

# Preambles in L<sup>A</sup>T<sub>E</sub>X

Anders B. Clausen

<https://github.com/anbclausen>

July 14, 2023

## Contents

<b>1</b>	<b>Setup</b>	<b>1</b>
<b>2</b>	<b>Preambles</b>	<b>2</b>
2.1	Articles and Presentations . . . . .	2
2.2	Default . . . . .	2
2.2.1	Common Text Commands . . . . .	2
2.2.2	Special Sections . . . . .	2
2.2.3	Colors . . . . .	3
2.2.4	Links and Email Addresses . . . . .	3
2.2.5	Citing and Bibliography . . . . .	3
2.2.6	Images . . . . .	3
2.2.7	Plots . . . . .	4
2.2.8	TikZ . . . . .	4
2.2.9	Math . . . . .	4
2.3	Handin . . . . .	4
2.3.1	Question Areas . . . . .	5
2.4	Code . . . . .	5
2.4.1	Code Blocks . . . . .	5
2.5	Logic . . . . .	5
2.5.1	LTL and CTL . . . . .	6
2.5.2	Similarity and Bisimilarity . . . . .	6
2.6	Protocols . . . . .	6
2.6.1	Protocol Diagrams . . . . .	7
2.6.2	Features . . . . .	8

## 1 Setup

Here is how to get going with an article:

```
1 \input{preambles/article}
2
3 \title{Your Title}
4 \author{Author}
5 \date{\today}
6
7 \begin{document}
8 \maketitle
9 ...
10 \end{document}
```

## 2 Preambles

### 2.1 Articles and Presentations

Specify the type of document you are writing by choosing a preamble. For writing articles, use the `article` preamble at the top of your document:

```
1 \input{preambles/article}
```

For writing presentations, use the `presentation` preamble at the top of your document:

```
1 \input{preambles/presentation}
```

### 2.2 Default

The preambles for articles and presentations offer many kinds of features. Here are all the ones that are enabled by default! If you do not want to use the article or presentation preamble, you can use the `default` preamble at the top of your document:

```
1 \input{preambles/default}
```

#### 2.2.1 Common Text Commands

	Usage	Example	Result
Italic Text	<code>\I{...}</code>	<code>\I{This is italic.}</code>	<i>This is italic.</i>
Bold Text	<code>\B{...}</code>	<code>\B{This is bold.}</code>	<b>This is bold.</b>
Underlined Text	<code>\U{...}</code>	<code>\U{This is underlined.}</code>	<u>This is underlined.</u>
Teletype (Monospace) Text	<code>\T{...}</code>	<code>\T{This is teletype.}</code>	<b>This is teletype.</b>
Striked Out Text	<code>\X{...}</code>	<code>\X{This is striked out.}</code>	<del>This is striked out.</del>

#### 2.2.2 Special Sections

	Usage	Example	Result
Todo Marker	<code>\todo</code>	<code>\todo</code>	<b>TODO</b>
Comment	<code>\comment{...}</code>	<code>\comment{This is a comment.}</code>	<b>Comment:</b> This is a comment.

### 2.2.3 Colors

	Usage	Example	Result
Colors from <code>xcolor</code> package	<code>\color{...}</code>	<code>\color{blue} I'm blue... Da ba dee da ba daa...</code>	I'm blue... Da ba dee da ba daa...

### 2.2.4 Links and Email Addresses

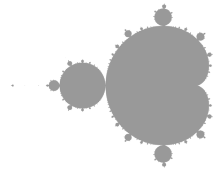
	Usage	Example	Result
Hyper References from <code>hyperref</code> package	<code>\href{...}{...}</code>	<code>\href{https://aekvi.com}{ aekvi}</code>	<a href="https://aekvi.com">aekvi</a>
Emails	<code>\email{...}{...}</code>	<code>\email{abc@aekvi.com}{abc}</code>	<a href="mailto:abc@aekvi.com">abc</a>

### 2.2.5 Citing and Bibliography

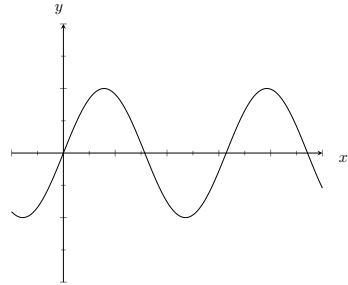
	Usage	Example	Result
Bibliography	<code>\bibliography {...}</code>	<code>\bibliography{refs}</code>	Renders bibliography assuming <code>refs.bib</code> file exists. See an example at the end of this document.
Cite	<code>\cite{...}</code>	<code>\cite{clausen}</code>	[1]

All citation features are from the `natbib` package. Learn more about it [here](#).

