Optimizing the Spherical Geometry Program

Clock Cycle Count per Point





Clock Cycle Count Ratios





Clock Cycle Count Ratios vs. Cortex-A72 (Arm32)





Clock Cycle Count Ratio: Cortex-A72 Arm32 to Arm64



Analysis Process

- Build, run, update spreadsheet
 - Review spreadsheet
- Gather performance data
 - sudo perf record ./sg
 - sudo perf annotate
 - P saves to text file <symbol_name>.annotation
- Use Ghidra to analyze sg executable
 - Examine top functions

Arm64 Performance: Why so good? And why not even better?

alex@raspberrypi: ~/AES-24/Speed/Scalar/SG2					
Samples: Overhead	2K of eve Command	nt 'cycles:P', Even Shared Object	t count (approx.): 1002378189 Symbol		
37.97%	sg	sg	<pre>[.] Find_Nearest_Waypoint</pre>		
21.01%	sg	sg	[.]cosf		
9.91%	sg	sg	[.] strcmp		
9.25%	sg	[kernel.kallsyms]	[k] el0_svc		
8.61%	sg	[kernel.kallsyms]	<pre>[k] _raw_spin_unlock_irgrestore</pre>		
5.50%	sg	[vdso]	<pre>[.]kernel_clock_gettime</pre>		
0.78%	sg	[kernel.kallsyms]	[k] _raw_spin_lock_irqsave		
0.52%	sg	[kernel.kallsyms]	<pre>[k]arm64_sys_clock_gettime</pre>		
0.48%	sg	sg	[.]atan2f		
0.45%	sg	[kernel.kallsyms]	<pre>[k] el0_svc_common.constprop.0</pre>		
0.41%	sg	sg	[.]sincosf		
0.41%	sg	sg	<pre>[.]clock_gettime</pre>		
0.41%	sq	[kernel.kallsyms]	[k] posix_cpu_clock_get		

alex@raspberrypi: ~/AES-24/Speed/Scalar/SG2

Samples:	13K of eve	ent 'cycles:P', Eve	nt count (approx.): 1024123918
overneau	Commanu	Shared Object	Symbol
	sg	sg	[.] Find_Nearest_Waypoint
	sg	sg	[.]cosf
	sg	sg	[.] strcmp
	sg	[kernel.kallsyms]	<pre>[k] _raw_spin_unlock_irqrestor</pre>
	sg	[kernel.kallsyms]	[k] el0_svc
4.59%	sg	[vdso]	<pre>[.]kernel_clock_gettime</pre>
0.56%	sg	sg	[.]atan2f
0.50%	sg	[kernel.kallsyms]	<pre>[k]arm64_sys_clock_gettime</pre>
0.46%	sg	[kernel.kallsyms]	[k] _raw_spin_lock_irqsave
0.44%	sg	sg	[.]sincosf
0.39%	sg	sg	<pre>[.]clock_gettime</pre>
0.39%	sg	sg	[.]ieee754_atan2f

Sanity check on results

- Find_Nearest_Waypoint OK, expected
- __cosf OK, expected
- strcmp OK, surprised it takes so much time
- el0_svc What is this? 9.25%
- _raw_spin_unlock_irqrestore What is this? 8.61%
- __kernel_clock_gettime Big! 5.50%
- P: Save this to perf.hist.N
- Rerun with higher sampling frequency to get more than 2K samples:
 - sudo perf record -F 20000 ./sg
 - get 13K samples
- Repeat the sanity check
 - Similar results, so probably ok

