The papermas package

H.-Martin Münch
<Martin.Muench at Uni-Bonn.de>

2023-04-12 v1.1a

Abstract

This LaTeX package allows to compute the number of sheets of paper needed to print a document as well as the mass of that printed version of the document, useful e. g. to determine the postage when sending it by snail mail.

Disclaimer for web links: The author is not responsible for any contents referred to in this work unless having full knowledge of illegal contents. If any damage occurs by the use of information presented there, only the author of the respective pages might be liable, not the one who has referred to those pages.

Contents

1 Introduction

This LaTeX package allows to compute the number of sheets of paper needed to print a document (useful when the paper is running out) as well as the mass of that printed version of the document, useful e. g. to determine the postage when sending it by snail mail.

Warning/Disclaimer: The mass of (printer's) ink has to be added and that of envelope, address sticker, stamps,... Thus this is only an estimation without guarantee – do not sue me, if you have got to pay excess postage!

The name papermas is short for paper mass but written with only one s, because historically some software had problems with names with more than eight letters.
It is mass and gives a result in grammes [g], because the weight $F = m \cdot g$ (really $\overrightarrow{F} = m \cdot \overrightarrow{g}$) [N] would require the knowledge of the gravitational acceleration g (depending on place and time, in central Europe approximately $9.81\,\mathrm{m/s^2}$) and give a result in Newton, which probably is not very useful.

2 Usage

Just load the package placing

```
\usepackage[<options>]{papermas}
```

in the preamble of your LaTeX 2_ε source file.
Now you can say

```
This document consists of $\papermassPages$~pages.
When printing $\papermaspagespersheet$~pages on one sheet of
paper, $\papermassheets$~sheets will be needed. For
ISO~A~\papermasformat{} paper of $\papermasmasss \unit{g}/\!\unit{m}^{2}$
specific mass, the printout will have a mass of about
$\papermasstotal \unit{g}$.
```

to get e. g.

> This document consists of 101 pages. When printing 4 pages on one sheet of paper, 26 sheets will be needed. For ISO A 4 paper of 80 g/m^2 specific mass, the printout will have a mass of about 130 g.

This information is also presented at the screen while compiling your document (look for papermas), in the `log` file (search for *** Paper mass ***), and can be found near the end of the `aux` file.

2.1 Options

`options` The papermas package takes the following options:

2.1.1 format

`format` The option `format` wants to know the ISO A... format of the paper used for printing, i. e. `format=4` means ISO A4 paper format (which is also the default).

2.1.2 masss

`masss` The option `masss` wants to know the specific (therefore the third `s`) mass of the paper used for printing in g/m^2. The default is `masss=80`, i. e. 80 g/m^2.

2.1.3 pagespersheet

`pagespersheet` The option `pagespersheet` wants to know, how many pages are to be printed on one sheet of paper. `pagespersheet=2` could mean duplex printing or printing two pages on one side of paper while keeping the back side blank. This does not influence the real printing process! So, if this number differs from the one chosen for printing, the result will be wrong, of course.

2.1.4 decimalsep

`decimalsep` The option `decimalsep` wants to know, what should be used for the decimal separator. In English this is ".", while in German it is ",". Enclose this in brackets, e. g. `decimalsep={.}` or `decimalsep={,}`. The default is ".". This is used for the mass of the printed document, and this value is given at the screen during compilation as well as in the `log` and `aux` files. Therefore something like `decimalsep={,\,}` would cause trouble there.

3 Alternatives

With the totpages package optionally the number of sheets of paper needed to print the document can be computed, but not the mass; https://ctan.org/pkg/totpages.

You programmed or found another alternative, which is available at https://www.CTAN.org/? OK, send an e-mail to me with the name, location at CTAN, and a short notice, and I will probably include it in the list above.

