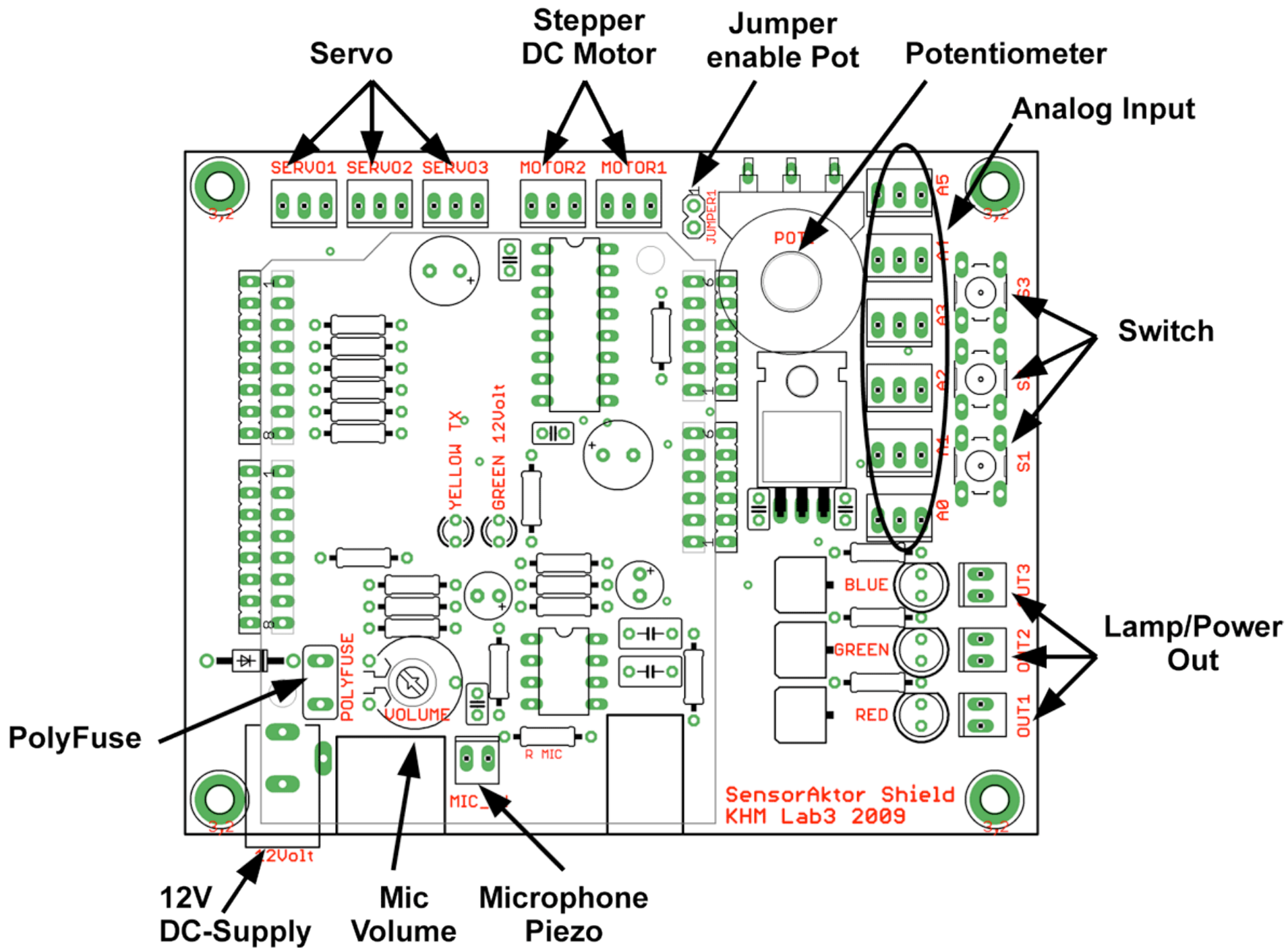


# Code & Material

2. Stunde, 24. Okt. 2012

Georg Trogemann



```
/* general setting when using Sensor-Aktor  
Board */
```

```
/* pin mapping */
```

```
// analog inputs
```

```
#define analog0 0
```

```
#define analog1 1
```

```
#define analog2 2
```

```
#define analog3 3
```

```
#define analog4 4
```

```
#define analog5 5
```

```
// poti
```

```
#define poti 5
```

```
// amplified input
```

```
#define mic 0
```

```
// switches
```

```
#define switch1 15
```

```
#define switch2 16
```

```
#define switch3 17
```

```
// led outs (same pins than power outs)
```

```
#define led1 3
```

```
#define led2 5
```

```
#define led3 6
```

```
// power outs (same pins than led outs)
```

```
#define out1 3
```

```
#define out2 5
```

```
#define out3 6
```

```
// motor outs
```

