

Versuch EP11

Datenerfassung mit dem Computer

I. Zielsetzung des Versuches

In den ersten 7 Versuchen des Elektronikpraktikums haben Sie sich hauptsächlich mit der Analogtechnik befaßt, wo Signale jeden beliebigen (Spannungs-)wert annehmen können. Über die Funktionsweise eines Digitalmultimeters haben wir dabei nicht weiter nachgedacht.

In den Versuchen EP8, EP9 und EP10 wurde andererseits nur die Digitaltechnik betrachtet, wo es nur zwei logische Zustände gibt.

In diesem Versuch werden Sie nun lernen, wie sich beide Bereiche miteinander verbinden lassen. Wir werden dabei untersuchen:

- Wie können digitale Werte in analoge Signale umgewandelt werden? Wie kann z.B. ein Computer kontinuierliche Spannungswerte erzeugen?
- Wie können analoge Werte digitalisiert werden? Wie kann z.B. ein Digitalvoltmeter kontinuierliche Spannungswerte erfassen und digital anzeigen?
- Wie lassen sich Zeiten oder Frequenzen messen?

II. Vorkenntnisse

Grundlagen der Digitaltechnik (Versuch EP8) und Programmierung von Mikrocontrollern (EP10)

Widerstandsnetzwerke (Spannungsteiler) und Schaltungen mit Operationsverstärkern (EP4)

III. Theorie zum Versuch

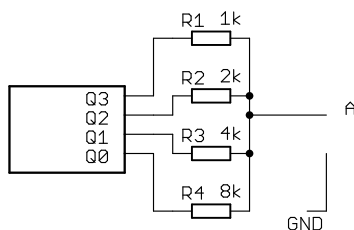
1. Umwandlung digitaler in analoge Signale, DAC

Die Umwandlung digitaler Signale in analoge Werte (Digital-Analog-Converter, DAC) kann über einen Spannungsteiler erfolgen, also ein Netzwerk von Widerständen, das von Digitalausgängen umgeschaltet wird.

Dabei gibt es verschiedene Lösungsansätze:

1.1. DAC mit binärem Netzwerk

Betrachten Sie die folgende Schaltung:



Die Ausgänge eines Binärzählers (hier z.B. 4 Ausgänge Q0 bis Q3) sind mit verschiedenen Widerständen versehen, die zu einem gemeinsamen Knoten A führen. Dabei stehen die Widerstandswerte im Verhältnis 1:2:4:8 zueinander. Der niederwertigste Ausgang Q0 (Binärwert 1) hat den größten Widerstand, der höchstwertige Ausgang Q3 (Binärwert 8) hat den kleinsten Widerstand.

Welche Spannung stellt sich an A für die 16 unterschiedlichen Binärwerte ein?

Betrachten wir zunächst die beiden einfachsten Fälle:

Wenn alle Q-Ausgänge auf 0 V liegen, ist A sicher auch auf 0 V.

Wenn alle Q-Ausgänge auf logisch 1 (Ausgangsspannung = U_0) liegen, ist A sicher auch auf U_0 . Wir gehen davon aus, daß A nicht mit einem Meßstrom belastet wird.

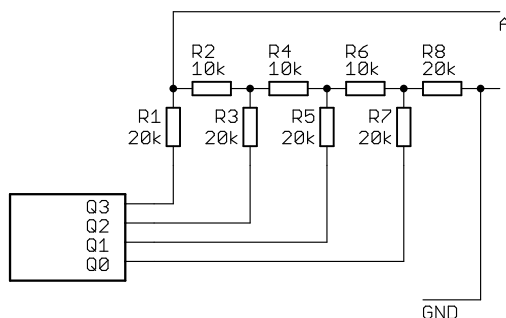
Für alle anderen Zustände läßt sich zeigen (siehe Anhang), daß sich aufgrund der unterschiedlichen Stromanteile der 4 Widerstände am Punkt A eine Spannung einstellt, die proportional zum Binärwert an Q0... Q3 ist:

$$U_A = U_0 \frac{\text{Binärwert}}{15} \quad \text{mit} \quad \text{Binärwert} = 0,1,2,\dots,15$$

Der Nachteil dieser Methode ist, daß mit zunehmender Auflösung die Widerstände beliebig genau werden müssen. Bei einem 10-Bit-System müssen der erste und der letzte Widerstand im Verhältnis von genau 1:512 zueinander stehen. Solch präzise Widerstände sind relativ teuer.

1.2. DAC mit R-2R-Netzwerk

Betrachten Sie die folgende Schaltung:



Die Ausgänge Q0 bis Q3 des 4-Bit-Binärzählers sind mit einem Netzwerk aus Widerständen versehen, die zu einem gemeinsamen Knoten A führen. Dabei werden nur Widerstandswerte im Verhältnis 1:2 verwendet. Der niederwertigste Ausgang ist wieder Q0 (Binärwert 1).

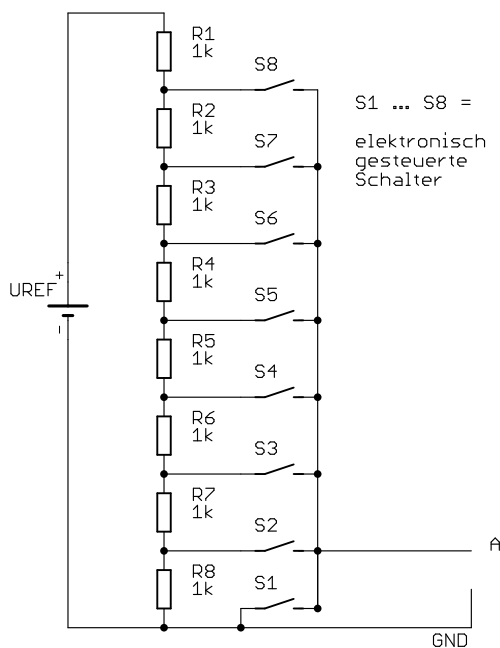
Auch hier läßt sich zeigen (siehe Anhang), daß sich am Punkt A eine Spannung einstellt, die wie folgt proportional zum Binärwert an Q0...Q3 ist (beachte: Nenner ist hier 16 statt 15):

$$U_A = U_0 \frac{\text{Binärwert}}{16} \quad \text{mit} \quad \text{Binärwert} = 0,1,2,\dots,15$$

Der Vorteil dieser Methode ist, daß auch bei zunehmender Auflösung immer nur Widerstände im Verhältnis 1:2 benötigt werden. Eine extreme Genauigkeit muß dieses Verhältnis nicht haben. Es läßt sich (gerade bei integrierten Schaltungen) auch sehr einfach herstellen, indem ausschließlich Widerstände *eines* Wertes R verwendet werden; an den Stellen mit 2R wird einfach eine Serienschaltung zweier dieser Widerstände eingesetzt.

1.3. DAC mit großem Netzwerk und Multiplexer

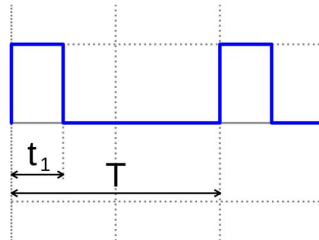
Eine weitere, sehr aufwendige Methode besteht darin, mit n Digitalausgängen über einen Dekoder ein System aus 2^n elektronischen Schaltern (einen n -nach- 2^n -Multiplexer) anzusteuern. Immer nur ein Schalter ist geschlossen und schaltet aus einem langen Spannungsteiler mit 2^n gleichen Widerständen ($2^n - 1$ Knoten und dazu noch GND) eine Knotenspannung an den Ausgang durch.



Der Nachteil dieser Methode ist der hohe Aufwand. Vorteilhaft ist, daß die Umwandlung garantiert monoton ist, d.h. größere Binärwerte führen garantiert zu größeren Ausgangsspannungen.

