VRIJE UNIVERSITEIT BRUSSEL

Master thesis submitted in order to be awarded the degree of
Master of Science in Engineering Technology:
Electromechanical Engineering - sustainable energy

TITLE OF YOUR WORK
optional subtitle

Your name

20xx-20xx

promoter: prof. …
supervisor: …

ENGINEERING SCIENCES

BRUFACE
BRUSSELS FACULTY
OF ENGINEERING

ULB

VUB

Title of your work

Optional subtitle

Your name

Master thesis submitted under the supervision of
prof. . . .

In order to be awarded the Master's degree in
Electromechanical Engineering: Energy

ECOLE
POLYTECHNIQUE
DE BRUXELLES

ULB

Première ligne de titre du mémoire
Deuxième ligne de titre du mémoire
Ligne du sous-titre du mémoire

Mémoire présenté en vue de l'obtention du diplôme
d'Ingénieur Civil [...] à finalité [...]

[Prénom Nom]

Directeur
Professeur [Prénom Nom]

Co-Promoteur
Professeur [Prénom Nom]

Superviseur
[Prénom Nom]

Service
[Nom du service]

Année académique
20xx - 20xx

For a substantial undertaking such as a thesis, it is good practice to create a project plan and to maintain a logbook. For students enrolled in the Energy Technology program [Dutch: opleiding Industriële Wetenschappen], the creation of a project plan for a bachelor's or master's thesis is mandatory.

A template for a Gantt chart (a common format for a project plan) and a logbook is provided via the link below. Please visit this link and create your own online project plan according the instructions outlined in the `read me` worksheet. Afterwards, substitute the URL in the `\href` command at the bottom of this page, with the URL of your own online project plan.

Please keep your project plan page accessible for your supervisors during your project, but exclude it from the final version of your manuscript. You can accomplish this by commenting out `\include{extra/projectplan}` before compiling the final version of your manuscript.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line `\hideExplanations` in your main file.

My project plan and logbook can be consulted online via this link.

Here you can thank certain people, companies and/or organisations.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line `\hideExplanations` in your main file.

Your name

Place name, September 3, 2024

An abstract is a concise summary of your manuscript and must make sense by itself. It is probably the first part that a reader reads. It must describe the work and set accurate expectations for the reader. It is usually around 200 to 300 words and provides the following information: the context and reason for writing, the problem statement and scope, the used method(s) and/or performed experiments, the obtained results and the conclusions and implications.

An abstract in English is required for a master thesis in Engineering Technology and BRUFACE.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line \hideExplanations in your main file.

This is a template to make a thesis and/or slides in LaTeX, free software to compile high-quality scientific documents. The template contains the typical parts such as a title page (for VUB, BRUFACE and ULB), acknowledgements, abstract, table of contents, a nomenclature, the typical chapters, appendices and a bibliography. In the appendices you will find practical information about formatting and examples of useful packages.

This template is too large to compile with a free Overleaf plan. Consult section A.1.4 in appendix A for possible workarounds.

A set of videos is available that demonstrate the use of this template.

This template comes with four main files:

– `mainReport`: to compile a report and all the appendices of this template

– `mainSlides`: to compile slides. Only selected content is put on the slides though. To see all content, you must compile `mainReport` and/or inspect the source code.

– `PhD`: to compile a PhD without the appendices of this template

– `Thesis`: to compile a thesis without the appendices of this template

The template uses the class memoir. It is a class for typesetting mathematical works as books, reports and articles.

This template is written in English. If you want to write in Dutch, you must select Dutch as the spell checking language in Overleaf via the Menu-button in the upper left corner and replace `english` by `dutch` as option in the `\documentclass` command.

keywords: LaTeX, template, Ti*k*Z

An abstract in Dutch is required for a master thesis in Engineering Technology.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line \hideExplanations in your main file.

trefwoorden: LaTeX, template, TikZ

An abstract in French is required for a master thesis in Engineering Technology.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line `\hideExplanations` in your main file.

mots clés: LaTeX, template, Ti*k*Z

Use this section to document the AI tools you used. Check your program regulations to verify if such a disclosure is obligatory. If necessary, you can report by completing the table provided below. Dedicate a row for each tool or service used. Note that the use of a spellchecker and grammar checker doesn't need to be included in this report.

A Dutch version is available in the source code of this page, but is currently commented out.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line `\hideExplanations` in your main file.

Declaration

During the preparation of this work, the author(s) used the following tool(s) and service(s):

name of tool/service		reason
name of tool/service	in order to:	*reason*
name of tool/service	in order to:	*reason*
. . .		

After using this tool(s) and/or service(s), the author(s) reviewed and edited the content as needed. The author(s) take(s) full responsibility for the content of this work.

Contents

Listings

It is good practice to include a list of symbols in your report/thesis in a so-called nomenclature. Below you will find an example. In your own copy of the template, you must update the nomenclature to bring it in line with your report/thesis. This means that you have to:

1. *remove entries that are not used in your manuscript*

2. *update entries, in case you use other symbols*

3. *add missing entries that are used in your manuscript*

You can find the nomenclature in `extra/nomenclature`.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line `\hideExplanations` in your main file.

ACRONYMS

APU	Auxiliary Power Unit
BAT	Best Available Technology
BDC	Bottom Dead Centre
CC	Combined Cycle
CHP	Combined Heat and Power
COP	Coefficient of Performance
CS	Control Surface
CV	Controle Volume
FOD	Foreign Object Damage
GWP	Greenhouse Warming Potential
HE	Heat Engine
HHV	Higher Heating Value
HX	Heat Exchanger
IGV	Inlet Guide Vanes
ISA	International Standard Atmosphere
LHV	Lower Heating Value
ODP	Ozone Depletion Potential
RH	Relative Humidity
RPM	Rounds Per Minute
TDC	Top Dead Centre
TIT	Turbine Inlet Temperature

DIMENSIONLESS NUMBERS

Ma	Mach number	$\mathrm{Ma} = V/V_{\mathrm{sound}}$

Nu	Nusselt number	$\mathrm{Nu} = hL/\lambda$
Pr	Prandl number	$\mathrm{Pr} = \nu/\alpha = \mu c_p/\lambda$
Re	Reynolds number	$\mathrm{Re} = \rho V L/\mu$

GREEK SYMBOLS

α	thermal diffusivity	m²/s
β	mass transfer coefficient	m/s
δ	infinitesimal quantity	
ϵ	emissivity	
ϵ	strain	
η	efficiency	%
γ	heat capacity ratio	
λ	thermal conduction coefficient	W/(m K)
μ	dynamic viscosity	Pa s
π	pressure ratio	
ν	kinematic viscosity	m²/s
ρ	mass density	kg/m³
ω	solid angle	sr
ω	angular velocity	rad/s

ROMAN SYMBOLS

A	area	m²
B	magnetic flux density	T
c	specific heat capacity	J/(kg K)
D	diameter	m
D	mass diffusion coefficient	m²/s
D	thermal dilation coefficient	
E	electromotive force	V
E	electric field strength	V/m
E	Young's modulus	N/m²
E	total energy	J
f	frequency	Hz
F	force	N
g	acceleration of gravity	m/s²
H	enthalpy	J
$\dot{H}$	enthalpy flow rate	J/s
h	specific enthalpy	J/kg
h	convection coefficient	W/(m² K)
I	electric current	A
L	length	m
M	molecular mass	kg/mol
m	mass	kg
$\dot{m}$	mass flow rate	kg/s
n	number of moles	mol
N	rotational speed	rpm
p	pressure	Pa
p	pole-pair number	
P	power	W
Q	heat	J

$\dot{Q}$	thermal power	W
q	specific heat	J/kg
R	electric resistance	Ω
R_u	universal gasconstant	J/(mol K)
R_s	specific gasconstant	J/(kg K)
r	compression ratio	
S	entropy	J/K
s	specific entropy	J/(kg K)
T	torque	Nm
T	thrust	N
T	temperature	K
U	voltage	V
U	internal energy	J
$\mathcal{V}$	volume	m^3
V	velocity	m/s
v	specific volume	m^3/kg
W	work	J
$\dot{W}$	mechanical power	W
w	specific work	J/kg
x	vapour fraction	
Y	mole fraction	
z	height	m
Z	compressibility factor	

SUPERSCRIPTS

$'$	per meter	1/m
$\cdot$	flow rate	1/s
$-$	per mole	1/mol
$*$	critical	
isen	isentropic	
o	at 25 °C	

SUBSCRIPTS

a(tm)	atmospheric
a	air
bb	blackbody
cr	critical point
elek	electric
f	fuel
g	generated
i	isentropic
0	dead state
p	at constant pressure
p(rim)	primary
pm	prime-mover
R	reduced
s	specific
sat	saturated
s(ec)	secundary

st	stoichiometric
t	total
th	theoretical
tot	total
u	universal
univ	universe
v	at constant volume

A thesis typically contains an introduction. A possible structure is provided below.

– *Context: situate your work within a broader background and explain the relevance of the problem your work addresses*

– *Topic: define the topic and scope of your work*

– *Research question: list the research question(s) that your work aims to answer; a master thesis and a PhD must have research questions*

– *Objectives: list the objectives of your work*

– *Thesis structure: briefly describe the thesis structure*

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line **\hideExplanations** *in your main file.*

1.1 CONTEXT

1.2 TOPIC AND SCOPE

1.3 RESEARCH QUESTION

1.4 OBJECTIVES

It is recommended to use action verbs to formulate objectives. It will make them actionable and easier to assess for progress. Examples of action verbs are: to analyze, assess, build, construct, create, demonstrate, design, develop, evaluate, enhance, implement, improve, increase, investigate, measure, optimize, produce, reduce, test, validate.

Try to formulate SMART objectives that are specific, measurable, achievable, relevant, and time-bound. This approach will help make your objectives clearer, easier to track, realistic, aligned with your research question and defined within a specific time frame for completion.

Example of a SMART objective: "To measure the PM emissions from a multiheat wood pellet boiler with an ELPI+ during 24 hours of continuous operation at 100% load in December 2025."

The non-SMART version: "Study boiler emissions"

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line **\hideExplanations** *in your main file.*

1.5 THESIS STRUCTURE

Literature review

It is common practice to have a literature review in a thesis where the state-of-the-art and the relevant literature are discussed. But it must not necessarily be done in a separate chapter. It can also be a section in another chapter.

In the appendix about useful packages, you will find more info about `biblatex`, a package used for managing and formatting citations and bibliographies. Please also consult the guidelines for using other people's work, in the appendix on formatting.

Feel free to alter the name of the chapter and/or the sections.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line `\hideExplanations` in your main file.

Materials and Methods

It is mandatory to explain what materials, experimental setups and methods were used.

The guidelines for formatting figures, tables and mathematical expressions can be found in the appendix on formatting.

Feel free to alter the name of the chapter and/or the sections.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line \hideExplanations in your main file.

Results

It is mandatory to report the obtained results from modelling work, calculations, simulations and experiments.

The guidelines for formatting figures, tables and mathematical expressions can be found in the appendix on formatting.

Feel free to alter the name of the chapter and/or the sections.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line \hideExplanations in your main file.

Discussion

It is mandatory to discuss the obtained results. But it must not necessarily be done in a separate chapter. It can also be part of the chapter with results.

Feel free to alter the name of the chapter and/or the sections.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line `\hideExplanations` in your main file.

Conclusions

This chapter is mandatory. Your work must have conclusions.

The conclusions contain your main findings and the answer to the research question(s) in a clear and concise way. Conclusions should not contain results that have not yet been discussed in your work.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line \hideExplanations in your main file.

Future work

It is common practice to formulate future work in a thesis. But it must not necessarily be done in a separate chapter. It can also be a section in the chapter with conclusions.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line `\hideExplanations` in your main file.

This appendix contains a very short description of LaTeX and explains the content and use of this template.

A.1   About LaTeX

LaTeX is software to make high-quality technical and scientific documents. It is not a "what you see is what you get" (WYSIWYG) editor such as Microsoft Word or OpenOffice. It is a typesetting engine, which means that it works more like a programming language.

LaTeX requires an input file that contains the source code. The structure of that file is shown in listing A.1 and is discussed in more detail here.

Listing A.1: Structure of a LaTeX file

```
\documentclass[⟨options⟩]{⟨class name⟩}
   % This part is the PREAMBLE, here:
   %  packages are loaded that enhance the capabilities of LaTeX
   %  settings are defined
\begin{document}
   % This part is the DOCUMENT, here:
   %  content is provided in plain text format
\end{document}
```

A.1.1   Document class

The input file starts with the command `\documentclass[⟨options⟩]{⟨class name⟩}`, where ⟨*options*⟩ is a placeholder for a list of options, such as for instance font size, paper size, etc. that can be given to customize the behaviour of the document class and ⟨*class name*⟩ is a placeholder for the name of a class file, recognised by the extension `.cls`. In a class file the layout standard is defined which tells LaTeX how to format your content.

This template uses the class file memoir but many other class files exist such as article, report, book,.... A comprehensive list of class files can be consulted at CTAN.

A.1.2   Preamble

The next part in the input file is called the *preamble*. In this part, packages are loaded that enhance the capabilities of LaTeX. This is typically done in so-called style files with the extension `.sty`. The style files that are loaded in this template are found in `styles/myStyleFiles`. There are three style files that load packages:

– `styles/myBibSettings`: to load the biblatex package to specify settings related to citing and the bibliography

– `styles/myListings`: to load the listings package to specify settings related to formatting computer code

– `styles/myPackages`: to specify settings for other packages

The purpose and use of some of the loaded packages are discussed further in this chapter. Currently there exist around 4000 packages for LaTeX for all kind of applications. The comprehensive list of all packages can be consulted at CTAN.

In the *preamble* a user can also define his/her own commands and environments. In this template, this is done in the other style files in the folder `styles`.

A.1.3 ENTERING CONTENT

The actual content of your work is enclosed between the commands `\begin{document}` and `\end{document}`. When you have a lot of content, you will typically divide it over several files. Files with content have the extension `.tex`. Content is added as plain text, combined with embedded LaTeX commands for specific typeset results. To include files in your input file, you can use the command: `\include{⟨filename⟩}` or `\input{⟨filename⟩}`.

The command `\include{⟨filename⟩}` is used to include chapters. It will first process all the floats (figures and tables) from the previous chapter, cause a page break so that the next chapter can start on a new page and import the contents from the file named ⟨*filename*⟩ into the target file. It will also start a new auxiliary file in the background to keep track of the locations of figures, tables and equations within the chapter for proper cross-referencing. Inspect the file `mainReport` to see what chapters are included in this template.

The command `\input{⟨filename⟩}` only imports the contents from the file named ⟨*filename*⟩ into the target file. It's equivalent to typing all the content from the file ⟨*filename*⟩ into the target file at the location of the `\input{⟨filename⟩}` command.

The content is typically structured for the reader by using commands such as:
`\chapter{⟨name of chapter⟩}`
`\section{⟨name of section⟩}`
`\subsection{⟨name of subsection⟩}`
`\subsubsection{⟨name of subsubsection⟩}`

More info on how to add content such as text, images, tables and math can be found in Learn LaTeX in 30 minutes.

A.1.4 COMPILING YOUR INPUT FILE

You have to compile you input file to obtain a typeset PDF document. The processing of your input file is done by a so-called TeX engine. It will parse your input file, line by line, detect the embedded LaTeX commands in your input file and act upon them, according the settings in your class file and packages. This means that you only have to concentrate on the content, while LaTeX will take care of the formatting.

There are four possible outcomes of a compilation:

Successful compilation This is the desired outcome. The compilation process completed without warning or errors. The output PDF corresponds to the specified content.

Compilation with warnings The compilation process completed but there were minor issues in your code. The feedback button in Overleaf will colour orange and show the number of

warnings. If there are 3 warnings for instance, you will see `3`. Click on the box to get more info about the warning. Warnings can be caused by duplicate labels, text extending beyond margins, etc. The output PDF might have issues with formatting, cross-referencing or citing. You should review your code and address the issues that gave rise to the warnings. This might not be an immediate necessity, but it's certainly important to do so before finalizing your document.

Compilation with errors The compilation was halted because serious issues were encountered that could not be resolved. The feedback button in Overleaf will colour red and show the number of errors. If there are 8 errors for instance, you will see `8`. Click on the box to get more info about the errors. Errors can be caused by syntax errors, missing files, incorrect commands, etc. You must immediately address these errors because the produced output PDF can not be trusted or will even be missing. Good to know: an error can induce a cascade of other errors. So don't be alarmed by the potential abundance of reported errors. Always start by addressing the first listed error. Once rectified, you might find that other errors also vanish. See the help pages of Overleaf for more info on compilation errors.

Compile time out warning In 2023, Overleaf reduced the compile time for free plan users from 1 minute to 20 seconds. That is not enough to compile this entire template. Possible workarounds are:

– Comment out some of the large appendices in the template and compile again. This will reduce the required compilation time.
– Or compile the file `myThesis` instead. This file contains the full structure of a thesis but without the appendices, so it takes less time to compile.
– Transfer the ownership of your project to your supervisor or promoter. If he/she has a premium account on Overleaf, enough compilation time will be available to compile the entire template.
– Download the template and compile it offline.
– Get a paid Overleaf subscription.

A.2   ABOUT THIS TEMPLATE

A.2.1   CONTENT OF THE TEMPLATE

This template will get you started with writing your thesis, report or slides in LaTeX. The template contains:

– a title page for VUB, BRUFACE and ULB
– a project plan
– a jury (for a PhD)
– acknowledgements
– an abstract in English, Dutch and French
– a disclosure on the use of AI
– a table of contents
– a list of figures, tables and listings
– a nomenclature
– the typical chapters
– appendices with useful information on various topics
– your own publications (for a PhD)
– a bibliography
– a demo bibliography with common entry types
– a to-do list

A.2.2 CONTENT OF THE APPENDICES

In the appendices you will find useful information about

    – LaTeX and this template
    – formatting of figures, tables, equations and source code
    – useful LaTeX packages and commands
    – examples of different types of figures and plots
    – mathematical and Greek symbols
    – example of a table in appendix
    – example of an included PDF file

The template uses the class memoir. It is a class for typesetting mathematical works as books, reports and articles.

A.2.3 HOW TO START USING IT

To use this template: surf to this template on the Overleaf website and click on the button 'Open as Template'. This will create a project with the name *Tips for writing (in LaTeX)*. You can then change the name of the project according to your choice. FYI: the name *Tips for writing (in LaTeX)* was inherited from the precursor to this template, a document with tips on formatting numbers, units, and figures, hence the name.

A set of videos is available that demonstrate the use of this template.

This template can generate a report and slides from the same source code. It comes with three main files:

    – `mainReport`: to compile a report and all the appendices of this template

    – `mainSlides`: to compile slides. Only selected content is put on the slides though. To see all content, you must compile `mainReport` and/or inspect the source code.

    – `PhD`: to compile a PhD but without the appendices of this template

    – `Thesis`: to compile a thesis but without the appendices of this template

A.2.4 HOW TO CHANGE THE LANGUAGE

This template is written in English. If you write in another language, you have to adapt the language in two ways (example below is for Dutch):

1. Select `Dutch` as spell check language in Overleaf, via the `Menu` in the upper left corner. Words that do not belong to the selected spell check language will be underlined in red in the editor. The selected spell checking language has no effect on the output PDF.

2. Replace the option `english` by `dutch` in the command `\documentclass`. Options supplied to `\documentclass` are global options and are available to all packages.