4 Example

```
⟨*example⟩
\NeedsTeXFormat{LaTeX2e}[2022-11-01]
\documentclass[british,a4paper]{article}[2022/07/02]% v1.4n Standard LaTeX document class
\pdfinfo{
  /Author (H.-Martin Muench)
  /Title (papermas package example)
  /Subject (Example for the papermas package)
  /Keywords (LaTeX;papermas;papermass;paper mass;paper;mass;weight)
}
\usepackage[format=4,masss=80,pagespersheet=2,decimalsep={.}]{papermas}[2023-04-12]
%% v1.1a Computes paper mass of a printout (HMM)
%% These are the default options.
\DeclareMathAlphabet{\mathup}{OT1}{\familydefault}{m}{n}
\DeclareRobustCommand{\unit}[1]{\ensuremath{\mathord{\thinspace\mathup{#1}}}}
\listfiles
\begin{document}
\pagenumbering{arabic}
\section*{Example for papermas}
\markboth{Example for papermas}{Example for papermas}

This example demonstrates the use of package\newline
\textsf{papermas}, v1.1a as of 2023-04-12 (HMM).\newline
The used options were \texttt{format=4} (ISO~A4),
\texttt{masss=80} ($\unit{g}\unit{m}^{-2}$), and\newline
\texttt{pagespersheet=2} (pages per sheet of paper,
i.\,e. either duplex printing or\newline
printing two pages on one side of a sheet of paper with blank back side).\newline
(These are the default options.)\newline
For more details please see the documentation!
\bigskip

This document consists of $\papermassPages$~pages.
When printing $\papermaspagespersheet$~pages on one sheet of paper,
$\papermassheets$~sheets will be needed. For ISO~A~\papermasformat{}
paper of $\papermasmasss \unit{g}\unit{m}^{-2}$ specific mass,
the printout will have a mass of about $\papermasstotal \unit{g}$.

\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage
\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage
\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage
\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage
\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage
\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage
\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage
```

```
\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage
\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage
\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage\newpage Page \thepage
\newpage Last page \thepage.
\end{document}
```
⟨/example⟩

5 The implementation

We start off by checking that we are loading into LaTeX 2ε and announcing the name and version of this package.

⟨*package⟩

```
\NeedsTeXFormat{LaTeX2e}[2022-11-01]
\ProvidesPackage{papermas}[2023-04-12 v1.1a Computes paper mass of a printout (HMM)]

```

A short description of the papermas package:

```
%% Allows to compute the number of sheets of paper
%% needed to print a document as well as the
%% mass of that printed version of the document,
%% useful e.g. to determine the postage when sending it by snail mail.
%% Warning/Disclaimer: Mass of (printer's) ink has to be added
%% and that of envelope, address sticker, stamps,...!
%% So, this is only an estimation without guarantee -
%% do not sue me, if you have got to pay excess postage!

```

`\NeedsTeXFormat{LaTeX2e}[2022-11-01]` was demanded. Check for it:

```
\providecommand\IfFormatAtLeastTF{\@ifl@t@r\fmtversion}

\IfFormatAtLeastTF{2022/11/01}{\relax}{%
  \PackageError{papermas}{%
      Newer LaTeX format needed or older papermas package%
    }{Needed LaTeX format version: 2022-11-01 or newer.\MessageBreak%
      Found\space\space LaTeX format version: \fmtversion.\MessageBreak%
      Either update your TeX distribution\MessageBreak%
      or use an archived version of papermas\MessageBreak%
      (see section History in the documentation).\MessageBreak%
     }
 }

```

For the handling of the options we need the kvoptions package and for `\intcalcPow` the intcalc package:

```
\RequirePackage{kvoptions}[2022-06-15]% v3.15 Key value format for package options  (HO)
\RequirePackage{intcalc}[2019/12/15]%   v1.3  Expandable calculations with integers (HO)

```

See subsection 6.1 about how to get them.

The options are introduced:

```
\SetupKeyvalOptions{family = papermas,prefix = papermas@}
\DeclareStringOption[4]{format}[4]%        paper format, ISO A..., default: (ISO A) 4
\DeclareStringOption[80]{masss}[80]%       specific mass of the paper, default: 80 (g/(m^2))
\DeclareStringOption[2]{pagespersheet}[2]% number of pages per sheet, for duplex printing this is 2.
\DeclareStringOption[.]{decimalsep}[.]%    decimal separator,
%%             e.g. "." or ",": decimalsep={,} - brackets are needed!!!
%%             decimalsep={,\,} does not work for screen, aux, log output!

\ProcessKeyvalOptions*

```

We pre-define some commands as (still) undefined:

```
\def\papermas@undefined{\textbf{??}}
```

