

Karlsruhe, tt.mm.yyyy <- Beginn der Arbeit
eintragen

BACHELORARBEIT/MASTERARBEIT

TITEL DER ARBEIT

Bearbeiter: cand. el. Vorname Name

Prof. Dr.-Ing. M. XYZ

Betreuer: Betreuer

Abgabetermin: tt.mm.yyyy

Erklärung:

Hiermit versichere ich, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, sowie die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht und die Satzung des Karlsruher Institut für Technologie (KIT) zur Sicherung guter wissenschaftlicher Praxis in der jeweils gültigen Fassung beachtet habe.

NAME

Danksagung

Text der Danksagung

Kurzfassung

Deutsche Kurzfassung der Arbeit

Der Abstract/Kurzfassung soll den Leser informieren und möglichst neugierig darauf machen, was in der Abschlussarbeit zu erwarten ist. Sinnvoll ist es dabei, explizit die Lesergruppen anzusprechen, für die der Text besonders geeignet ist. Der Abstract ist sprachlich nüchtern und sachlich zu halten. Den Leser interessieren v. a. folgende Fragen: Was sind die wichtigsten Ergebnisse? Welche Methodik wurde wie angewendet? Was sind die wichtigsten Schlussfolgerungen usw.?

Hierbei ist es von Vorteil, wenn Stichwörter wie z.B. 'Multilevel-Umrichter' oder 'Asynchronmaschine' benutzt werden, da es die Suche erleichtert.

Abstract

Englische Kurzfassung der Arbeit

Inhaltsverzeichnis

Abkürzungsverzeichnis	xi
------------------------------	-----------

Formelverzeichnis	xiii
--------------------------	-------------

1 Setup	1
1.1 Installation	1
1.1.1 MiKTeX	1
1.1.2 Perl	1
1.1.3 TeXstudio	1
2 Latex Grundlagen mit dieser Vorlage	3
2.1 Absätze und Zeilenumbrüche	3
2.2 Aufzählungen	4
2.2.1 Punktiert	4
2.2.2 Nummeriert	4
2.3 Referenzen im Dokument	4
2.4 Abkürzungs- und Formelverzeichnis	5
2.4.1 Abkürzungen erstellen	6
2.4.2 Formelzeichen erstellen	6
2.4.3 Referenzierung	7
2.5 Literaturverweise	8
2.6 Pfade	9
2.7 Gleichungen	9
2.7.1 Neue Mathe-Befehle in dieser Vorlage	10
2.8 Zahlen und Einheiten	11
2.9 Abbildungen	12
2.9.1 includegraphics	12
2.9.2 figure	13
2.9.3 sidewaysfigure	13
2.9.4 subfigure	15
2.9.5 Extra	18
2.10 Tabellen	19
2.10.1 table	20
2.10.2 tabular	20
2.10.3 Beispieltabelle	21
2.10.4 Tabellen Generator	22
2.11 Quelltext	22

2.12 Extras	23
2.12.1 Paket Dokumentationen	23
2.12.2 Kein Perl	24
A Schaltpläne	25
B Abbildungsverzeichnis	27
C Tabellenverzeichnis	29
D Quellcodeverzeichnis	31

Abkürzungsverzeichnis

ETI Elektrotechnisches Institut

Formelverzeichnis

γ_R Winkel des Rotorfluss der Asynchronmaschine

\underline{i}_R komplexer Rotorstrom

$\underline{\Psi}_R$ komplexer Rotorfluss

R_R Rotorwiderstand

1 Setup

1.1 Installation

In diesem Kapitel wird beschrieben, welche Programme benutzt werden und wie diese Konfiguriert werden müssen, um diese Vorlage in vollem Umfang zu nutzen.

1.1.1 MiKTeX

Es wird die neuste Version von MiKTeX benötigt. Diese kann von <https://miktex.org/> für das jeweilige Betriebssystem heruntergeladen werden.

1.1.2 Perl

Falls diese Vorlage auf einem Windows System benutzt wird, ist zusätzlich Perl notwendig. Diese Vorlage wurde mit Strawberryperl getestet und erstellt (<http://strawberryperl.com/>). Strawberryperl kann auch genutzt werden, wenn keine Admin-Rechte auf dem jeweiligen Computer vorhanden sind. Auf Linux Systemen kann Perl falls notwendig mit `sudo apt-get install perl` nachinstalliert werden.

1.1.3 TeXstudio

Es wird die neuste Version von TeXstudio benötigt. <https://www.texstudio.org/>

Achtung: MiKTeX sollte vor TeXstudio installiert werden!

Damit TeXstudio das Abkürzungs- sowie das Symbolverzeichnis automatisch mit erstellt, muss TeXstudio noch etwas konfiguriert werden. Die einzelnen Schritte zur genauen Konfiguration sind in Abbildung 1.1 zu sehen.

