Praktikum Digitaltechnik

IR - Remote

Benedikt Schnörr, Sebastian Pasinski

# Gliederung

[**Aufgabenstellung**](#_o416kcmjjpvl) **3**

[**Systemarchitektur**](#_841qwzd4aejy) **3**

[Schaltplanskizze](#_ft8f5nvmgoa1) 3

[Aktivitätsdiagramm](#_33u8boua5oge) 4

[**Schaltplan**](#_7m2oro76sre9) **5**

[Mikrocontroller](#_283vly4wbg3p) 6

[Taktgebung](#_empozeo1zpah) 6

[Programmier-/Debugging-Schnittstelle](#_pmyhfgs8gakh) 6

[Spannungsregulierung](#_defazytsrpdu) 7

[Schalter](#_1d90gz86k1p9) 7

[IR-LED](#_3d9j6dopixnc) 8

[**Platinendesign**](#_96yi9dicc93z) **8**

[**Quellen**](#_bytjappq86zf) **10**

# 

# Aufgabenstellung

Die Aufgabenstellung für das Digitaltechnik-Praktikum ist es, eine Infrarot-Fernbedienung zu bauen. Diese soll einen Receiver ansteuern, welcher das RC5-Protokoll unterstützt. Es sollen die wichtigsten Funktionen eines CD-Players mit der Fernbedienung angesprochen werden können. Dazu gehören:

* Play
* Stop
* Pause
* Previous, vorheriger Titel
* Next, nächster Titel

Die Anforderungen für dieses System sind, dass es auf einer eigenen Platine aufgebaut und batteriebetrieben arbeiten soll.

Bei unserem Vorgehen werden wir zunächst einen Schaltplan entwerfen, der die elektrischen Bauteile und den Mikrocontroller enthält, sodass daraufhin eine Platine designt und die Bauteile bestellt werden können. Im zweiten Schritt werden wir dann den Code entwickeln, mit dem die gedrückte Taste auf der Fernbedienung ausgelesen wird und das passende Signal als RC5-Code gesendet wird.

# Systemarchitektur

## Schaltplanskizze

Wir haben zunächst ein Blockschaltbild erstellt, das die wichtigsten Komponenten unserer Schaltung darstellt. Dazu gehören die Schalter mit ihren einzelnen Funktionen, der Mikrocontroller, der die IR-LED ansteuert, und der Receiver, der die gesendeten Befehle empfängt.



Abb. 1: Blockschaltbild der Fernbedienung

## Aktivitätsdiagramm

Der Ablauf der Benutzung der fertigen Fernbedienung wird in einer Tabelle dargestellt.

Tabelle 1: Ablauf bei Interaktion des Benutzers

| **Aktoren** | **Aktivität** |
| --- | --- |
| Benutzer | drückt einen Schalter in der Schaltung |
| Schalter [0-4] | sendet beim Drücken elektrische Signale an einen Mikrocontroller-Eingang |
| Microcontroller | empfängt Spannung am Eingang und sendet elektrische Signale an den Ausgang mit IR-LED |
| IR Led | Wandelt elektrische Spannung in Lichtsignale um |
| Receiver | empfängt die Lichtsignale und führt z.B “Play”-Befehl aus |
| Benutzer | Kann entsprechenden Befehl am Receiver erkennen |

