

Blinkenlights Workshop

Stephan Messlinger, Valentin Ochs

20. März 2017

Startpunkt digitaler Output

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Blink Beispiel: File → Examples → Basics → Blink

```
void setup() {  
    pinMode(13, OUTPUT);  
}  
  
void loop() {  
    digitalWrite(13, HIGH);  
    delay(1000);  
    digitalWrite(13, LOW);  
    delay(1000);  
}
```

Setup

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

`pinMode(pin, modus)` wählt für den Pin mit Nummer `pin` eine von drei Betriebsarten:

- **OUTPUT:** wird für Ausgabe verwendet, z.B. um LEDs zu schalten oder mit anderen Microcontrollern zu sprechen.
- **INPUT:** Die Spannung am Pin kann gelesen werden.
- **INPUT_PULLUP:** Wie INPUT, aber der Pin wird intern auf die Versorgungsspannung gezogen.

digitalWrite und delay

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

`digitalWrite(pin, zustand)` setzt bei einem auf Output gestellten Pin die Ausgangsspannung:

- 0 Volt für LOW
- 5 Volt für HIGH (oder was auch immer die aktuelle Versorgungsspannung ist)

`delay(ms)` tut ms Millisekunden lang nichts.

Andere Blink Muster

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Zwei Sekunden lang an, eine halbe aus.

Andere Blink Muster

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Zwei Sekunden lang an, eine halbe aus.

```
digitalWrite(13, HIGH);  
delay(2000);  
digitalWrite(13, LOW);  
delay(500);
```

Schnelleres Blinken

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Was passiert, wenn man die Zeiten ganz niedrig setzt?

Schnelleres Blinken

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Was passiert, wenn man die Zeiten ganz niedrig setzt?
→ Man sieht kein Blinken mehr

Schnelleres Blinken

Blinkenlights Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Was passiert, wenn man die Zeiten ganz niedrig setzt?

→ Man sieht kein Blinken mehr

Was passiert, wenn die Zeitverhältnisse geändert werden?

Schnelleres Blinken

Blinkenlights Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Was passiert, wenn man die Zeiten ganz niedrig setzt?

→ Man sieht kein Blinken mehr

Was passiert, wenn die Zeitverhältnisse geändert werden?

→ Dimmen

analogWrite

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

`analogWrite(pin, wert)` schaltet den Pin automatisch an und aus, mit variablen An-/Aus-Zeiten
→ Pulsweitenmodulation

- Frequenz: Etwa 490 Hz
- Wertebereich: 0 bis 255
- Nur auf Pins 3, 5, 6, 9, 10, und 11.
- Die PWM Pins sind auf dem Arduino mit ~ markiert.

PWM Funktionsweise: Zähler + Vergleich

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

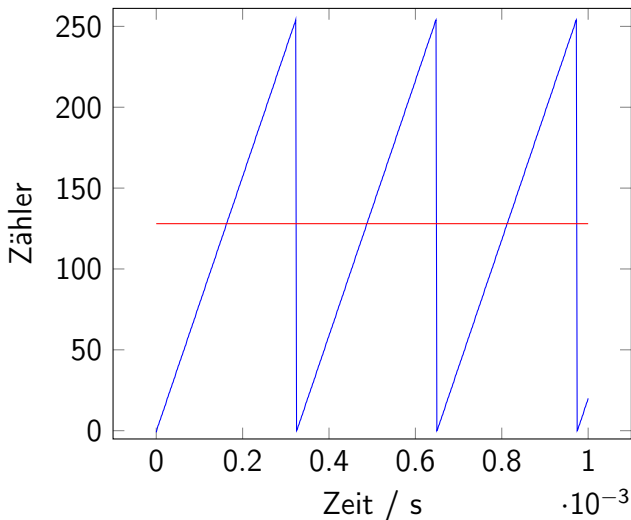
Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren



PWM, Schwellwert 128

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

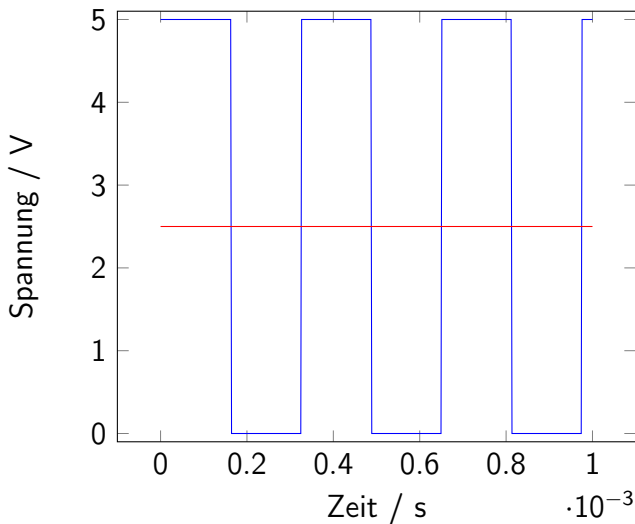
Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren



PWM, Schwellwert 16

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

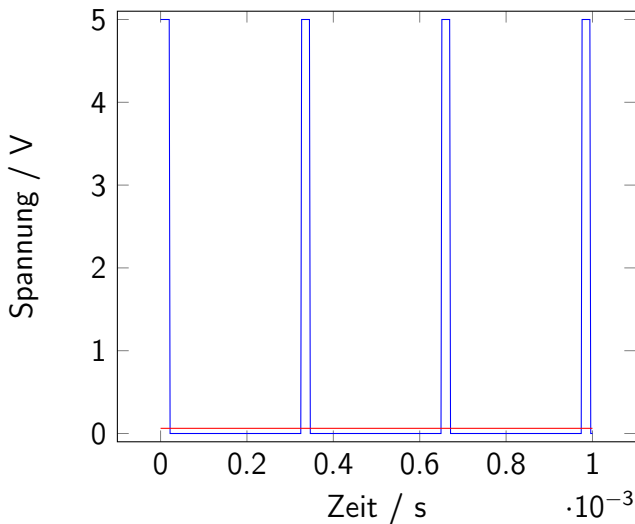
Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren



Einfacher PWM Code

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

```
int const led_pin = 11;
void setup() {
    pinMode(led_pin, OUTPUT);
}
void loop() {
    // Zeit seit Beginn des Programms
    unsigned long time = millis();
    // Berechne eine Sägezahn mit 0.1 Hz
    int value = 255 * time / 4000;
    // Verwende den Wert als Schwellwert
    analogWrite(led_pin, value);
}
```

Datentypen (1)

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

- `unsigned long time` und `int value` definieren Variablen.
- `unsigned long` und `int` sind die Typen, `time` und `value` die Namen, bzw. Identifier.
- Normal sind Typen vorzeichenbehaftet, durch `unsigned` haben sie einen nicht-negativen Wertebereich
- Kleinere Datentypen sind schneller

Typ	Wertebereich	unsigned Wertebereich
<code>char</code>	-2^7 bis $2^7 - 1$	0 bis $2^8 - 1$
<code>int</code>	-2^{15} bis $2^{15} - 1$	0 bis $2^{16} - 1$
<code>long</code>	-2^{31} bis $2^{31} - 1$	0 bis $2^{32} - 1$

Datentypen (2)

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

- `float` für Gleitkommazahlen (sehr langsam!)
- `double` für genauere Gleitkommazahlen (unglaublich langsam)
- `const` Suffix (z.B. `int const`) für Werte, die sich nach ihrer Definition nicht ändern. Vorteile:
 - Etwas lesbarer
 - Kann zu schnelleren Programmen führen
- Zu große (oder kleine) Werte führen zu Überlauf:
 - Bei `char`: $127+1 \rightarrow -128$
 - Bei `unsigned char`: $0 - 1 \rightarrow 255$

PWM Frequenz

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

490 Hz sind bei schnellen Bewegungen sichtbar.

Bestimmung der Frequenz: $\text{Taktfrequenz} / \text{Vorteiler} / \text{Zählergröße}$

- Taktfrequenz: 16 MHz
- Zählergröße:
 - 256 für Pins 5 und 6
 - 510 für 3, 9, 10, 11

PWM Vorteile: Timer 0, Pins 5 und 6

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Einstellung	Teiler	Frequenz
0x01	1	62500
0x02	8	7813
0x03	64	977
0x04	256	244
0x05	1024	61

Einstellen durch

`TCCR0B = (TCCR0B & 0b11111000) | Einstellung`

PWM Vorteile: Timer 1, Pins 9 und 10

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Einstellung	Teiler	Frequenz
0x01	1	31373
0x02	8	3921
0x03	64	490
0x04	256	123
0x05	1024	31

Einstellen durch

`TCCR1B = (TCCR0B & 0b11111000) | Einstellung`

PWM Vorteile: Timer 2, Pins 11 und 3

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Einstellung	Teiler	Frequenz
0x01	1	31373
0x02	8	3921
0x03	32	980
0x04	64	490
0x05	128	245
0x06	256	123
0x07	1024	31

Einstellen durch

`TCCR2B = (TCCR2B & 0b11111000) | Einstellung`

Vorsicht

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Frequenzänderung beeinflusst nicht nur LEDs, sondern alles, was an dem Timer hängt! Servos, Tonerzeugung, etc.