1.4. DAC mit Pulsweitenmodulation (PWM)

Zusätzlich zur Digital-Analog-Wandlung mit Widerstandsnetzwerken werden Sie in diesem Versuch auch die Pulsweitenmodulation (PWM) als einfache Methode mit nur einem digitalen Ausgang und einem RC-Tiefpass kennenlernen. Bei der Pulsweitenmodulation wird ein digitales Signal mit einer bestimmten Frequenz erzeugt, dessen Tastverhältnis (Verhältnis t_1/T) sich einstellen lässt:



Dieses Signal wird anschließend mit einem Tiefpass gefiltert, sodass die PWM-Frequenz möglichst stark unterdrückt wird. Da sich die (ideale) Tiefpassfilterung wie eine Integration des Signals verhält, lässt sich also für die gefilterte Spannung der folgende Zusammenhang finden:

$$U_a = \frac{1}{T} \int_0^T U(t) dt = \frac{1}{T} \int_0^{t_1} U_0 dt = U_0 \frac{t_1}{T} \quad (1)$$

Damit bestimmt letztlich das Tastverhältnis t_1/T den Spannungswert nach der Tiefpassfilterung. Die Grenzfrequenz des Tiefpasses muss dabei so gewählt werden, dass die Modulationsfrequenz genügend stark unterdrückt wird, schnelle Signaländerungen aber nicht zu stark gedämpft werden.

2. Umwandlung analoger in digitale Signale, ADC

Die Umwandlung analoger Werte in digitale Signale erfolgt durch einen Analog-Digital-Converter (ADC). Er ist zentraler Bestandteil z.B. jedes Digitalvoltmeters.

Auch für einen ADC gibt es verschiedene Schaltungsmöglichkeiten:

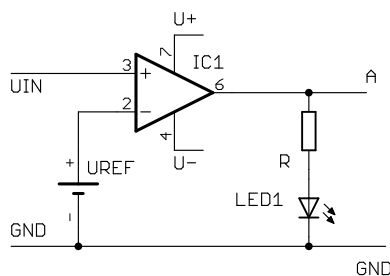
2.1. Spannungsvergleich mit Komparator

Die simpelste Methode zur Digitalisierung einer Spannung besteht darin, die Spannung über einen Komparator mit einer anderen zu vergleichen.

Ein Komparator ist im wesentlichen ein Operationsverstärker ohne Gegenkopplung:

- Sein Ausgang ist *logisch 1*, wenn sein nichtinvertierter (+) Eingang eine *höhere* Spannung hat als sein invertierter (-) Eingang.
- Sein Ausgang ist *logisch 0*, wenn sein nichtinvertierter (+) Eingang eine *niedrigere* Spannung hat als sein invertierter (-) Eingang.
- Der Fall, daß sein nichtinvertierter (+) Eingang die *gleiche* Spannung hat wie sein invertierter (-) Eingang, tritt praktisch nicht auf. (Selbst wenn beide Eingänge elektrisch miteinander verbunden sind, tritt eine interne Offsetspannung im Millivoltbereich hinzu.)

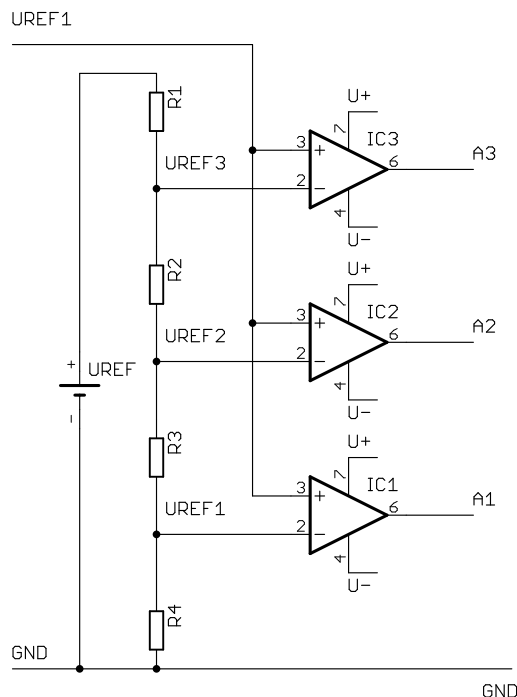
Die folgende Schaltung verdeutlicht das Prinzip. Die zu messende Eingangsspannung U_{IN} wird mit einer Referenzspannung U_{REF} verglichen. (U_+ und U_- sind die Versorgungsanschlüsse.) Ist U_{IN} größer als U_{REF} , so geht der Ausgang A auf 1 (LED leuchtet).



2.2. ADC mit Komparatoren, Flash-ADC

Möchte man die Eingangsspannung mit mehr als nur einem Bit digitalisieren, nimmt man mehrere Komparatoren. Für n Bits müssen es $2^n - 1$ Komparatoren sein. Über einen langen Spannungsteiler mit 2^n gleichen Widerständen werden $2^n - 1$ Knotenspannungen erzeugt.

Die folgende Schaltung verdeutlicht das Prinzip. Die zu messende Eingangsspannung U_{IN} (im Schaltplan U_{REF1}) wird mit mehreren Referenzspannungen verglichen.



Es ergeben sich auf diese Weise 2^n Zustände:

0. = alle Komparatorausgänge auf 0

1. = Komparator 1 auf 1

2. = Komparatoren 1 und 2 auf 1

3. = Komparatoren 1, 2, 3 auf 1

...

$2^n - 1$ = alle $2^n - 1$ Komparatoren auf 1

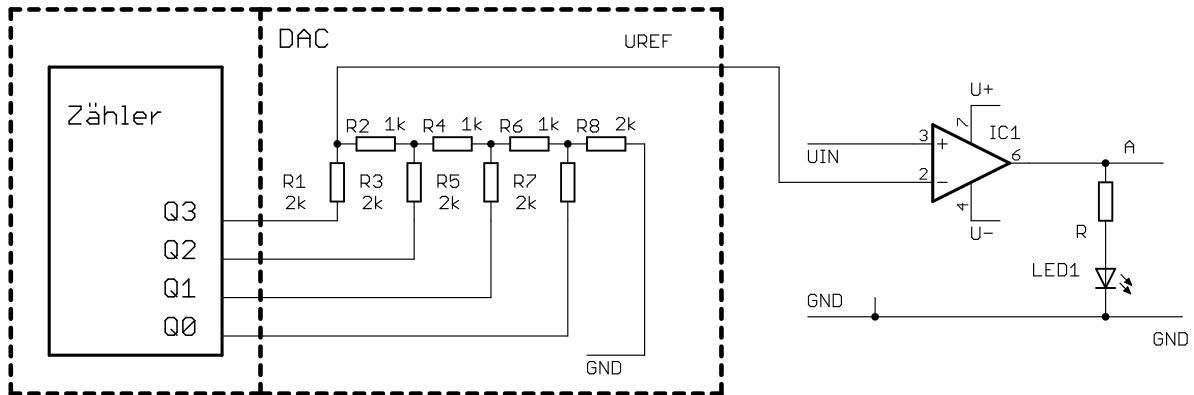
Über einen 2^n -nach- n -Dekoder werden diese 2^n Zustände in n Bits zusammengefaßt.

Der Nachteil dieser Methode ist der hohe Aufwand. Vorteilhaft ist, daß die Umwandlung sofort erfolgt. Man spricht auch von Flash-ADCs und setzt diese immer dann ein, wenn mit sehr hoher Geschwindigkeit, aber relativ niedriger Auflösung umgewandelt werden soll. Beispiel: Digitaloszilloskope haben oft nur 8 Bit Auflösung (=256 Stufen), müssen aber im GHz-Bereich abtasten/wandeln können.

2.3. ADC mit Zählverfahren

Möchte man mit sehr hoher Auflösung digitalisieren, läßt sich dies mit dem Flash-ADC nur mit erheblichem Aufwand realisieren: Man müßte für 16 Bit $2^{16} = 65536$ Widerstände und Komparatoren aufbauen.

