};
[rotor_diameter] 0.0260 {m};
[rotor_length] 0.0270 {m};
[rotor_mass] 112.9 {g};
```

Software: Windows-GUI (Daniel Herfert)



Installation

WaveHit

- USB-Bootlader installieren über AVRISPmkII (dann ist der Modalhammer ein Arduino-Bord vom Eigenbau-Typ "AVR32U4 micro 2ff1 7801") siehe GFaI-Wiki "USB-VID"
  oder
- Programm (moha_v34.ino) incl. USB-Bootlader über AVRISPmkII installieren
  und sofort arbeiten

USB-VCP unter Windows

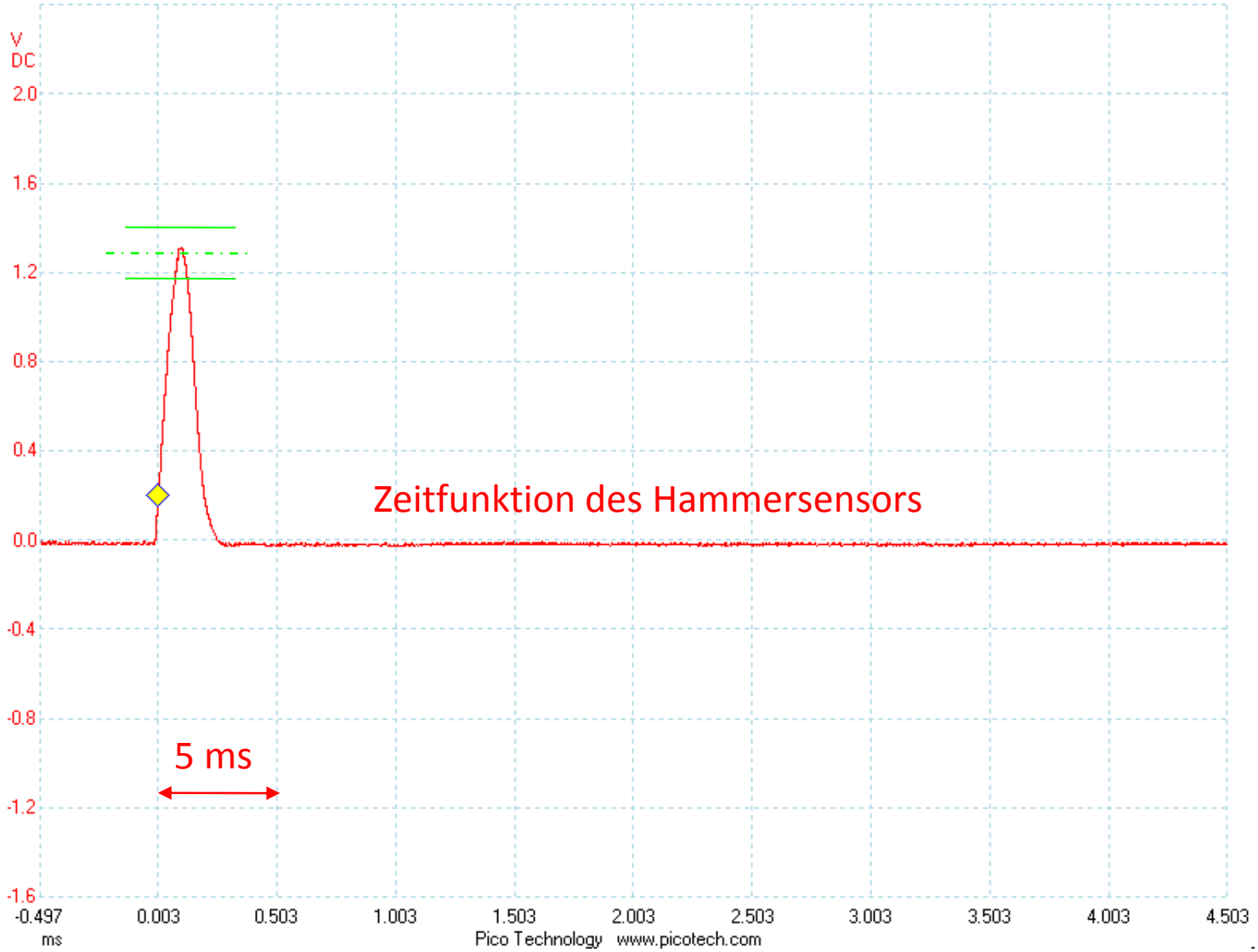
- Windows ungleich 7:
  USB stecken → ding/dong → "Neues Gerät gefunden!"
