

# Wireless Electric Lock

Hardware-nahe Systemprogrammierung

Aaron Winziers

Benedikt Lüken-Winkels

9. August 2018

## Inhaltsverzeichnis

<b>1</b>	<b>Hardware</b>	<b>2</b>
1.1	WLAN-Modul . . . . .	2
1.2	Vorteile gegenüber Arduino . . . . .	2
<b>2</b>	<b>Programmierung</b>	<b>3</b>
2.1	Messen des Klingelrhythmus . . . . .	3
2.2	1602A (HD44780) LCD - Anzeige . . . . .	3
2.2.1	Verkabelung . . . . .	3
2.2.2	LiquidCristal Library . . . . .	3

# **1 Hardware**

## **1.1 WLAN-Modul**

### **Lua Module V2 ESP8266 ESP-12E**

- 4 MB Flash-Speicher
- Programmierung in C
- NodeMCU-Board
- ESP-12E
- Programmierung per Arduino möglich
- Selbstinduziertes Schlafen und Aufwecken des Chips (Stromsparen)

## **1.2 Vorteile gegenüber Arduino**

- Höhere Taktfrequenz (80-160 Mhz)
- Vielfältige Speicherressourcen

## 2 Programmierung

### 2.1 Messen des Klingelrhythmus

#### Ideen

- Unterscheidung der verschiedenen Klingellängen durch Anpassung an den Klingler
  - Datenerhebung durch verschiedene Testpersonen, wie sich die Klingellängen bei lang und kurz unterscheiden
  - Bsp.  $Time(Kurz) = \frac{2}{3}Time(Lang)$
  - ⇒ Dafür entweder Heuristik oder einmal lang, einmal kurz zum justieren.
- Orientierung für den Klingler durch visuelle Elemente
  - Display, das die Klingellänge durch Balken anzeigt 2.2
  - LED, das ein Tempo angibt

### 2.2 1602A (HD44780) LCD - Anzeige

5 Pixel breit und 8 Pixel hoch. Zwischen den Feldern sind Lücken, die 1 Pixel groß sind.

**Idee** Darstellung der Klingellänge als horizontal aufsteigende Balken

- Probleme bei der Darstellung
  - Dynamische Aktualisierung der Pixel
  - Custom Characters (keine Standardbelegung)

#### 2.2.1 Verkabelung

Um die Kommunikation zwischen Arduino und dem LCD zu gewährleisten musste das Breadboard auch entsprechend verkabelt werden. Dazu musste ein Weiteres Power-supply angesteckt werden da der Arduino nur 3V3 Output hat und 5V gebraucht sind um das LCD anzusteuern.

Das Prozess die Einheiten zu verbinden war jedoch eine große Herausforderung, da

#### 2.2.2 LiquidCrystal Library