Die Lösung besteht darin, die Referenzspannung durch einen DAC zu erzeugen und nur noch einen Komparator zu verwenden.



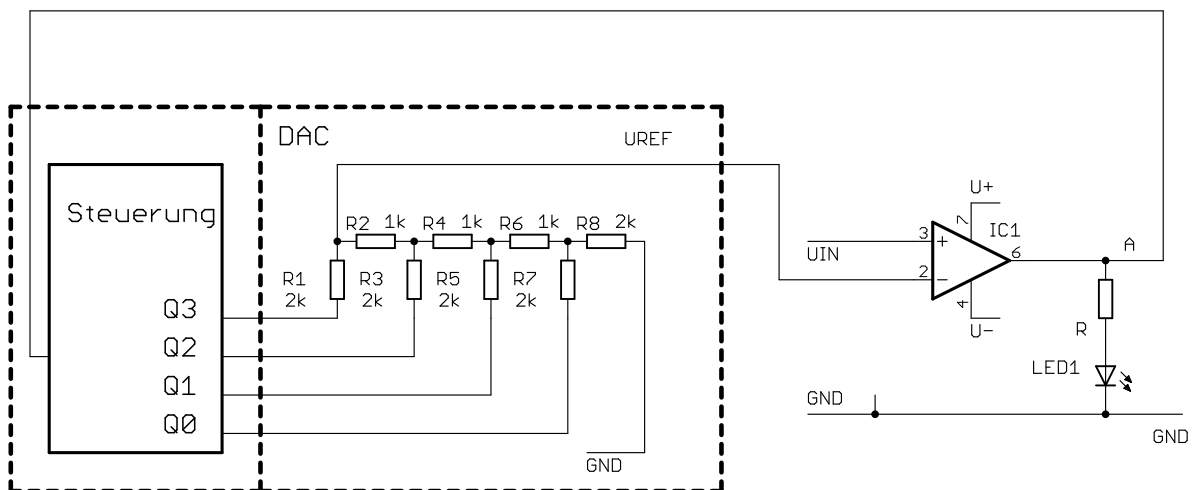
Beim Zählverfahren läuft ein Zähler kontinuierlich aufwärts und erzeugt mit seinem Widerstandsnetzwerk nacheinander 2^n Referenzspannungen, d.h. eine zunehmende Referenzspannung. Sobald die Referenzspannung die Eingangsspannung U_{IN} übersteigt, springt der Komparatorausgang auf 0. Den zugehörigen Zählerzustand speichert man ab und hat damit den gesuchten Digitalwert.

Der Vorgang wiederholt sich endlos, denn nach dem Zählerzustand $2^n - 1$ kommt wieder der Zustand 0.

Der Nachteil dieser Methode ist, daß man bis zu 2^n Zählschritte für die Digitalisierung benötigt. Das Verfahren ist also recht langsam. Vorteilhaft ist aber der geringe Aufwand.

2.4. ADC mit Wägeverfahren, Sukzessive Approximation

Man kann einen aus DAC und Komparator bestehenden ADC deutlich beschleunigen, wenn die Steuerung des DACs intelligenter gestaltet wird. Anstatt den DAC mit einem einfach nur hochlaufenden Zähler anzusteuern, wird Bit für Bit gesetzt und der Komparatorausgang immer wieder abgefragt. Die Schaltung:

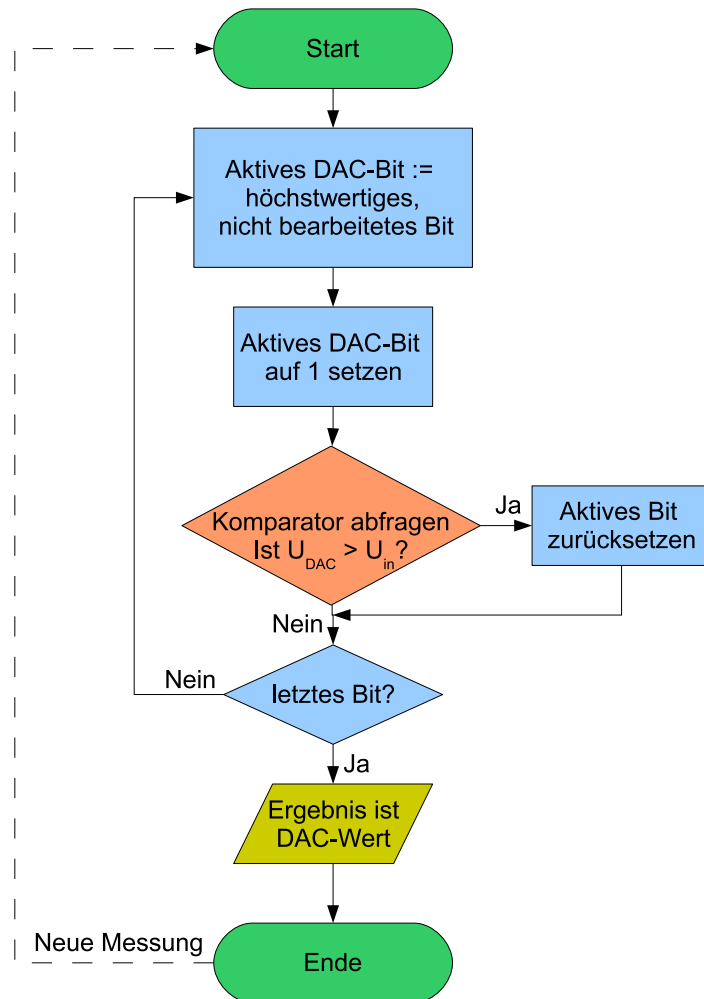


Der Ablauf ist wie folgt:

- Man setzt den DAC zunächst auf halbe Maximalspannung, indem man seine Bits auf halben Maximalwert setzt. Dazu wird allein das höchstwertige Bit gesetzt, beim 4-Bit-DAC also 1000.
 - Falls der Komparatorausgang (A) auf 1 geht, ist U_{IN} größer als diese DAC-Spannung (die halbe Maximalspannung). Die DAC-Spannung wird jetzt auf $3/4$ des Maximalwertes gesetzt, beim 4-Bit-DAC also 1100.

- Falls aber der Komparatorausgang (A) auf 0 geht, ist UIN kleiner als die DAC-Spannung (die halbe Maximalspannung). Das höchstwertige Bit wird wieder auf 0 gesetzt und die DAC-Spannung wird jetzt auf 1/4 des Maximalwertes gesetzt, beim 4-Bit-DAC also 0100.
- Wieder wird der Komparatorausgang abgefragt und je nachdem ob er 0 oder 1 bleibt das entsprechende Bit gesetzt oder wird wieder zurückgesetzt.
- So wird fortgefahren, bis nacheinander alle Bits bestimmt wurden. Das Ergebnis wird in einem Register abgespeichert.

Anschaulicher wird dies Verfahren durch folgendes Flußdiagramm:



Der Vorteil dieser Methode ist, daß man statt 2^n nur noch n Zählschritte für die Digitalisierung benötigt. Das Verfahren ist zwar n -mal langsamer als der Flash-ADC, aber bedeutend schneller als das Zählverfahren.

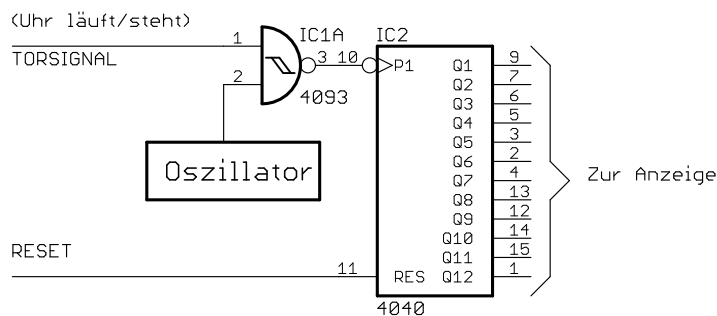
3. Zeit- und Frequenzmessung

Mit Hilfe von Binärzählern aus der Digitaltechnik lassen sich auch Zeiten und Frequenzen messen. Dazu benötigt man zusätzlich noch eine genaue Referenzzeit, die von einem präzisen Oszillator (z.B. einem Quarzoszillator) geliefert wird.

