

Dokumentation zur Projektarbeit

Remote Tic-Tac-Toe ESP8266 über ein Web-Interface steuern



1 Inhaltsverzeichnis

1	INHALTSVERZEICHNIS.....	2
2	MOTIVATION	3
3	AUFGABENSTELLUNG	3
4	ENTWURF.....	4
4.1	USE CASE DIAGRAMM	4
4.2	SYSTEMENTWURF	5
4.3	PROGRAMM	6
4.3.1	<i>Demobetrieb.....</i>	<i>7</i>
4.3.2	<i>Webserver (Spielbetrieb)</i>	<i>8</i>
4.3.3	<i>Gewinner anzeigen.....</i>	<i>11</i>
4.3.4	<i>IP-Adresse anzeigen</i>	<i>11</i>
4.4	WEBINTERFACE	12
4.5	HARDWARE	13
4.5.1	<i>LED-Anzeige.....</i>	<i>13</i>
4.5.2	<i>Schaltplan.....</i>	<i>14</i>
5	VERWENDETE KOMPONENTEN	14
5.1	ESP 8266.....	14
5.2	RGB-LEDs.....	15
5.3	SCHALTNETZTEIL	15
5.4	USB-STEP-DOWN-CONVERTER.....	16
6	ABBILDUNGSVERZEICHNIS	17

2 Motivation

Meine Motivation für diese Projektarbeit ist das Vertiefen meiner Programmierkenntnisse in C/C++, sowie die neue Erfahrung im Erstellen des Webinterfaces mit html, css und javascript. Diese Projektarbeit hat aber auch einen Nutzen für die Fakultät, indem dieses Tic-Tac-Toe Spiel zur Demonstration bei Hochschultagen verwendet werden kann. Die LED-Anzeige wird so erstellt, dass sie auch anderweitig, z.B. zur Anzeige von Texten oder anderen Spielen verwendet werden kann.

3 Aufgabenstellung

Für diese Projektarbeit gibt es Vorgaben. Es soll ein ESP8266 verwendet werden. Das Spiel muss über ein Webinterface spielbar sein und zur Visualisierung soll zusätzlich eine etwa 60 x 60 cm große Anzeige erstellt werden. Die Anzeige soll über den ESP angesteuert werden. Für das Tic-Tac-Toe Spiel soll es verschiedene Modis geben. Zur Demonstration soll der Computer gegen sich selbst spielen, man soll aber auch selbst gegen den Computer oder eine weitere Person spielen können.

4 Entwurf

In diesem Kapitel wird der Entwurf des Tic-Tac-Toe Spiels in unterschiedlichen Detailstufen beschrieben.

4.1 Use Case Diagramm

Der Entwurf beginnt mit der Festlegung und Beschreibung der Use Cases. Abbildung 1 zeigt das Use Case Diagramm. Die Erläuterung der dieser Cases erfolgt im Anschluss.

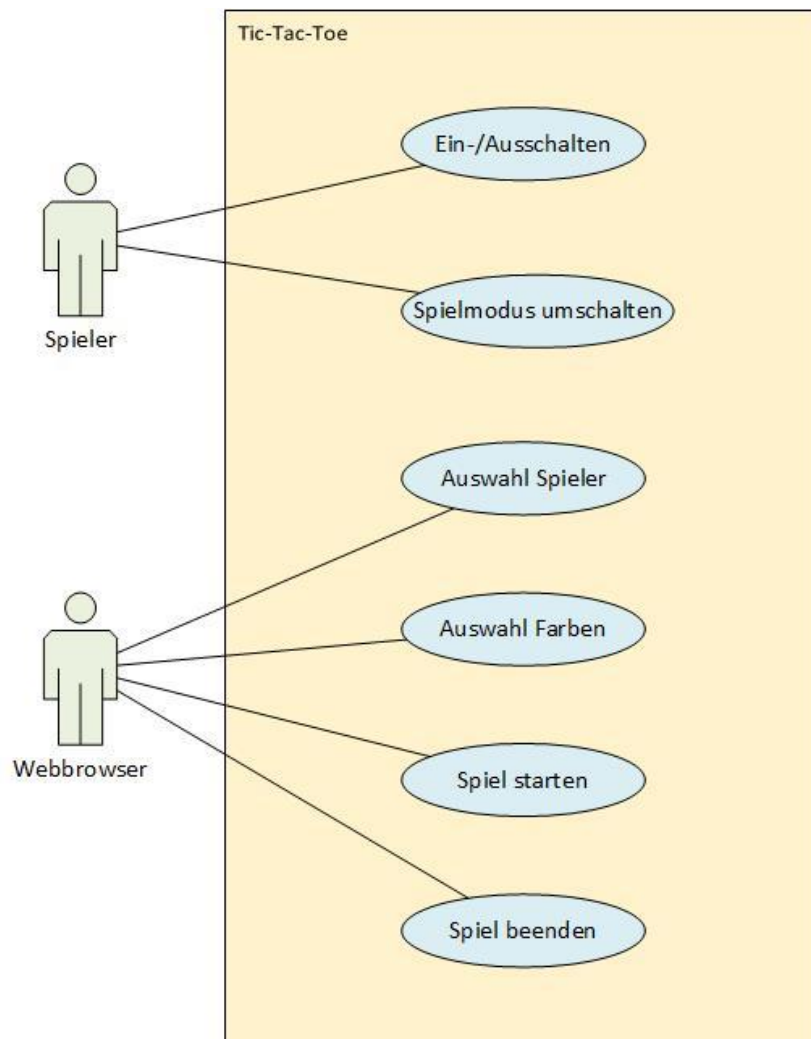


Abbildung 1: Use Case Diagramm des Tic-Tac-Toe Spiels

- **Ein-/Ausschalten**

Der Spieler soll in der Lage sein das Gerät ein- und auszuschalten.

- **Spielmodus umschalten**

Der Spieler soll die Möglichkeit haben durch einen zweiten Schalter zwischen den beiden Spielmodis „Demobetrieb“ und „Spielbetrieb“ umzuschalten.

