

# RIOTNerf

---

WS 2016/17 - RIOT im Internet of Things - Bachelor Project  
of Computer Science and Electrical Engineering



# Inhaltsverzeichnis

- Allgemeines
  - Intro
  - Mitarbeiter
- Komponenten
  - Arm
  - Zielscheibe
  - Server

# Inhaltsverzeichnis

- Metainformationen
  - Entwicklungsprozess
  - Erkenntnisgewinn
  - Strukturen
  - Zwischenmenschliches
  - Workflow
  - Tooling

# Allgemeines

---

# Intro

Webgesteuertes Multiplayer-Spiel

2 Laserguns

2 Zielscheiben

# Mitarbeiter

- Andreas - Communication IoT side
- Jerom - Server side
- Michael - Server side
- Phillip - IoT development
- Darjush - IoT development
- Martin - IoT development, HW Setup



# Andreas

- + Planung und Definition der Kommunikationsschnittstellen
- + Planung und Definition der Kommunikationsabläufe
- + Tätigkeiten im Entwicklungsprozess
  - ◆ Git Issue-tracking
  - ◆ C-Programmierung
    - CoAP Kommunikation Target
    - CoAP Kommunikation Lasergun

# Martin

- + Planung und Konstruktion des gesamten Aufbaus
- + Erstellung der Stückliste
- + Erstellung der Schaltpläne
- + Wikipflege
- + Tätigkeiten im Entwicklungsprozess
  - ◆ Planung Schnittstelle IoT <> Servo
  - ◆ Planung Schnittstelle IoT <> isl29125
  - ◆ C-Programmierung
    - Target Treiber
    - Interrupt-Implementierung für isl29125
    - Kommunikation IoT <> isl29125



# Michael

- + Aufsetzen des Raspberry Pi
  - + Raspian
  - + Einrichtung von 6LoWPAN / Coap
- + Bereitstellen notwendiger Software
  - + Node.js
  - + Socket.io
- + Tätigkeiten im Entwicklungsprozess
  - + GUI
  - + Web- / Coapserver
- + Game Logic

# Jerom

- + Planung der Kommunikationsschnittstelle Server ↔ RiotBoard
- + Tätigkeiten im Entwicklungsprozess
  - ◆ NodeJS-Server (RaspberryPi) implementiert
    - HTTP-Server
      - Implementation Spiellogik
      - Socketschnittstelle Client - Server
    - Coap-Server
      - Steuerung der Laser
      - Dispatching von Zielscheiben

# Darjush

- + HW Aufbau
- + Planung Wiki
- + Tätigkeiten im Entwicklungsprozess
  - ◆ IoT Development
    - Entwicklung des gun controllers
    - Testen und Kalibrieren der Servos

# Phillip

- + Festkommabibliothek
- + Unit testing framework
- + Tätigkeiten im Entwicklungsprozess
  - ◆ Git
  - ◆ C-Programmierung
    - Code review
    - Refactoring
    - Fehlerbehebung
    - Beratung und Hilfe
      - API-Design
      - Implementation

# Phillip

## + Erforschung von RIOT

- ◆ UDP
- ◆ C++-Entwicklung
  - Threadpool
  - UDP socket

## + Tooling

- ◆ Code indentation script

## + Dokumentation

# Komponenten

---

# 2-Achsen Lasergun

Der Arm kann durch zwei Servomotoren, die jeweils über einen PWM-Ausgang vom RIOT-Device gesteuert werden, je 180° in x- und y-Richtung bewegt werden.

Die Laserdiode ist über einen GPIO-Ausgang schaltbar (Transistorschaltung).

Den RIOT-eigenen Servo-Treiber nutzend, existiert eine projektspezifische Middleware (pHAL), diese bietet Calls wie z.B. ein winkelgenaues justieren oder eine schrittweise Steuerung innerhalb der Schranken.

