

Re: shift and failure to exceed compiler in speed

Source: <http://coding.derkeiler.com/Archive/Assembler/alt.lang.asm/2004-07/1160.html>

From: beta (*beta_at_s.l*)

Date: 07/27/04

Date: Tue, 27 Jul 2004 17:21:41 GMT

On 25 Jul 2004 17:38:04 -0700, 933191278@terra.es (Octavio) wrote:

>omega <beta@s.l> wrote in message news:<c6f2g0h3u5oseoq9pkl9a3oqrjf9o95m9a@4ax.com>...

>> I'd like to translate the shift of division "shift_u1()" from C[++] to

>> assembly [NASM]. Shift_u1() takes 3 args

>> 1) the result: a struct where there is the number [num_u]

>> 2) the number I have to shift: a struct num_u

>> 3) the shift: unsigned dword

>

>>this routine assumes that num size is multiple of 4 bytes

>>if num size is a fixed value or both numbers have the same size

>>the code can be smaller and faster.

>>i'm not a nasm or c++ programmer , and this code is not tested.

>

>pushad

>mov ebx,[esp+32+4] ;num_u*

>mov edx,[esp+32+8] ;const num_u*

>mov ecx,[esp+32+12] ;shift

>mov edi,[ebx+4] ;num

>mov ebx,[ebx] ;num size

>mov esi,[edx+4]

>mov edx,[edx]

>mov eax,ecx

>and ecx,7

>shr eax,3

>add esi,eax

>shr ebx,2

>sub edx,eax

>jnz zeros

>add edx,3

>shr edx,2

>||:

forse sono più chiaro in italiano: Mi sembra che

esi punta al vettore da shiftare di lunghezza edx

edi punta al vettore risultato di 'lunghezza' ebx

>lods

lodsd <=> { mov eax, [esi]; add esi, 4; }

```
>shld eax,[esi],cl
```

is it

```
eax = [esi]>>cl
```

?

se è così è un errore

```
>stosd
```

```
stosd <=> { mov [edi], eax; add edi, 4; }
```

```
>dec ebx
```

```
>jz end
```

```
>dec edx
```

```
>jnz ll
```

è

qui c'è un errore da come vedo io i numeri: struct

```
dword len
```

```
dword num ; pointer
```

```
>zeros:
```

```
>xor eax,eax
```

```
>mov ecx,ebx
```

```
>rep stosd
```

```
>end:
```

```
>popad
```

```
>ret
```