Im Demobetrieb spielt der Computer gegen sich selbst. Aus diesem Grund eine Verbindung zum Webbrowser unnötig und damit auch das WLAN-Signal. In diesem Modus wird das WLAN ausgeschaltet.

Im Spielbetrieb betreibt der ESP einen WLAN- Access Point, mit dem sich der Spieler

direkt verbinden kann. Nachdem sicher der Spieler mit dem ESP verbunden hat, kann das Tic-Tac-Toe-Spiel über den Webbrowser gespielt werden

- **Auswahl Spieler**
Es wird eingestellt, ob der Spieler 1 der Computer ist oder ob ein Mensch spielt. Das Gleiche gilt auch für Spieler 2. Die Einstellung, dass Spieler 1 und Spieler 2 der Computer sind, wird nicht akzeptiert. Dafür gibt es den Demobetrieb.
- **Auswahl Farben**
Die LED-Anzeige wird mit RGB-LEDs realisiert, aus diesem Grund können die Farben der Spieler, also X und O, sowie die Farbe des Gitters, dass die neun Spielfelder voneinander trennt, verändert werden.
- **Spiel starten**
Durch den Start des Spiels werden die Spieleinstellung, die im Webbrowser vorgenommen worden, an den ESP gesendet und das entsprechende Spiel gestartet.
- **Spiel beenden**
Das aktuelle Spiel wird beendet. Die Spieleinstellungen werden nicht zurückgesetzt und können für das nächste Spiel übernommen werden.

4.2 Systementwurf

Mit dem Systementwurf soll die Kommunikation der einzelnen Komponenten untereinander dargestellt werden. Abbildung 2 zeigt den Systementwurf.

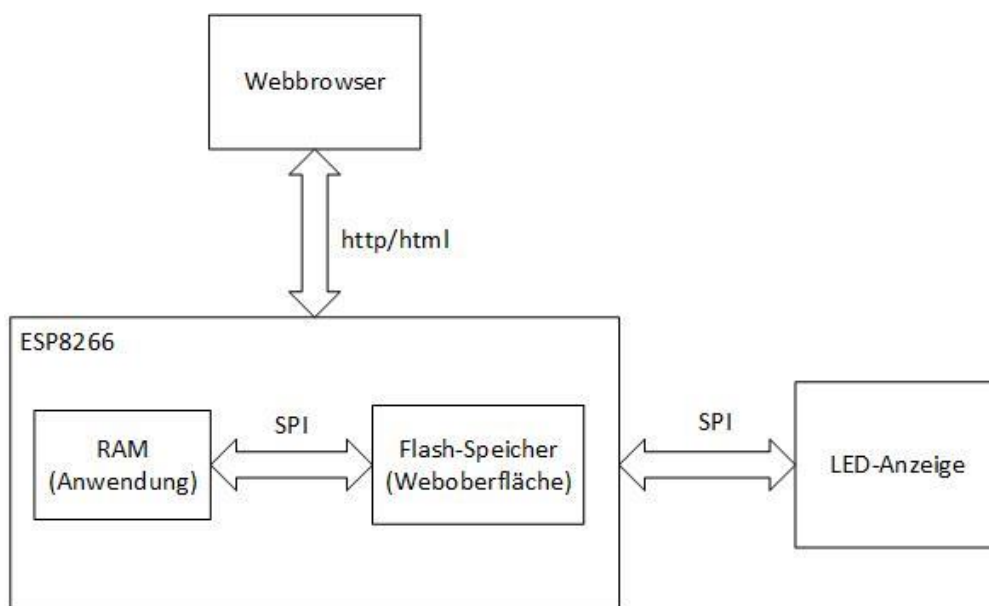


Abbildung 2: Systementwurf des Spiels

Der ESP8266 betreibt einen WLAN-Access Point, über den man sich mit einem WLAN fähigen Endgerät direkt verbinden kann. Die Kommunikation zwischen dem ESP und dem Webbrowser erfolgt über http-Requests und html. Um die Auslastung auf dem RAM des ESPs möglichst niedrig zu halten, wird die Webpage des Spiels im Flashspeicher des ESP abgelegt und bei Anforderung durch das Programm mittels SPI geladen und an den Webbrowser gesendet. Die LED-Anzeige wird mittels SPI über eine serielle Schnittstelle am ESP gesteuert.

4.3 Programm

Das gesamte Programm verteilt sich auf mehrere Tasks, die in Abbildung 3 dargestellt sind. Der Task „Demo“ beinhaltet den Demobetrieb, der ohne Weboberfläche vom Computer gespielt wird. Der Task „Webserver“ beinhaltet den Spielbetrieb mit Weboberfläche. Zwischen den beiden Tasks wird mit dem Betriebswahlschalter umgeschaltet. Dabei wird ein Interrupt aufgerufen, in dem beispielsweise der Task des Demobetriebs deaktiviert wird und der Task des Webserver gestartet wird. Beim Start des Webservertasks wird außerdem auch das WLAN aktiviert, im Demobetrieb ist dieses ausgeschaltet. Die Spielelogik wird in den Tasks „Demo“ und „Webserver“ umgesetzt. Die Tasks „Anzeige aktualisieren“, „Gewinner anzeigen“ und „IP-Adresse anzeigen“ werden aus den vorherigen beiden Tasks heraus gestartet. „Anzeige aktualisieren“ und „Gewinner anzeigen“ deaktivieren sich nach einmaligem Ausführen selbst. Der Task „IP-Adresse anzeigen“ wird aus dem „Webserver“ heraus deaktiviert.

