

## TEST—Logical Compare

Opcode	Instruction	Op/En	64-Bit Mode	Compat/Leg Mode	Description
A8 ib	TEST AL, imm8	I	Valid	Valid	AND imm8 with AL; set SF, ZF, PF according to result.
A9 iw	TEST AX, imm16	I	Valid	Valid	AND imm16 with AX; set SF, ZF, PF according to result.
A9 id	TEST EAX, imm32	I	Valid	Valid	AND imm32 with EAX; set SF, ZF, PF according to result.
REX.W + A9 id	TEST RAX, imm32	I	Valid	N.E.	AND imm32 sign-extended to 64-bits with RAX; set SF, ZF, PF according to result.
F6 /0 ib	TEST r/m8, imm8	MI	Valid	Valid	AND imm8 with r/m8; set SF, ZF, PF according to result.
REX + F6 /0 ib	TEST r/m8 <sup>1</sup> , imm8	MI	Valid	N.E.	AND imm8 with r/m8; set SF, ZF, PF according to result.
F7 /0 iw	TEST r/m16, imm16	MI	Valid	Valid	AND imm16 with r/m16; set SF, ZF, PF according to result.
F7 /0 id	TEST r/m32, imm32	MI	Valid	Valid	AND imm32 with r/m32; set SF, ZF, PF according to result.
REX.W + F7 /0 id	TEST r/m64, imm32	MI	Valid	N.E.	AND imm32 sign-extended to 64-bits with r/m64; set SF, ZF, PF according to result.
84 /r	TEST r/m8, r8	MR	Valid	Valid	AND r8 with r/m8; set SF, ZF, PF according to result.
REX + 84 /r	TEST r/m8 <sup>1</sup> , r8 <sup>1</sup>	MR	Valid	N.E.	AND r8 with r/m8; set SF, ZF, PF according to result.
85 /r	TEST r/m16, r16	MR	Valid	Valid	AND r16 with r/m16; set SF, ZF, PF according to result.
85 /r	TEST r/m32, r32	MR	Valid	Valid	AND r32 with r/m32; set SF, ZF, PF according to result.
REX.W + 85 /r	TEST r/m64, r64	MR	Valid	N.E.	AND r64 with r/m64; set SF, ZF, PF according to result.

### NOTES:

1. 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.

### Instruction Operand Encoding

Op/En	Operand 1	Operand 2	Operand 3	Operand 4
I	AL/AX/EAX/RAX	imm8/16/32	N/A	N/A
MI	ModRM:r/m (r)	imm8/16/32	N/A	N/A
MR	ModRM:r/m (r)	ModRM:reg (r)	N/A	N/A

### 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.