

tipps_fuer_schaltungserstellung

Student Group

First Name	Surname	Matrikel Nr.



Table of Contents

- Generelles 2
- 101 - Schaltplan zeichnen 2
 - Verbindungen und Potentiale* 2
 - passive Elemente* 3
 - Mikrocontroller* 3
 - Analoge Eingangswerte (ADC)* 4
 - Digitale Schnittstellen* 4
- Zum Abschluss 5

Generelles

- **Alle ICs mit Kondensatoren an den Stromversorgungspins versehen!!!** Diese müssen nahe am IC liegen. Die Kondensatoren kompensieren Störungen auf der Versorgungsleitung. Sinnvoll sind hier Keramikkondensatoren mit 100 nF, da diese einen geringen **äquivalenten Serienwiderstand** haben. Auch bei Operationsverstärker, ADC-ICs, ... sind diese vorzusehen.
- Theoretisch können Sie alle Ports zu Ein/Ausgänge umkonfigurieren. In der Praxis sind die verwendeten Ports **weise** zu wählen:
 - Es ist umständlich, aus einem mit JTAG belegten Pin einen Ausgabe-Pin zu machen.
 - Es ist manchmal besser einen Analoginput zur Eingabe von mehreren Binärwerten zu nutzen (z.B. über **R-2R-Netzwerk**, **Falstad**).
- Berücksichtigen Sie bereits bei Der Schaltungserstellung das Layout. Es ist einfacher von einer vorhandenen Schematic **und Board** auszugehen und dort die nicht benötigten Komponenten zu löschen.
- Falls Sie Eagle per Tastatur bedienen wollen, können Sie auf den Seiten des **MIT** die Editor-Commands finden

101 - Schaltplan zeichnen

- Füllen Sie den Kommentarbereich in der Zeichnung aus und halten Sie ihn aktuell (Autor, Schaltungszeichnung, Version, Datum). schlechte Beschriftung: übereinander und über Komponente 
- Legen Sie zunächst das grobe Layout auf Basis der Funktion fest. Eingänge sollten links, Ausgänge rechts zu positioniert sein. Bei kleineren Boards bietet sich die Struktur an, wie sie zukünftig auf der Platine wiederzufinden sein wird. 
- Der Name (unter Eigenschaften) von Bauteilen sollte möglichst kurz sein. Die Bezeichnung von Anschlüsse (z.B. Motor, RXD, ADC) und Bauteile (z.B. ATmega88) sollte sinnvoll und aussagekräftig sein. schlechte Beschriftung: doppelte Beschriftung
- Name und Bezeichnung sollten möglichst nahe am Bauteil positioniert sein. Vermeiden Sie Überlagerungen von Text bzw von Text und Komponenten. Falls Sie eine doppelte Beschriftung haben, entfernen Sie eine davon.
- Falls Sie bei den Bezeichnern eine negierte Benennung benötigen (Z.B. Out Nicht Reset), so erhalten Sie einen Überstrich mittels Ausrufezeichen: $OUT_!\text{Reset} = OUT_!\overline{\text{Reset}}$
- Nutzen Sie aussagekräftige Darstellungen für Bauteile, also z.B. Transistoren statt Blackboxen.
- Falls Sie neue Bibliotheken verwenden: Exportieren Sie neue Libraries zunächst über Datei > Exportieren > Libraries > In eine Bibliothek integrieren. Der Ursprung ("Origin") soll mittig positioniert werden; die neue Komponente sollte möglichst kompakt gestaltet sein.
- Zeichnen Sie nur Bauteile ein, welche auch auf der Platine zu finden sein werden. Also keine externen Sensoren oder Widerstände.

Verbindungen und Potentiale

- Geben Sie den Verbindungen kurze und aussagekräftige Namen, bevorzugt eine Funktionsbezeichnung ("MISO"). Alternativ kann auch der Pinnamen vom Mikrocontroller ("PB4") oder eine Kombination ("PB4_MISO") verwendet werden.