```
#define motor1_1 8
```

```
#define motor1_2 13
```

```
#define motor2_1 11
```

```
#define motor2_2 12
```

```
// pwms motor
```

```
#define pwmMotor1 10
```

```
#define pwmMotor2 9
```

```
// servo outs
```

```
#define servo1 2
```

```
#define servo2 4
```

```
#define servo3 7
```

# Signalverarbeitung

- Die Welt der *operativen Symbolsysteme* (Algorithmen) und die *physikalische Welt* (Material) werden durch **Signale** vermittelt.
- Unter **Signalverarbeitung** verstehen wir alle Bearbeitungsschritte, die benötigt werden um Informationen aus einem gemessenen Signal zu extrahieren oder Informationen in Signale zu transformieren, die auf der Ausgabenseite in der Lage sind, Aktoren (und damit Handlungen) in der Umgebung auszuführen.

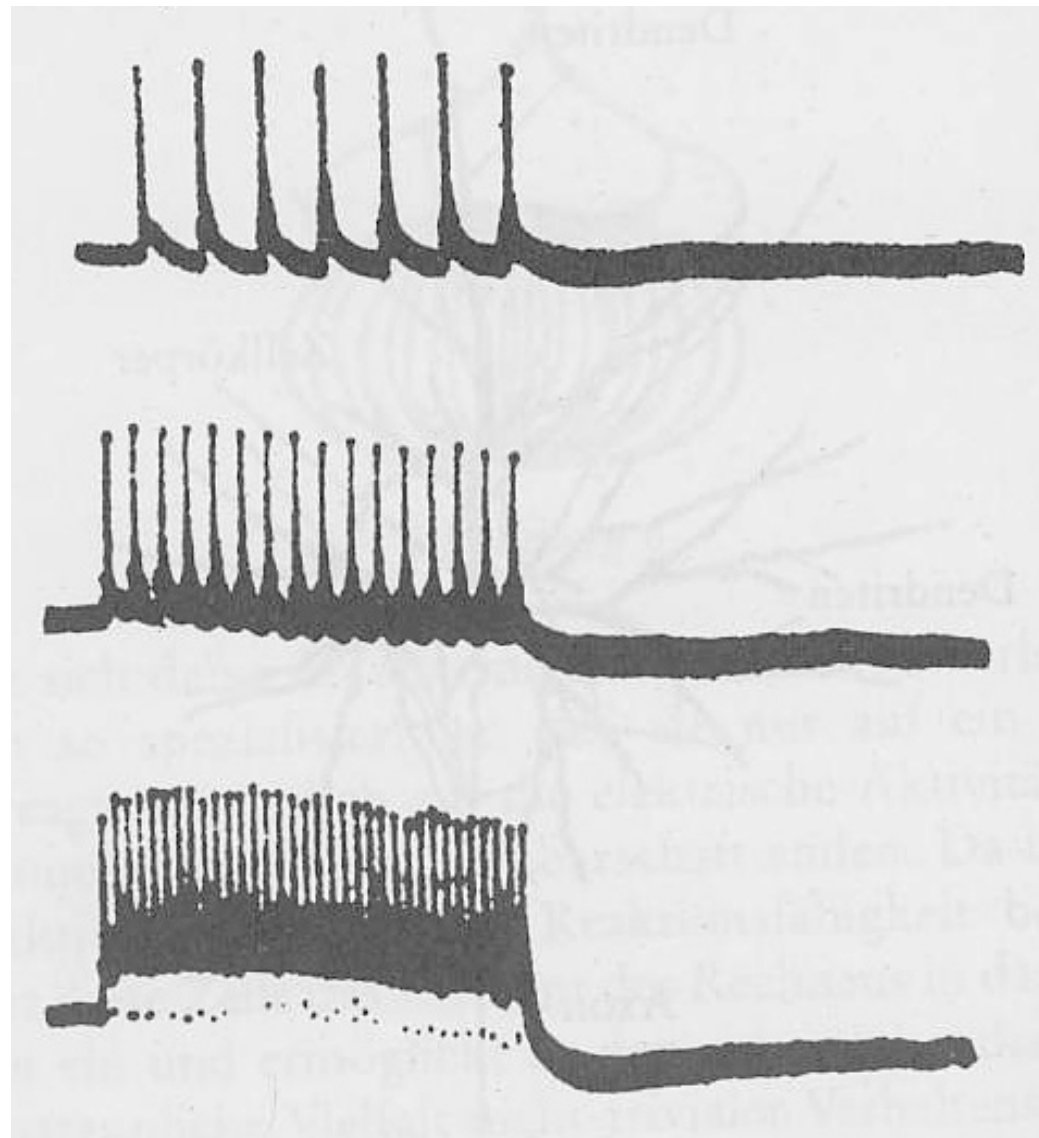
# Undifferenzierte Codierung beim Menschen

