

The trig package*

David Carlisle

1999/03/16

1 Introduction

These macros implement the trigonometric functions, sin, cos and tan. In each case two commands are defined. For instance the command `\CalculateSin{33}` may be issued at some point, and then anywhere later in the document, the command `\UseSin{33}` will return the decimal expansion of $\sin(33^\circ)$.

The arguments to these macros do not have to be whole numbers, although in the case of whole numbers, L^AT_EX or plain T_EX counters may be used. In T_EXBook syntax, arguments must be of type: *<optional signs><factor>*

Some other examples are:

`\CalculateSin{22.5}`, `\UseTan{\value{mycounter}}`, `\UseCos{\count@}`.

Note that unlike the psfig macros, these save all previously computed values. This could easily be changed, but I thought that in many applications one would want many instances of the same value. (eg rotating all the headings of a table by the *same* amount).

I don't really like this need to pre-calculate the values, I originally implemented `\UseSin` so that it automatically calculated the value if it was not pre-stored. This worked fine in testing, until I remembered why one needs these values. You want to be able to say `\dimen2=\UseSin{30}\dimen0`. Which means that `\UseSin` must *expand* to a *<factor>*.

2 The Macros

```
1 <*package>
\nin@ty Some useful constants for converting between degrees and radians.
\@clxx
\@lxxi
\@mmmlxviii
2 \chardef\nin@ty=90
3 \chardef\@clxx=180
4 \chardef\@lxxi=71
5 \mathchardef\@mmmlxviii=4068
```

The approximation to sin. I experimented with various approximations based on Tchebicheff polynomials, and also some approximations from a SIAM handbook 'Computer Approximations' However the standard Taylor series seems sufficiently accurate, and used by far the fewest T_EX tokens, as the coefficients are all rational.

$$\begin{aligned}\sin(x) &\simeq x - (1/3!)x^3 + (1/5!)x^5 - (1/7!)x^7 + (1/9!)x^9 \\ &\simeq \frac{(((7!/9!x^2 - 7!/7!)x^2 + 7!/5!)x^2 + 7!/3!)x^2 + 7!/1!)x}{7!} \\ &= \frac{(((1/72x^2 - 1)x^2 + 42)x^2 + 840)x^2 + 5040)x}{5040}\end{aligned}$$

*This file has version number v1.09, last revised 1999/03/16.

The nested form used above reduces the number of operations required. In order to further reduce the number of operations, and more importantly reduce the number of tokens used, we can precompute the coefficients. Note that we can not use $9!$ as the denominator as this would cause overflow of \TeX 's arithmetic.

```

\@coeffz Save the coefficients as \(\math)chars.
\@coeffa 6 \chardef\@coeffz=72
\@coeffb 7 %\chardef\@coefa=1
\@coeffc 8 \chardef\@coefb=42
\@coeffd 9 \mathchardef\@coefc=840
          10 \mathchardef\@coefd=5040

\TG@rem@pt The standard trick of getting a real number out of a  $\langle dimen \rangle$ . This gives a maximum accuracy of approx. 5 decimal places, which should be sufficient. It puts a space after the number, perhaps it shouldn't.
          11 {\catcode't=12\catcode'p=12\gdef\noPT#1pt{#1}}
          12 \def\TG@rem@pt#1{\expandafter\noPT\the#1\space}

\TG@term Compute one term of the above nested series. Multiply the previous sum by  $x^2$  (stored in \@tempb, then add the next coefficient, #1.
          13 \def\TG@term#1{%
          14   \dimen@\@tempb\dimen@
          15   \advance\dimen@ #1p@}

\TG@series Compute the above series. the value in degrees will be in \dimen@ before this is called.
          16 \def\TG@series{%
          17   \dimen@\@lxxi\dimen@
          18   \divide \dimen@ \@mmmlxviii

\dimen@ now contains the angle in radians, as a  $\langle dimen \rangle$ . We need to remove the units, so store the same value as a  $\langle factor \rangle$  in \@tempa.
          19 \edef\@tempa{\TG@rem@pt\dimen@}%

Now put  $x^2$  in \dimen@ and \@tempb.
          20 \dimen@\@tempa\dimen@
          21 \edef\@tempb{\TG@rem@pt\dimen@}%

The first coefficient is  $1/72$ .
          22 \divide\dimen@\@coeffz
          23 \advance\dimen@\m@ne\p@
          24 \TG@term\@coefb
          25 \TG@term{-\@coefc}%
          26 \TG@term\@coefd

Now the cubic in  $x^2$  is completed, so we need to multiply by  $x$  and divide by  $7!$ .
          27 \dimen@\@tempa\dimen@
          28 \divide\dimen@ \@coefd}

\CalculateSin If this angle has already been computed, do nothing, else store the angle, and call \TG@@sin.
          29 \def\CalculateSin#1{%
          30   \expandafter\ifx\csname sin(\number#1)\endcsname\relax
          31     \dimen@=#1p@\TG@@sin
          32     \expandafter\xdef\csname sin(\number#1)\endcsname
          33       {\TG@rem@pt\dimen@}%
          34   \fi}}

\CalculateCos As above, but use the relation  $\cos(x) = \sin(90 - x)$ .
          35 \def\CalculateCos#1{%
          36   \expandafter\ifx\csname cos(\number#1)\endcsname\relax
          37     \dimen@=\ninety\p@