### 2.2.6 Images

	Usage	Example	Result
Images	<code>\img[scale]{...}</code>	<code>\img[0.2]{assets/mandelbrot_ set}</code>	
			assuming <code>assets/mandelbrot_set.png</code> exists. <code>scale</code> is optional.

### 2.2.7 Plots

	Usage	Example	Result
Plots	<pre>\plot[xmin]       [xmax]       [ymin]       [ymax]{f}</pre>	<pre>\plot[-2][10][-2][2]       {\sin(deg(x))}</pre>	 <p><code>xmin</code>, <code>xmax</code>, <code>ymin</code>, <code>ymax</code> are optional.</p>

### 2.2.8 TikZ

See OverLeafs tutorial on TikZ [here](#). Using TikZ libraries `arrows`, `shapes` and `automata` by default.

### 2.2.9 Math

	Usage	Example	Result
<code>mathbb</code> symbols	<pre>\Xb for   X ∈ A...Z</pre>	<code>\Ab</code> , <code>\Bb</code> , <code>\Cb</code> , <code>\Db</code>	$\mathbb{A}$ , $\mathbb{B}$ , $\mathbb{C}$ , $\mathbb{D}$
<code>mathcal</code> symbols	<pre>\Xc for   X ∈ A...Z</pre>	<code>\Ac</code> , <code>\Bc</code> , <code>\Cc</code> , <code>\Dc</code>	$\mathcal{A}$ , $\mathcal{B}$ , $\mathcal{C}$ , $\mathcal{D}$
XOR	<code>\$_{xor}\$</code>	<code>\$_{xor}\$</code>	$\oplus$
Boolean Equality	<code>\$_{beq}\$</code>	<code>\$_a \_{beq} b\$</code>	$a \stackrel{?}{=} b$
(a nicely behaving) Modulo	<code>\$_{mod}\$</code>	<code>\$_a \_{mod} b\$</code>	$a \bmod b$
Q.E.D.	<code>\QED</code>	<code>\QED</code>	$\square$

## 2.3 Handin

The handin preamble contains useful features for writing handins. Use the `handin` preamble at the top of your document:

```
1 \input{preambles/handin}
```

### 2.3.1 Question Areas

Question areas are used to highlight exercise descriptions and solutions. You can define a question like so:

```
1 \question{Prove  $SP = NP$ .}
```

This will render a question area like this:

**Question:** Prove  $P = NP$ .

## 2.4 Code

The code preamble contains useful features for writing code. Use the `code` preamble at the top of your document:

```
1 \input{preambles/code}
```

### 2.4.1 Code Blocks

The `listings` package is used to render code blocks. One can define a code block like so:

```
1 \begin{lstlisting}[language=Haskell]
2   main = putStrLn "Hello World!"
3 \end{lstlisting}
```

This will render a code block like this:

```
1 main = putStrLn "Hello World!"
```

Note that everything inside dollar symbols (\$) is rendered as math. This means that you can use math symbols in your code blocks. For example, the following code block:

```
1 \begin{lstlisting}
2 Area( $r$ ) =  $\pi r^2$ 
3 \end{lstlisting}
```

Will render like this:

```
1 Area( $r$ ) =  $\pi r^2$ 
```

## 2.5 Logic

The logic preamble contains useful features for writing LTL (Linear Temporal Logic) and CTL (Computation Tree Logic) amongst other things. Use the `logic` preamble at the top of your document:

```
1 \input{preambles/logic}
```

### 2.5.1 LTL and CTL

	Usage	Example	Result
Eventually	<code>\eventually</code>	<code>\eventually</code>	$\Diamond$
Always	<code>\always</code>	<code>\always</code>	$\Box$
Until	<code>\until</code>	<code>\until</code>	$U$
Weak Until	<code>\weakuntil</code>	<code>\weakuntil</code>	$W$
Release	<code>\release</code>	<code>\release</code>	$R$
Next	<code>\nex</code>	<code>\nex</code>	$\bigcirc$
True	<code>\true</code>	<code>\true</code>	true
False	<code>\false</code>	<code>\false</code>	false

### 2.5.2 Similarity and Bisimilarity

	Usage	Example	Result
Similarity	<code>\similar</code>	<code>\similar</code>	$\simeq$
Bisimilarity	<code>\bisimilar</code>	<code>\bisimilar</code>	$\sim$
Simulated By	<code>\simulatedby</code>	<code>\simulatedby</code>	$\sqsubseteq$

## 2.6 Protocols

The protocols preamble contains a useful macro for drawing protocols. Use the `protocols` preamble at the top of your document:

```
1 \input{preambles/protocols}
```