`\papermas@undefined` will also be needed later!

```
\let\papermassPages\papermas@undefined
\let\papermasstotal\papermas@undefined
\let\papermasformat\papermas@undefined
\let\papermasmasss\papermas@undefined
\let\papermaspagespersheet\papermas@undefined
\let\papermassheets\papermas@undefined
\let\papermas@rerun\z@

```

`\papermas@totmass` This is the internal command, which computes the total paper mass of the printed document.

```
\DeclareRobustCommand\papermas@totmass{%
  \newcounter{papermasA}% paper mass for ISO A...
  \setcounter{papermasA}{\papermas@format}% e.g. 4
```

We check whether `papermasA` has a resonable value:

```
  \ifnum \value{papermasA}<0%
    \PackageError{papermas}{Option format has no valid value}%
     {The format option of the papermas package\MessageBreak%
      only takes whole, non-negative numbers (0, 1, 2, 3,...),\MessageBreak%
      because this should be the paper format\MessageBreak%
      ISO A 0, 1, 2, 3,...\MessageBreak%
      Found instead: \papermas@format \MessageBreak%
     }%
  \else%
```

`papermasA` has a resonable value. We introduce a new counter `papermasmasss` and initialize it with the value given in option `masss`, i. e. `\papermas@masss`.

```
    \newcounter{papermasmasss}%
    \setcounter{papermasmasss}{\papermas@masss}% default   80
```

Counters are integers, but the amount of the mass of a single sheet of paper in most cases is not an integer, therefore we multiply with 100 to get two digits behind the decimal separator.
(Later we need to divide by 100 again, of course.)

```
    \multiply \value{papermasmasss} 100 %        default 8000
```

We check whether `papermasmasss` has a resonable value, i. e. > 0:

```
    \ifnum \value{papermasmasss}<1%
      \PackageError{papermas}{Option masss has no valid value}%
       {The masss option of the papermas package\MessageBreak%
        only takes positive numbers,\MessageBreak%
        because this should be the mass per square meter\MessageBreak%
        of a single sheet of your paper.\MessageBreak%
        Found instead: \papermas@masss\MessageBreak%
       }%
    \else%
```

`masss` has a resonable value, and therefore also `\papermas@masss` and `papermasmasss`.
We check whether option `pagespersheet` has a resonable value, i. e. ≥ 1:

```
      \newcounter{papermasPPS}%
      \setcounter{papermasPPS}{\papermas@pagespersheet}% default 2
      \ifnum \value{papermasPPS}<1%
        \PackageError{papermas}{The number of pages per sheet must be positive.}{%
          You cannot print less than one TeX page per sheet of paper.\MessageBreak%
          The value found was \papermas@pagespersheet .\MessageBreak%
          }%
      \else%
```

`pagespersheet` has a resonable value, and therefore also `\papermas@pagespersheet` and `papermasTmpA`.
We introduce a new counter `papermas@sheets` for the number of sheets printed.
`\gdef\@abspage@last{...}` has not happened yet in first run.

```
        \newcounter{papermas@sheets}%
        \let\papermassPages\m@ne%
        \xdef\papermassmd{\number\maxdimen}%
        \ifnum \@abspage@last=\papermassmd\relax%
          \setcounter{papermas@sheets}{1}%
          \gdef\papermas@rerun{1}%
        \else%
          \setcounter{papermas@sheets}{\@abspage@last}%
          \xdef\papermassPages{\@abspage@last}%
        \fi%
```

When more than one page is printed on one sheet of paper, the number of sheets needed for printing is decreased:

```
        \divide \value{papermas@sheets} by \value{papermasPPS}%
```

`\divide` cuts off all digits behind the decimal separator, but if there are digits > 0, this means that there is an additional, last sheet, which is only partially covered with print (e. g. only one side of it for duplex printing an odd number of pages). In that case, we have to add one sheet of paper to the number of sheets needed.

```
\newcounter{papermas@tmpn}%
\setcounter{papermas@tmpn}{\arabic{papermas@sheets}}%
\multiply \value{papermas@tmpn} \value{papermasPPS}%
\ifnum \value{papermas@tmpn}=\papermassPages%
\else \addtocounter{papermas@sheets}{1}%
\fi%
```

Now we can multiply the specific mass of 100 sheets with the number of sheets needed for printing:

```
\multiply \value{papermasmasss} \value{papermas@sheets}%
% default:     8000              (no default for this)
```

The result is in $\mathrm{g\,m^{-2}}$.
A sheet with format ISO A0 has a size of $1\,\mathrm{m}^2$,
a sheet with format ISO A1 has a size of $1\,\mathrm{m}^2 \cdot 2^{-1}$,
a sheet with format ISO A2 has a size of $1\,\mathrm{m}^2 \cdot 2^{-2}$, ..., and
a sheet with format ISO An has a size of $1\,\mathrm{m}^2 \cdot 2^{-n}$.