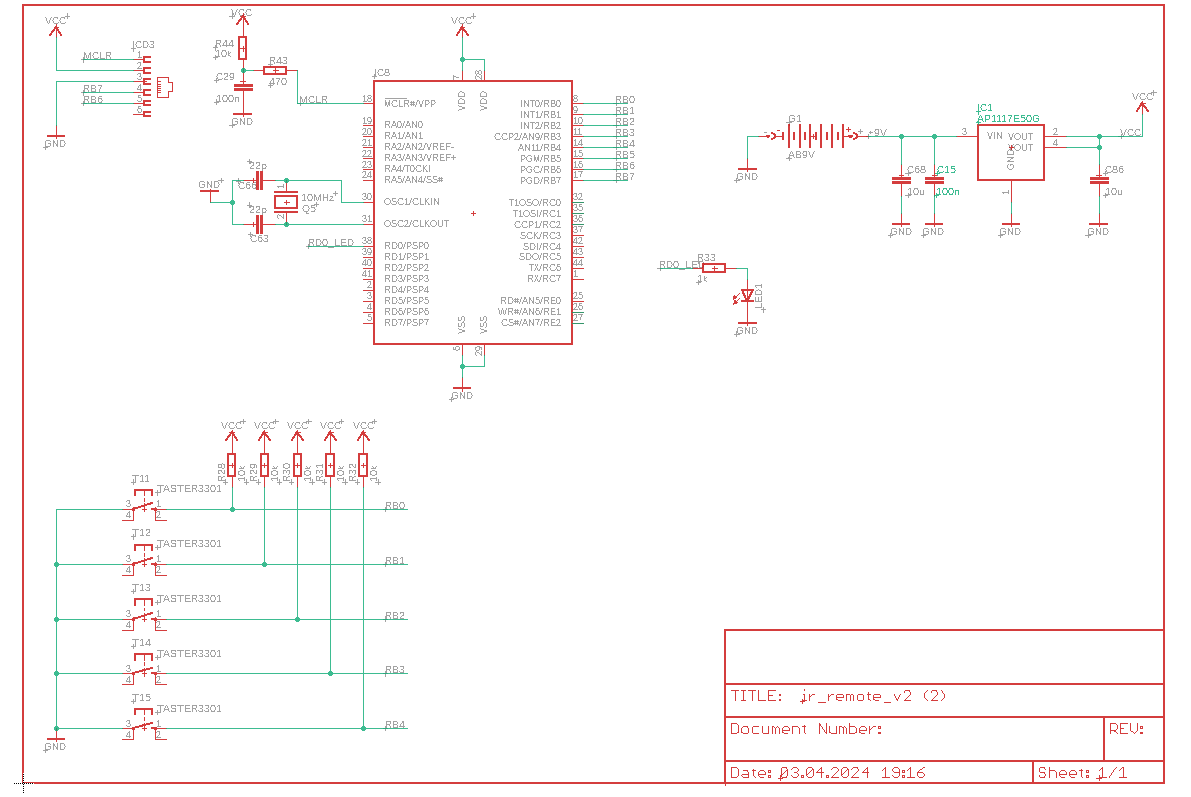
Das Aktivitätsdiagramm stellt den Ablauf noch einmal bildlich dar. Interessant dabei ist, dass sich der Mikrocontroller immer im Sleep-Modus befindet und von einem Schalter aufgeweckt wird, um Energie zu sparen, wenn keine Taste gedrückt wird.



Abb. 2: Aktivitätsdiagramm für den Ablauf der Benutzereingabe

# Schaltplan

Um die Funktionsweise des Systems auch in Hardware umzusetzen, wird eine Schaltung entwickelt und daraus eine Platine gestaltet.

Abb. 3: Schaltplan in Eagle

## Mikrocontroller

Der in der Schaltung verwendete Mikrocontroller ist ein PIC18F4525, an ihn sind neben Schaltern und Infrarot-LED noch weitere Schaltkreise angeschlossen. In Eagle wird zur Darstellung ein PIC18F4520 verwendet, der aber die gleiche Pin-Belegung und die gleichen Abmessungen wie der PIC18F4525 besitzt und daher als Alternative im Schaltplan benutzt werden kann.

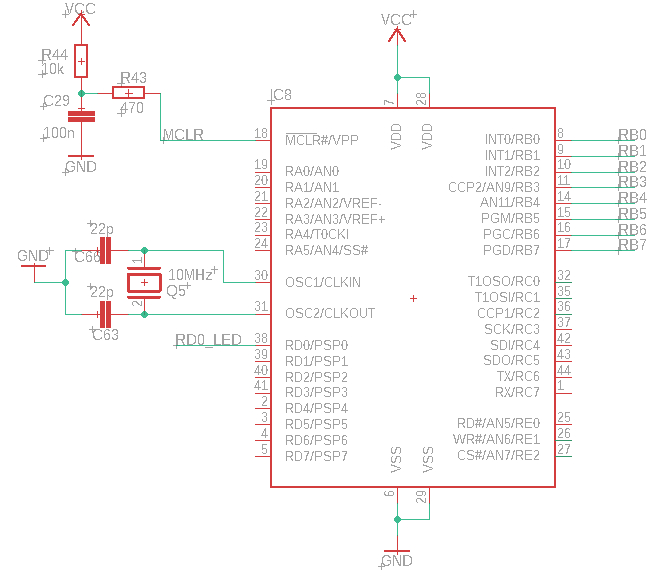


Abb. 4: Beschalteter PIC18F2545

### Taktgebung

Der Quarz dient dazu, dem Mikrocontroller eine Taktfrequenz bereitzustellen, nach der sich dieser richten kann. Das Datenblatt des PIC18F4525 empfiehlt für die Oszillator-Konfiguration “High Speed Crystal/Resonators” mit einer Frequenz von 10 MHz Kondensatoren eine Kapazität von 15 pF. In der vorgegebenen Schaltung wurde eine höhere Kapazität verwendet, dies verbessert die Stabilität des Schwingkreises, kann aber zur Vergrößerung der Startzeit führen [1].