Nice Perf Annotate Features

P alex@	raspberrypi: ~/AES-24/Speed	d/Scalar/SG2
Samples	: 13K of event 'cyc	les:P', 20000 Hz, Event count (approx.): 1024123918
0.04	fmul s8, ldp s11 fdiv s8, while (strcmo)	s8, s11 , s10, [sp, #152] s8, s12 (waypoints[i].Name, "END")) {
	b d4 Calc_Closeness	s():
5.05	a8: UP/DOWN/PGUP PGDN/SPACE q/ESC/CTRL+C	Navigate Move to prev/next symbol Exit
3.12	 ENTER H TAB/shift+TAB j J	Go to target Go to hottest instruction Cycle thru hottest instructions Toggle showing jump to target arrows Toggle showing number of jump sources on targets Search post crime
1 36	0 0 S t c	Toggle disassembler output/simplified view Bump offset level (jump targets -> +call -> all -> cycle thru) Toggle source code view Circulate percent, total period, samples view Show min/max cycle Search string
1.50), k 1 P P P p	Toggle line numbers Show full source file location Print to [symbol_name].annotation file. Run available scripts Toggle percent type [local/global] Toggle percent base [period/hits]
84.36	? cc: f d4: Press any key	Search string backwards Toggle showing offsets to full address

 h: Get this help screen p: Toggle percentage between local (this function) and global (entire program) t: Cycle percent -> period -> # samples -> o, s, k: Toggle code listing 				
	P: Sav	ve to [symbol_nameJ.annotation file	
📔 C:\Us	ers\Alex\Docume	ents\Teaching\EC	E785\AES-2024\Modules\Speed Optimization\Optimizing SG\Find_Nearest_Waypo – 🛛 🗙	
File Edi	t Search View	Encoding L	anguage Settings Tools Macro Run Plugins Window ? + 🔻 🗙	
2 🗗 🛙	3 🖻 🗟 🕞	8 4 6 6) 🕽 C 📾 🎭 🔍 🔍 🖾 🖓 🔜 🖬 🖬 🐺 🥥 💹 🖾 🔊 🔍 🗉 🕑 🔤 🖛 👻	
Eind N	learest Waypoint	annotation 🕅		
1	Find Neares	t Waynoint	() /home/alex/AFS-24/Speed/Scalar/SG2/sg	
2	Event: cvcl	es:P	////inine/atex/Ac5-24/Speed/Scatal/Sd2/Sg	
3				
4	0.10	stp	x29, x30, [sp, #-160]!	
5		mov	x29, sp	
6	0.01	stp	x19, x20, [sp, #16]	
7	0.01	stp	x21, x22, [sp, #32]	
8	0.04	stp	x23, x24, [sp, #48]	
9	0.01	stp	x25, x26, [sp, #64]	
10	0.01	stp	x27, x28, [sp, #80]	
11	0.02	stp	d8, d9, [sp, #96]	
12	0.02	stp	d10, d11, [sp, #112]	
13	0.02	str	d12, [sp, #128]	
14		xpaclr		
15	0.02	fmov	s10, s0	
16		fmov	s8, s1	
17		mov	x26, x0	
18		movi	v9.2s, #0x0	
19		mov	x0, x30	
20		marr	V0E v1	

Try to Fill in Holes Between Samples

- Do these instructions *really* all complete in the same clock cycle?
- Maybe they don't but are missed by sampling
 - Program was sampled at 20 kHz, got 13,000 samples.
- Get more samples
 - Raise sampling frequency even more to 100 kHz
 - Increase program run time by raising NUM_TESTS to 1,000,000
- Now get 888,290 samples
- Eliminate possible truncation by display sample count, not percentage

님 Find_N	earest_Way	point.anno	tation 🔣	cosf.annotation 🗵
43	0.20		ldp	s11, s10, [sp, #152]
44			fdiv	s8, s8, s12
45		1	b	d4
46	1.99	a8:	ldur	s0, [x19, #-4]
47			fsub	s0, s0, s8
48		→	bl	cosf
49	1.23		ldp	s2, s1, [x19, #-12]
50			†mu1	s1, s11, s1
51			fmul	s0, s1, s0
52			fmadd	s0, s10, s2, s0
53	0.54		fcmpe	s9, s0
54		1	b.mi	194
55	33.26	cc:	add	w20, w20, #0x1
56			add	x19, x19, #0x28
57		d4:	mov	x1, x21
58			mov	x0, x19
59	0.08	→	bl	strcmp
60	0.06	î	cbnz	w0, a8
61			fmov	s0, s9
62			sbfiz	x19, x22, #2, #32
63			add	x19, x19, w22, sxtw
64		→	bl	acosf
65	0.03		mov	W0. #0x1800