3.1. Zeitmessung

Bei der Zeitmessung wird ein Binärzähler so genutzt, dass dieser mit einer fest definierten und präzisen Taktfrequenz zählt. Treten nun Ereignisse auf, die man zeitlich messen möchte, so muss man nur den jeweiligen Zählerstand zum Zeitpunkt des Ereignisses auslesen. Hat man zwei Ereignisse beobachtet, so ist die Differenz der Zählerstände der zeitliche Abstand beider Signale in Taktzyklen des Binärzählers. Dessen Frequenz bestimmt somit auch die Auflösung mit der sich Zeiten messen lassen.

Die folgende Schaltung verdeutlicht das Prinzip eines Zeitmessers:

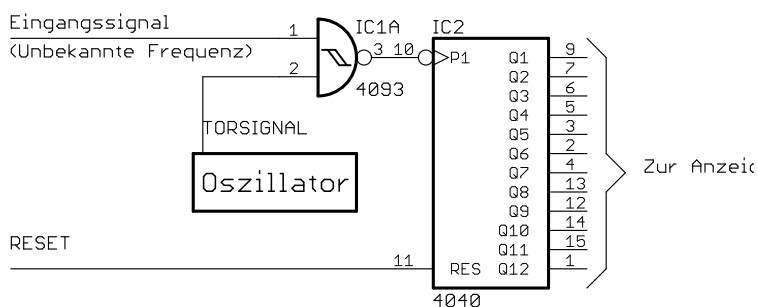


Hierbei ist das *Torsignal* ein zu messendes Signal, welches angibt, ob der Zähler läuft oder nicht. Außerdem erlaubt es ein Reset-Signal, den Zähler wieder auf Null zurückzusetzen.

3.2. Frequenzmessung

Die Frequenzmessung arbeitet nach einem ähnlichen Prinzip wie die Zeitmessung. Allerdings werden hier Ereignisse (z.B. Flanken eines Signals) in einer festgelegten Zeitspanne gezählt. Wählt man als Zeitspanne genau 1 s, so ist der Zählerstand der Ereignisse direkt die (mittlere) Frequenz des Signals in Hz.

Die folgende Schaltung verdeutlicht das Prinzip eines Frequenzzählers.

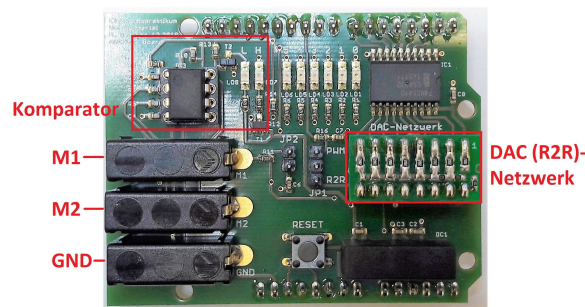


- Das *Torsignal*, welches bestimmt, ob die Uhr läuft oder steht, kommt von einem präzisen Oszillator. Gegebenenfalls wird dessen Frequenz heruntergeteilt, so daß Meßzeiten (Torzeiten) von z.B. 1 s entstehen.
- Das Eingangssignal (mit zu messender, unbekannter Frequenz) wird in einer AND-Verknüpfung mit dem Torsignal auf den Zählereingang gegeben.
- Eine Ablaufsteuerung sorgt dafür, daß das Torsignal nur einmal gegeben wird bzw. vor einem neuen Torsignal der Zähler zurückgesetzt wird (Reset).

IV. Versuchsdurchführung

1. Allgemeines zum Mikrocontrollerboard

Für diesen Versuch verwenden Sie wieder den Arduino Uno als Mikrocontrollerboard. Dieser wird um einen speziellen Shield ergänzt, der den Komparator sowie das Widerstandsnetzwerk für die Digital-Analog-Wandlung enthält. Der Shield ist wie folgt aufgebaut:



Auf der rechten Seite des Shields wird das Widerstandsnetzwerk (kleine grüne Platine mit 17 Widerständen) in einen 16-poligen Sockel gesteckt. Achten Sie unbedingt darauf, dass die Kerbe *rechts* ist! Das Widerstandsnetzwerk lässt eine Auflösung von 8 Bit zu, wobei in diesem Versuch aus technischen Gründen nur 6 Bit verwendet werden. Die untersten 2 Bit sind auf Masse gelegt und werden daher nicht verwendet. Welcher Binärwert gerade am Netzwerk anliegt, sehen Sie an den 6 LEDs in der Mitte. Die rechte (LD1) hat den niedrigsten Wert, also 1, die linke (LD6) hat den höchsten Wert, also 32. Damit hat das DAC-Netzwerk einen Wertebereich von 0 bis 63, also 6 Bit. Oben links ist der Komparator (LM311), dessen Ausgangszustand Sie rechts daneben an den LEDs ablesen können. Die grüne LED leuchtet, wenn der Komparatorausgang auf 1 ist, die rote LED leuchtet, wenn der Komparatorausgang auf 0 ist.

Die zu messende Spannung geben Sie über die Buche M1 (links oben) an den Komparator. Vergessen Sie nicht, die zu vermessende Spannungsquelle auch mit GND zu verbinden (Buchse GND)! An der Buchse M2 ist außerdem der Ausgang des DAC verfügbar.

Der Arduino-Shield unterstützt zwei Arten für die Digital-Analog-Wandlung. Über den Jumper JP1 kann zwischen beiden Varianten ausgewählt werden. Eine Variante ist die D/A-Wandlung über das R2R-Netzwerk. Die andere Variante ist die Wandlung über eine Pulsweitenmodulation (PWM).

Die vom Widerstandsnetzwerk oder PWM-Generator erzeugte DAC-Spannung gelangt auf den invertierten Eingang des Komparators. Auf den nichtinvertierten Eingang des Komparators geben Sie über Buchse M1 eine einstellbare Spannung aus dem HAMEG-Netzgerät. Stellen Sie zunächst etwa 2 V ein. Das ist Ihre zu messende Spannung.

Die Ausgangsspannung des Komparators gelangt auf den Eingang D9 Ihres Arduino Uno.

2. Arduino-Bibliothek

Für diesen Versuch können Sie auf eine vorgefertigte Arduino-Bibliothek zurückgreifen, die Ihnen die grundlegenden Funktionen des Shields als fertige Funktion zur Verfügung stellt. Bevor Sie beginnen, müssen Sie die Bibliothek eventuell noch installieren. Dazu laden Sie sich die EP11Shield.zip vom Praktikums-USB-Stick und wählen unter *Sketch - Bibliothek einbinden - .ZIP Bibliothek hinzufügen*. Anschließend öffnet sich ein Fenster, in dem Sie die Bibliothek vom USB-Stick einfach auswählen.

Möchten Sie die Bibliothek im Anschluss dann nutzen, so nutzen Sie bitte folgende Vorlage für den Sketch:

```
#include <EP11Shield.h>

void setup() {
  // EP11 Shield initialisieren
  EP11Shield::begin();

  // weitere Initialisierung hier
}

void loop() {
}
```

Die EP11Shield-Bibliothek stellt Ihnen folgende Funktionen zur Verfügung:

- EP11Shield::begin()
Initialisierung der EP11Shield-Bibliothek. Muss zwingend in setup() aufgerufen werden!
- EP11Shield::end()
Setzt alle von der Bibliothek veränderten Konfiguration auf Standardeinstellungen zurück.
- EP11Shield::DACwrite(value)
Setzt den DAC-Wert (Wertebereich 0..63)
- EP11Shield::COMPread()
Liest den Komparatorausgang ein. Rückgabe: HIGH/LOW
- EP11Shield::PULSEread()
Misst die Pulslänge eines Impulses an M1 in Mikrosekunden.
- EP11Shield::FREQread(gatetime)
Misst die Frequenz des Signals an M1. Optional kann die Torzeit in ms (gatetime) angegeben werden. Ist nichts angegeben, gilt 1000 ms=1 s für die Torzeit.

