# Priedas NR.1 – x86-64+ operacijų kodai

Oranžine spalva pažymėti operacijų kodai yra įgyvendinti VM operacinėje sistemoje. Yra laikoma kad RM operacinė sistema palaiko visus RM registrus.

| Mnemonika | op1 | op2 | op3 | op4 | I.R. | pre-fix | 0F | OPK | OPK II | flds | o | x | Grupė1 | Grupė2 | Grupė3 | Pakeisti FLAGS | Apibr. FLAGS | Aprašymas |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ADC | Eb | Gb | | | | | | 10 | | dw | r | L | gen | arith | binary | o..szapc | o..szapc | Add with Carry |
| ADC | Evqp | Gvqp | | | | | | 11 | | dW | r | L | gen | arith | binary | o..szapc | o..szapc | Add with Carry |
| ADC | Gb | Eb | | | | | | 12 | | Dw | r | | gen | arith | binary | o..szapc | o..szapc | Add with Carry |
| ADC | Gvqp | Evqp | | | | | | 13 | | DW | r | | gen | arith | binary | o..szapc | o..szapc | Add with Carry |
| ADC | AL | Ib | | | | | | 14 | | w | | | gen | arith | binary | o..szapc | o..szapc | Add with Carry |
| ADC | rAX | Ivds | | | | | | 15 | | W | | | gen | arith | binary | o..szapc | o..szapc | Add with Carry |
| ADC | Eb | Ib | | | | | | 80 | | w | 2 | L | gen | arith | binary | o..szapc | o..szapc | Add with Carry |
| ADC | Evqp | Ivds | | | | | | 81 | | W | 2 | L | gen | arith | binary | o..szapc | o..szapc | Add with Carry |
| ADC | Evqp | Ibs | | | | | | 83 | | SW | 2 | L | gen | arith | binary | o..szapc | o..szapc | Add with Carry |
| ADD | Eb | Gb | | | | | | 00 | | dw | r | L | gen | arith | binary | o..szapc | o..szapc | Add |
| ADD | Evqp | Gvqp | | | | | | 01 | | dW | r | L | gen | arith | binary | o..szapc | o..szapc | Add |
| ADD | Gb | Eb | | | | | | 02 | | Dw | r | | gen | arith | binary | o..szapc | o..szapc | Add |
| ADD | Gvqp | Evqp | | | | | | 03 | | DW | r | | gen | arith | binary | o..szapc | o..szapc | Add |
| ADD | AL | Ib | | | | | | 04 | | w | | | gen | arith | binary | o..szapc | o..szapc | Add |
| ADD | rAX | Ivds | | | | | | 05 | | W | | | gen | arith | binary | o..szapc | o..szapc | Add |
| ADD | Eb | Ib | | | | | | 80 | | w | 0 | L | gen | arith | binary | o..szapc | o..szapc | Add |
| ADD | Evqp | Ivds | | | | | | 81 | | W | 0 | L | gen | arith | binary | o..szapc | o..szapc | Add |
| ADD | Evqp | Ibs | | | | | | 83 | | SW | 0 | L | gen | arith | binary | o..szapc | o..szapc | Add |
| ADDPD | Vpd | Wpd | | | sse2 | 66 | 0F | 58 | | | r | | pcksclr | arith | | | | Add Packed Double-FP Values |
| ADDPS | Vps | Wps | | | sse1 | | 0F | 58 | | | r | | simdfp | arith | | | | Add Packed Single-FP Values |
| ADDSD | Vsd | Wsd | | | sse2 | F2 | 0F | 58 | | | r | | pcksclr | arith | | | | Add Scalar Double-FP Values |
| ADDSS | Vss | Wss | | | sse1 | F3 | 0F | 58 | | | r | | simdfp | arith | | | | Add Scalar Single-FP Values |
| ADDSUBPD | Vpd | Wpd | | | sse3 | 66 | 0F | D0 | | | r | | simdfp | arith | | | | Packed Double-FP Add/Subtract |
| ADDSUBPS | Vps | Wps | | | sse3 | F2 | 0F | D0 | | | r | | simdfp | arith | | | | Packed Single-FP Add/Subtract |
| AND | Eb | Gb | | | | | | 20 | | dw | r | L | gen | logical | | o..szapc | o..sz.pc | Logical AND |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AND | Evqp | Gvqp | | | | | | 21 | | d W | r | L | gen | logical | | o..szapc | o..sz.pc | Logical AND |
| AND | Gb | Eb | | | | | | 22 | | D w | r | | gen | logical | | o..szapc | o..sz.pc | Logical AND |
| AND | Gvqp | Evqp | | | | | | 23 | | D W | r | | gen | logical | | o..szapc | o..sz.pc | Logical AND |
| AND | AL | Ib | | | | | | 24 | | w | | | gen | logical | | o..szapc | o..sz.pc | Logical AND |
| AND | rAX | Ivds | | | | | | 25 | | W | | | gen | logical | | o..szapc | o..sz.pc | Logical AND |
| AND | Eb | Ib | | | | | | 80 | | w | 4 | L | gen | logical | | o..szapc | o..sz.pc | Logical AND |
| AND | Evqp | Ivds | | | | | | 81 | | W | 4 | L | gen | logical | | o..szapc | o..sz.pc | Logical AND |
| AND | Evqp | Ibs | | | | | | 83 | | S W | 4 | L | gen | logical | | o..szapc | o..sz.pc | Logical AND |
| ANDNPD | Vpd | Wpd | | | sse2 | 66 | 0F | 55 | | | r | | pcksclr | logical | | | | Bitwise Logical AND NOT of Packed Double-FP Values |
| ANDNPS | Vps | Wps | | | sse1 | | 0F | 55 | | | r | | simdfp | logical | | | | Bitwise Logical AND NOT of Packed Single-FP Values |
| ANDPD | Vpd | Wpd | | | sse2 | 66 | 0F | 54 | | | r | | pcksclr | logical | | | | Bitwise Logical AND of Packed Double-FP Values |
| ANDPS | Vps | Wps | | | sse1 | | 0F | 54 | | | r | | simdfp | logical | | | | Bitwise Logical AND of Packed Single-FP Values |
| BLENDPD | Vpd | Wpd | Ib | | sse41 | 66 | 0F | 3A | 0D | | r | | simdfp | datamov | | | | Blend Packed Double-FP Values |
| BLENDPS | Vps | Wps | Ib | | sse41 | 66 | 0F | 3A | 0C | | r | | simdfp | datamov | | | | Blend Packed Single-FP Values |
| BSF | Gvqp | Evqp | | | | | 0F | BC | | | r | | gen | bit | | o..szapc | ....z... | Bit Scan Forward |
| BSR | Gvqp | Evqp | | | | | 0F | BD | | | r | | gen | bit | | o..szapc | ....z... | Bit Scan Reverse |
| BSWAP | Zvqp | | | | | | 0F | C8 | | +r | | | gen | datamov | | | | Byte Swap |
| BT | Evqp | Gvqp | | | | | 0F | A3 | | | r | | gen | bit | | o..szapc | .....c | Bit Test |
| BT | Evqp | Ib | | | | | 0F | BA | | | 4 | | gen | bit | | o..szapc | .....c | Bit Test |
| BTC | Evqp | Ib | | | | | 0F | BA | | | 7 | L | gen | bit | | o..szapc | .....c | Bit Test and Complement |
| BTC | Evqp | Gvqp | | | | | 0F | BB | | | r | L | gen | bit | | o..szapc | .....c | Bit Test and Complement |
| BTR | Evqp | Gvqp | | | | | 0F | B3 | | | r | L | gen | bit | | o..szapc | .....c | Bit Test and Reset |
| BTR | Evqp | Ib | | | | | 0F | BA | | | 6 | L | gen | bit | | o..szapc | .....c | Bit Test and Reset |
| BTS | Evqp | Gvqp | | | | | 0F | AB | | | r | L | gen | bit | | o..szapc | .....c | Bit Test and Set |
| BTS | Evqp | Ib | | | | | 0F | BA | | | 5 | L | gen | bit | | o..szapc | .....c | Bit Test and Set |
| CALL | Jvds | | | | | | | E8 | | | | | gen | branch stack | | | | |
| CALL | Ev | | | | | | | FF | | | 2 | | gen | branch stack | | | | |
| CALL | Eq | | | | | | | FF | | | 2 | | gen | branch stack | | | | |
| CALLF | Mptp | | | | | | | FF | | | 3 | | gen | branch stack | | | | |
| CBW | AX | AL | | | | | | 98 | | | | | gen | conver | | | | Convert |
| CWDE | EAX | AX | | | | | | | | | | | | | | | | |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CDQE | RAX | EAX | | | | | | | | | | | | | | | | |
| CLC | | | | | | | | F8 | | | | | gen | flgctrl | | .....c | .....c | Clear Carry Flag |
| CLD | | | | | | | | FC | | | | | gen | flgctrl | | .d..... | .d..... | Clear Direction Flag |
| CLFLUSH | Mb | | | | sse2 | | 0F | AE | | | 7 | | cachect | | | | | Flush Cache Line |
| CLI | | | | | | | | FA | | | | | gen | flgctrl | | ..i..... | ..i..... | Clear Interrupt Flag |
| CLTS | CR0 | | | | | | 0F | 06 | | | | | system | | | | | Clear Task-Switched Flag in CR0 |
| CMC | | | | | | | | F5 | | | | | gen | flgctrl | | .....c | .....c | Complement Carry Flag |
| CMOVB | Gvqp | Evqp | | | | | 0F | 42 | | ttTn | r | | gen | datamov | | | | Conditional Move - below/not above or equal/carry (CF=1) |
| CMOVNAE | Gvqp | Evqp | | | | | | | | | | | | | | | | |
| CMOVC | Gvqp | Evqp | | | | | | | | | | | | | | | | |
| CMOVBE | Gvqp | Evqp | | | | | 0F | 46 | | tTTn | r | | gen | datamov | | | | Conditional Move - below or equal/not above (CF=1 AND ZF=1) |
| CMOVNA | Gvqp | Evqp | | | | | | | | | | | | | | | | |
| CMOVL | Gvqp | Evqp | | | | | 0F | 4C | | TTtn | r | | gen | datamov | | | | Conditional Move - less/not greater (SF!=OF) |
| CMOVNGE | Gvqp | Evqp | | | | | | | | | | | | | | | | |
| CMOVLE | Gvqp | Evqp | | | | | 0F | 4E | | TTTn | r | | gen | datamov | | | | Conditional Move - less or equal/not greater ((ZF=1) OR (SF!=OF)) |
| CMOVNG | Gvqp | Evqp | | | | | | | | | | | | | | | | |
| CMOVNB | Gvqp | Evqp | | | | | 0F | 43 | | ttTN | r | | gen | datamov | | | | Conditional Move - not below/above or equal/not carry (CF=0) |
| CMOVAE | Gvqp | Evqp | | | | | | | | | | | | | | | | |
| CMOVNC | Gvqp | Evqp | | | | | | | | | | | | | | | | |
| CMOVNBE | Gvqp | Evqp | | | | | 0F | 47 | | tTTN | r | | gen | datamov | | | | Conditional Move - not below or equal/above (CF=0 AND ZF=0) |
| CMOVA | Gvqp | Evqp | | | | | | | | | | | | | | | | |
| CMOVNL | Gvqp | Evqp | | | | | 0F | 4D | | TTtN | r | | gen | datamov | | | | Conditional Move - not less/greater or equal (SF=OF) |
| CMOVGE | Gvqp | Evqp | | | | | | | | | | | | | | | | |
| CMOVNLE | Gvqp | Evqp | | | | | 0F | 4F | | TTTN | r | | gen | datamov | | | | Conditional Move - not less nor equal/greater ((ZF=0) AND (SF=OF)) |
| CMOVG | Gvqp | Evqp | | | | | | | | | | | | | | | | |
| CMOVNO | Gvqp | Evqp | | | | | 0F | 41 | | tttN | r | | gen | datamov | | | | Conditional Move - not overflow (OF=0) |
| CMOVNP | Gvqp | Evqp | | | | | 0F | 4B | | TtTN | r | | gen | datamov | | | | Conditional Move - not parity/parity odd |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CMOVPO | Gvqp | Evqp | | | | | | | | | | | | | | | | |
| CMOVNS | Gvqp | Evqp | | | | | 0F | 49 | | TttN | r | | gen | datamov | | | | Conditional Move - not sign (SF=0) |
| CMOVNZ | Gvqp | Evqp | | | | | 0F | 45 | | tTtN | r | | gen | datamov | | | | Conditional Move - not zero/not equal (ZF=1) |
| CMOVNE | Gvqp | Evqp | | | | | | | | | | | | | | | | |
| CMOVO | Gvqp | Evqp | | | | | 0F | 40 | | tttn | r | | gen | datamov | | | | Conditional Move - overflow (OF=1) |
| CMOVP | Gvqp | Evqp | | | | | 0F | 4A | | TtTn | r | | gen | datamov | | | | Conditional Move - parity/parity even (PF=1) |
| CMOVPE | Gvqp | Evqp | | | | | | | | | | | | | | | | |
| CMOVS | Gvqp | Evqp | | | | | 0F | 48 | | Tttn | r | | gen | datamov | | | | Conditional Move - sign (SF=1) |
| CMOVZ | Gvqp | Evqp | | | | | 0F | 44 | | tTtn | r | | gen | datamov | | | | Conditional Move - zero/equal (ZF=0) |
| CMOVE | Gvqp | Evqp | | | | | | | | | | | | | | | | |
| CMP | Eb | Gb | | | | | | 38 | | dw | r | | gen | arith | binary | o..szapc | o..szapc | Compare Two Operands |
| CMP | Evqp | Gvqp | | | | | | 39 | | dW | r | | gen | arith | binary | o..szapc | o..szapc | Compare Two Operands |
| CMP | Gb | Eb | | | | | | 3A | | Dw | r | | gen | arith | binary | o..szapc | o..szapc | Compare Two Operands |
| CMP | Gvqp | Evqp | | | | | | 3B | | DW | r | | gen | arith | binary | o..szapc | o..szapc | Compare Two Operands |
| CMP | AL | Ib | | | | | | 3C | | w | | | gen | arith | binary | o..szapc | o..szapc | Compare Two Operands |
| CMP | rAX | Ivds | | | | | | 3D | | W | | | gen | arith | binary | o..szapc | o..szapc | Compare Two Operands |
| CMP | Eb | Ib | | | | | | 80 | | w | 7 | | gen | arith | binary | o..szapc | o..szapc | Compare Two Operands |
| CMP | Evqp | Ivds | | | | | | 81 | | W | 7 | | gen | arith | binary | o..szapc | o..szapc | Compare Two Operands |
| CMP | Evqp | Ibs | | | | | | 83 | | SW | 7 | | gen | arith | binary | o..szapc | o..szapc | Compare Two Operands |
| CMPPD | Vpd | Wpd | Ib | | sse2 | 66 | 0F | C2 | | | r | | pcksclr | compar | | | | Compare Packed Double-FP Values |
| CMPPS | Vps | Wps | Ib | | sse1 | | 0F | C2 | | | r | | simdfp | compar | | | | Compare Packed Single-FP Values |
| CMPS | Yb | Xb | | | | | | A6 | | w | | | gen | arith string | binary | o..szapc | o..szapc | Compare String Operands |
| CMPSB | Yb | Xb | | | | | | | | | | | | | | | | |
| CMPS | Yvqp | Xvqp | | | | | | A7 | | W | | | gen | arith string | binary | o..szapc | o..szapc | Compare String Operands |
| CMPSW | Ywo | Xwo | | | | | | | | | | | | | | | | |
| CMPSD | Ydo | Xdo | | | | | | | | | | | | | | | | |
| CMPSQ | Yqp | Xqp | | | | | | | | | | | | | | | | |
| CMPSD | Vsd | Wsd | Ib | | sse2 | F2 | 0F | C2 | | | r | | pcksclr | compar | | | | Compare Scalar Double-FP Values |
| CMPSS | Vss | Wss | Ib | | sse1 | F3 | 0F | C2 | | | r | | simdfp | compar | | | | Compare Scalar Single-FP Values |