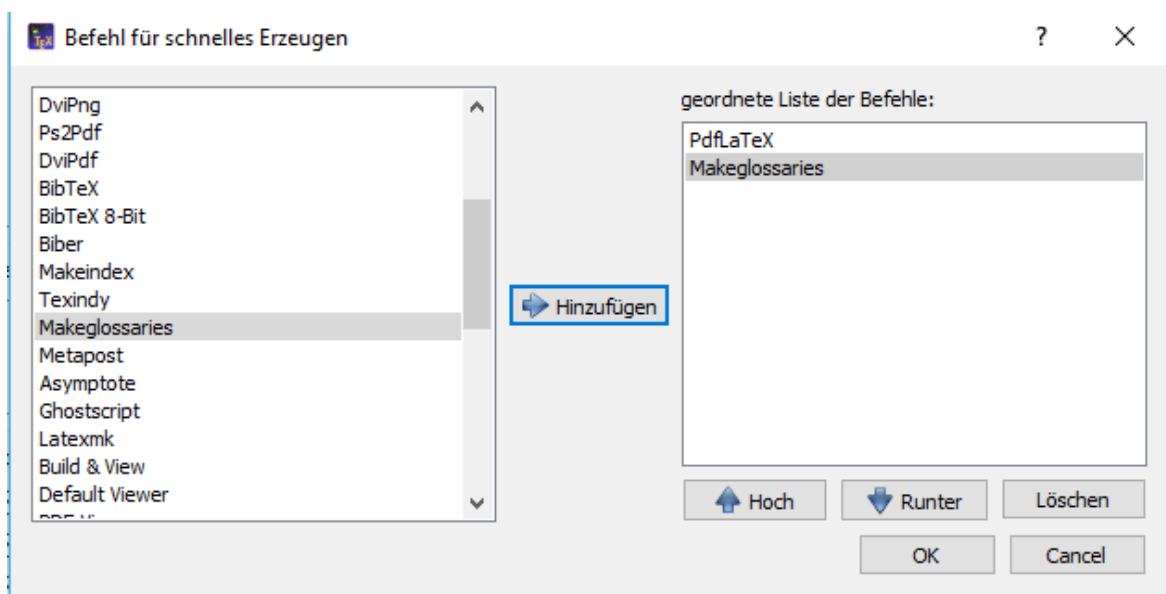
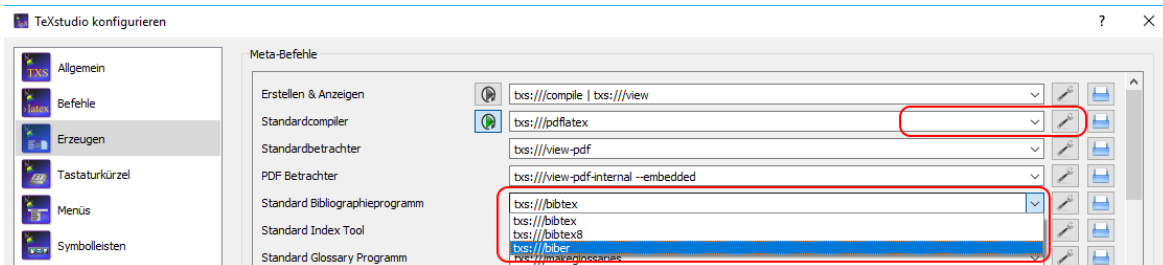
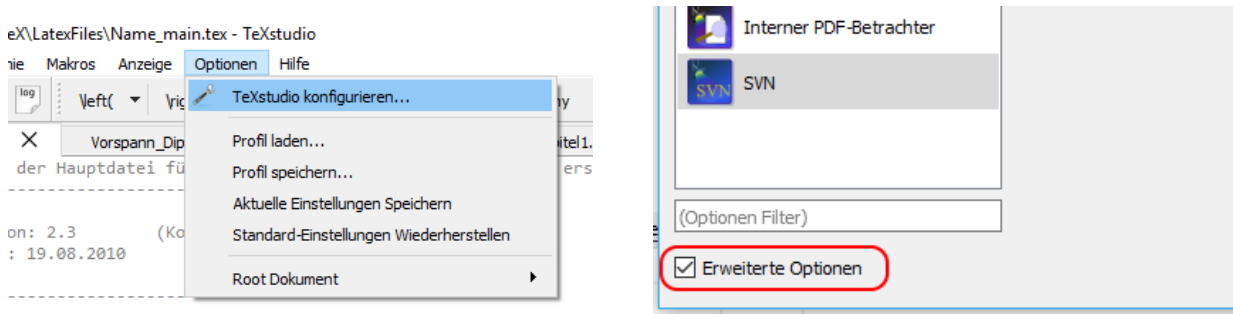


Abbildung 1.1: TeXstudio konfigurieren

2 Latex Grundlagen mit dieser Vorlage

Das Dokument wird durch die Hauptdatei `Name_main.tex` erzeugt. Zum bearbeiten sollte diese Datei immer als erstes geöffnet werden. TeXstudio erkennt dann automatisch welche Dateien zusätzlich eingebunden werden und zeigt diese auf der linken Seite an. Das Erstellen der Vorlage lässt sich von jeder Hierarchiestufe auslösen.

In der Hauptdatei folgt der eigentliche Inhalt des Dokuments. Zunächst wird die Titelseite erstellt, die fast unverändert übernommen werden kann. Es müssen lediglich einige Daten, wie Name, Betreuer, Titel, etc. Weiter unten in `Titelseite_Name.tex` muss auch noch die Kurzfassung und die Danksagung verfasst werden. Es empfiehlt sich die einzelnen Kapitel der Arbeit zur besseren Übersichtlichkeit in separaten Dateien zu verfassen, die mit `\include{}` oder `\input{}` in das Dokument eingefügt werden.

Dieses Beispielkapitel sollte aus der Arbeit entfernt werden. Sollten jedoch Fragen und Unklarheiten auftreten, soll dieses Kapitel als Beispiel dienen, wie die einzelnen Inhalte erstellt werden.

2.1 Absätze und Zeilenumbrüche

Zeilenumbrüche werden von L^AT_EX automatisch erzeugt und sollten **nicht** mit einem Befehl, wie `\\` erzeugt werden. Um einen Absatz zu erzeugen, wird einfach eine Leerzeile zwischen den Textblöcken gelassen.

Möchte man die Silbentrennung eines Wortes verändern bzw. korrigieren kann man die Trennstellen des Wortes so angeben: `Trenn\-\stel\-\len`. Möchte man das kein Zeilenumbruch zwischen zwei Wörtern, einem Wort und deinem Formelzeichen oder einem Wort und einer Zahl, so setzt man anstelle eines Leerzeichens eine Tilde (`~`).

2.2 Aufzählungen

2.2.1 Punktiert

Punktierte Aufzählungen werden mit der `itemize`-Umgebung erzeugt.

```
1 \begin{itemize}
2   \item Erster Punkt.
3   \item Zweiter Punkt.
4 \end{itemize}
```

- Erster Punkt.
- Zweiter Punkt.

2.2.2 Nummeriert

Nummerierte Aufzählungen werden mit der `enumerate`-Umgebung erzeugt.

```
1 \begin{enumerate}
2   \item Erster Punkt.
3   \item Zweiter Punkt.
4 \end{enumerate}
```

1. Erster Punkt.
2. Zweiter Punkt.

2.3 Referenzen im Dokument

Referenzen können auf verschiedenste Objekte mit `\label{präfix:name}` erzeugt und mit `\autoref{präfix:name}` referenziert werden. Dabei ist es sinnvoll die Labels mit einem sinnvollen Namen und einem Präfix, der angibt um was für eine Referenz es sich handelt, zu versehen.

`\autoref` fügt mithilfe des `Babel`-Pakets automatisch den Typ der Referenzierung z.B. Abbildung oder Tabelle vor der dazugehörigen Nummer ein. Der `\label`-Befehl muss bei Kapiteln immer hinter dem Kapitel stehen.

-
- 1 `\chapter{name}\label{sec:kapitelname}`
 - 2 `\section{name}\label{sec:sectionname}`
-

Wo bei Abbildungen, Tabellen, usw. der `\label`-Befehl stehen muss, wird in dem jeweiligen Kapitel erläutert.

Der Präfix hilft lediglich dem Autor den Überblick über das Dokument zu behalten und spielt keine Rolle dabei, welcher Typ wirklich referenziert wird. Damit das ganze aber etwas einheitlich bleibt hat man sich auf einige standardisierte Präfixe geeinigt. Die Liste der Präfixe ist in Tabelle 2.1 zu sehen.

Tabelle 2.1: Tabelle gängiger Präfixe für Referenzen.