3. Umwandlung digitaler in analoge Signale, DAC

3.1. DAC mit R2R-Netzwerk (8-Bit-Modul)

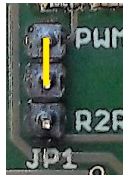
Untersuchen Sie zunächst die D/A-Wandlung mit Hilfe eines R2R-Netzwerks. Stellen Sie sicher, dass sich der Jumper JP1 in der Position R2R befindet:



Erstellen Sie nun einen einfachen Arduino-Sketch, der alle möglichen DAC-Werte von 0 bis 63 ausgibt. Nutzen Sie dazu die DACwrite()-Funktion aus der EP11Shield-Bibliothek. Messen Sie anschließend mit dem Oszilloskop den Spannungsverlauf am Ausgang des DAC (Buchse M2). Welchen zeitlichen Verlauf erwarten Sie? Können Sie die Stufenhöhe des DAC mit dem Oszilloskop messen? Welche Höhe erwarten Sie aus der Theorie?

3.2. DAC mit PWM

Stecken Sie nun den Jumper JP1 um, sodass Sie den PWM-Generator als DAC nutzen:



Messen Sie nun wieder den Signalverlauf mit dem Oszilloskop. Welche Unterschiede zum R2R-Netzwerk können Sie erkennen und haben Sie eine Erklärung dafür?

4. ADCs mit dem Mikrocontroller

4.1. ADC mit Zählverfahren

Im vorherigen Versuchsteil haben Sie gelernt, wie sie einfach digitale Werte mit Hilfe eines D/A-Wandlers in eine analoge Spannung wandeln können. Wir möchten nun analoge Signale digitalisieren. Dazu sollen Sie zunächst einen A/D-Wandler auf Basis des Zählverfahrens entwickeln.

Entwickeln Sie also eine Funktion `adcZaehl()`, in der ein Zähler von 0 bis 63 hochgezählt wird. Der Zählerwert wird über den D/A-Wandler in eine Spannung gewandelt, die der Komparator vergleichen kann. Sobald diese Spannung über der Spannung des unbekanntes Eingangssignals liegt, wird der Komparator das anzeigen und Sie können den digitalisierten Wert am Zähler ablesen. Den Komparator können Sie mit der Funktion `COMPread()` aus der EP11Shield-Bibliothek abfragen. Diese Funktion gibt `HIGH` zurück, wenn der DAC-Wert oberhalb Ihrer Eingangsspannung ist. Anschließend soll die Funktion den letzten Zählerwert als ADC-Ergebnis zurückgeben.

Testen Sie Ihre ADC-Funktion, indem Sie am Netzgerät verschiedene Spannungen einstellen und auf den Eingang M1 geben. Geben Sie das Ergebnis der A/D-Wandlung auf der seriellen Schnittstelle aus. Vergleichen Sie diese mit dem Wert, den Sie aufgrund der von außen angelegten Spannung erwarten würden. Stimmen die ADC-Messwerte mit Ihrer Erwartung überein? Wie verhält es sich mit der Wandlungszeit, die Ihr ADC für eine Wandlung benötigt? Ist diese immer konstant?

4.2. ADC mit Approximationsverfahren

In diesem Versuchsteil verändern Sie das Verfahren mit dem Ihr ADC arbeitet. Implementieren Sie nun analog zur `adcZaehl()`-Funktion eine Funktion `adcApprox()`, die eine A/D-Wandlung mit dem Verfahren der sukzessiven Approximation durchführt. Sie können sich an dem Ablaufdiagramm aus dem Theorieteil orientieren.

Hinweis: Sie können mit der Arduino-Bibliothek einzelne Bits in einer Variablen setzen und löschen. Nutzen Sie dafür die Funktionen `bitSet(x, n)` bzw. `bitClear(x, n)` um in der Variablen `x` das `n`-te Bit zu setzen oder zu löschen, wobei `n=0` das geringstwertigste Bit (Wertigkeit 1, ganz rechts!) ist.

Stellen Sie am Netzgerät wieder verschiedene Messspannungen ein und geben Sie den ADC-Wert auf der seriellen Schnittstelle aus. Beobachten Sie am Oszilloskop die DAC-Spannung an M2. Was erkennen Sie?

Wie verhält sich die Wandlungszeit im Vergleich zur Wandlungszeit bei dem Zählverfahren?

5. Zeit- und Frequenzmessung

5.1. Allgemeines

Als zu messende Impulse (Zeit bzw. Frequenz) verwenden Sie Ausgangsimpulse des HAMEG-Funktionsgenerators in einer bestimmten Einstellung. Die folgende Abbildung zeigt die Frontplatte des HAMEG-Funktionsgenerators.



Sie verwenden als Kurvenform: Rechteckgenerator mit einstellbarem Impuls-Pausen-Verhältnis, das ist die mit der Anzeige ganz rechts (im Bild oben leuchtet diese grün). Sehen Sie sich das Signal mit dem Oszilloskop an. Stellen Sie eine Signalamplitude von 5 V ein. Drücken Sie den Schalter links neben dem Amplitudenknopf herein, dann können Sie mit dem Knopf links neben diesem Schalter auch das Impuls-Pausen-Verhältnis verändern.

Hinweis: Stellen Sie sicher, dass der Jumper JP2 auf dem Shield für die folgenden Messungen **NICHT** gesteckt ist. Dieser Jumper aktiviert einen Tiefpassfilter, der Ihre Messung deutlich verschlechtert.

5.2. Zeitmessung

Schließen Sie das Signal des Funktionsgenerators (Buchse 50-Ω-OUTP.) an die **Buchsen M1 und GND** an. Verbinden Sie schwarz wie üblich mit GND. Achtung: Über das USB-Kabel und den Computer liegt das Board auf gemeinsamer Masse mit dem Funktionsgenerator.

In der EP11Shield-Bibliothek gibt es bereits eine fertige Funktion PULSEread(), die die Länge eines eingehenden Impulses (die Zeit von der ansteigenden bis zur abfallenden Flanke) in Mikrosekunden messen kann. Erstellen Sie einen Sketch, der diese Funktion nutzt und die gemessene Pulslänge auf der seriellen Schnittstelle ausgibt.

Machen Sie Messungen mit verschiedenen Pulslängen und vergleichen Sie die vom Mikrocontroller gemessene Zeit mit dem Meßwert auf dem Oszilloskop. Sollten Sie eine Pulslänge von 0 herausbekommen, so prüfen Sie, ob ein Impuls anliegt.

Was ist die kürzeste meßbare Zeit?

5.3. Frequenzzähler

Das Signal des Funktionsgenerators (Buchse 50-Ω-OUTP.) ist weiterhin an den **Buchsen M1 und GND**.

Nutzen Sie nun analog zur Zeitmessung die fertige Funktion FREQread() aus der EP11Shield-Bibliothek. Diese Funktion misst standardmäßig die Anzahl der steigenden Flanken innerhalb eines Messzeitraums von einer Sekunde. Dies ist dann die Frequenz des Eingangssignals in Hertz. Erstellen Sie einen Sketch, der diese Funktion nutzt und die gemessene Frequenz auf der seriellen Schnittstelle ausgibt.

Was ist die höchste Frequenz, die Sie messen können und was ist die niedrigste Frequenz? Können Sie mit der Funktion Frequenzen niedriger als 1 Hz messen? Wenn nein, warum ist das nicht so?