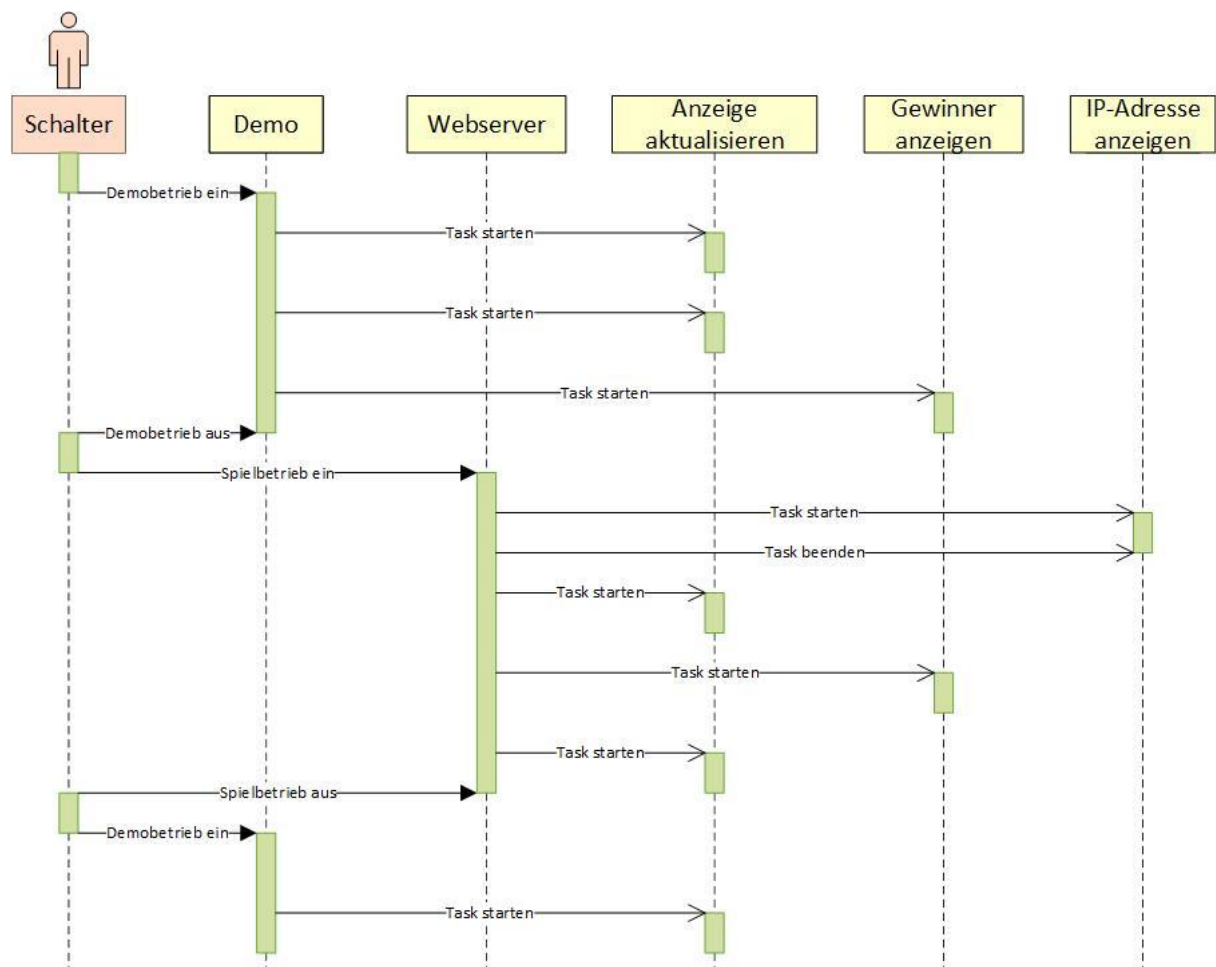


Abbildung 3: Sequenzdiagramm der einzelnen Tasks

4.3.1 Demobetrieb

Um den Ablauf des Demobetriebs zu erläutern, wird in diesem Unterkapitel in Abbildung 4 die State-Machine des Demobetriebs dargestellt.