The option `dutch` will be transferred to the following two packages, loaded in `styles/myPackages`:

    – `\usepackage{babel}`
    This will automatically translate standard names:
    chapter → hoofdstuk
    contents → inhoudsopgave

bibliography → bibliografie

. . .

– `\usepackage{cleveref}`
This will automatically translate cross-referencing names:
figure → figuur
table → tabel
equation → vergelijking
. . .

The option `dutch` will also cause the nomenclature to be displayed in Dutch. If you want to know how this works, inspect the first lines of the file `extra/nomenclature`.

Your report or thesis will most probably contain elements such as figures, tables, numbers, units, symbols, mathematical expressions, citations and a bibliography. This chapter contains some guidelines and good practices for the proper formatting of these elements.

B.1 FIGURES

Figures are visual representations such as drawings, schemes, photos, charts and graphs. Every figure must be referenced in your text, preferably before the figure appears. Some important aspects of a figure are shown in fig. B.1 and are discussed below.

First of all, your figure must be of good quality, which means that careful consideration must be given to graphical excellence and integrity (e.g. resolution, data-ink ratio, lie-factor, legend,...). See the excellent work of Tufte (2001) and Doumont (2009) for more information on these topics.

Secondly, your figure must have an unique identifier and a descriptive caption. The identifier is typically the word *Figure* followed by a number. For a short report, these numbers are typically the counting numbers 1, 2, 3, etc. For a thesis or book with different chapters, the numbering of figures restarts at 1 at the beginning of each new chapter, but the chapter number is included in the figure number. See for example the number 4.7 in fig. B.1, which means that this is the seventh figure in chapter 4. The automatic assigning of figure numbers in LaTeX is taken care of by the `\caption` command. If you want the caption to appear under the figure, you must place the `\caption` command on the last line in the figure environment. You can inspect the source code of fig. B.1 to verify this.

A descriptive caption is a stand-alone text that must provide enough information to understand the figure, without consulting the text. The first word of the caption should be capitalized. The caption should contain a *what* part and, if possible, a *so-what* part. The *what* part clearly

a numbered identifier under the figure, referenced in the text

a good-quality figure

Eppler 625 airfoil

Figure 4.7: A plot of an Eppler 625 airfoil, to illustrate the plotting of data points from a file, source: UIUC Department of Aerospace Engineering, 2020

a caption describing: the *what*   the *so-what*   the source

Figure B.1: Some important formatting aspects of a figure and its caption
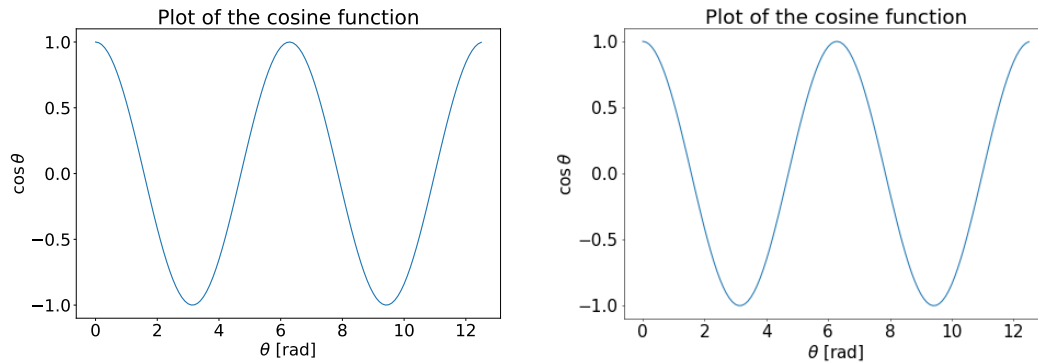
Figure B.2: Plot of a cosine function, included from a file in vector format (PDF, left plot) and a file in bitmap format (PNG, right plot), to illustrate the superiority in image quality of PDF over PNG

explains what is shown on the figure. Try to precise and complete, it's better to have a longer caption than a short incomplete or unclear caption. The *so-what* part is a statement and explains what the reader should pay attention to and/or contains a justification for showing the picture. For figures that are shown on a slide (with the beamer class), the *so-what* message should be put as title of the slide, as explained in appendix B.7.

If you did not make the figure yourself, you must add the reference to the source of the figure at the end of your caption, preceded for instance by the word `source:`. If the figure came from an external source but was adapted by you, you can precede the reference to the source by `adapted from:`.

B.1.1 Include graphical files in vector format

To enhance the quality and scalability of graphics in documents, you should include graphical files in vector format like PDF whenever possible, rather than in bitmap format like PNG. Vector graphics maintain their sharpness and clarity at any size or resolution, making them ideal for high-quality printing and zooming in on details. In Python for instance, plots can be saved in PDF format with the command `plt.savefig(`‘$\langle filename\rangle$`.pdf’)`.

Zoom in on fig. B.2 and observe the difference in quality of the two plots.

B.2 Tables

Tables are used to present data in an orderly manner. They can contain different kind of elements such as text, numbers and mathematical symbols and expressions. Every table must be referenced in your text, preferably before it appears. Some important aspects of a table are shown in table B.1 and are discussed below.

First of all, your table must be of good quality and should provide clarity on the presented data. This means that careful consideration must be given to graphical excellence and integrity (e.g. readability, data formatting and alignment, data-ink ratio, legend,...).

Secondly, your table must have an unique identifier and a descriptive caption. The identifier is typically the word `Table` followed by a number. For a short report, these are typically ordinal numbers such as 1, 2, 3, etc. For a thesis or book with different chapters, the numbering of tables restarts at 1 at the beginning of each new chapter, but the chapter number is included in the table number. Table 3.2 for example refers to the second table in chapter 3. The automatic assigning of table numbers in LaTeX is taken care of by the `\caption` command. If you want the caption to appear above the table, you must place it on the first line in the figure environment.

Table B.1: Some important formatting aspects of a table and its caption

a numbered identifier above the table, referenced in the text

a caption describing: the *what*   the *so-what*

Table 5.5: Technical specifications of the T100 CHP from Turbec, these values were used for the design calculations, source: Turbec, 2001

a top border —

singular column headers

some space

a good-quality table —

a bottom border —

description	symbol	value	unit	comment
turbine inlet temperature	TIT	950	°C	calculated
compressor pressure	p_2	6.5	bar	
mass flow rate air	$\dot{m}_{air}$	1.79	kg/s	
thermal power natural gas	$\dot{Q}_{fuel}$	330	kW	
net electrical power	$\dot{W}_{net}$	100	kW	
net thermal power	$\dot{Q}_{net}$	167	kW	
CHP efficiency	η	80.9	%	at full load

the source

text = left align   number = right align   text = left align

You can inspect the source code of table B.1 to verify this. In Microsoft Word, captions should be added via the `insert caption` menu command.

A descriptive caption is a stand-alone text that must provide enough information to understand the data in the table, without the need to consult the text. The first word of the caption should be capitalised. The caption should contain a *what* part and, if possible, a 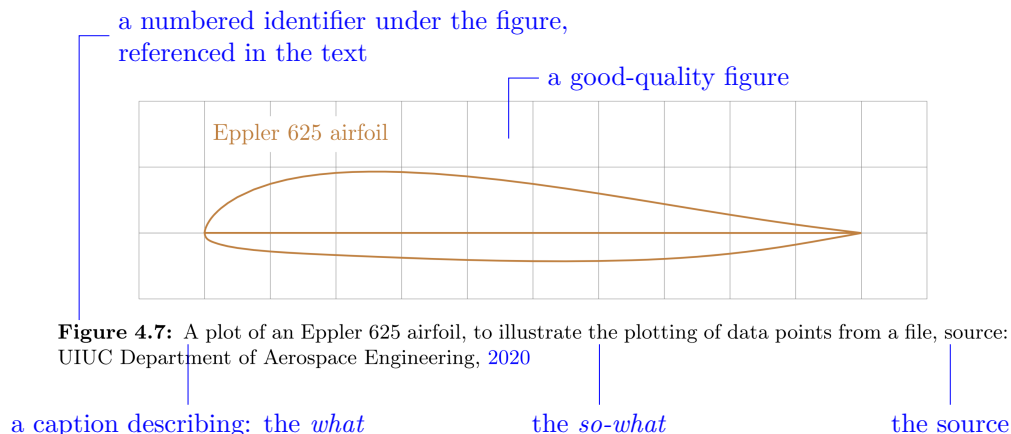*so-what* part. The *what* part clearly explains what is shown in the table. Try to precise and complete, it's better to have a longer caption than a short incomplete or unclear caption. The *so-what* part is a statement and explains what the reader should pay attention to and/or contains a justification for showing the table.

If you did not obtain the data in the table yourself, you must add the reference to the source of the data at the end of your caption, preceded for instance by the word `source:`

If the table came from an external source but was adapted by you, you can precede the reference to the source by `adapted from:`

B.2.1 HEADER ROW

A simple table will typically have a header row with a descriptor per column. Descriptors are not capitalized, unless it is required by grammar rules. Descriptors should also be written in the singular form. See for example the descriptors in table B.6. You will see: parameter, value, unit, comment, and not: Parameter or Parameters or parameters,...

Tables can also have multiple header rows, for example a row for the descriptors and a row for the units. More complex tables can also have header columns.

B.2.2 COLUMN ALIGNMENT

The data in the columns should be aligned. Columns that contain text should be left-aligned (column type `l`). That makes the text easier to read, because we read from left to right. See for example the improved readability of the text in the first column of table B.3 compared to table B.2. Columns that contain numbers should be right-aligned (column type `r`) or aligned at the decimal marker (column type `S`). This makes it easier to compare numbers in the same column as illustrated in the second column in tables B.2 and B.3. Columns with symbols and

Table B.2: Example of a table with centered columns, which are not well suited for text and numbers

text	number	unit
relative humidity outside	62	%
mass flow rate of fresh air	20 000	kg/h
total pressure at the intake of the HVAC system	1	bar

Table B.3: Improved table with a left-aligned text column and a right-aligned number column

text	number	unit
relative humidity outside	62	%
mass flow rate of fresh air	20 000	kg/h
total pressure at the intake of the HVAC system	1	bar

Table B.4: A table with wrapped text in the first column to reduce the column width

text	number	unit
relative humidity outside	62	%
mass flow rate of fresh air	20 000	kg/h
total pressure at the intake of the HVAC system	1	bar

columns with units can be centred (column type `c`), but they are easier to read when they are left-aligned.

Columns with a lot of text can become too wide, causing the table to become too wide. This can be solved by wrapping the text in the column. You can achieve this by limiting the width of a column to a specified value with the paragraph column type `p`, see example in table B.4.

B.2.3   Horizontal and vertical lines

Having too many horizontal and/or vertical lines in a table is considered as visual noise and is distracting to the eye. It also lowers the data-ink ratio. Most of the time, two horizontal lines will do, one at the top of the table (`\toprule` command) and another at the bottom (`\bottomrule` command) as in table B.1. These lines visually separate the data in the table from the main text. The column headers are separated from the data by some extra space. This separation can also be achieved with a horizontal line (`\midrule` command). Vertical lines are most of the time not required but can be added in some occasions to improve readability.

Compare for instance table B.5 and table B.6. Both tables contain exactly the same data but table B.6 has the better data-ink ratio and is therefore recommended.

B.3   Symbols

Table B.5: Example of a table with unnecessary horizontal and vertical lines, giving rise to a low data-ink ratio

parameter	value	unit	comment
voltage	12	V	given
current	2	A	given
power	24	W	calculated

Table B.6: Improved table with a top and a bottom line only, giving rise to an improved data-ink ratio

parameter	value	unit	comment
voltage	12	V	given
current	2	A	given
power	24	W	calculated

B.3.1 USE OF SYMBOLS

Symbols are typically used to represent *(physical) properties*. Physical properties play an important role in physics and engineering. Well-known physical properties are for example the mass, temperature, velocity, pressure and electric current, but there are many others.

Physical properties can be scalars (e.g. the mass m and temperature T) but they can also be vectors (e.g. the force $\vec{F}$ and the velocity $\vec{V}$), matrices or tensors.

The value of a scalar physical properties is expressed with a number and an associated unit. One can say, for example: the ambient temperature is 298 kelvin. Note that both the name of the physical property (ambient temperature) and the name of the unit (kelvin) were written in full. That is perfectly acceptable but not so practical to use in calculations and formulas. That is why we often switch to a compact notation with *symbols*.

For example, one can rewrite the statement regarding the ambient temperature as follows: the ambient temperature $T_{\text{environment}}$ is 298 kelvin. Note that we used the symbol T for the temperature. This is merely an agreement and other symbols for the temperature are also possible. For example, some books use the symbol θ for the temperature. It is therefore recommended to always consult the symbol list or nomenclature of a scientific publication to find out which symbols were used.

Useful information about the physical quantity can be mentioned in the subscript; in this example the fact that it concerns the temperature of the environment). So $_{\text{environment}}$ was added as a subscript to the symbol T.

There are also symbols for units. For example, the symbol for the kelvin is the uppercase letter K. The above statement concerning the ambient temperature can be rewritten as follows: the ambient temperature $T_{\text{environment}} = 298\,\text{K}$.

B.3.2 FORMATTING OF SYMBOLS

One-letter symbols that represent (physical) quantities – such as for example the temperature T and the voltage U – are written in italics. Everything else is written upright, as illustrated in the flowchart in fig. B.3. Symbols that consist of more than one letter – such as for instance the Mach number (Ma) or the Net Present Values (NPV) – are written upright, otherwise they might be mistaken as products. That's why for instance the Reynolds number is written as Re and not as Re, because that could be interpreted as the product of the radius R with the eccentricity e.

The fact that many symbols must be written upright will give you some kind of struggle, because in the math mode of LaTeX letters are written in italics by default, unless you manually force them upright. The same is true for the Equation editor of Microsoft Word, there too, the standard way of writing is in italics. But no worries, how to deal with this and how to get things right is explained below.

Some examples are given below and in table B.7. So, as already mentioned, only one-letter
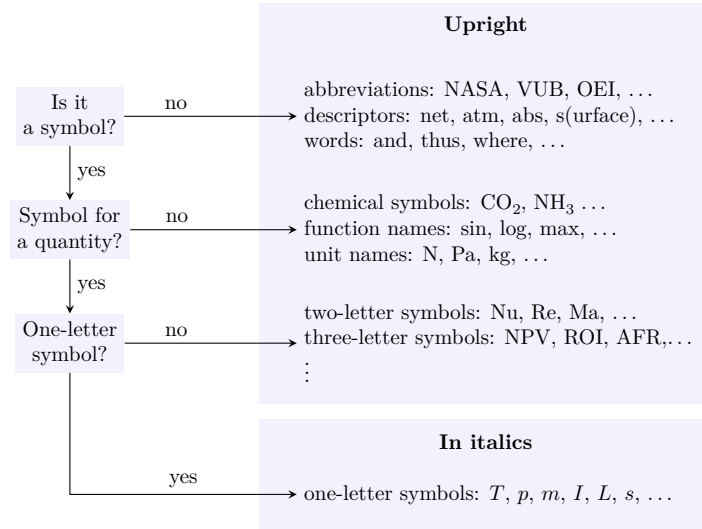
Figure B.3: Flowchart to determine if something must be written upright or in italics: only one-letter symbols for quantities must be written in italics

Table B.7: Some things are always written upright

type	example	LaTeX example	LaTeX result
descriptions	abs, net, …	`$p_\up{atm}$`	p_{atm}
symbols for units	Pa, kg, …	`\unit{\square\m}`	m^2
math functions	log, max, …	`$\sin\alpha$`	$\sin\alpha$
chemical formulas	NH_3, CO_2, …	`\ch{CH4}`	CH_4
multi-letter symbols	Ma, Nu, …	`$\up{Re}=2300$`	$\mathrm{Re} = 2300$
acronyms and abbreviations	NPV, NASA, …	`$\up{AFR}= 50$`	$\mathrm{AFR} = 50$

symbols for quantities are written in italics:

$$\text{correct:} \quad T = 298\,\mathrm{K}$$
$$\text{not correct:} \quad \mathrm{T} = 298\,\mathrm{K}$$

The same goes for one-letter symbols that appear in subscripts and superscripts. The subscript p for example in the specific heat capacity c_p is written in italics because it represents a quantity:

$$\text{correct:} \quad c_p \quad \text{(because } p \text{ is a symbol for a quantity)}$$
$$\text{not correct:} \quad c_{\mathrm{p}}$$
$$\text{correct:} \quad p_{\mathrm{gauge}} = p_{\mathrm{abs}} - p_{\mathrm{atm}} \quad \text{(because the subscripts are not symbols for quantities)}$$
$$\text{not correct:} \quad p_{gauge} = p_{abs} - p_{atm}$$

Natural constants are also quantities and their one-letter symbols are therefore also written in italics, as for example, the universal gas constant[1] R_{u} or the speed of light in vacuum c.

---

[1] the u in the subscript is written upright because it is not a symbol for a quantity, it's just the first letter of the word 'universal'

More-letter symbols for quantities are written upright:

  correct: $\mathrm{Ma} = 0.80$
  not correct: $Ma = 0.80$ (undesired spacing + looks like M times a)
  correct: $\mathrm{NPV} = 3000\,\text{€}$
  not correct: $NPV = 3000\,\text{€}$ (undesired spacing + looks like N times P times V)

Descriptions are written upright:

  correct: $T_{\text{environment}} = 298\,\mathrm{K}$
  not correct: $T_{environment} = 298\,\mathrm{K}$

Even if you shorten the descriptive text to make the notation more compact, you still write it upright:

  correct: $T_{\text{env}} = 298\,\mathrm{K}$
  not correct: $T_{env} = 298\,\mathrm{K}$

Even descriptions reduced to a single letter are written upright:

  correct: $T_{\text{e}} = 298\,\mathrm{K}$ (because e is not a symbol for a quantity)
  not correct: $T_{e} = 298\,\mathrm{K}$

Words are written upright, even when they appear between mathematical expressions:

  correct: $U = IR$ with the current $I = 2\,\mathrm{A}$
  not correct: $U = IR$ *and the current* $I = 2\,\mathrm{A}$

Words are written upright, even when they appear in mathematical expressions:

  correct: $\eta = \dfrac{\text{work produced}}{\text{heat received}}$

  not correct: $\eta = \dfrac{work\ produced}{heat\ received}$

Symbols for mathematical functions are written upright:

  correct: $\sin \alpha$
  not correct: $sin\alpha$

Chemical formulas are written upright:

  correct: $\mathrm{CO_2}$
  not correct: CO_2

B.4   NUMBERS

B.4.1   DECIMAL MARKER

We advice you to use the dot as the decimal marker for your entire document, as is common practice in English scientific literature (e.g. $\pi \approx 3.14$).

In Belgium it is customary to use the comma as the decimal marker (e.g. $\pi \approx 3{,}14$), but this will give rise to inconsistencies, as you will probably also have figures and tables from English-language literature (that contain numbers with the dot as decimal marker). Maintaining consistency in the selected decimal marker throughout your document is however essential to avoid confusing the reader. Also, the comma serves other purposes in the math mode in LaTeX, such as for instance the separation of function arguments. For these reasons, we recommend to use the dot as the decimal marker.

B.4.2 THOUSANDS SEPARATOR

It is good practise to separate the thousands in a number with a small non-breakable space. This makes a number with many digits easier to read. The siunitx package can take care of this, because it provides the command `\num`, see appendix C.16 for more info.

Observe for instance how the readability of the number 10 million can be improved by using the command `\num{10000000}`

hard to read:	10000000
easier to read:	10 000 000

Tip: do not use the comma as a thousands separator, to avoid confusion for people in Belgium who are used to the comma as a decimal marker.