- Nicht benötigte Verbindungen sollten entfernt werden.
- Versuchen Sie Verbindungen möglichst kurz und kreuzungsfrei zu zeichnen.
- Gegen zuviel Spaghetti auf der Schaltung, hilft
 - die "Name"-Funktion und Linien-Stummel,
 - die Verwendung von vorgegebenen Potentialbauteilen wie GND, -5V oder +5V (weitere in den libs supply1 und supply2)
- Als Bezeichnungen für die verschiedenen Versorgungsspannungen von IC werden in der Industrie i.d.R. entweder VDD5, VDD33, VDD18 oder VDD5V, VDD3V3, VDD1V8 geschrieben. Bei älteren Schaltungen wurde VCC (Common Collector Voltage) statt VDD verwendet.

Symbol	Anwendung	Empfehlung
✘	<p>AGND: Analog Ground: ungestörte Masse für Analogsignale</p> <p>GND: Ground: Masse für digitale und schaltende ICs</p> <p>GNDINT: Masse für Bauteil-interne Logiken</p> <p>GNDIO: Masse für (hochspannungs)abgeschirmte Datenübertragung, Bezug für IO-Pins</p> <p>GNDI eingangsseitige Masse (Alternative für AGND falls belegt)</p> <p>GND A eingangsseitige Masse (Alternative für AGND falls belegt)</p>	Verwendung von GND und AGND
✘	<p>+5V: positive Spannungsversorgung (auch als 3,3V o.ä.)</p> <p>-5V: negative Spannungsversorgung</p> <p>VCC: Common Collector Voltage, auch als positive Versorgungsspannung nutzbar</p> <p>VCC/2 Mittenspannung bei unipolar versorgten Schaltungen</p> <p>VCCINT: Versorgung für Bauteil-interne Logiken</p> <p>VCCIO: Versorgung für IO-Pins</p> <p>-5V: sollte als Pfeilsymbol wg. Verwechslungsgefahr nicht verwendet werden</p>	Verwendung von VCC-Symbol mit Spannungsbeschriftung bei unipolar gespeisten Schaltungen. Verwendung von positiver und negativer Spannungsversorgung bei bipolar gespeisten Schaltungen

passive Elemente

- Verwenden Sie als Wert für Widerstände, Kondensatoren und Induktivitäten für die Größenbeschreibung folgende Symbole: f, p, n, u, m, k, M, G, T (kein μ , MEGA, o.ä.). Die physikalische Einheit wie Ω , F , H ist wegzulassen (im Gegensatz zu Rechnungen).

Mikrocontroller



Die folgenden Tipps beziehen sich allgemein auf Schaltungen mit Mikrocontroller.

- Resetleitung

- Der Resetpin sollte über einen Pull-up/-down Widerstand auf der nicht resetierenden Spannung gehalten werden (bei ATmega32U4 per Pull-up auf High). Weiterhin ist ein Mikrotaster von Vorteil, über welchen der Pin auf die resetierende Spannung gezogen werden kann (=“Reset-Knopf”, bei ATmega32U4 per Pulldown auf Low).
- Taktgeber
 - Überlegen Sie sich, ob Ihre Anwendung eine genaue Taktfrequenz oder kompakten/kostengünstigen Aufbau benötigt

Eigenschaften	interner RC-Oszillator	Quarz	(Quarz-)Oszillator
Genauigkeit	$\pm 1\%$ ¹⁾	$<0,01\%$ ²⁾	$<0,01\%$ ³⁾
Auswahl mögl. Frequenzen	eine feste (z.B. Atmega32U4 8 MHz), die per Prescaler geteilt werden kann (CLKPR)	große Auswahl (alle Baudraten)	kleine(re) Auswahl
notwendige externe Bauteile	keine	2 weitere Kondensatoren und Quarz	nur Oszillator
Geeignet für	kostenkritische Anwendungen	zeitkritische Anwendungen: UART, USART, genaue Uhr	

Details sind im Data Sheet unter Kapitel “Clock Sources” (bei ATmega32U4 Kapitel 6.2) zu finden. Für energiesparende Anwendungen sollte der Takt reduziert werden (siehe Tabelle).