Besonders wichtig: Timer 0 für `millis()` und `delay()`.
Standardvorteiler: 64. Bei Änderungen Zeiten entsprechend anpassen (Vervierfachen bei 256...)

Startpunkt digitaler Input

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Button Beispiel: File → Examples → Digital → Button
Geht nicht nur mit einfachen Schaltern und Tastern,
sondern auch z.B. einer Lichtschranke oder
Reed-Schaltern.

`digitalRead(pin):`

- HIGH falls Spannung an pin etwa 2.6 V oder höher
- LOW falls Spannung an pin 2.1 V oder tiefer
- Nur bei 5 V Versorgungsspannung, sonst andere Werte

Schaltplanvarianten

Blinkenlights Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

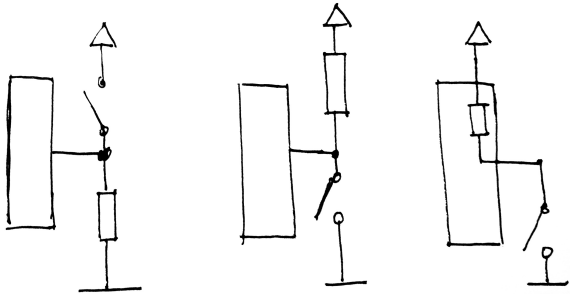
Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren



Unterbrechbare Abläufe starten (1)

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

```
unsigned long button_time = 0;
bool running = false;
void loop() {
    if(digitalRead(button_pin) == HIGH) {
        running = true;
        button_time = millis();
    }
    if(running) {
        running = do_stuff(millis() - start_time);
    }
}
```

Unterbrechbare Abläufe starten (2)

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

```
bool do_stuff(unsigned long time_point)
{
    if(time_point < 100) {
        digitalWrite(led_pin, HIGH);
    } else if(time_point < 200) {
        digitalWrite(led_pin, LOW);
    } else if(time_point < 1000) {
        digitalWrite(led_pin, HIGH);
    } else {
        return false;
    }
    return true;
}
```

Prellen

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

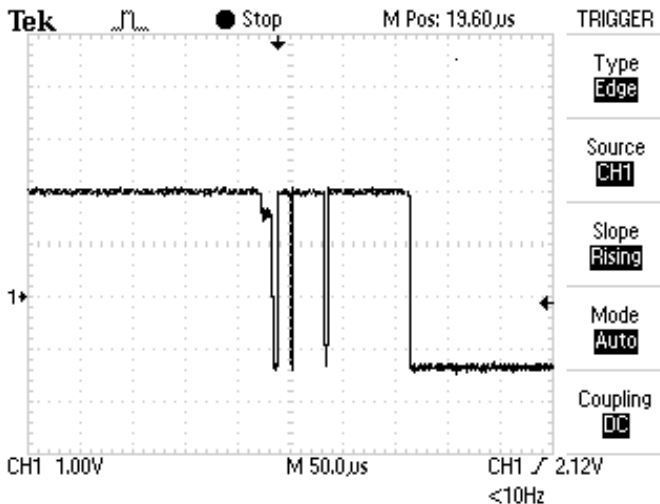
Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren



Entprellen

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Auch: Debouncing

- Hardware Lösung: Tiefpassfilter mit Kondensator
- Software Lösung: Mehrmals Wert auslesen und warten, bis er sich nicht mehr ändert
- Hier ohne weitere Vertiefung, aber ihr wisst jetzt, wonach man suchen muss :)

Startpunkt analoger Input

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

AnalogInput Beispiel: File → Examples → Analog →
AnalogInput

analogRead

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

`analogRead(pin)`: 0-1023 für 0-5 Volt an Pin `pin`.

Kombination mit analogWrite

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

```
void loop() {  
    int value = analogRead(A0) / 4;  
    analogWrite(3, value);  
}
```


An den PC senden

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

```
void setup() {  
    Serial.begin(115200);  
}  
void loop() {  
    Serial.print("Aktueller Wert: ");  
    Serial.println(analogRead(A0));  
}
```

Auch zur Fehlersuche nützlich!

Die Arduino IDE hat einen Plotter, mit dem man den zeitlichen Verlauf von Zahlen beobachten kann.

Spannungsbereich

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Maximale Spannung: Versorgungsspannung

Spannungsbereich

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Maximale Spannung: Versorgungsspannung
Darüber: Spannungsteiler

Stephan

Blinkenlights Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Valentin darf sich ausruhen. Ihr nicht.