Therefore we compute $2^{\value\{\mathrm{papermasA}\}}$ and divide the specific paper mass by that value:

```
\divide \value{papermasmasss} by \intcalcPow{2}{\value{papermasA}}%
% default:    16000           /              2^(\value{papermasA})
```

We need to get the division by 100 and the digits after the decimal separator right:

```
% for the example 297 is used
\newcounter{papermas@tmpm}%
\setcounter{papermas@tmpm}{\arabic{papermasmasss}}%   m:297 n:  0 o:0 p: 0 q:0
\setcounter{papermas@tmpn}{\arabic{papermasmasss}}%   m:291 n:291 o:0 p: 0 q:0
\divide \value{papermas@tmpn} by 100%                 m:297 n:  2 o:0 p: 0 q:0
\newcounter{papermas@tmpo}%
\setcounter{papermas@tmpo}{\arabic{papermas@tmpn}}%   m:291 n:  2 o:2 p: 0 q:0
\multiply \value{papermas@tmpn} 10%                   m:297 n: 20 o:2 p: 0 q:0
\divide \value{papermas@tmpm} by 10%                  m: 29 n: 20 o:2 p: 0 q:0
\newcounter{papermas@tmpp}%
\setcounter{papermas@tmpp}{\arabic{papermas@tmpm}}%
\addtocounter{papermas@tmpp}{-\arabic{papermas@tmpn}}%m: 29 n: 20 o:2 p: 9 q:0
%                  29        -  20 = 9
\multiply \value{papermas@tmpm} 10%                   m:290 n: 20 o:2 p: 9 q:0
\newcounter{papermas@tmpq}%
\setcounter{papermas@tmpq}{\arabic{papermasmasss}}%
\addtocounter{papermas@tmpq}{-\arabic{papermas@tmpm}}%m:290 n: 20 o:2 p: 9 q:7
%                 297        - 290 = 7
```

Now rounding mathematically correct, i. e. ≥ 0.5 becomes 1 (and remember a possible amount carried forward!) and < 0.5 becomes 0.

```
    \ifnum\value{papermas@tmpq}>4%
      \addtocounter{papermas@tmpp}{1}%                  m:290 n: 20 o:2 p:10 q:7
      \ifnum\value{papermas@tmpp}>9%                    m:290 n: 20 o:2 p:10 q:7
        \addtocounter{papermas@tmpo}{1}%                m:290 n: 20 o:3 p:10 q:7
        \setcounter{papermas@tmpp}{0}%                  m:290 n: 20 o:3 p: 0 q:7
      \fi%
    \fi%
    \ifx\papermassPages\m@ne\relax\let\papermassPages\papermas@undefined\fi%
```

The result in the example above is $297/100 = 2.97 \approx 3.0$. We write this into `\papermastmpr` (where `\papermas@decimalsep`) is the decimal separator) and the (other) options' values into temporary definitions, as well as the number of sheets:

```
    \edef\papermastmpr{\arabic{papermas@tmpo}\papermas@decimalsep\arabic{papermas@tmpp}}%
    \xdef\papermas@mbs{\arabic{papermas@tmpo}}%
    \edef\papermastmpformat{\papermas@format}%
    \edef\papermastmpmasss{\papermas@masss}%
    \edef\papermastmppagespersheet{\papermas@pagespersheet}%
    \edef\papermastmpt{\arabic{papermas@sheets}}%
```

We check for the counter `papermassttl`. If it exists, nothing is done, if it does not exist, it is declared as `\newcounter` (and by default set to zero).

```
    \@ifundefined{c@papermassttl}{\newcounter{papermassttl}}{\relax}%
```

