

# Blinkenlights Workshop

Stephan Messlinger  
Valentin Ochs

2017-03-20

# Startpunkt digitaler Output

Blinkenlights  
Workshop

Stephan  
Messlinger  
Valentin Ochs

Digital Out

PWM

Digital In

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

PWM

Digital In

`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.

TODO: Bild zu Pullups?

# digitalWrite und delay

Blinkenlights  
Workshop

Stephan  
Messlinger  
Valentin Ochs

Digital Out

PWM

Digital In

`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

PWM

Digital In

Zwei Sekunden lang an, eine halbe aus.

# Andere Blink Muster

Blinkenlights  
Workshop

Stephan  
Messlinger  
Valentin Ochs

Digital Out

PWM

Digital In

Zwei Sekunden lang an, eine halbe aus.

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

# Mehrere LEDs

Blinkenlights  
Workshop

Stephan  
Messlinger  
Valentin Ochs

Digital Out

PWM

Digital In

hier einfuegen?  
Timer + if

# Schnelleres Blinken

Blinkenlights  
Workshop

Stephan  
Messlinger  
Valentin Ochs

Digital Out

PWM

Digital In

auf 1 ms oder so setzen, Verhaeltnisse aendern



# analogWrite

Blinkenlights  
Workshop

Stephan  
Messlinger  
Valentin Ochs

Digital Out

PWM

Digital In

`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

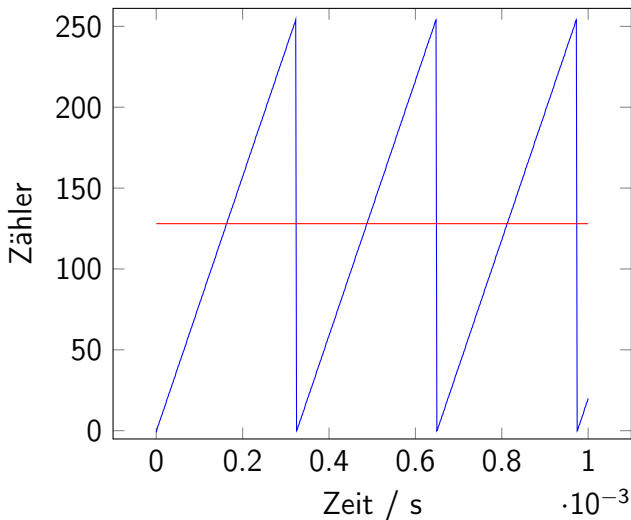
Blinkenlights  
Workshop

Stephan  
Messlinger  
Valentin Ochs

Digital Out

PWM

Digital In



# PWM, Schwellwert 128

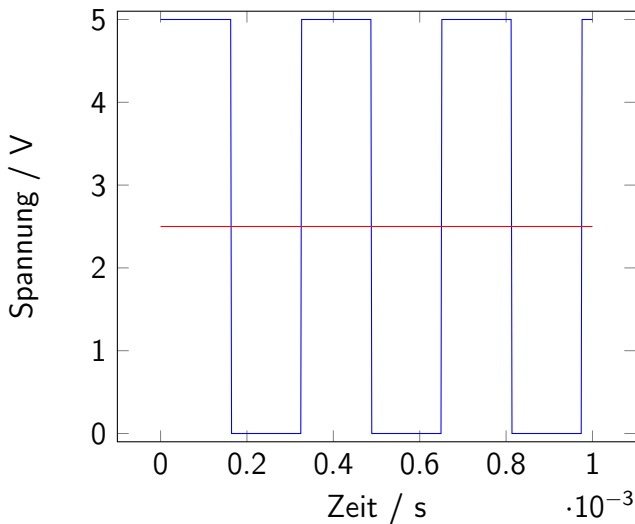
Blinkenlights  
Workshop

Stephan  
Messlinger  
Valentin Ochs

Digital Out

PWM

Digital In



# PWM, Schwellwert 16

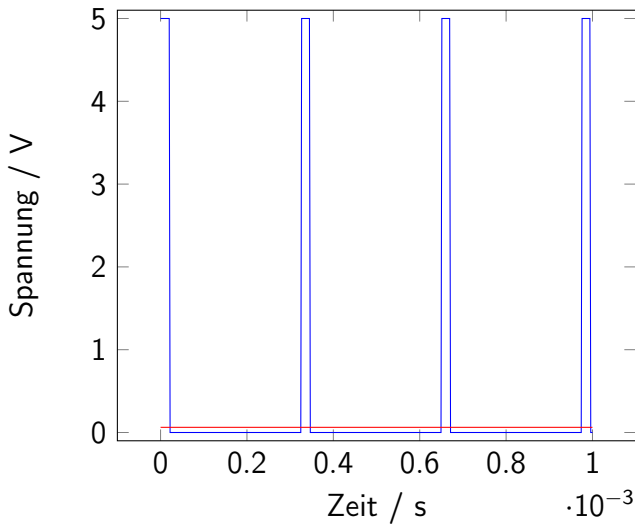
Blinkenlights  
Workshop

Stephan  
Messlinger  
Valentin Ochs

Digital Out

PWM

Digital In



# Einfacher PWM Code

Blinkenlights  
Workshop

Stephan  
Messlinger  
Valentin Ochs

Digital Out

PWM

Digital In

```
void setup() {  
    pinMode(11, OUTPUT);  
}  
void loop() {  
    // Zeit seit Beginn des Programms  
    unsigned long time = millis();  
    // Berechne einen Sinus mit 0.1 Hz  
    int value = 127.5 *  
        (1+sin(millis * 0.001 * 0.1 * 2 * 3.1416));  
    // Verwende den Wert als Schwellwert  
    analogWrite(11, value);  
}
```

# PWM Frequenz

Blinkenlights  
Workshop

Stephan  
Messlinger  
Valentin Ochs

Digital Out

PWM

Digital In

490 Hz sind bei schnellen Bewegungen sichtbar.

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

Taktfrequenz: 16 MHz Maximalwert: 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

PWM

Digital In

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

PWM

Digital In

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

PWM

Digital In

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

PWM

Digital In

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

PWM

Digital In

Button Beispiel: File → Examples → Digital → Blink

# Schaltplan

Blinkenlights  
Workshop

Stephan  
Messlinger  
Valentin Ochs

Digital Out

PWM

Digital In

- pulldown, extern
- pullup, extern
- pullup, intern

# Unterbrechbare Abläufe starten

Blinkenlights  
Workshop

Stephan  
Messlinger  
Valentin Ochs

Digital Out

PWM

Digital In

millis() und so?