Vorteile:

- Einfach
- PWM (bei bis zu 6) möglich

Nachteile:

- 1 Pin pro LED
- Ab 7 LEDs kein PWM mehr (oder nur in Gruppen)
- 1 RGB LED braucht 3 Pins

Matrix

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Vorteile:

- Kann je nach Methode mit n Pins bis zu $n^2 - n$ LEDs ansteuern

Nachteile:

- Kompliziert
- Niedrige Wiederholrate
- Reduzierte Helligkeit
- Bei größeren Spitzenströmen werden externe Treiber benötigt
- Kein (hardware-beschleunigtes) Dimmen

Schieberegister

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

- Englisch: Shift register
- Mehrere Ausgänge, z.B. 8
- Digitale Steuerung, z.B. SPI oder I2C
- Zu viele Werte → alte Werte werden weitergeschoben

Vorteile:

- Einfach
- Benötigt wenige (i.d.R. < 4) Pins
- Leicht erweiterbar

Nachteile:

- Kein (hardware-beschleunigtes) Dimmen
- Wiederholrate sinkt mit $1/n$

WS2812, APA102...

Blinkenlights Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

- Mehrere LEDs auf Streifen
- Ähnlich zu Schieberegistern
- Eingebaute Logik zum Dimmen
- Ansteuerung durch fertige Libraries

Libraries

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

- Sketch → Include Library → Manage Libraries
- Modularer Code, bei Arduino häufig zum Ansteuern von externer Hardware
- Für WS2812: Adafruit NeoPixel
- Für APA102: APA102

Beispielcode

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

```
#include <Adafruit_NeoPixel.h>

Adafruit_NeoPixel strip(144, 13, NEO_GRB + NEO_K
int i = 0;
void setup() {
    strip.begin();
}
void loop() {
    strip.setPixelColor(i, 255, 0, 0);
    strip.show();
    delay(10);
    strip.setPixelColor(i, 0, 0, 0);
    i++;
    if(i == 144) i = 0;
}
```

Arrays

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

... speichern viele Werte gleichen Typs unter einem Namen. Das erste Element hat Index 0.

Beispiel:

```
int many_values[20];  
for(int i = 0; i < 20; i++)  
    many_values[i] = i;  
Serial.print(many_values[0]+many_values[19]);
```

Laufender Regenbogen (1)

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

```
#include <Adafruit_NeoPixel.h>
Adafruit_NeoPixel strip(144, 13, NEO_GRB + NEO_KHZ800);
uint32_t colors[144];
int i = 0;
void setup() {
    strip.begin();
    for(i = 0; i < 48; i++) {
        unsigned char v = i*255/48;
        colors[i] = strip.Color(255-v, v, 0);
        colors[i+48] = strip.Color(0, 255-v, v);
        colors[i+96] = strip.Color(v, 0, 255-v);
    }
}
void loop() {
    for(int j = i; j < 144-i; j++)
        strip.setPixelColor(i+j, colors[j]);
    for(int j = 144-i; j < 144; j++)
        strip.setPixelColor(i+j-144, colors[j]);
    strip.show();
    i++;
    if(i == 144) i = 0;
}
```

Macht Stephan

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

**Stromversor-
gung**

Sensoren

Däumchen dreh

Anschluss von Sensoren

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

- Analog: Sensor gibt eine Spannung aus, die gemessen wird
 - Unkompliziert, aber durch den Arduino eingeschränkte Genauigkeit, Präzision, Geschwindigkeit, Anzahl von Sensoren
- Digital: Sensor wird durch ein serielles Interface (häufig SPI oder I2C) an den Arduino angeschlossen.
 - Erlaubt manchmal auch Einstellungen (Messfrequenz, -bereich)
 - Etwas komplizierter zu programmieren
 - Viele Sensoren an wenigen Pins möglich

Sensorbeispiele

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

- Beschleunigung
- Drehrate
- Magnetfeld
- Spannung
- Luftfeuchtigkeit, Temperatur, Druck
- Licht
- Position (GPS)

Sensoren im Arduino

Blinkenlights
Workshop

Stephan
Messlinger,
Valentin Ochs

Digital Out

Analog Out

Digital In

Analog In

Stephan

Mehrere LEDs

LED Streifen

Stromversor-
gung

Sensoren

Spannung (analoger Input) und Temperatur (interne Temperatur, wird über den Analog-Digital-Wandler gemessen).

```
void setup() {  
    Serial.begin(115200);  
    // Temperaturmessung einrichten:  
    ADMUX = (_BV(REFS1) | _BV(REFS0) | _BV(MUX3));  
    ADCSRA |= _BV(ADEN);  
}  
  
void loop() {  
    ADCSRA |= _BV(ADSC); // Messung starten  
    while(ADCSRA & _BV(ADSC)) { } // Warte  
    Serial.println(ADCW); // Wert ausgeben  
}
```