If the `papermassttl` counter value already has the value of `papermasmasss`, everything is fine.

```
    \ifnum\value{papermassttl}=\value{papermasmasss}%
```

Otherwise we need another run of LaTeX.

```
    \else\gdef\papermas@rerun{1}%
    \fi%
```

In any case, we set the counter `papermassttl` to the current value of `papermasmasss`.

```
    \setcounter{papermassttl}{\arabic{papermasmasss}}%
```

Because we want to write out into the aux-file, we need the expanded value (as string) of `papermasmasss`:

```
    \edef\papermastmps{\arabic{papermasmasss}}%
```

If we are allowed to write into the aux-file, we do it here. If we are not allowed to do it, we give an error message.

```
    \if@filesw%
```

When it is read from the aux-file and when its content is processed, the counter `papermassttl` might not have been defined yet, in which case we define it.

```
      \immediate\write\@auxout{\string\@ifundefined{c@papermassttl}{\string\newcounter{papermassttl}}{\string\relax}}%
```

We set the counter `papermassttl` to the value `\papermastmps`, i. e. `\arabic{papermasmasss}`. In the next compilation run it will be checked, whether `\value{papermassttl}=\value{papermasmasss}` (see above).
If this is the case, everything is OK, no changes happened, and no rerun is necessary (at least not for papermas).

```
      \immediate\write\@auxout{\string\setcounter{papermassttl}{\papermastmps}}%
```

What we do need, is to get the determined `\papermastmpr` to the user. Therefore

1. we define `\papermasstotal` in the `aux`-file, where the user can look it up
2. we define `\papermasstotal`, so the user can e. g. write

```
This document consists of $\papermassPages$~pages.
When printing $\papermaspagespersheet$~pages on one sheet of
paper, $\papermassheets$~sheets will be needed. For
ISO~A~\papermasformat{} paper of $\papermasmasss \unit{g}/\!\unit{m}^{2}$
specific mass, the printout will have a mass of about
$\papermasstotal \unit{g}$.
```

```
193               \immediate\write\@auxout{\string\gdef\string\papermassPages{\papermassPages}}%
194               \immediate\write\@auxout{\string\gdef\string\papermasstotal{\papermastmpr}}%
195               \immediate\write\@auxout{\string\gdef\string\papermasformat{\papermastmpformat}}%
196               \immediate\write\@auxout{\string\gdef\string\papermasmasss{\papermastmpmasss}}%
197               \immediate\write\@auxout{\string\gdef\string\papermaspagespersheet{\papermastmppagespersheet}}%
```

3. we give at the screen the information about the `\papermasstotal`
4. which will also appear in the `log`-file.

We want to give also `\papermastmpt = \arabic{papermas@sheets}` to the user, i. e. the number of sheets needed to print the document. Therefore we follow the same procedure:

```
198           \immediate\write\@auxout{\string\gdef\string\papermassheets{\papermastmpt}}%
199         \else\PackageError{papermas}{No writing to auxiliary file allowed}{%
200            It was not allowed to write to an .aux file.\MessageBreak%
201            This package does not work without access to an .aux file.\MessageBreak%
202            But it is OK if the .aux file was already updated by previouse compiler runs\MessageBreak%
203            and would not have changed anyway. In that case just press Enter or Return\MessageBreak%
204            to continue the compilation.}%
205         \fi%
206       \fi%
207     \fi%
208   \fi%
209   }
210
```

For pretty printing the message of papermas three internal commands are needed, needing another counter:

```
211 \newcounter{papermassSpace}
212
```

`\papermas@log`

```
213 \DeclareRobustCommand{\papermas@log}[1]{%
```

```
  \ifnum#1>9%
    \addtocounter{papermassSpace}{1}%
    \papermas@log{\intcalcDiv{#1}{10}}%
  \fi%
  }

\papermas@spaces
\DeclareRobustCommand{\papermas@spaces}[2]{%
  \setcounter{papermassSpace}{1}%
  \papermas@log{#1}%
  \addtocounter{papermassSpace}{-#2}%
  \multiply\value{papermassSpace} -1%
  \papermas@space{\arabic{papermassSpace}}%
  \message{*^^J}%
  }

\papermas@space
\DeclareRobustCommand{\papermas@space}[1]{%
  \ifnum\value{papermassSpace}>0\relax\message{}\fi%
  \setcounter{papermassSpace}{#1}%
  \addtocounter{papermassSpace}{-1}%
  \ifnum\value{papermassSpace}>0\relax\papermas@space{\arabic{papermassSpace}}\fi%
  }
```