Das Prinzip der **undifferenzierten Codierung** wurde Mitte des 19. Jahrhunderts vom deutschen Neurophysiologen *Johannes Müller* formuliert.

Reize der Außenwelt, gleich welcher Qualität, werden im Gehirn ausschließlich quantitativ codiert. Je höher die „*Reizung*“, desto mehr Impulse (Aktionspotentiale) entstehen und werden weiter geleitet.

„Die Erregungszustände einer Nervenzelle codieren nur die Intensität, aber nicht die Natur der Erregungsursache. Codiert wird nur: ‘So-undso viel an dieser Stelle meines Körpers’ aber nicht ‘was’.“

# Frequenzcodierung von Rezeptoren



# Amplitudencodierung

- Undifferenzierte Codierung bei Maschinen
  - Verhältnisse der Außenwelt werden als Größen (Zahlen) codiert.
  - Je höher die „*Reizung*“, desto größer der erzeugte Wert der Zahl.
  - Die Welt wird im Computer meistens durch Größen codiert, d.h. sie ist *Amplitudencodiert*.
  - Gerade in der Signalverarbeitung ist die Frequenzcodierung der Informationen jedoch genauso wichtig wie die Amplitudencodierung.

# Arduino Eingaben

Wichtig sind folgende Größen:

- Eingabe:
  - Wo wird eingelesen? (an welchem Pin, Pin-Nummer oder Bezeichner)
  - An welche Variable wird der Wert übergeben?
- Digitale Werte = 1 Bit, **HIGH** und **LOW** (entspricht 0 und 1)  
`Wert = digitalRead(pin); // Wert ist 0 oder 1`
- Analoge Werte = 8 Bit, liegen zwischen 0 und 255  
`Wert = analogRead(pin); // Intervall [0 – 1023]`



# Arduino Ausgaben

Wichtig sind folgende Größen:

- Ausgabe:
  - Wo wird ausgegeben? (an welchem Pin, Pin-Nummer oder Bezeichner)
    - Digitale Ein-Ausgabe beim Arduino an allen 20 Pins
    - Analoge Ausgabe nur an PWM Pins (~)
  - Was wird ausgegeben?
    - Digitale Werte = 1 Bit, HIGH und LOW (entspricht 0 und 1)  
`digitalWrite(pin, WERT); // HIGH or LOW`
    - Analoge Werte = 8 Bit, liegen zwischen 0 und 255  
`analogWrite(pin, Wert); // Intervall [0 – 255]`

# Erste Programme

- sensorAktor Shield `#include, #define`
- Modus der Pins definieren `pinMode();`
- Leuchtdiode an- und ausschalten  
(Relay steuern) `digitalWrite();`
- Serielle Kommunikation `Serial.begin(); Serial.print();`
- Leuchtdioden über Schalter steuern `digitalRead();`
- Poti einlesen  
(Lichtsensoren einlesen) `analogRead();`
- Leuchtdiode über poti steuern  
(DC Motor steuern) `analogWrite();`