Präfix	Beschreibung
sec	Kapitel und Unterkapitel
fig abb	Abbildungen
eq	Formeln und Gleichungen
tab	Tabellen
lst	Quellcode

2.4 Abkürzungs- und Formelverzeichnis

Wenn TeXstudio, wie in Unterabschnitt 1.1.3 beschrieben, konfiguriert wurde und Perl installiert ist, wird das Abkürzungs- und das Formelverzeichnis automatisch erstellt. Falls es nicht möglich ist Perl zu benutzen schauen Sie in Unterabschnitt 2.12.2.

Alle Abkürzungen und Formelzeichen werden in dieser Vorlage in `glossaretries.tex` definiert. Es landen aber automatisch nur die Abkürzungen und Formelzeichen in den Verzeichnissen, die mindestens eine Referenzierung im Dokument haben.

Es ist manchmal sinnvoll das Dokument mehrmals zu kompilieren, damit alle Verzeichnisse richtig erstellt werden.

2.4.1 Abkürzungen erstellen

Generell kann man alle Abkürzungen auch mit `\newglossaryentry{label}` aus Unterabschnitt 2.4.2 erstellen. Das gibt einem etwas mehr Freiheiten und auch die Möglichkeit mehr als einen Kasus zu definieren.

```

1 \newacronym[plural=LEDs,                                %1
2     longplural={light-emitting diodes}]                %2
3     {led}                                                %3
4     {LED}                                                %4
5     {light-emitting diode}                              %5

```

Tabelle 2.2: Beschreibung des newacronym-Befehls

Nummer	Beschreibung
1	Pluralform der Abkürzung (optional)
2	Pluralform der ausgeschriebenen Abkürzung (optional)
3	Label (wird nur zur Referenzierung verwendet)
4	Abkürzung
5	Langform/Bedeutung

2.4.2 Formelzeichen erstellen

```

1 \newglossaryentry{sym:rotorflusskompl}{                %1
2     name={\ensuremath{\uline{\Psi}_\mathrm{R}}},        %2
3     user1={\ensuremath{\uline{\dot{\Psi}}_\mathrm{R}}}, %3
4     user2={\ensuremath{\uline{\Psi}_\mathrm{R}^{\prime}}}, %4
5     user3={\ensuremath{\uline{\dot{\Psi}}_\mathrm{R}^{\prime}}}, %5
6     description={komplexer Rotorfluss},                %6
7     sort={Psir},                                       %7
8     type={symbolslist}                                %8
9 }

```

Um etwas zu unterstreichen muss in der dem `\ensuremath`-Befehl den `\uline{...}`-Befehl und **nicht** den `\underline{...}`-Befehl verwendet werden!

Tabelle 2.3: Beschreibung des newglossaryentry-Befehls für das Formelverzeichnis

Nummer	Beschreibung
1	Label (wird nur zur Referenzierung verwendet)
2	Das Formelzeichen. <code>\ensuremath</code> damit es im Text und in Gleichungen funktioniert.
3	User-definierte Darstellung für die erste Ableitung.
4	User-definierte Darstellung für die Transformierte.
5	User-definierte Darstellung für die transformierte, erste Ableitung.
6	Beschreibung
7	Name als Text (bestimmt die Sortierung)
8	Für das Formelverzeichnis immer <code>symbolslist</code> verwenden. Das Stichwort bestimmt in welchem Verzeichnis, der Eintrag landet.

2.4.3 Referenzierung

Das Erstellen eines Verweises im Text ist eine Trivialität. Es wird einfach der `\gls{label}`-Befehl verwendet. Beim Erstauftreten einer Abkürzung im Text wird automatisch die Langform in Klammern mit angegeben. Es kann sein, dass man im Text einen anderen Kasus benötigt, aber diesen nicht so in seinem Abkürzungsverzeichnis stehen haben will. Dazu kann `first` in der Definition überschrieben werden. Für Beispiele siehe in `glossaretries.tex`.

Tabelle 2.4: Beschreibung Referenzbefehle

Befehl	Beschreibung
<code>\gls{label}</code>	Abkürzung / Formelzeichen
<code>\glspl{label}</code>	Pluralform der Abkürzung
<code>\Gls{label}</code>	Abkürzung am Anfang des Satzes, damit der Anfang falls nötig groß geschrieben wird.
<code>\Glspl{label}</code>	Pluralform am Anfang des Satzes, damit der Anfang falls nötig groß geschrieben wird.
<code>\glssymbol{label}</code>	Symboldarstellung. Nur wenn die Option <code>symbol</code> angegeben wurde.
<code>\glsfirst{label}</code>	Erzwingt die Darstellung, die auch beim Erstaufruf gezeigt wird.
<code>\glsuseri{label}</code>	Erlaubt zugriff auf bis zu 6 verschiedene, user-definierte Darstellungen der Formelzeichen. Das wird oft benötigt, wenn man z.B. die Ableitung einer Größe darstellen möchte, ohne die Ableitung explizit im Verzeichnis aufführen zu müssen.
$a_{ref} \Rightarrow \dot{a}_{ref}$	
<code>\glsuservi{label}</code>	

Hier ein Beispiel, wie so etwas am Ende aussehen soll:

```

1 \begin{equation}
2   \num{0} =
3   \gluserii{sym:rotorwiderstand}\cdot\gluserii{sym:rotorstromkompl}
4   -\ju\gluserii{sym:gammarrotor}\gluserii{sym:rotorflusskompl}
5   +\gluseriii{sym:rotorflusskompl}\label{eq:formel}
6 \end{equation}

```

$$0 = R'_R \cdot i'_R - j\dot{\gamma}_R \underline{\Psi}'_R + \underline{\dot{\Psi}}'_R \quad (2.1)$$

2.5 Literaturverweise

Die verwendete Literatur wird in `Literaturverzeichnis.bib` eingetragen und mit dem `\cite`-Befehl referenziert. Die `.bib`-Datei kann mit TeXstudio bearbeitet werden. TeXstudio unterstützt das eintragen mit einigen hilfreichen Einträgen, die in dem `Bibliographie` Reiter zu finden sind. Dort einfach einen Eintrag z.B. Buch auswählen und alle bekannten Informationen eintragen. Einträge, die mit `OPT` beginnen, sind Optional. Falls so ein Eintrag doch angegeben wird, muss das `OPT` am Anfang entfernt werden.

Wenn mehrere Autoren an einem Werk beteiligt sind, werden diese einfach mit einem „and“ verbunden.

```

1 @book{MeyerLeistung, %ID für \cite
2   author = {Meyer, Manfred}, % mehr aut.: {Hans, Peter and Franz, Otto}
3   title = {Leistungselektronik - Einführung, Grundlagen, Überblick},
4   year = {2005},
5   edition = {1. Auflage},
6   publisher = {Springer-Verlag},
7   isbn = {9783642873485}
8 }

```

Des Weiteren sollte `date` mit `year` ersetzt werden, außer man hat ein genaues Datum, wie z.B. beim Abrufdatum einer Internetseite.

```

1 \cite[Seitenzahl Option]{ID}
2 \cite[12]{BeuthLeistung}
3 \cite[52\psq]{DoppelbauerHEF} % und nächste Seite

```

```
4 \cite[84\psqq]{HoffmannHybrid} % und folgende Seiten
```

Im Text wird mit dem `cite`-Befehl auf Buchquellen **BeuthLeistung**, Konferenzbeiträge **DoppelbauerHEF** und Internetquellen **HoffmannHybrid** verwiesen. **MeyerLeistung** Damit am Ende auch der Eintrag im Verzeichnis erscheint, muss mindestens einmal im Text darauf verwiesen werden.