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CMPXCHG | Eb | AL | Gb | | | | 0F | B0 | | dw | r | L | gen | datamov arith | binary | o..szapc | o..szapc | Compare and Exchange |
| CMPXCHG | Evqp | rAX | Gvqp | | | | 0F | B1 | | dW | r | L | gen | datamov arith | binary | o..szapc | o..szapc | Compare and Exchange |
| CMPXCHG8B | Mq | EAX | EDX | ... | | | 0F | C7 | | | 1 | L | gen | datamov arith | binary | ....z... | ....z... | Compare and Exchange Bytes |
| CMPXCHG8B | Mq | EAX | EDX | ... | | | 0F | C7 | | | 1 | L | gen | datamov arith | binary | ....z... | ....z... | Compare and Exchange Bytes |
| CMPXCHG16B | Mdq | RAX | RDX | ... | | | | | | | | | | | | | | |
| COMISD | Vsd | Wsd | | | sse2 | 66 | 0F | 2F | | | r | | pcksclr | compar | | ....z.pc | ....z.pc | Compare Scalar Ordered Double-FP Values and Set EFLAGS |
| COMISS | Vss | Wss | | | sse1 | | 0F | 2F | | | r | | simdfp | compar | | ....z.pc | ....z.pc | Compare Scalar Ordered Single-FP Values and Set EFLAGS |
| CPUID | I... | EAX | ECX | ... | | | 0F | A2 | | | | | gen | control | | | | CPU Identification |
| CRC32 | Gdqp | Eb | | | sse42 | F2 | 0F | 38 | F0 | | r | | | | | | | Accumulate CRC32 Value |
| CRC32 | Gdqp | Evqp | | | sse42 | F2 | 0F | 38 | F1 | | r | | | | | | | Accumulate CRC32 Value |
| CVTDQ2PD | Vpd | Wdq | | | sse2 | F3 | 0F | E6 | | | r | | pcksclr | conver | | | | Convert Packed DW Integers to Double-FP Values |
| CVTDQ2PS | Vps | Wdq | | | sse2 | | 0F | 5B | | | r | | pcksp | | | | | Convert Packed DW Integers to Single-FP Values |
| CVTPD2DQ | Vdq | Wpd | | | sse2 | F2 | 0F | E6 | | | r | | pcksclr | conver | | | | Convert Packed Double-FP Values to DW Integers |
| CVTPD2PI | Ppi | Wpd | | | sse2 | 66 | 0F | 2D | | | r | | pcksclr | conver | | | | Convert Packed Double-FP Values to DW Integers |
| CVTPD2PS | Vps | Wpd | | | sse2 | 66 | 0F | 5A | | | r | | pcksclr | conver | | | | Convert Packed Double-FP Values to Single-FP Values |
| CVTPI2PD | Vpd | Qpi | | | sse2 | 66 | 0F | 2A | | | r | | pcksclr | conver | | | | Convert Packed DW Integers to Double-FP Values |
| CVTPI2PS | Vps | Qpi | | | sse1 | | 0F | 2A | | | r | | conver | | | | | Convert Packed DW Integers to Single-FP Values |
| CVTPS2DQ | Vdq | Wps | | | sse2 | 66 | 0F | 5B | | | r | | pcksp | | | | | Convert Packed Single-FP Values to DW Integers |
| CVTPS2PD | Vpd | Wps | | | sse2 | | 0F | 5A | | | r | | pcksclr | conver | | | | Convert Packed Single-FP Values to Double-FP Values |
| CVTPS2PI | Ppi | Wpsq | | | sse1 | | 0F | 2D | | | r | | conver | | | | | Convert Packed Single-FP Values to DW Integers |
| CVTSD2SI | Gdqp | Wsd | | | sse2 | F2 | 0F | 2D | | | r | | pcksclr | conver | | | | Convert Scalar Double-FP Value to DW Integer |
| CVTSD2SS | Vss | Wsd | | | sse2 | F2 | 0F | 5A | | | r | | pcksclr | conver | | | | Convert Scalar Double-FP Value to Scalar Single-FP |

| | | | | | | | | | | | | | | | | | | Value |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CVTSI2SD | Vsd | Edqp | | | sse2 | F2 | 0F | 2A | | | r | | pcksclr | conver | | | | Convert DW Integer to Scalar Double-FP Value |
| CVTSI2SS | Vss | Edqp | | | sse1 | F3 | 0F | 2A | | | r | | conver | | | | | Convert DW Integer to Scalar Single-FP Value |
| CVTSS2SD | Vsd | Wss | | | sse2 | F3 | 0F | 5A | | | r | | pcksclr | conver | | | | Convert Scalar Single-FP Value to Scalar Double-FP Value |
| CVTSS2SI | Gdqp | Wss | | | sse1 | F3 | 0F | 2D | | | r | | conver | | | | | Convert Scalar Single-FP Value to DW Integer |
| CVTTPD2DQ | Vdq | Wpd | | | sse2 | 66 | 0F | E6 | | | r | | pcksclr | conver | | | | Convert with Trunc. Packed Double-FP Values to DW Integers |
| CVTTPD2PI | Ppi | Wpd | | | sse2 | 66 | 0F | 2C | | | r | | pcksclr | conver | | | | Convert with Trunc. Packed Double-FP Values to DW Integers |
| CVTTPS2DQ | Vdq | Wps | | | sse2 | F3 | 0F | 5B | | | r | | pcksp | | | | | Convert with Trunc. Packed Single-FP Values to DW Integers |
| CVTTPS2PI | Ppi | Wpsq | | | sse1 | | 0F | 2C | | | r | | conver | | | | | Convert with Trunc. Packed Single-FP Values to DW Integers |
| CVTTSD2SI | Gdqp | Wsd | | | sse2 | F2 | 0F | 2C | | | r | | pcksclr | conver | | | | Conv. with Trunc. Scalar Double-FP Value to Signed DW Int |
| CVTTSS2SI | Gdqp | Wss | | | sse1 | F3 | 0F | 2C | | | r | | conver | | | | | Convert with Trunc. Scalar Single-FP Value to DW Integer |
| CWD | DX | AX | | | | | | 99 | | | | | gen | conver | | | | Convert |
| CDQ | EDX | EAX | | | | | | | | | | | | | | | | |
| CQO | RDX | RAX | | | | | | | | | | | | | | | | |
| DEC | Eb | | | | | | | FE | | w | 1 | | gen | arith | binary | o..szap. | o..szap. | Decrement by 1 |
| DEC | Evqp | | | | | | | FF | | W | 1 | | gen | arith | binary | o..szap. | o..szap. | Decrement by 1 |
| DIV | AL | AH | AX | Eb | | | | F6 | | w | 6 | | gen | arith | binary | o..szapc | | Unsigned Divide |
| DIV | rDX | rAX | Evqp | | | | | F7 | | w | 6 | | gen | arith | binary | o..szapc | | Unsigned Divide |
| DIVPD | Vpd | Wpd | | | sse2 | 66 | 0F | 5E | | | r | | pcksclr | arith | | | | Divide Packed Double-FP Values |
| DIVPS | Vps | Wps | | | sse1 | | 0F | 5E | | | r | | simdfp | arith | | | | Divide Packed Single-FP Values |
| DIVSD | Vsd | Wsd | | | sse2 | F2 | 0F | 5E | | | r | | pcksclr | arith | | | | Divide Scalar Double-FP Values |
| DIVSS | Vss | Wss | | | sse1 | F3 | 0F | 5E | | | r | | simdfp | arith | | | | Divide Scalar Single-FP Values |
| DPPD | Vpd | Wpd | | | sse41 | 66 | 0F | 3A | 41 | | r | | simdfp | arith | | | | Dot Product of Packed |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | | | | | | | Double-FP Values |
| DPPS | Vps | Wps | | | sse41 | 66 | 0F | 3A | 40 | | r | | simdfp | arith | | | | Dot Product of Packed Single-FP Values |
| EMMS | | | | | mmx | | 0F | 77 | | | | | x87fpu | control | | | | Empty MMX Technology State |
| ENTER | rBP | Iw | Ib | | | | | C8 | | | | | gen | stack | | | | Make Stack Frame for Procedure Parameters |
| EXTRACTPS | Ed | Vdq | Ib | | sse41 | 66 | 0F | 3A | 17 | | r | | simdfp | datamov | | | | Extract Packed Single-FP Value |
| F2XM1 | ST | | | | | | | D9 | F0 | | 6 | | x87fpu | trans | | 0123 | .1.. | Compute 2x-1 |
| FABS | ST | | | | | | | D9 | E1 | | 4 | | x87fpu | arith | | 0123 | .1.. | Absolute Value |
| FADD | ST | Msr | | | | | | D8 | | mf | 0 | | x87fpu | arith | | 0123 | .1.. | Add |
| FADD | ST | EST | | | | | | | | | | | | | | | | |
| FADD | ST | Mdr | | | | | | DC | | Mf | 0 | | x87fpu | arith | | 0123 | .1.. | Add |
| FADD | EST | ST | | | | | | DC | | | 0 | | x87fpu | arith | | 0123 | .1.. | Add |
| FADDP | EST | ST | | | | | | DE | | | 0 | p | x87fpu | arith | | 0123 | .1.. | Add and Pop |
| FADDP | ST1 | ST | | | | | | DE | C1 | | 0 | p | x87fpu | arith | | 0123 | .1.. | Add and Pop |
| FBLD | ST | Mbcd | | | | | | DF | | | 4 | s | x87fpu | datamov | | 0123 | .1.. | Load Binary Coded Decimal |
| FBSTP | Mbcd | ST | | | | | | DF | | | 6 | p | x87fpu | datamov | | 0123 | .1.. | Store BCD Integer and Pop |
| FCHS | ST | | | | | | | D9 | E0 | | 4 | | x87fpu | arith | | 0123 | .1.. | Change Sign |
| FCLEX | | | | | | 9B | | DB | E2 | | 4 | | x87fpu | control | | 0123 | | Clear Exceptions |
| FCMOVB | ST | EST | | | | | | DA | | | 0 | | x87fpu | datamov | | 0123 | .1.. | FP Conditional Move - below (CF=1) |
| FCMOVBE | ST | EST | | | | | | DA | | | 2 | | x87fpu | datamov | | 0123 | .1.. | FP Conditional Move - below or equal (CF=1 or ZF=1) |
| FCMOVE | ST | EST | | | | | | DA | | | 1 | | x87fpu | datamov | | 0123 | .1.. | FP Conditional Move - equal (ZF=1) |
| FCMOVNB | ST | EST | | | | | | DB | | | 0 | | x87fpu | datamov | | 0123 | .1.. | FP Conditional Move - not below (CF=0) |
| FCMOVNBE | ST | EST | | | | | | DB | | | 2 | | x87fpu | datamov | | 0123 | .1.. | FP Conditional Move - below or equal (CF=0 and ZF=0) |
| FCMOVNE | ST | EST | | | | | | DB | | | 1 | | x87fpu | datamov | | 0123 | .1.. | FP Conditional Move - not equal (ZF=0) |
| FCMOVNU | ST | EST | | | | | | DB | | | 3 | | x87fpu | datamov | | 0123 | .1.. | FP Conditional Move - not unordered (PF=0) |
| FCMOVU | ST | EST | | | | | | DA | | | 3 | | x87fpu | datamov | | 0123 | .1.. | FP Conditional Move - unordered (PF=1) |
| FCOM | ST | ESsr | | | | | | D8 | | mf | 2 | | x87fpu | compar | | 0123 | 0123 | Compare Real |
| FCOM | ST | ST1 | | | | | | D8 | D1 | | 2 | | x87fpu | compar | | 0123 | 0123 | Compare Real |
| FCOM | ST | Mdr | | | | | | DC | | Mf | 2 | | x87fpu | compar | | 0123 | 0123 | Compare Real |
| FCOM2 alias | ST | EST | | | | | | DC | | | 2 | | x87fpu | compar | | 0123 | 0123 | Compare Real |
| FCOMI | ST | EST | | | | | | DB | | | 6 | | x87fpu | compar | | o...z.pc .1.. | o...z.pc .1.. | Compare Floating Point Values and Set EFLAGS |