B.4.3 SCIENTIFIC NOTATION

The scientific notation is used to write very large or very small numbers in a more compact form. It is also used to report numbers with the appropriate level of precision without overwhelming the reader, as illustrated in the Reynolds number example in the next subsection. Here is an example of a correctly formatted number in scientific notation: 6.02×10^{23}. It consists of:

- a coefficient, here: 6.02

- a multiplication sign, preceded and followed by a non-breaking space: $\times$

- a power of 10, here: 10^{23}

B.4.4 SIGNIFICANT DIGITS

Calculations performed with a calculator or computer code typically produce numeric results with many digits. But beware, it's important to understand that not all of these digits are significant. The number of significant digits depends on the uncertainty of the used input data for the calculations and on the accuracy and validity of the model(s) used in the calculations. When you include such numerical results in your report, you should only write the significant digits, being the digits that are meaningful and reliable. Writing more digits creates a false impression of precision and accuracy.

In engineering, as a rule of thumb, we typically consider 3 significant digits. But exception are of course possible. With high-precision instruments, we can perform measurements that leverage more than 3 significant digits. For instance a voltage of 239.41 V with 5 significant digits. And with low-precision instruments, we get less significant digits, for instance a temperature of 19 °C with only 2 significant digits. In such cases, you must of course adapt the reported number of digits accordingly.

So how to not-report the non-significant digits? Well, there are three simple methods:

1. Simply not write them, but this only works for digits behind the decimal marker

2. Switch the prefix of the unit, for instance from kWh to MWh

3. Switch to exponential notation

You will find an example of each method below.

Imagine that you calculated a flow velocity and obtained a value of $2.385\,494\,828\,187\,\mathrm{m/s}$ with your calculator or computer code. Well, do not report this result will all its digits in your manuscript! Most probably, only the first 3 non-zero digits are significant. Hiding the other digits is easy, because they are located behind the decimal marker.

> not correct: $V = 2.385\,494\,828\,187\,\mathrm{m/s}$ (too many digits, false impression of accuracy)
> correct: $V = 2.39\,\mathrm{m/s}$ (assuming 3 significant digits)

Imagine that you calculated the yearly heat demand of a space, based on outside temperature data (with 3 or 2 significant digits) and that your computer code produced $23\,765.324\,\mathrm{kWh/year}$ as a result. Again, do not report this result will all its digits in your manuscript. Only the first 3 or 2 non-zero digits are significant. Hiding digits after the decimal marker is easy, but not sufficient. More digits must be hidden. We can achieve this be switching the prefix in the unit.

> not correct: $Q = 23\,765.324\,\mathrm{kWh/year}$ (too many digits, false impression of accuracy)
> not correct: $Q = 23\,765\,\mathrm{kWh/year}$ (still too many digits, false impression of accuracy)
> correct: $Q = 23.8\,\mathrm{MWh/year}$ (assuming 3 significant digits)
> correct: $Q = 24\,\mathrm{MWh/year}$ (assuming 2 significant digits)

Imagine that you calculated the Reynolds number, based on measured flow velocity data (with 2 significant digits) and that your computer code produced $\mathrm{Re} = 567\,214$ as a result. You can guess, do not report this result will all its digits in your manuscript. Only the first 2 non-zero digits are significant. We can achieve this be switching to exponential notation. Switching the prefix in the unit is not an option here, because the Reynolds number has no units.

> not correct: $\mathrm{Re} = 567\,214$ (too many digits, false impression of accuracy)
> correct: $\mathrm{Re} = 5.7 \times 10^5$ (assuming 2 significant digits)

B.4.5 UNCERTAINTY

The uncertainty of a number can be put in parentheses behind the number (Newell and Tiesinga, 2019, section 5.4.5). Consider for example this length: $1.236(5)\,\mathrm{m}$. The number in parentheses refers to the last digit of the quoted value 1.236, so in this example, the uncertainty is $0.005\,\mathrm{m}$ or $5\,\mathrm{mm}$. An alternative notation is $1.236 \pm 0.005\,\mathrm{m}$.

B.5 UNITS

Units must be formatted according to the International System of Units (SI) guidelines, see the rules and style conventions in section 5 of Newell and Tiesinga, 2019. Except for dimensionless quantities, numbers without units make no sense. Units are always written upright, never in italics:

> correct: $L = 7\,\mathrm{meter}$
> not correct: $L = 7\,meter$

Unit names are never capitalized unless they are at the beginning of a sentence. This is to distinguish them from the scientists they are named after. So joule is the SI unit for energy, while Joule is the name of the English physicist and mathematician James Prescott Joule. You will find examples of other unit names in Tables 2 and 4 in Newell and Tiesinga, 2019.

correct:	300 joule (or watt, newton, pascal, volt, bar, kelvin, . . .)
not correct:	300 Joule (or Watt, Newton, Pascal, Volt, Bar, Kelvin. . .)

Special attention goes to the degree Celsius. The first word is not capitalized since the names of units are not capitalized. The second word is capitalized as it refers to the Swedish astronomer Anders Celsius. Another unit for temperature that is often used is the kelvin, written with a lowercase k, to distinguish it from the name of the British mathematician and engineer William Thomson Kelvin, also known as Lord Kelvin. It is important to note that there is no such thing as a degree Kelvin, just as there is no degree kilogram or degree meter. See the examples below.

correct:	0 degree Celsius (small d + capital C)
not correct:	0 Celsius (the word degree is missing)
not correct:	0 Degree Celsius
not correct:	0 degree celsius
correct:	273 kelvin
not correct:	273 Kelvin (that is the name of Lord Kelvin)
not correct:	273 degree kelvin (there is no such thing as a degree kelvin)
not correct:	273 degree Kelvin (there is no such thing as a degree Kelvin)

Symbols for units, like the letter m for meter, are always written upright:

correct:	$L = 7\,\mathrm{m}$
not correct:	$L = 7\,m$

In this way there can be no confusion between the symbol m that is used for mass and the symbol m that is used for meter. And so, for example, A represents an area in m^2 and the A written upright represents the unit ampere for electric current.

Do not use abbreviations for units. Use the symbol or the full name:

correct:	s or second
not correct:	sec
correct:	$\dfrac{\mathrm{m}}{\mathrm{s}}$ or meter per second
not correct:	mps, m/sec

The symbol used for a unit is never written in the plural form:

correct:	$z = 120\,\mathrm{cm}$
not correct:	$z = 120$ cms

No full stop is put after a unit, unless that unit is at the end of a sentence:

correct:	The length of the tube is 70 cm.
correct:	The tube is 70 cm long.
not correct:	The tube is 70 cm. long.

Table B.8: Some standard prefixes in SI units

factor	prefix	symbol	factor	prefix	symbol
10^{-1}	deci	d	10^{1}	deca	da
10^{-2}	centi	c	10^{2}	hecto	h
10^{-3}	milli	m	10^{3}	kilo	k
10^{-6}	micro	μ	10^{6}	mega	M
10^{-9}	nano	n	10^{9}	giga	G
10^{-12}	pico	p	10^{12}	terra	T
10^{-15}	femto	f	10^{15}	peta	P
10^{-18}	atto	a	10^{18}	exa	E
10^{-21}	zepto	z	10^{21}	zetta	Z

Prefixes representing a multiplication factor (table B.8) are also written upright and written together with the symbol of the unit

$$\text{correct:} \quad p_1 = 700\,\text{kPa}$$
$$\text{not correct:} \quad p_1 = 700\,\text{k}\,\text{Pa}$$

B.5.1 Spaces

A number must be kept together with its unit. This can be achieved by putting an unbreakable space between the number and the unit. This will 'glue' the unit to the number, so that the number doesn't end up alone at the end of a line with the unit at the start of the next line.

To achieve this in LaTeX you use the tilde `~` in text mode and the `\;` in math mode. So for instance `80~kg` will appear as 80 kg. Or even better, you can use the package `siunitx` that will take care of the proper spacing for you, amongst many other things related to numbers and units (more info in appendix C.16).

In Microsoft Word you must type `shift-ctrl-space` to produce an unbreakable space.

The unit can also be 'glued' to the number by suppressing the space between them, but this is considered as bad practice.

$$\text{correct:} \quad \mathcal{V} = 3\,\text{m}^3$$
$$\text{bad practice:} \quad \mathcal{V} = 3\text{m}^3$$

B.6 Mathematical expressions

B.6.1 Equations

It is good practice to number your equations, because it allows to cross-reference them from elsewhere in your text. Automatic numbering can be achieved by using the `equation` environment, as shown in the first law of thermodynamics for closed systems below

$$\Delta E = Q + W. \tag{B.1}$$

To establish a cross-reference, you must first add a label to the equation environment. You can then use this label in a `\cref` command elsewhere. Inspect the source code of eq. (B.1) to see how this is done.

There are many other math-environments available in LaTeX that can do more complicated things with equations such as splitting, grouping and aligning. More info is available on the Overleaf help page about aligning equations with amsmath.

B.6.2 MULTIPLICATION SIGN

To write a multiplication of two variables, leave a small space between them or put a proper multiplication operator between them. In math mode in LaTeX, use `\cdot` to produce the center dot · or use `\times` to produce the multiplication sign ×.

In Microsoft Word, you can use `\bullet` in the equation editor to produce the center dot ·.

Other symbols, such as the asterisk *, the letter x and the full stop, should be avoided.

$$
\begin{aligned}
\text{correct:} \quad & U = I\,R \\
\text{correct:} \quad & U = I \cdot R \\
\text{acceptable:} \quad & U = I \times R \quad \text{might be confused with vector product} \\
\text{bad practice:} \quad & U = I * R \\
\text{bad practice:} \quad & U = IxR \\
\text{bad practice:} \quad & U = I.R
\end{aligned}
$$

B.6.3 PUNCTUATION

Equations are part of the sentence structure, which means punctuation must be respected. See the example below.

Newton's second law is

$$
\vec{F} = m \cdot \vec{a}, \tag{B.2}
$$

where $\vec{F}$ is the force, m the mass and $\vec{a}$ the acceleration.

B.7 SLIDES

You can also make slides with this template. To do so, select the file `mainSlides` as your main document via the Menu-button in the upper left corner in Overleaf and compile it. You will see that the first line of `mainSlides` loads the class `beamer` which is a powerful LaTeX class to create presentations. More info about this class can be found on the Overleaf help pages on Beamer.

In beamer, slides are called frames. When you compile `mainSlides`, LaTeX will process the content that is embedded in all the frame environments and put that content on slides. Each frame environment becomes a separate slide. The syntax of a frame environment is shown in listing B.1.

Listing B.1: Syntax of a frame environment

```
\begin{frame}[⟨options⟩]{⟨title of your slide⟩}
    %  content is provided in plain text format
\end{frame}
```

There are some things that are good to know, when making slides. They are discussed by Jean-Luc Doumont in a talk he gave at Stanford University in 2013. The video is still available on YouTube and is worth watching. Some of his recommendations are listed below. If you inspect the source code, you will see that they are embedded in a frame environment. This means that besides being visible in this report (compiled with the class `memoir`), they will also end up on a slide (when compiled with the class `beamer`).

A slide should:
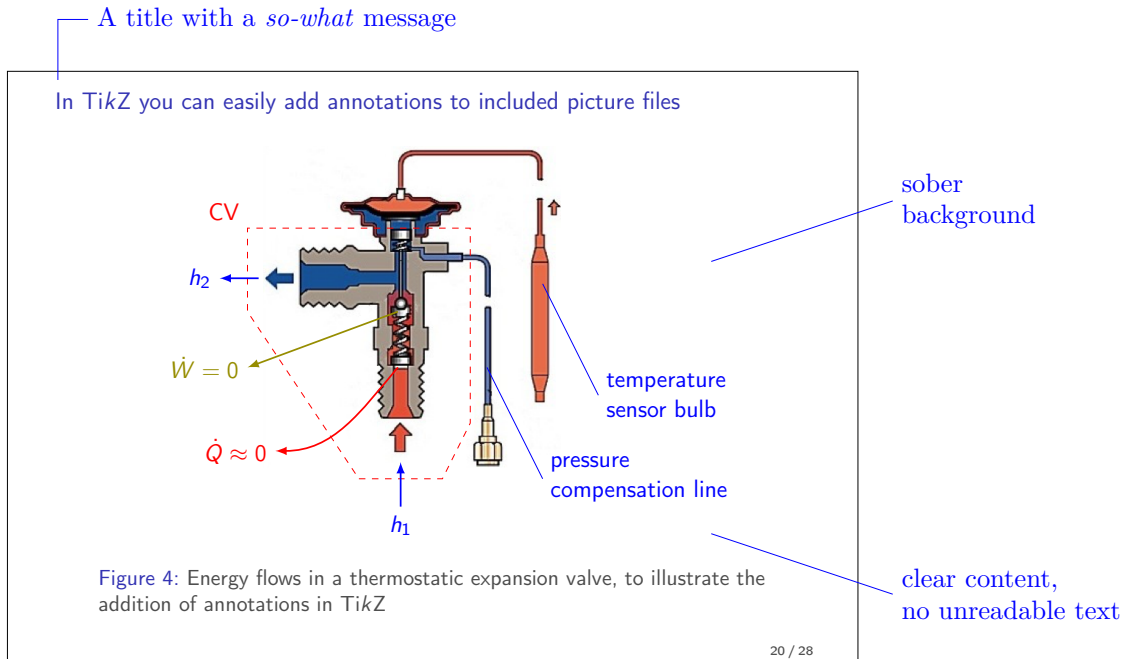
Figure B.4: An example of a slide with a *so-what* message in the title, a sober background and some clear content with no unreadable text

- have a clear title

- have a sober background

- contain clear content

- have no unreadable text

A title should be:

- a *so-what* message, telling the reason that you are showing the slide, rather than just a *what* message about the content of the slide

- left-aligned, because that is easier to read

- written as a full sentence, so with a verb, to get your message across

- two lines maximum, more lines will probably not be read

You can see an example in fig. B.4.

B.7.1 MAKING CONTENT FOR SLIDES ONLY

It is also possible to create content that will only show up on a slide, but not in your report. You can achieve this by specifying the mode `<presentation>` in the frame environment. In the source code beneath this sentence, a frame environment is specified that is invisible in this report, as you can not see :-) But it will show up on a slide, when compiled with the class `beamer`.

Table B.9: Example of a table generated in Python

parameter	value	unit
inputs:		
nominal power	39	kW
electricity produced	90.4	MWh/year
calculated:		
capacity factor	26.5	%
equiv. full load hours	2318	h

B.7.2 CREATING CONTENT IN PYTHON

It is possible to generate content for your report and/or slides using a programming language such as Python. For instance, table B.9 is entirely generated within Python with the script `scripts/capacityfactor.py` that creates the file `scripts/capacityfactor.tex`. The values presented in this table result from calculations performed in the script. By combining calculations and reporting in a single script, the need to manually input static results into your report is eliminated. The same script also generates content for a slide. Compile `mainSlides` to see this slide. Inspect the script `scripts/capacityfactor.py` and the accompanying output file `scripts/capacityfactor.tex` for more info.

FYI: with a paid Overleaf subscription you can synchronise Overleaf with Dropbox, allowing seamless two-way syncing. All your Overleaf projects will appear in your Dropbox, and vice versa. This feature allows you to semi-automate the inclusion of output generated with Python into your final PDF document, requiring only one **execute** and one **recompile**. The procedure is as follows:

1. Place your Python script in the corresponding Dropbox folder of your Overleaf project, ideally in a dedicated `scripts` folder.

2. **Execute** the Python script and save the generated tex file in the same folder as your script. The tex file will automatically sync with Overleaf within a few seconds.

3. In Overleaf, add the following lines of code:
   `\mode<all>`
   `\input{`⟨*name of generated tex file*⟩`}`
   `\mode*`

4. **Recompile** your project in Overleaf to incorporate the Python-generated content into your PDF.

B.7.3 SOME EXAMPLE SLIDES

Compile `mainSlides` to see examples of slide templates.

B.8 USING OTHER PEOPLE'S WORK

In your report or thesis you will probably use information from other sources. If you do so, you have to explain to the reader where this information came from and where this information can be found. This is of great importance for the following three reasons:

1. it gives proper credit to the authors or creators of your sources

2. it allows the reader of your work to look-up and consult the source to learn more about a subject

3. it avoids that you are being suspected of plagiarism

B.8.1 CITING

Citing is the crediting of author(s) of a source, directly in your text. This is done by putting in-text citations in the body of your text. They are typically placed immediately after the information being cited. In LaTeX we use the package BibLaTeX to take care of the citations, see appendix C.2 for more info. There are three commands you should know when citing.

The `\cite` command is used to refer to a work itself, as part of the sentence. For example:

The visual display of quantitative information is discussed in Tufte, 2001.

The `\textcite` command is used to refer to the author(s) of a work, as part of the sentence.

Tufte (2001) discusses how quantitative information is visually displayed.

The `\parencite` command is used to cite a work when the sentence is already complete. In this case, the citation is placed in parentheses at the end of the sentence, but before the full stop. If you remove the parentheses and their content, the sentence should still make grammatical sense.

Quantitative information is visually displayed in different ways (Tufte, 2001).

A citation must always be placed inside a sentence, never behind the full stop.

correct: This is a sentence [1].
bad practice: This is a sentence. [1]

There are many different citation styles. Table C.2 contains information on three popular citation styles (author-year, numeric and alphabetic) and shows the effect on the three cite commands that were mentioned above.

B.8.2 SOURCES

You should use reliable sources, such as articles from scientific journals. Wikipedia for instance, can certainly be useful as inspiration, and help you understand certain things better, but can't be regarded as primary scientific source material. It's better to consult and use the underlying sources that Wikipedia is citing. So in principle, no reference to Wikipedia should appear in your reference list. But of course exceptions are possible.

B.8.3 BIBLIOGRAPHY

The bibliography is a list with all the cited sources in your work. It is typically placed at the very end of your document. It must contain the necessary information to find back the references. Typically: the names of the authors, the title of the work, the publication date, details of the publisher and the name of the journal.

In this template, the command `\printbibliography` at the end of the main file produces the bibliography. There are many different styles for a bibliography but we advise you to use one of the standard styles. As was already the case with the citations, the package BibLaTeX will take care of the bibliography. See appendix C.2 for more info on this package.

Table B.10: Some examples of words in British and American spelling; in the EU, British English is recommended

British	American
analyse	analyze
aerofoil	airfoil
aeroplane	airplane
aluminium	aluminum
behaviour	behavior
catalogue	catalog
centre	center
cheque	check
colour	color
defence	defense
fibre	fiber
grey	gray
initialise	initialize
labelled	labeled
licence	license
litre	liter
manoeuvre	maneuver
metre	meter
modelling	modeling
modelled	modeled
mould	mold
neighbour	neighbor
optimise	optimize
organisation	organization
parameterise	parameterize
pressurise	pressurize
programme	program
realise	realize
standardise	standardize
tonne	ton
tyre	tire
vapour	vapor

B.9 ENGLISH

In the European Union, British English is mostly used. American English can also be used, if there are reasons to do so. In any case, you should consistently use one type of spelling throughout your entire document, so American and British spelling cannot be mixed. Some examples of the differences between the British and American spelling are shown in table B.10.

This chapter contains an alphabetic list of some useful packages. All the packages that are mentioned here are loaded in this template.