2.6 Pfade

In L^AT_EX werden immer wieder andere Dateien, wie Bilder, weitere TeX- und/oder Quellcode-Dateien, eingebunden. Die Pfade sind dabei am besten **immer** Relativ zu der TeX-Datei anzugeben, die die Dateien einbindet.

Für Grafiken, die mit `\includegraphics` (Siehe Abschnitt 2.9) eingebunden werden wurde der Suchpfad schon automatisch in den Bilderordner gelegt. Das heißt für Bilder muss nur noch der Dateiname und nicht mehr der Pfad in den Bilderordner angegeben werden.

Es ist möglich Leerzeichen und weitere Punkte im Dateinamen zu haben, dafür muss dieser aber noch einmal gekapselt werden. Beispiel:

```
1 \includegraphics[width=0.9\textwidth]{1.5 Beispiel}.pdf
```

2.7 Gleichungen

Bei den Gleichungen ist auf die Notation nach DIN zu achten. Diese kann in den beigelegten Dateien im Überordner nachgelesen werden!

Mit `equation` können einzelne Formeln, wie Gleichung 2.2 aus **DoesselLEN** und **SiegeLES**, erstellt werden.

```
1 \begin{equation}
2   P = U \cdot I \label{eq:leistung}
3 \end{equation}
```

$$P = U \cdot I \quad (2.2)$$

Mit `align` können mehrere Formeln an einem Zeichen, z.B. dem Gleichheitszeichen, ausgerichtet werden.

```

1 \begin{align}
2 \quad \int_{-\infty}^{\infty} \exp\{-x^2\} \mathrm{d}x &= \sqrt{\pi} \\
3 \quad \int_0^{\infty} \frac{x}{e^x - 1} \mathrm{d}x &= \frac{\pi^2}{6} \\
4 \end{align}

```

$$\int_{-\infty}^{\infty} \exp -x^2 \, dx = \sqrt{\pi} \quad (2.3)$$

$$\int_0^{\infty} \frac{x}{e^x - 1} \, dx = \frac{\pi^2}{6} \quad (2.4)$$

Ein Stern `*` hinter `equation` oder `align` kann die Formel Nummerierung für diese Formel deaktivieren. Dies ist jedoch in den meisten Fällen nicht zu empfehlen.

Mathematische Symbole können auch in den Fließtext eingebunden werden, indem man das Symbol mit `$` Zeichen einschließt. Damit teilt man `LaTeX` mit, dass es sich hier um mathematische Symbole handelt. Das sollte nicht nur für Symbole, wie z.B. Ω , benutzt werden, sondern auch, um Formelzeichen, wie z.B. Widerstand R , hervorzuheben.

2.7.1 Neue Mathe-Befehle in dieser Vorlage

Die neuen Befehle wurden in `newcommands.tex` definiert.

Tabelle 2.5: Beschreibung der neuen Befehle

Befehl	Beispiel	Beschreibung
<code>\eqdef</code>	$a \stackrel{\text{def}}{=} b$	<code>def</code> über dem Gleichheitszeichen.
<code>\eqexcl</code>	$a \stackrel{!}{=} b$	Ausrufezeichen über dem Gleichheitszeichen.
<code>\geqexcl</code>	$a \stackrel{!}{\geq} b$	Ausrufezeichen über dem Vergleichszeichen.
<code>\leexcl</code>	$a \stackrel{!}{<} b$	Ausrufezeichen über dem Vergleichszeichen.
<code>\ju</code>	$a + jb$	Darstellung der imaginären Einheit j .
<code>\abs{...}</code>	$ a + jb $	Betrag.
<code>\norm{...}</code>	$\ a + jb\ $	Normiert.
<code>\Re{...}</code>	$\text{Re}\{a + jb\}$	Realteil.
<code>\Im{...}</code>	$\text{Im}\{a + jb\}$	Imaginärteil.

2.8 Zahlen und Einheiten

Für Einheiten steht das Paket `siunitx` zu Verfügung. Das Paket sorgt einerseits für die richtige Darstellung der Einheiten (Nicht kursiv geschrieben) und andererseits passt es die Notation an die lokale Spracheigenschaft an (Komma/Punkt der Tausenderstellen Deutsch/Englisch, Leerzeichen zwischen Zahl und Einheit, usw.).

Es empfiehlt sich alle Zahlen in einem Dokument, egal ob mit oder ohne Einheit mit den Paket befehlen einzubinden, da man so mit wenig Aufwand eine einheitliche Formatierung der Zahlen erreichen kann. Des Weiteren sollten die Abkürzungsbefehle für die Einheiten verwendet werden.

Komplexe Zahlen können direkt verwendet (Siehe Beispiel).

```

1 \SI{1}{\micro\meter\per\second} = \SI{3.6e-6}{\kilo\meter\per\hour}
2 \SI{1}{\per\second} = \SI{60}{\per\minute}
3 \num{5e3} = \num{5000}
4 (\complexnum{1+i2})\,,\si{\ampere} = \complexqty{1+i2}{\ampere}
5 \ang{90} = \SI{90}{\degree} = \SI[parse-numbers = false]{\pi/2}{\radian}

```

$$1 \mu\text{m s}^{-1} = 3,6 \cdot 10^{-6} \text{ km h}^{-1} \quad (2.5)$$

$$1 \text{ s}^{-1} = 60 \text{ min}^{-1} \quad (2.6)$$

$$5 \cdot 10^3 = 5000 \quad (2.7)$$

$$(1 + 2i) \text{ A} = (1 + 2i) \text{ A} \quad (2.8)$$

$$90^\circ = 90^\circ = \pi/2 \text{ rad} \quad (2.9)$$

Achtung: Die Kommastelle muss mit einem `.` geschrieben werden. Das Paket passt dann die Notation automatisch an die lokalen Spracheinstellungen an! Außerdem ist die Autovervollständigung der Einheit `\degree` falsch. Diese wird zu `\degres` korrigiert und das existiert nicht und gibt einen Fehler. Es wird empfohlen `\ang` zu benutzen.

2.9 Abbildungen

Hier ist ein einfaches Beispiel, wie man Abbildungen hinzufügt. Was man dabei alles Einstellen kann und was man dabei beachten muss wird in den Folgenden Unterkapiteln erklärt.