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FCOMIP | ST | EST | | | | | | DF | | | 6 | p | x87fpu | compar | | o...z.pc<br>.1.. | o...z.pc<br>.1.. | Compare Floating Point Values and Set EFLAGS and Pop |
| FCOMP | ST | ESsr | | | | | | D8 | | mf | 3 | p | x87fpu | compar | | 0123 | 0123 | Compare Real and Pop |
| FCOMP | ST | ST1 | | | | | | D8 | D9 | | 3 | p | x87fpu | compar | | 0123 | 0123 | Compare Real and Pop |
| FCOMP | ST | Mdr | | | | | | DC | | Mf | 3 | p | x87fpu | compar | | 0123 | 0123 | Compare Real and Pop |
| FCOMP3 alias | ST | EST | | | | | | DC | | | 3 | p | x87fpu | compar | | 0123 | 0123 | Compare Real and Pop |
| FCOMP5 alias | ST | EST | | | | | | DE | | | 2 | p | x87fpu | compar | | 0123 | 0123 | Compare Real and Pop |
| FCOMPP | ST | ST1 | | | | | | DE | D9 | | 3 | P | x87fpu | compar | | 0123 | 0123 | Compare Real and Pop Twice |
| FCOS | ST | | | | | | | D9 | FF | | 7 | | x87fpu | trans | | 0123 | .12. | Cosine |
| FDECSTP | | | | | | | | D9 | F6 | | 6 | | x87fpu | control | | 0123 | .1.. | Decrement Stack-Top Pointer |
| FDIV | ST | Msr | | | | | | D8 | | mf | 6 | | x87fpu | arith | | 0123 | .1.. | Divide |
| FDIV | ST | EST | | | | | | | | | | | | | | | | |
| FDIV | ST | Mdr | | | | | | DC | | Mf | 6 | | x87fpu | arith | | 0123 | .1.. | Divide |
| FDIV | EST | ST | | | | | | DC | | | 7 | | x87fpu | arith | | 0123 | .1.. | Divide and Pop |
| FDIVP | EST | ST | | | | | | DE | | | 7 | p | x87fpu | arith | | 0123 | .1.. | Divide and Pop |
| FDIVP | ST1 | ST | | | | | | DE | F9 | | 7 | p | x87fpu | arith | | 0123 | .1.. | Divide and Pop |
| FDIVR | ST | Msr | | | | | | D8 | | mf | 7 | | x87fpu | arith | | 0123 | .1.. | Reverse Divide |
| FDIVR | ST | EST | | | | | | | | | | | | | | | | |
| FDIVR | EST | ST | | | | | | DC | | | 6 | | x87fpu | arith | | 0123 | .1.. | Reverse Divide |
| FDIVR | ST | Mdr | | | | | | DC | | Mf | 7 | | x87fpu | arith | | 0123 | .1.. | Reverse Divide |
| FDIVRP | EST | ST | | | | | | DE | | | 6 | p | x87fpu | arith | | 0123 | .1.. | Reverse Divide and Pop |
| FDIVRP | ST1 | ST | | | | | | DE | F1 | | 6 | p | x87fpu | arith | | 0123 | .1.. | Reverse Divide and Pop |
| FFREE | EST | | | | | | | DD | | | 0 | | x87fpu | control | | 0123 | | Free Floating-Point Register |
| FFREEP | EST | | | | | | | DF | | | 0 | p | x87fpu | control | | 0123 | | Free Floating-Point Register and Pop |
| FIADD | ST | Mdi | | | | | | DA | | mF | 0 | | x87fpu | arith | | 0123 | .1.. | Add |
| FIADD | ST | Mwi | | | | | | DE | | M<br>F | 0 | | x87fpu | arith | | 0123 | .1.. | Add |
| FICOM | ST | Mdi | | | | | | DA | | mF | 2 | | x87fpu | compar | | 0123 | 0123 | Compare Integer |
| FICOM | ST | Mwi | | | | | | DE | | M<br>F | 2 | | x87fpu | compar | | 0123 | 0123 | Compare Integer |
| FICOMP | ST | Mdi | | | | | | DA | | mF | 3 | p | x87fpu | compar | | 0123 | 0123 | Compare Integer and Pop |
| FICOMP | ST | Mwi | | | | | | DE | | M<br>F | 3 | p | x87fpu | compar | | 0123 | 0123 | Compare Integer and Pop |
| FIDIV | ST | Mdi | | | | | | DA | | mF | 6 | | x87fpu | arith | | 0123 | .1.. | Divide |
| FIDIV | ST | Mwi | | | | | | DE | | M<br>F | 6 | | x87fpu | arith | | 0123 | .1.. | Divide |
| FIDIVR | ST | Mdi | | | | | | DA | | mF | 7 | | x87fpu | arith | | 0123 | .1.. | Reverse Divide |
| FIDIVR | ST | Mwi | | | | | | DE | | M<br>F | 7 | | x87fpu | arith | | 0123 | .1.. | Reverse Divide |
| FILD | ST | Mdi | | | | | | DB | | mF | 0 | s | x87fpu | datamov | | 0123 | .1.. | Load Integer |
| FILD | ST | Mwi | | | | | | DF | | M | 0 | s | x87fpu | datamov | | 0123 | .1.. | Load Integer |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | F | | | | | | | | |
| FILD | ST | Mqi | | | | | | DF | | | 5 | s | x87fpu | datamov | | 0123 | .1.. | Load Integer |
| FIMUL | ST | Mdi | | | | | | DA | | mF | 1 | | x87fpu | arith | | 0123 | .1.. | Multiply |
| FIMUL | ST | Mwi | | | | | | DE | | M F | 1 | | x87fpu | arith | | 0123 | .1.. | Multiply |
| FINCSTP | | | | | | | | D9 | F7 | | 6 | | x87fpu | control | | 0123 | .1.. | Increment Stack-Top Pointer |
| FINIT | | | | | | 9B | | DB | E3 | | 4 | | x87fpu | control | | 0123 | | Initialize Floating-Point Unit |
| FIST | Mdi | ST | | | | | | DB | | mF | 2 | | x87fpu | datamov | | 0123 | .1.. | Store Integer |
| FIST | Mwi | ST | | | | | | DF | | M F | 2 | | x87fpu | datamov | | 0123 | .1.. | Store Integer |
| FISTP | Mdi | ST | | | | | | DB | | mF | 3 | p | x87fpu | datamov | | 0123 | .1.. | Store Integer and Pop |
| FISTP | Mwi | ST | | | | | | DF | | M F | 3 | p | x87fpu | datamov | | 0123 | .1.. | Store Integer and Pop |
| FISTP | Mqi | ST | | | | | | DF | | | 7 | p | x87fpu | datamov | | 0123 | .1.. | Store Integer and Pop |
| FISTTP | Mdi | ST | | | sse3 | | | DB | | mF | 1 | p | x87fpu | conver | | 0123 | .1.. | Store Integer with Truncation and Pop |
| FISTTP | Mqi | ST | | | sse3 | | | DD | | | 1 | p | x87fpu | conver | | 0123 | .1.. | Store Integer with Truncation and Pop |
| FISTTP | Mwi | ST | | | sse3 | | | DF | | M F | 1 | p | x87fpu | conver | | 0123 | .1.. | Store Integer with Truncation and Pop |
| FISUB | ST | Mdi | | | | | | DA | | mF | 4 | | x87fpu | arith | | 0123 | .1.. | Subtract |
| FISUB | ST | Mwi | | | | | | DE | | M F | 4 | | x87fpu | arith | | 0123 | .1.. | Subtract |
| FISUBR | ST | Mdi | | | | | | DA | | mF | 5 | | x87fpu | arith | | 0123 | .1.. | Reverse Subtract |
| FISUBR | ST | Mwi | | | | | | DE | | M F | 5 | | x87fpu | arith | | 0123 | .1.. | Reverse Subtract |
| FLD | ST | ESsr | | | | | | D9 | | mf | 0 | s | x87fpu | datamov | | 0123 | .1.. | Load Floating Point Value |
| FLD | ST | Mer | | | | | | DB | | | 5 | s | x87fpu | datamov | | 0123 | .1.. | Load Floating Point Value |
| FLD | ST | Mdr | | | | | | DD | | Mf | 0 | s | x87fpu | datamov | | 0123 | .1.. | Load Floating Point Value |
| FLD1 | ST | | | | | | | D9 | E8 | | 5 | s | x87fpu | ldconst | | 0123 | .1.. | Load Constant +1.0 |
| FLDCW | Mw | | | | | | | D9 | | | 5 | | x87fpu | control | | 0123 | | Load x87 FPU Control Word |
| FLDENV | Me | | | | | | | D9 | | | 4 | | x87fpu | control | | 0123 | 0123 | Load x87 FPU Environment |
| FLDL2E | ST | | | | | | | D9 | EA | | 5 | s | x87fpu | ldconst | | 0123 | .1.. | Load Constant log2e |
| FLDL2T | ST | | | | | | | D9 | E9 | | 5 | s | x87fpu | ldconst | | 0123 | .1.. | Load Constant log210 |
| FLDLG2 | ST | | | | | | | D9 | EC | | 5 | s | x87fpu | ldconst | | 0123 | .1.. | Load Constant log102 |
| FLDLN2 | ST | | | | | | | D9 | ED | | 5 | s | x87fpu | ldconst | | 0123 | .1.. | Load Constant loge2 |
| FLDPI | ST | | | | | | | D9 | EB | | 5 | s | x87fpu | ldconst | | 0123 | .1.. | Load Constant π |
| FLDZ | ST | | | | | | | D9 | EE | | 5 | s | x87fpu | ldconst | | 0123 | .1.. | Load Constant +0.0 |
| FMUL | ST | Msr | | | | | | D8 | | mf | 1 | | x87fpu | arith | | 0123 | .1.. | Multiply |
| FMUL | ST | EST | | | | | | | | | | | | | | | | |
| FMUL | ST | Mdr | | | | | | DC | | Mf | 1 | | x87fpu | arith | | 0123 | .1.. | Multiply |
| FMUL | EST | ST | | | | | | DC | | | 1 | | x87fpu | arith | | 0123 | .1.. | Multiply |
| FMULP | EST | ST | | | | | | DE | | | 1 | p | x87fpu | arith | | 0123 | .1.. | Multiply and Pop |
| FMULP | ST1 | ST | | | | | | DE | C9 | | 1 | p | x87fpu | arith | | 0123 | .1.. | Multiply and Pop |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FNCLEX | | | | | | | | DB | E2 | | 4 | | x87fpu | control | | 0123 | | Clear Exceptions |
| FNDISI nop | | | | | | | | DB | E1 | | 4 | | obsol | control | | | | Treated as Integer NOP |
| FNENI nop | | | | | | | | DB | E0 | | 4 | | obsol | control | | | | Treated as Integer NOP |
| FNINIT | | | | | | | | DB | E3 | | 4 | | x87fpu | control | | 0123 | | Initialize Floating-Point Unit |
| FNOP | | | | | | | | D9 | D0 | | 2 | | x87fpu | control | | 0123 | | No Operation |
| FNSAVE | Mst | ST | ST1 | ... | | | | DD | | | 6 | | x87fpu | control | | 0123 | 0123 | Store x87 FPU State |
| FNSETPM nop | | | | | | | | DB | E4 | | 4 | | obsol | control | | | | Treated as Integer NOP |
| FNSTCW | Mw | | | | | | | D9 | | | 7 | | x87fpu | control | | 0123 | | Store x87 FPU Control Word |
| FNSTENV | Me | | | | | | | D9 | | | 6 | | x87fpu | control | | 0123 | | Store x87 FPU Environment |
| FNSTSW | Mw | | | | | | | DD | | | 7 | | x87fpu | control | | 0123 | | Store x87 FPU Status Word |
| FNSTSW | AX | | | | | | | DF | E0 | | 4 | | x87fpu | control | | 0123 | | Store x87 FPU Status Word |
| FPATAN | ST1 | ST | | | | | | D9 | F3 | | 6 | p | x87fpu | trans | | 0123 | .1.. | Partial Arctangent and Pop |
| FPREM | ST | ST1 | | | | | | D9 | F8 | | 7 | | x87fpu | arith | | 0123 | 0123 | Partial Remainder (for compatibility with i8087 and i287) |
| FPREM1 | ST | ST1 | | | | | | D9 | F5 | | 6 | | x87fpu | arith | | 0123 | 0123 | IEEE Partial Remainder |
| FPTAN | ST | | | | | | | D9 | F2 | | 6 | s | x87fpu | trans | | 0123 | .12. | Partial Tangent |
| FRNDINT | ST | | | | | | | D9 | FC | | 7 | | x87fpu | arith | | 0123 | .1.. | Round to Integer |
| FRSTOR | ST | ST1 | ST2 | ... | | | | DD | | | 4 | | x87fpu | control | | 0123 | 0123 | Restore x87 FPU State |
| FS | FS | | | | | 64 | | | | | | | prefix | segreg | | | | FS segment override prefix |
| FSAVE | Mst | ST | ST1 | ... | | 9B | | DD | | | 6 | | x87fpu | control | | 0123 | 0123 | Store x87 FPU State |
| FSCALE | ST | ST1 | | | | | | D9 | FD | | 7 | | x87fpu | arith | | 0123 | .1.. | Scale |
| FSIN | ST | | | | | | | D9 | FE | | 7 | | x87fpu | trans | | 0123 | .12. | Sine |
| FSINCOS | ST | | | | | | | D9 | FB | | 7 | s | x87fpu | trans | | 0123 | .12. | Sine and Cosine |
| FSQRT | ST | | | | | | | D9 | FA | | 7 | | x87fpu | arith | | 0123 | .1.. | Square Root |
| FST | Msr | ST | | | | | | D9 | | mf | 2 | | x87fpu | datamov | | 0123 | .1.. | Store Floating Point Value |
| FST | Mdr | ST | | | | | | DD | | Mf | 2 | | x87fpu | datamov | | 0123 | .1.. | Store Floating Point Value |
| FST | ST | EST | | | | | | DD | | | 2 | | x87fpu | datamov | | 0123 | .1.. | Store Floating Point Value |
| FSTCW | Mw | | | | | 9B | | D9 | | | 7 | | x87fpu | control | | 0123 | | Store x87 FPU Control Word |
| FSTENV | Me | | | | | 9B | | D9 | | | 6 | | x87fpu | control | | 0123 | | Store x87 FPU Environment |
| FSTP | Msr | ST | | | | | | D9 | | mf | 3 | p | x87fpu | datamov | | 0123 | .1.. | Store Floating Point Value and Pop |
| FSTP | Mer | ST | | | | | | DB | | | 7 | p | x87fpu | datamov | | 0123 | .1.. | Store Floating Point Value and Pop |
| FSTP | Mdr | ST | | | | | | DD | | Mf | 3 | p | x87fpu | datamov | | 0123 | .1.. | Store Floating Point Value and Pop |
| FSTP | ST | EST | | | | | | DD | | | 3 | p | x87fpu | datamov | | 0123 | .1.. | Store Floating Point Value and Pop |
| FSTP1 part alias4 | EST | ST | | | | | | D9 | | | 3 | p | x87fpu | datamov | | 0123 | .1.. | Store Floating Point Value and Pop |
| FSTP8 alias | EST | ST | | | | | | DF | | | 2 | p | x87fpu | datamov | | 0123 | .1.. | Store Floating Point Value and Pop |
| FSTP9 alias | EST | ST | | | | | | DF | | | 3 | p | x87fpu | datamov | | 0123 | .1.. | Store Floating Point Value and Pop |
| FSTSW | Mw | | | | | 9B | | DD | | | 7 | | x87fpu | control | | 0123 | | Store x87 FPU Status Word |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FSTSW | AX | | | | | 9B | | DF | E0 | | 4 | | x87fpu | control | | 0123 | | Store x87 FPU Status Word |
| FSUB | ST | Msr | | | | | | D8 | | mf | 4 | | x87fpu | arith | | 0123 | .1.. | Subtract |
| FSUB | ST | EST | | | | | | | | | | | | | | | | |
| FSUB | ST | Mdr | | | | | | DC | | Mf | 4 | | x87fpu | arith | | 0123 | .1.. | Subtract |
| FSUB | EST | ST | | | | | | DC | | | 5 | | x87fpu | arith | | 0123 | .1.. | Subtract |
| FSUBP | EST | ST | | | | | | DE | | | 5 | p | x87fpu | arith | | 0123 | .1.. | Subtract and Pop |
| FSUBP | ST1 | ST | | | | | | DE | E9 | | 5 | p | x87fpu | arith | | 0123 | .1.. | Subtract and Pop |
| FSUBR | ST | Msr | | | | | | D8 | | mf | 5 | | x87fpu | arith | | 0123 | .1.. | Reverse Subtract |
| FSUBR | ST | EST | | | | | | | | | | | | | | | | |
| FSUBR | EST | ST | | | | | | DC | | | 4 | | x87fpu | arith | | 0123 | .1.. | Reverse Subtract |
| FSUBR | ST | Mdr | | | | | | DC | | Mf | 5 | | x87fpu | arith | | 0123 | .1.. | Reverse Subtract |
| FSUBRP | EST | ST | | | | | | DE | | | 4 | p | x87fpu | arith | | 0123 | .1.. | Reverse Subtract and Pop |
| FSUBRP | ST1 | ST | | | | | | DE | E1 | | 4 | p | x87fpu | arith | | 0123 | .1.. | Reverse Subtract and Pop |
| FTST | ST | | | | | | | D9 | E4 | | 4 | | x87fpu | compar | | 0123 | 0123 | Test |
| FUCOM | ST | EST | | | | | | DD | | | 4 | | x87fpu | compar | | 0123 | 0123 | Unordered Compare Floating Point Values |
| FUCOM | ST | ST1 | | | | | | DD | E1 | | 4 | | x87fpu | compar | | 0123 | 0123 | Unordered Compare Floating Point Values |
| FUCOMI | ST | EST | | | | | | DB | | | 5 | | x87fpu | compar | | o...z.pc .1.. | o...z.pc .1.. | Unordered Compare Floating Point Values and Set EFLAGS |
| FUCOMIP | ST | EST | | | | | | DF | | | 5 | p | x87fpu | compar | | o...z.pc .1.. | o...z.pc .1.. | Unordered Compare Floating Point Values and Set EFLAGS and Pop |
| FUCOMP | ST | EST | | | | | | DD | | | 5 | p | x87fpu | compar | | 0123 | 0123 | Unordered Compare Floating Point Values and Pop |
| FUCOMP | ST | ST1 | | | | | | DD | E9 | | 5 | p | x87fpu | compar | | 0123 | 0123 | Unordered Compare Floating Point Values and Pop |
| FUCOMPP | ST | ST1 | | | | | | DA | E9 | | 5 | P | x87fpu | compar | | 0123 | 0123 | Unordered Compare Floating Point Values and Pop Twice |
| FWAIT | | | | | | | | 9B | | | | | x87fpu | control | | 0123 | | Check pending unmasked floating-point exceptions |
| WAIT | | | | | | | | | | | | | | | | | | |
| FXAM | ST | | | | | | | D9 | E5 | | 4 | | x87fpu | | | 0123 | 0123 | Examine |
| FXCH | ST | EST | | | | | | D9 | | mf | 1 | | x87fpu | datamov | | 0123 | .1.. | Exchange Register Contents |
| FXCH | ST | ST1 | | | | | | D9 | C9 | | 1 | | x87fpu | datamov | | 0123 | .1.. | Exchange Register Contents |
| FXCH4 alias | ST | EST | | | | | | DD | | | 1 | | x87fpu | datamov | | 0123 | .1.. | Exchange Register Contents |
| FXCH7 alias | ST | EST | | | | | | DF | | | 1 | | x87fpu | datamov | | 0123 | .1.. | Exchange Register Contents |
| FXRSTOR | ST | ST1 | ST2 | ... | | | 0F | AE | | | 1 | | sm | | | | | Restore x87 FPU, MMX, XMM, and MXCSR State |
| FXRSTOR | ST | ST1 | ST2 | ... | | | 0F | AE | | | 1 | | sm | | | | | Restore x87 FPU, MMX, XMM, and MXCSR State |

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FXSAVE | Mstx | ST | ST1 | ... | | | 0F | AE | | | 0 | | sm | | | | | Save x87 FPU, MMX, XMM, and MXCSR State |
| FXSAVE | Mstx | ST | ST1 | ... | | | 0F | AE | | | 0 | | sm | | | | | Save x87 FPU, MMX, XMM, and MXCSR State |
| FXTRACT | ST | | | | | | | D9 | F4 | | 6 | s | x87fpu | arith | | 0123 | .1.. | Extract Exponent and Significand |
| FYL2X | ST1 | ST | | | | | | D9 | F1 | | 6 | p | x87fpu | trans | | 0123 | .1.. | Compute y × log2x and Pop |
| FYL2XP1 | ST1 | ST | | | | | | D9 | F9 | | 7 | p | x87fpu | trans | | 0123 | .1.. | Compute y × log2(x+1) and Pop |
| GETSEC | EAX | | | | smx | | 0F | 37 | | | | | | | | | | GETSEC Leaf Functions |
| GS | GS | | | | | 65 | | | | | | | prefix | segreg | | | | GS segment override prefix |
| HADDPD | Vpd | Wpd | | | sse3 | 66 | 0F | 7C | | | r | | simdfp | arith | | | | Packed Double-FP Horizontal Add |
| HADDPS | Vps | Wps | | | sse3 | F2 | 0F | 7C | | | r | | simdfp | arith | | | | Packed Single-FP Horizontal Add |
| HINT_NOP | Ev | | | | | | 0F | 18 | | | 4 | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 18 | | | 5 | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 18 | | | 6 | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 18 | | | 7 | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 19 | | | | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 1A | | | | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 1B | | | | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 1C | | | | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 1D | | | | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 1E | | | | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 1F | | | 1 | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 1F | | | 2 | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 1F | | | 3 | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 1F | | | 4 | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 1F | | | 5 | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 1F | | | 6 | | gen | control | | | | Hintable NOP |
| HINT_NOP | Ev | | | | | | 0F | 1F | | | 7 | | gen | control | | | | Hintable NOP |
| HLT | | | | | | | | F4 | | | | | system | | | | | Halt |
| HSUBPD | Vpd | Wpd | | | sse3 | 66 | 0F | 7D | | | r | | simdfp | arith | | | | Packed Double-FP Horizontal Subtract |
| HSUBPS | Vps | Wps | | | sse3 | F2 | 0F | 7D | | | r | | simdfp | arith | | | | Packed Single-FP Horizontal Subtract |
| IDIV | AL | AH | AX | Eb | | | | F6 | | w | 7 | | gen | arith | binary | o..szapc | | Signed Divide |
| IDIV | rDX | rAX | Evqp | | | | | F7 | | w | 7 | | gen | arith | binary | o..szapc | | Signed Divide |
| IMUL | Gvqp | Evqp | Ivds | | | | | 69 | | | r | | gen | arith | binary | o..szapc | o.....c | Signed Multiply |
| IMUL | Gvqp | Evqp | Ibs | | | | | 6B | | S | r | | gen | arith | binary | o..szapc | o.....c | Signed Multiply |
| IMUL | AX | AL | Eb | | | | | F6 | | w | 5 | | gen | arith | binary | o..szapc | o.....c | Signed Multiply |
| IMUL | rDX | rAX | Evqp | | | | | F7 | | w | 5 | | gen | arith | binary | o..szapc | o.....c | Signed Multiply |