### Programmier-/Debugging-Schnittstelle

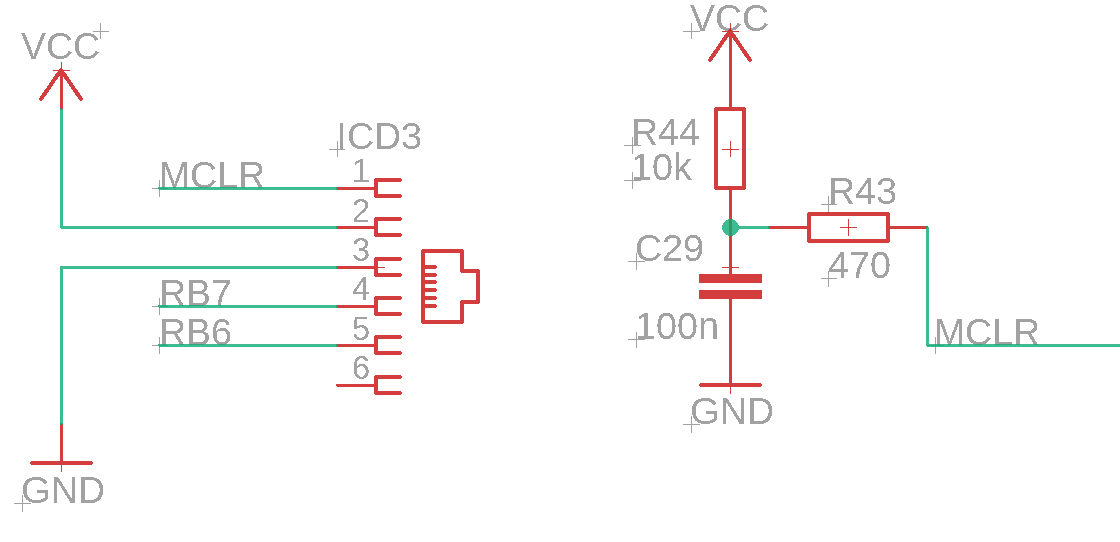


Abb. 6: Schnittstellen des Mikrocontrollers zum Debuggen/Programmieren

Mit diesem Schaltungsblock wird eine Schnittstelle zwischen der Software zum Programmieren und den Pins RB6 und RB7 des Mikrocontrollers hergestellt. Das Debugging des PIC kann dadurch gesteuert werden, während dieser in der Schaltung eingebaut ist [1].

### Spannungsregulierung

Der AP1117E50G wird eingesetzt, um die Spannung von einer 9 V-Versorgung auf 5 V als Betriebsspannung des PIC zu senken. Die 10 μF Kondensatoren gleichen dabei mögliche Schwingungen in der erzeugten 5 V-Spannung aus. Der 100 nF Kondensator soll mit seiner größeren Kapazität die Eingangsspannung halten, wenn die Versorgungsspannung zu stark abfällt. Der Spannungsregulierer kann nämlich nur bei einer Eingangsspannung von 1,3V über der Ausgangsspannung korrekt arbeiten [2].

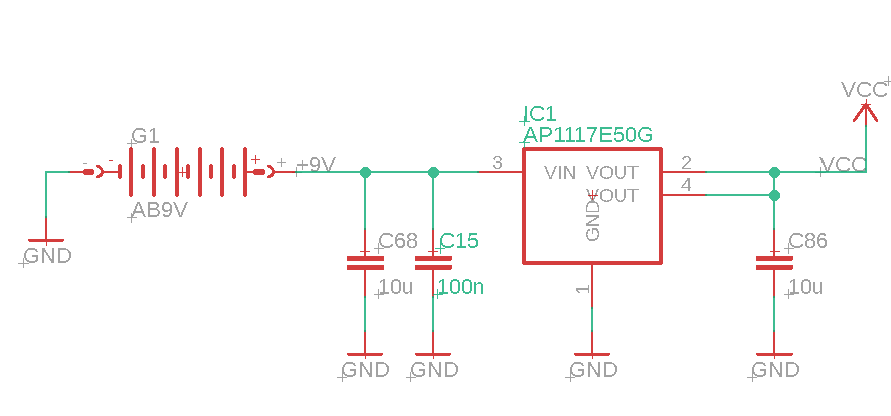


Abb. 7: Spannungsversorgung

## Schalter

Die Schalter, welche an verschiedenen Pins an Port B des Mikrocontrollers angeschlossen sind, sorgen dafür, dass im ausgeschalteten Zustand die Versorgungsspannung von 5 V am jeweiligen Pin anliegt. Im angeschalteten Zustand werden die Pins auf Masse gezogen.