Comparison: 13k vs. 888k samples

📄 Find_N	earest_Way	point.anno	tation 🔀	🔚 _cosf.annotation 🗵	🔚 Find_N	learest_Waypoint.an	notation	×	
43	0.20		ldp	s11, s10, [sp, #152]	43	765056		ldp	s11, s10, [sp, #152]
44			fdiv	s8, s8, s12	44			fdiv	s8, s8, s12
45		1	b	d4	45		1	b	d4
46	1.99	a8:	ldur	s0, [x19, #-4]	46	248473989	a8:	ldur	s0, [x19, #-4]
47			fsub	s0, s0, s8	47			fsub	s0, s0, s8
48		→	bl	cosf	48		→	bl	cosf
49	1.23		ldp	s2, s1, [x19, #-12]	49	141652362		ldp	s2, s1, [x19, #-12]
50			fmul	s1, s11, s1	50			fmul	s1, s11, s1
51			fmul	s0, s1, s0	51			fmul	s0, s1, s0
52			fmadd	s0, s10, s2, s0	52			fmadd	s0, s10, s2, s0
53	0.54		fcmpe	s9, s0	53	68382855		fcmpe	s9, s0
54		1	b.mi	194	54		1	b.mi	194
55	33.26	cc:	add	w20, w20, #0x1	55	4021122350	cc:	add	w20, w20, #0x1
56			add	x19, x19, #0x28	56			add	x19, x19, #0x28
57		d4:	mov	x1, x21 New	57	13359	d4:	mov	x1, x21
58			mov	x0, x19	58			mov	x0, x19
59	0.08	+	bl	strcmp	59	4093965	+	bl	strcmp
60	0.06	1	cbnz	w0, a8	60	3981251	1	cbnz	w0, a8
61			fmov	s0, s9 New	61	1299935		fmov	s0, s9
62			sbfiz	x19, x22, #2, #32	62			sbfiz	x19, x22, #2, #32
63			add	x19, x19, w22, sxtw	63			add	x19, x19, w22, sxtw
64		→	bl	acosf	64		→	bl	acosf
65	0.03		mov	w0, #0x1800	65	2891705		mov	w0, #0x1800

Latest Profile

Palex@raspberrypi: ~/AES-24/Speed/Scalar/SG2

Samples:	888K of e	vent 'cycles:P', Ev	ent count (approx.): 11898345696
Overhead	Command	Shared Object	Symbol
38.73%	sg	sg	[.] Find_Nearest_Waypoint
22.84%	sg	sg	[.]cosf
10.56%	sg	[kernel.kallsyms]	<pre>[k] _raw_spin_unlock_irqrestore</pre>
10.43%	sg	sg	[.] strcmp
7.27%	sg	[kernel.kallsyms]	[k] el0_svc
3.43%	sg	[vdso]	<pre>[.]kernel_clock_gettime</pre>
0.76%	sg	[kernel.kallsyms]	[k] _raw_spin_lock_irqsave
0.64%	sg	sg	[.]atan2f
0.56%	sg	[kernel.kallsyms]	<pre>[k]arm64_sys_clock_gettime</pre>
0.40%	sg	sg	[.]ieee754_atan2f
0.37%	sg	[kernel.kallsyms]	<pre>[k] posix_cpu_clock_get</pre>
0.36%	sg	[kernel.kallsyms]	[k] invoke_syscall
0.34%	sg	[kernel.kallsyms]	<pre>[k] put_timespec64</pre>
0.32%	sg	[kernel.kallsyms]	<pre>[k] el0_svc_common.constprop.0</pre>
0.29%	sg	[kernel.kallsyms]	[k] do_el0_svc
0.29%	sg	sg	[.]sincosf
0.27%	sg	sg	[.]atanf
0.27%	sg	[kernel.kallsyms]	[k]arch_copy_to_user