C.1 BEAMERARTICLE

This package is essential when you want to create both a presentation and a corresponding report or article from the same source file. When you compile such a dual-purpose document using the `beamer` class, as in the first line of `mainSlides`, all content within frame environments will be placed on slides, resulting in a PDF for a presentation. For more information on creating slides in LaTeX, refer to appendix B.7.

However, if you compile the same document using the `memoir` class to create a PDF of a report, as in the first line of `mainReport`, the frame environments become irrelevant and are not recognized as valid commands by `memoir`. The `beamerarticle` package addresses this issue by ensuring that commands specific to the `beamer` class are ignored. This is why `beamerarticle` is the first package loaded in `MainReport`. So do not remove it.

More info at CTAN: package beamer.

C.2 BIBLATEX

This package is used for citing and formatting the bibliography. It is a complete implementation of the bibliographic facilities provided by LaTeX. BibLaTeX provides several standard citations styles, as for example: `authoryear`, `numeric` and `alphabetic` amongst many other non-standard citation styles that are popular in different journals.

You can inspect the current options that are loaded for the BibLaTeX package in the style file `styles/myBibSettings`. Some common styles and sorting options are shown in table C.1.

Is is good practice to keep your bibliographic references in a separate file, a so-called bibliographic information file, recognised by the extension `.bib`. In this template, this file can be found here: `bib/myBibliography.bib`.

Table C.1: Some examples of citing styles and sorting options in BibLaTeX

style	sorting	\cite example	cite number	order in bibliography
authoryear	nyt	Rees, 2017		by name, year, title
authoryear	none	Rees, 2017		by cite order
numeric	nyt	[8]	as in bibliography	by name, year, title
numeric	none	[1]	as order in text	by cite order
alphabetic	nyt	[Ree17]		by name, year, title
alphabetic	none	[Ree17]		by cite order

Table C.2: Some examples of other cite commands

style:	current style	authoryear	numeric	alphabetic
\cite{Rees2017}	Rees, 2017	Rees, 2017	[1]	[Ree17]
\textcite{Rees2017}	Rees (2017)	Rees (2017)	Rees [1]	Rees [Ree17]
\parencite{Rees2017}	(Rees, 2017)	(Rees, 2017)	[1]	[Ree17]
\citeauthor{Rees2017}	Rees	Rees	Rees	Rees

This file is imported with the command `\addbibresource{bib/myBibliography.bib}` in the style file `styles/myBibSettings`.

It is convenient and recommended to use separate software to collect and manage your references. A well-known free tool to manage your references is Mendeley Reference Manager. It works well together with Mendeley Web Importer. This is a browser extension to import reference material that you are watching in your browser into your Mendeley database. Other well-known free tools are Zotero and JabRef, but there exist many others.

Every reference in the bib file must have a unique identifier, a so-called a `BibTeX key`. You can choose this key yourself. An often-used approach to construct this key is to combine the name of the first author and the year of publishing. You can then refer to a BibTeX key with commands such as `\cite`, `\textcite`, `\parencite`, `\citeauthor` or `\fullcite`.

See table C.2 for some examples of these cite commands. The second column in this table shows the results for the style that is currently loaded in BibLaTeX. `\cite` is used in sentences where the reference is part of the sentence, and the year is there to identify what is being cited. `\parencite` is used to put the citation between parentheses. It is used when the sentence is already complete, and the citation is being added in support. `\textcite` is used in sentences where the name of the author is part of the sentence, and the year is there to identify what is being cited. With `\citeauthor` only the name of the author is shown without a hyperlink to the reference in the bibliography.

Rees (2017) made a good overview of the available commands and options in this cheat sheet.

More info at CTAN: package BibLaTeX and at Overleaf: getting started with BibLaTeX.

C.3   CANCEL

With this package you can easily show which terms are canceled in equations. For instance, the command `$\cancel{x}$` will produce $\cancel{x}$ and `$\cancelto{0}{z}$` will produce $\cancelto{0}{z}$.

Both commands are also demonstrated in an equation below.

For example, for a stationary [Dutch: stilstaand] process:

$$\Delta E_{cv} = \Delta U_{cv} + \cancel{\Delta KE_{cv}} \quad + \quad \cancelto{0\,J}{\Delta PE_{cv}} \tag{C.1}$$

and the first law for a closed and stationary system simplifies to

$$\Delta E_{cv} = \Delta U_{cv} = Q + W \quad [\text{J}]. \tag{C.2}$$

More info at CTAN: package cancel.

Table C.3: Various attempts to write the chemical formula of water

| LaTeX command	result	remark
H20	H2O	not correct, 2 not in subscript
H_20	error	underscore not allowed outside math mode
H_20	H_2O	bad practice to write chemicals in italics
$\mathrm{H_20}$	H_2O	correct, but heavy syntax
\ch{H20}	H_2O	correct and easy syntax
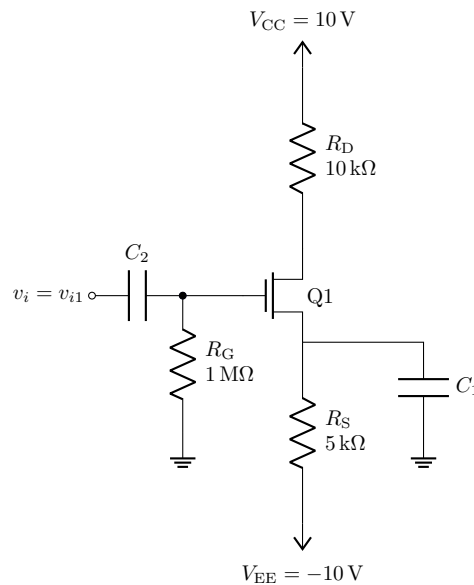
Figure C.1: Example of an electric circuit drawn with circuitikz (adapted from Redaelli et al., 2023, p. 25)
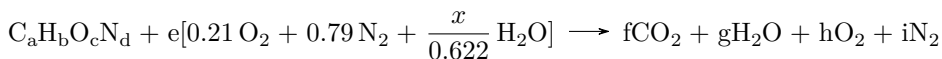
C.4 CHEMFORMULA

This package can typeset chemical formulas and reactions with an intuitive syntax. Consider for instance the formula H_2O for water. In table C.3 you will see some bad and good ways to write it in LaTeX.

Writing chemical reactions is also possible. For instance:

\ch{H20 + CO3^2- <=> OH- + HCO3-}   produces: $H_2O + CO_3^{2-} \rightleftharpoons OH^- + HCO_3^-$
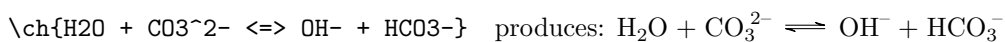
Or for instance this combustion equation:

$$C_aH_bO_cN_d + e[0.21\,O_2 + 0.79\,N_2 + \frac{x}{0.622}\,H_2O] \longrightarrow fCO_2 + gH_2O + hO_2 + iN_2$$

More info at CTAN: package chemformula.

C.5 CIRCUITIKZ

The package circuitikz extends the capabilities of the package TikZ. It allows users to draw electrical and electronic circuit diagrams in their LaTeX documents. It provides a wide range of components, symbols, and styles to create professional-looking circuit diagrams. An example is shown in fig. C.1. Inspect the source code to see how this electronic circuit diagram is defined.

Additional examples that are relevant for students in the Engineering technology program [Dutch: opleiding Industriële wetenschappen] can be found on pages 25–29 of the circuitikz user

Table C.4: Some examples of cross-referencing with `\cref` and `\Cref`

command	old syntax with `\ref`	result
`\cref{ch:conclusions}`	`chapter~\ref{ch:conclusions}`	chapter 6
`\cref{sec:siunitx}`	`section~\ref{sec:siunitx}`	appendix C.16
`\cref{fig:psat}`	`fig.~\ref{fig:psat}`	fig. E.16
`\cref{tab:mathsymbols}`	`table~\ref{tab:mathsymbols}`	table F.1
`\cref{eq:newton}`	`eq.~(\ref{eq:newton})`	eq. (B.2)
`\cref{list:factorial}`	`listing~(\ref{list:factorial})`	listing C.2
`\Cref{ch:conclusions}`	`Chapter~\ref{ch:conclusions}`	Chapter 6
`\Cref{sec:siunitx}`	`Section~\ref{sec:siunitx}`	Appendix C.16
`\Cref{fig:psat}`	`Figure~\ref{fig:psat}`	Figure E.16
`\Cref{tab:mathsymbols}`	`Table~\ref{tab:mathsymbols}`	Table F.1
`\Cref{eq:newton}`	`Equation~(\ref{eq:newton})`	Equation (B.2)
`\Cref{list:factorial}`	`Listing~(\ref{list:factorial})`	Listing C.2

manual (Redaelli et al., 2023). More info at CTAN: package circuitikz.

C.6 CLEVEREF

The cleveref package enhances LaTeX's cross-referencing features. It automatically formats cross-references depending on what they refer to (chapter, section, table, figure, equation,...). The normal procedure in LaTeX is to use the `\ref` command preceded by the type of object that is being referred to. With the `\cref` command, this is no longer necessary because the type of object will be automatically detected. This will make your cross-references more consistent. In table C.4 you will see the difference. For instance, in the case of figures, you don't have to explicitly write `fig.` in front of every reference to a figure. For references at the beginning of a sentence, that thus must be capitalized, the `\Cref` command must be used.

The format for each cross-reference type (chapter, section, figure, table and equation) can be customised in the style file `styles/myPackages`.

More info at CTAN: package cleveref.

C.7 CSQUOTES

Quotations must be appropriately formatted with quotation marks. The required quotation marks depend on the language of the document. With this package, you can focus on the content of the quote; the appropriate quotation style will automatically be employed based on the specified language option in `\documentclass`. The package can format both inline and display quotations. It provides several commands, such as `\enquote` for simple inline quotes, `\textquote` for inline quotes that also mention the author and the `\displayquote` environment that will separate the quote from the main text.

Here is an example of a quote from Rudolf Clausius that will appear inline in the text with quotations marks that are in agreement with the document language: "The energy of the universe is constant". The next inline quote also mentions the author: "Absence of evidence is not the evidence of absence" (William Wright, 1888). The example below is a quote that is separated from the main text.

> The best scientists are open to the possibility that they may be wrong, and they are willing to change their minds in the face of new evidence. (Richard Feynman)

More info at CTAN: package csquotes.

C.8   DRAFTWATERMARK

The package draftwatermark will put a watermark on the pages of your report when it is loaded with the `stamp` option as: `\usepackage[stamp]{draftwatermark}`. You can customize the watermark text to suit your needs. For example, you can use watermarks like 'draft', 'confidential', or 'under NDA'. Additionally, you can adjust the size, lightness, and angle of the watermark. See the example in the style file `styles/myPackages`.

In this template, no watermark is applied because the package is loaded with the `nostamp` option as: `\usepackage[nostamp]{draftwatermark}`. More info can be found at CTAN: package draftwatermark.

C.9   FONTAWESOME

The package fontawesome allows you to incorporate icons into your LATEX document. It provides a large number of web-related icons in the form of a font. Each icon has its unique command. They can be used in text mode, in math mode and in your drawings made with Ti*k*Z. Here are some examples: file , wifi , folder , GitHub , image , industry and lock . Check the source code to see the syntax. You can also customize their size and color, as you would do with other fonts. The same folder icon, but now in a larger font size and in blue looks like this:

The complete list of available icons can be consulted at CTAN: package fontawesome.

C.10   HYPERREF

With this package you can add hyperlinks in your document. For instance, the command: `\href{https://canvas.vub.be/}{Canvas learning platform}` will produce this clickable link: Canvas learning platform. More info can be found at CTAN: package hyperref.

C.11   LISTINGS

With the package listings you can typeset source code in your document. The proper typesetting of your code will make it easier to read. General info about typesetting source code in LATEX can be found in the Overleaf help pages. You will read that the package listings is well suited for this propose. The package provides the `lstlisting` environment as shown in the "Hello world!" example in listing C.1.

```
\begin{lstlisting}[language=Matlab,label={list:hello},
caption={Hello world in Matlab}]
    disp('Hello, World!');
\end{lstlisting}
```

As can be seen in the source code of this example, you can specify the language your code is written in, specify a caption and add a label for cross-referencing. The result is shown in listing C.1. Many programming languages can be typeset with this package. The names of the recognised languages can be consulted in Table 1 in the package documentation.

Listing C.1: Hello world in Matlab

```
1  disp('Hello world!');
```

The typesetting of code in this document is customised in the style file `styles/myListings`.

Another example, this time in Python, is shown in listing C.2. Many typesetting properties (such as: font, font size, indents, colours, . . .) can be specified. The package settings in this template can be consulted and adapted in the style file `styles/myListings`.

Listing C.2: Python code to calculate factorial, adapted from programiz

```python
1  # Factorial of a number via recursion
2  def factorial_recursive(n):
3      if n == 1:
4          return n
5      else:
6          return n*factorial_recursive(n-1)
7
8  num = int(input("Enter a number: "))
9  if num < 0:    # check if the number is negative
10     print("Sorry, factorial does not exist for negative numbers")
11 elif num == 0: # check if the number is zero
12     print("The factorial of 0 is 1")
13 else:
14     print("The factorial of", num, "is", factorial_recursive(num))
```

It is also possible to import source code from a file (that you maybe are still working on). Importing for example the first 18 lines from a Python script would look like this:

`\lstinputlisting[language=Python, linerange={1-18}]{⟨filename⟩.py}`.

More info at CTAN: package listings.

C.12 LONGTABLE

With this package you can make tables that are larger than one page and continue to the next page(s).

More info can be found at CTAN: package longtable.

C.13 PDFPAGES

The package `pdfpages` provides the command `\includepdf[⟨key=val⟩]{⟨filename⟩}` to insert pages from an external PDF document into your LaTeX document. ⟨*key=val*⟩ stands for a comma separated list of options using the ⟨*key*⟩=⟨*value*⟩ syntax. Consult the package documentation to learn about the available options. Here are some examples:
include page 3 and 7: `\includepdf[pages={3,7}]{⟨filename⟩}`
include page 1 to 4: `\includepdf[pages={1-4}]{⟨filename⟩}`
2 x 2 and add frames: `\includepdf[pages={1-4},nup=2x2,frame=true]{⟨filename⟩}`

See examples in appendix I. More info can be found at CTAN: package pdfpages.

C.14 PFDICONS

The package `pfdicons` provides drawings for elements used in process flow diagrams (PFDs) and instrumentation diagrams (PIDs), such as a valve, pump, compressor, turbine, heat exchanger, reactor, vessel and column. See example in appendix E.3. More info can be found at CTAN: package pfdicons.

Table C.5: Various attempts to correctly format a unit, illustrated with the unit for mass density, because it contains both a fraction and an exponent

LaTeX command	result	remark
`kg/m3`	kg/m3	not correct: 3 not in superscript
`kg/m^3`	error	ˆ not allowed outside math mode
`$kg/m^3$`	kg/m^3	bad practice to write units in italics
`$\mathrm{kg/m^3}$`	$\mathrm{kg/m^3}$	correct but heavy syntax, no horizontal bar
`$\mathrm{\frac{kg}{m^3}}$`	$\frac{\mathrm{kg}}{\mathrm{m^3}}$	correct but heavy syntax, font too small
`\unit{\kg\per\cubic\m}`	$\frac{\mathrm{kg}}{\mathrm{m^3}}$	correct and easy syntax, font size ok

C.15 PGFPLOTS

With this package you can make high-quality plots in normal or logarithmic scaling. They are typically drawn in the `axes` environment and plots are added with the command `\addplot`. See for instance fig. E.14 with some plots of analytical functions and fig. E.20 with a plot of external data from a file. See this gallery for many more examples.

More info can be found at CTAN: package pgfplots.

C.16 SIUNITX

This package correctly formats numbers and units, according to the rules and style convention from the International system of units (SI) (Newell and Tiesinga, 2019, section 5). The package can handle different types of numbers as well as complex units with ease, including units that contain fractions, exponents in superscript and/or descriptors in subscript.

To typeset units with superscripts and subscripts, without using macros provided by this package, you would typically need to switch to math mode. However, in math mode, everything is displayed in italics by default, unless specified otherwise. Since units always have to be written upright, you would have to employ the `\mathrm` command to ensure their upright appearance, which results in cumbersome syntax. This is illustrated in table C.5 where, as an example, various attempts are shown to properly format the unit of mass density. The final row in this table demonstrates the simplified syntax achievable by employing the `\unit` macro from this package.

Below you will find the most important macros the **siunitx** package provides. These macros can be used in text mode as well as in math mode.

C.16.1 \NUM

Besides the `\num` macro to format numbers, the package also provides macros to format lists of numbers, number ranges, products of numbers, angles and complex numbers. It can group digits for better readability, format numbers in scientific notation and numbers with uncertainties. See the examples in table C.6. In this table, the option `separate-uncertainty` and `range-phrase` have been set locally just for demonstration purposes. Normally, such options are set globally in the preamble, to ensure their uniform application throughout the entire document. More info on global options can be found in appendix C.16.7.

Table C.6: Examples of macros to format numbers

LaTeX command	result
\num{23000}	23 000
\num{4.5(1)}	4.5(1)
\num[separate-uncertainty=true]{4.5(1)}	4.5 ± 0.1
\num{6.02E23}	6.02×10^{23}
\numlist{50;70;90}	50, 70 and 90
\numrange{50}{70}	50 to 70
\numrange[range-phrase = --]{50}{70}	50–70
\numproduct{5 x 7 x 9}	$5 \times 7 \times 9$
\ang{50;7;33}	$50°7'33''$
\complexnum{3+-2i}	$3 \pm 2i$

Table C.7: Unit macro versus literal units

approach	example
unit name macro	\joule
unit abbreviation macro	\J
literal unit	J

C.16.2 \unit

Units can be formatted using the \unit macro. To obtain the symbol for a unit, there are three possible approaches, as illustrated in table C.7. If you choose to use unit macros (preceded by a backslash), the package will automatically typeset your units in accordance with the SI rules. If you prefer to use literal units, you will need to format them correctly yourself.

If you follow the instructions listed below, the \unit macro will typeset your unit in accordance with the SI rules and style conventions. Examples are shown in table C.8.

1. Write the units as you would pronounce them, for instance \joule\per\kilogram.

2. Use unit macros by preceding all unit names with a backslash, as in \joule or \kilogram. The package will automatically substitute the correct symbol for each unit. Alternatively, you can use unit abbreviations, such as \J or \kg, with the same output result[1].