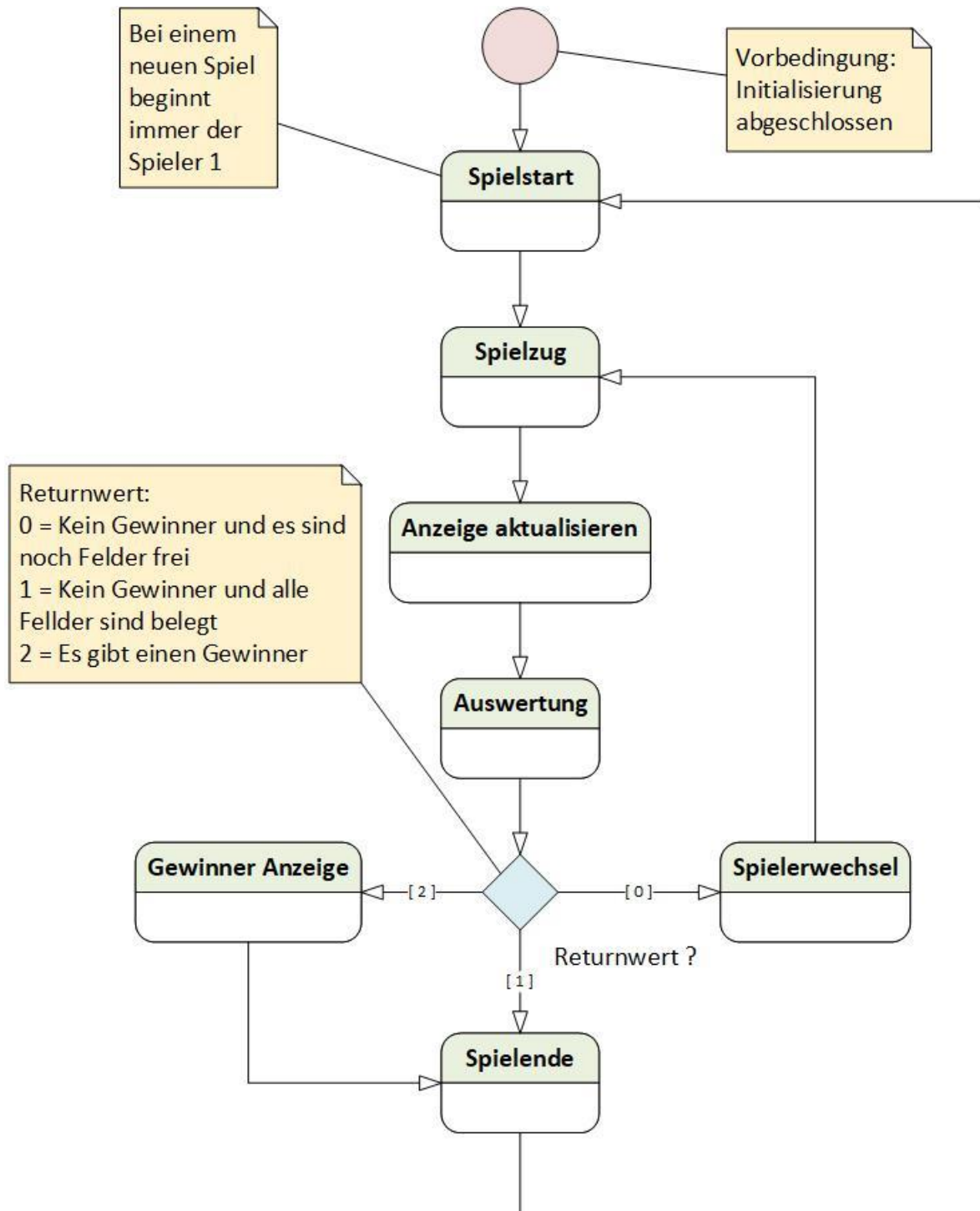


Abbildung 4: State-Machine des Demobetriebs

4.3.2 Webserver (Spielbetrieb)

Der Ablauf eines Spiels wird nicht wie im Demobetrieb durch eine State-Machine gesteuert. Im Webserver sind alle notwendigen Methoden hinterlegt, die für ein Spiel notwendig sind. Diese Methoden werden dann über die Weboberfläche aufgerufen. Die folgende Tabelle zeigt die registrierten Methoden und wie sie aufgerufen werden.

Kommando	Methode	Zeitpunkt
„/“	Schickt die Weboberfläche an den Browser.	Nach der Eingabe der IP-Adresse im Browser oder durch aktualisieren des Browserfensters
„/start“	Start ein neues Spiel mit den eingestellten Parametern.	Bei Betätigung der Starttaste.
„/s_fld“	Sendet den über die Weboberfläche gemachten Spielzug an den ESP. Diese Methode enthält auch die Spielelogik.	Durch anklicken eines noch nicht belegten Spielfelds.
„/end“	Beendet das aktuelle Spiel.	Bei Betätigung der Endetaste.

Im Folgenden werden zur Erläuterung der hinterlegten Methoden die Ablaufpläne in den Abbildungen 5, 6 und 7 dargestellt.

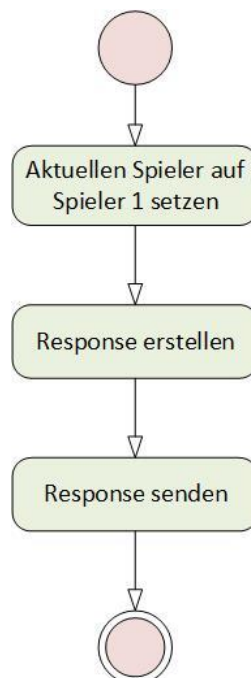


Abbildung 5: "Spiel beenden" - Methode, die durch "/end" aufgerufen wird.