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| IMUL | Gvqp | Evqp | | | | | 0F | AF | | D W | r | | gen | arith | binary | o..szapc | o.....c | Signed Multiply |
| IN | AL | Ib | | | | | | E4 | | w | | | gen | inout | | | | Input from Port |
| IN | eAX | Ib | | | | | | E5 | | W | | | gen | inout | | | | Input from Port |
| IN | AL | DX | | | | | | EC | | w | | | gen | inout | | | | Input from Port |
| IN | eAX | DX | | | | | | ED | | W | | | gen | inout | | | | Input from Port |
| INC | Eb | | | | | | | FE | | w | 0 | | gen | arith | binary | o..szap. | o..szap. | Increment by 1 |
| INC | Evqp | | | | | | | FF | | W | 0 | | gen | arith | binary | o..szap. | o..szap. | Increment by 1 |
| INS | Yb | DX | | | | | | 6C | | w | | | gen | inout string | .d..... | | | |
| INSB | Yb | DX | | | | | | | | | | | | | | | | |
| INS | Ywo | DX | | | | | | 6D | | W | | | gen | inout string | .d..... | | | |
| INSW | Ywo | DX | | | | | | | | | | | | | | | | |
| INS | Yv | DX | | | | | | 6D | | W | | | gen | inout string | .d..... | | | |
| INSD | Ydo | DX | | | | | | | | | | | | | | | | |
| INSERTPS | Vps | Ups | Ib | | sse41 | 66 | 0F | 3A | 21 | | r | | simdfp | datamov | | | | Insert Packed Single-FP Value |
| INSERTPS | Vps | Md | Ib | | | | | | | | | | | | | | | |
| INT alias | 3 | Fv | | | | | | CC | | | | | gen | break stack | | ..i..... | | |
| INT | Ib | Fv | | | | | | CD | | | | | gen | break stack | | ..i..... | | |
| INT1 part alias9 | Fv | | | | | | | F1 | | | | | gen | break stack | | ..i..... | | |
| ICEBP part alias9 | Fv | | | | | | | | | | | | | | | | | |
| INTO | Fv | | | | | | | CE | | | | | gen | break stack | o..... | ..i..... | | |
| INVD | | | | | | | 0F | 08 | | | | | system | | | | | Invalidate Internal Caches |
| INVEPT | Gq | Mdq | | | vmx | 66 | 0F | 38 | 80 | | r | | | | | o..szapc | o..szapc | Invalidate Translations Derived from EPT |
| INVLPG | M | | | | | | 0F | 01 | | | 7 | | system | | | | | Invalidate TLB Entry |
| INVVPID | Gq | Mdq | | | vmx | 66 | 0F | 38 | 81 | | r | | | | | o..szapc | o..szapc | Invalidate Translations Based on VPID |
| IRET | Fwo | | | | | | | CF | | | | | gen | break stack | | | | |
| IRETD | Fdo | | | | | | | | | | | | | | | | | |
| IRETQ | Fqp | | | | | | | | | | | | | | | | | |
| JB | Jbs | | | | | | | 72 | | ttT n | | | gen | branch | cond | | | Jump short if below/not above or equal/carry (CF=1) |
| JNAE | Jbs | | | | | | | | | | | | | | | | | |
| JC | Jbs | | | | | | | | | | | | | | | | | |
| JB | Jvds | | | | | | 0F | 82 | | ttT | | | gen | branch | cond | | | Jump short if below/not |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | n | | | | | | | | above or equal/carry (CF=1) |
| JNAE | Jvds | | | | | | | | | | | | | | | | | |
| JC | Jvds | | | | | | | | | | | | | | | | | |
| JBE | Jbs | | | | | | | 76 | | tT Tn | | | gen | branch | cond | | | Jump short if below or equal/not above (CF=1 AND ZF=1) |
| JNA | Jbs | | | | | | | | | | | | | | | | | |
| JBE | Jvds | | | | | | 0F | 86 | | tT Tn | | | gen | branch | cond | | | Jump short if below or equal/not above (CF=1 AND ZF=1) |
| JNA | Jvds | | | | | | | | | | | | | | | | | |
| JECXZ | Jbs | ECX | | | | | | E3 | | | | | gen | branch | cond | | | Jump short if rCX register is 0 |
| JRCXZ | Jbs | RCX | | | | | | | | | | | | | | | | |
| JL | Jbs | | | | | | | 7C | | TT tn | | | gen | branch | cond | | | Jump short if less/not greater (SF!=OF) |
| JNGE | Jbs | | | | | | | | | | | | | | | | | |
| JL | Jvds | | | | | | 0F | 8C | | TT tn | | | gen | branch | cond | | | Jump short if less/not greater (SF!=OF) |
| JNGE | Jvds | | | | | | | | | | | | | | | | | |
| JLE | Jbs | | | | | | | 7E | | TT Tn | | | gen | branch | cond | | | Jump short if less or equal/not greater ((ZF=1) OR (SF!=OF)) |
| JNG | Jbs | | | | | | | | | | | | | | | | | |
| JLE | Jvds | | | | | | 0F | 8E | | TT Tn | | | gen | branch | cond | | | Jump short if less or equal/not greater ((ZF=1) OR (SF!=OF)) |
| JNG | Jvds | | | | | | | | | | | | | | | | | |
| JMP | Jvds | | | | | | | E9 | | | | | gen | branch | | | | Jump |
| JMP | Jbs | | | | | | | EB | | | | | gen | branch | | | | Jump |
| JMP | Ev | | | | | | | FF | | | 4 | | gen | branch | | | | Jump |
| JMP | Eq | | | | | | | FF | | | 4 | | gen | branch | | | | Jump |
| JMPE | | | | | | | 0F | 00 | | | 6 | | system | branch | | | | Jump to IA-64 Instruction Set |
| JMPE | | | | | | | 0F | B8 | | | | | system | branch | | | | Jump to IA-64 Instruction Set |
| JMPF | Mptp | | | | | | | FF | | | 5 | | gen | branch | | | | Jump |
| JNB | Jbs | | | | | | | 73 | | ttT N | | | gen | branch | cond | | | Jump short if not below/above or equal/not carry (CF=0) |
| JAE | Jbs | | | | | | | | | | | | | | | | | |
| JNC | Jbs | | | | | | | | | | | | | | | | | |
| JNB | Jvds | | | | | | 0F | 83 | | ttT N | | | gen | branch | cond | | | Jump short if not below/above or equal/not carry (CF=0) |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| JAE | Jvds | | | | | | | | | | | | | | | | | |
| JNC | Jvds | | | | | | | | | | | | | | | | | |
| JNBE | Jbs | | | | | | | 77 | | tT TN | | | gen | branch | cond | | | Jump short if not below or equal/above (CF=0 AND ZF=0) |
| JA | Jbs | | | | | | | | | | | | | | | | | |
| JNBE | Jvds | | | | | | 0F | 87 | | tT TN | | | gen | branch | cond | | | Jump short if not below or equal/above (CF=0 AND ZF=0) |
| JA | Jvds | | | | | | | | | | | | | | | | | |
| JNL | Jbs | | | | | | | 7D | | TT tN | | | gen | branch | cond | | | Jump short if not less/greater or equal (SF=OF) |
| JGE | Jbs | | | | | | | | | | | | | | | | | |
| JNL | Jvds | | | | | | 0F | 8D | | TT tN | | | gen | branch | cond | | | Jump short if not less/greater or equal (SF=OF) |
| JGE | Jvds | | | | | | | | | | | | | | | | | |
| JNLE | Jbs | | | | | | | 7F | | TT TN | | | gen | branch | cond | | | Jump short if not less nor equal/greater ((ZF=0) AND (SF=OF)) |
| JG | Jbs | | | | | | | | | | | | | | | | | |
| JNLE | Jvds | | | | | | 0F | 8F | | TT TN | | | gen | branch | cond | | | Jump short if not less nor equal/greater ((ZF=0) AND (SF=OF)) |
| JG | Jvds | | | | | | | | | | | | | | | | | |
| JNO | Jbs | | | | | | | 71 | | ttt N | | | gen | branch | cond | | | Jump short if not overflow (OF=0) |
| JNO | Jvds | | | | | | 0F | 81 | | ttt N | | | gen | branch | cond | | | Jump short if not overflow (OF=0) |
| JNP | Jbs | | | | | | | 7B | | Tt TN | | | gen | branch | cond | | | Jump short if not parity/parity odd |
| JPO | Jbs | | | | | | | | | | | | | | | | | |
| JNP | Jvds | | | | | | 0F | 8B | | Tt TN | | | gen | branch | cond | | | Jump short if not parity/parity odd |
| JPO | Jvds | | | | | | | | | | | | | | | | | |
| JNS | Jbs | | | | | | | 79 | | Ttt N | | | gen | branch | cond | | | Jump short if not sign (SF=0) |
| JNS | Jvds | | | | | | 0F | 89 | | Ttt N | | | gen | branch | cond | | | Jump short if not sign (SF=0) |
| JNZ | Jbs | | | | | | | 75 | | tTt N | | | gen | branch | cond | | | Jump short if not zero/not equal (ZF=1) |
| JNE | Jbs | | | | | | | | | | | | | | | | | |
| JNZ | Jvds | | | | | | 0F | 85 | | tTt N | | | gen | branch | cond | | | Jump short if not zero/not equal (ZF=1) |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| JNE | Jvds | | | | | | | | | | | | | | | | | |
| JO | Jbs | | | | | | | 70 | | ttt<br>n | | | gen | branch | cond | | | Jump short if overflow (OF=1) |
| JO | Jvds | | | | | | 0F | 80 | | ttt<br>n | | | gen | branch | cond | | | Jump short if overflow (OF=1) |
| JP | Jbs | | | | | | | 7A | | Tt<br>Tn | | | gen | branch | cond | | | Jump short if parity/parity even (PF=1) |
| JPE | Jbs | | | | | | | | | | | | | | | | | |
| JP | Jvds | | | | | | 0F | 8A | | Tt<br>Tn | | | gen | branch | cond | | | Jump short if parity/parity even (PF=1) |
| JPE | Jvds | | | | | | | | | | | | | | | | | |
| JS | Jbs | | | | | | | 78 | | Ttt<br>n | | | gen | branch | cond | | | Jump short if sign (SF=1) |
| JS | Jvds | | | | | | 0F | 88 | | Ttt<br>n | | | gen | branch | cond | | | Jump short if sign (SF=1) |
| JZ | Jbs | | | | | | | 74 | | tTt<br>n | | | gen | branch | cond | | | Jump short if zero/equal (ZF=0) |
| JE | Jbs | | | | | | | | | | | | | | | | | |
| JZ | Jvds | | | | | | 0F | 84 | | tTt<br>n | | | gen | branch | cond | | | Jump short if zero/equal (ZF=0) |
| JE | Jvds | | | | | | | | | | | | | | | | | |
| LAHF | AH | | | | | | | 9F | | | | | gen | datamov<br>flgctrl | ...szap<br>c | | | |
| LAR | Gvqp | Mw | | | | | 0F | 02 | | | r | | system | | | ....z... | ....z... | Load Access Rights Byte |
| LAR | Gvqp | Rv | | | | | | | | | | | | | | | | |
| LDDQU | Vdq | Mdq | | | sse3 | F2 | 0F | F0 | | | r | | cachect | | | | | Load Unaligned Integer 128 Bits |
| LDMXCSR | Md | | | | sse1 | | 0F | AE | | | 2 | | mxcsrsm | | | | | Load MXCSR Register |
| LEA | Gvqp | M | | | | | | 8D | | | r | | gen | datamov | | | | Load Effective Address |
| LEAVE | rBP | | | | | | | C9 | | | | | gen | stack | | | | High Level Procedure Exit |
| LFENCE | | | | | sse2 | | 0F | AE | | | 5 | | order | | | | | Load Fence |
| LFS | FS | Gvqp | Mp<br>tp | | | | 0F | B4 | | Sr<br>e | r | | gen | datamov<br>segreg | | | | |
| LGDT | GDTR | Ms | | | | | 0F | 01 | | | 2 | | system | | | | | Load Global Descriptor Table Register |
| LGS | GS | Gvqp | Mp<br>tp | | | | 0F | B5 | | Sr<br>E | r | | gen | datamov<br>segreg | | | | |
| LIDT | IDTR | Ms | | | | | 0F | 01 | | | 3 | | system | | | | | Load Interrupt Descriptor Table Register |
| LLDT | LDTR | Ew | | | | | 0F | 00 | | | 2 | | system | | | | | Load Local Descriptor Table Register |
| LMSW | MSW | Ew | | | | | 0F | 01 | | | 6 | | system | | | | | Load Machine Status Word |
| LOCK | | | | | | F0 | | | | | | | prefix | | | | | Assert LOCK# Signal Prefix |
| LODS | AL | Xb | | | | | | AC | | w | | | gen | datamov<br>string | .d..... | | | |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| LODSB | AL | Xb | | | | | | | | | | | | | | | | |
| LODS | rAX | Xvqp | | | | | | AD | | W | | | gen | datamov string | .d..... | | | |
| LODSW | AX | Xwo | | | | | | | | | | | | | | | | |
| LODSD | EAX | Xdo | | | | | | | | | | | | | | | | |
| LODSQ | RAX | Xqp | | | | | | | | | | | | | | | | |
| LOOP | rCX | Jbs | | | | | | E2 | | | | | gen | branch | cond | | | Decrement count; Jump short if count!=0 |
| LOOPNZ | rCX | Jbs | | | | | | E0 | | | | | gen | branch | cond | | | Decrement count; Jump short if count!=0 and ZF=0 |
| LOOPNE | rCX | Jbs | | | | | | | | | | | | | | | | |
| LOOPZ | rCX | Jbs | | | | | | E1 | | | | | gen | branch | cond | | | Decrement count; Jump short if count!=0 and ZF=1 |
| LOOPE | rCX | Jbs | | | | | | | | | | | | | | | | |
| LSL | Gvqp | Mw | | | | | 0F | 03 | | | r | | system | | | ....z... | ....z... | Load Segment Limit |
| LSL | Gvqp | Rv | | | | | | | | | | | | | | | | |
| LSS | SS | Gvqp | Mp tp | | | | 0F | B2 | | sR e | r | | gen | datamov segreg | | | | |
| LTR | TR | Ew | | | | | 0F | 00 | | | 3 | | system | | | | | Load Task Register |
| MASKMOVDQU | BDdq | Vdq | Ud q | | sse2 | 66 | 0F | F7 | | | r | | cachect | | | | | Store Selected Bytes of Double Quadword |
| MASKMOVQ | BDq | Pq | Nq | | sse1 | | 0F | F7 | | | r | | cachect | | | | | Store Selected Bytes of Quadword |
| MAXPD | Vpd | Wpd | | | sse2 | 66 | 0F | 5F | | | r | | pcksclr | arith | | | | Return Maximum Packed Double-FP Values |
| MAXPS | Vps | Wps | | | sse1 | | 0F | 5F | | | r | | simdfp | arith | | | | Return Maximum Packed Single-FP Values |
| MAXSD | Vsd | Wsd | | | sse2 | F2 | 0F | 5F | | | r | | pcksclr | arith | | | | Return Maximum Scalar Double-FP Value |
| MAXSS | Vss | Wss | | | sse1 | F3 | 0F | 5F | | | r | | simdfp | arith | | | | Return Maximum Scalar Single-FP Value |
| MFENCE | | | | | sse2 | | 0F | AE | | | 6 | | order | | | | | Memory Fence |
| MINPD | Vpd | Wpd | | | sse2 | 66 | 0F | 5D | | | r | | pcksclr | arith | | | | Return Minimum Packed Double-FP Values |
| MINPS | Vps | Wps | | | sse1 | | 0F | 5D | | | r | | simdfp | arith | | | | Return Minimum Packed Single-FP Values |
| MINSD | Vsd | Wsd | | | sse2 | F2 | 0F | 5D | | | r | | pcksclr | arith | | | | Return Minimum Scalar Double-FP Value |
| MINSS | Vss | Wss | | | sse1 | F3 | 0F | 5D | | | r | | simdfp | arith | | | | Return Minimum Scalar Single-FP Value |
| MONITOR | BAb | ECX | EDX | | sse3 | | 0F | 01 | C8 | | 1 | | sync | | | | | Set Up Monitor Address |
| MOV | Eb | Gb | | | | | | 88 | | dw | r | | gen | datamov | | | | Move |
| MOV | Evqp | Gvqp | | | | | | 89 | | d W | r | | gen | datamov | | | | Move |
| MOV | Gb | Eb | | | | | | 8A | | D | r | | gen | datamov | | | | Move |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | w | | | | | | | | |
| MOV | Gvqp | Evqp | | | | | | 8B | | D w | r | | gen | datamov | | | | Move |
| MOV | Mw | Sw | | | | | | 8C | | d | r | | gen | datamov | | | | Move |
| MOV | Rvqp | Sw | | | | | | | | | | | | | | | | |
| MOV | Sw | Ew | | | | | | 8E | | D | r | | gen | datamov | | | | Move |
| MOV | AL | Ob | | | | | | A0 | | w | | | gen | datamov | | | | Move |
| MOV | rAX | Ovqp | | | | | | A1 | | W | | | gen | datamov | | | | Move |
| MOV | Ob | AL | | | | | | A2 | | w | | | gen | datamov | | | | Move |
| MOV | Ovqp | rAX | | | | | | A3 | | W | | | gen | datamov | | | | Move |
| MOV | Zb | Ib | | | | | | B0 | | +r | | | gen | datamov | | | | Move |
| MOV | Zvqp | Ivqp | | | | | | B8 | | +r | | | gen | datamov | | | | Move |
| MOV | Eb | Ib | | | | | | C6 | | w | 0 | | gen | datamov | | | | Move |
| MOV | Evqp | Ivds | | | | | | C7 | | W | 0 | | gen | datamov | | | | Move |
| MOV | Rq | Cq | | | | | 0F | 20 | | | r | | system | | | o..szapc | | Move to/from Control Registers |
| MOV | Hq | Cq | | | | | 0F | 20 | | | r | | system | | | o..szapc | | Move to/from Control Registers |
| MOV | Rq | Dq | | | | | 0F | 21 | | | r | | system | | | o..szapc | | Move to/from Debug Registers |
| MOV | Hq | Dq | | | | | 0F | 21 | | | r | | system | | | o..szapc | | Move to/from Debug Registers |
| MOV | Cq | Rq | | | | | 0F | 22 | | | r | | system | | | o..szapc | | Move to/from Control Registers |
| MOV | Cq | Hq | | | | | 0F | 22 | | | r | | system | | | o..szapc | | Move to/from Control Registers |
| MOV | Dq | Rq | | | | | 0F | 23 | | | r | | system | | | o..szapc | | Move to/from Debug Registers |
| MOV | Dq | Hq | | | | | 0F | 23 | | | r | | system | | | o..szapc | | Move to/from Debug Registers |
| MOVAPD | Vpd | Wpd | | | sse2 | 66 | 0F | 28 | | | r | | pcksclr | datamov | | | | Move Aligned Packed Double-FP Values |
| MOVAPD | Wpd | Vpd | | | sse2 | 66 | 0F | 29 | | | r | | pcksclr | datamov | | | | Move Aligned Packed Double-FP Values |
| MOVAPS | Vps | Wps | | | sse1 | | 0F | 28 | | | r | | simdfp | datamov | | | | Move Aligned Packed Single-FP Values |
| MOVAPS | Wps | Vps | | | sse1 | | 0F | 29 | | | r | | simdfp | datamov | | | | Move Aligned Packed Single-FP Values |
| MOVBE | Gvqp | Mvqp | | | | | 0F | 38 | F0 | | r | | gen | datamov | | | | Move Data After Swapping Bytes |
| MOVBE | Mvqp | Gvqp | | | | | 0F | 38 | F1 | | r | | gen | datamov | | | | Move Data After Swapping Bytes |
| MOVD | Pq | Ed | | | mmx | | 0F | 6E | | | r | | datamov | | | | | Move Doubleword/Quadword |
| MOVQ | Pq | Eqp | | | | | | | | | | | | | | | | |