3. Use \per to put units in the denominator.

4. Use \of to refer to descriptors.

5. Use \square for exponent 2, \cubic for exponent 3 and \tothe for other exponents.

6. Use SI prefixes such as \mega, \giga, . . .

C.16.3 PER-MODE

Units that are placed in the denominator with the \per macro can be displayed as reciprocal powers (the default) or as fraction or as symbol, depending om the setting of the per-mode

---

[1] Table 5 in the user manual of siunitx contains all available abbreviations

Table C.8: Examples of unit abbreviation macros and unit name macros

unit abbreviation macros	result
\unit{\celsius}	°C
\unit{\um}	µm
\unit{\kg\per\s}	$\frac{\text{kg}}{\text{s}}$
\unit{\MPa}	MPa
\unit{\kg\of{vap}\per\cubic\m\of{air}}	$\frac{\text{kg}_{\text{vap}}}{\text{m}^3_{\text{air}}}$
\unit{\W\per\square\m\per\K\tothe{4}}	$\frac{\text{W}}{\text{m}^2\,\text{K}^4}$
unit name macros	result
\unit{\degreeCelsius}	°C
\unit{\degree}	°
\unit{\micro\meter}	µm
\unit{\kilogram\per\second}	$\frac{\text{kg}}{\text{s}}$
\unit{\mega\pascal}	MPa
\unit{\kilogram\of{vap}\per\cubic\meter\of{air}}	$\frac{\text{kg}_{\text{vap}}}{\text{m}^3_{\text{air}}}$
\unit{\watt\per\square\meter\per\kelvin\tothe{4}}	$\frac{\text{W}}{\text{m}^2\,\text{K}^4}$

Table C.9: Three modes to display units in the denominator

per-mode	result	display method
\unit[per-mode=power]{\kJ\per\kg\per\K}	$\text{kJ}\,\text{kg}^{-1}\,\text{K}^{-1}$	reciprocal powers
\unit[per-mode=fraction]{\kJ\per\kg\per\K}	$\frac{\text{kJ}}{\text{kg}\,\text{K}}$	horizontal bar
\unit[per-mode=symbol]{\kJ\per\kg\per\K}	$\text{kJ}/(\text{kg}\,\text{K})$	forward slash

switch, as illustrated in table C.9. In the examples in this table, the `per-mode` switch is given as a local option for demonstration purposes. Normally, it is set globally in the preamble, as was done in the style file `styles/myPackages` where the package siunitx is loaded. Such a global setting makes more sense because it will give consistent formatting of units throughout your document.

It is also straightforward to modify the formatting of your units within your manuscript at a later time by adjusting this switch, eliminating the necessity to modify the source code of every unit throughout the document.

It is even possible to configure the `per-mode` switch separately for units written in display mode (such as stand alone equations) using `display-per-mode` and for units written in inline text using `inline-per-mode`.

The `per-mode` switch only functions in conjunction with unit macros; it has no impact on literal units. Refer to table C.10 for examples of literal units. As you can observe in this table, it is your responsibility to format literal units correctly to prevent ambiguity.

Table C.10: Examples of literal units

example	result	remark
`\unit{whatev3r}`	whatev3r	string appears as written
`\unit{N.m}`	N m	dots are converted to inter-unit products
`\unit{wha^tev_3r}`	wha^{t}ev$_3$er	superscripts and subscripts can be specified
`\unit{kJ/kgK}`	kJ/kgK	space between kg and K missing
`\unit{kJ/kg.K}`	kJ/kg K	ambiguous use of dot in denominator
`\unit{kJ/(kg.K)}`	kJ/(kg K)	literal unit correctly specified

Table C.11: Examples of the use of `\qty` macros

macro	result
`\qty{9.81}{\m\per\s\squared}`	$9.81\,\frac{\text{m}}{\text{s}^2}$
`\qty{9.81}{m/s^2}`	$9.81\,\text{m/s}^2$
`\qty{4}{\euro\per\kg}`	$4\,\text{€/kg}$
`\qty{2E-4}{\mol\per\square\cm\per\s}`	$2 \times 10^{-4}\,\frac{\text{mol}}{\text{cm}^2\,\text{s}}$
`\qtylist{50;70;90}{\cm}`	50 cm, 70 cm and 90 cm
`\qtyrange{50}{70}{\celsius}`	50 °C to 70 °C
`\qtyproduct{5 x 7 x 9}{\m}`	$5\,\text{m} \times 7\,\text{m} \times 9\,\text{m}$

C.16.4 \QTY

The macro `\qty` is used to format a number and its unit simultaneously; it is a combination of `\num` and `\unit`. For instance $4.18\,\frac{\text{kJ}}{\text{kg K}}$. You will find more examples in table C.11.

C.16.5 DEFINING YOUR OWN UNITS

You can define your own units with the command `\DeclareSIUnit`. This makes sense if you often use a unit in your manuscript that is not yet available in siunitx. Some extra units are defined in this template. You will find them in `styles/mypackages` where the package siunitx is loaded. For example:

`\DeclareSIUnit\kWh{kWh}`
`\DeclareSIUnit\emission{\ch{CO2}eq}`

They can then be used as in this example: $85\,\frac{\text{gCO}_2\text{eq}}{\text{kWh}}$.

C.16.6 ALIGNING NUMBERS IN TABLES

The package also introduces a column type `S` to align numbers in tables at the decimal separator. Inspect the source code of table C.12 to see the application of these column types. Column `unit` is centred and uses the command `\unit` to write the units. The columns indicating the initial end final state are of type `S`.

C.16.7 GLOBAL SETTINGS

Many other global settings can be made, for instance settings related to the rounding of numbers, the formatting of the exponential notation, the displaying of digits, to name a few. All

additional info can be found in the user guide siunitx.pdf. You will also find some examples in `styles/mypackages` where the package siunitx is loaded.

More info can be found at CTAN: package siunitx.

C.16.8   OLDER VERSIONS

The commands `\si`, `\SI`, `\SIlist` and `\SIrange` from version 2 of siunitx remain available, but are not recommended for use in new documents. Use the new commands instead.

C.17   TIKZ

Ti*k*Z is a powerful package for creating high-quality graphics and diagrams within LaTeX documents. With it, you can design and draw a wide range of visual elements, including geometric shapes, graphs, charts, and illustrations. It comes with an extensive library of predefined styles and shapes and is often used in scientific publications by anyone who needs precise graphics in LaTeX documents.

Here are useful links to learn more

– Overleaf documentation on Ti*k*Z

– A very minimal introduction to Ti*k*Z

– PGF/Ti*k*Z Graphics for LaTeX

– A very short introduction to Ti*k*Z

– Creating Diagrams with Ti*k*Z

You can make simple inline drawings with the command `\tikz{ }`. For instance the command `\tikz{\draw[VUBblue] (0,0) rectangle (2mm,2mm)}` and the command `\tikz{\draw[VUBorange] (0,0) circle (2mm)}` will produce ■ and ● respectively.

You can also define your own drawing commands. For example: the command `\drawpumpicon`, defined in `extra/myDrawingStuff`, will draw a pump icon at a given location with a given name tag. Below is an example of a drawing of two pumps, named A and B, placed side by side with the single Ti*k*Z command `\tikz{\drawpumpicon{0}{0}{A};\drawpumpicon{3}{0}{B}}`.

More complex drawings are better made within the `tikzpicture` environment. See appendix E in this document for some examples. Many more examples can be found on TeXample.net.

More info can be found at CTAN: package Ti*k*Z.

Table C.12: Example of a property table with type `S` columns to align numbers at decimal separator

property	unit	initial state	process	end state	comment
m	kg	1.0	$=$	1.0	closed
T	°C	25.0	$=$	25.0	isotherm
$\mathcal{V}$	m^3	1.00	$>$	0.83	compression

C.18   TODONOTES

The `todonotes` package provides the command `\todo` to insert to-do items in your document. For instance the command `\todo{example of a margin note}` renders to the note in the right margin of the PDF.

example of a margin note

The package also provides the `\missingfigure` command to create a placeholder for a picture you still need to add. For example: `\missingfigure{Search for better picture}`

Missing figure

Search for better picture
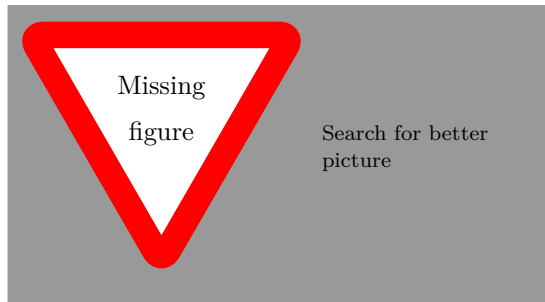
At any point in your document, a list of all the inserted to–do items and missing figures can be listed with the `\listoftodos` command. In this template, the `\listoftodos` command can be found at the last line of `mainReport.tex` and thus you will find the list of to do's at the last page of this PDF.

If you wish to temporarily hide all the to-do items in your document, you can utilize the `disable` option when loading the package as follows: `\usepackage[disable]{todonotes}`. This option allows you to work on your document without the distraction of to-do notes until you decide to enable them again. More info can be found at CTAN: package todonotes.

Useful commands

This chapter contains an alphabetic list of some useful commands. All the commands that are mentioned here are included in the style file `styles/myCommands`.

D.1   Basic commands in LaTeX

There are several good sources to learn about commands in LaTeX. See for instance the list below

- LaTeX command summary
- LaTeX cheat sheet
- Wikibook: LaTeX command glossary

Below we will mention some commands that might be of interest for you.

D.2   \ChapterOrPart

This command is defined in `styles/myCommands.sty`. It starts a new chapter when compiled with the class `memoir` to make a report or thesis. But it will start a new part, instead of a new chapter, when compiled with the class `beamer` to make slides. The reason being that the concept of a chapter does not exist in a presentation made with `beamer`. The `\ChapterOrPart` command also executes a `\mode*` command so that content outside frame environments is ignored when compiling with `beamer`, see also appendix D.14.

D.3   \def

This command is typically used to define constants that can be accessed and employed across your entire document. The constants can be numerical or text strings. The syntax is `\def\commandname{replacement}` where `\commandname` is the name of the command you want to define, and `replacement` is the value or text that will be substituted when the command is used. In this template, you can find several examples of the use of `\def`.

For an example of a definition of a numerical constant, see the command `\def\arcr{0.8cm}` in the source code of fig. E.10. It assigns the value of 0.8 cm to `\arcr` and is then used as the radius of the arc to indicate angles in the free body diagram. Feel free to adapt this value and observe the effect on the free body diagram after a recompile.

For defining of text strings, inspect the files `extra/titleInfoVUB` and `extra/titleInfoULB`, where you will find definitions of text strings that are used to compose the VUB and ULB title pages. Other examples can be found in the file `extra/nomenclature`, where section names for the nomenclature are defined, depending on the document language. The document language is part of the options of the command `\documentclass[...]` at the first line of the main

41

file. For your info: the current language of a document can be displayed with the command `\languagename`. In this document, it is currently set to: `english`[1].

Warning: the command `\def` does not check if a command name already exists, so it does not prevent accidental redefinition of existing commands. So exercise caution to prevent unintentional overwriting of existing commands. If you want error checking, use `\newcommand` instead, as explained in appendix D.15.

D.4 \DEFINECOLOR

You can define your own colours. For instance:

`\definecolor{VUBorange}{cmyk}{0.00, 0.78, 1.00, 0.00}`

defines the colour VUBorange. This colour can now be used in your document, as is done on the title page. For instance, the command

`\textcolor{VUBorange}{This text is written in VUB orange}`

will produce: This text is in VUB orange.

D.5 \ENSUREMATH

The command `\ensuremath` ensures that its argument is enclosed in math mode, regardless of whether the surrounding text is in math mode or regular text mode. It is typically used inside `\newcommand`. For instance, the command `$\psat$` will produce p_{sat} but the command without the dollar-signs `\psat` will produce exactly the same result. This is because the command `\ensuremath` is used inside the definition of `\psat` (you can verify this in the style file `styles/myCommands`). It will force a switch to math mode, if not yet in math mode. This is very convenient because you are no longer obliged to explicitly activate math mode with the $-signs, every time you use such a command when writing.

D.6 \HIDEEXPLANATIONS

The command `\hideExplanations` will hide the explanations in blue at the beginning of chapters. You will find this command in the preamble of `mainReport` where is currently command out so the explanations are visible.

D.7 \INCLUDE

The command `\include{⟨filename⟩}` is used to include chapters. It will process all the floats (figures and tables) from the previous chapter, cause a page break so that the next chapter can start on a new page and import the content from the file named ⟨filename⟩ into the document body. It will also start a new auxiliary file in the background to keep track of the locations of figures, tables and equations within the chapter for proper cross-referencing. It is not possible to nest `\include` commands.

D.8 \INCLUDEONLY

This command is used to reduce the compilation time of your Overleaf project. There are two ways to reduce compilation time:

---

[1]Reminder: names of language options are not capitalized in LaTeX.

1. Reduce the amount of content in the document part. This can be achieved by disabling `\include{⟨`*path/filename*`⟩}` commands. Just put a %-sign at the beginning of each line that you want to exclude. For instance: `%\include{chapters/introduction}` will no longer compile the `introduction` file. As a result:

   – The content of `introduction` will no longer be known to LaTeX, and thus will no longer show up in the output PDF

   – The content of `introduction` will no longer show up in the table of contents

   – Labels in `introduction` will no longer be known to LaTeX, so cross-references to these labels, from anywhere in your document, will result in broken links, indicated by a double question mark: ??

2. Use the command `\includeonly{⟨`*path/filename*`⟩}` in the *preamble*. But beware, you must proceed as follows:

   a) First compile your entire project. You can achieve this by putting a %-sign in front of the `\includeonly` command in the *preamble*. This will allow LaTeX to detect the entire document structure (chapters, sections, labels, . . .) in all the included files.

   b) After successful compilation, you can now restrict compilation to the file you are currently working in. Let's assume it is the `introduction` file. In that case, you add the command `\includeonly{chapters/introduction}` to your *preamble*[2]. As a result:

      – Only the content of `introduction` will be recompiled by LaTeX. The other included files are not recompiled. This will speed up compilation! The content of all the other compiled files is however still known to LaTeX from the previous full compile.

      – Only the content of `introduction` will show up in the output PDF.

      – The content of all included files will show up in the table of contents, because LaTeX remembers the document structure from the first entire compile.

      – All the labels in all compiled files are known to LaTeX, so cross-references to such labels, from anywhere in your document, will work.

D.9   \INCLUDEGRAPHICS

The `\includegraphics` command is used to include images or other types of graphics in your document. Often included file types are PDF, PNG and JPG. PDF is preferred over PNG or JPG because it is a vector format and thus it retains its image quality on all zoom levels. The general form of the command is `\includegraphics[⟨`*options*`⟩]{⟨`*filename*`⟩}`. Often used options are listed in table D.1. When more than one option is specified, they will be processed in the order they are specified, so from left to right. This is important when combining sizing and rotating of an image.

The command is provided by the package `graphicx`, a widely used package in LaTeX for handling graphics. More info at CTAN: package graphicx.

D.10   \INCLUDESVG

The `\includesvg` command is used to include Scalable Vector Graphics (SVG) files in your document. SVG is a widely used vector image format that allows for high-quality images at any scale without loss of resolution.

---

[2]Do not add the `\includeonly` command to the document part, it does not belong there and will cause an error.

Table D.1: Often used options for `\includegraphics`

option example	explanation
`width=0.8\textwidth`	set image width to 80% of the textwidth, height is scaled proportionally
`height=0.7\textheight`	set image height to 70% of the textheight, width is scaled proportionally
`keepaspectratio`	ensures that the aspect ratio of the image is maintained when both width and height are specified
`scale=0.6`	scale image to 60% of size
`angle=30`	rotate image 30° counterclockwise
`frame`	add frame around image
`trim={1cm 2cm 3cm 4cm}`	hide parts of image: 1 cm from left, 2 cm from bottom, 3 cm from right, 4 cm from top
`clip`	remove parts of image that were hidden via trim

The general form of the command is `\includesvg[⟨options⟩]{⟨filename⟩}`. It is provided by the package `svg`. More info at CTAN: package svg.

D.11 \INPUT

The `\input{⟨filename⟩}` command is used to insert the contents of another file named ⟨*filename*⟩ into the document body. It's equivalent to typing all the content from the file named ⟨*filename*⟩ into the target file at the location of the `\input{⟨filename⟩}` command. It is used to break up your project into smaller, more manageable parts. It will not start on a new page. It is possible to nest `\input` commands.

D.12 \INTERTEXT

The command `\intertext` is useful to write lines of text between equations. It is applied in the `align` environment. You can then use one single `align` environment, instead of a series of separate `equation` environments. Inspect the source code in the example below. This approach is also very well suited for writing equations on slides.

For an adiabatic CV, the entropy balance is

$$\frac{\mathrm{d}S_{\mathrm{cv}}}{\mathrm{d}t} = \sum_{\mathrm{in}} \dot{m}_i \cdot s_i - \sum_{\mathrm{out}} \dot{m}_j \cdot s_j + \dot{S}_{\mathrm{gen}} \tag{D.1}$$

for a closed CV with k heat flows, it takes the following form

$$\frac{\mathrm{d}S_{\mathrm{cv}}}{\mathrm{d}t} = \sum \frac{\dot{Q}_k}{T_k} + \dot{S}_{\mathrm{gen}} \tag{D.2}$$

and for an isolated CV it reduces to

$$\frac{\mathrm{d}S_{\mathrm{cv}}}{\mathrm{d}t} = \dot{S}_{\mathrm{gen}} \geq 0 \qquad \text{and thus } S \text{ reaches a maximum at equilibrium.} \tag{D.3}$$
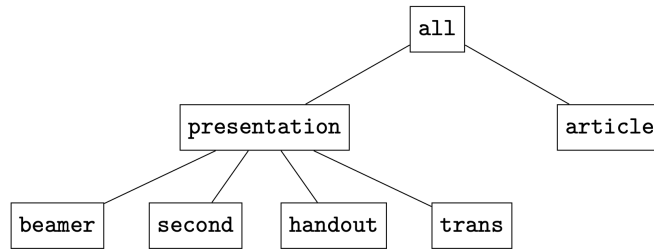
Figure D.1: The seven modes in beamer (beamer manual, p. 205)

D.13   \MATHRM

The command `\mathrm` is used to write upright symbols in mathematical expressions, such as for example the Reynolds number in eq. (D.4). See also the shorter command `\up` with the same effect, discussed in appendix D.19.

$$\text{Re} = \frac{V \cdot L}{\nu} \tag{D.4}$$

D.14   \MODE

Read this section if you want to understand how one source file can be used for different outputs. The `\mode` command is used to control the inclusion of content based on the mode in which the document is being compiled. This is useful for source files that are used to serve multiple purposes, for instance to generate slides and to make a report/thesis.

The beamer class supports seven modes, as shown in fig. D.1, each designed for different output formats. As you likely already know, this template utilizes two classes. When you compile with class beamer to create slides, the <presentation> mode is active. Conversely, when you compile with class memoir to produce a report or thesis, the <article> mode is active. The name of the <article> mode is somewhat misleading, because it is not only active when you compile a project with the article class. It is active for all output types that are not presentations. So when you compile with classes like book, article, report, or memoir, the <article> mode is active.

By using `\mode<presentation>{⟨content⟩}` or `\mode<article>{⟨content⟩}`, you can specify content that will only be processed when the respective mode is active. This is for instance used in `styles/myPageSetup.sty` to perform a separate page setup for the slides and for the report/thesis.

There are two additional commands related to modes that are frequently used throughout this template. The first one is the `\mode*` command which is particularly useful when compiling with the beamer class, as it ensures that the compiler ignores all content that is not within frame environments. This command is typically placed at the beginning of tex file, and in this template, it is incorporated into the `\chapterOrPart` command, which is defined in `styles/myCommands.sty`. The second command is `\mode<all>` which instructs the compiler to process all content, and it is generally written at the end of tex file. Without `\mode<all>` at the end of a tex file, the subsequent `\chapterOrPart` command in the next tex file would be ignored, as it would fall outside a frame environment.

