

# Subnumbering of equations\*

Donald Arsenau      Johannes Braams

2026/01/02

## 1 Introduction

Sometimes it is necessary to be able to refer to subexpressions of an equation. In order to do that these subexpressions should be numbered. In standard L<sup>A</sup>T<sub>E</sub>X there is no provision for this. To solve this problem Stephen Gildea once wrote `subeqn.sty` for L<sup>A</sup>T<sub>E</sub>X 2.09; Donald Arsenau rewrote the macros and Johannes Braams made them available for L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>.

Note that this package is *not* compatible with the package `subeqnarray`, written by Johannes Braams.

This package can be used together with the L<sup>A</sup>T<sub>E</sub>X options `leqno` and `fleqn`.

## 2 Available environments

`subequations` (*env.*) Inside the `subequations` environment L<sup>A</sup>T<sub>E</sub>X's equation environments such as `equation` and `eqnarray` are numbered as subexpressions. At the same time the number of the (main) equation is kept the same.

`subeqnarray` (*env.*) `\begin{subeqnarray}` works like `\begin{subequations}\begin{eqnarray}`, but saves typing. A `\label` command given at the very beginning of the first entry defines a `label` for the overall equation number, as if you had typed `\begin{subequations}\label{xxx}\begin{eqnarray}`.

## 3 Available commands

`\thesubequation` The command `\thesubequation` controls the labelling of the subexpressions of an equation. You can change the labelling by redefining this command, but the names of the counters may be confusing: The sub-number is given by counter `equation`, while the overall equation number is given by `mainequation`.

There are two ways to reference the overall equation number: through its value, as in `\Roman{mainequation}`, or through `\themainequation`, which gives the text of the normal `\theequation`. Refer to the local sub-number through the value of the `equation` counter, as in `\alph{equation}`. The default numbering is like 13c, given by:

```
\newcommand*{\thesubequation}{\themainequation\alph{equation}}
```

---

\*This file has version number v2.0d, last revised 2026/01/02.

Some alternatives:

A number such as 13.C is achieved by

```
\newcommand*\thesubequation{\thema inequation.\Alph{equation}}
```

A number such as 13-iii is achieved by

```
\newcommand*\thesubequation{\thema inequation-\roman{equation}}
\newcommand*\thesubequation{\thema inequation.\Alph{equation}}
```

When the document class which is used has declared

```
\renewcommand{\@eqnnum}{\theequation}
\renewcommand{\theequation}{(\arabic{equation})}
```

which puts parentheses around *all* equation numbers, including those produced by the `\ref` command, you can use:

```
\newcommand*\thesubequation{(\arabic{ma inequation}\alph{equation})}
```

## 4 The implementation

```
1 (*package)
```

`subequations` (*env.*) Within the `subequations` the equation numbers consist of two parts. The first part is a representation of the current value of the `equation` counter when the environment is entered, ie the number of the equation; the second part indicates the number of the subexpression of the equation.

```
2 \newenvironment{subequations}{%
```

First we update the `equation` counter,

```
3 \refstepcounter{equation}%
```

then we save its current value in `\c@ma inequation` and define `\thema inequation` to be the current representation of the `equation` counter.

```
4 \mathchardef\c@ma inequation\c@equation
```

```
5 \protected@edef\thema inequation{\theequation}%
```

Then we change the representation of the `equation` counter to represent the subexpression number. Finally we set the `equation` counter to zero as we use it for counting the subexpressions.

```
6 \let\theequation\thesubequation
```

```
7 \global\c@equation\z@
```

```
8 }{%
```

When the environment is finished we restore the value of the `equation` counter.

```
9 \global\c@equation\c@ma inequation
```

```
10 \global\@ignoretrue
```

```
11 }
```

`\thesubequation` By default the subexpressions will be numbered with lower case letters. The representation of the `equation` counter also includes the saved value of the `equation` counter. This can be changed by redefining this command.

```
12 \newcommand{\thesubequation}{\thema inequation\alph{equation}}
```

subeqnarray (*env.*)

```

13 \newenvironment{subeqnarray}{%
14   \subequations
15   \@ifnextchar\label{\@lab@subeqnarray}{\eqnarray}
16 }{%
17   \endeqnarray\endsubequations
18 }
```

`\@lab@subeqnarray` This macro picks up the `\label` command and its argument and re-inserts it *before* starting the `eqnarray` environment.

```

19 \newcommand*\@lab@subeqnarray}[2]{#1{#2}\eqnarray}
20 \</package>
```

## 5 An example of the use of this package

When you run the following document through L<sup>A</sup>T<sub>E</sub>X you will see the difference between the `subeqnarray` and `eqnarray` environments.

```

21 \<sample>
22 \documentclass{article}
23 \usepackage{subeqn}
24
25 \begin{document}
26 \title{Sample sub-equations}
27 \author{Johannes L. Braams}
28 \date{\today}
29 \maketitle
30
31 \noindent
32 This is an example of the use of the \texttt{subequations}
33 package. First we have a normal \textsf{equation} environment.
34 \begin{equation}
35   \label{a}
36   a^2 + b^2 = c^2
37 \end{equation}
38 Now we start sub-numbering.
39 \begin{subequations}
40   \label{b}
41   \begin{equation}
42     \label{b1}
43     d^2 + e^2 = f^2
44   \end{equation}
45   We can refer to equation~\ref{a}, \ref{b} and~\ref{b1}.
46   \begin{equation}
47     \label{b2}
48     g^2 + h^2 = i^2
49   \end{equation}
50   This was equation~\ref{b2}.
51   \begin{eqnarray}
52     \label{c}
53     x &=& y+z\label{c1}\\
54     u &=& v+w\label{c2}
55   \end{eqnarray}
```

```
56 This was expression~\ref{c}, consisting of parts~\ref{c1}
57 and~\ref{c2}.
58 \end{subequations}
59
60 \noindent
61 Now lets start a \textsf{subeqnarray} environment.
62 \begin{subeqnarray}
63   \label{d}
64   x &=& y+z\label{d1}\
65   u &=& v+w\label{d2}
66 \end{subeqnarray}
67 This was equation~\ref{d}, with parts~\ref{d1} and~\ref{d2}.
68 \end{document}
69 </sample>
```