

atei: Laser-Schießstand.ino

Autor: Lukas Aaron Sperhake und Ivan Kleiböhmer

Erstellung: 07.10.2020

Letzte Änderung: 09.12.2020

Implementation: Arduino

Funktion:

Laser Schießstand mit vier Zielpunkten. Ziel ist es mit Hilfe eines Lasers alle Ziele zu treffen. Treffer sollen

über LED's, Display und Geräusch signalisiert werden.

```
#include <PCM.h>
```

```
#include <SPI.h>
```

```
#include <Wire.h>
```

```
#include <Adafruit_GFX.h>
```

```
#include <Adafruit_SSD1306.h>
```

```
Adafruit_SSD1306 display(OLED_RESET);
```

```
// _____ *L_A_S_E_R* _____
```

```
#define SOUND 8 // pin 2 für die Soundausgabe "BUZZER" _
```

```
#define LASER1 3 // pin 3 für den Sensor
```

```
#define LASER2 4 // pin 4 für den Sensor
```

```
#define LASER3 5 // pin 5 für den Sensor
```

```
#define LASER4 6 // pin 6 für den Sensor "RESET"
```

```
#define Playback 2 // pin 2 für den Lautsprecher
```

```
// _____
```

```
// _____ *L_E_D* _____
```

```
byte LED1 = 12; //
```

```
byte LED2 = 10; //
```

```
byte LED3 = 13; //
```

```
byte LED4 = 9; //
```

```
//_____
```

```
//_____ *S_O_U_N_D___C_O_D_E* _____
```

```
const unsigned char sample [] PROGMEM = {
```

```
129, 129, 129, 127, 127, 127, 129, 128, 128, 129, 129, 131, 128, 129, 129, 129, 130, 129, 128, 125,  
126, 126, 126, 128, 126, 126, 125, 127, 130, 128, 126, 123, 124, 125, 126, 129, 126, 129, 129, 128,  
128, 129, 130, 127, 128, 127, 126, 127, 124, 126, 128, 127, 127, 125, 125, 125, 129, 127, 123, 124,  
121, 123, 127, 128, 130, 129, 128, 125, 130, 130, 126, 126, 123, 125, 127, 127, 128, 128, 131, 130,  
134, 131, 130, 136, 131, 128, 131, 131, 126, 127, 130, 127, 128, 127, 125, 127, 125, 128, 130, 129,  
127, 134, 132, 126, 131, 131, 125, 126, 130, 128, 126, 128, 123, 126, 125, 125, 128, 125, 125, 128,  
131, 124, 126, 128, 120, 126, 127, 124, 124, 126, 128, 126, 133, 127, 127, 129, 125, 129, 126, 125,  
128, 131, 127, 126, 132, 125, 130, 136, 130, 128, 130, 129, 125, 128, 126, 121, 124, 124, 128, 128,  
123, 126, 133, 130, 130, 135, 129, 126, 131, 130, 124, 130, 130, 127, 129, 126, 128, 127, 125, 126,  
127, 125, 120, 130, 128, 124, 128, 123, 123, 121, 124, 121, 122, 129, 124, 130, 122, 121, 128, 125,  
125, 122, 129, 129, 133, 137, 129, 134, 130, 133, 136, 130, 127, 122, 128, 127, 133, 132, 121, 127,  
133, 140, 137, 131, 130, 128, 130, 126, 128, 127, 128, 134, 132, 129, 130, 133, 133, 132, 133, 127,  
122, 121, 124, 127, 123, 119, 121, 127, 132, 135, 130, 123, 122, 119, 119, 125, 132, 135, 131, 125,  
120, 124, 127, 123, 118, 115, 119, 124, 124, 127, 128, 126, 124, 130, 131, 126, 127, 123, 124, 125,  
122, 121, 121, 133, 139, 138, 126, 120, 134, 131, 132, 128, 122, 121, 123, 133, 127, 130, 129, 129,  
141, 135, 140, 136, 131, 131, 133, 137, 122, 129, 132, 133, 138, 126, 129, 126, 125, 122, 124, 128,  
122, 135, 129, 120, 125, 126, 125, 121, 126, 124, 125, 126, 124, 132, 125, 122, 130, 127, 119, 126,  
131, 127, 135, 139, 128, 124, 132, 132, 131, 126, 117, 124, 122, 121, 127, 124, 120, 126, 138, 125,  
129, 134, 123, 131, 133, 131, 124, 128, 128, 125, 133, 122, 130, 136, 131, 139, 131, 126, 126, 136,  
124, 120, 133, 118, 123, 134, 131, 125, 127, 130, 124, 128, 120, 120, 125, 118, 128, 129, 117, 116,  
130, 121, 114, 133, 129, 124, 132, 134, 127, 126, 126, 119, 125, 118, 121, 128, 116, 119, 128, 124,  
113, 130, 134, 123, 138, 139, 137, 143, 139, 138, 139, 140, 127, 127, 124, 120, 141, 136, 126, 133,  