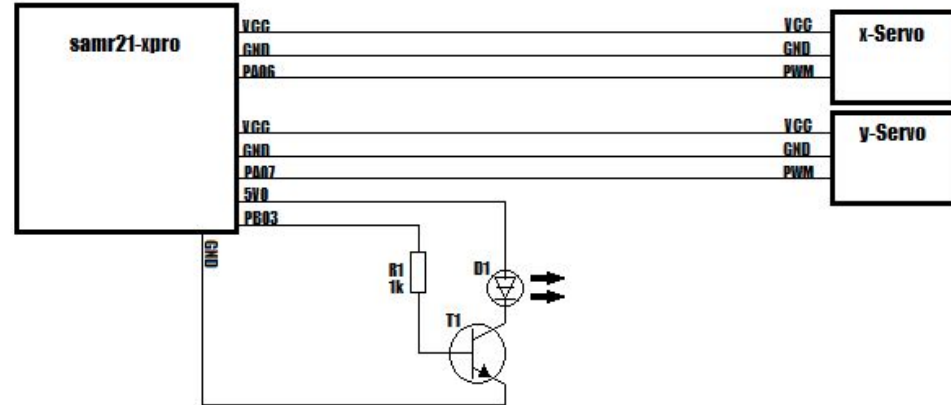


# 2-Achsen Lasergun

Der Arm kann durch zwei Servomotoren, die jeweils über einen PWM-Ausgang vom RIOT-Device gesteuert werden, je 180° in x- und y-Richtung bewegt werden.

Die Laserdiode ist über einen GPIO-Ausgang schaltbar (Transistorschaltung).

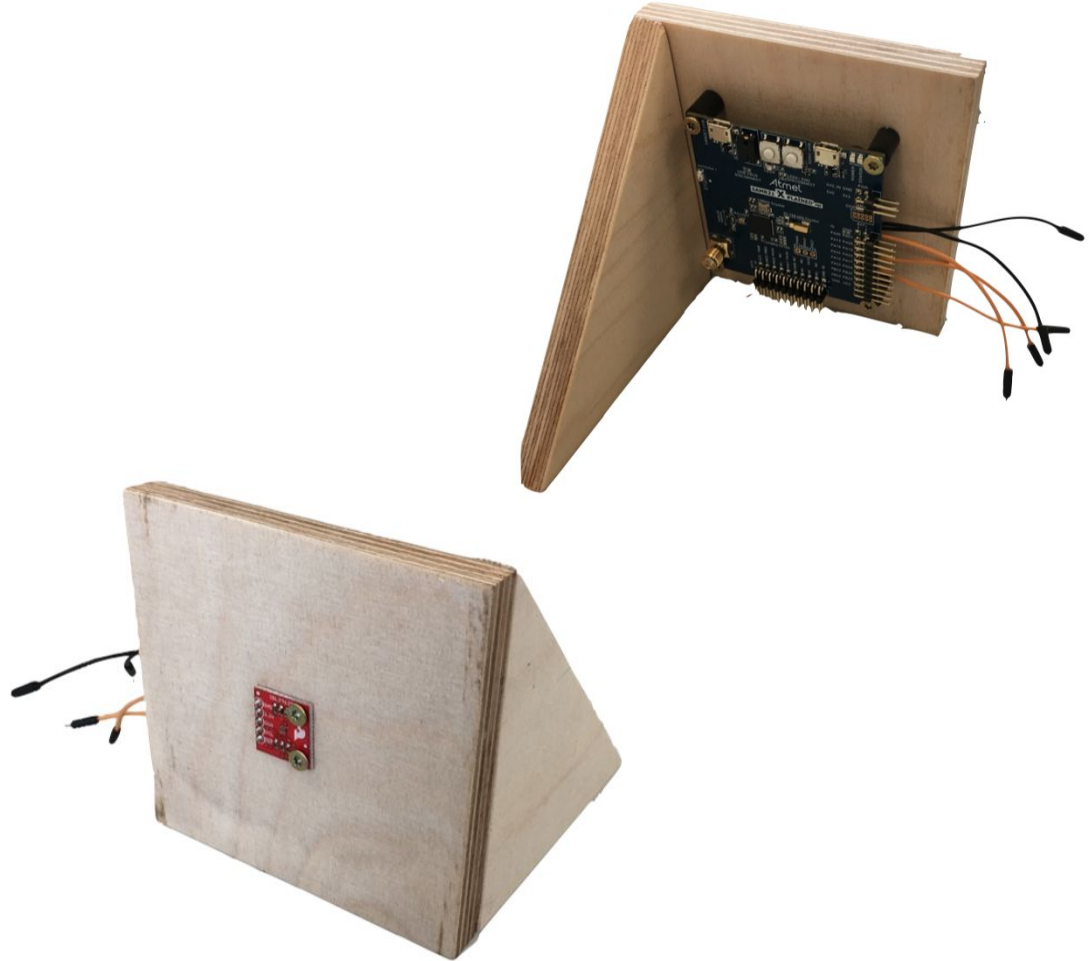
Den RIOT-eigenen Servo-Treiber nutzend, existiert eine projektspezifische Middleware (pHAL), diese bietet Calls wie z.B. ein winkelgenaues justieren oder eine schrittweise Steuerung innerhalb der Schranken.