Im vorletzten Unterkapitel wird gezeigt, wie man Grafiken gedreht einbindet. Des Weiteren wird im letzten Unterkapitel erläutert, wie man mehrere Grafiken neben und untereinander einbindet.

```

1 \begin{figure}[htb]
2     \centering
3     \includegraphics[width=0.9\textwidth]{SDSB.pdf}
4     \caption{Selbstgeführte Drehstrombrückenschaltung}
5     \label{abb:stromrichter}
6 \end{figure}

```

2.9.1 includegraphics

Abbildungen werden mit dem Befehl `\includegraphics[options]{dateiname}` eingebunden. Die Dateierweiterung muss dabei nicht explizit angegeben werden. Mit `pdflatex` können direkt `.jpg`, `.png`, `.bmp` oder `.pdf` Dateien eingefügt werden.

Es wird empfohlen Vektorgrafiken zu verwenden. Leider können `.svg` Dateien nicht ohne Weiteres eingefügt werden, deshalb sollten die Grafiken einfach als `.pdf` gespeichert und eingebunden werden. Wie man Vektorgrafiken aus den am Elektrotechnisches Institut (ETI) gebräuchlichen Programmen bekommt wird im Wiki <https://wiki.eti.kit.edu> beschrieben. Ansonsten Fragen Sie ihren Betreuer.

Der wichtigste Parameter von `options` ist dabei `width`. Damit lässt sich die Breite des anzuzeigenden Bildes festlegen. Es können dabei direkte Angaben in z.B. cm, oder auch anderen Einheiten, gemacht werden. Es wird aber Empfohlen den Befehl `textwidth` zu benutzen und die Breite als Faktor der benutzbaren Seitenbreite zu definieren. Das ist besonders praktisch für variable Designs, wie `subfigure` oder `minipage`, da dort `textwidth` automatisch die zur Verfügung stehende Breite angibt.

```

1 \includegraphics[width=0.9\textwidth]{SDSB.pdf}

```

2.9.2 figure

Die `figure`-Umgebung ist ein Wrapper der dazu dient Abbildung anhand verschiedener interner Regeln automatisch an einer vorteilhaften Stelle innerhalb des Textes platzieren.

```

1 \begin{figure}[options]
2     % ...
3 \end{figure}

```

Dabei ist es möglich die bevorzugte Position/en in den Optionen anzugeben.

Tabelle 2.6: Optionen der figure-Umgebung.

Option	Beschreibung
h	here - also etwa dort wo die Abbildung im Text definiert wird
t	top - am Seitenanfang
b	bottom - am Seitenende
p	page - auf einer eigenen Seite einzeln oder mit anderen Abbildungen

2.9.3 sidewaysfigure

Das Paket `rotating` stellt die `sidewaysfigure`-Umgebung zur Verfügung, mit der man Bilder um 90° gedreht darstellen kann. Das ist besonders Praktisch für sehr breite Bilder, die am besten eine ganze Seite benötigen. Die Drehrichtung wird dabei anhand der Seitenzahl automatisch bestimmt, so dass das Bild immer nach Außen zeigt und die Unterschrift gut zu lesen ist.

```

1 \begin{sidewaysfigure}
2     \centering
3     \includegraphics[width=\textwidth]{testwide.png}
4     \caption[Bild 0]{\small Bild 0 seitlich dargestellt.}
5     \label{abb:bild0}
6 \end{sidewaysfigure}

```

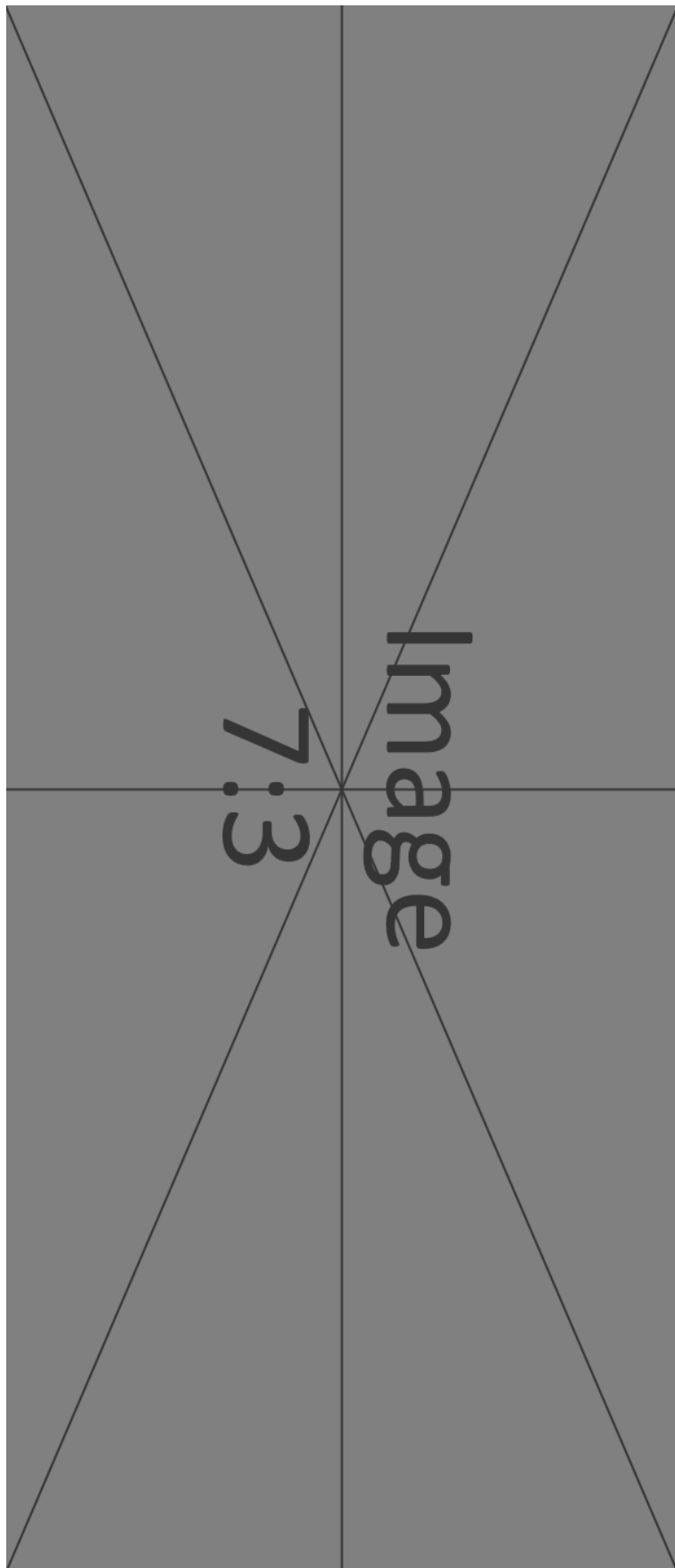


Abbildung 2.1: Bild 0 seitlich dargestellt.

2.9.4 subfigure

Mit dem `subcaption`-Paket wird die `subfigure`-Umgebung definiert mit der sich Grafiken mit zugehöriger Bildunterschrift in Container packen lassen, die dann in der `figure`-Umgebung eingebunden werden können.