Frequenz	Anwendung
32,768 kHz	für energiesparende Schaltungen (Stimmgabelquarz)
18,432 MHz	Baudratenquarz , notwendig um viele Schnittstellentakte zu generieren. Maximalwert für die meisten AVR-Chips

- Weitere passive Komponenten
 - Überprüfen Sie das Data Sheet auf weitere notwendige passive Elemente, z.B. für interne Regler o.ä.
Diese sind unter anderem über Suchwörter wie “Design Guidelines” zu finden.
- USB Anschluss / Anschluss an externe Elektronik
 - Bei digitalen Anschlüssen an externe Elektronik bietet es sich an ESD Schutzdioden zu verwenden, um die Komponenten zu schützen.

Analoge Eingangswerte (ADC)

- Nutzen Sie für alle analoge Komponenten (Operationsverstärker, MOSFETs, ...) die Signale “Analog Vcc” (AVCC) und “Analog Ground” (AGND). Damit reduzieren sich Störungen auf der analogen Seite.
- Analog Ground und Digital Ground sollten nahe am Prozessor nur an einer Stelle verbunden werden. Dazu ist am einfachsten ein kurzgeschlossener Lötjumper zwischen AGND und GND zu verwenden.
- Sinnvoll beim Einlesen von Analogwerten ist es weiterhin ein Tiefpass vorzuschalten, um Spannungsspitzen abzdämpfen. Z.B. (Grenzfrequenz beachten!)
 - 1. Ordnung: Widerstand mit 1 kOhm und 100 nC gegen Masse, oder
 - 2. Ordnung: Induktivität mit 10 µH und 100 nC gegen Masse.

Digitale Schnittstellen

- Schalter als Input werden sinnvollerweise bei Schaltungen mit ATmega-Prozessoren gegen GND geschaltet. Dann können die internen Pull-up Widerstände genutzt werden.

- Falls Sie einen USB-Anschluss robust auslegen wollen, so sollten Sie sich an die üblichen Design Guides richten (z.B. [Design Guide von DataKey.com](#))
- **Wir wollen einen RasPi über I2C anschließen ODER Wir wollen 5V und 3,3V I2C an einem Bus haben.** Es ist zu beachten, dass der I2C des RasPi für High Pegel 3,3V liefert. Die meisten älteren Schaltungen benötigen 5V. Als Lösung können Sie den Levelshifter in folgende Application Note betrachten: [an10441.pdf](#)
- **Wie kann ich einen zweiten SPI an meinen ATmega anschließen?** Verwenden Sie dazu die USART Schnittstelle als zweite SPI. Details dazu finden Sie im Datenblatt des Prozessors unter USART. zweite Alternative: nutzen Sie einen geeigneten AVR-Chip (z.B. ATmega328PB)
- **Ich will viele Schalter einlesen. Ich will einen möglichst energiesparenden Aufbau.** dazu bietet sich ein Blick in folgende Dateien [PIC-Tipps](#). Einige der Tipps sind inzwischen überholt, können aber genutzt werden.
- Falls Sie einen SMD Mikrocontroller verwenden, sollten Sie nicht vergessen einen Programmierport herauszuführen - am besten [ISP](#) über [debugWire](#) (6-polige Buchse). Ansonsten wird das Programm nie den Weg auf den Chip finden.

Zum Abschluss

- **Prüfen, ob alle Teile in der Schaltung im Haupttraster liegen:** in der Kommandozeile "run snap-on-grid-sch.ulp" eingeben
- **Electrical-Rule-Check durchführen:** in der Kommandozeile "ERC" eingeben.
- Spannungsniveaus beschreiben: Schreiben Sie, für welche Spannung(en) die Schaltung ausgelegt ist. z.B. VCC: 3.3 ... 5.0 V, V_power: 20..36 V, Vin: 1.8 V

1)

Datenblatt ATmega32U4

2)

<https://www.mouser.de/search/refine.aspx?N=11749875>

3)

https://www.mouser.de/Passive-Components/Frequency-Control-Timing-Devices/Oscillators/_/N-7jdf/

From:

<https://wiki.mexle.org/> - **MEXLE Wiki**

Permanent link:

https://wiki.mexle.org/elektronik_labor/tipps_fuer_schaltungserstellung?rev=1585091795

Last update: **2021/05/09 10:07**