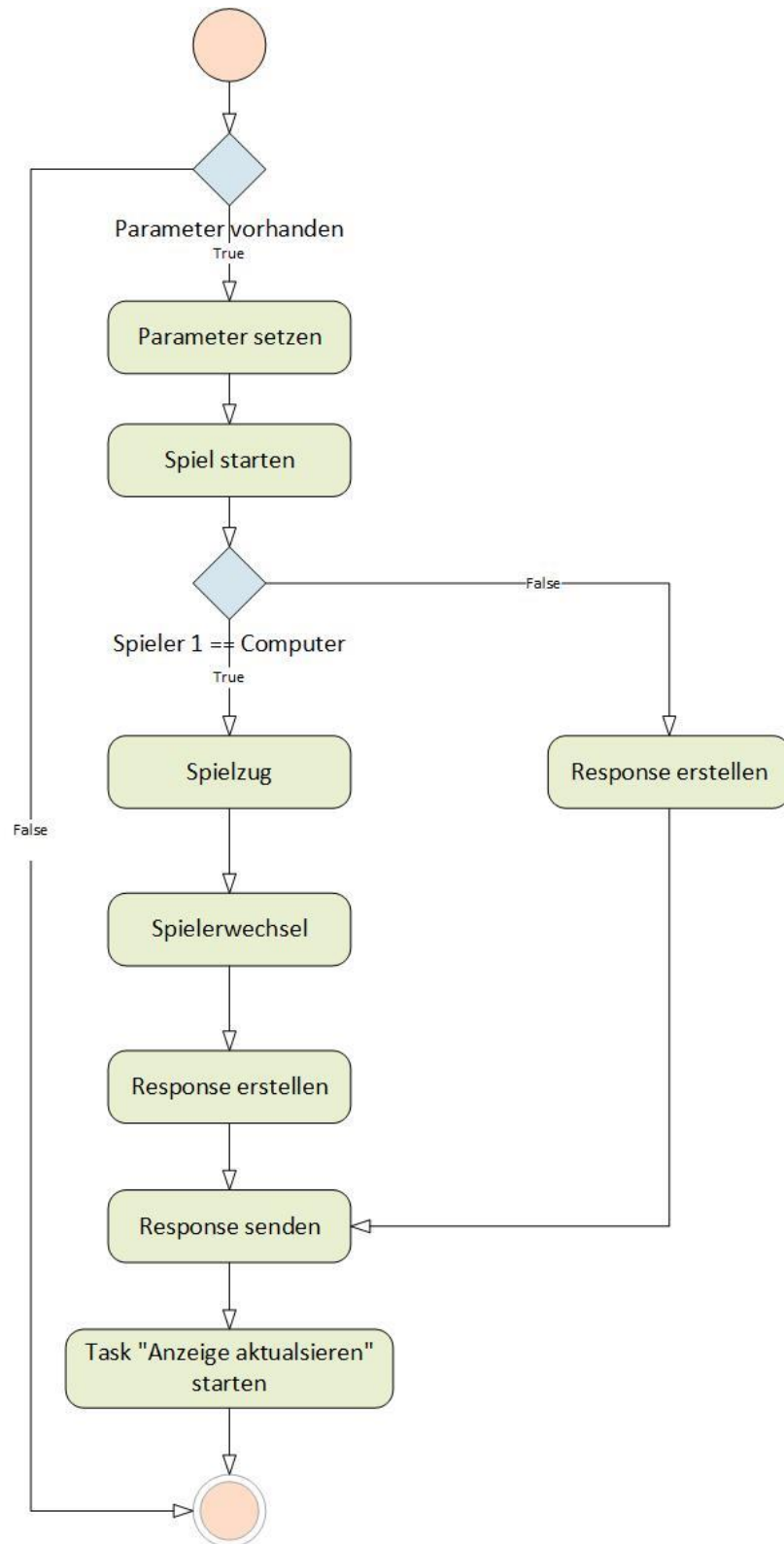


Abbildung 6: "Spiel starten" - Methode, die durch "/start" aufgerufen wird.

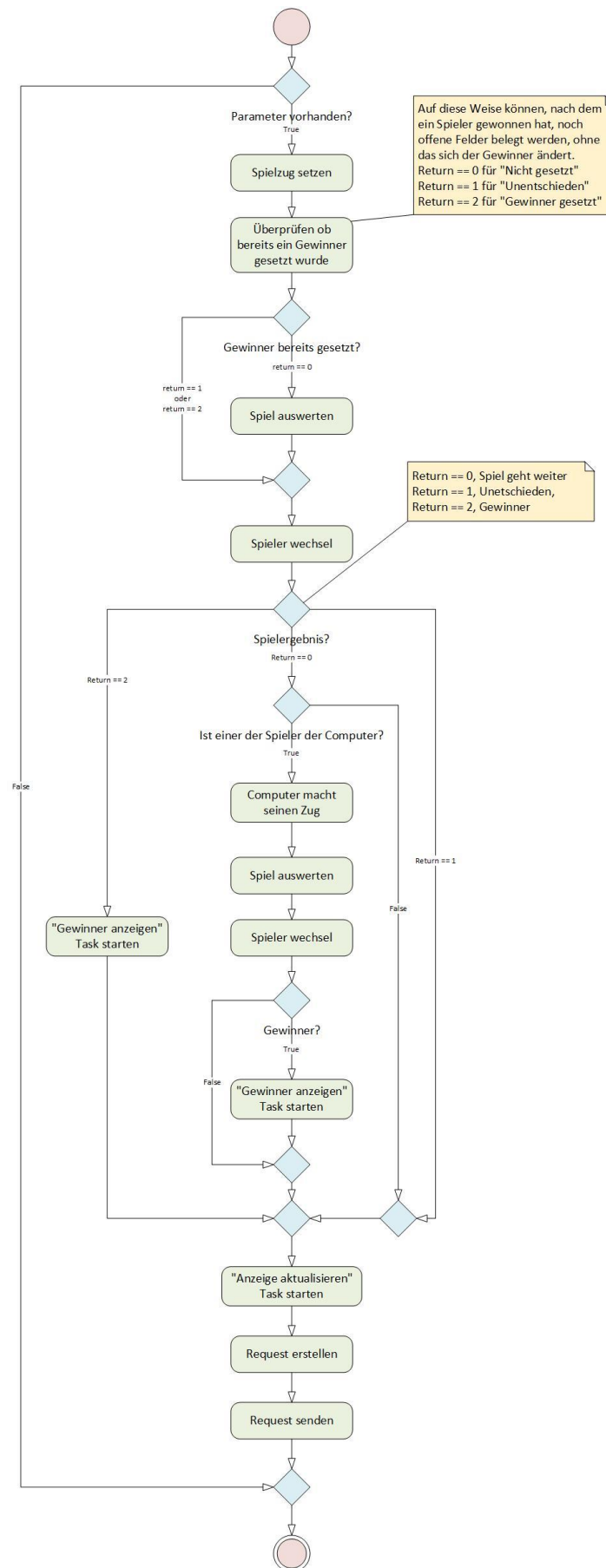


Abbildung 7: "Spielzug machen" - Methode, die durch "/s_fld" aufgerufen wird.

4.3.3 Gewinner anzeigen

Im Task „Gewinner anzeigen“ werden auf der Anzeige abwechselnd ein leeres Spielfeld und die Gewinnerfelder angezeigt.

4.3.4 IP-Adresse anzeigen

Der Task „IP-Adresse anzeigen“ wird gestartet, sobald der „Webserver“ – Task gestartet wurde und der Server sich eine IP - Adresse gegeben hat. „IP-Adresse anzeigen“ wird erst beendet, wenn sich ein Client mit dem Webserver verbunden hat und die Weboberfläche des Spiels aufgerufen hat. Abbildung 8 zeigt den Ablauf dieses Tasks.