00400b00 - LAB_... 💅 🍷 🛄 📜

..0b00 mov w0.#0x43b40000

.0b04 fmov sl,w0

.. 0b0c b

.0b08 fadd s0.s0.s1

LAB 00400b00

LAB_00400ab4

Find Nearest Waypoint: 38.7% of Program's Time

void Find Nearest Waypoint(float cur pos lat, float cur pos lon, float * distance, float * bearing, 00400960 - Find_Nearest_Waypoint char * * name) { 1 · 🎕 🗖 🔲 00400a44 **3-**0 I undefined Find_Nearest_Waypoint() ..0a44 fmov s0,s9 // cur pos lat and cur pos lon are in degrees undefined w0:1 <RETURN> .0a48 sbfizx19,x22,#0x2,#0x20 undefined4 Stack[-0x8]:4 local_8 ... 0a4c add x19,x19,w22, SXTW Stack[-0x20]:8 local 20 undefined8 ... 0a50 bl acosf // distance is in kilometers undefined8 Stack[-0x30]:8 local_30 ..0a54 mov w0, #0x1800 Stack[-0x40]:8 local 40 undefined8 .. 0a58 movk w0, #0x45c7, LSL #16 Stack[-0x50]:8 local 50 // bearing is in degrees undefined8 00400a34 - LAB_00400a34 **1** - 🗆 I ... 0a5c add x19, x24, x19, LSL #0x3 Stack[-0x60]:8 local 60 undefined8 ... 0a60 fmov s1, s0 LAB 00400a34 Stack[-0x70]:8 local 70 undefined8 ...0a64 mov x1.x28 .0a34 mov x1=>DAT_00458e80,x21 undefined8 Stack[-0x80]:8 local 80 ... 0a68 fmov s2.w0 ... 0a38 mov x0=>s_ALBATROSS_BNK_00458e.. undefined8 Stack[-0x90]:8 local 90 ... 0a6c mov x0, x27 int i=0. closest i=0: .. 0a3c bl strcmp undefined8 Stack[-0xa0]:8 local a0 ..0a70 ldr s0.[x19. #0xc]=>DAT 00458e... ...0a40 cbnz w0,LAB_00400a08 Find Nearest Waypo0a74 fsub s0.s0.s8 PT T ref; ...0960 stp x29,x30,[sp, #local_a0]! ...0a78 fmul s8.s1.s2 ...0964 mov x29, sp 00400a08 - LAB_00400a08 **X-** 🗆 🖬 ...0968 stp x19,x20,[sp, #local_90] float d, b, c, max c=0, closest d=1E10; LAB 00400a08 ...0a80 ldp s1,s0, [x19, #0x4]=>DAT_004... ...096c stp x21,x22,[sp, #local_80] .0a08 ldur s0, [x19, #-0x4]=>DAT_00458... ...0a84 ldp s2,s3,[sp, #local 8] ...0970 stp x23,x24, [sp, #local 70] .. OaOc fsub s0, s0, s8 ... 0a88 fmul s10, s10, s0 ...0974 stp x25,x26,[sp, #local_60] .. Oal0 bl cosf .. 0a8c fmul s0.s3.s0 while (strcmp(wavpoints[i].Name, "END")) { ...0978 stp x27.x28.[sp. #local 50] ...0a90 fmul s10.s10.s2 ..0al4 ldp s2,s1,[x19, #-0xc]=>DAT_00... ...097c stp d8,d9,[sp, #local_40] .. 0al8 fmul sl.sll.sl .. 0a94 fnm.. s1, s11, s1, s10 c = Calc Closeness(&ref, &(waypoints[i])); ...0980 stp d10,d11,[sp, #local_30] ... 0a98 bl atan2f ... Oalc fmul s0, s1, s0 ..0984 str d12, [sp, #local_20] ...0a20 fmadd s0, s10, s2, s0 .. 0a9c mov w0, #0x2ee0 float Calc Closeness(PT T * p1, ...0988 xpa... ...0a24 fcmpes9,s0 ... 0aa0 movk w0, #0x4265, LSL #16 if (c>max c) { ..098c fmov s10, s0 ...0a28 b.mi LAB 00400af4 .0aa4 fmov sl,w0 ...0990 fmov s8,s1 .0aa8 fmul s0.s0.sl // calculates closeness (decreases $max_c = c;$...0994 mov x26.x0 .Oaac fcmpe s0. #0.0 ...0998 movi v9.25,#0x0 00400af4 - LA... 