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MOVD | Vdq | Ed | | | sse2 | 66 | 0F | 6E | | | r | | simdint | datamov | | | | Move Doubleword/Quadword |
| MOVQ | Vdq | Eqp | | | | | | | | | | | | | | | | |
| MOVD | Ed | Pq | | | mmx | | 0F | 7E | | | r | | datamov | | | | | Move Doubleword/Quadword |
| MOVQ | Eqp | Pq | | | | | | | | | | | | | | | | |
| MOVD | Ed | Vdq | | | sse2 | 66 | 0F | 7E | | | r | | simdint | datamov | | | | Move Doubleword/Quadword |
| MOVQ | Eqp | Edq | | | | | | | | | | | | | | | | |
| MOVDDUP | Vq | Wq | | | sse3 | F2 | 0F | 12 | | | r | | simdfp | datamov | | | | Move One Double-FP and Duplicate |
| MOVDQ2Q | Pq | Uq | | | sse2 | F2 | 0F | D6 | | | r | | simdint | datamov | | | | Move Quadword from XMM to MMX Technology Register |
| MOVDQA | Vdq | Wdq | | | sse2 | 66 | 0F | 6F | | | r | | simdint | datamov | | | | Move Aligned Double Quadword |
| MOVDQA | Wdq | Vdq | | | sse2 | 66 | 0F | 7F | | | r | | simdint | datamov | | | | Move Aligned Double Quadword |
| MOVDQU | Vdq | Wdq | | | sse2 | F3 | 0F | 6F | | | r | | simdint | datamov | | | | Move Unaligned Double Quadword |
| MOVDQU | Wdq | Vdq | | | sse2 | F3 | 0F | 7F | | | r | | simdint | datamov | | | | Move Unaligned Double Quadword |
| MOVHLPS | Vq | Uq | | | sse1 | | 0F | 12 | | | r | | simdfp | datamov | | | | Move Packed Single-FP Values High to Low |
| MOVHPD | Vq | Mq | | | sse2 | 66 | 0F | 16 | | | r | | pcksclr | datamov | | | | Move High Packed Double-FP Value |
| MOVHPD | Mq | Vq | | | sse2 | 66 | 0F | 17 | | | r | | pcksclr | datamov | | | | Move High Packed Double-FP Value |
| MOVHPS | Vq | Mq | | | sse1 | | 0F | 16 | | | r | | simdfp | datamov | | | | Move High Packed Single-FP Values |
| MOVHPS | Mq | Vq | | | sse1 | | 0F | 17 | | | r | | simdfp | datamov | | | | Move High Packed Single-FP Values |
| MOVLHPS | Vq | Uq | | | sse1 | | 0F | 16 | | | r | | simdfp | datamov | | | | Move Packed Single-FP Values Low to High |
| MOVLPD | Vq | Mq | | | sse2 | 66 | 0F | 12 | | | r | | pcksclr | datamov | | | | Move Low Packed Double-FP Value |
| MOVLPD | Mq | Vq | | | sse2 | 66 | 0F | 13 | | | r | | pcksclr | datamov | | | | Move Low Packed Double-FP Value |
| MOVLPS | Vq | Mq | | | sse1 | | 0F | 12 | | | r | | simdfp | datamov | | | | Move Low Packed Single-FP Values |
| MOVLPS | Mq | Vq | | | sse1 | | 0F | 13 | | | r | | simdfp | datamov | | | | Move Low Packed Single-FP Values |
| MOVMSKPD | Gdqp | Upd | | | sse2 | 66 | 0F | 50 | | | r | | pcksclr | datamov | | | | Extract Packed Double-FP Sign Mask |
| MOVMSKPS | Gdqp | Ups | | | sse1 | | 0F | 50 | | | r | | simdfp | datamov | | | | Extract Packed Single-FP |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | | | | | | | Sign Mask |
| MOVNTDQ | Mdq | Vdq | | | sse2 | 66 | 0F | E7 | | | r | | cachect | | | | | Store Double Quadword Using Non-Temporal Hint |
| MOVNTI | Mdqp | Gdqp | | | sse2 | | 0F | C3 | | | r | | cachect | | | | | Store Doubleword Using Non-Temporal Hint |
| MOVNTPD | Mpd | Vpd | | | sse2 | 66 | 0F | 2B | | | r | | cachect | | | | | Store Packed Double-FP Values Using Non-Temporal Hint |
| MOVNTPS | Mps | Vps | | | sse1 | | 0F | 2B | | | r | | cachect | | | | | Store Packed Single-FP Values Using Non-Temporal Hint |
| MOVNTQ | Mq | Pq | | | sse1 | | 0F | E7 | | | r | | cachect | | | | | Store of Quadword Using Non-Temporal Hint |
| MOVQ | Pq | Qq | | | mmx | | 0F | 6F | | | r | | datamov | | | | | Move Quadword |
| MOVQ | Vq | Wq | | | sse2 | F3 | 0F | 7E | | | r | | simdint | datamov | | | | Move Quadword |
| MOVQ | Qq | Pq | | | mmx | | 0F | 7F | | | r | | datamov | | | | | Move Quadword |
| MOVQ | Wq | Vq | | | sse2 | 66 | 0F | D6 | | | r | | simdint | datamov | | | | Move Quadword |
| MOVQ2DQ | Vdq | Nq | | | sse2 | F3 | 0F | D6 | | | r | | simdint | datamov | | | | Move Quadword from MMX Technology to XMM Register |
| MOVS | Yb | Xb | | | | | | A4 | | w | | | gen | datamov string | .d..... | | | |
| MOVSB | Yb | Xb | | | | | | | | | | | | | | | | |
| MOVS | Yvqp | Xvqp | | | | | | A5 | | W | | | gen | datamov string | .d..... | | | |
| MOVSW | Ywo | Xwo | | | | | | | | | | | | | | | | |
| MOVSD | Ydo | Xdo | | | | | | | | | | | | | | | | |
| MOVSQ | Yqp | Xqp | | | | | | | | | | | | | | | | |
| MOVSD | Vsd | Wsd | | | sse2 | F2 | 0F | 10 | | | r | | pcksclr | datamov | | | | Move Scalar Double-FP Value |
| MOVSD | Wsd | Vsd | | | sse2 | F2 | 0F | 11 | | | r | | pcksclr | datamov | | | | Move Scalar Double-FP Value |
| MOVSHDUP | Vq | Wq | | | sse3 | F3 | 0F | 16 | | | r | | simdfp | datamov | | | | Move Packed Single-FP High and Duplicate |
| MOVSLDUP | Vq | Wq | | | sse3 | F3 | 0F | 12 | | | r | | simdfp | datamov | | | | Move Packed Single-FP Low and Duplicate |
| MOVSS | Vss | Wss | | | sse1 | F3 | 0F | 10 | | | r | | simdfp | datamov | | | | Move Scalar Single-FP Values |
| MOVSS | Wss | Vss | | | sse1 | F3 | 0F | 11 | | | r | | simdfp | datamov | | | | Move Scalar Single-FP Values |
| MOVSX | Gvqp | Eb | | | | | 0F | BE | | D w | r | | gen | conver | | | | Move with Sign-Extension |
| MOVSX | Gvqp | Ew | | | | | 0F | BF | | D W | r | | gen | conver | | | | Move with Sign-Extension |
| MOVSXD | Gdqp | Ed | | | | | | 63 | | D | r | | gen | conver | | | | Move with Sign-Extension |

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MOVUPD | Vpd | Wpd | | | sse2 | 66 | 0F | 10 | | | r | | pcksclr | datamov | | | | Move Unaligned Packed Double-FP Value |
| MOVUPD | Wpd | Vpd | | | sse2 | 66 | 0F | 11 | | | r | | pcksclr | datamov | | | | Move Unaligned Packed Double-FP Values |
| MOVUPS | Vps | Wps | | | sse1 | | 0F | 10 | | | r | | simdfp | datamov | | | | Move Unaligned Packed Single-FP Values |
| MOVUPS | Wps | Vps | | | sse1 | | 0F | 11 | | | r | | simdfp | datamov | | | | Move Unaligned Packed Single-FP Values |
| MOVZX | Gvqp | Eb | | | | | 0F | B6 | | D w | r | | gen | conver | | | | Move with Zero-Extend |
| MOVZX | Gvqp | Ew | | | | | 0F | B7 | | D W | r | | gen | conver | | | | Move with Zero-Extend |
| MPSADBW | Vdq | Wdq | Ib | | sse41 | 66 | 0F | 3A | 42 | | r | | simdint | arith | | | | Compute Multiple Packed Sums of Absolute Difference |
| MUL | AX | AL | Eb | | | | | F6 | | w | 4 | | gen | arith | binary | o..szapc | o.....c | Unsigned Multiply |
| MUL | rDX | rAX | Evqp | | | | | F7 | | W | 4 | | gen | arith | binary | o..szapc | o.....c | Unsigned Multiply |
| MULPD | Vpd | Wpd | | | sse2 | 66 | 0F | 59 | | | r | | pcksclr | arith | | | | Multiply Packed Double-FP Values |
| MULPS | Vps | Wps | | | sse1 | | 0F | 59 | | | r | | simdfp | arith | | | | Multiply Packed Single-FP Values |
| MULSD | Vsd | Wsd | | | sse2 | F2 | 0F | 59 | | | r | | pcksclr | arith | | | | Multiply Scalar Double-FP Values |
| MULSS | Vss | Wss | | | sse1 | F3 | 0F | 59 | | | r | | simdfp | arith | | | | Multiply Scalar Single-FP Value |
| MWAIT | EAX | ECX | | | sse3 | | 0F | 01 | C9 | | 1 | | sync | | | | | Monitor Wait |
| NEG | Eb | | | | | | | F6 | | w | 3 | | gen | arith | binary | o..szapc | o..szapc | Two's Complement Negation |
| NEG | Evqp | | | | | | | F7 | | W | 3 | | gen | arith | binary | o..szapc | o..szapc | Two's Complement Negation |
| NOP | | | | | | | | 90 | | | | | gen | control | | | | No Operation |
| NOP | Ev | | | | | | 0F | 0D | | | | | gen | control | | | | No Operation |
| NOP | Ev | | | | | | 0F | 1F | | | 0 | | gen | control | | | | No Operation |
| NOT | Eb | | | | | | | F6 | | w | 2 | | gen | logical | | | | One's Complement Negation |
| NOT | Evqp | | | | | | | F7 | | W | 2 | | gen | logical | | | | One's Complement Negation |
| OR | Eb | Gb | | | | | | 08 | | dw | r | L | gen | logical | | o..szapc | o..sz.pc | Logical Inclusive OR |
| OR | Evqp | Gvqp | | | | | | 09 | | d W | r | L | gen | logical | | o..szapc | o..sz.pc | Logical Inclusive OR |
| OR | Gb | Eb | | | | | | 0A | | D w | r | | gen | logical | | o..szapc | o..sz.pc | Logical Inclusive OR |
| OR | Gvqp | Evqp | | | | | | 0B | | D W | r | | gen | logical | | o..szapc | o..sz.pc | Logical Inclusive OR |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| OR | AL | Ib | | | | | | 0C | | w | | | gen | logical | | o..szapc | o..sz.pc | Logical Inclusive OR |
| OR | rAX | Ivds | | | | | | 0D | | W | | | gen | logical | | o..szapc | o..sz.pc | Logical Inclusive OR |
| OR | Eb | Ib | | | | | | 80 | | w | 1 | L | gen | logical | | o..szapc | o..sz.pc | Logical Inclusive OR |
| OR | Evqp | Ivds | | | | | | 81 | | W | 1 | L | gen | logical | | o..szapc | o..sz.pc | Logical Inclusive OR |
| OR | Evqp | Ibs | | | | | | 83 | | S W | 1 | L | gen | logical | | o..szapc | o..sz.pc | Logical Inclusive OR |
| ORPD | Vpd | Wpd | | | sse2 | 66 | 0F | 56 | | | r | | pcksclr | logical | | | | Bitwise Logical OR of Double-FP Values |
| ORPS | Vps | Wps | | | sse1 | | 0F | 56 | | | r | | simdfp | logical | | | | Bitwise Logical OR of Single-FP Values |
| OUT | Ib | AL | | | | | | E6 | | w | | | gen | inout | | | | Output to Port |
| OUT | Ib | eAX | | | | | | E7 | | W | | | gen | inout | | | | Output to Port |
| OUT | DX | AL | | | | | | EE | | w | | | gen | inout | | | | Output to Port |
| OUT | DX | eAX | | | | | | EF | | W | | | gen | inout | | | | Output to Port |
| OUTS | DX | Xb | | | | | | 6E | | w | | | gen | inout string | .d..... | | | |
| OUTSB | DX | Xb | | | | | | | | | | | | | | | | |
| OUTS | DX | Xwo | | | | | | 6F | | W | | | gen | inout string | .d..... | | | |
| OUTSW | DX | Xwo | | | | | | | | | | | | | | | | |
| OUTS | DX | Xv | | | | | | 6F | | W | | | gen | inout string | .d..... | | | |
| OUTSD | DX | Xdo | | | | | | | | | | | | | | | | |
| PACKSSDW | Pq | Qq | | | mmx | | 0F | 6B | | | r | | conver | | | | | Pack with Signed Saturation |
| PACKSSDW | Vdq | Wdq | | | sse2 | 66 | 0F | 6B | | | r | | simdint | conver | | | | Pack with Signed Saturation |
| PACKSSWB | Pq | Qd | | | mmx | | 0F | 63 | | | r | | conver | | | | | Pack with Signed Saturation |
| PACKSSWB | Vdq | Wdq | | | sse2 | 66 | 0F | 63 | | | r | | simdint | conver | | | | Pack with Signed Saturation |
| PACKUSWB | Pq | Qq | | | mmx | | 0F | 67 | | | r | | conver | | | | | Pack with Unsigned Saturation |
| PACKUSWB | Vdq | Wdq | | | sse2 | 66 | 0F | 67 | | | r | | simdint | conver | | | | Pack with Unsigned Saturation |
| PADDB | Pq | Qq | | | mmx | | 0F | FC | | | r | | arith | | | | | Add Packed Integers |
| PADDB | Vdq | Wdq | | | sse2 | 66 | 0F | FC | | | r | | simdint | arith | | | | Add Packed Integers |
| PADDD | Pq | Qq | | | mmx | | 0F | FE | | | r | | arith | | | | | Add Packed Integers |
| PADDD | Vdq | Wdq | | | sse2 | 66 | 0F | FE | | | r | | simdint | arith | | | | Add Packed Integers |
| PADDQ | Pq | Qq | | | sse2 | | 0F | D4 | | | r | | simdint | arith | | | | Add Packed Quadword Integers |
| PADDQ | Vdq | Wdq | | | sse2 | 66 | 0F | D4 | | | r | | simdint | arith | | | | Add Packed Quadword Integers |
| PADDSB | Pq | Qq | | | mmx | | 0F | EC | | | r | | arith | | | | | Add Packed Signed Integers with Signed Saturation |
| PADDSB | Vdq | Wdq | | | sse2 | 66 | 0F | EC | | | r | | simdint | arith | | | | Add Packed Signed Integers with Signed Saturation |
| PADDSW | Pq | Qq | | | mmx | | 0F | ED | | | r | | arith | | | | | Add Packed Signed Integers with Signed Saturation |