D.15   \NEWCOMMAND

With the command `\newcommand` you can define your own commands. That can be useful to simplify repetitive and/or complex formatting. See the Overleaf documentation on newcommand for more info and the examples below.

```
\newcommand{\vel}[0]{\ensuremath{V}}
```
example: \vel= \qty{2}{m/s}

result: $V = 2\,\mathrm{m/s}$

```
\newcommand{\psat}[0]{\ensuremath{\pres_\up{sat}}}
```
example: \psat = \qty{3.2}{\kPa}

result: $p_\mathrm{sat} = 3.2\,\mathrm{kPa}$

```
\newcommand{\units}[1]{\quad\left[\unit{#1}\right]}
```
example: $T \ud s = \delta q + \delta d \units{\J\per\kg}$

result: $T\mathrm{d}s = \delta q + \delta d \quad \left[\frac{\mathrm{J}}{\mathrm{kg}}\right]$

```
\newcommand{\dutch}[1]{[{Dutch:} {#1}]}
```
example: The flow is steady \dutch{stationair}.

result: The flow is steady [Dutch: stationair].

```
\newcommand{\cvwatervalue}[0]{\qty{4.18}{\kJ\per\kg\per\K}}
```
example: \cvwatervalue

result: $4.18\,\frac{\mathrm{kJ}}{\mathrm{kg\,K}}$

D.16 \NEWTOGGLE

Toggles are boolean variables that can be set to true or false. They allow conditional processing based on whether a toggle is set to true or false, enabling the inclusion or exclusion of specific content and formatting in a document.

For instance, the command `\newtoggle{showResults}` defines the toggle `showResults`. You can set this toggle to true with the command `\toggletrue{showResults}` and to false with `\togglefalse{showResults}`. With the command `\iftoggle` we can then perform some conditional processing, for example the showing or hiding of some text, as shown below. The command

`$1+1=3$ \iftoggle{showResults}{for very large values of 1.}{for \dots}`

will produce:

$1 + 1 = 3$ for ... (when `showResults` is set to FALSE)

$1 + 1 = 3$ for very large values of 1 (when `showResults` is set to TRUE)

D.17 \TEXT

The command `\text` is used to write normal text when you are in math mode. Assume for instance that your want to add Newton's second law of motion as a numbered equation in your manuscript and that you want to add a sentence behind it. The correct way of doing this is shown in eq. (D.5). Another example of its use can be seen in eq. (D.3).

$$F = m \cdot a \quad \text{is Newton's second law of motion.} \quad \text{(correct text formatting)} \tag{D.5}$$

If you forget to apply the command `\text` for the sentence, LATEX will consider the sentence as a math expression. As a consequence, three things will happen to it. First, all the letters will be written in italics, because they are all considered as math symbols. Second, the spaces between the words will disappear, because LATEX does not consider this series of letters as a sentence consisting of separate words. Third, since LATEX thinks that this sentence is a mathematical expression, it will add spaces between some symbols, for instance around the letter f. See this unacceptable result:

$$F = m \cdot a \quad isNewton'ssecondlawofmotion. \quad \text{(text formatting not acceptable)}$$

Using the command `\mathrm` to put text upright does not produce the desired outcome. It will put all the letters of the text upright, but is is still considered as a mathematical expression, hence the strange lack of spacing, as shown here:

$$F = m \cdot a \quad isNewton'ssecondlawofmotion. \quad \text{(text formatting not acceptable)}$$

For the correct use of the command `\mathrm`, see appendix D.13.

D.18 \TEXTCOLOR

With this command you can colour text. For instance the command

`\textcolor{blue}{This text is coloured}`

will produce: This text is coloured.

D.19 \UP

The command `\up` is identical to the command `\mathrm` but is more convenient because it is shorter. It is defined in `styles/myCommands` as `\newcommand\up[1]{\mathrm{#1}}`. For the use of `\newcommand` see appendix D.15. Inspect the source code of eq. (D.6) to see how `\up` is used to write Re upright in math mode. More info on writing in italics versus upright can be found in fig. B.3.

$$\mathrm{Re} = 2300 \tag{D.6}$$

D.20 \TEXTSUPERSCRIPT

This command is used to write superscripts in your text, such as for example: 1st, 2nd and 3rd.

This chapter contains some examples of figures and charts. More will be added in the future.

It's possible to make all kinds of drawings in LaTeX with the package TikZ. The style of objects can be predefined. See for instance the source code for fig. E.1 where the blue fill colour and the red dashed border of the three systems are defined with the command `\tikzstyle` in the style file `styles/myDrawingStuff`.
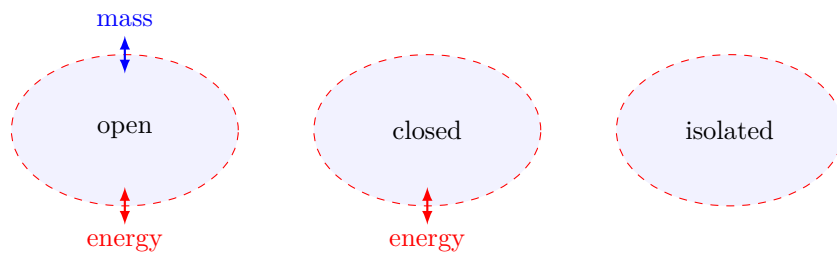
E.1 Simple TikZ drawing

Figure E.1: Mass and energy transfer in thermodynamic systems, to illustrate the use of `tikzstyle` for defining object properties

E.2 Drawing block diagrams

It's possible to draw block diagrams in TikZ, as for example in fig. E.2. In this figure, the properties of the blocks and connectors are defined globally in `styles/myDrawingStuff`, so that they can also be used in other diagrams in the document.
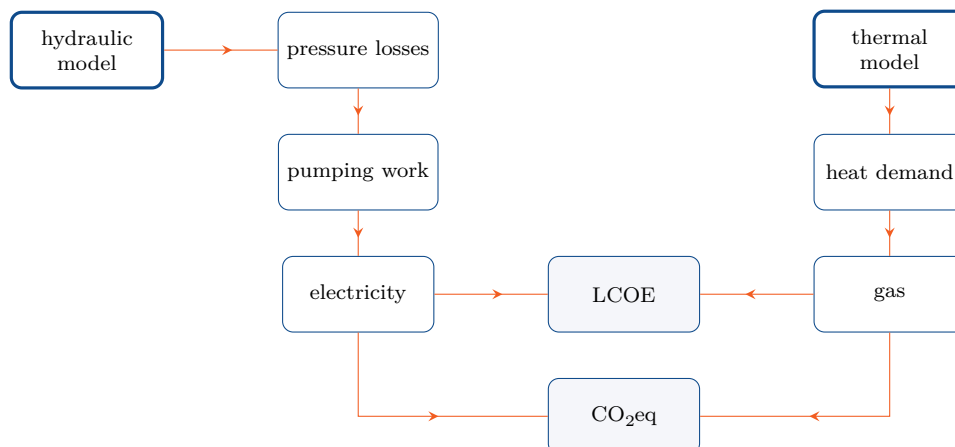

Figure E.2: Block diagram drawn in TikZ, to illustrate the use of drawing styles and connectors
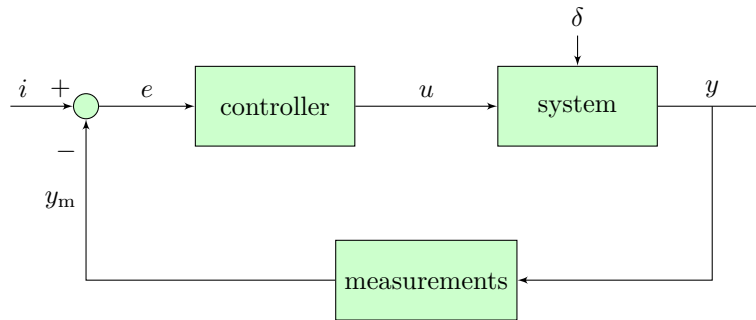
Figure E.3: Example of a system control diagram drawn in Ti*k*Z

Another type of block diagram, a process control scheme, is shown in fig. E.3. The original source code for fig. E.3 can be found on TeXample.net, built by Fauske (2020) and was adapted in this template.

E.3   PROCESS FLOW DIAGRAM

Process flow diagrams (PFDs) can be drawn with the help of the **pfdicons** package. An example of a PFD of a Rankine cycle is shown in fig. E.4. See also appendix C.14.
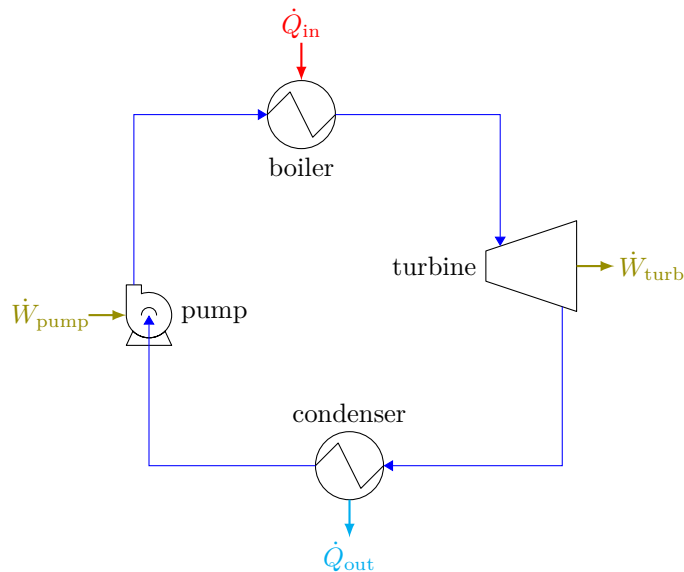

Figure E.4: Process flow diagram of a Rankine cycle (adapted from Drews, 2021, p. 19), to illustrate the capacity of the package **pfdicons** to draw PFDs

E.4   EXTRACTING A SPECIFIC REGION FROM A PDF FILE

It's easy to extract a specific rectangular region from a PDF file while preserving the image quality in vector format. This is useful for incorporating figures and tables from publications into your work without sacrificing image quality, as often happens when resorting to screenshots. This method is illustrated in fig. E.5 where a specific figure from a certain page of a PDF file is included twice. The left figure is included with the command `includegraphics` with the desired `page` and `trim` values, combined with the option `clip`. By doing so, the vector format will be retained. Consequently, the figure remains sharp, even when scaled up for placement on a poster. The right figure is a screenshot of the same region. As you can see, its resolution
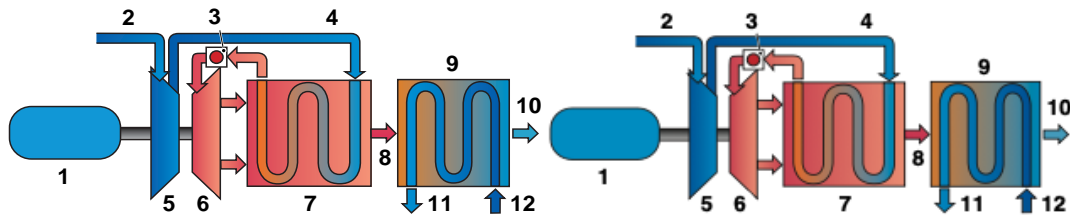
Figure E.5: Process flow diagram of a microturbine included as PDF (left) and as PNG (right), to demonstrate the superior quality of PDF over PNG

is limited, making it less suited for magnification and placement on a poster. You can verify this by zooming in on the two figures.

E.5 ADDING ANNOTATIONS ON AN INCLUDED PICTURE

An example of an included picture file (`JPEG, PNG, ...`) with annotations added in LaTeX is shown in fig. E.6. The image of the expansion valve is included with the `\includegraphics` command. The dotted CV line, the arrows and symbols are added within LaTeX with simple TikZ drawing commands. Inspect the source file to see how. Another example is shown in fig. E.7 where a scale is added to an included photo. The properties of the scale (position, color, unit, length, number of ticks) can be customized in the source code.

E.6 PROPER SIZING OF A FIGURE ON A SLIDE

The aspect ratio of the A4 page format differs from the aspect ratio of a slide in beamer. As a consequence, choosing the proper dimensions of figures, to make them look good on A4 and also on a slide, is sometimes challenging. The trick is to define both the height and width, in combination with the option `keepaspectratio`. It will freeze the aspect ratio of an included picture, to make it fit on A4 and on a slide. Inspect the source code of fig. E.8 to find out how.
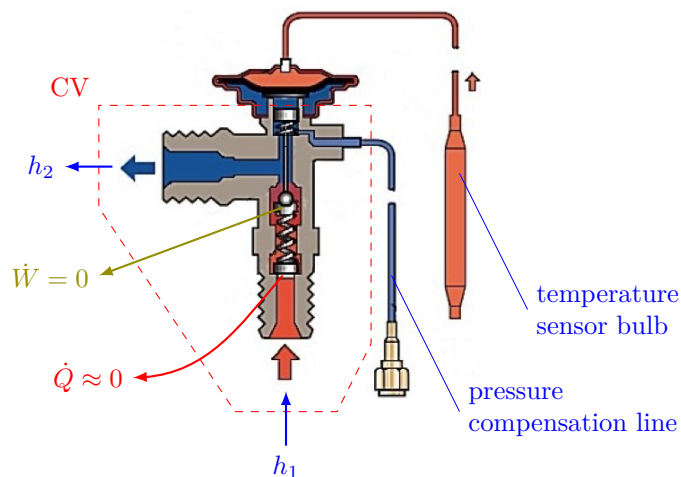

Figure E.6: Energy flows in a thermostatic expansion valve, to illustrate the addition of annotations in TikZ
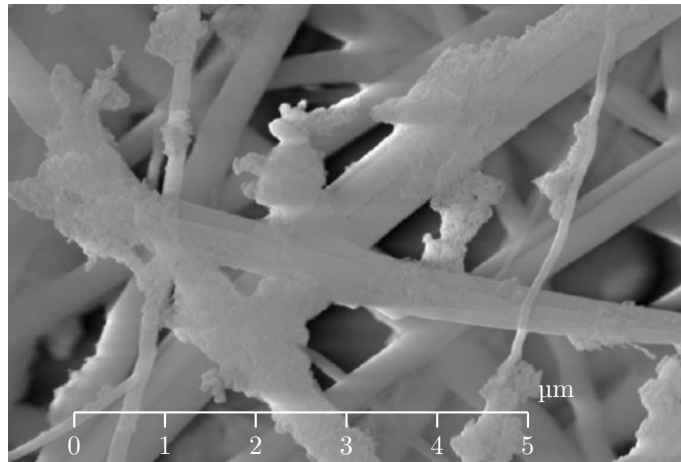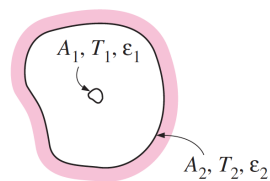
Figure E.7: SEM image of particles on the fibres of a filter, collected in the flue gas from a biomass boiler, to illustrate the addition of a customized scale with Ti*k*Z
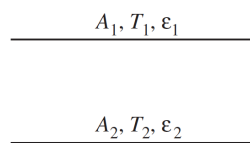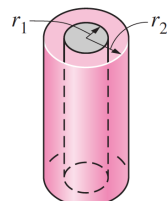

Figure E.8: Exchange of thermal radiation for some common geometries (Cengel et al., 2016), to illustrate the use of `\keepaspectratio` to correctly size an included picture file in a manuscript and on a slide

Figure E.9: Velocity triangles at the root (1) and tip (2) of a impeller blade in a centrifugal pump, to illustrate the drawing of vectors and angles in Ti*k*Z

E.7 DRAWING VECTORS

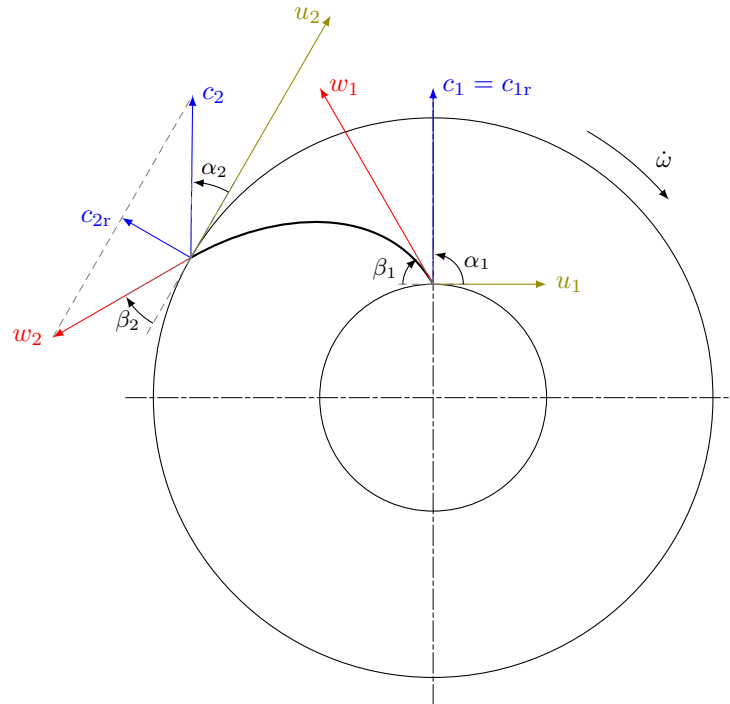Vectors can easily be drawn with Ti*k*Z. See for instance the velocity triangles in a centrifugal pump in fig. E.9. In this drawing, several pump parameters are defined with the command `\def`. The goniometric calculations to determine the orientation and magnitude of the velocities are performed with the command `\pgfmathsetmacro`. Inspect the source code of the figure in the file `figs/centrifugalpump` for more info. Altering the parameters of the pump will cause the velocity triangles to be automatically recalculated and redrawn.

E.8 DRAWING FREE-BODY DIAGRAMS

An example of a mechanical sketch of two bodies on a slope connected with a wire and the related free-body diagrams are shown in fig. E.10. The original Ti*k*Z code was written by Fauske, 2008 and can be found on TeXample net. Drawings in Ti*k*Z can be parametric. In this case, the inclination angle of the slope can be altered by changing the value in the definition `\def\iangle{45}` in the source code. The angle in fig. E.10 is 45° while the angle in the original drawing was 30°.

E.9 DRAWINGS WITH A 3D LOOK

Drawings that create a 3D look, are also easy to make, as shown in fig. E.11.

E.10 PLOTTING AIRFOILS

The Applied Aerodynamics Group of the university of Illinois has an online database of more than 1600 airfoils (*Airfoil Coordinates Database* 2020). The airfoil data is made available in
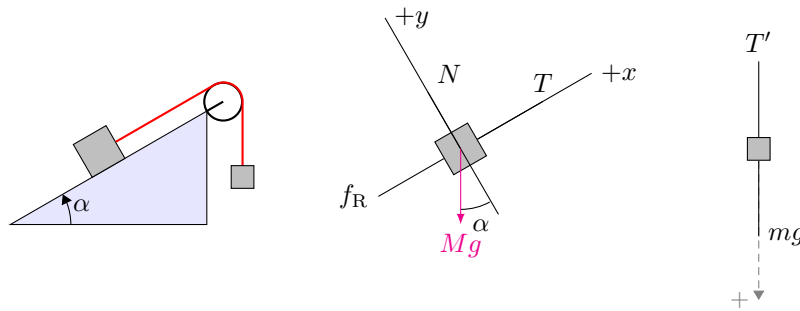
Figure E.10: A mechanical sketch and two corresponding free-body diagrams made with Ti*k*Z (adapted from Fauske, 2008), to illustrate the use of parameters in drawings
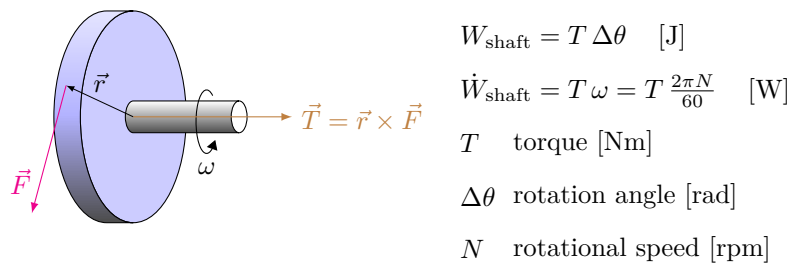

$$W_{\text{shaft}} = T \, \Delta\theta \quad [\text{J}]$$

$$\dot{W}_{\text{shaft}} = T \, \omega = T \, \frac{2\pi N}{60} \quad [\text{W}]$$

T torque [Nm]

$\Delta\theta$ rotation angle [rad]

N rotational speed [rpm]

Figure E.11: Transfer of mechanical power with a shaft, to illustrate the capacity of Ti*k*Z to make 3D drawings

data files with the extension `.dat`. Data files can be plotted directly in LATEX if the headers are removed. See fig. E.12 where the Eppler 625 airfoil data is plotted, combined with a grid.

