© 1992 by Christian Franz. All rights reserved

Christian Franz

Hi-Performance Trigs

for Programmers

vers 1.0

```
x :  1.11385   sin(x) :  0.89740   FSin(x) :  0.89664 || Δ =  0.00075
x : -6.83590   sin(x) : -0.52500   FSin(x) : -0.52459 || Δ =  0.00041
x :  1.94773   sin(x) :  0.92979   FSin(x) :  0.93132 || Δ =  0.00152
x : -11.3244   sin(x) :  0.94639   FSin(x) :  0.94709 || Δ =  0.00069
x :  10.9891   sin(x) : -0.99997   FSin(x) : -0.99995 || Δ =  0.00002
x : -1.71395   sin(x) : -0.98977   FSin(x) : -0.99027 || Δ =  0.00050
x : -0.51351   sin(x) : -0.49124   FSin(x) : -0.49022 || Δ =  0.00101
x : -0.78333   sin(x) : -0.70564   FSin(x) : -0.70384 || Δ =  0.00180
x :  0.87740   sin(x) :  0.76908   FSin(x) :  0.76812 || Δ =  0.00096
x : -0.82378   sin(x) : -0.73371   FSin(x) : -0.73265 || Δ =  0.00106
x :  1.07238   sin(x) :  0.87834   FSin(x) :  0.87754 || Δ =  0.00079
x :  0.90463   sin(x) :  0.78619   FSin(x) :  0.78455 || Δ =  0.00163
x :  0.48897   sin(x) :  0.46972   FSin(x) :  0.46868 || Δ =  0.00103

Maximum Δ at  2.66487 with  0.00438

Mean Δ at       50 tries :  0.00107

TableResulution (Sine   ) was at     4096 Values for 2π.
```

© 1992 by Christian Franz. All rights reserved

Notice:

You may use this software and its documentation free of charge for any non-commercial use. This includes using it for writing public-domain or other *freeware* programs. If you use this software in your non-commercial programs you *must* include the line

"**uses Christian Franz Hi-Performance Trigs ©1992 by Christian Franz**"

in both the program's documentation and 'About...' dialog. *You must also write me an email or postcard telling me if you like the Trigs.*

Permission is granted to freely distribute this package and its accompanying documentation as long as neither is modified in any way and no fees are charged other than the usual downloading fees on commercial bulletin boards.

For commercial use of this software (for shareware programs and any other purpose) or its documentation you must contact me and have my written consent. Usually all I want in return is a free registered copy of your finished work.

My address is

Christian Franz
Sonneggstrasse 61
CH-8006 Zurich

Switzerland

email cfranz@iiic.ethz.ch
tel. +1-261 26 96 (+ = your code for Switzerland)

If you have any questions or bug reports or would like to see other features implemented, please feel free to contact me at above address.

Note: As you will notice throughout the documentation, English is not my primary language. There are bound to be many mistakes. If you find some, please take the time to write them down and

© 1992 by Christian Franz. All rights reserved

(e)mail them to me so I can correct them.

© 1992 by Christian Franz. All rights reserved

What it is:
Hi-Performance Trigs is a library for THINK Pascal and THINK C programmers. The library contains code for vary fast Sine, Cosine and Tangens functions. These are implemented via a look-up table, so no calculation is actually done. This is what makes the routines so fast.
Test results (full results are shown in Appendix A) show a dramatic Speed increase of a factor of almost 5 on a Mac SE. Note that although fast, the HiPerf Trigs cannot compete with the speed of a Math-Coprocessor (aka FPU). Code written specifically for Computers with FPUs are still twice as fast as the HiPerf Trigs. Code written for all machines using SANE to call the FPU however is slower than the HiPerf Trig Package.

Accuracy
Speed comes at a price, though. The price in this case is accuracy. The current version supports look-up tables of a maximum size of 32K (or 16384 look-up values, called resolution). This means that the best average accuracy would be ±0.00012207. In other words, using the 32K look-up table, you can expect the first three digits to be correct and the fourth to carry an error of about one. Note that this suffices for most real-time programs that don't require accuracy but speed. For choosing the correct accuracy, please refer to Appendix B.

