Limiting the bottom byte of an XMM register and clearing the other bytes

devblogs.microsoft.com/oldnewthing/20150112-00

January 12, 2015

Raymond Chen

Suppose you have a value in an XMM register and you want to limit the bottom byte to a particular value *and* set all the other bytes to zero. (Yes, I needed to do this.)

One way to do this is to apply the two steps in sequence:

```
; value to truncate/limit is in xmm0

; First, zero out the top 15 bytes
    pslldq  xmm0, 15
    psrldq  xmm0, 15

; Now limit the bottom byte to N
    mov     al, N
    movd    xmm1, eax
    pminub  xmm0, xmm1
```

But you can do it all in one step by realizing that min(*x*, 0) = 0 for all unsigned values *x*.

```
; value to truncate/limit is in xmm0
    mov     eax, N
    movd    xmm1, eax
    pminub  xmm0, xmm1
```

In pictures:

xmm0		xmm1		xmm0
?	min	0	=	0
?	min	0	=	0
?	min	0	=	0
?	min	0	=	0

?	min	0	=	0
?	min	0	=	0
?	min	0	=	0
?	min	0	=	0
?	min	0	=	0
?	min	0	=	0
?	min	0	=	0
?	min	0	=	0
?	min	0	=	0
?	min	0	=	0
?	min	0	=	0
x	min	*N*	=	min(*x*, *N*)

In intrinsics:

```
__m128i min_low_byte_and_set_upper_bytes_to_zero(__m128i x, uint8_t N)
{
 return _mm_min_epi8(x, _mm_cvtsi32_si128(N));
}
```

Raymond Chen

Follow