E.11   Repeating patterns

Technical and scientific drawings often contain repeating patterns. The command `\foreach` can be used to repeat drawing commands, to create patterns like for instance the pattern of tubes in fig. E.13. The number of rows and layers of tubes can easily be adapted in the source code. Other examples can be found in the Pgf/Ti*k*Z manual.

E.12   Plotting analytical functions

It's also easy to plot analytical functions as shown in fig. E.14. In this example, three analytical functions are plotted. Inspect the source code to see the various plot settings, such as the domain, the minimum and maximum values on the axes, the size of the plot, the legend and line styles.
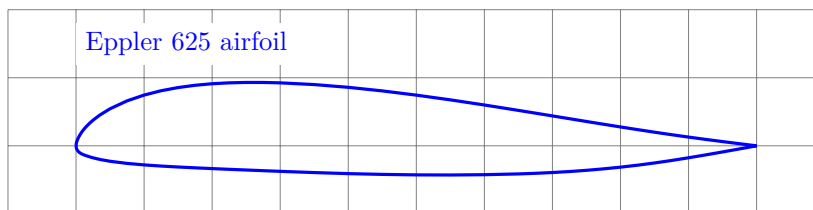

Figure E.12: A plot of an Eppler 625 airfoil with Ti*k*Z (data from *Airfoil Coordinates Database* 2020), to illustrate the plotting of data from a DAT-file
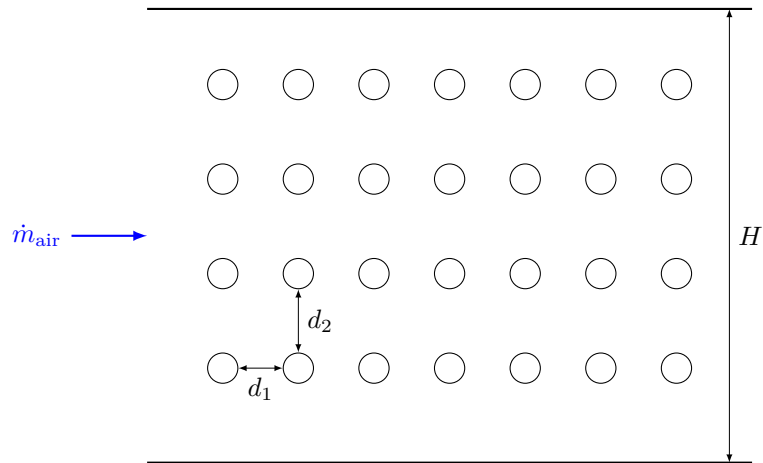
Figure E.13: A cross-flow heat exchanger with the pipes arranged in 4 rows and 7 layers, to illustrate the use of `\foreach` in Ti*k*Z to draw patterns
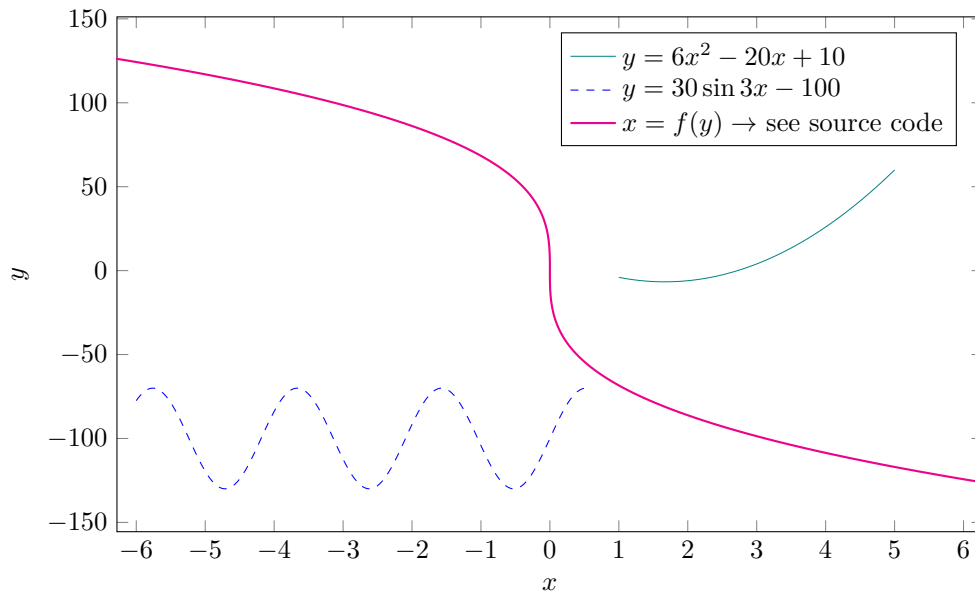
Figure E.14: Plot of three analytical functions, to illustrate various types of plot settings in Ti*k*Z

E.13   PLOTTING RANDOM DATA

E.14   PLOTTING EXPERIMENTAL DATA AGAINST A MODEL

It is also possible to combine analytic functions with experimentally obtained data in a single plot, with inclusion of the measurement errors. This is shown in fig. E.16 where experimental data is plotted against a simple parabolic model.

E.15   PREDEFINED PLOT SETTINGS

It's also possible to predefine the settings of axes. See for instance the definition of `axespsychro` in the style file `styles/myDrawingStuff` and how easy it is to use it as an option in: `\begin{axis}[axespsychro]` in the source code of fig. E.17 and fig. E.18. Both charts have exactly the same appearance because of the same predefined settings.
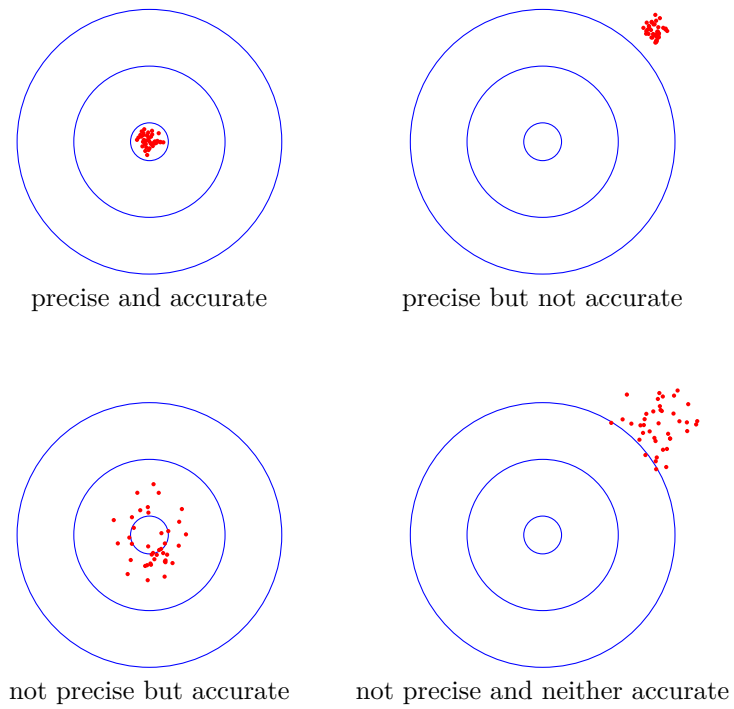
Figure E.15: Four combinations of accuracy and precision, to illustrate the plotting of random data in Ti*k*Z, adapted from StackExchange
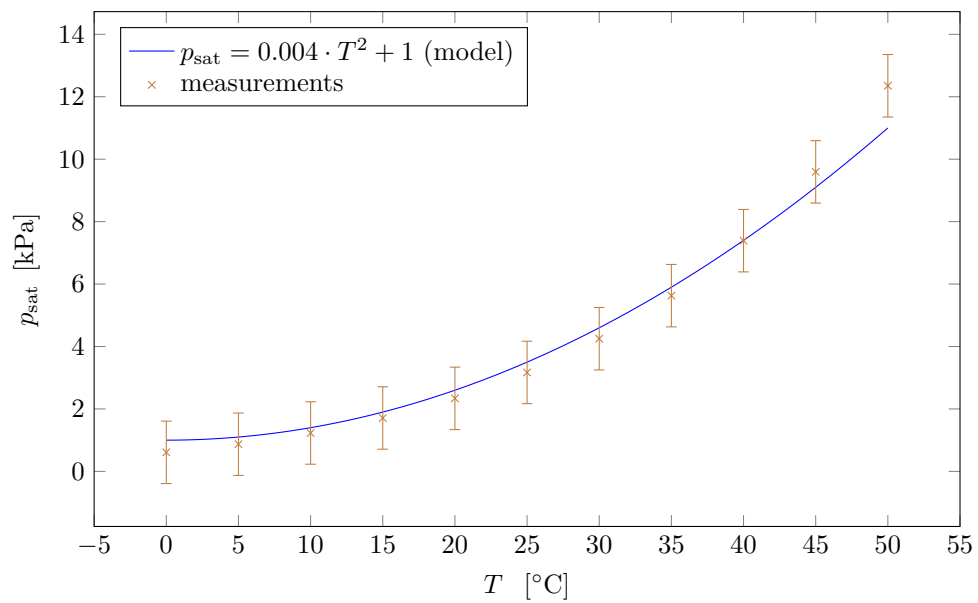
Figure E.16: The saturation pressure of water as function of temperature, to demonstrate the plotting of model data against experimental data with error bars
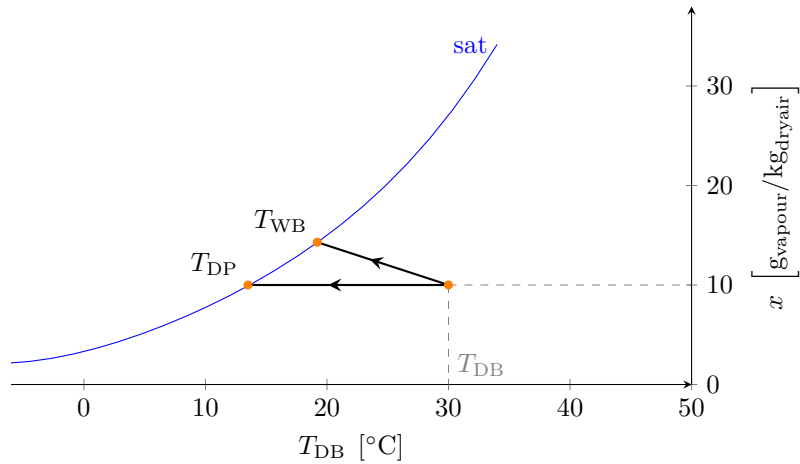
Figure E.17: A simplified psychrometric chart with indication of the dew point and the wet bulb temperature of humid air, to illustrate the use of predefined plot settings
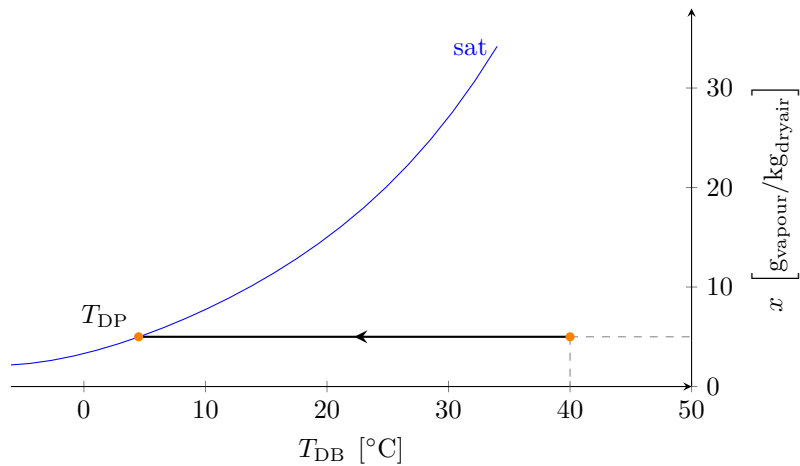
Figure E.18: A simplified psychrometric chart with indication of the dew point of humid air, to illustrate the use of the same predefined plot settings as in fig. E.17

E.16 Grouping of plots

Plots can be grouped in such a way that they share the same x-axis and/or y-axis. An example where both axes are shared is shown in fig. E.19.

E.17 Plotting data from an external file

Data can also be read from external files and then plotted, as in fig. E.20 and fig. E.21. Even if the data in the file contains date- and timestamps (like for instance of the type `YYYY-MM-DD HH:MM`) they can still be plotted by using the Ti*k*Z library `dateplot`. An example of such a plot is shown in fig. E.22. The weather data was fetched from *Iowa Environmental Mesonet* 2020, an ever-growing archive of automated airport weather observations from around the world.
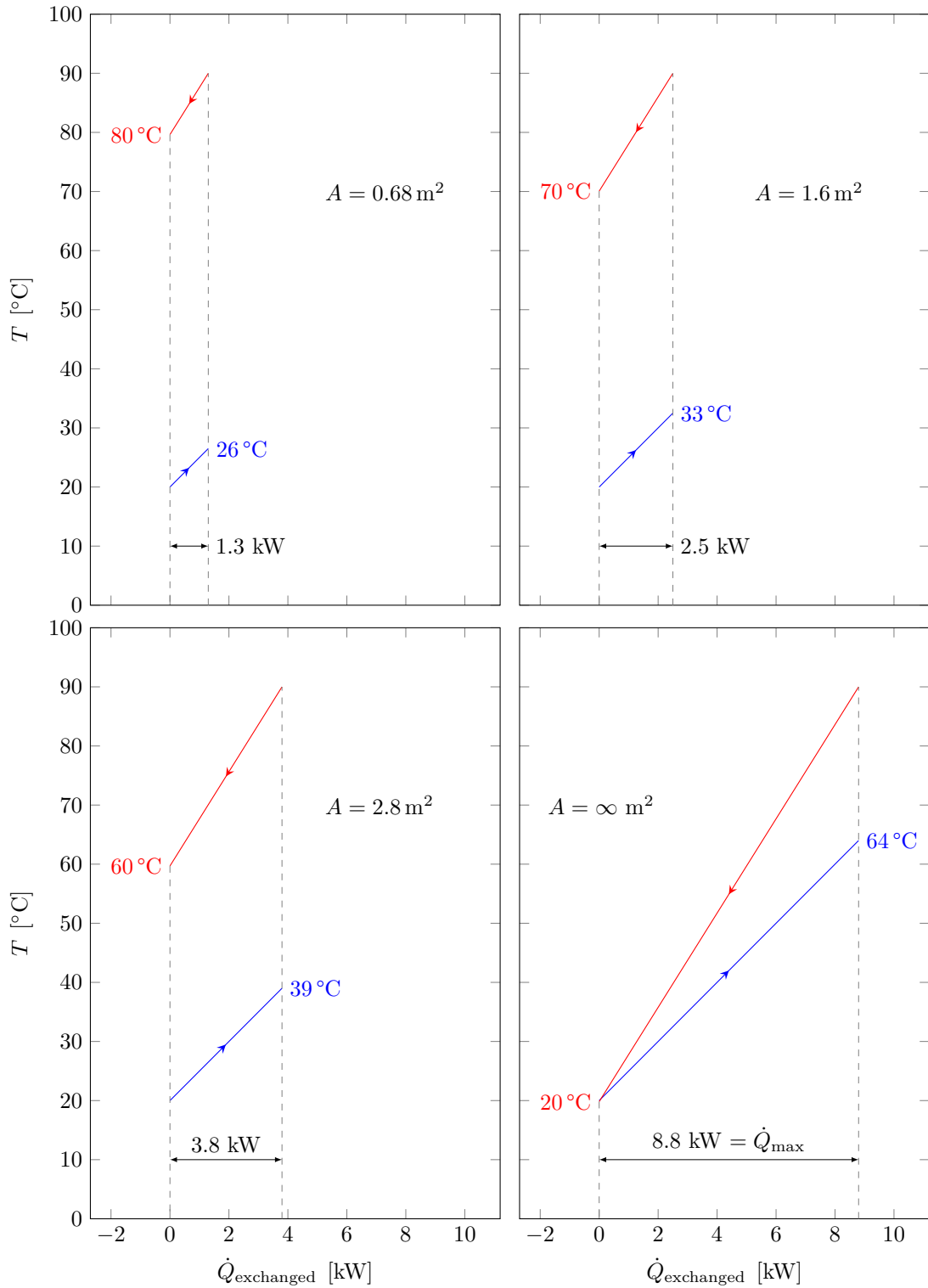
Figure E.19: The effect of heat exchanger surface on heat exchanger duty, to illustrate the use of shared axes in group plots
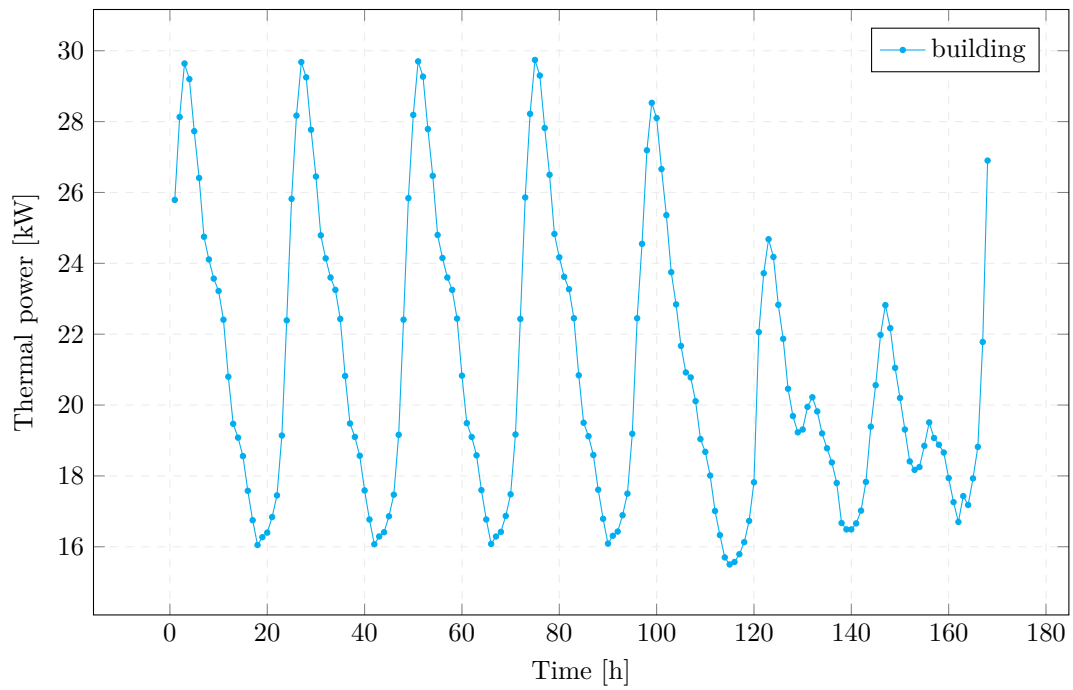
Figure E.20: Thermal power demand for a building during one week, to illustrate the plotting of time series data from a csv-file
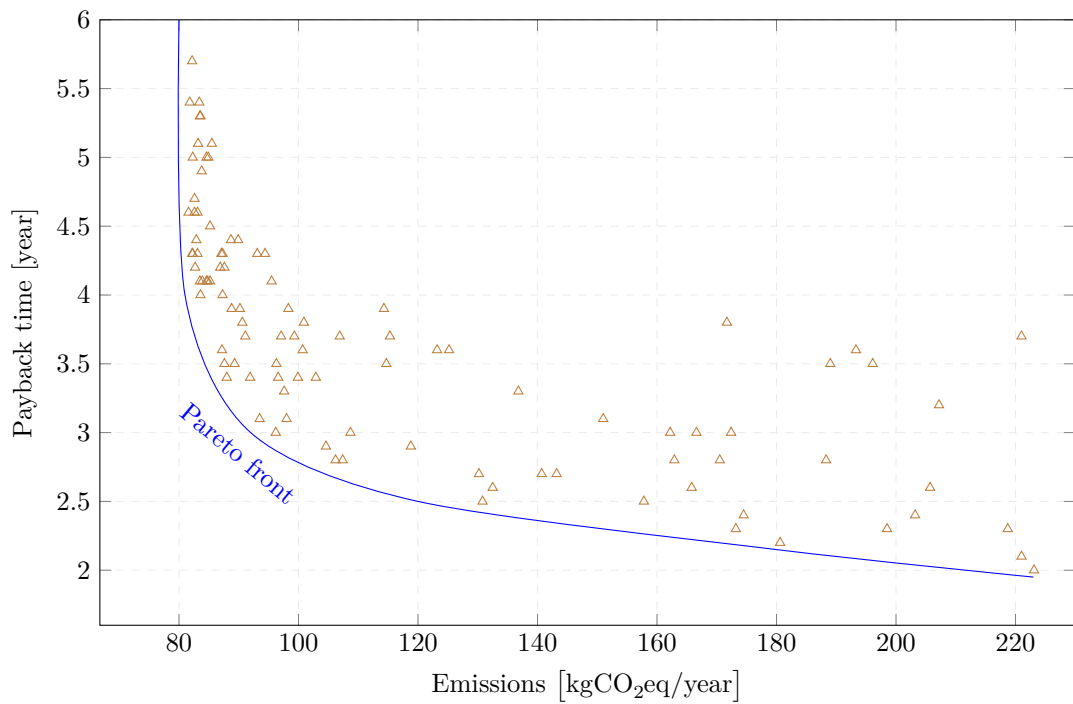
Figure E.21: The payback time and the CO_2 emissions for different configurations of a solar panel installation, to illustrate the creation of scatter plots from data on a csv-file