Note also that due to the way the Sine and Cosine functions work, the error is usually much smaller. The Tangents however bear a significantly greater error close to $\pi/2$ and $3\pi/2$ (where a division by zero due to $\tan x = \sin x / \cos x$ occures).

Intended Useage:
HiPerf Trigs was originally written for my 3D GrafSys, a library similar to this for programmers to faciliate easy use of 3D graphics in their programs. These routines should be available at any good FTP site and are on the AMUG's BBS-in-a-Box CD.

The problem was that THINK's Pascal and C compilers where producing somewhat different code and C's notorious way of calling trigonometric functions made it very hard for programmers to use the libraries (that were written in Pascal). I thought about this and since I was not too overwhelmed with the performance of SANE, I decided to write my own very fast procedures. It worked out pretty well, since with 3D graphics you don't really need high accuracy but high speed.

© 1992 by Christian Franz. All rights reserved

After finishing the implementation, I realized that maybe a lot of people needed some high-speed trig functions and decided to publish these as well.

How to use HiPerf Trigs:
The usage is quite simple. Include the Tables of desired accuracy into the resource file for your project, Include the files HiPerfTrigs.lib and HiPerfTrigs.int into your projectfile and we are ready to roll.

The HiPerf Package comes with different look-up tables indicating their resolution. 'Tables 8096' would mean that this file contains the tables that have 8096 entries for 2π. Use Appendix B to find the table with the necessary resolution. Note that tablesize has no effect on speed, only on the memory it occupies. If you use the maximum resolution tables, expect them to occupy 96K (= 3 x 32K).

Since the HiPerf Package is self-configuring, you can mix tables of different resolution . Note, however, that you always *must* include all three tables, one for Sine, one for Cosine and one for Tangens for the package to work.

Note that you still can use any of the system's trigonometric functions while HiPerf Trig is loaded.

© 1992 by Christian Franz. All rights reserved

HiPerf Trigs Routines

```
function InitTrigs: OSErr;
```

Call this procedure before calling any of the other functions. InitTrig will load the look-up tables into memory and configure itself to the corresponding table resolutions.
If InitTrigs returns any other error than noErr, you must not use any of the fast trig functions.

```
procedure CloseTrigs;
```

When you are done with all trigonometric calculations, call this procedure to deallocate the look-up tables.

```
function GetSinResolution: integer;
```

This function simply returns the resolution (i.e. number of values in the table) for the Sine table.

```
function GetCosResolution: integer;
```

This function simply returns the resolution (i.e. number of values in the table) for the Cosine table.

```
function GetTanResolution: integer;
```

This function simply returns the resolution (i.e. number of values in the table) for the Tangens table.

```
function FSin (x: real): real;
```

This function returns the sine of x (in radiants). Please refer to Appendix B for the accuracy of the returned value.

© 1992 by Christian Franz. All rights reserved

```
function FCos (x: real): real;
```

This function returns the cosine of x (in radiants). Please refer to Appendix B for the accuracy of the returned value.

© 1992 by Christian Franz. All rights reserved

```
function FTan (x: real): real;
```

This function returns the tangens of x (in radiants). Please refer to Appendix B for the accuracy of the returned value.

© 1992 by Christian Franz. All rights reserved

Routines

```
const
        TableLoadErr = resNotFound;
        WrongVersionErr = -2;


function InitTrigs: OSErr;

procedure CloseTrigs;

function GetSinResolution: integer;

function GetCosResolution: integer;

function GetTanResolution: integer;

function FSin (x: real): real;

function FCos (x: real): real;

function FTan (x: real): real;
```

© 1992 by Christian Franz. All rights reserved

Appendix A
Performance Tests

The performance tests were done using he three supplied TimeTrig programs. Each program called 100'000 times the Sine, Cosine and Tangens functions and compared the results with the time it took the system to perform the same task.

A performance factor greater than one means that the HiPerf Trigs are faster by that factor.

