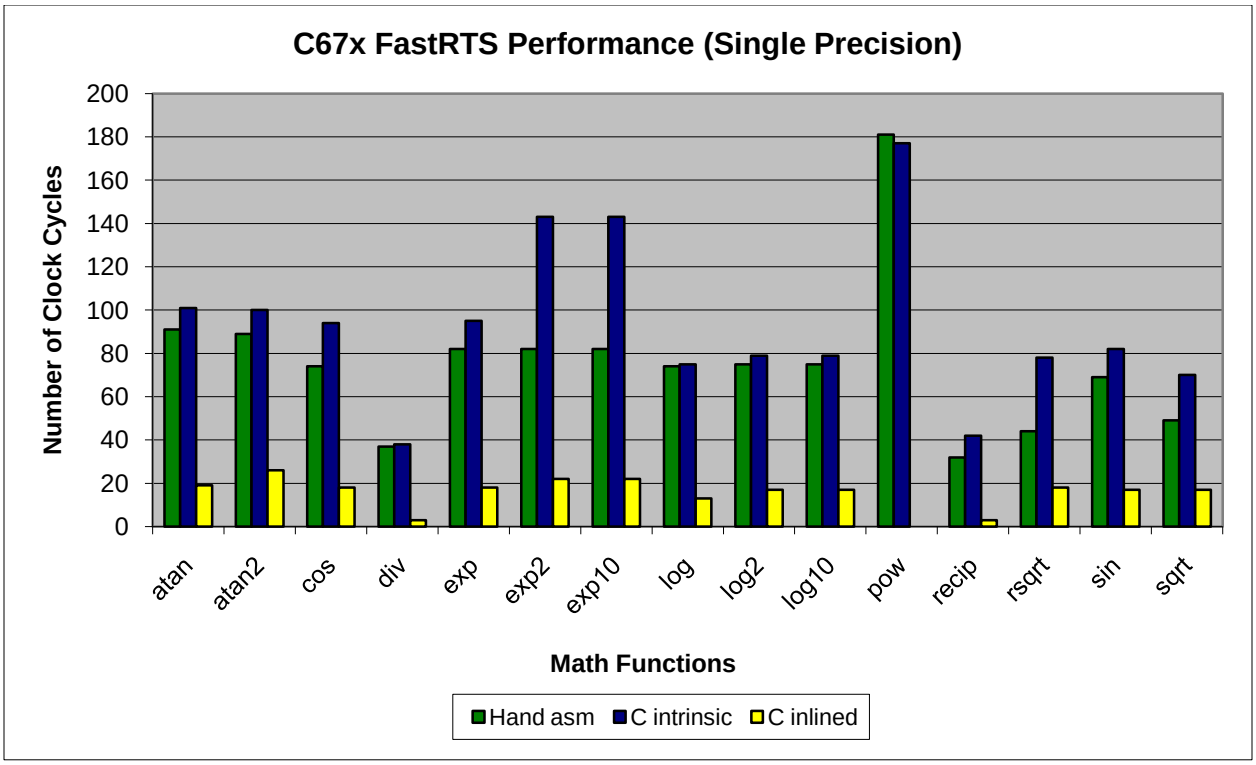
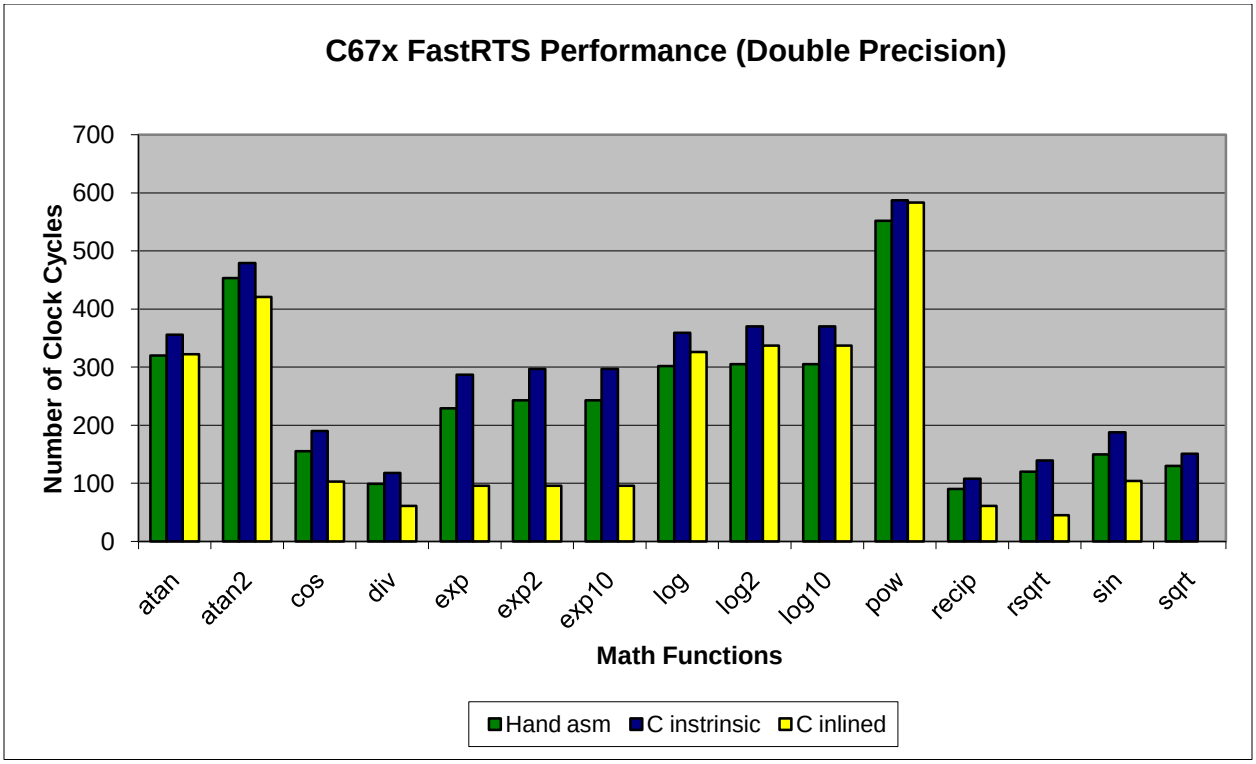
C67x FastRTS vs. RTS Benchmarking

