

64 Bit vs. 32 Bit

- ▶ 64 Bit Wortbreite
- ▶ 64 Bit Adressen
 - ▶ 2^{64} Byte \approx 16 EB Hauptspeicher adressierbar
- ▶ 64 Bit Register

32 Bit Register

eax	ax	
	ah	al
ecx	cx	
	ch	cl
edx	dx	
	dh	dl
ebx	bx	
	bh	bl
esp	sp	
ebp	bp	
esi	si	
edi	di	

64 Bit Register

rax	eax	ax	
		ah	al
rcx	ecx	cx	
		ch	cl
rdx	edx	dx	
		dh	dl
rbx	ebx	bx	
		bh	bl
rsp	esp	sp	spl
rbp	ebp	bp	bpl
rsi	esi	si	sil
rdi	edi	di	dil

r8	r8d	r8w	r8b
r9	r9d	r9w	r9b
r10	r10d	r10w	r10b
r11	r11d	r11w	r11b
r12	r12d	r12w	r12b
r13	r13d	r13w	r13b
r14	r14d	r14w	r14b
r15	r15d	r15w	r15b

Immediate-Operanden

- ▶ x86-32: Alle immediates 32 Bit, also im Wertebereich $[-2^{31}, 2^{31} - 1]$
- ▶ x86-64: 64 Bit immediates nur bei `mov`-Instruktionen erlaubt
32 Bit immediates sign-extended

<code>mov rax, 0xaaaabbbbccccdddd</code>	erlaubt
<code>add rax, 0xaaaabbbbccccdddd</code>	nicht erlaubt

Quiz

Wie kann man `rax = rax + 0xffffffff` berechnen?

☐

`add rax, 0xffffffff`

☐

`mov rcx, 0xffffffff`
`add rax, rcx`

☐

`mov ecx, -1`
`add rax, rcx`