

**GoFast<sup>®</sup> for SPARC C Compilers**

**Features**

- "Link and Go" compiler support
- Conforms to IEEE 754
- Includes single and double precision
- Supports SPARClite designs
- Includes test programs and make files

**Description**

GOFAST<sup>®</sup> for SPARC is designed to allow programs developed for the SPARC architecture to access high performance floating point numerics. GOFAST for SPARC provides "link and go" compatibility with the Microtec C, SUN and GNU C SPARC compilers. GOFAST provides full IEEE and ANSI compatible floating point support.

**Floating Point Technology**

GOFAST for SPARC is based on U S Software's "Architecture Independent Technology" (AIT) and proven floating point algorithms the company has been developing for over decade. The algorithms have been thoroughly tested using automated methods. GOFAST was specifically optimized and integrated for high performance on SPARC processors.

GOFAST libraries are designed for "link and go" operation with each supported compiler.

**Processor Support**

The GOFAST for SPARC package includes different versions for different SPARC

- |  |   |
|--|---|
| <ul style="list-style-type: none"><li>• V8</li><li>• V7</li><li>• splite</li><br/><li>• fuj934</li></ul> | <p>SPARC version 8: floating-point unit, 32-bit divide</p> <p>SPARC version 7: no floating-point unit, 32-bit divide</p> <p>Fujitsu MB86930 (SPARClite): no floating point unit, 1-bit divide, scan instructions</p> <p>Fujitsu MB86934: floating point unit, 1-bit divide, scan instruction.</p> |
|--|---|

Each version also emulates floating-point instructions and 32-bit divide instructions. Two register interfaces are also supported: o0 and f0. The difference between these interfaces is in which registers are used for return values.

## **Functionality**

GOFAST supports C runtime floating point including the following single and double precision routines:

- + - \* /
- conversion operations
- sqrt, sin, cos, tan
- asin, acos, atan, atan2
- sinh, cosh, tanh
- log, log10, exp, pow
- floor, ceil, fabs
- modf, fmod, frexp, ldexp

Additional support includes test programs and make files.

## **Timings**

The following table gives approximate timings, in microseconds, for some emulated floating-point instructions. The times were measured on the 49 MHz Fujitsu SPARC lite evaluation board, using static RAM with cache both disabled and enabled.

Instruction	SRAM single cache		SRAM double cache	
branch, not taken	1.60	0.85	2.55	0.85
branch, taken	1.75	0.90	2.70	0.90
load	2.70	1.30	2.65	1.35
store	2.55	1.30	2.95	1.55
add	6.80	3.30	7.90	4.25
subtract	6.80	3.30	9.05	4.45
multiply	6.60	3.25	8.20	4.30
divide	8.35	4.75	12.10	6.10
compare	4.75	2.35	4.65	2.25
integer to float	4.40	2.10	4.50	2.25
float to integer	4.75	2.35	5.40	4.15
double to single	5.45	2.95		
single to double			4.15	2.30
square root	12.95	7.30	22.60	13.05

## **Compiler Support**

GOFAST for SPARC includes libraries for "link and go", seamless operation with the Microtec C, SUN, and GNU C compilers.

The GOFAST routines directly replace the compiler's floating point runtime library routines. In addition, routines normally not supported by the compiler are available with GOFAST. An

example might be the single precision version of the functions.

### **Capabilities and Accuracy**

GOFAST is reentrant. There is no need to perform any special state save/restore.  
GOFAST is ROMable. GOFAST is IEEE 754 and ANSI C compatible.

The basic functions + - \* / and the conversions perform the IEEE 754 "nearest or even" rounding exactly. The other functions are accurate within one mantissa unit.

GOFAST routines make no distinction between quiet and signaling Not-a-Numbers (NaNs). In the case of an invalid operation, the answer is always a standard quiet NaN.

GOFAST routines support IEEE 754 masked exception handling for overflows and invalid operations. No unmasked exceptions are supported. Underflow and loss of precision are not reported. Division by zero is treated as an invalid operation.