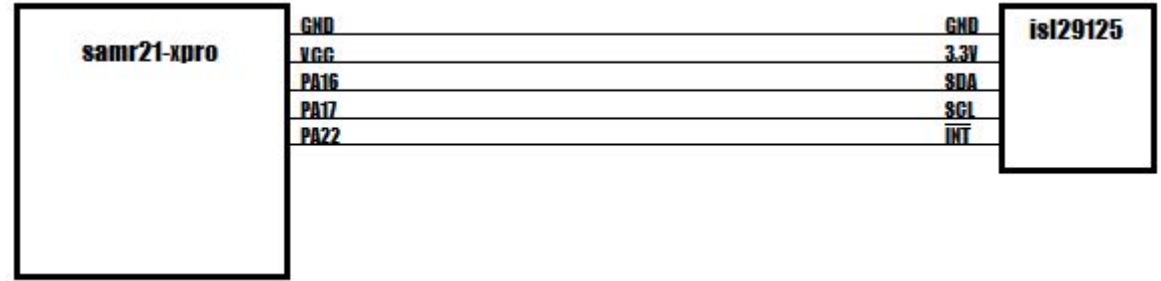
# Zielscheibe

Der auf einer Holzplatte verbaute und über I<sup>2</sup>C verbundene RGB-Sensor isl29125 setzt ein IRQ für das RIOT-Device, sobald der rote Laser des Arms auf den RGB-Sensor gerichtet ist und dadurch ein stark erhöhter Rotwert registriert wird. Ausgelöst durch den Interrupt wird via Microcoap ein Treffersignal an das RIOT-Device der gerade aktiven Lasergun gefunkt.



# Zielscheibe

Der auf einer Holzplatte verbaute und über I<sup>2</sup>C verbundene RGB-Sensor isl29125 setzt ein IRQ für das RIOT-Device, sobald der rote Laser des Arms auf den RGB-Sensor gerichtet ist und dadurch ein stark erhöhter Rotwert registriert wird. Ausgelöst durch den Interrupt wird via Microcoap ein Treffersignal an das RIOT-Device der gerade aktiven Lasergun gefunkt.



# Server

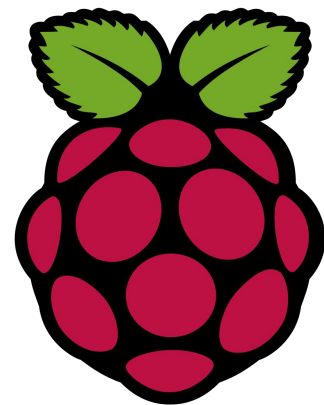
Raspberry Pi 1B & Transceiver für 6LoWPAN

Raspbian Jessie Lite & tmux

Tutorial (RIOT-Makers/Wpan-Raspbian)

Channel & RADVD & Coap

Node.js / socket.io



# Server

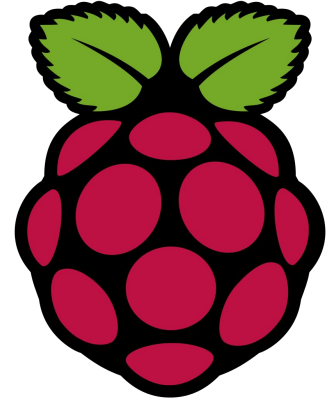
Das Backend ist als NodeJS Server realisiert

Läuft auf Raspberry Pi

Website ermöglicht Nutzern die Steuerung  
der Laser

Median der Abstimmungen pro Laser  
ermittelt.