Da es sehr viele Anordnungsmöglichkeiten gibt, wird hier nur auf einige nützliche Zusatzbefehle eingegangen gefolgt von einem großen Beispiel (siehe Abbildung 2.2), welches das Vorgehen veranschaulichen soll.

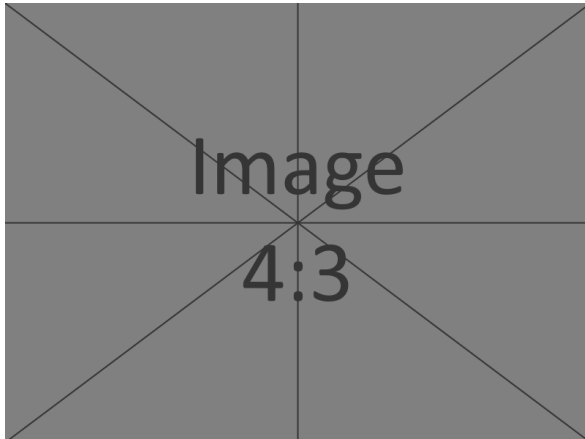
Tabelle 2.7: Nützliche Befehle für `subfigure`.

Befehl	Beschreibung
<code>\hfill</code>	Damit wird der horizontale Platz, der übrig bleibt, frei gelassen. Das erstellt einen Abstand und macht das linke Bild links und das rechte rechtsbündig. Bei noch mehr Bildern wird der Restabstand gleichmäßig aufgeteilt.
<code>\vskip\baselineskip</code>	Beginn einer neuen Bildzeile.

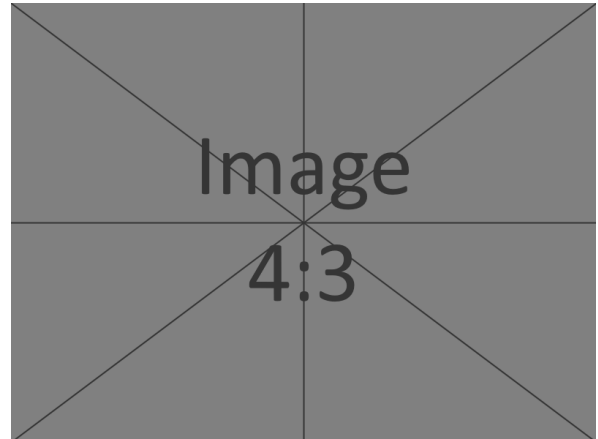
Zu beachten ist hier die Einstellung der Bildbreite von `subfigure`. Eine Zeile an Bildern sollte maximal zu 100% `\textwidth` aufaddiert werden können. Es ist aber meist schöner etwas mehr Platz zu lassen.

Des Weiteren sieht so eine Anordnung sehr viel besser aus, wenn die Bilder in der selben Zeile gleich groß sind. Man kann eine leichte Abweichung mit einem weißen Bildhintergrund kaschieren.

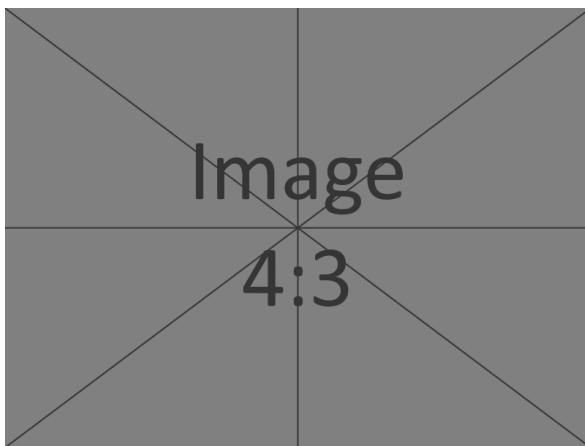
```
1 \begin{figure*}[p]
2   \centering
3   \begin{subfigure}[b]{0.475\textwidth} \centering
4     \includegraphics[width=\textwidth]{test.png}
5     \caption[Bild 1]{\small Bild 1}
6     \label{abb:bild1}
7   \end{subfigure}
8   \hfill
9   \begin{subfigure}[b]{0.475\textwidth} \centering
10     \includegraphics[width=\textwidth]{test.png}
11     \caption[Bild 2]{\small Bild 2}
12     \label{abb:bild2}
13   \end{subfigure}
14   \vskip\baselineskip
15   \begin{subfigure}[b]{0.475\textwidth} \centering
16     \includegraphics[width=\textwidth]{test.png}
17     \caption[Bild 3]{\small Bild 3}
18     \label{abb:bild3}
19   \end{subfigure}
20   \hfill
21   \begin{subfigure}[b]{0.475\textwidth} \centering
22     \includegraphics[width=\textwidth]{test.png}
23     \caption[Bild 4]{\small Bild 4}
24     \label{abb:bild4}
25   \end{subfigure}
26   \vskip\baselineskip
27   \begin{subfigure}[b]{\textwidth} \centering
28     \includegraphics[width=\textwidth]{testwide.png}
29     \caption[Bild 5]{\small Bild 5}
30     \label{abb:bild5}
31   \end{subfigure}
32   \caption[Bilder Matrix]{\small Bilder Matrix}
33   \label{fig:subfigbsp}
34 \end{figure*}
```