Figure E.22: Weather data for Brookneal Virginia (USA) in the first week of April '20, to illustrate the sizing of plots, here the plot is sized to fill the entire page

Math symbols

Table F.1 contains a selection of often used mathematical symbols. The comprehensive list of all LaTeX symbols can be found at CTAN.

Table F.1: Some often used mathematical symbols in LaTeX

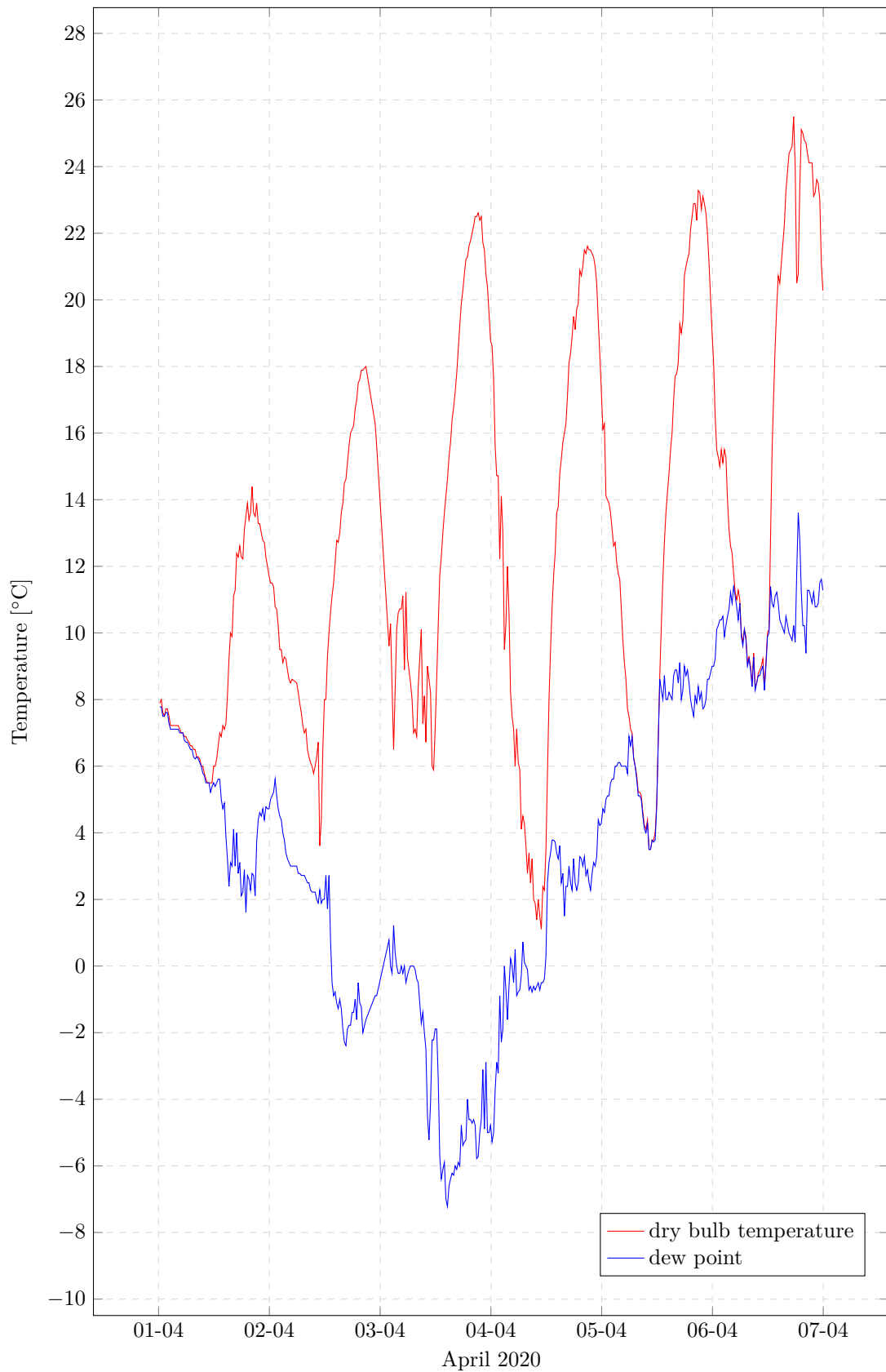| symbol	LaTeX	description	symbol	LaTeX	description
relations					
$=$	`=`	equal to	$\equiv$	`\equiv`	equivalent to
$\neq$	`\neq`	not equal to	$\approx$	`\approx`	approximately
$\sim$	`\sim`	similar	$\bot$	`\bot`	perpendicular
$<$	`<`	less than	$>$	`>`	greater than
$\ll$	`\ll`	much less than	$\gg$	`\gg`	much greater than
$\leq$	`\leq`	less or equal to	$\geq$	`\geq`	greater or equal to
math					
$\sum$	`\sum`	sum	$\sin$	`\sin`	sine
$\prod$	`\prod`	product	$\cos$	`\cos`	cosine
$\int$	`\int`	integral	$\tan$	`\tan`	tangent
$\sqrt[y]{x}$	`\sqrt[y]{x}`	root	$\arcsin$	`\arcsin`	inverse sine
$\ln$	`\ln`	natural logarithm	$\arccos$	`\arccos`	inverse cosine
$\log$	`\log`	logarithm base 10	$\arctan$	`\arctan`	inverse tangent
$\frac{a}{b}$	`\frac{a}{b}`	fraction	∞	`\infty`	infinity
$\times$	`\times`	multiplied by	$\cdot$	`\cdot`	centre dot
a^x	`a^x`	to the power	$\dots$	`\dots`	horizontal dots
$\dot{x}$	`\dot{x}`		$\vdots$	`\vdots`	vertical dots
$\vec{x}$	`\vec{x}`		∂	`\partial`	partial
$\hat{x}$	`\hat{x}`		$\hbar$	`\hbar`	h bar
$($	`\left(`	resizing left brace	$\pm$	`\pm`	plus minus
$)$	`\right)`	resizing right brace	∇	`\nabla`	nabla operator
sets and logic					
$\mathbb{N}$	`\mathbb{N}$`	natural numbers	$\exists$	`\exists`	there exists at least 1
$\mathbb{R}$	`\mathbb{R}$`	real numbers	$\nexists$	`\nexists`	there does not exist
$\cup$	`\cup`	union	$\forall$	`\forall`	for all
$\cap$	`\cap`	intersection	$\implies$	`\implies`	implies
$\subset$	`\subset`	subset of	$\iff$	`\iff`	if and only if
$\in$	`\in`	member of	$\to$	`\to`	right arrow
$\notin$	`\notin`	not a member of	$\gets$	`\gets`	left arrow
$\emptyset$	`\emptyset`	empty set	$\leftrightarrow$	`\leftrightarrow`	left right arrow
miscellaneous					
$\checkmark$	`\checkmark`	check mark	$\%$	`\%`	percent
$\copyright$	`\copyright`	copyright	$\&$	`\&`	ampersand
$\$$	`\$`	dollar	$\#$	`\#`	hash tag
$\euro$	`\euro`	euro			

Table G.1: Greek symbols in LaTeX

lowercase	LaTeX	uppercase	LaTeX
α	\alpha		
β	\beta		
γ	\gamma	Γ	\Gamma
δ	\delta	Δ	\Delta
ϵ	\epsilon		
ζ	\zeta		
η	\eta		
θ	\theta	Θ	\Theta
ι	\iota		
κ	\kappa		
λ	\lambda	Λ	\Lambda
μ	\mu		
µ	\textmu[1]		
ν	\nu		
ξ	\xi	Ξ	\Xi
π	\pi	Π	\Pi
ρ	\rho		
σ	\sigma	Σ	\Sigma
τ	\tau		
υ	\upsilon	Υ	\Upsilon
ϕ	\phi	Φ	\Phi
φ	\varphi	$\varPhi$	\varPhi
χ	\chi		
ψ	\psi	Ψ	\Psi
ω	\omega	Ω	\Omega

---

[1]This symbol is used for the prefix `micro` in units, for example: µm

International Standard Atmosphere

This is an example of an appendix with a table.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line \hideExplanations in your main file.

Table H.1: Some properties of the International Standard Atmosphere (ISA) as function of the altitude above sea level, ranging from -1000 ft to 30000 ft

altitude		T	p	ρ	V_s	μ
ft	m	K	mbar	$\frac{\text{kg}}{\text{m}^3}$	$\frac{\text{m}}{\text{s}}$	Pa s
-1000	-304.8	290.13	1050.4	1.2613	341.5	1.822×10^{-5}
0	0.0	288.15	1013.3	1.2250	340.3	1.812×10^{-5}
1000	304.8	286.17	977.2	1.1896	339.1	1.802×10^{-5}
2000	609.6	284.19	942.1	1.1549	337.9	1.792×10^{-5}
3000	914.4	282.21	908.1	1.1210	336.8	1.782×10^{-5}
4000	1219.2	280.23	875.1	1.0879	335.6	1.772×10^{-5}
5000	1524.0	278.24	843.1	1.0556	334.4	1.762×10^{-5}
6000	1828.8	276.26	812.0	1.0239	333.2	1.752×10^{-5}
7000	2133.6	274.28	781.9	0.9930	332.0	1.742×10^{-5}
8000	2438.4	272.30	752.6	0.9629	330.8	1.732×10^{-5}
9000	2743.2	270.32	724.3	0.9334	329.6	1.722×10^{-5}
10000	3048.0	268.34	696.8	0.9046	328.4	1.712×10^{-5}
11000	3352.8	266.36	670.2	0.8766	327.2	1.701×10^{-5}
12000	3657.6	264.38	644.4	0.8491	326.0	1.691×10^{-5}
13000	3962.4	262.39	619.4	0.8224	324.7	1.681×10^{-5}
14000	4267.2	260.41	595.2	0.7963	323.5	1.670×10^{-5}
15000	4572.0	258.43	571.8	0.7708	322.3	1.660×10^{-5}
16000	4876.8	256.45	549.2	0.7460	321.0	1.650×10^{-5}
17000	5181.6	254.47	527.2	0.7218	319.8	1.639×10^{-5}
18000	5486.4	252.49	506.0	0.6981	318.5	1.629×10^{-5}
19000	5791.2	250.51	485.5	0.6751	317.3	1.618×10^{-5}
20000	6096.0	248.53	465.6	0.6527	316.0	1.608×10^{-5}
21000	6400.8	246.55	446.5	0.6308	314.8	1.597×10^{-5}
22000	6705.6	244.56	427.9	0.6095	313.5	1.586×10^{-5}
23000	7010.4	242.58	410.0	0.5888	312.2	1.576×10^{-5}
24000	7315.2	240.60	392.7	0.5686	311.0	1.565×10^{-5}
25000	7620.0	238.62	376.0	0.5489	309.7	1.554×10^{-5}
26000	7924.8	236.64	359.9	0.5298	308.4	1.543×10^{-5}
27000	8229.6	234.66	344.3	0.5112	307.1	1.533×10^{-5}
28000	8534.4	232.68	329.3	0.4931	305.8	1.522×10^{-5}
29000	8839.2	230.70	314.9	0.4754	304.5	1.511×10^{-5}
30000	9144.0	228.71	300.9	0.4583	303.2	1.500×10^{-5}

How to include a PDF file

This appendix contains an included page from an external PDF file. Inspect the source code to see the syntax. More examples can be found in the comments. Uncomment and recompile to make them appear.

Do not forget to remove all explanations in blue from your final document. This can be achieved by uncommenting the line `\hideExplanations` in your main file.

page 3 of the

included dummy PDF file

Airfoil Coordinates Database (2020). UIUC Department of Aerospace Engineering. URL: m-selig.ae.illinois.edu/ads/coord_database.html (visited on 03/15/2020).

Cengel, Y., Turner, R., and Cimbala, J. (2016). *Fundamentals of Thermal-Fluid Sciences*. McGraw Hill. ISBN: 9781259151323.

Doumont, J. (2009). *Trees, Maps, and Theorems: Effective Communication for Rational Minds*. Principiæ. ISBN: 9789081367707. URL: books.google.be/books?id=O2dFPgAACAAJ.

Drews, A. (2021). *Documentation of the pfdicons package*. URL: https://ctan.tetaneutral.net/graphics/pgf/contrib/pfdicons/pfdicons-doc.pdf (visited on 01/04/2024).

Fauske, K. M. (2008). *Free body diagrams*. URL: www.texample.net/tikz/examples/free-body-diagrams/ (visited on 04/16/2020).

– (2020). *Control system principles*. URL: www.texample.net/tikz/examples/control-system-principles (visited on 03/15/2020).

Iowa Environmental Mesonet (2020). Iowa State University. URL: mesonet.agron.iastate.edu/request/download.phtml (visited on 10/20/2022).

Newell, D. and Tiesinga, E. (2019). *The International System of Units (SI), 2019 Edition*. explaining SI. DOI: 10.6028/NIST.SP.330-2019.

Redaelli, M. A., Lindner, S., Erhardt, S., and Giannetti, R. (2023). *Circuitikz manual*. URL: https://texdoc.org/serve/circuitikzmanual.pdf/0 (visited on 12/27/2023).

Rees, C. F. (2017). *Biblatex cheat sheet*. URL: tug.ctan.org/info/biblatex-cheatsheet/biblatex-cheatsheet.pdf (visited on 03/31/2021).

Tufte, E. (2001). *The Visual Display of Quantitative Information*. Graphics Press. ISBN: 978-1930824133. URL: books.google.be/books?id=qmjNngEACAAJ.

This demo bibliography shows the content of the file `bib/entrytypes.bib` which contains common entry types. It serves two purposes:

1. In the file `bib/entrytypes.bib`, you can inspect how these entry types are specified.

2. In this printed bibliography, you can see how they appear.

The entry type fields are filled with fictitious information, except for the `title` field, which contains an explanation of the use of that entry type.

Do not forget to remove this demo bibliography from your final document. You can do this by commenting out the line `\input{bib/entrytypes}` in your main file.

Black, S. (2020). "A title of an article from a journal, magazine, newspaper, or other periodicals". In: *Journal of Celestial Studies* 10.2. The content of this note field is visible in the bibliography, pp. 1–10. DOI: 10.1234/key.doi. URL: http://www.example.com/article.

Blue, A. (2011). "A title of a document that has not been formally published". The content of this note field is visible in the bibliography.

Brown, M. (2019). *A title of a book with an explicit publisher.* The content of this note field is visible in the bibliography. New York, NY: City Cruising Press. ISBN: 978-3-16-148410-0.

Cyan, M. (2018). "A title of a part of a book, usually with its own title. this could be a chapter or a specific section". In: *Advances in Bird Strike Assessment.* Ed. by M. Enta. The content of this note field is visible in the bibliography. Bird Publishing. Chap. 2, pp. 15–30.

Gray, C. (2016). "A title of a general type for theses and dissertations. use 'type' to specify the type of thesis (e.g., 'phdthesis' or 'mastersthesis')". The content of this note field is visible in the bibliography. PhD thesis. City, Country: Meteorological University.

Green, E. (2014). *A title of technical documentation.* The content of this note field is visible in the bibliography. Nature Publishing. City, Country.

Magenta, E. (May 12, 2010). "A title of a patent". Patent US123456789 (United States). N. of holder. The content of this note field is visible in the bibliography.

Orange, J. (2021). "A title of an article in a conference proceedings". In: *Proceedings of the Botany Conference.* Ed. by S. Rose. The content of this note field is visible in the bibliography. Botanical Society. San Francisco, CA, pp. 100–110.

Pink, M. (2015). *A title of a report published by a school or other institution, usually numbered within a series.* TR-123. The content of this note field is visible in the bibliography. City, Country: Lunar Research Institute.

Purple, H. (2017). "A title of a part of a book that is a collection of self-contained works, usually by multiple authors". In: *Collected Studies on Optical Phenomena.* Ed. by S. White. Series on Optical Science. The content of this note field is visible in the bibliography. Optics Press, pp. 45–60.

Red, T. (2013). *A title of a fallback type for entries that do not fit into any other category.* The content of this note field is visible in the bibliography.

Rose, S., ed. (2021). *A title of the proceedings of a conference.* The content of this note field is visible in the bibliography. Marine Biology Association. Seattle, WA: Marine Biology Press.

White, I. (2012). *A title of an online resource, such as a website.* The content of this note field is visible in the bibliography. URL: http://www.example.com (visited on 03/04/2012).

Yellow, S. (2022). *A title of a work that is printed and bound, but without a named publisher or sponsoring institution.* Distributed at the conference. The content of this note field is visible in the bibliography. Conference Location.

Todo list