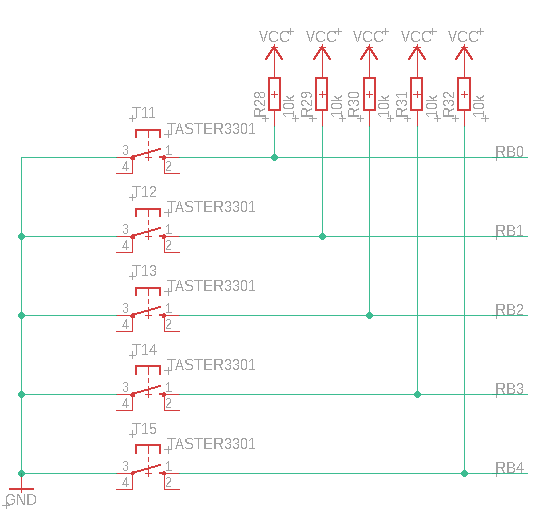


Abb. 8: Verschaltung der Taster

## IR-LED

Für das Aussenden der Lichtsignale wird eine 5mm Vischay IR-LED verwendet. Der Vorwiderstand berechnet sich mit dem Spannungsabfall von 1 V an der LED und dem Betriebsstrom von 50 mA zu 80 Ω. Als erhältlicher Widerstand können dann z.B. 82 Ω oder 100 Ω gewählt werden [3].

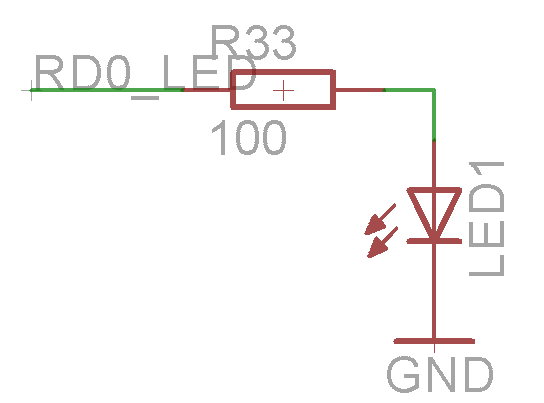


Abb. 9: Verschaltung der IR-LED

# Platinendesign

Das Platinen-Design wurde genau wie die Erstellung des Schaltplans in Eagle gemacht. Die Vorgaben des Labors bezüglich der Designregeln waren an die Standardeinstellungen des Editors zu befolgen.

Anforderungen an den Platinenaufbau sollten allerdings sein, dass die Schalter möglichst aufgereiht beieinander sein sollten und der Quarz-Schwingkreis möglichst nah am Mikrocontroller sein sollte. Zudem sollte die IR-Led möglichst weit am Rande einer Plattenkante befinden.

Die erste Platzierung der Bauelemente auf einer Leiterplatte ist in Abb. 10 zu sehen:

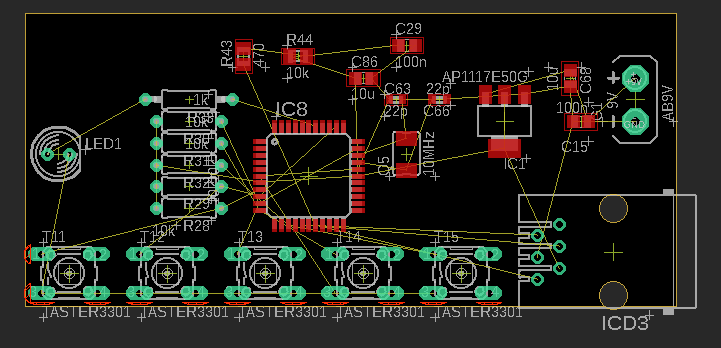


Abb. 10: Erste Platzierung der Bauteile in Eagle

Ein endgültiges Layout sollte allerdings erst nach Rücksprache mit dem Labor und dem Dozenten angefertigt werden.

# Quellen

[1] Datenblatt PIC: <https://ww1.microchip.com/downloads/en/devicedoc/39626e.pdf>

[2] Datenblatt AP1117: <https://www.diodes.com/assets/Datasheets/AP1117.pdf>

[3] Datenblatt LED: <https://docs.rs-online.com/175a/0900766b80de9fc1.pdf>