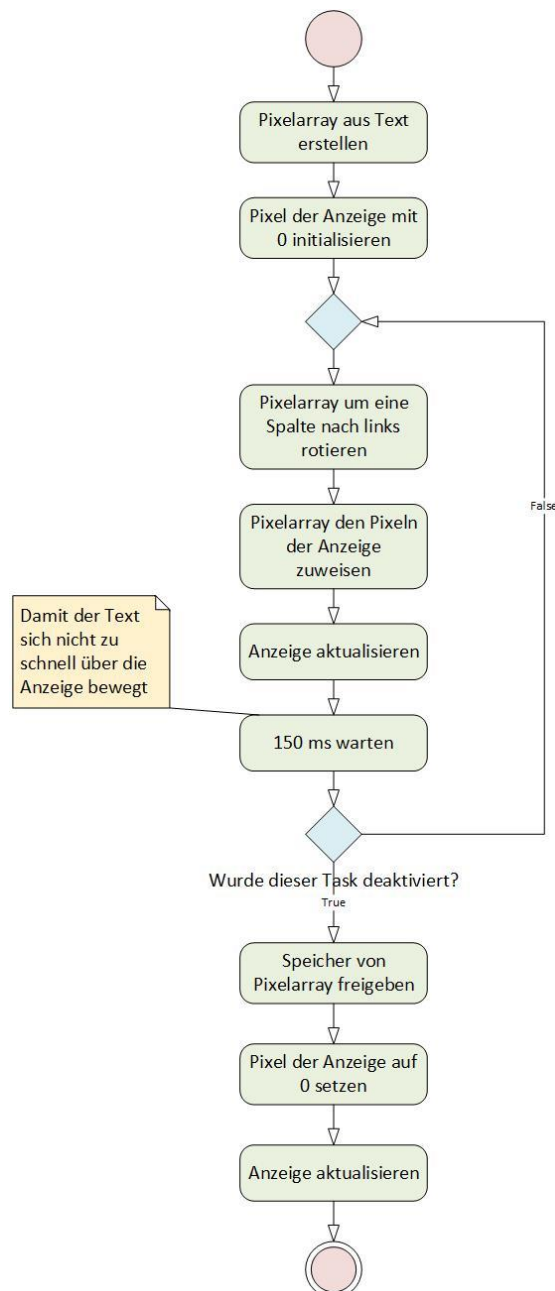


Abbildung 8: Ablaufplan des "IP-Adresse anzeigen" Tasks.

4.4 Webinterface

Tic Tac Toe

X Spieler 1 O Spieler 2

Spieler: Spieler Spieler: Spieler

Farbwahl: Farbwahl:

Farbwahl Spielfeld:

Spiel starten

Tic Tac Toe

X	O	

Spieler 1 ist am Zug

X Spieler 1 O Spieler 2

Spieler: Spieler Spieler: Spieler

Farbwahl: Farbwahl:

Farbwahl Spielfeld:

Spiel beenden

Projektarbeit Daniel Wunsch Matr.-Nr. 59527

Projektarbeit Daniel Wunsch Matr.-Nr. 59527

Abbildung 9: Startseite der Weboberfläche (links) und Oberfläche während eines Spiels (rechts)

Abbildung 9 links zeigt die Startseite. Solange kein Spiel gestartet wurde, sind die einzelnen Felder des Spielfelds deaktiviert und können nicht angeklickt werden. Auf der Startseite können Einstellungen für den Spieler vorgenommen werden. Es kann eingestellt werden, ob zwei Menschen gegeneinander oder ein Mensch gegen einen Computer spielt. Zwei Computer können nicht gegeneinander spielen. Ein Spiel wird immer von Spieler 1 begonnen. Für jeden Spieler kann die Farbe verändert werden. Diese Farbeinstellung wird später für die LED-Anzeige übernommen. Durch Betätigen der Spielstarttaste werden die Spieleinstellungen an den ESP gesendet und ein Spiel gestartet. Die Oberfläche passt sich nun an und sieht jetzt aus wie in Abbildung 9 rechts. Es können nun keine Einstellungen mehr vorgenommen werden. Die einzelnen Felder sind jetzt als Schaltflächen aktiviert. Belegte Felder können nicht noch einmal angeklickt werden. Die „Spiel starten“ Schaltfläche wechselt ihre Beschriftung zu „Spiel beenden“. In der Box unterhalb des Spielfelds wird angezeigt welcher Spieler am Zug ist und ob ein Spieler gewonnen hat. Durch Betätigen der „Spiel beenden“ Schaltfläche wechselt die Oberfläche wieder zurück zur Startseite Abbildung 9 links, nur die zuvor gemachten Einstellungen bleiben erhalten, um nicht wieder alles neu einstellen zu müssen, wenn man ein neues Spiel spielen möchte. Alle Veränderungen an der Weboberfläche werden im Hintergrund geladen.

4.5 Hardware

4.5.1 LED-Anzeige

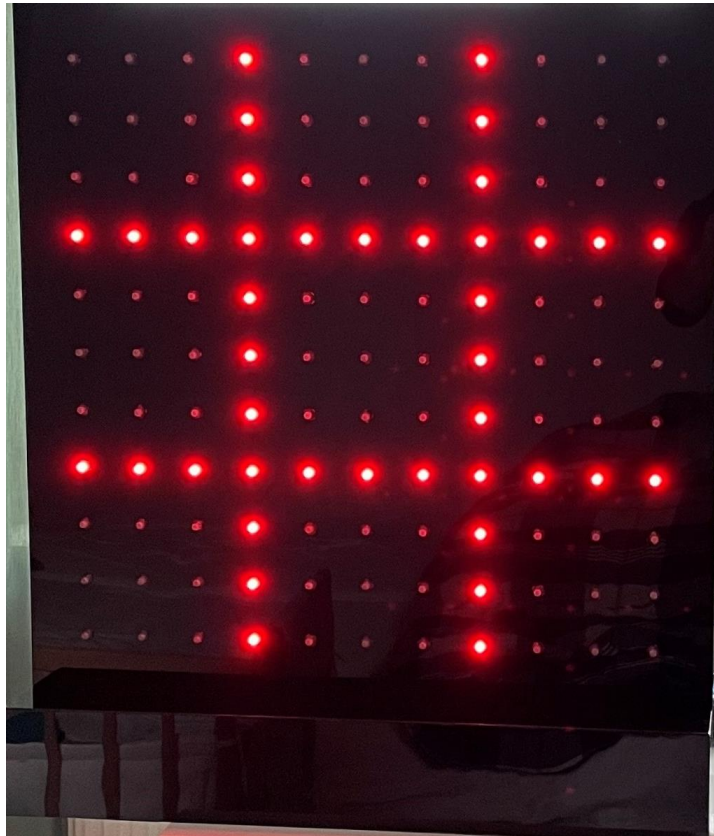


Abbildung 10: LED-Anzeige

Bei der LED-Anzeige habe ich mich für eine 11x11 Matrix entschieden. Durch den Einsatz von RGB LEDs kann die Übersichtlichkeit, die sich durch den großen Pixelabstand (6 cm) verschlechtert, mit unterschiedlichen Farben kompensiert werden. Die 11x11 Matrix bietet die beste Auslastung, da alle Pixel verwendet werden.