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PADDSW | Vdq | Wdq | | | sse2 | 66 | 0F | ED | | | r | | simdint | arith | | | | Add Packed Signed Integers with Signed Saturation |
| PADDUSB | Pq | Qq | | | mmx | | 0F | DC | | | r | | arith | | | | | Add Packed Unsigned Integers with Unsigned Saturation |
| PADDUSB | Vdq | Wdq | | | sse2 | 66 | 0F | DC | | | r | | simdint | arith | | | | Add Packed Unsigned Integers with Unsigned Saturation |
| PADDUSW | Pq | Qq | | | mmx | | 0F | DD | | | r | | arith | | | | | Add Packed Unsigned Integers with Unsigned Saturation |
| PADDUSW | Vdq | Wdq | | | sse2 | 66 | 0F | DD | | | r | | simdint | arith | | | | Add Packed Unsigned Integers with Unsigned Saturation |
| PADDW | Pq | Qq | | | mmx | | 0F | FD | | | r | | arith | | | | | Add Packed Integers |
| PADDW | Vdq | Wdq | | | sse2 | 66 | 0F | FD | | | r | | simdint | arith | | | | Add Packed Integers |
| PALIGNR | Pq | Qq | | | ssse3 | | 0F | 3A | 0F | | r | | simdint | | | | | Packed Align Right |
| PALIGNR | Vdq | Wdq | | | ssse3 | 66 | 0F | 3A | 0F | | r | | simdint | | | | | Packed Align Right |
| PAND | Pq | Qd | | | mmx | | 0F | DB | | | r | | logical | | | | | Logical AND |
| PAND | Vdq | Wdq | | | sse2 | 66 | 0F | DB | | | r | | simdint | logical | | | | Logical AND |
| PANDN | Pq | Qq | | | mmx | | 0F | DF | | | r | | logical | | | | | Logical AND NOT |
| PANDN | Vdq | Wdq | | | sse2 | 66 | 0F | DF | | | r | | simdint | logical | | | | Logical AND NOT |
| PAUSE | | | | | sse2 | F3 | | 90 | | | | | cachect | | | | | Spin Loop Hint |
| PAVGB | Pq | Qq | | | sse1 | | 0F | E0 | | | r | | simdint | | | | | Average Packed Integers |
| PAVGB | Vdq | Wdq | | | sse1 | 66 | 0F | E0 | | | r | | simdint | | | | | Average Packed Integers |
| PAVGW | Pq | Qq | | | sse1 | | 0F | E3 | | | r | | simdint | | | | | Average Packed Integers |
| PAVGW | Vdq | Wdq | | | sse1 | 66 | 0F | E3 | | | r | | simdint | | | | | Average Packed Integers |
| PBLENDW | Vdq | Wdq | Ib | | sse41 | 66 | 0F | 3A | 0E | | r | | simdint | datamov | | | | Blend Packed Words |
| PCMPEQB | Pq | Qq | | | mmx | | 0F | 74 | | | r | | compar | | | | | Compare Packed Data for Equal |
| PCMPEQB | Vdq | Wdq | | | sse2 | 66 | 0F | 74 | | | r | | simdint | compar | | | | Compare Packed Data for Equal |
| PCMPEQD | Pq | Qq | | | mmx | | 0F | 76 | | | r | | compar | | | | | Compare Packed Data for Equal |
| PCMPEQD | Vdq | Wdq | | | sse2 | 66 | 0F | 76 | | | r | | simdint | compar | | | | Compare Packed Data for Equal |
| PCMPEQW | Pq | Qq | | | mmx | | 0F | 75 | | | r | | compar | | | | | Compare Packed Data for Equal |
| PCMPEQW | Vdq | Wdq | | | sse2 | 66 | 0F | 75 | | | r | | simdint | compar | | | | Compare Packed Data for Equal |
| PCMPESTRI | rCX | Vdq | Wdq | ... | sse42 | 66 | 0F | 3A | 61 | | r | | strtxt | | | o..szapc | o..szapc | Packed Compare Explicit Length Strings, Return Index |
| PCMPESTRM | XMM0 | Vdq | Wdq | ... | sse42 | 66 | 0F | 3A | 60 | | r | | strtxt | | | o..szapc | o..szapc | Packed Compare Explicit Length Strings, Return Mask |
| PCMPGTB | Pq | Qd | | | mmx | | 0F | 64 | | | r | | compar | | | | | Compare Packed Signed |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | | | | | | | Integers for Greater Than |
| PCMPGTB | Vdq | Wdq | | | sse2 | 66 | 0F | 64 | | | r | | simdint | compar | | | | Compare Packed Signed Integers for Greater Than |
| PCMPGTD | Pq | Qd | | | mmx | | 0F | 66 | | | r | | compar | | | | | Compare Packed Signed Integers for Greater Than |
| PCMPGTD | Vdq | Wdq | | | sse2 | 66 | 0F | 66 | | | r | | simdint | compar | | | | Compare Packed Signed Integers for Greater Than |
| PCMPGTW | Pq | Qd | | | mmx | | 0F | 65 | | | r | | compar | | | | | Compare Packed Signed Integers for Greater Than |
| PCMPGTW | Vdq | Wdq | | | sse2 | 66 | 0F | 65 | | | r | | simdint | compar | | | | Compare Packed Signed Integers for Greater Than |
| PCMPISTRI | rCX | Vdq | Wdq | Ib | sse42 | 66 | 0F | 3A | 63 | | r | | strtxt | | | o..szapc | o..szapc | Packed Compare Implicit Length Strings, Return Index |
| PCMPISTRM | XMM0 | Vdq | Wdq | Ib | sse42 | 66 | 0F | 3A | 62 | | r | | strtxt | | | o..szapc | o..szapc | Packed Compare Implicit Length Strings, Return Mask |
| PEXTRB | Mb | Vdq | Ib | | sse41 | 66 | 0F | 3A | 14 | | r | | simdint | datamov | | | | Extract Byte |
| PEXTRB | Rdqp | Vdq | Ib | | | | | | | | | | | | | | | |
| PEXTRD | Ed | Vdq | Ib | | sse41 | 66 | 0F | 3A | 16 | | r | | simdint | datamov | | | | Extract Dword/Qword |
| PEXTRQ | Eqp | Vdq | Ib | | | | | | | | | | | | | | | |
| PEXTRW | Mw | Vdq | Ib | | sse41 | 66 | 0F | 3A | 15 | | r | | simdint | datamov | | | | Extract Word |
| PEXTRW | Rdqp | Vdq | Ib | | | | | | | | | | | | | | | |
| PEXTRW | Gdqp | Nq | Ib | | sse1 | | 0F | C5 | | | r | | simdint | | | | | Extract Word |
| PEXTRW | Gdqp | Udq | Ib | | sse1 | 66 | 0F | C5 | | | r | | simdint | | | | | Extract Word |
| PINSRB | Vdq | Mb | Ib | | sse41 | 66 | 0F | 3A | 20 | | r | | simdint | datamov | | | | Insert Byte |
| PINSRB | Vdq | Rdqp | Ib | | | | | | | | | | | | | | | |
| PINSRD | Vdq | Ed | Ib | | sse41 | 66 | 0F | 3A | 22 | | r | | simdint | datamov | | | | Insert Dword/Qword |
| PINSRQ | Vdq | Eqp | Ib | | | | | | | | | | | | | | | |
| PINSRW | Pq | Rdqp | Ib | | sse1 | | 0F | C4 | | | r | | simdint | | | | | Insert Word |
| PINSRW | Pq | Mw | Ib | | | | | | | | | | | | | | | |
| PINSRW | Vdq | Rdqp | Ib | | sse1 | 66 | 0F | C4 | | | r | | simdint | | | | | Insert Word |
| PINSRW | Vdq | Mw | Ib | | | | | | | | | | | | | | | |
| PMADDWD | Pq | Qd | | | mmx | | 0F | F5 | | | r | | arith | | | | | Multiply and Add Packed Integers |
| PMADDWD | Vdq | Wdq | | | sse2 | 66 | 0F | F5 | | | r | | simdint | arith | | | | Multiply and Add Packed Integers |
| PMAXSW | Pq | Qq | | | sse1 | | 0F | EE | | | r | | simdint | | | | | Maximum of Packed Signed Word Integers |
| PMAXSW | Vdq | Wdq | | | sse1 | 66 | 0F | EE | | | r | | simdint | | | | | Maximum of Packed Signed Word Integers |
| PMAXUB | Pq | Qq | | | sse1 | | 0F | DE | | | r | | simdint | | | | | Maximum of Packed Unsigned Byte Integers |
| PMAXUB | Vdq | Wdq | | | sse1 | 66 | 0F | DE | | | r | | simdint | | | | | Maximum of Packed Unsigned Byte Integers |
| PMINSW | Pq | Qq | | | sse1 | | 0F | EA | | | r | | simdint | | | | | Minimum of Packed Signed Word Integers |

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PMINSW | Vdq | Wdq | | | sse1 | 66 | 0F | EA | | | r | | simdint | | | | | Minimum of Packed Signed Word Integers |
| PMINUB | Pq | Qq | | | sse1 | | 0F | DA | | | r | | simdint | | | | | Minimum of Packed Unsigned Byte Integers |
| PMINUB | Vdq | Wdq | | | sse1 | 66 | 0F | DA | | | r | | simdint | | | | | Minimum of Packed Unsigned Byte Integers |
| PMOVMSKB | Gdqp | Nq | | | sse1 | | 0F | D7 | | | r | | simdint | | | | | Move Byte Mask |
| PMOVMSKB | Gdqp | Udq | | | sse1 | 66 | 0F | D7 | | | r | | simdint | | | | | Move Byte Mask |
| PMULHUW | Pq | Qq | | | sse1 | | 0F | E4 | | | r | | simdint | | | | | Multiply Packed Unsigned Integers and Store High Result |
| PMULHUW | Vdq | Wdq | | | sse1 | 66 | 0F | E4 | | | r | | simdint | | | | | Multiply Packed Unsigned Integers and Store High Result |
| PMULHW | Pq | Qq | | | mmx | | 0F | E5 | | | r | | arith | | | | | Multiply Packed Signed Integers and Store High Result |
| PMULHW | Vdq | Wdq | | | sse2 | 66 | 0F | E5 | | | r | | simdint | arith | | | | Multiply Packed Signed Integers and Store High Result |
| PMULLW | Pq | Qq | | | mmx | | 0F | D5 | | | r | | arith | | | | | Multiply Packed Signed Integers and Store Low Result |
| PMULLW | Vdq | Wdq | | | sse2 | 66 | 0F | D5 | | | r | | simdint | arith | | | | Multiply Packed Signed Integers and Store Low Result |
| PMULUDQ | Pq | Qq | | | sse2 | | 0F | F4 | | | r | | simdint | arith | | | | Multiply Packed Unsigned DW Integers |
| PMULUDQ | Vdq | Wdq | | | sse2 | 66 | 0F | F4 | | | r | | simdint | arith | | | | Multiply Packed Unsigned DW Integers |
| POP | Zvq | | | | | | | 58 | | +r | | | gen | stack | | | | Pop a Value from the Stack |
| POP | Ev | | | | | | | 8F | | W | 0 | | gen | stack | | | | Pop a Value from the Stack |
| POP | Evq | | | | | | | 8F | | W | 0 | | gen | stack | | | | Pop a Value from the Stack |
| POP | FS | | | | | | 0F | A1 | | Sr e | | | gen | stack segreg | | | | |
| POP | GS | | | | | | 0F | A9 | | Sr E | | | gen | stack segreg | | | | |
| POPCNT | Gvqp | Evqp | | | | F3 | 0F | B8 | | | r | | gen | bit | | o..szapc | | Bit Population Count |
| POPF | Fws | | | | | | | 9D | | | | | gen | stack flgctrl | | | | |
| POPFQ | Fqs | | | | | | | | | | | | | | | | | |
| POR | Pq | Qq | | | mmx | | 0F | EB | | | r | | logical | | | | | Bitwise Logical OR |
| POR | Vdq | Wdq | | | sse2 | 66 | 0F | EB | | | r | | simdint | logical | | | | Bitwise Logical OR |
| PREFETCHNTA | Mb | | | | sse1 | | 0F | 18 | | | 0 | | fetch | | | | | Prefetch Data Into Caches |
| PREFETCHT0 | Mb | | | | sse1 | | 0F | 18 | | | 1 | | fetch | | | | | Prefetch Data Into Caches |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PREFETCHT1 | Mb | | | | sse1 | | 0F | 18 | | | 2 | | fetch | | | | | Prefetch Data Into Caches |
| PREFETCHT2 | Mb | | | | sse1 | | 0F | 18 | | | 3 | | fetch | | | | | Prefetch Data Into Caches |
| PSADBW | Pq | Qq | | | sse1 | | 0F | F6 | | | r | | simdint | | | | | Compute Sum of Absolute Differences |
| PSADBW | Vdq | Wdq | | | sse1 | 66 | 0F | F6 | | | r | | simdint | | | | | Compute Sum of Absolute Differences |
| PSHUFD | Vdq | Wdq | Ib | | sse2 | 66 | 0F | 70 | | | r | | simdint | shunpck | | | | Shuffle Packed Doublewords |
| PSHUFHW | Vdq | Wdq | Ib | | sse2 | F3 | 0F | 70 | | | r | | simdint | shunpck | | | | Shuffle Packed High Words |
| PSHUFLW | Vdq | Wdq | Ib | | sse2 | F2 | 0F | 70 | | | r | | simdint | shunpck | | | | Shuffle Packed Low Words |
| PSHUFW | Pq | Qq | Ib | | sse1 | | 0F | 70 | | | r | | simdint | | | | | Shuffle Packed Words |
| PSLLD | Nq | Ib | | | mmx | | 0F | 72 | | | 6 | | shift | | | | | Shift Packed Data Left Logical |
| PSLLD | Udq | Ib | | | sse2 | 66 | 0F | 72 | | | 6 | | shift | | | | | Shift Packed Data Left Logical |
| PSLLD | Pq | Qq | | | mmx | | 0F | F2 | | | r | | shift | | | | | Shift Packed Data Left Logical |
| PSLLD | Vdq | Wdq | | | sse2 | 66 | 0F | F2 | | | r | | simdint | shift | | | | Shift Packed Data Left Logical |
| PSLLDQ | Udq | Ib | | | sse2 | 66 | 0F | 73 | | | 7 | | simdint | shift | | | | Shift Double Quadword Left Logical |
| PSLLQ | Nq | Ib | | | mmx | | 0F | 73 | | | 6 | | shift | | | | | Shift Packed Data Left Logical |
| PSLLQ | Udq | Ib | | | sse2 | 66 | 0F | 73 | | | 6 | | shift | | | | | Shift Packed Data Left Logical |
| PSLLQ | Pq | Qq | | | mmx | | 0F | F3 | | | r | | shift | | | | | Shift Packed Data Left Logical |
| PSLLQ | Vdq | Wdq | | | sse2 | 66 | 0F | F3 | | | r | | simdint | shift | | | | Shift Packed Data Left Logical |
| PSLLW | Nq | Ib | | | mmx | | 0F | 71 | | | 6 | | shift | | | | | Shift Packed Data Left Logical |
| PSLLW | Udq | Ib | | | sse2 | 66 | 0F | 71 | | | 6 | | shift | | | | | Shift Packed Data Left Logical |
| PSLLW | Pq | Qq | | | mmx | | 0F | F1 | | | r | | shift | | | | | Shift Packed Data Left Logical |
| PSLLW | Vdq | Wdq | | | sse2 | 66 | 0F | F1 | | | r | | simdint | shift | | | | Shift Packed Data Left Logical |
| PSRAD | Nq | Ib | | | mmx | | 0F | 72 | | | 4 | | shift | | | | | Shift Packed Data Right Arithmetic |
| PSRAD | Udq | Ib | | | sse2 | 66 | 0F | 72 | | | 4 | | shift | | | | | Shift Packed Data Right Arithmetic |
| PSRAD | Pq | Qq | | | mmx | | 0F | E2 | | | r | | shift | | | | | Shift Packed Data Right Arithmetic |
| PSRAD | Vdq | Wdq | | | sse2 | 66 | 0F | E2 | | | r | | simdint | shift | | | | Shift Packed Data Right Arithmetic |

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PSRAW | Nq | Ib | | | mmx | | 0F | 71 | | | 4 | | shift | | | | | Shift Packed Data Right Arithmetic |
| PSRAW | Udq | Ib | | | sse2 | 66 | 0F | 71 | | | 4 | | shift | | | | | Shift Packed Data Right Arithmetic |
| PSRAW | Pq | Qq | | | mmx | | 0F | E1 | | | r | | shift | | | | | Shift Packed Data Right Arithmetic |
| PSRAW | Vdq | Wdq | | | sse2 | 66 | 0F | E1 | | | r | | simdint | shift | | | | Shift Packed Data Right Arithmetic |
| PSRLD | Nq | Ib | | | mmx | | 0F | 72 | | | 2 | | shift | | | | | Shift Double Quadword Right Logical |
| PSRLD | Udq | Ib | | | sse2 | 66 | 0F | 72 | | | 2 | | shift | | | | | Shift Double Quadword Right Logical |
| PSRLD | Pq | Qq | | | mmx | | 0F | D2 | | | r | | shift | | | | | Shift Packed Data Right Logical |
| PSRLD | Vdq | Wdq | | | sse2 | 66 | 0F | D2 | | | r | | simdint | shift | | | | Shift Packed Data Right Logical |
| PSRLDQ | Udq | Ib | | | sse2 | 66 | 0F | 73 | | | 3 | | simdint | shift | | | | Shift Double Quadword Right Logical |
| PSRLQ | Nq | Ib | | | mmx | | 0F | 73 | | | 2 | | shift | | | | | Shift Packed Data Right Logical |
| PSRLQ | Udq | Ib | | | sse2 | 66 | 0F | 73 | | | 2 | | shift | | | | | Shift Packed Data Right Logical |
| PSRLQ | Pq | Qq | | | mmx | | 0F | D3 | | | r | | shift | | | | | Shift Packed Data Right Logical |
| PSRLQ | Vdq | Wdq | | | sse2 | 66 | 0F | D3 | | | r | | simdint | shift | | | | Shift Packed Data Right Logical |
| PSRLW | Nq | Ib | | | mmx | | 0F | 71 | | | 2 | | shift | | | | | Shift Packed Data Right Logical |
| PSRLW | Udq | Ib | | | sse2 | 66 | 0F | 71 | | | 2 | | shift | | | | | Shift Packed Data Right Logical |
| PSRLW | Pq | Qq | | | mmx | | 0F | D1 | | | r | | shift | | | | | Shift Packed Data Right Logical |
| PSRLW | Vdq | Wdq | | | sse2 | 66 | 0F | D1 | | | r | | simdint | shift | | | | Shift Packed Data Right Logical |
| PSUBB | Pq | Qq | | | mmx | | 0F | F8 | | | r | | arith | | | | | Subtract Packed Integers |
| PSUBB | Vdq | Wdq | | | sse2 | 66 | 0F | F8 | | | r | | simdint | arith | | | | Subtract Packed Integers |
| PSUBD | Pq | Qq | | | mmx | | 0F | FA | | | r | | arith | | | | | Subtract Packed Integers |
| PSUBD | Vdq | Wdq | | | sse2 | 66 | 0F | FA | | | r | | simdint | arith | | | | Subtract Packed Integers |
| PSUBQ | Pq | Qq | | | sse2 | | 0F | FB | | | r | | simdint | arith | | | | Subtract Packed Quadword Integers |
| PSUBQ | Vdq | Wdq | | | sse2 | 66 | 0F | FB | | | r | | simdint | arith | | | | Subtract Packed Quadword Integers |
| PSUBSB | Pq | Qq | | | mmx | | 0F | E8 | | | r | | arith | | | | | Subtract Packed Signed Integers with Signed Saturation |