138, 141, 146, 148, 131, 130, 128, 116, 135, 137, 126, 126, 125, 127, 129, 136, 124, 122, 130, 117,  
124, 119, 118, 133, 127, 120, 120, 135, 128, 131, 148, 129, 127, 115, 105, 113, 122, 132, 119, 120,  
113, 124, 138, 117, 121, 111, 102, 105, 115, 124, 116, 124, 112, 117, 136, 129, 131, 127, 135, 137,  
135, 127, 119, 135, 137, 141, 138, 126, 129, 131, 128, 125, 139, 136, 130, 139, 141, 135, 126, 125,  
122, 131, 137, 137, 136, 127, 140, 142, 135, 133, 141, 143, 126, 135, 137, 133, 124, 112, 117, 115,  
131, 134, 127, 119, 118, 136, 126, 125, 123, 118, 117, 116, 132, 128, 123, 120, 127, 134, 124, 128,  
125, 129, 137, 139, 128, 117, 128, 137, 141, 122, 110, 111, 109, 115, 123, 131, 118, 129, 139, 132,  
136, 128, 131, 125, 127, 122, 115, 122, 115, 138, 131, 118, 132, 132, 140, 133, 142, 127, 127, 146,  
122, 127, 118, 117, 128, 127, 131, 115, 125, 118, 127, 134, 101, 116, 118, 112, 120, 125, 119, 112,  
146, 134, 118, 132, 119, 125, 121, 120, 126, 121, 123, 127, 141, 116, 119, 161, 142, 141, 154, 132,  
119, 140, 156, 121, 118, 127, 137, 162, 139, 146, 158, 143, 148, 157, 137, 107, 145, 141, 104, 122,  
112, 110, 136, 163, 155, 132, 113, 99, 145, 150, 128, 131, 110, 109, 138, 142, 102, 117, 126, 93, 110,  
109, 105, 119, 124, 100, 105, 128, 99, 122, 126, 114, 120, 103, 95, 100, 131, 103, 115, 140, 97, 126,  
126, 107, 117, 137, 108, 55, 103, 112, 135, 141, 98, 122, 139, 152, 147, 149, 121, 116, 159, 118, 127,  
138, 110, 117, 144, 157, 132, 144, 127, 141, 180, 152, 149, 140, 131, 138, 157, 136, 106, 132, 123,  
115, 138, 138, 136, 166, 134, 119, 161, 123, 118, 116, 145, 163, 110, 121, 117, 137, 113, 135, 167, 89,  
132, 136, 128, 119, 122, 159, 109, 153, 141, 149, 150, 123, 185, 116, 123, 110, 111, 159, 87, 129, 124,  
111, 113, 125, 158, 109, 138, 131, 128, 142, 150, 130, 104, 170, 106, 116, 137, 120, 165, 130, 178,
```



```
57, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 120, 218, 60, 0, 0, 0, 129, 255, 163, 63, 255, 255, 255, 255, 255, 243, 0,
0, 19, 24, 210, 212, 203, 113, 0, 233, 255, 77, 0, 0, 23, 0, 13, 38, 0, 0, 96, 0, 0, 0, 0, 46, 134, 130, 0, 88,
255, 255, 255, 255, 255, 4, 0, 0, 0, 0, 70, 0, 0, 0, 0, 0, 0, 0, 50, 86, 0, 0, 83, 0, 0, 0, 0, 7, 134, 64,
0, 82, 15, 0, 0, 0, 0, 0, 0, 61, 76, 0, 0, 147, 163, 111, 156, 255, 255, 76, 0, 37, 255, 255, 255, 19, 0,
0, 0, 0, 0, 0, 0, 0, 146, 255, 255, 255, 255, 230, 114, 0, 118, 182, 0, 121, 255, 255, 255, 255, 255,
255, 191, 72, 179, 114, 244, 255, 114, 92, 215, 67, 0, 0, 0, 0, 35, 0, 0, 16, 0, 0, 0, 22, 70, 204, 255,
255, 255, 255, 255, 255, 115, 0, 174, 255, 65, 0, 85, 197, 166, 255, 255, 255, 255, 255, 255,
174, 252, 51, 41, 255, 203, 182, 171, 10, 68, 139, 189, 245, 74, 0, 45, 135, 134, 206, 5, 0, 0, 0, 0, 0,
0, 157, 255, 210, 255, 128, 0, 12, 0, 31, 223, 253, 158, 146, 253, 255, 255, 255, 255, 240, 172, 206,
239, 255, 255, 255, 255, 249, 151, 80, 149, 255, 221, 124, 0, 0, 0, 0, 23, 0, 0, 0, 0, 146, 59, 0, 243, 3, 0,
71, 0, 0, 0, 0, 0, 0, 0, 28, 98, 185, 164, 142, 103, 0, 41, 212, 115, 100, 165, 131, 59, 0, 22,
110, 232, 255, 208, 130, 149, 183, 129, 48, 42, 72, 115, 136, 174, 219, 153, 113, 194, 216, 175, 171,
175, 235, 255, 158, 158, 145, 93, 170, 131, 103, 158, 31, 93, 239, 54, 30, 141, 28, 41, 139, 119, 133,
94, 48, 122, 122, 98, 196, 194, 90, 112, 105, 68, 185, 214, 164, 253, 255, 152, 91, 81, 83, 123, 157,
123, 100, 107, 107, 143, 158, 163, 135, 24, 13, 112, 205, 193, 83, 116, 163, 82, 87, 151, 117, 0, 0, 28,