- Windows 7:
  (nichts für Warmduscher)

Daniels Windows-GUI

- muß installiert werden

Hitparade

Am 1.11.2017
war es soweit:
Der Modalhammer
schlug beliebig viele
"Single Hits"
nacheinander,
siehe Film →



Verteilung

Mittelwert μ = 1.35

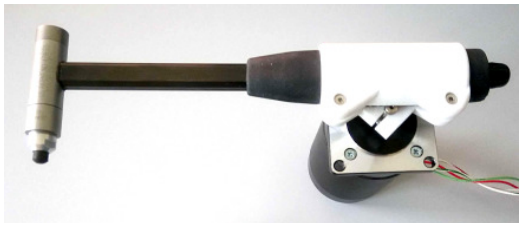
Varianz σ = 0.11

Streuung kleiner

+/- 9 %

Wie geht es weiter?

- Präzision verbessern

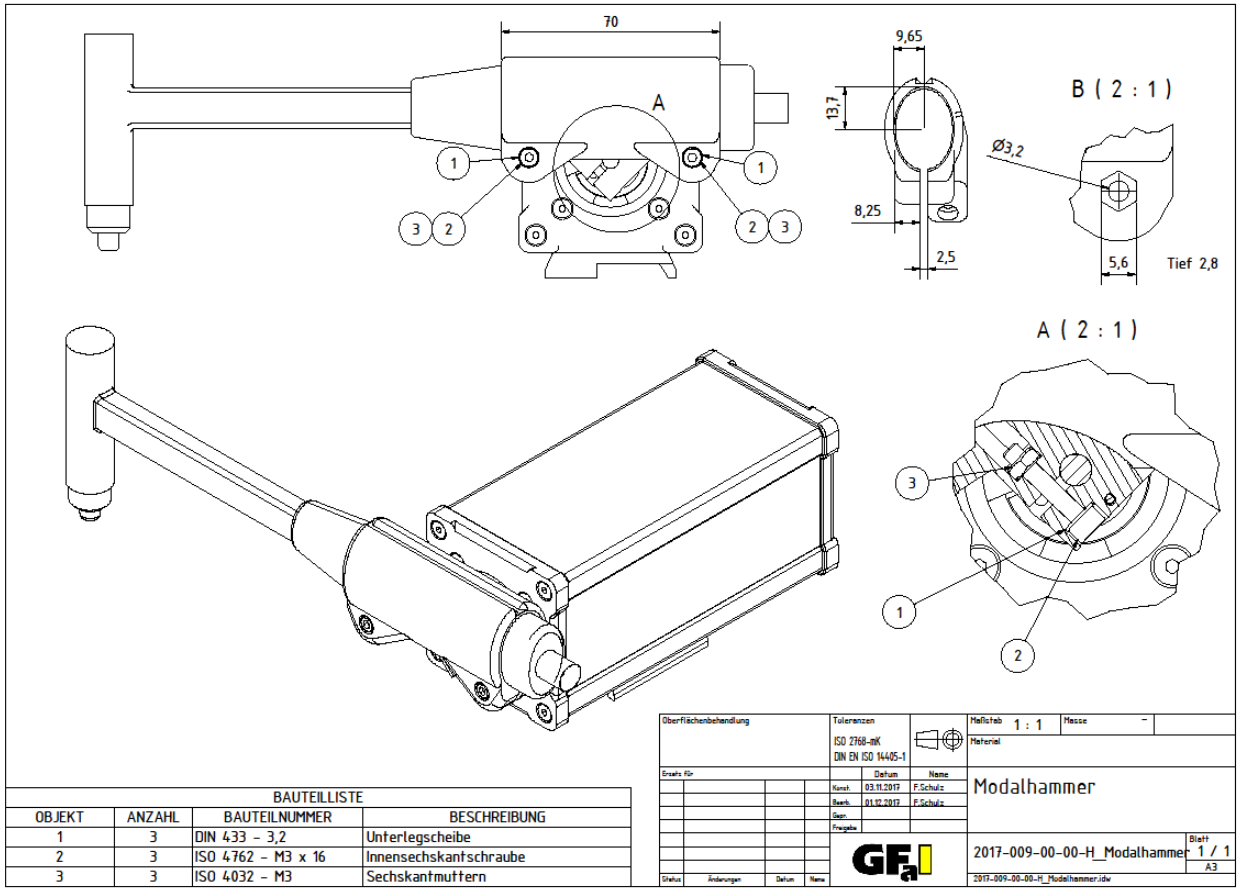


Stall Detection:

- Motoranschlag elektronisch registrieren
  → anderer Typ von Motortreiber nötig

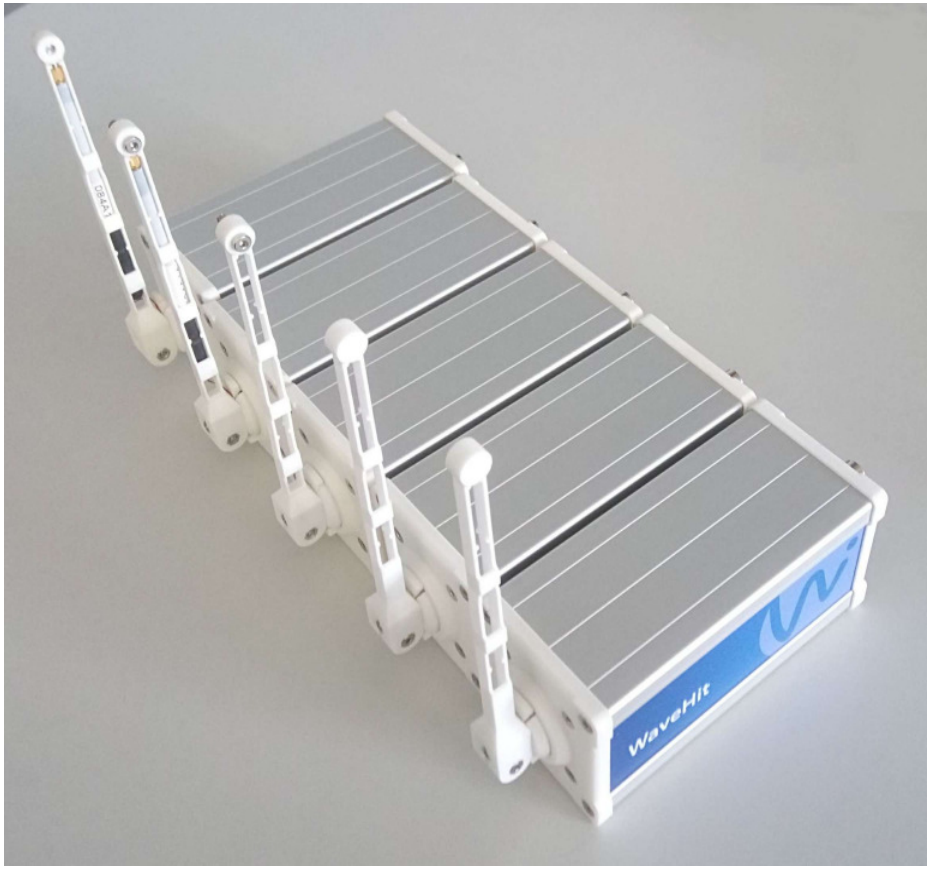
Großer Hammer:

- Hammer mit elektronischer Blockierung
  → Weiterentwicklung mit *externer* Power-MOS Brückenschaltung (zur Abführung der Wärme)
- neue Konstruktion



Vielen Dank für Ihre Aufmerksamkeit,

herzlichen Dank auch an die Mitstreiter Florian Schulz und Daniel Herfert!



Gesellschaft zur Förderung angewandter Informatik e.V.

Volmerstraße 3
12489 Berlin-Adlershof

Telefon : +49 30 814563-300
Telefax: +49 30 814563-302
eMail: info@gfai.de
Internet: www.gfai.de

Kontakt:
Dr. Gerd Heinz
Bereich: SDM
Telefon: +49 30 814563-490
eMail: heinz@gfai.de