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PSUBSB | Vdq | Wdq | | | sse2 | 66 | 0F | E8 | | | r | | simdint | arith | | | | Subtract Packed Signed Integers with Signed Saturation |
| PSUBSW | Pq | Qq | | | mmx | | 0F | E9 | | | r | | arith | | | | | Subtract Packed Signed Integers with Signed Saturation |
| PSUBSW | Vdq | Wdq | | | sse2 | 66 | 0F | E9 | | | r | | simdint | arith | | | | Subtract Packed Signed Integers with Signed Saturation |
| PSUBUSB | Pq | Qq | | | mmx | | 0F | D8 | | | r | | arith | | | | | Subtract Packed Unsigned Integers with Unsigned Saturation |
| PSUBUSB | Vdq | Wdq | | | sse2 | 66 | 0F | D8 | | | r | | simdint | arith | | | | Subtract Packed Unsigned Integers with Unsigned Saturation |
| PSUBUSW | Pq | Qq | | | mmx | | 0F | D9 | | | r | | arith | | | | | Subtract Packed Unsigned Integers with Unsigned Saturation |
| PSUBUSW | Vdq | Wdq | | | sse2 | 66 | 0F | D9 | | | r | | simdint | arith | | | | Subtract Packed Unsigned Integers with Unsigned Saturation |
| PSUBW | Pq | Qq | | | mmx | | 0F | F9 | | | r | | arith | | | | | Subtract Packed Integers |
| PSUBW | Vdq | Wdq | | | sse2 | 66 | 0F | F9 | | | r | | simdint | arith | | | | Subtract Packed Integers |
| PUNPCKHBW | Pq | Qq | | | mmx | | 0F | 68 | | | r | | unpack | | | | | Unpack High Data |
| PUNPCKHBW | Vdq | Wdq | | | sse2 | 66 | 0F | 68 | | | r | | simdint | shunpck | | | | Unpack High Data |
| PUNPCKHDQ | Pq | Qq | | | mmx | | 0F | 6A | | | r | | unpack | | | | | Unpack High Data |
| PUNPCKHDQ | Vdq | Wdq | | | sse2 | 66 | 0F | 6A | | | r | | simdint | shunpck | | | | Unpack High Data |
| PUNPCKHQDQ | Vdq | Wdq | | | sse2 | 66 | 0F | 6D | | | r | | simdint | shunpck | | | | Unpack High Data |
| PUNPCKHWD | Pq | Qq | | | mmx | | 0F | 69 | | | r | | unpack | | | | | Unpack High Data |
| PUNPCKHWD | Vdq | Wdq | | | sse2 | 66 | 0F | 69 | | | r | | simdint | shunpck | | | | Unpack High Data |
| PUNPCKLBW | Pq | Qd | | | mmx | | 0F | 60 | | | r | | unpack | | | | | Unpack Low Data |
| PUNPCKLBW | Vdq | Wdq | | | sse2 | 66 | 0F | 60 | | | r | | simdint | shunpck | | | | Unpack Low Data |
| PUNPCKLDQ | Pq | Qd | | | mmx | | 0F | 62 | | | r | | unpack | | | | | Unpack Low Data |
| PUNPCKLDQ | Vdq | Wdq | | | sse2 | 66 | 0F | 62 | | | r | | simdint | shunpck | | | | Unpack Low Data |
| PUNPCKLQDQ | Vdq | Wdq | | | sse2 | 66 | 0F | 6C | | | r | | simdint | shunpck | | | | Unpack Low Data |
| PUNPCKLWD | Pq | Qd | | | mmx | | 0F | 61 | | | r | | unpack | | | | | Unpack Low Data |
| PUNPCKLWD | Vdq | Wdq | | | sse2 | 66 | 0F | 61 | | | r | | simdint | shunpck | | | | Unpack Low Data |
| PUSH | Zvq | | | | | | | 50 | | +r | | | gen | stack | | | | Push Word, Doubleword or Quadword Onto the Stack |
| PUSH | Ivs | | | | | | | 68 | | | | | gen | stack | | | | Push Word, Doubleword or Quadword Onto the Stack |
| PUSH | Ibss | | | | | | | 6A | | S | | | gen | stack | | | | Push Word, Doubleword or Quadword Onto the Stack |
| PUSH | Ev | | | | | | | FF | | | 6 | | gen | stack | | | | Push Word, Doubleword or Quadword Onto the Stack |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PUSH | Evq | | | | | | | FF | | | 6 | | gen | stack | | | | Push Word, Doubleword or Quadword Onto the Stack |
| PUSH | FS | | | | | | 0F | A0 | | Sre | | | gen | stack segreg | | | | |
| PUSH | GS | | | | | | 0F | A8 | | SrE | | | gen | stack segreg | | | | |
| PUSHF | Fws | | | | | | | 9C | | | | | gen | stack flgctrl | | | | |
| PUSHFQ | Fqs | | | | | | | | | | | | | | | | | |
| PXOR | Pq | Qq | | | mmx | | 0F | EF | | | r | | logical | | | | | Logical Exclusive OR |
| PXOR | Vdq | Wdq | | | sse2 | 66 | 0F | EF | | | r | | simdint | logical | | | | Logical Exclusive OR |
| RCL | Eb | Ib | | | | | | C0 | | w | 2 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| RCL | Evqp | Ib | | | | | | C1 | | W | 2 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| RCL | Eb | 1 | | | | | | D0 | | w | 2 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| RCL | Evqp | 1 | | | | | | D1 | | W | 2 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| RCL | Eb | CL | | | | | | D2 | | w | 2 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| RCL | Evqp | CL | | | | | | D3 | | W | 2 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| RCPPS | Vps | Wps | | | sse1 | | 0F | 53 | | | r | | simdfp | arith | | | | Compute Reciprocals of Packed Single-FP Values |
| RCPSS | Vss | Wss | | | sse1 | F3 | 0F | 53 | | | r | | simdfp | arith | | | | Compute Reciprocal of Scalar Single-FP Values |
| RCR | Eb | Ib | | | | | | C0 | | w | 3 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| RCR | Evqp | Ib | | | | | | C1 | | W | 3 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| RCR | Eb | 1 | | | | | | D0 | | w | 3 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| RCR | Evqp | 1 | | | | | | D1 | | W | 3 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| RCR | Eb | CL | | | | | | D2 | | w | 3 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| RCR | Evqp | CL | | | | | | D3 | | W | 3 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| RDMSR | rAX | rDX | rCX | MSR | | | 0F | 32 | | | | | system | | | | | Read from Model Specific Register |
| RDPMC | EAX | EDX | PMC | | | | 0F | 33 | | | | | system | | | | | Read Performance-Monitoring Counters |
| RDTSC | EAX | EDX | I... | | | | 0F | 31 | | | | | system | | | | | Read Time-Stamp Counter |
| RDTSCP | EAX | EDX | ECX | ... | | | 0F | 01 | F9 | | 7 | | system | | | | | Read Time-Stamp Counter and Processor ID |
| REP | rCX | | | | | F2 | | | | | | | prefix | string | | | | Repeat String Operation Prefix |
| REP | rCX | | | | | F3 | | | | | | | prefix | string | | | | Repeat String Operation Prefix |
| REPNZ | rCX | | | | | F2 | | | | | | | prefix | string | | | | Repeat String Operation Prefix |
| REPNE | rCX | | | | | | | | | | | | | | | | | |
| REPZ | rCX | | | | | F3 | | | | | | | prefix | string | | | | Repeat String Operation Prefix |
| REPE | rCX | | | | | | | | | | | | | | | | | |
| RETF | Iw | | | | | | | CA | | | | | gen | branch | | | | |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | | | stack | | | | |
| RETF | | | | | | | | CB | | | | | gen | branch stack | | | | |
| RETN | Iw | | | | | | | C2 | | | | | gen | branch stack | | | | |
| RETN | | | | | | | | C3 | | | | | gen | branch stack | | | | |
| REX | | | | | | 40 | | | | | | | prefix | | | | | Access to new 8-bit registers |
| REX.B | | | | | | 41 | | | | | | | prefix | | | | | Extension of r/m field, base field, or opcode reg field |
| REX.R | | | | | | 44 | | | | | | | prefix | | | | | Extension of ModR/M reg field |
| REX.RB | | | | | | 45 | | | | | | | prefix | | | | | REX.R and REX.B combination |
| REX.RX | | | | | | 46 | | | | | | | prefix | | | | | REX.R and REX.X combination |
| REX.RXB | | | | | | 47 | | | | | | | prefix | | | | | REX.R, REX.X and REX.B combination |
| REX.W | | | | | | 48 | | | | | | | prefix | | | | | 64 Bit Operand Size |
| REX.WB | | | | | | 49 | | | | | | | prefix | | | | | REX.W and REX.B combination |
| REX.WR | | | | | | 4C | | | | | | | prefix | | | | | REX.W and REX.R combination |
| REX.WRB | | | | | | 4D | | | | | | | prefix | | | | | REX.W, REX.R and REX.B combination |
| REX.WRX | | | | | | 4E | | | | | | | prefix | | | | | REX.W, REX.R and REX.X combination |
| REX.WRXB | | | | | | 4F | | | | | | | prefix | | | | | REX.W, REX.R, REX.X and REX.B combination |
| REX.WX | | | | | | 4A | | | | | | | prefix | | | | | REX.W and REX.X combination |
| REX.WXB | | | | | | 4B | | | | | | | prefix | | | | | REX.W, REX.X and REX.B combination |
| REX.X | | | | | | 42 | | | | | | | prefix | | | | | Extension of SIB index field |
| REX.XB | | | | | | 43 | | | | | | | prefix | | | | | REX.X and REX.B combination |
| ROL | Eb | Ib | | | | | | C0 | | w | 0 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| ROL | Evqp | Ib | | | | | | C1 | | W | 0 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| ROL | Eb | 1 | | | | | | D0 | | w | 0 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| ROL | Evqp | 1 | | | | | | D1 | | W | 0 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| ROL | Eb | CL | | | | | | D2 | | w | 0 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| ROL | Evqp | CL | | | | | | D3 | | W | 0 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| ROR | Eb | Ib | | | | | | C0 | | w | 1 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| ROR | Evqp | Ib | | | | | | C1 | | W | 1 | | gen | shftrot | | o..szapc | o..szapc | Rotate |

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ROR | Eb | 1 | | | | | | D0 | | w | 1 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| ROR | Evqp | 1 | | | | | | D1 | | W | 1 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| ROR | Eb | CL | | | | | | D2 | | w | 1 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| ROR | Evqp | CL | | | | | | D3 | | W | 1 | | gen | shftrot | | o..szapc | o..szapc | Rotate |
| ROUNDPD | Vps | Wpd | Ib | | sse41 | 66 | 0F | 3A | 09 | | r | | simdfp | conver | | | | Round Packed Double-FP Values |
| ROUNDPS | Vps | Wps | Ib | | sse41 | 66 | 0F | 3A | 08 | | r | | simdfp | conver | | | | Round Packed Single-FP Values |
| ROUNDSD | Vsd | Wsd | Ib | | sse41 | 66 | 0F | 3A | 0B | | r | | simdfp | conver | | | | Round Scalar Double-FP Values |
| ROUNDSS | Vss | Wss | Ib | | sse41 | 66 | 0F | 3A | 0A | | r | | simdfp | conver | | | | Round Scalar Single-FP Values |
| RSM | Fw | | | | | | 0F | AA | | | | | system | branch | | | | Resume from System Management Mode |
| RSQRTPS | Vps | Wps | | | sse1 | | 0F | 52 | | | r | | simdfp | arith | | | | Compute Recipr. of Square Roots of Packed Single-FP Values |
| RSQRTSS | Vss | Wss | | | sse1 | F3 | 0F | 52 | | | r | | simdfp | arith | | | | Compute Recipr. of Square Root of Scalar Single-FP Value |
| SAHF | AH | | | | | | | 9E | | | | | gen | datamov flgctrl | | ...szapc | | |
| SAL alias | Eb | Ib | | | | | | C0 | | w | 6 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SHL alias | Eb | Ib | | | | | | | | | | | | | | | | |
| SAL alias | Evqp | Ib | | | | | | C1 | | w | 6 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SHL alias | Evqp | Ib | | | | | | | | | | | | | | | | |
| SAL alias | Eb | 1 | | | | | | D0 | | w | 6 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SHL alias | Eb | 1 | | | | | | | | | | | | | | | | |
| SAL alias | Evqp | 1 | | | | | | D1 | | W | 6 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SHL alias | Evqp | 1 | | | | | | | | | | | | | | | | |
| SAL alias | Eb | CL | | | | | | D2 | | w | 6 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SHL alias | Eb | CL | | | | | | | | | | | | | | | | |
| SAL alias | Evqp | CL | | | | | | D3 | | W | 6 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SHL alias | Evqp | CL | | | | | | | | | | | | | | | | |
| SAR | Eb | Ib | | | | | | C0 | | w | 7 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SAR | Evqp | Ib | | | | | | C1 | | W | 7 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SAR | Eb | 1 | | | | | | D0 | | w | 7 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SAR | Evqp | 1 | | | | | | D1 | | W | 7 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SAR | Eb | CL | | | | | | D2 | | w | 7 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SAR | Evqp | CL | | | | | | D3 | | W | 7 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SBB | Eb | Gb | | | | | | 18 | | dw | r | L | gen | arith | binary | o..szapc | o..szapc | Integer Subtraction with Borrow |
| SBB | Evqp | Gvqp | | | | | | 19 | | dW | r | L | gen | arith | binary | o..szapc | o..szapc | Integer Subtraction with Borrow |
| SBB | Gb | Eb | | | | | | 1A | | D | r | | gen | arith | binary | o..szapc | o..szapc | Integer Subtraction with |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | w | | | | | | | | Borrow |
| SBB | Gvqp | Evqp | | | | | | 1B | | D W | r | | gen | arith | binary | o..szapc | o..szapc | Integer Subtraction with Borrow |
| SBB | AL | Ib | | | | | | 1C | | w | | | gen | arith | binary | o..szapc | o..szapc | Integer Subtraction with Borrow |
| SBB | rAX | Ivds | | | | | | 1D | | W | | | gen | arith | binary | o..szapc | o..szapc | Integer Subtraction with Borrow |
| SBB | Eb | Ib | | | | | | 80 | | w | 3 | L | gen | arith | binary | o..szapc | o..szapc | Integer Subtraction with Borrow |
| SBB | Evqp | Ivds | | | | | | 81 | | W | 3 | L | gen | arith | binary | o..szapc | o..szapc | Integer Subtraction with Borrow |
| SBB | Evqp | Ibs | | | | | | 83 | | S W | 3 | L | gen | arith | binary | o..szapc | o..szapc | Integer Subtraction with Borrow |
| SCAS | Yb | AL | | | | | | AE | | w | | | gen | arith string | binary | o..szapc | o..szapc | Scan String |
| SCASB | Yb | AL | | | | | | | | | | | | | | | | |
| SCAS | Yvqp | rAX | | | | | | AF | | W | | | gen | arith string | binary | o..szapc | o..szapc | Scan String |
| SCASW | Ywo | AX | | | | | | | | | | | | | | | | |
| SCASD | Ydo | EAX | | | | | | | | | | | | | | | | |
| SCASQ | Yqp | RAX | | | | | | | | | | | | | | | | |
| SETB | Eb | | | | | | 0F | 92 | | ttT n | 0 | | gen | datamov | | | | Set Byte on Condition - below/not above or equal/carry (CF=1) |
| SETNAE | Eb | | | | | | | | | | | | | | | | | |
| SETC | Eb | | | | | | | | | | | | | | | | | |
| SETBE | Eb | | | | | | 0F | 96 | | tT Tn | 0 | | gen | datamov | | | | Set Byte on Condition - below or equal/not above (CF=1 AND ZF=1) |
| SETNA | Eb | | | | | | | | | | | | | | | | | |
| SETL | Eb | | | | | | 0F | 9C | | TT tn | 0 | | gen | datamov | | | | Set Byte on Condition - less/not greater (SF!=OF) |
| SETNGE | Eb | | | | | | | | | | | | | | | | | |
| SETLE | Eb | | | | | | 0F | 9E | | TT Tn | 0 | | gen | datamov | | | | Set Byte on Condition - less or equal/not greater ((ZF=1) OR (SF!=OF)) |
| SETNG | Eb | | | | | | | | | | | | | | | | | |
| SETNB | Eb | | | | | | 0F | 93 | | ttT N | 0 | | gen | datamov | | | | Set Byte on Condition - not below/above or equal/not carry (CF=0) |
| SETAE | Eb | | | | | | | | | | | | | | | | | |
| SETNC | Eb | | | | | | | | | | | | | | | | | |
| SETNBE | Eb | | | | | | 0F | 97 | | tT TN | 0 | | gen | datamov | | | | Set Byte on Condition - not below or equal/above (CF=0 AND ZF=0) |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SETA | Eb | | | | | | | | | | | | | | | | | |
| SETNL | Eb | | | | | | 0F | 9D | | TTtN | 0 | | gen | datamov | | | | Set Byte on Condition - not less/greater or equal (SF=OF) |
| SETGE | Eb | | | | | | | | | | | | | | | | | |
| SETNLE | Eb | | | | | | 0F | 9F | | TTTN | 0 | | gen | datamov | | | | Set Byte on Condition - not less nor equal/greater ((ZF=0) AND (SF=OF)) |
| SETG | Eb | | | | | | | | | | | | | | | | | |
| SETNO | Eb | | | | | | 0F | 91 | | tttN | 0 | | gen | datamov | | | | Set Byte on Condition - not overflow (OF=0) |
| SETNP | Eb | | | | | | 0F | 9B | | TtTN | 0 | | gen | datamov | | | | Set Byte on Condition - not parity/parity odd |
| SETPO | Eb | | | | | | | | | | | | | | | | | |
| SETNS | Eb | | | | | | 0F | 99 | | TttN | 0 | | gen | datamov | | | | Set Byte on Condition - not sign (SF=0) |
| SETNZ | Eb | | | | | | 0F | 95 | | tTtN | 0 | | gen | datamov | | | | Set Byte on Condition - not zero/not equal (ZF=1) |
| SETNE | Eb | | | | | | | | | | | | | | | | | |
| SETO | Eb | | | | | | 0F | 90 | | tttn | 0 | | gen | datamov | | | | Set Byte on Condition - overflow (OF=1) |
| SETP | Eb | | | | | | 0F | 9A | | TtTn | 0 | | gen | datamov | | | | Set Byte on Condition - parity/parity even (PF=1) |
| SETPE | Eb | | | | | | | | | | | | | | | | | |
| SETS | Eb | | | | | | 0F | 98 | | Tttn | 0 | | gen | datamov | | | | Set Byte on Condition - sign (SF=1) |
| SETZ | Eb | | | | | | 0F | 94 | | tTtn | 0 | | gen | datamov | | | | Set Byte on Condition - zero/equal (ZF=0) |
| SETE | Eb | | | | | | | | | | | | | | | | | |
| SFENCE | | | | | | sse1 | | 0F | AE | | | 7 | | order | | | | | Store Fence |
| SGDT | Ms | GDTR | | | | | | 0F | 01 | | | 0 | | system | | | | | Store Global Descriptor Table Register |
| SHL | Eb | Ib | | | | | | | C0 | | w | 4 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SAL | Eb | Ib | | | | | | | | | | | | | | | | | |
| SHL | Evqp | Ib | | | | | | | C1 | | W | 4 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SAL | Evqp | Ib | | | | | | | | | | | | | | | | | |
| SHL | Eb | 1 | | | | | | | D0 | | w | 4 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SAL | Eb | 1 | | | | | | | | | | | | | | | | | |
| SHL | Evqp | 1 | | | | | | | D1 | | W | 4 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SAL | Evqp | 1 | | | | | | | | | | | | | | | | | |
| SHL | Eb | CL | | | | | | | D2 | | w | 4 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SAL | Eb | CL | | | | | | | | | | | | | | | | | |
| SHL | Evqp | CL | | | | | | | D3 | | W | 4 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SAL | Evqp | CL | | | | | | | | | | | | | | | | | |
| SHLD | Evqp | Gvqp | Ib | | | | | 0F | A4 | | d | r | | gen | shftrot | | o..szapc | o..sz.pc | Double Precision Shift Left |