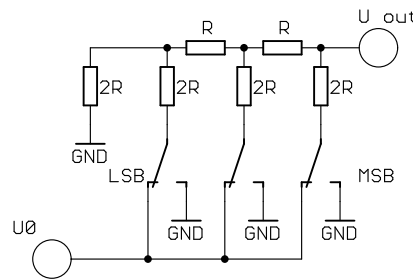
ANHANG

A Theorie zum R2R-Netzwerk

1. Aufbau eines R2R-Netzwerks

Ein R2R-Netzwerk besteht aus einer Anordnung von Widerständen der Größe R und der doppelten Größe, also $2R$. Dabei gilt (siehe Schaltskizze):

- Es werden n Widerstände der Größe $2R$, über zugehörige Schalter entweder mit einer Referenzspannung U_0 oder mit 0 Volt (Ground, GND, Masse) verbunden.
- Jeder der Schalter wird durch ein bestimmtes Bit eines aus n Bit bestehenden Datenwortes gesteuert.
- Der Schalter ist mit der Referenzspannung U_0 verbunden, wenn das zugehörige Steuerbit 1 ist.
- Der Schalter ist mit GND verbunden, wenn das zugehörige Steuerbit 0 ist.
- Die $2R$ -Widerstände sind untereinander über Widerstände der Größe R verbunden.
- Der $2R$ -Widerstand des niederwertigsten Bits (LSB) ist außerdem über einen Widerstand $2R$ mit GND verbunden.
- Der $2R$ -Widerstand des höchstwertigsten Bits (MSB) stellt an seiner Verbindung mit dem R -Widerstand die Ausgangsspannung U_{out} zur Verfügung.

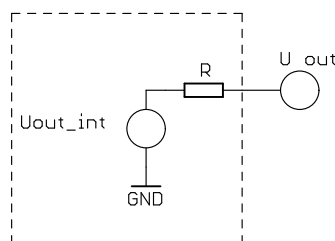


2. Behauptung:

Diese Ausgangsspannung ist (unbelastet)

$$U_{out,int} = U_0 \frac{d}{2^n} \quad \text{mit dem Datenwort:} \quad d = 0, \dots, 2^n - 1$$

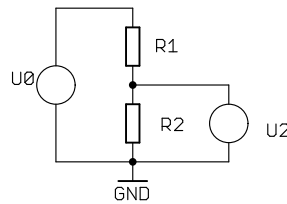
und sie erscheint als U_{out} mit einem Innenwiderstand der Größe $R_i = R$.



3. Beweis

3.1. Allgemeine Voraussetzungen (Netzwerktheorie)

Für einen einfachen Spannungsteiler gilt unbelastet:

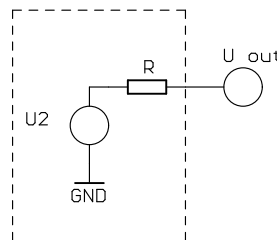


Die Ausgangsspannung über R_2 ist unbelastet $U_2 = U_0 \frac{R_2}{R_1 + R_2}$

Ein solcher Spannungsteiler entspricht einer Spannungsquelle mit:

- Quellspannung = unbelasteter Ausgangsspannung = U_2
- und einem Innenwiderstand R_i , der der Parallelschaltung der beiden Teilerwiderstände R_1 und R_2 entspricht, also

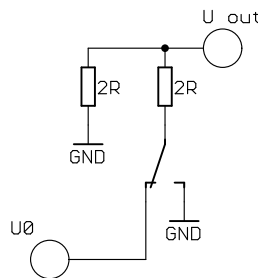
$$R_i = R_1 \parallel R_2 = \frac{1}{\frac{1}{R_1} + \frac{1}{R_2}} = \frac{R_1 \cdot R_2}{R_1 + R_2}$$



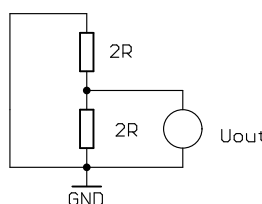
(Beweis siehe Anhang B: Theorie Spannungsteiler)

3.2. Beweis für R2R-Netzwerk mit $n = 1$

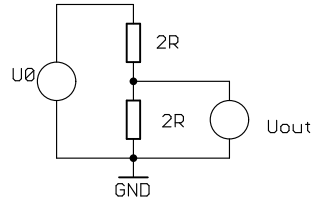
Es gibt nur ein Bit, das Datum kann $d = 0$ oder $d = 1$ sein:



Für $d = 0$ ist $2R$ mit GND verbunden und die Ausgangsspannung offenbar 0. Der Ausgang ist über die Parallelschaltung $2R \parallel 2R = R$ mit GND verbunden:



Für $d = 1$ ist $2R$ mit U_0 verbunden und die Ausgangsspannung offenbar (trivialer Spannungsteiler!) $\frac{1}{2}U_0$:



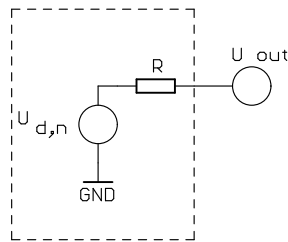
Auch hier ist der Innenwiderstand $R_i = R$.

Damit ist die Behauptung für $n = 1$ bewiesen.

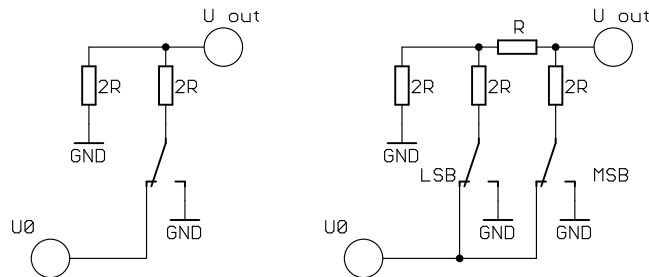
3.3. Beweis für R2R-Netzwerk für $n+1$, wenn es für n bewiesen wurde

Allgemeines Gegeben sei ein R2R-Netzwerk mit n Stufen. Es sei bewiesen worden, daß

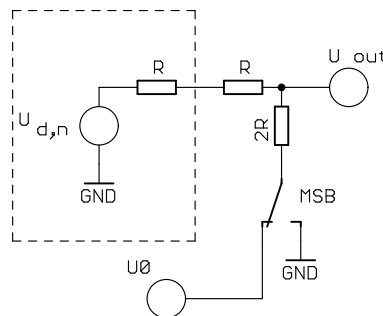
- es die Zustände $d = 0, \dots, 2^n - 1$ gibt,
- der Innenwiderstand $R_i = R$ ist,
- die Ausgangsspannung den Wert $U_{d,n} = \frac{d}{2^n}U_0$ hat



Wenn es nun eine Stufe (1 Bit) mehr gibt, so ändert sich die Schaltung wie folgt (Beispiel: von $n = 1$ nach $n = 2$):



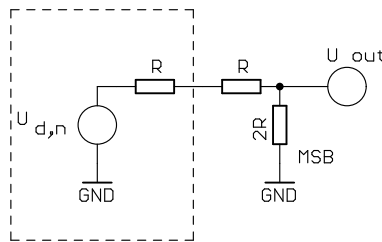
Mit dem bereits bewiesenen R2R-Netzwerk aus n Stufen sieht diese Schaltung so aus:



Fallunterscheidung

Jetzt gibt es 2 Fälle:

- Das MSB ist 0. Dann ist der $2R$ -Widerstand mit GND verbunden und es ergibt sich folgende Schaltung:

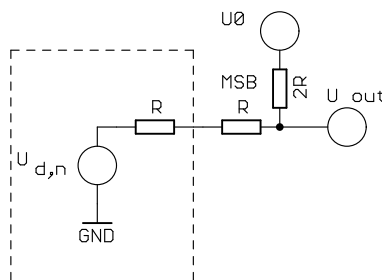


Offensichtlich ist die Ausgangsspannung:

$$U_{out,n+1,0} = \frac{1}{2}U_{out,n}$$

und der Innenwiderstand R_i ist R , da der Spannungsteiler offensichtlich über und unter dem Knoten aus jeweils $2R$ besteht (siehe oben, Netzwerktheorie).

