

# Python II - GPIO

Gerätesteuerung mit Python und RaspberryPi

# GPIO

- General Purpose Input/Output
- *Digitale* Ein-/Ausgänge
- Raspberry Pi: teilweise doppelte Belegung (UART, i2c, SPI)

# Libraries

- RPi.GPIO
- python-sysfs-gpio
- pyA20
- wiringpi2 / wiring-x86
- webiopi

# Sicherheit

- `RPi.GPIO.cleanup()`
  - bei Exceptions
  - am Ende des Skripts

```
try:
```

```
    ... code ...
```

```
except:
```

```
    RPi.GPIO.cleanup()
```

# Sicherheit

- GPIO-Pins mit 3.3V
  - auf keinen Fall 5V anlegen!
- 3.3V out (Pin 1) nicht mit mehr als 50mA belasten
- 5V nur soweit nötig benutzen
  - Läuft über Polyfuse, zu viel Last kann den Pi (temporär) lahmlegen

# Lesen und Schreiben

- nur digitale Werte → 1 oder 0
  - `RPi.GPIO.LOW` | `RPi.GPIO.HIGH`

```
RPi.GPIO.setup(channel, GPIO.IN[, pull_up_down=GPIO.  
PUD_UP])
```

```
RPi.GPIO.setup(channel, GPIO.OUT)
```

```
RPi.GPIO.input(channel)
```

```
RPi.GPIO.output(channel, value)
```

# Lesen und Schreiben

- nur digitale Werte → 0 oder 1
  - `RPi.GPIO.LOW` | `RPi.GPIO.HIGH`

```
RPi.GPIO.wait_for_edge(channel, RPi.GPIO.RISING)
```

```
RPi.GPIO.add_event_detect(channel, RPi.GPIO.RISING, callback)
```

```
RPi.GPIO.remove_event_detect(channel)
```

# Ereigniserkennung

- Buttons, Schalter, ...
- Debouncing notwendig
  - 0.1 $\mu$ F Kondensator
  - Software-Debouncing

```
RPi.GPIO.add_event_detect(  
    channel, RPi.GPIO.RISING,  
    callback=cb_function,  
    bouncetime=200)
```



# Hello World

- LED blinken lassen
- GPIO → LED → Widerstand ( $270\Omega$ ) → GND

# Hello World

```
from RPi import GPIO
import time

PIN_LED = 11
GPIO.setmode(GPIO.BOARD)
GPIO.setup(PIN_LED, GPIO.OUT)

try:
    while True:
        GPIO.output(PIN_LED, GPIO.HIGH)
        time.sleep(1)
        GPIO.output(PIN_LED, GPIO.LOW)
        time.sleep(1)
except:
    GPIO.cleanup()
```

# Hello World - Mit PWM

```
PIN_LED = 11
GPIO.setmode(GPIO.BOARD)
GPIO.setup(PIN_LED, GPIO.OUT)
pwm = GPIO.PWM(PIN_LED, 100) // 100Hz PWM

try:
    pwm.start(0) // 0% duty cycle
    while True:
        for i in range(0,100):
            pwm.changeDutyCycle(i) // i% duty cycle
            time.sleep(0.3)
except:
    GPIO.cleanup()
```

50% duty cycle



75% duty cycle



25% duty cycle



# Performance

- **Polling mit while True Schleifen**
  - Verbraucht CPU
  - sollte nur mit sleep verwendet werden
- **Threading.Timer**
  - geringer CPU-Verbrauch
  - hoher Ressourcen-Verbrauch bei vielen Timer-Objekten

# Performance

- Interrupts
  - `RPi.GPIO.add_event_detect`
  - `RPi.GPIO.wait_for_edge`
  - seit Version 0.5.2a

<http://pad.technikkultur-erfurt.de/p/Python-II-GPIO>

- \* Python auf dem Raspbian - GPIO <http://www.thirdeyevis.com/pi-page-1.php>
- \* GPIO Geschwindigkeiten mit unterschiedlichen Sprachen <http://codeandlife.com/2012/07/03/benchmarking-raspberry-pi-gpio-speed/>
- \* PWM mit Python <https://www.youtube.com/watch?v=uUn0KWwwkq8>
- \* GPIO cheat sheet [https://www.cl.cam.ac.uk/projects/raspberrypi/tutorials/robot/cheat\\_sheet/](https://www.cl.cam.ac.uk/projects/raspberrypi/tutorials/robot/cheat_sheet/)
- \* RaspberryPi Pinout <http://pi.gadgetoid.com/pinout>
- \* RaspberryPi Low-Level Peripherals [http://elinux.org/RPi\\_Low-level\\_peripherals#Introduction](http://elinux.org/RPi_Low-level_peripherals#Introduction)
- \* <https://github.com/derekstavis/python-sysfs-gpio>
- \* <https://pypi.python.org/pypi/pyA20>
- \* <https://github.com/emutex/wiring-x86>
- \* <https://code.google.com/p/webiopi/>
- \* <https://github.com/quick2wire/quick2wire-python-api>
- \* Raspberry Pi 2 B (neu): <http://de.rs-online.com/web/p/processor-microcontroller-development-kits/8326274/>
- \* AdaFruit Raspi Python Repository <https://github.com/adafruit/Adafruit-Raspberry-Pi-Python-Code>
- \* Raspberry Pi programmieren mit Python <http://owncloud.nincns.de/public.php?service=files&t=048e7ada683159d301ebe409c24e00e8>
- \* Raspberry Pi A Beginners Guide <http://owncloud.nincns.de/public.php?service=files&t=690d94b4818ce7e03c3e5d7c2616beeb>
- \* Raspberry Pi Guide for Simple Python Projects <http://owncloud.nincns.de/public.php?service=files&t=c0b436ea9b57fbddda73188677af77f4>