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SHLD | Evqp | Gvqp | CL | | | | 0F | A5 | | d | r | | gen | shftrot | | o..szapc | o..sz.pc | Double Precision Shift Left |
| SHR | Eb | Ib | | | | | | C0 | | w | 5 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SHR | Evqp | Ib | | | | | | C1 | | W | 5 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SHR | Eb | 1 | | | | | | D0 | | w | 5 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SHR | Evqp | 1 | | | | | | D1 | | W | 5 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SHR | Eb | CL | | | | | | D2 | | w | 5 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SHR | Evqp | CL | | | | | | D3 | | W | 5 | | gen | shftrot | | o..szapc | o..sz.pc | Shift |
| SHRD | Evqp | Gvqp | Ib | | | | 0F | AC | | d | r | | gen | shftrot | | o..szapc | o..sz.pc | Double Precision Shift Right |
| SHRD | Evqp | Gvqp | CL | | | | 0F | AD | | d | r | | gen | shftrot | | o..szapc | o..sz.pc | Double Precision Shift Right |
| SHUFPD | Vpd | Wpd | Ib | | sse2 | 66 | 0F | C6 | | | r | | pcksclr | shunpck | | | | Shuffle Packed Double-FP Values |
| SHUFPS | Vps | Wps | Ib | | sse1 | | 0F | C6 | | | r | | simdfp | shunpck | | | | Shuffle Packed Single-FP Values |
| SIDT | Ms | IDTR | | | | | 0F | 01 | | | 1 | | system | | | | | Store Interrupt Descriptor Table Register |
| SLDT | Mw | LDTR | | | | | 0F | 00 | | | 0 | | system | | | | | Store Local Descriptor Table Register |
| SLDT | Rvqp | LDTR | | | | | | | | | | | | | | | | |
| SMSW | Mw | MSW | | | | | 0F | 01 | | | 4 | | system | | | | | Store Machine Status Word |
| SMSW | Rvqp | MSW | | | | | | | | | | | | | | | | |
| SQRTPD | Vpd | Wpd | | | sse2 | 66 | 0F | 51 | | | r | | pcksclr | arith | | | | Compute Square Roots of Packed Double-FP Values |
| SQRTPS | Vps | Wps | | | sse1 | | 0F | 51 | | | r | | simdfp | arith | | | | Compute Square Roots of Packed Single-FP Values |
| SQRTSD | Vsd | Wsd | | | sse2 | F2 | 0F | 51 | | | r | | pcksclr | arith | | | | Compute Square Root of Scalar Double-FP Value |
| SQRTSS | Vss | Wss | | | sse1 | F3 | 0F | 51 | | | r | | simdfp | arith | | | | Compute Square Root of Scalar Single-FP Value |
| STC | | | | | | | | F9 | | | | | gen | flgctrl | | .......c | .......c | Set Carry Flag |
| STD | | | | | | | | FD | | | | | gen | flgctrl | | .d..... | .d..... | Set Direction Flag |
| STI | | | | | | | | FB | | | | | gen | flgctrl | | ..i..... | ..i..... | Set Interrupt Flag |
| STMXCSR | Md | | | | sse1 | | 0F | AE | | | 3 | | mxcsrsm | | | | | Store MXCSR Register State |
| STOS | Yb | AL | | | | | | AA | | w | | | gen | datamov string | .d..... | | | |
| STOSB | Yb | AL | | | | | | | | | | | | | | | | |
| STOS | Yvqp | rAX | | | | | | AB | | W | | | gen | datamov string | .d..... | | | |
| STOSW | Ywo | AX | | | | | | | | | | | | | | | | |
| STOSD | Ydo | EAX | | | | | | | | | | | | | | | | |
| STOSQ | Yqp | RAX | | | | | | | | | | | | | | | | |
| STR | Mw | TR | | | | | 0F | 00 | | | 1 | | system | | | | | Store Task Register |
| STR | Rvqp | TR | | | | | | | | | | | | | | | | |
| SUB | Eb | Gb | | | | | | 28 | | dw | r | L | gen | arith | binary | o..szapc | o..szapc | Subtract |
| SUB | Evqp | Gvqp | | | | | | 29 | | d W | r | L | gen | arith | binary | o..szapc | o..szapc | Subtract |

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SUB | Gb | Eb | | | | | | 2A | | Dw | r | | gen | arith | binary | o..szapc | o..szapc | Subtract |
| SUB | Gvqp | Evqp | | | | | | 2B | | DW | r | | gen | arith | binary | o..szapc | o..szapc | Subtract |
| SUB | AL | Ib | | | | | | 2C | | w | | | gen | arith | binary | o..szapc | o..szapc | Subtract |
| SUB | rAX | Ivds | | | | | | 2D | | W | | | gen | arith | binary | o..szapc | o..szapc | Subtract |
| SUB | Eb | Ib | | | | | | 80 | | w | 5 | L | gen | arith | binary | o..szapc | o..szapc | Subtract |
| SUB | Evqp | Ivds | | | | | | 81 | | W | 5 | L | gen | arith | binary | o..szapc | o..szapc | Subtract |
| SUB | Evqp | Ibs | | | | | | 83 | | SW | 5 | L | gen | arith | binary | o..szapc | o..szapc | Subtract |
| SUBPD | Vpd | Wpd | | | sse2 | 66 | 0F | 5C | | | r | | pcksclr | arith | | | | Subtract Packed Double-FP Values |
| SUBPS | Vps | Wps | | | sse1 | | 0F | 5C | | | r | | simdfp | arith | | | | Subtract Packed Single-FP Values |
| SUBSD | Vsd | Wsd | | | sse2 | F2 | 0F | 5C | | | r | | pcksclr | arith | | | | Subtract Scalar Double-FP Values |
| SUBSS | Vss | Wss | | | sse1 | F3 | 0F | 5C | | | r | | simdfp | arith | | | | Subtract Scalar Single-FP Values |
| SWAPGS | GS | I... | | | | | 0F | 01 | F8 | | 7 | | system | | | | | Swap GS Base Register |
| SYSCALL | RCX | R11 | SS | ... | | | 0F | 05 | | | | | system | branch | | | | Fast System Call |
| SYSENTER | SS | RSP | I... | ... | | | 0F | 34 | | Sr | | | system | branch | | ..i..... | ..i..... | Fast System Call |
| SYSEXIT | SS | eSP | I... | ... | | | 0F | 35 | | Sr | | | system | branch | trans | | | Fast Return from Fast System Call |
| SYSRET | SS | Fd | R11 | ... | | | 0F | 07 | | | | | system | branch | trans | | | Return From Fast System Call |
| TEST | Eb | Gb | | | | | | 84 | | dw | r | | gen | arith | binary | o..szapc | o..sz.pc | Logical Compare |
| TEST | Evqp | Gvqp | | | | | | 85 | | dW | r | | gen | arith | binary | o..szapc | o..sz.pc | Logical Compare |
| TEST | AL | Ib | | | | | | A8 | | w | | | gen | logical | | o..szapc | o..sz.pc | Logical Compare |
| TEST | rAX | Ivds | | | | | | A9 | | W | | | gen | logical | | o..szapc | o..sz.pc | Logical Compare |
| TEST | Eb | Ib | | | | | | F6 | | w | 0 | | gen | logical | | o..szapc | o..sz.pc | Logical Compare |
| TEST alias | Eb | Ib | | | | | | F6 | | w | 1 | | gen | logical | | o..szapc | o..sz.pc | Logical Compare |
| TEST | Evqp | Ivqp | | | | | | F7 | | W | 0 | | gen | logical | | o..szapc | o..sz.pc | Logical Compare |
| TEST alias | Evqp | Ivqp | | | | | | F7 | | W | 1 | | gen | logical | | o..szapc | o..sz.pc | Logical Compare |
| UCOMISD | Vsd | Wsd | | | sse2 | 66 | 0F | 2E | | | r | | pcksclr | compar | | ....z.pc | ....z.pc | Unordered Compare Scalar Double-FP Values and Set EFLAGS |
| UCOMISS | Vss | Wss | | | sse1 | | 0F | 2E | | | r | | simdfp | compar | | ....z.pc | ....z.pc | Unordered Compare Scalar Single-FP Values and Set EFLAGS |
| UD | G | E | | | | | 0F | B9 | | | r | | gen | control | | | | Undefined Instruction |
| UD2 | | | | | | | 0F | 0B | | | | | gen | control | | | | Undefined Instruction |
| UNPCKHPD | Vpd | Wpd | | | sse2 | 66 | 0F | 15 | | | r | | pcksclr | shunpck | | | | Unpack and Interleave High Packed Double-FP Values |
| UNPCKHPS | Vps | Wq | | | sse1 | | 0F | 15 | | | r | | simdfp | shunpck | | | | Unpack and Interleave High |

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | | | | | | | | | Packed Single-FP Values |
| UNPCKLPD | Vpd | Wpd | | | sse2 | 66 | 0F | 14 | | | r | | pcksclr | shunpck | | | | Unpack and Interleave Low Packed Double-FP Values |
| UNPCKLPS | Vps | Wq | | | sse1 | | 0F | 14 | | | r | | simdfp | shunpck | | | | Unpack and Interleave Low Packed Single-FP Values |
| VERR | Ew | | | | | | 0F | 00 | | | 4 | | system | | | ....z... | ....z... | Verify a Segment for Reading |
| VERW | Ew | | | | | | 0F | 00 | | | 5 | | system | | | ....z... | ....z... | Verify a Segment for Writing |
| VMCALL | | | | | vmx | | 0F | 01 | C1 | | 0 | | | | | o..szapc | o..szapc | Call to VM Monitor |
| VMCLEAR | Mq | | | | vmx | 66 | 0F | C7 | | | 6 | | | | | o..szapc | o..szapc | Clear Virtual-Machine Control Structure |
| VMLAUNCH | | | | | vmx | | 0F | 01 | C2 | | 0 | | | | | o..szapc | o..szapc | Launch Virtual Machine |
| VMPTRLD | Mq | | | | vmx | | 0F | C7 | | | 6 | | | | | o..szapc | o..szapc | Load Pointer to Virtual-Machine Control Structure |
| VMPTRST | Mq | | | | vmx | | 0F | C7 | | | 7 | | | | | o..szapc | o..szapc | Store Pointer to Virtual-Machine Control Structure |
| VMREAD | Eq | Gq | | | vmx | | 0F | 78 | | | r | | | | | o..szapc | o..szapc | Read Field from Virtual-Machine Control Structure |
| VMRESUME | | | | | vmx | | 0F | 01 | C3 | | 0 | | | | | o..szapc | o..szapc | Resume Virtual Machine |
| VMWRITE | Gq | Eq | | | vmx | | 0F | 79 | | | r | | | | | o..szapc | o..szapc | Write Field to Virtual-Machine Control Structure |
| VMXOFF | | | | | vmx | | 0F | 01 | C4 | | 0 | | | | | o..szapc | o..szapc | Leave VMX Operation |
| VMXON | Mq | | | | vmx | F3 | 0F | C7 | | | 6 | | | | | o..szapc | o..szapc | Enter VMX Operation |
| WBINVD | | | | | | | 0F | 09 | | | | | system | | | | | Write Back and Invalidate Cache |
| WRMSR | MSR | rCX | rAX | rDX | | | 0F | 30 | | | | | system | | | | | Write to Model Specific Register |
| XADD | Eb | Gb | | | | | 0F | C0 | | dw | r | L | gen | datamov arith | binary | o..szapc | o..szapc | Exchange and Add |
| XADD | Evqp | Gvqp | | | | | 0F | C1 | | dW | r | L | gen | datamov arith | binary | o..szapc | o..szapc | Exchange and Add |
| XCHG | Gb | Eb | | | | | | 86 | | Dw | r | L | gen | datamov | | | | Exchange Register/Memory with Register |
| XCHG | Gvqp | Evqp | | | | | | 87 | | DW | r | L | gen | datamov | | | | Exchange Register/Memory with Register |
| XCHG | Zvqp | rAX | | | | | | 90 | | +r | | | gen | datamov | | | | Exchange Register/Memory with Register |
| XGETBV | EDX | EAX | ECX | XCR | | | 0F | 01 | D0 | | 2 | | system | | | | | Get Value of Extended Control Register |
| XLAT | AL | BBb | | | | | | D7 | | | | | gen | datamov | | | | Table Look-up Translation |
| XLATB | AL | BBb | | | | | | | | | | | | | | | | |
| XOR | Eb | Gb | | | | | | 30 | | dw | r | L | gen | logical | | o..szapc | o..sz.pc | Logical Exclusive OR |
| XOR | Evqp | Gvqp | | | | | | 31 | | dW | r | L | gen | logical | | o..szapc | o..sz.pc | Logical Exclusive OR |

| | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| XOR | Gb | Eb | | | | | | 32 | | D w | r | | gen | logical | | o..szapc | o..sz.pc | Logical Exclusive OR |
| XOR | Gvqp | Evqp | | | | | | 33 | | D W | r | | gen | logical | | o..szapc | o..sz.pc | Logical Exclusive OR |
| XOR | AL | Ib | | | | | | 34 | | w | | | gen | logical | | o..szapc | o..sz.pc | Logical Exclusive OR |
| XOR | rAX | Ivds | | | | | | 35 | | W | | | gen | logical | | o..szapc | o..sz.pc | Logical Exclusive OR |
| XOR | Eb | Ib | | | | | | 80 | | w | 6 | L | gen | logical | | o..szapc | o..sz.pc | Logical Exclusive OR |
| XOR | Evqp | Ivds | | | | | | 81 | | W | 6 | L | gen | logical | | o..szapc | o..sz.pc | Logical Exclusive OR |
| XOR | Evqp | Ibs | | | | | | 83 | | S W | 6 | L | gen | logical | | o..szapc | o..sz.pc | Logical Exclusive OR |
| XORPD | Vpd | Wpd | | | sse2 | 66 | 0F | 57 | | | r | | pcksclr | logical | | | | Bitwise Logical XOR for Double-FP Values |
| XORPS | Vps | Wps | | | sse1 | | 0F | 57 | | | r | | simdfp | logical | | | | Bitwise Logical XOR for Single-FP Values |
| XRSTOR | ST | ST1 | ST2 | ... | | | 0F | AE | | | 5 | | system | | | | | Restore Processor Extended States |
| XSAVE | M | EDX | EAX | ... | | | 0F | AE | | | 4 | | system | | | | | Save Processor Extended States |
| XSAVE | M | EDX | EAX | ... | | | 0F | AE | | | 4 | | system | | | | | Save Processor Extended States |
| XSETBV | XCR | ECX | EDX | EA X | | | 0F | 01 | D1 | | 2 | | system | | | | | Set Extended Control Register |