At `begindocument` it is checked whether some commands, which are/will be defined via the `aux`-file, are undefined yet. If this is the case, at the end a rerun warning is given.

```
\AddToHook{begindocument}{%
 \@bsphack%
  \ifx\papermassPages\papermas@undefined       \gdef\papermas@rerun{1}\fi%
  \ifx\papermasstotal\papermas@undefined       \gdef\papermas@rerun{1}\fi%
  \ifx\papermasformat\papermas@undefined       \gdef\papermas@rerun{1}\fi%
  \ifx\papermasmasss\papermas@undefined        \gdef\papermas@rerun{1}\fi%
  \ifx\papermaspagespersheet\papermas@undefined\gdef\papermas@rerun{1}\fi%
  \ifx\papermassheets\papermas@undefined       \gdef\papermas@rerun{1}\fi%
 \@esphack%
```

What we did not do yet, is to really *call* the command `\papermas@totmass`.
We do this after the last page, because we need the total number of pages.

```
 \AddToHook{enddocument}{%
  \AddToHook{enddocument/afterlastpage}{%
   \papermas@totmass%
```

After the final execution of the aux-file, everything should be defined. Otherwise a rerun is needed.

```
\AddToHook{enddocument/afteraux}{%
 \ifx\papermassPages\m@ne\relax\gdef\papermas@rerun{1}%
 \else\ifx\papermassPages\@abspage@last\else\gdef\papermas@rerun{1}\fi%
 \fi%
 \AddToHook{enddocument/info}[kernel/filelist]{%
```

The final \PackageInfo is given, either giving values or a rerun warning.

```
  \message{^^J}%
  \message{papermas: ******************** Paper mass ********************^^J}%
  \ifx\papermas@rerun\z@\relax%
    \edef\papermassETest{\the\ReadonlyShipoutCounter\thetotalpages\PreviousTotalPages\@abspage@last}%
    \edef\papermassEmpty{0001}%
    \ifx\papermassEmpty\papermassETest\relax%
      \PackageWarningNoLine{papermas}{Your document seems to consists of zero pages.\MessageBreak%
        When printing no pages, no paper will be needed.\MessageBreak%
        No paper has no mass.\MessageBreak%
        Maybe just rerun or fix any mistakes}%
      \gdef\papermassPages{0}%
      \gdef\papermaspagespersheet{0}%
      \gdef\papermasstotal{0}%
    \else%
      \ifnum \papermassPages=1\relax\message{papermas: * This document consists of 1 page.}%
                                    \papermas@spaces{1}{16}% 16 because page vs. pages
      \else \message{papermas: * This document consists of \papermassPages\space pages.}%
            \papermas@spaces{\papermassPages}{15}%
      \fi%
      \ifnum \papermaspagespersheet=1\relax \message{papermas: * When printing one page on one sheet of paper,}%
             \papermas@spaces{100}{6}%
      \else \message{papermas: * When printing \papermaspagespersheet\space pages on one sheet of paper,}%
            \papermas@spaces{\papermaspagespersheet}{5}%
      \fi%
      \ifnum \papermassheets=1\relax \message{papermas: * one sheet will be needed.}%
             \papermas@spaces{100}{26}%
      \else \message{papermas: * \papermassheets\space sheets will be needed.}%
            \papermas@spaces{\papermassheets}{25}%
      \fi%
      \message{papermas: * For ISO A \papermasformat\space paper of \papermasmasss\space g/m^2 specific mass,}%
        \papermas@spaces{\papermasmasss}{6}%
      \message{papermas: * the printout will have a mass of about \papermasstotal\space g.}%
        \papermas@spaces{\papermas@mbs}{4}%
    \fi%
   \else \PackageWarningNoLine{papermas}{Variables have changed.\MessageBreak Rerun to get them right}%
   \fi%
   \message{papermas: **************************************************^^J^^J}%
  }}}}%
```