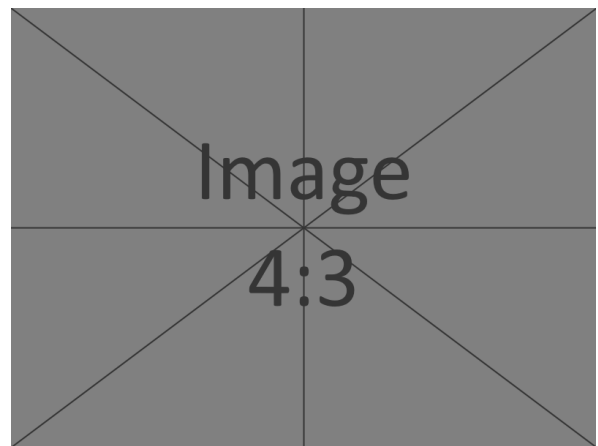
(a) Bild 1



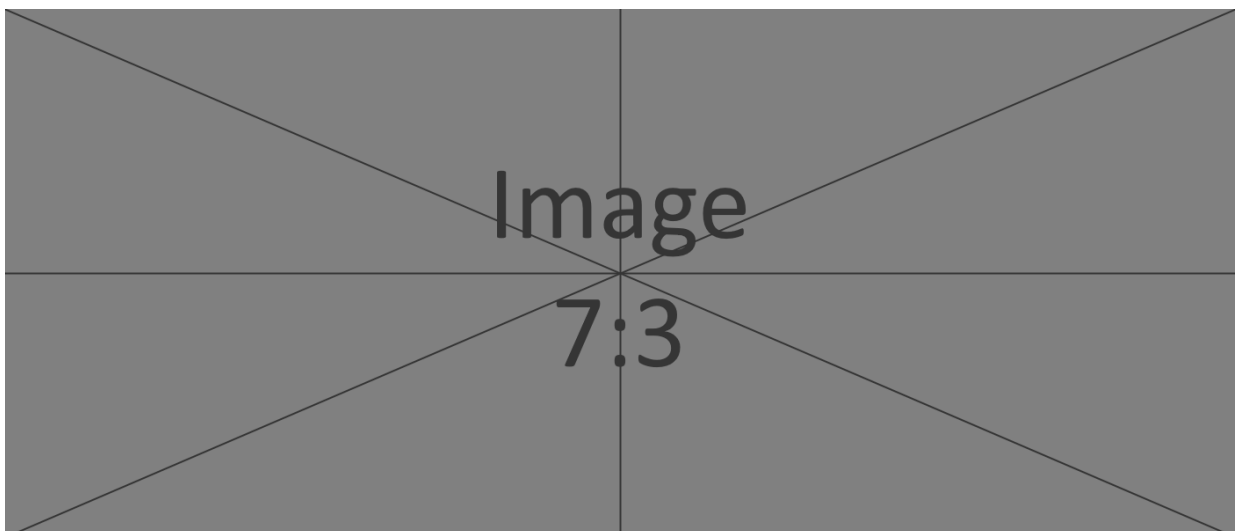
(b) Bild 2



(c) Bild 3



(d) Bild 4



(e) Bild 5

Abbildung 2.2: Bilder Matrix

2.9.5 Extra

Das Beispiel kommt von David Carlisle **David Carlisle**. Möchte man zwei Bilder unterschiedlicher Größe nebeneinander darstellen und dabei sicher gehen, dass keines der beiden Bilder verzerrt wird und am Ende beide gleich hoch sind und nur eine maximale Breite haben, kann einem dieses Beispiel helfen:

```

1  \begin{figure}[h]
2      % preliminary
3      \sbox\twosubbox{
4          \resizebox{\dimexpr.80\textwidth-1em}{!}{
5              \includegraphics[height=200pt]{test.png}
6              \includegraphics[height=200pt]{test2.png}
7          }
8      }
9      \setlength{\twosubht}{\ht\twosubbox}
10
11     % typeset
12     \centering
13     \subcaptionbox{$4:3$-Bild, Größe=800x600\label{abb:testimage_1}}{
14         \includegraphics[height=\twosubht]{test.png}%
15     }\quad
16     \subcaptionbox{$1:1$-Bild, Größe=400x400\label{abb:testimage_2}}{
17         \includegraphics[height=\twosubht]{test2.png}
18     }
19     \caption[Beide Abbildungen]{\small Beide Abbildungen}
20     \label{fig:both_testimages}
21 \end{figure}

```

In Zeile 4 kann der Faktor vor `\textwidth` angepasst werden und damit die gesamte Breite beider Bilder. Die Höhe wird dann automatisch errechnet und angepasst. Der Wert `height=200pt` in Zeile 5 und 6 kann angepasst werden, beeinflusst aber höchstens die Auflösung mit der die Bilder eingebunden werden. Es ist wichtig, dass dieser Wert bei beiden gleich ist!

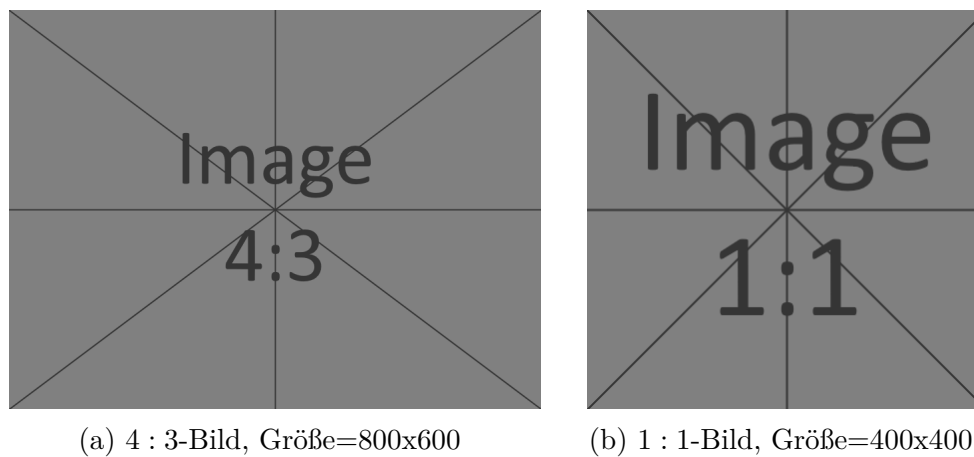


Abbildung 2.3: Beide Abbildungen

2.10 Tabellen

Tabellen werden in \LaTeX ähnlich wie Bilder gehandhabt. Man definiert die `table`-Umgebung als äußeren Wrapper für die Tabelle, ähnlich wie die `figure`-Umgebung bei Abbildungen. In dieser Umgebung erstellt man die `tabular`-Umgebung, in der man nun den Inhalt der einzelnen Zeilen und Spalten definiert.

```

1 \begin{table}[htb]
2   \centering
3   \caption{Tabellen Beispiel}
4   \label{tab:beispieltabelle}
5   \begin{tabular}{rrr} % 3 spalten rechtsbündig
6     \toprule
7     Zeit & $\sin$-Funktion & $\cos$-Funktion\\
8     \midrule
9     $0.0000$ & $0.00000$ & $1.0000$ \\
10    $1.0000$ & $0.84147$ & $0.54030$ \\
11    $2.0000$ & $0.90930$ & $-0.41615$ \\
12    $3.0000$ & $0.14112$ & $-0.98999$ \\
13    $4.0000$ & $-0.75680$ & $-0.65364$ \\
14    $5.0000$ & $-0.95892$ & $0.28366$ \\
15    $6.0000$ & $-0.27942$ & $0.96017$ \\
16    \bottomrule
17  \end{tabular}

```

18 `\end{table}`

2.10.1 table

Die `table`-Umgebung ist so ziemlich das gleiche, wie die `figure`-Umgebung für Abbildungen. Es kann wieder mit den Optionen (siehe Tabelle 2.6) die Position bestimmt werden. Einzig zu beachten ist, dass bei Tabellen die `caption` oberhalb der Tabelle steht.

2.10.2 tabular

Mit `tabular` wird die eigentlich Tabelle erzeugt. Der zweite Parameter konfiguriert dabei die einzelnen Spalten und die Trennstriche dazwischen.