Single Precision				
Function	RTS6740**	FastRTS	FastRTS (Intrinsics)	FastRTS (Inlining) Pipelining w/128 Calls
atan	651	91	*101*	19
atan2	1267	89	*100*	26
cos	167	74	*94*	18
div	540	37	*38*	3
exp	184	82	*95*	18
exp2	190	82	*143*	22
exp10	190	82	*143*	22
log	128	74	*75*	13
log2	153	75	*79*	17
log10	153	75	*79*	17
pow	937	181	*177*	82*
recip	542	32	*42*	3
rsqrt	165	44	*78*	18
sin	142	69	*82*	17
sqrt	161	49	*70*	17

Double Precision				
Function	RTS6740**	FastRTS	FastRTS (Intrinsics)	FastRTS (Inlining) Pipelining w/128 Calls
atan	1915	320	*356*	322
atan2	2016	453	*479*	421
cos	327	155	*190*	103
div	853	99	*118*	61
exp	1136	229	*287*	96
exp2	1083	243	*297*	96
exp10	1157	243	*297*	96
log	1871	302	*359*	326
log2	1887	305	*370*	337
log10	1891	305	*370*	337
pow	1570	552	*587*	583
recip	856	90	*108*	61
rsqrt	325	120	*139*	45
sin	309	150	*188*	104
sqrt	337	130	*151*	81*

Notes: The benchmarking was performed using the RTS, fastRTS, fastRTS w/intrinsics and fastRTS w/inlining of the different math functions. The inlining was performed to allow for pipelining in a loop iterated 128 times. The average clock cycles was then determined for the respective fastRTS (Inlining) column. The functions atan2 and pow require 2 arguments. *Special case modified to optimize. ** Average clocks measured may vary depending on the range of the data


C67x FastRTS Performance (Single Precision)
Number of Clock Cycles
200
180
160
140
120
100
80
60
40
20
0
atan
atan2
cos
div
exp
exp2
exp10
log
log2
log10
pow
recip
rsqrt
sin
sqrt
Math Functions
Hand asm
C intrinsic
C inlined


C67x FastRTS Performance (Double Precision)
Number of Clock Cycles
700
600
500
400
300
200
100
0
atan
atan2
cos
div
exp
exp2
exp10
log
log2
log10
pow
recip
rsqrt
sin
sqrt
Math Functions
Hand asm
C instrinsic
C inlined