\papermas@powerof The command \papermas@powerof is obsolete since papermas 2011/06/02 v1.0f. As there was ample time to remove it from documents, the replacement has been removed from the package in version 2023-04-12 v1.1a. If you really need to use it, here is the code:

```
\makeatletter%
\DeclareRobustCommand\papermas@powerof[2]{%
  \@ifundefined{c@papermas@result}{\newcounter{papermas@result}}{\relax}%
  \setcounter{papermas@result}{\intcalcPow{#1}{#2}}}%
\makeatother%
```

291 ⟨/package⟩

6 Installation

6.1 Downloads

Everything is available at https://ctan.org, but may need additional packages themselves.

papermas.dtx For unpacking the papermas.dtx file and constructing the documentation it is required:

- TEX-format LATEX 2ε 2022-11-01 or newer: https://www.CTAN.org/
- document class ltxdoc, 2022/06/22, v2.1i, https://ctan.org/pkg/ltxdoc
- package geometry, 2020/01/02, v5.9, https://ctan.org/pkg/geometry
- package holtxdoc, 2019/12/09, v0.30, https://ctan.org/pkg/holtxdoc

papermas.sty The papermas.sty for LATEX 2ε (i. e. each document using the papermas package) requires:

- TEX-format LATEX 2ε 2022-11-01 or newer, https://www.CTAN.org/
- package kvoptions, 2022-06-15, v3.15, https://ctan.org/pkg/kvoptions
- package intcalc, 2019/12/15, v1.3, https://ctan.org/pkg/intcalc
- package papermas, 2023-04-12, v1.1a, https://ctan.org/pkg/papermas (Because you are reading the documentation for the papermas package, it can be assumed that you already have some version of it – is it the current one?)

papermas-example.tex The papermas-example.tex requires the same files as all documents using the papermas package and additionally:

- class article, 2022/07/02, v1.4n, from classes: https://ctan.org/pkg/classes

totpages As possible alternative in section 3 there is listed

- package totpages, 2005/09/19, v2.00, https://ctan.org/pkg/totpages

Oberdiek
holtxdoc
kvoptions

All packages of the 'oberdiek' bundle (especially holtxdoc and kvoptions) are also available in a TDS compliant ZIP archive: https://mirror.ctan.org/install/macros/latex/contrib/oberdiek.tds.zip. It is probably best to download and use this, because the packages in there are quite probably both recent and compatible among themselves.

intcalc

The intcalc is available at https://www.ctan.org/pkg/intcalc, also as tds: https://mirrors.ctan.org/install/macros/latex/contrib/intcalc.tds.zip.

hyperref

hyperref is not included in that bundle and needs to be downloaded separately, https://mirror.ctan.org/install/macros/latex/contrib/hyperref.tds.zip.

Münch

A hyperlinked list of my (other) packages can be found at https://ctan.org/author/muench-hm.

6.2 Package, unpacking TDS

Package. This package is available on https://www.CTAN.org.

https://mirror.ctan.org/macros/latex/contrib/papermas/papermas.dtx The source file.

https://mirror.ctan.org/macros/latex/contrib/papermas/papermas.pdf The documentation.

https://mirror.ctan.org/macros/latex/contrib/papermas/papermas-example.pdf The compiled example file, as it should look like.

https://mirror.ctan.org/macros/latex/contrib/papermas/README The README file.

There is also a `papermas.tds.zip` available:

https://mirror.ctan.org/install/macros/latex/contrib/papermas.tds.zip Everything in TDS compliant, compiled format.

which additionally contains

papermas.ins	The installation file.
papermas.drv	The driver to generate the documentation.
papermas.sty	The `.style` file.
papermas-example.tex	The example file.

For required other packages please see the preceding subsection.

Unpacking. The `.dtx` file is a self-extracting docstrip archive. The files are extracted by running the `.dtx` through plain TEX:

```
tex papermas.dtx
```

About generating the documentation see paragraph 6.4 below.

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

```
papermas.sty          → tex/latex/papermas.sty
papermas.pdf          → doc/latex/papermas.pdf
papermas-example.tex → doc/latex/papermas-example.tex
papermas-example.pdf → doc/latex/papermas-example.pdf
papermas.dtx          → source/latex/papermas.dtx
```

If you have a `docstrip.cfg` that configures and enables docstrip's TDS installing feature, then some files can already be in the right place, see the documentation of docstrip.