per Coap-Protokoll mittels 6LoWPAN an die  
Laser geschickt



# Kommunikation

## Layer 7: CoAP

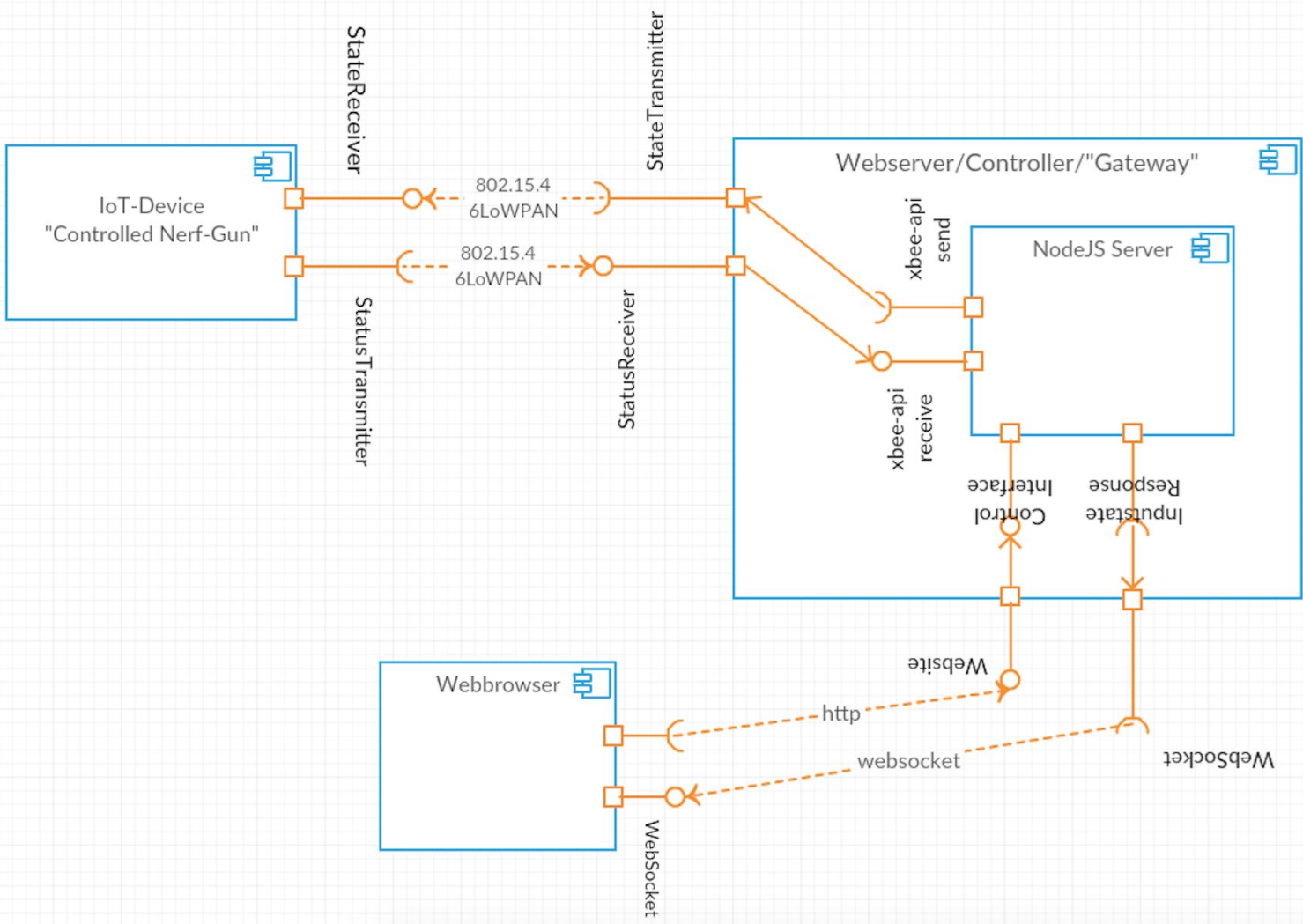
“The Constrained Application Protocol (CoAP) is a specialized web transfer protocol for use with constrained nodes and constrained networks in the **Internet of Things**.”

## Layer 4: UDP

## Layer 3: 6LoWPAN

*IPv6 over Low power Wireless Personal Area Networks*

[...]mechanisms that allow IPv6 packets to be sent and received over IEEE 802.15.4[...]