- Das MSB ist 1. Dann ist der $2R$ -Widerstand mit U_0 verbunden und es ergibt sich folgende Schaltung:



Anwendung der Regeln für den Spannungsteiler liefert für die Ausgangsspannung:

$$U_{out,n+1,1} = U_{out,n} + \frac{1}{2}(U_0 - U_{out,n}) = \frac{1}{2}U_{out,n} + \frac{1}{2}U_0$$

Auch hier ist der Innenwiderstand R_i wieder R .

Es gibt somit die beiden neuen Gruppen möglicher Spannungswerte:

für MSB = 0: $\frac{1}{2}U_{out,n}$

und für MSB = 1: $\frac{1}{2}U_{out,n} + \frac{1}{2}U_0$

Zusammenfassung

- Für n Bit gilt:

$$U_{out,n,d} = U_0 \frac{d}{2^n} \quad \text{mit dem Datenwort: } d = 0, \dots, 2^n - 1$$

- Für $(n+1)$ Bit gibt es die beiden Gruppen (auch hier zunächst mit dem n -Bit Datenwort $d = 0, \dots, 2^n - 1$)

$$\text{für MSB} = 0 \quad \frac{1}{2} U_{out,n,d} = U_0 \frac{d}{2^{n+1}}$$

$$\text{für MSB} = 1 \quad \frac{1}{2} U_{out,n,d} + \frac{1}{2} U_0 = U_0 \left[\frac{d}{2^{n+1}} + \frac{1}{2} \right] = U_0 \left[\frac{d}{2^{n+1}} + \frac{1 \cdot 2^n}{2 \cdot 2^n} \right] = U_0 \left[\frac{d + 2^n}{2^{n+1}} \right]$$

- Es gibt im Zähler des Quotienten hinter U_0 demnach zwei Zahlenbereiche:

Für MSB = 0 gibt es $[0 \dots (2^n - 1)]$

und für MSB = 1 gibt es $[2^n + 0 \dots 2^n + (2^n - 1)] = [2^n + 0 \dots 2^{n+1} - 1]$

zusammen also den Gesamtbereich $d' = [0 \dots (2^{n+1} - 1)]$

Fazit:

Für $n + 1$ gilt:

$$U_{out,n+1,d'} = U_0 \frac{d'}{2^{n+1}} \quad \text{mit } d' = [0 \dots (2^{n+1} - 1)]$$

und der Innenwiderstand ist $R_i = R$.

Damit ist der Beweis erbracht.

B Theorie Spannungsteiler

1. Behauptung

Ein Spannungsteiler aus den Widerständen R_1 und R_2 , der an die Spannung U_0 angeschlossen ist, liefert an seinem Knoten (über R_2) eine Spannung

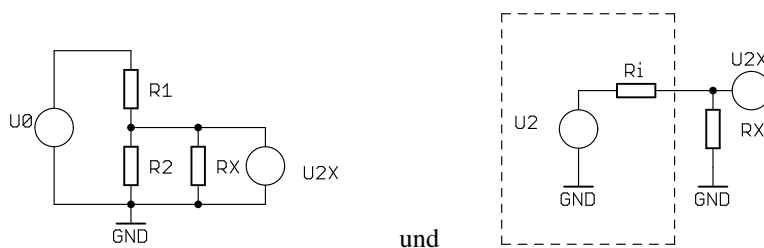
$$U_2 = U_0 \frac{R_2}{R_1 + R_2} \quad \text{mit einem Innenwiderstand} \quad R = R_1 \parallel R_2 = \frac{R_1 R_2}{R_1 + R_2}$$

also:

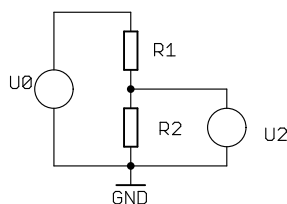


2. Beweis

Zeige, daß sich bei Belastung mit einem Widerstand R_x die Ausgangsspannung identisch ändert, beweise also die Äquivalenz der Schaltungen:



2.1. Vorbemerkung: Unbelasteter Spannungsteiler

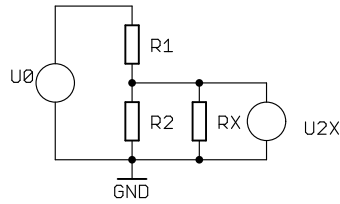


Die Eingangsspannung U_0 liegt an der Serienschaltung der Widerstände R_1 und R_2 , d.h. am Gesamtwiderstand $(R_1 + R_2)$.

Daher fließt ein Strom $I = \frac{U_0}{R_1 + R_2}$.

Dieser Strom führt an R_2 zu einem Spannungsabfall $U_2 = R_2 I = R_2 \frac{U_0}{R_1 + R_2} = U_0 \frac{R_2}{R_1 + R_2}$.

2.2. Belasteter Spannungsteiler (Rechnung über Parallelschaltung)



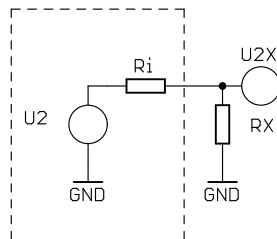
Beim belasteten Spannungsteiler liegt parallel zu R_2 der Lastwiderstand R_X . In der Formel für den Spannungsteiler kann also der bisherige Widerstand R_2 durch den Wert $R_{2X} = R_2 \parallel R_X = \frac{R_2 \cdot R_X}{R_2 + R_X}$ ersetzt werden. Dann ist:

$$U_{2X} = U_0 \frac{R_{2X}}{R_1 + R_{2X}}$$

Auflösen nach R_1, R_2, R_X liefert:

$$U_0 \frac{R_{2X}}{R_1 + R_{2X}} = U_0 \frac{\frac{R_2 R_X}{R_2 + R_X}}{R_1 + \frac{R_2 R_X}{R_2 + R_X}} = U_0 \frac{\frac{R_2 R_X}{R_2 + R_X}}{\frac{R_1(R_2 + R_X) + R_2 R_X}{R_2 + R_X}} = U_0 \frac{R_2 R_X}{R_1(R_2 + R_X) + R_2 R_X} = U_0 \frac{R_2 R_X}{R_1 R_2 + R_1 R_X + R_2 R_X}$$

2.3. Belasteter Spannungsteiler (Rechnung über Innenwiderstand)



Geht man hingegen von einer Spannungsquelle $U_2 = U_0 \frac{R_2}{R_1 + R_2}$ mit Innenwiderstand $R_i = R_1 \parallel R_2 = \frac{R_1 \cdot R_2}{R_1 + R_2}$ aus, die mit dem Widerstand R_X belastet wird, so lautet die Rechnung:

$$\begin{aligned} U_{2X} &= U_2 \frac{R_X}{R_i + R_X} = \left(U_0 \frac{R_2}{R_1 + R_2} \right) \frac{R_X}{R_i + R_X} = U_0 \frac{R_2 R_X}{(R_1 + R_2)(R_i + R_X)} = U_0 \frac{R_2 R_X}{R_1 R_i + R_1 R_X + R_2 R_i + R_2 R_X} = \\ &= U_0 \frac{R_2 R_X}{R_i(R_1 + R_2) + R_1 R_X + R_2 R_X} = U_0 \frac{R_2 R_X}{\frac{R_1 R_2}{R_1 + R_2}(R_1 + R_2) + R_1 R_X + R_2 R_X} = U_0 \frac{R_2 R_X}{R_1 R_2 + R_1 R_X + R_2 R_X} \end{aligned}$$

Das stimmt mit dem Ergebnis der Rechnung über Parallelschaltung überein!