Tabelle 2.8: Spaltenoptionen in einem anderen Design

Option	Beschreibung
<code>l</code>	Spalte mit linksbündigem Text
<code>c</code>	Spalte mit mittigem Text
<code>r</code>	Spalte mit rechtsbündigem Text
<code> </code>	Trennstrich zwischen der linken und rechten Spalte
<code>L{space}</code>	<code>space</code> -Breite Spalte mit linksbündigem Text
<code>C{space}</code>	<code>space</code> -Breite Spalte mit mittigem Text
<code>R{space}</code>	<code>space</code> -Breite Spalte mit rechtsbündigem Text

Der eigentliche Inhalt erfolgt dann Zeilenweise, die jeweils, mit Ausnahme der letzten, mit einem `\\` beendet wird. (Außer ihr fügt am Ende noch eine Endlinie hinzu, dann wird auch die letzte Zeile mit einem `\\` beendet.) Die einzelnen Spalten werden dabei mit dem `&` Zeichen getrennt. Es müssen in jeder Zeile alle Spalten angegeben werden, auch wenn diese leer bleiben.

Mit `\multirow` und `\multicolumn` können auch Zellen verbunden werden.

Für eine genaue Erklärung der Funktionsweise schauen Sie in die Dokumentation zu `multirow` und `multicolumn`.

```

1 \begin{tabular}{|l|c|r|}
2   \hline
3   1 & 2 & 3\\
4   \hline
5   4 & 5 & 6\\
6   \hline
7   7 & 8 & 9\\
8   \hline
9   & 0 & \\
10  \hline
11 \end{tabular}

```

Beispiel 1

1	2	3
4	5	6
7	8	9
	0	

```

1 \begin{tabular}{|c|c|c|}\hline
2   1 & 2 & 3\\
3   \hline
4   & \multicolumn{2}{c|}{5}\\
5   \cline{2-3}
6   \multirow{-2}{*}{4} & 6 & 7\\
7   \hline
8 \end{tabular}

```

Beispiel 2

1	2	3
4	5	
	6	7

2.10.3 Beispieltabelle

```

1 \begin{table}[h]
2   \centering
3   \caption{Beispieltabelle 1}
4   \label{tab:bsp1}
5   \begin{tabular}{@{}C{0.15\textwidth}L{0.75\textwidth}@{}}
6     \rowcolor[HTML]{000000}
7     {\color[HTML]{FFFFFF} \textbf{Nummer}} & {\color[HTML]{FFFFFF} \textbf{Beschreibung}}\\
8     \rowcolor[HTML]{FFFFFF}
9     \num{1} & Beschreibung 1\\
10    \rowcolor[HTML]{C0C0C0}
11    \num{2} & Beschreibung 2\\
12    \rowcolor[HTML]{FFFFFF}
13    \num{3} & Beschreibung 3\\
14    \rowcolor[HTML]{C0C0C0}
15    \num{4} & Beschreibung 4
16  \end{tabular}

```

17 `\end{table}`

Tabelle 2.9: Beispieltabelle 1

Nummer	Beschreibung
1	Beschreibung 1
2	Beschreibung 2
3	Beschreibung 3
4	Beschreibung 4

2.10.4 Tabellen Generator

Wer ein anderes Design für seine Tabellen haben möchte, kann sich ein eigenes Design mithilfe der Webseite <https://www.tablesgenerator.com/#> erstellen. Es sei aber Anmerkt, dass darauf geachtet werden sollte, dass die Arbeit in sich konsistent und nicht zu überladen ist, also nicht jede Tabelle ein anderes Design mit einer ausgiebigen Farbüberflutung hat.

2.11 Quelltext

Quellcode kann mit dem `lstlisting`-Paket eingebunden werden. Für C++, Matlab, VHDL und Latex-Code wird der Quelltext auch automatisch entsprechend dargestellt.

Quelltext kann auf zwei verschiedene arten eingebunden werden. Zum einen kann mit dem `\lstinputlisting`-Befehl direkt eine Quelldatei angegeben werden, zum anderen ist es auch möglich den Quelltext direkt in der `lstlisting`-Umgebung zu schreiben (Siehe Quellcode 2.2).

```
1 \lstinputlisting[style=matlab,caption={Beispiel}]{Quelltext/Beispiel.m}
```

```
1 %Beispiel zum Einfügen von MATLAB-Code
2 M(1,:)=[0:2*pi];
3 M(2,:)=sin(M(1,:));
```

Quellcode 2.1: Beispiel

```

1 \begin{lstlisting}[style=cpp,caption={Beispiel C++},label=lst:bsp]
2     ..Quelltext..
3 \end{lstlisting}

```

```

1 #include <iostream>
2 int main{
3     std::cout << "Hallo Welt!" << std::endl;
4     return 0;
5 }

```

Quellcode 2.2: Beispiel C++

Hier eine liste aller Styles die bereits in `Vorspann_Dipl.tex` definiert wurden.

Tabelle 2.10: Styles für die `lstlisting`-Umgebung

Kürzel	Sprache
Matlab	Matlab
cpp	C++
vhdl	VHDL
latex	Latex

2.12 Extras

2.12.1 Paket Dokumentationen

Um die Dokumentation eines installierten Pakets anzusehen ist der `texdoc`-Befehl hilfreich.

```
texdoc paketname
```

Bei Windows einfach in der Eingabeaufforderung oder im Ausführen Dialog eingeben und es sollte sich die jeweilige Dokumentation öffnen. Bei Linux und Mac sollte das im Terminal genauso funktionieren.

2.12.2 Kein Perl

Wenn Perl keine Option ist, kann alternativ versucht werden das **Makeindex**-Skript zu benutzen. Dazu geht man mit der Kommandozeile in den Ordner mit den *.tex*-Dateien und versucht es mit:

```
makeindex.exe -s %.ist -t %.alg -o %.acr %.acn  
makeindex.exe -s %.ist -o %.sym %.sbl
```

Achtung: % steht für den Namen der Hauptdatei! Sollte das nicht funktionieren bemühen Sie die Suchmaschine ihrer Wahl und schauen Sie sich die Dokumentation des **glossaries**-Pakets an.

A Schaltpläne

B Abbildungsverzeichnis

1.1	TeXstudio konfigurieren	2
2.1	Bild 0	14
2.2	Bilder Matrix	17
2.3	Beide Abbildungen	19

C Tabellenverzeichnis

2.1	Tabelle gängiger Präfixe für Referenzen.	5
2.2	Beschreibung des newacronym-Befehls	6
2.3	Beschreibung des newglossaryentry-Befehls für das Formelverzeichnis . . .	7
2.4	Beschreibung Referenzbefehle	7
2.5	Beschreibung der neuen Befehle	10
2.6	Optionen der figure-Umgebung.	13
2.7	Nützliche Befehle für subfigure.	15
2.8	Spaltenoptionen in einem anderen Design	20
2.9	Beispieltabelle 1	22
2.10	Styles für die lstlisting-Umgebung	23

D Quellcodeverzeichnis

2.1	Beispiel	22
2.2	Beispiel C++	23