Macintosh SE, System 7.1, no FPU, no extensions
(68000 Processor, 8 MHz)

Test Program: 'Test Trig plain'

Test	System	HiPerf Trigs	Performance Factor
Sin	89603	23191	3.86
Cos	81815	23186	3.53
Tan	105345	21704	4.85

Macintosh LC, System 7.1, no FPU, no extensions
(68020 Processor, 16 MHz)

Test Program: 'Test Trig plain'

Test	System	HiPerf Trigs	Performance Factor
Sin	17913	6312	2.84
Cos	15912	6324	2.52
Tan	19908	5935	3.35

Test Program: Test Trig 030'

Test	System	HiPerf Trigs	Performance Factor
Sin	17908	6111	2.93
Cos	15906	6079	2.62

© 1992 by Christian Franz. All rights reserved

Tan	19904	5731	3.48

© 1992 by Christian Franz. All rights reserved

Macintosh IIcx, System 7.1, FPU, no extensions
(68030 Processor, 16 MHz)

Test Program: 'Test Trig plain'

Test	System	HiPerf Trigs	Performance Factor
Sin	9174	4577	2.00
Cos	7679	4575	1.68
Tan	10545	4314	2.44

Test Program: 'Test Trig 030'

Test	System	HiPerf Trigs	Performance Factor
Sin	8859	4460	1.99
Cos	7428	4465	1.66
Tan	10161	4347	2.34

Test Program 'Test Trig 030/881'

Test	System	HiPerf Trigs	Performance Factor
Sin	176	315	0.56
Cos	176	325	0.54
Tan	204	323	0.63

© 1992 by Christian Franz. All rights reserved

Appendix B
Accuracy Table

Supplied with the package are 5 sets of tables. Generally speaking we have the rule that the larger the table, the better the accuracy. You may mix tables of different accuracies since the program is self-configuring.

Mean average error was calculated taking (always the same) 10000 random real numbers and passing them to the FSin, FCos and FTan routines. Then the same number was passed to the SANE routines and the difference was calculated and summed in a variable. After ten thousand tests, the sum was divided by ten thousand giving the average error.
The program 'CalcErr' supplied with this package does exactly this.

Resolution	Epsilon	Sin	Cos	Tan
1024	0.006136	0.00652	0.00708	0.01776
2048	0.003068	0.00276	0.00312	0.00793
4096	0.001534	0.00144	0.00162	0.00401
8192	0.000767	0.00084	0.00083	0.00227
16250	0.000387	0.00020	0.00024	0.00152

Resolution is number of entries in the table. Epsilon is the accuracy of the argument, i.e. only arguments passed to the trig functions that differ more than Epsilon from each other will generate different results. Epsilon is calculated by the formula 2π/Resolution.
The Sin, Cos and Tan values give the mean error for the functions at this resolution. This means you can expect an error of 0.00084 for a Sin call at a resolution of 8192.

© 1992 by Christian Franz. All rights reserved

Appendix C
Worst-Case Table

This appendix talks alot about some arcane things. Look at the supplied table to find the worst possible error. The remainder of this text just explains how the numbers were derived and have no importance for the programmer except if s/he wishes to use these functions for engineering software (which I don't recommend).

Note that the worst possible error is usually smaller than 4*Epsilon.

The following table contains the previous Accuracy Table plus together with the maximum Delta (error) found. Tests were done using an Epsilon of 0.0001 on the range of -2π..2π (this means 10000 tests on the range of 0..1 for a total of 125000 tests on the range of 4π).

Resolution	Epsilon	Sin	Cos	Tan
1024	0.006136	0.00652 Δ 0.02513	0.00708 Δ 0.01659	0.01776 Δ *
2048	0.003068	0.00276 Δ 0.01218	0.00312 Δ 0.00814	0.00793 Δ *
4096	0.001534	0.00144 Δ 0.00583	0.00162 Δ 0.00392	0.00401 Δ *
8192	0.000767	0.00084 Δ 0.00265	0.00083 Δ 0.00181	0.00227 Δ *
16250	0.000387	0.00020 Δ 0.00038	0.00024 Δ 0.00046	0.00152 Δ *

Note that there is no relieable maximum Delta for the Tangens function since the Tangens of kπ/2 (or k*90°) [k= 1,2,...] is not defined and approaces ±∞.
The entry preceeded by the Δ-sign contains the worst possible error at that resolution.