C Theorie zum binären Netzwerk

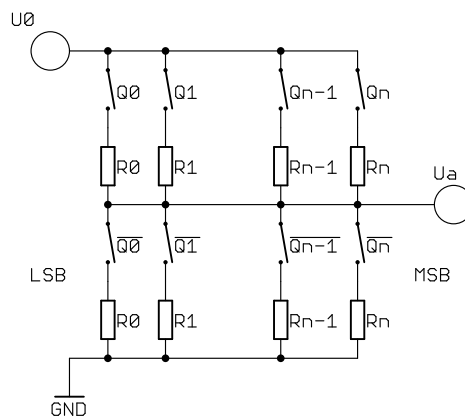
1. Aufbau eines binären Netzwerks

Ein binäres Netzwerk besteht aus einer Anordnung von Widerständen der Größe R , der doppelten Größe $2R$, der vierfachen Größe $4R$ usw. bis zur Größe $2^n R$ bei n Bit Auflösung. Man kann aber auch sagen: Die Widerstandswerte sind R_0 als größter verwendeter Widerstand sowie $\frac{R_0}{2}$, sowie $\frac{R_0}{4}$ usw. bis $\frac{R_0}{2^n}$.

Die Widerstände werden über binäre Ausgänge z.B. eines Binärzählers entweder mit 0 (0 Volt) oder 1 (Spannung U_0) verbunden.

Dabei gilt (siehe Schaltskizze unten, vgl. auch Schaltbild auf auf der ersten Seite):

- Der Widerstand $R_i = \frac{R_0}{2^i}$ wird vom i -ten Bit gesteuert.
- Dabei wird er entweder über einen Schalter Q_i mit einer Referenzspannung U_0 verbunden oder über einen Schalter \bar{Q}_i mit 0 Volt (Ground, GND, Masse) verbunden.
- Jeder der Schalter wird durch ein bestimmtes Bit eines aus n Bit bestehenden Datenwortes gesteuert.
- Der Schalter Q_i ist mit der Referenzspannung U_0 verbunden, wenn das zugehörige Steuerbit 1 ist.
- Der Schalter \bar{Q}_i ist mit GND verbunden, wenn das zugehörige Steuerbit 0 ist.
- Die Ausgangsspannung U_a ergibt sich als Spannung eines Spannungsteilers aus den beiden Widerständen R_a und R_b , wobei R_a die Parallelschaltung aller Widerstände mit geschlossenem Schalter Q_i ist und R_b die Parallelschaltung aller Widerstände mit geschlossenem Schalter \bar{Q}_i ist.



2. Berechnung dieses Spannungsteilers

Wie erwähnt, gilt für jeden Teilwiderstand:

$$R_i = \frac{R_0}{2^i}$$

Für die Widerstände R_a und R_b gilt:

$$R_a = \left[\sum_{i=0}^N Q_i \frac{1}{R_i} \right]^{-1} \quad \text{und} \quad R_b = \left[\sum_{i=0}^N \bar{Q}_i \frac{1}{R_i} \right]^{-1}$$

Für den Binärwert q , mit dem das Netzwerk (die Schalter Q_i) angesteuert wird, gilt:

$$q = \sum Q_i 2^i$$

Für das entsprechende Komplement des Binärwertes \bar{q} , mit dem das Netzwerk (die Schalter \bar{Q}_i) angesteuert wird, gilt:

$$\bar{q} = \sum \bar{Q}_i 2^i = 2^N - 1 - q$$

Für die Ausgangsspannung gilt die bekannte Spannungsteilerformel:

$$U_a = \frac{R_b}{R_a + R_b} U_0$$

wobei

$$R_a = \left[\sum_{i=0}^N \frac{Q_i 2^i}{R_0} \right]^{-1} = \frac{R_0}{\sum_{i=0}^N Q_i 2^i} = \frac{R_0}{q} \quad \text{und} \quad R_b = \frac{R_0}{\sum \bar{Q}_i 2^i} = \frac{R_0}{2^N - 1 - q}$$

Somit ist

$$U_a = \frac{\frac{R_0}{2^N - 1 - q}}{\frac{R_0}{q} + \frac{R_0}{2^N - 1 - q}} U_0 = \frac{q}{2^N - 1 - q + q} U_0 = U_0 \frac{q}{2^N - 1}$$

was zu beweisen war.

(MW 2/2017)

D ADCs in kommerziellen Digitalmultimetern

In den Digitalmultimetern, die Sie im Praktikum verwenden, befindet sich natürlich auch ein ADC. Er funktioniert nach dem Zählverfahren (siehe Seite 6), aber ohne einen DAC. Stattdessen wird ein Operationsverstärker in Integratorschaltung verwendet. Der Schaltkreis hat außerdem einen Taktgeber (RC-Oszillator). Die Messung funktioniert so:

1. Zuerst wird der Integrator-Kondensator entladen.
2. Dann wird die zu messende Spannung (z.B. im Bereich 0 bis 200 mV) für eine bestimmte Zeit (1000 Takte) integriert.
3. Danach wird eine präzise Referenzspannung (z.B. 100 mV) an den Integrator gelegt, und zwar mit negativer Polarität, damit sich der Integrator entlädt. Die Zeit (Anzahl der Takte) bis zur Entladung auf 0 Volt wird gemessen.
4. - Falls beide Spannungen gleich groß sind, werden beim Entladen ebenfalls 1000 Takte gezählt und es erscheint 1000 im Display, was (mit einen zusätzlichen Komma) als 100,0 mV zu lesen ist.
- Falls die zu messende Spannung das x -fache der Referenzspannung ist, braucht das Entladen x -mal länger, es werden also $x \times 1000$ Takte gezählt und es erscheint eine Zahl im Display, die dem Meßspannungswert entspricht.
Beispiele: Ist die Meßspannung nur halb so groß (50 mV), erscheint 500 im Display, was (mit einen zusätzlichen Komma) als 50,0 mV zu lesen ist. Ist die Meßspannung anderthalbmal so groß (150 mV), erscheint 1500 im Display, was (mit einen zusätzlichen Komma) als 150,0 mV zu lesen ist.
5. Für negative Eingangsspannungen wird die Polarität des Auf- und Entladevorgangs automatisch angepaßt und ein Minuszeichen im Display gezeigt.

Die Schritte 1 bis 5 wiederholen sich dauernd, typ. 3 bis 4mal pro Sekunde.

Der maximale Meßbereich ist (bei 100 mV Referenzspannung) ± 200 mV. Höhere Meßspannungen werden über einen Spannungsteiler (1:10, 1:100 usw.) reduziert.

Ströme werden über den Spannungsabfall an einem Meßwiderstand gemessen (z.B. 200 mA an 1 Ω werden zu 200 mV).

Widerstände werden gemessen, indem der unbekannte Widerstand und ein präziser Referenzwiderstand (z.B. 100 Ω , 1 k Ω , 10 k Ω usw.) in Serie geschaltet, d.h. vom gleichen Strom durchflossen werden. Der Spannungsabfall am unbekanntem Widerstand ist dann die Meßspannung, der Spannungsabfall am Referenzwiderstand ist dann die Referenzspannung. Der angezeigte Wert im Display ist dann das Verhältnis der Widerstände zueinander, und da man als Referenzwiderstände glatte 10er-Werte nimmt, ist die die Anzeige direkt als Widerstands-Meßwert ablesbar.

Kondensatoren kann man messen, indem eine Wechselfspannung an eine Operationsverstärker-Differenzierschaltung gelegt wird (unbekannter C an den Eingang, R in der Rückkopplung). Die Ausgangsspannung ist bekanntlich proportional zu RC und wird gemessen.

Eine solche Differenzierschaltung kann man auch mit Induktivitäten (Spulen) aufbauen. Induktivitäten kann man daher messen, indem eine Wechselfspannung an eine Operationsverstärker-Differenzierschaltung gelegt wird (unbekanntes L in der Rückkopplung, R an den Eingang). Die Ausgangsspannung ist proportional zu RL und wird gemessen.

Für weitere Details schauen Sie sich im Internet das Datenblatt des Bausteins ICL7106 oder ICM7106 an, der seit den 80er Jahren der Standardschaltkreis für einfache Digitalmultimeter ist.