95, 183, 235, 204, 169, 119, 139, 138, 159, 185, 110, 152, 192, 206, 255, 245, 255, 253, 157, 198, 189,
112, 71, 107, 211, 203, 105, 155, 209, 109, 108, 126, 106, 134, 101, 159, 133, 98, 255, 255, 251, 131,
20, 139, 115, 80, 136, 119, 217, 157, 68, 155, 83, 51, 77, 0, 0, 0, 0, 0, 0, 11, 108, 176, 171, 140, 156,
197, 162, 137, 141, 126, 112, 74, 68, 76, 93, 168, 201, 223, 214, 134, 115, 88, 71, 113, 93, 130, 205,
124, 49, 91, 146, 199, 172, 78, 66, 51, 20, 92, 133, 98, 71, 10, 0, 47, 137, 160, 141, 89, 61, 54, 100,
191, 142, 90, 141, 97, 57, 102, 116, 94, 115, 165, 171, 221, 255, 179, 159, 177, 127, 58, 32, 68, 96,
106, 118, 85, 82, 93, 63, 138, 173, 77, 71, 101, 134, 125, 98, 174, 157, 138, 109, 32, 98, 55, 94, 253,
171, 152, 181, 130, 119, 163, 210,
```

```
};
```

```
// _____
```

```
void setup()
```

```
{
```

```
display.begin(SSD1306_SWITCHCAPVCC, 0x3C);
```

```
Serial.begin(9600);
```

```
pinMode(SOUND, OUTPUT); //definiert die Soundausgabe Pin 2
```

```
pinMode(LASER1, INPUT); // definiert den Laserinput Pin 3
```

```
pinMode(LASER2, INPUT); // definiert den Laserinput Pin 4
```

```
pinMode(LASER3, INPUT);// definiert den Laserinput Pin 5
pinMode(LASER4, INPUT);// definiert den Laserinput Pin 6

pinMode(LED1, OUTPUT);// Pin 9
pinMode(LED2, OUTPUT);// Pin 10
pinMode(LED3, OUTPUT);// Pin 11
pinMode(LED4, OUTPUT);// Pin 12

}

#define DRAW_DELAY 118
#define D_NUM 47
int i;

void loop() {

#define Size 4

int laserinput[Size] = {digitalRead(LASER1), digitalRead(LASER2), digitalRead(LASER3),
digitalRead(LASER4)};

    display.setTextColor(WHITE);
    display.setTextSize(2);
    display.setCursor(1,0);
    display.println("Viel Spass");
    display.setCursor(40,20);
    display.display();
    display.clearDisplay();

for(int i = 0; i < Size; i++)
{
```

```
if( laserinput[0] == HIGH)
{

    Serial.println("Input1");
    digitalWrite(SOUND,HIGH);

    startPlayback(sample, sizeof(sample));
    digitalWrite(LED1,HIGH);
    Serial.println("Detected!");
    display.setTextColor(WHITE);
    display.setTextSize(2);
    display.setCursor(1,0);
    display.println("Treffer1");
    display.setCursor(40,20);
    display.display();
    display.clearDisplay();

}
```

```
if( laserinput[1] == HIGH)
{

    Serial.println("Input2");
    digitalWrite(SOUND,HIGH);
    /*
    startPlayback(sample, sizeof(sample));
    digitalWrite(LED2,HIGH);
    Serial.println("Detected!");
    display.setTextColor(WHITE);
    display.setTextSize(2);
```

```
display.setCursor(1,0);
display.println("Treffer2");
display.setCursor(40,20);
display.display();
display.clearDisplay();

}

if( laserinput[2] == HIGH)
{

    Serial.println("Input3");
    digitalWrite(SOUND,HIGH);
    startPlayback(sample, sizeof(sample));
    digitalWrite(LED3,HIGH);
    Serial.println("Detected!");
    display.setTextColor(WHITE);
    display.setTextSize(2);
    display.setCursor(1,0);
    display.println("Treffer3");
    display.setCursor(40,20);
    display.display();
    display.clearDisplay();

}

if( laserinput[3] == HIGH)
{
    Serial.println("Input4");
    digitalWrite(SOUND,HIGH);
    startPlayback(sample, sizeof(sample));
```

```
digitalWrite(LED4,HIGH);
Serial.println("Detected!");
display.setTextColor(WHITE);
display.setTextSize(2);
display.setCursor(1,0);
display.println("Treffer4");
display.setCursor(40,20);
display.display();
display.clearDisplay();

}
else
{

}
}

delay(200);
}
```