

# Re: Share Your Experience with 3DNow, SSE, SSE2 etc.

---

*Source:* <http://sci.tech-archive.net/Archive/sci.image.processing/2008-08/msg00055.html>

---

- *From:* aruzinsky <aruzinsky@xxxxxxxxxxxxxxxxxxxxxx>
  - *Date:* Thu, 7 Aug 2008 13:42:57 -0700 (PDT)
- 

On Aug 7, 7:17 am, Martin Brown <|||newspam...@xxxxxxxxxxxxxxxxxxxxxx> wrote:

aruzinsky wrote:

On Aug 5, 5:29 am, Martin Brown <|||newspam...@xxxxxxxxxxxxxxxxxxxxxx> wrote:

Did you establish experimentally that a loop unrolled by 4 was optimum?

I think so. Maybe, I copied it from a similar function that I optimized.

Worth checking. Cache behaviour can be rather quirky at times.

Tempting to suggest the usual peephole optimisations, and also to try out the shortest loop to see how well or badly that performs. eg.

```
$L2:
    movaps xmm0, [esi]
    movntpd [ebx+esi], xmm0
    add esi, 16
    dec ecx          ; or loop $L2
    jnz $L2
```

Re: Share Your Experience with 3DNow, SSE, SSE2 etc.

Registers used all different ebx contains old(eax-edx) etc.  
And then try loop unrolling.

But rather than optimise by trial and error, blindfolded why  
not enable  
the performance monitoring counters and do it properly by  
monitoring  
cache misses and stalls.

But, I still have to use trial and error to find the best code  
arrangement for prefetches because prefetches should be interleaved  
with computations.

If you are serious about coding for multiple CPUs with near optimality  
then the parameterised generated codelets approach used by FFTW and  
others is probably the way to go. See Bugbears post for details.

The Ring0 driver ia32.sys is at the University of Texas site  
(playing up  
at the moment) but I think Google cache still has a copy –  
including a  
new class library around it.

Try at your own peril but it can be very useful for tuning  
critical code.

<http://216.239.59.104/search?q=cache:H9fYu1GGy0EJ:iss.ices.utexas.edu...>

Thank you. I take it that you no longer have easy access to a C++  
compiler to do your own experiments?

More lack of time than anything else.  
I would only optimise at this sort of low level as a last resort. YMMV

Regards,  
Martin Brown

Re: Share Your Experience with 3DNow, SSE, SSE2 etc.

\*\* Posted from <http://www.teranews.com> \*\* - Hide quoted text -

- Show quoted text -

I checked and the loop unroll was unnecessary. I still get a 6% speed increase with prefetchnta.

I can get a 20% speed increase for conjugate gradients which my software, SAR Image Processor, often uses, but advertising SIMD would mostly serve as a sales gimmick.

To do better, I would have to use 2D float arrays with all rows beginning on 16 byte boundaries. Rewriting code for such arrays would be a major headache because my current matrix functions assume that rows are contiguous in RAM.

.