💅 🍷 🔲 📜 ...0ab0 b.mi LAB_00400b00 ...099c mov x0.x30 LAB 00400af4 closest i = i; ...09a0 mov x25,x1 .Oaf4 fmov s9,s0 return p1->SinLat * p2->SinLat + ...09a4 mov x23.x2 .. 0af8 mov w22, w20 } ...09a8 add x27.sp.#0x9c ...Oafc b LAB 00400a2c ...09ac adrp x24,0x49f000 p1->CosLat * p2->CosLat* ...09b0 ldr x24. [x24. #0xd50]=>->wavpo... i++; ...09b4 add x28.sp. #0x98 $cosf(p2 \rightarrow Lon - p1 \rightarrow Lon);$...09b8 adrp x21.0x458000 10400a2c - LAB... 💅 🍷 🔲 📜 } ...09bc add x21,x21,#0xe80 0400ab4 - LAB 00400ab4 2 · 🗆 🖬 ...09c0 mov w22.#0x0 LAB 00400a2c ...09c4 bl .0a2c add w20,w20,#0x1 mcount LAB 00400ab4 .. 09c8 mov w3, #0xfdb .0a30 add x19,x19,#0x28 .0ab4 mov w0.#0x28 .. 09cc movk w3, #0x4049, LSL #16 .0ab8 ldp x19,x20,[sp, #local_90] // Finish calcuations for the closest point ...09d0 mov w2,#0x43340000 .0abc sma... x22, w22, w0, x24 ...09d4 mov x0.x27 .. 0ac0 ldp x27, x28, [sp, #local 50] b = Calc_Bearing(&ref, &(waypoints[closest_i])); ...09d8 fmov sll,w3 ... 0ac4 add x22, x22, #0x10 ...09dc mov x1,x28 .. 0ac8 ldp d10, d11, [sp, #local_30] ..09e0 add x19.x24.#0x10 ... 0acc ldr d12, [sp, #local_20] // return information to calling function about closest ...09e4 fmov s12.w2 ... 0ad0 str s8, [x26] ..09e8 mov w20,#0x0 ... 0ad4 1dp d8.d9. [sp. #local 40] *distance = d; ...09ec fmul s0, s10, s11 ... 0ad8 str s0. [x25] ...09f0 fdiv s0, s0, s12 .. 0adc ldp x25, x26, [sp, #local 60] ...09f4 bl sincosf .. Oae0 str x22=>s_ALBATROSS_BNK_00458... *bearing = b; ...09f8 fmul s8.s8.s11 .. 0ae4 ldp x21,x22,[sp, #local 80] ...09fc ldp sll.sl0.[sp. #local 8] .. 0ae8 ldp x23,x24,[sp, #local 70] *name = (char *) (waypoints[closest_i].Name); ...0a00 fdiv s8.s8.s12 ..Oaec ldp x29=>local_a0,x30,[sp], #0... ...0a04 b LAB 00400a34 ... Oaf0 ret

Find_Nearest_Waypoint Loop

```
while (strcmp(waypoints[i].Name, "END")) {
  c = Calc_Closeness(&ref, &(waypoints[i]) );
  return p1->SinLat * p2->SinLat +
    p1->CosLat * p2->CosLat*
    cosf(p2->Lon - p1->Lon);
  if (c>max_c) {
    max_c = c;
    closest_i = i;
    }
    i++;
}
```

 What is going on here? Mark up hot instructions from annotation information



Find_Nearest_Waypoint Loop

```
while (strcmp(waypoints[i].Name, "END")) {
  c = Calc_Closeness(&ref, &(waypoints[i]) );
  return p1->SinLat * p2->SinLat +
    p1->CosLat * p2->CosLat*
    cosf(p2->Lon - p1->Lon);
  if (c>max_c) {
    max_c = c;
    closest_i = i;
    }
    i++;
}
```

 What is going on here? Mark up hot instructions from annotation information









