

TEST—Logical Compare

Opcode	Instruction	64-Bit Mode	Compat/ Leg Mode	Description
A8 <i>ib</i>	TEST AL, <i>imm8</i>	Valid	Valid	AND <i>imm8</i> with AL; set SF, ZF, PF according to result.
A9 <i>iw</i>	TEST AX, <i>imm16</i>	Valid	Valid	AND <i>imm16</i> with AX; set SF, ZF, PF according to result.
A9 <i>id</i>	TEST EAX, <i>imm32</i>	Valid	Valid	AND <i>imm32</i> with EAX; set SF, ZF, PF according to result.
REX.W + A9 <i>id</i>	TEST RAX, <i>imm32</i>	Valid	N.E.	AND <i>imm32</i> sign-extended to 64-bits with RAX; set SF, ZF, PF according to result.
F6 /0 <i>ib</i>	TEST <i>r/m8</i> , <i>imm8</i>	Valid	Valid	AND <i>imm8</i> with <i>r/m8</i> ; set SF, ZF, PF according to result.
REX + F6 /0 <i>ib</i>	TEST <i>r/m8*</i> , <i>imm8</i>	Valid	N.E.	AND <i>imm8</i> with <i>r/m8</i> ; set SF, ZF, PF according to result.
F7 /0 <i>iw</i>	TEST <i>r/m16</i> , <i>imm16</i>	Valid	Valid	AND <i>imm16</i> with <i>r/m16</i> ; set SF, ZF, PF according to result.
F7 /0 <i>id</i>	TEST <i>r/m32</i> , <i>imm32</i>	Valid	Valid	AND <i>imm32</i> with <i>r/m32</i> ; set SF, ZF, PF according to result.
REX.W + F7 /0 <i>id</i>	TEST <i>r/m64</i> , <i>imm32</i>	Valid	N.E.	AND <i>imm32</i> sign-extended to 64-bits with <i>r/m64</i> ; set SF, ZF, PF according to result.
84 / <i>r</i>	TEST <i>r/m8</i> , <i>r8</i>	Valid	Valid	AND <i>r8</i> with <i>r/m8</i> ; set SF, ZF, PF according to result.
REX + 84 / <i>r</i>	TEST <i>r/m8*</i> , <i>r8*</i>	Valid	N.E.	AND <i>r8</i> with <i>r/m8</i> ; set SF, ZF, PF according to result.
85 / <i>r</i>	TEST <i>r/m16</i> , <i>r16</i>	Valid	Valid	AND <i>r16</i> with <i>r/m16</i> ; set SF, ZF, PF according to result.
85 / <i>r</i>	TEST <i>r/m32</i> , <i>r32</i>	Valid	Valid	AND <i>r32</i> with <i>r/m32</i> ; set SF, ZF, PF according to result.
REX.W + 85 / <i>r</i>	TEST <i>r/m64</i> , <i>r64</i>	Valid	N.E.	AND <i>r64</i> with <i>r/m64</i> ; set SF, ZF, PF according to result.

NOTES:

- * In 64-bit mode, *r/m8* can not be encoded to access the following byte registers if a REX prefix is used: AH, BH, CH, DH.

Description

Computes the bit-wise logical AND of first operand (source 1 operand) and the second operand (source 2 operand) and sets the SF, ZF, and PF status flags according to the result. The result is then discarded.

In 64-bit mode, using a REX prefix in the form of REX.R permits access to additional registers (R8-R15). Using a REX prefix in the form of REX.W promotes operation to 64 bits. See the summary chart at the beginning of this section for encoding data and limits.

Operation

```
TEMP ← SRC1 AND SRC2;
SF ← MSB(TEMP);
```

```
IF TEMP = 0
    THEN ZF ← 1;
    ELSE ZF ← 0;
```

FI:

```
PF ← BitwiseXNOR(TEMP[0:7]);
CF ← 0;
OF ← 0;
(* AF is undefined *)
```

Flags Affected

The OF and CF flags are set to 0. The SF, ZF, and PF flags are set according to the result (see the “Operation” section above). The state of the AF flag is undefined.

Protected Mode Exceptions

#GP(0)	If a memory operand effective address is outside the CS, DS, ES, FS, or GS segment limit. If the DS, ES, FS, or GS register contains a NULL segment selector.
#SS(0)	If a memory operand effective address is outside the SS segment limit.
#PF(fault-code)	If a page fault occurs.
#AC(0)	If alignment checking is enabled and an unaligned memory reference is made while the current privilege level is 3.
#UD	If the LOCK prefix is used.

Real-Address Mode Exceptions

#GP	If a memory operand effective address is outside the CS, DS, ES, FS, or GS segment limit.
#SS	If a memory operand effective address is outside the SS segment limit.
#UD	If the LOCK prefix is used.

Virtual-8086 Mode Exceptions

#GP(0)	If a memory operand effective address is outside the CS, DS, ES, FS, or GS segment limit.
#SS(0)	If a memory operand effective address is outside the SS segment limit.
#PF(fault-code)	If a page fault occurs.
#AC(0)	If alignment checking is enabled and an unaligned memory reference is made.
#UD	If the LOCK prefix is used.

Compatibility Mode Exceptions

Same exceptions as in protected mode.

64-Bit Mode Exceptions

#SS(0)	If a memory address referencing the SS segment is in a non-canonical form.
#GP(0)	If the memory address is in a non-canonical form.
#PF(fault-code)	If a page fault occurs.
#AC(0)	If alignment checking is enabled and an unaligned memory reference is made while the current privilege level is 3.
#UD	If the LOCK prefix is used.