Aus dieser LED-Matrix ergeben sich 121 LEDs. Der Gesamtverbrauch für diese Anzeige liegt bei 36 W, da ein Pixel 0,3 W benötigt. Zum Flashen der LEDs sind pro Pixel 24 Bit notwendig, insgesamt 2904 Bit, wenn man die Start- und Endsequenz ein Flashvorgangs nicht mitzählt. Bei einer Frequenz von 80 MHz des ESP dauert das Übertragen der Daten auf die Anzeige etwa 36,3 μ s.

4.5.2 Schaltplan

Abbildung 11 zeigt den Schaltplan. Für die LED-Anzeige werden immer elf LEDs in Reihe geschaltet, um den Stromverbrauch zu reduzieren und damit auch die Erwärmung der Leitungen und des Schaltnetzteils. Die LEDs werden über die Datenleitung seriell angesteuert.

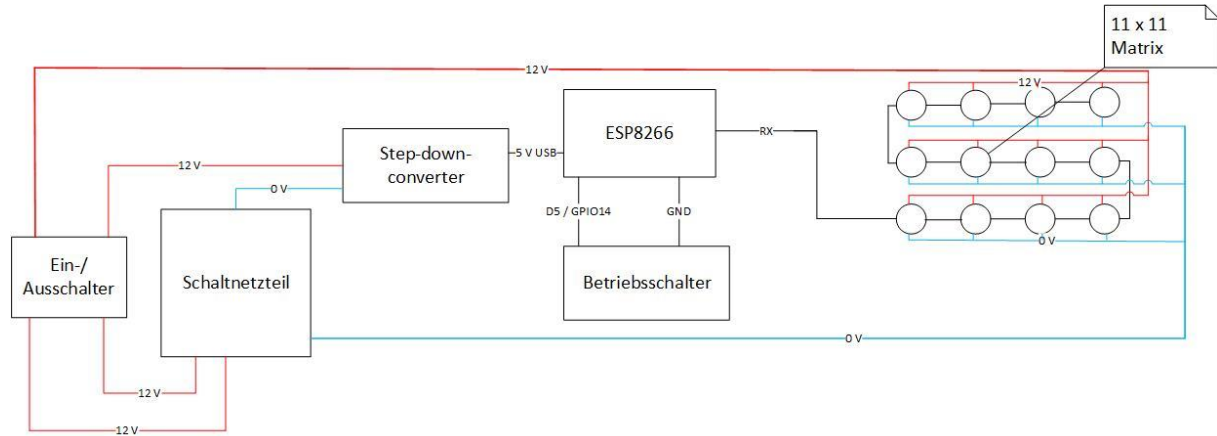


Abbildung 11: Schaltplan des Tic-Tac-Toe Spiels

5 Verwendete Komponenten

5.1 ESP 8266

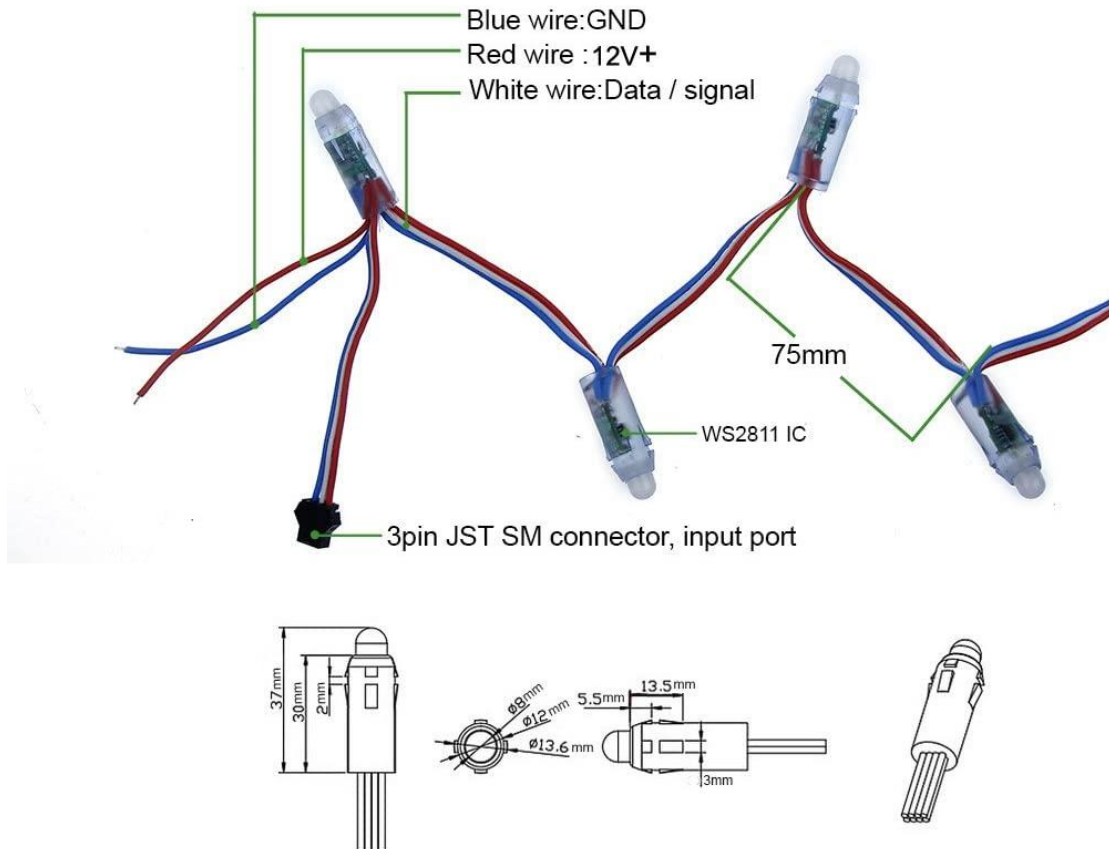
Der verwendete Mikrocontroller ist der ESP8266 von AZ-Delivery. Der MCU hat eine Taktfrequenz von 80 MHz und verfügt über einen 4 MB Flashspeicher. Die GPIOs des ESP nutzen eine 3,3 V Logik. Die Spannungsversorgung erfolgt über den 5 V USB-Anschluss. Der MCU kann unter anderem über die Arduino IDE oder Visual Studio Code programmiert werden. Abbildung 12 zeigt den ESP8266 von AZ-Delivery.