6.3 Refresh file name databases

If your TEX distribution (TEX Live, MiKTEX, ...) relies on file name databases, you must refresh these. For example, TEX Live users run `texhash` or `mktexlsr`.

6.4 Some details for the interested

Unpacking with LATEX. The `.dtx` chooses its action depending on the format:

plain TEX: Run docstrip and extract the files.

LATEX: Generate the documentation.

If you insist on using LATEX for docstrip (really, docstrip does not need LATEX), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{papermas.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

Generating the documentation. You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by a configuration file `ltxdoc.cfg`. For instance, put the following line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfLATEX:

```
pdflatex papermas.dtx
makeindex -s gind.ist papermas.idx
pdflatex papermas.dtx
makeindex -s gind.ist papermas.idx
pdflatex papermas.dtx
```

6.5 Compiling the example

The example file, papermas-example.tex, can be compiled via `(pdf)latex papermas-example.tex` and will need at least two compiler runs to get everything right.

7 Acknowledgements

I would like to thank Heiko Oberdiek for providing a lot of useful packages (from which I also got everything I know about creating a file in `.dtx` format, OK, say it: copying).

8 History

[2010/06/01 v1.0(a)]

- First version of this papermas package.

[2010/06/03 v1.0b]

- New `\papermassheets` and reruncheck introduced; several small changes.
- Example adapted to other examples of mine.
- TDS locations updated.
- Several changes in the documentation and the Readme file.

[2010/06/24 v1.0c]

- holtxdoc warning in `drv` updated.
- Corrected the location of the package at CTAN. (TDS was still missing due to packaging error.)
- Several changes to the documentation.
- Introduced new **option**: `decimalsep`.

[2010/07/29 v1.0d]

- Corrected given url of `papermas.tds.zip` and other urls.
- Included a `\CheckSum` [was leter removed again].

[2011/02/01 v1.0e]

- Removed wrong % from the driver file.
- Changed the `\unit` definition (got rid of an old `\rm`).
- Replaced the list of my packages with a link to a web page list of those, which has the advantage of showing the recent versions of all those packages.
- Now using `\@ifundefined`.

- Removed `/muench/` from the path at diverse locations.
- Some small changes.

[2011/06/02 v1.0f]

- The holtxdoc package was fixed (recent version: 2011/02/04, v0.21), therefore the warning in `drv` could be removed. – Adapted the style of this documentation to new OBERDIEK `dtx` style.
- The rerun warnings are given after the `filelist` (if that is called with `\listfiles`) and the final papermas information is presented `\AtVeryVeryEnd` (now only ones instead of twice) [later replaced by use of the new hook management].
- Instead of compiling "a to the power of b" itself, papermas now uses the intcalc package.
- Removed five counters.

[2011/08/08 v1.0g]

- The pagesLTS package has been renamed to pageslts: 2011/08/08, v1.2a.
- Replaced `\global\edef` by `\xdef`.

[2011/08/22 v1.0h]

- Hot fix: TEX 2011-06-27 changed `\enddocument` and thus broke the `\AtVeryVeryEnd` command/hooking of atveryend package as of 2011-04-23, v1.7. Until fixed, `\AtEndAfterFileList` was used. [changed back in v1.0i]
- This version was archived at https://web.archive.org/web/20190417084752/https://mirror.ctan.org/install/macros/latex/contrib/papermas.tds.zip.

[----/--/-- v1.0i]

- Made `\newcommand`s robust.
- Moved from `\AtEndAfterFileList` to `\AtVeryVeryEnd` again. [later replaced by use of new hook management]
- By error this version was never released to the public.

[2023-04-12 v1.1a]

- `\NeedsTeXFormat{LaTeX2e}[2022-11-01]`, older package version available from archive (see version 2011/08/22 v1.0h above).
- Using the new hook management and LATEX's `\@abspage@last`.
- Package pageslts is neither needed nor used by the papermas package any longer.
- Removed `\papermas@powerof` (obsolete since version 2011-06-02 v1.0f).

- Converted to UTF-8.
- A lot of details; extensive documentation updates.

When you find a mistake or have a suggestion for an improvement of this package, please send an e-mail to the maintainer, thanks! (Please see BUG REPORTS in the README.)

9 Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; plain numbers refer to the code lines where the entry is used.