```

```

38   \advance\dimen@-#1\p@
39   \TG@sin
40   \expandafter\xdef\csname cos(\number#1)\endcsname
41                                   {\TG@rem@pt\dimen@}%
42   \fi}}

\TG@reduce Repeatedly use one of the the relations  $\sin(x) = \sin(180 - x) = \sin(-180 - x)$ 
to get  $x$  in the range  $-90 \leq x \leq 90$ . Then call \TG@series.
43 \def\TG@reduce#1#2{%
44 \dimen@#1#2\nin@ty\p@
45   \advance\dimen@#2-\@clxx\p@
46   \dimen@-\dimen@
47   \TG@sin}

\TG@sin Slightly cryptic, but it seems to work...
48 \def\TG@sin{%
49   \ifdim\TG@reduce>+%
50   \else\ifdim\TG@reduce<-%
51   \else\TG@series\fi\fi}%

\UseSin Use a pre-computed value.
\UseCos 52 \def\UseSin#1{\csname sin(\number#1)\endcsname}
53 \def\UseCos#1{\csname cos(\number#1)\endcsname}

A few shortcuts to save space.
54 \chardef\z@num\z@
55 \expandafter\let\csname sin(0)\endcsname\z@num
56 \expandafter\let\csname cos(0)\endcsname\@ne
57 \expandafter\let\csname sin(90)\endcsname\@ne
58 \expandafter\let\csname cos(90)\endcsname\z@num
59 \expandafter\let\csname sin(-90)\endcsname\m@ne
60 \expandafter\let\csname cos(-90)\endcsname\z@num
61 \expandafter\let\csname sin(180)\endcsname\z@num
62 \expandafter\let\csname cos(180)\endcsname\m@ne

\CalculateTan Originally I coded the Taylor series for tan, but it seems to be more accurate to
just take the ratio of the sine and cosine. This is accurate to 4 decimal places
for angles up to  $50^\circ$ , after that the accuracy tails off, giving 57.47894 instead of
57.2900 for  $89^\circ$ .
63 \def\CalculateTan#1{%
64   \expandafter\ifx\csname tan(\number#1)\endcsname\relax
65     \CalculateSin{#1}%
66     \CalculateCos{#1}%
67     \@tempdima\UseCos{#1}\p@
68     \divide\@tempdima\@iv
69     \@tempdimb\UseSin{#1}\p@
70     \@tempdimb\two@fourteen\@tempdimb
71     \divide\@tempdimb\@tempdima
72     \expandafter\xdef\csname tan(\number#1)\endcsname
73                                   {\TG@rem@pt\@tempdimb}%
74   \fi}}

\UseTan Just like \UseSin.
75 \def\UseTan#1{\csname tan(\number#1)\endcsname}

\two@fourteen two constants needed to keep the division within TeX's range.
\@iv 76 \mathchardef\two@fourteen=16384
77 \chardef\@iv=4

```

Predefine $\tan(\pm 90)$ to be an error.

```

78 \expandafter\def\csname tan(90)\endcsname{\errmessage{Infinite tan !}}
79 \expandafter\let\csname tan(-90)\expandafter\endcsname
80                               \csname tan(90)\endcsname
81 \endpackage

```