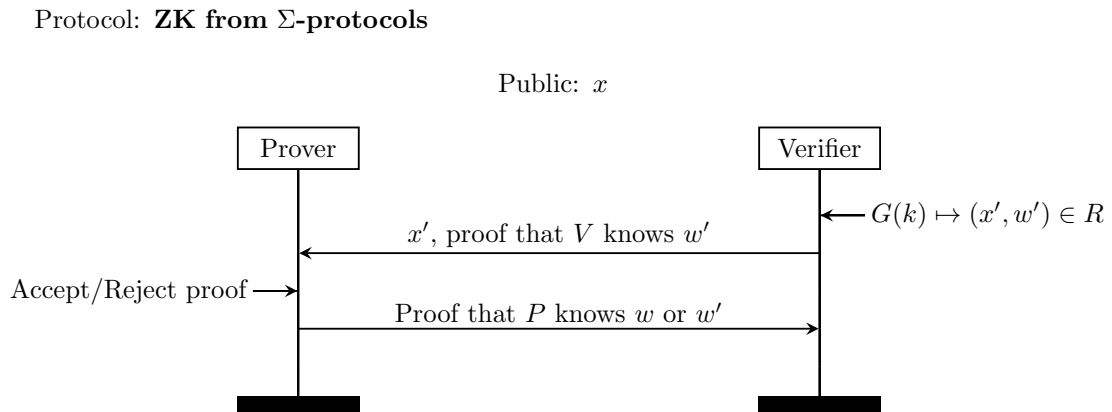
### 2.6.1 Protocol Diagrams

The `protocols` preamble contains a macro for drawing protocol diagrams. One can define a protocol diagram like so:

```

1 \protocol[ $x$ ]{ZK from  $\Sigma$ -protocols}{
2   \party{P}{Prover}
3   \party{V}{Verifier}
4
5   \knows{V}{ $G(k) \mapsto (x', w') \in R$ }
6   \msg{V}{ $x'$ , proof that  $V$  knows  $w'$ }{P}
7
8   \knows{P}{Accept/Reject proof}
9   \msg{P}{Proof that  $P$  knows  $w$  or  $w'$ }{V}
10 }
```

The result is a protocol diagram like this:



## 2.6.2 Features

	Usage	Example	Result
Protocol	<pre>\protocol   [public]   {name}   {body}    public is   optional.</pre>	<pre>\protocol[public]{name}   {\party{P}{Prover}    \party{V}{Verifier}}</pre>	See following.
Party (use in body)	<pre>\party   {symbol}   {name}</pre>	<pre>\party{P}{Prover}</pre>	See following.
Knows (use in body)	<pre>\knows   {symbol}   {message}   {side}</pre>	<pre>\knows{V}   {something private}   {left}</pre>	See following.
Message (use in body)	<pre>\msg{from}   {message}   {to}</pre>	<pre>\msg{V}{hello}{P}</pre>	See following.
Condition (use in body)	<pre>\cond   {message}   {parties}</pre>	<pre>\cond{Wait for P.}{P,V}</pre>	See following.

### Protocol example:

Protocol: **name**

Public: public

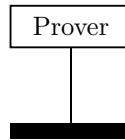


### Party example:



Protocol: **name**

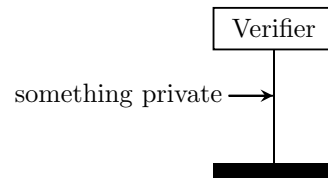
Public: public



**Knows example:**

Protocol: **name**

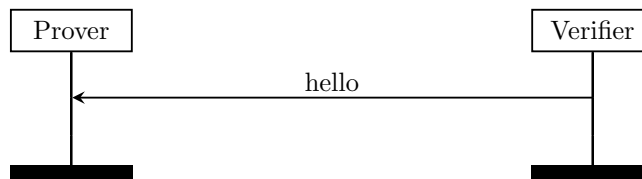
Public: public



**Message example:**

Protocol: **name**

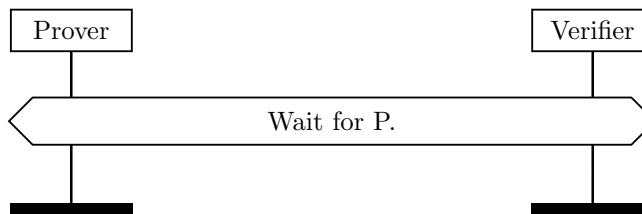
Public: public



**Condition example:**

Protocol: **name**

Public: public



## References

- [1] A. B. Clausen. *Preambles in L<sup>A</sup>T<sub>E</sub>X*, 2023.