# RIOTNerf

WS 2016/17 - RIOT im Internet of Things - Bachelor Project of Computer Science and Electrical Engineering

Main Page	Related Pages	Modules	Namespaces	Data Structures	<b>Files</b>	Examples
File List	Globals					

## Macros

- #define STEP\_SIZE (8)**  
Value that represents a single servo step. [More...](#)
- #define PHAL\_INIT\_LASER\_ERROR (-1)**  
Error code returned by pHAL\_init to indicate failure to initialize laser. [More...](#)
- #define PHAL\_INIT\_SERVO1\_ERROR (-2)**  
Error code returned by pHAL\_init to indicate failure to initialize servo 1. [More...](#)
- #define PHAL\_INIT\_SERVO2\_ERROR (-3)**  
Error code returned by pHAL\_init to indicate failure to initialize servo 2. [More...](#)

## Functions

- int pHAL\_init (void)**  
pHAL\_init The function that initializes the laser and servos to set up the pHAL. [More...](#)
- void set\_h (int angle)**  
set\_h Sets the horizontal servo to the angle passed to the function. [More...](#)
- void set\_v (int angle)**  
set\_v Sets the vertical servo to the angle passed to the function. [More...](#)
- void sethv (int h, int v)**  
sethv Sets both servos to positions given. [More...](#)
- void step\_l (void)**  
step\_l Performs a single step to the left. [More...](#)
- void step\_r (void)**  
step\_r Performs a single step to the right. [More...](#)
- void step\_u (void)**  
step\_u Performs a single upward step. [More...](#)

# Dokumentation

+ API-Dokumentation

# Metainformationen

---



# Entwicklungsprozess

In 1-Wochen-Abschnitten jeweils versucht lauffähiges Inkrement zu erarbeiten

Aufteilung in 3 Teams, mit dedizierten Schnittstellenbeauftragten

Priorisierung der einzelnen Arbeitspakete

**SOFTWARE  
DEVELOPMENT**



# Erkenntnisgewinn

*Plans are worthless, but planning is everything* - Dwight D. Eisenhower

RIOT bietet trotz des jungen Alters bereits eine erstaunlich verlässliche Basis

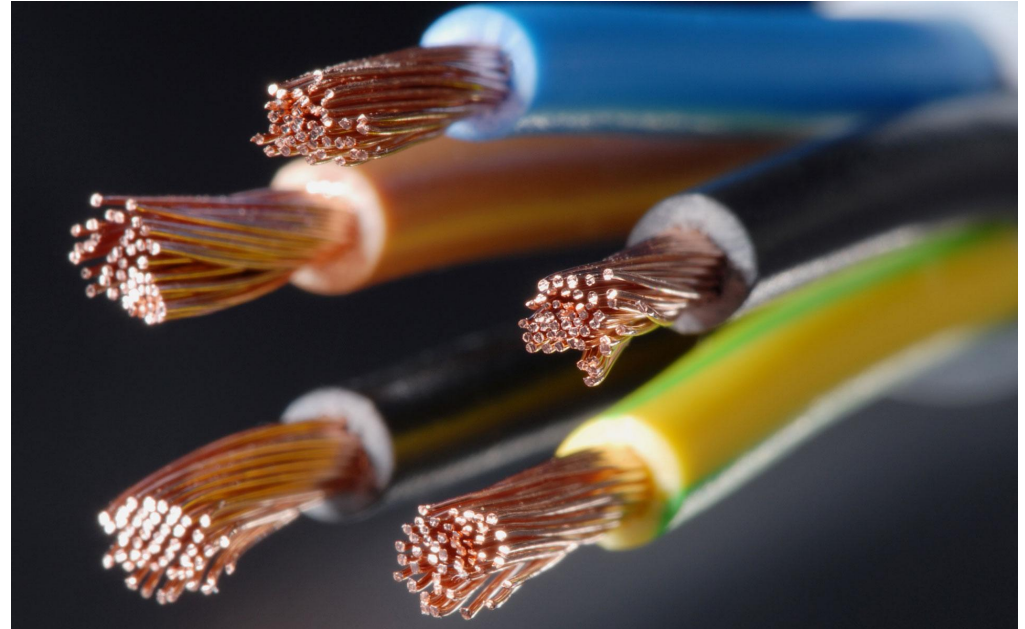
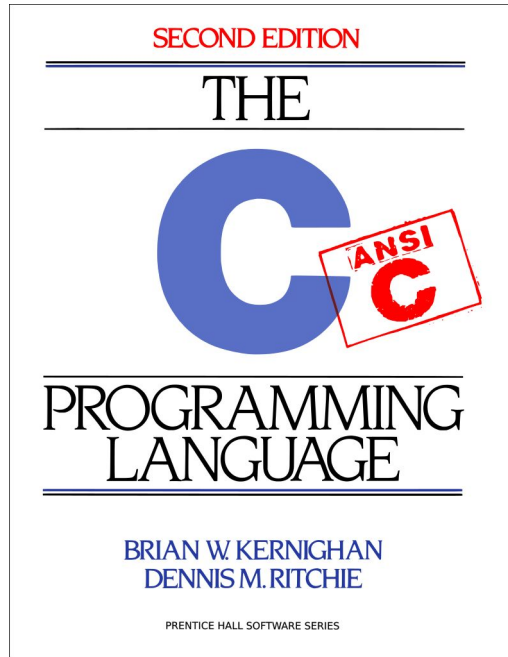
RGB-Sensor erkennt Treffer überraschend gut