Abbildung 12: ESP8266 von AZ-Delivery

5.2 RGB-LEDs

Die RGB-LEDs kommen von der Firma Altivo. Jeder Pixel dieser LED-Kette wird über den WS2811 IC gesteuert. Geflasht werden die LEDs durch SPI. Sie benötigen eine 12 V Spannungsversorgung und haben eine Leistungsaufnahme von 0,3 W pro Pixel. Abbildung 13 zeigt die RGB-LEDs der Firma Altivo.



- Waterproof IP68. It can be used outdoor or even underwater

Abbildung 13: RGB-LEDs der Firma Altivo

5.3 Schaltnetzteil

Das Schaltnetzteil kommt von der Firma Mean Well. Es besitzt eine Ausgangsleistung von 102 W und verfügt über zwei 12 V Anschlüsse, die zusammen bis 8,5 A Strom abgeben können. Die Ausgangsleistung des Schaltnetzteil ist bewusst deutlich größer gewählt als für den ESP und die LED-Anzeige notwendig wären, um die Auslastung und die Erwärmung im Einbauraum des Spiels niedrig zu halten. Abbildung 14 zeigt das verwendete Mean Well RS-100-12 Netzteil.



Abbildung 14: Mean Well RS-100-12 Netzteil

5.4 USB-Step-down-converter

Für die Spannungsversorgung des ESP wird ein Step down converter verwendet, der eine Eingangsspannung von 6 V – 40 V benötigt. Der Konverter ist auf der einen Seite mit dem 12 V Netzteil verbunden und auf der anderen Seite über ein USB-Kabel mit dem ESP. Abbildung 15 zeigt den verwendeten Konverter.

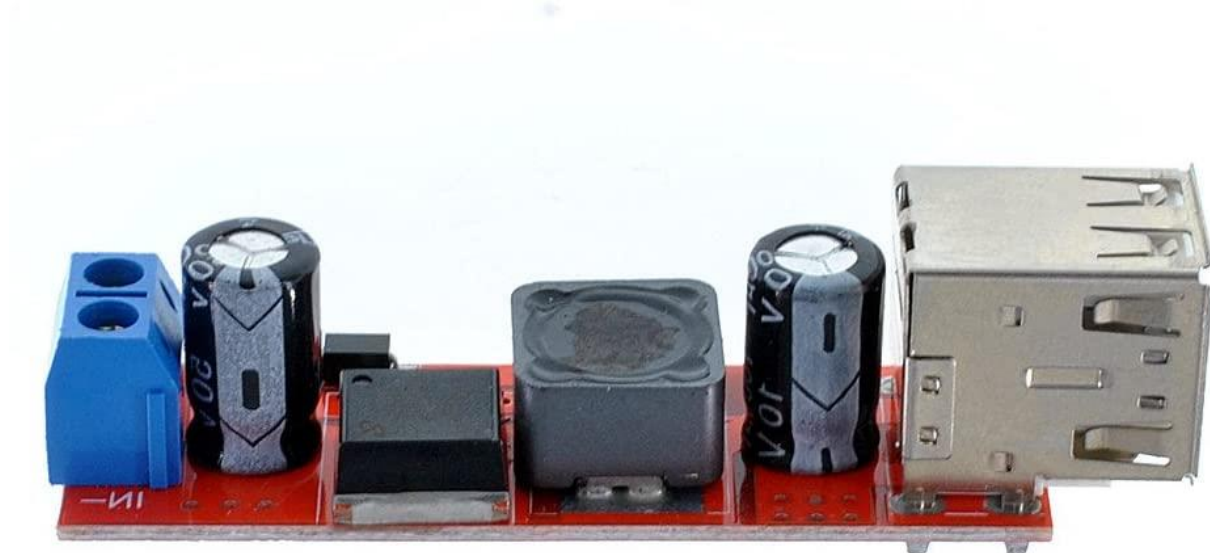


Abbildung 15: USB Step down converter

6 Abbildungsverzeichnis

Abbildung 1: Use Case Diagramm des Tic-Tac-Toe Spiels.....	4
Abbildung 2: Systementwurf des Spiels.....	5
Abbildung 3: Sequenzdiagramm der einzelnen Tasks.....	6
Abbildung 4: State-Machine des Demobetriebs	7
Abbildung 5: "Spiel beenden" - Methode, die durch "/end" aufgerufen wird.	8
Abbildung 6: "Spiel starten" - Methode, die durch "/start" aufgerufen wird.	9
Abbildung 7: "Spielzug machen" - Methode, die durch "/s_fld" aufgerufen wird.	10
Abbildung 8: Ablaufplan des "IP-Adresse anzeigen" Tasks.....	11
Abbildung 9: Startseite der Weboberfläche (links) und Oberfläche während eines Spiels (rechts)	12
Abbildung 10: LED-Anzeige	13
Abbildung 11: Schaltplan des Tic-Tac-Toe Spiels	14
Abbildung 12: ESP8266 von AZ-Delivery	14
Abbildung 13: RGB-LEDs der Firma Altivo	15
Abbildung 14: Mean Well RS-100-12 Netzteil.....	16
Abbildung 15: USB Step down converter	16