Servomotoren sind relativ unpräzise

The logo for RIOT features the letters 'R' and 'I' in a large, red, stylized font. The 'R' is a thick, rounded shape with a loop at the top. The 'I' is a thick, vertical bar with a small horizontal tick at the bottom. To the right of these, the letters 'O', 'T', and 'I' are in a green, sans-serif font. The 'O' is a simple circle, the 'T' is a simple vertical bar with a horizontal top bar, and the 'I' is a simple vertical bar. The overall logo is 'RIOT'.

# Zwischenmenschliches

- Gute Kommunikation => schnelle Fortschritte
- Zusammenarbeit zwischen Elektrotechnikern und Informatikern funktioniert gut

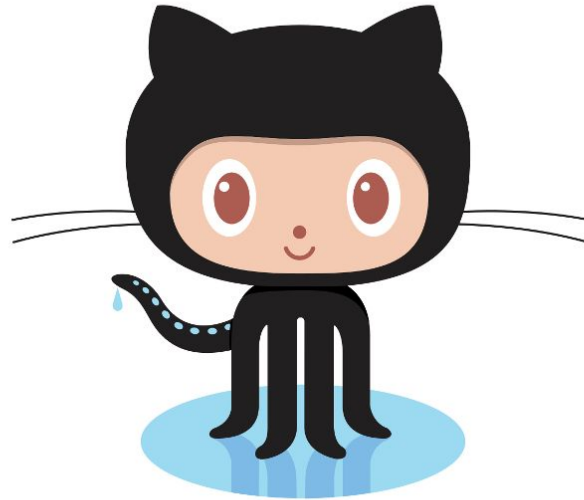


# Tooling



# GitHub

- Hosting von git repository
- Issue tracking
- Wiki
- Pull requests



# Slack

- Teamkommunikation
- Austausch von Dateien
- Austausch von Code snippets



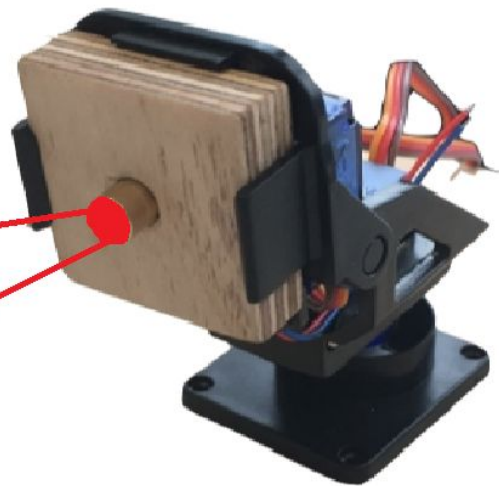
“Ein RIOT Sie zu knechten, ins Dunkle zu  
treiben und ewig zu binden”



- Martin Heusmann



**Danke**



Kontakt:

Team RIOTNerf

Slack:

RIOT\_WS16/17

[riotws1617.slack.com](https://riotws1617.slack.com)