The Or instruction performs a logical bit by bit "or" between a register and a fullword in memory. Operand 1, the target, is a register and Operand 2, the source, specifies a fullword in memory. The fullword in memory is or-ed internally with the fullword in the register and the result is placed in the register. The fullword in memory is not changed. The table below shows the results of "or-ing" two bits together.

<table>
<thead>
<tr>
<th>Bit 1</th>
<th>Bit 2</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

This instruction sets the condition code as follows:

0  if all target bits are set to 0. Test this condition with **BZ** or **BNZ**.
1  if any target bit is set to 1. Test this condition with **BM** or **BNM**.

Some Unrelated Ors

```
R4 = X’FFFFFFFF’  ALL 1’S
R5 = X’00000000’  ALL 0’S
R6 = X’0000148C’  00000000000000000001010010001100
FIELD1 DC X’0000148C’
FIELD2 DC X’00000000’
FIELD3 DC X’FFFFFFFF’
FIELD4 DC X’12345678’
```

```
O  R4,FIELD1  R4 = X’FFFFFFFF’  Condition Code = 1
O  R4,FIELD2  R4 = X’FFFFFFFF’  Condition Code = 1
O  R4,FIELD3  R4 = X’FFFFFFFF’  Condition Code = 1
O  R4,FIELD4  R4 = X’FFFFFFFF’  Condition Code = 1
O  R5,FIELD1  R5 = X’0000148C’  Condition Code = 1
O  R5,FIELD2  R5 = X’00000000’  Condition Code = 0
O  R5,FIELD3  R5 = X’FFFFFFFF’  Condition Code = 1
O  R5,FIELD4  R5 = X’12345678’  Condition Code = 1
O  R6,FIELD1  R6 = X’0000148C’  Condition Code = 1
O  R6,FIELD2  R6 = X’0000148C’  Condition Code = 1
```
O R6,FIELD3  R6 = X'FFFFFFFF'  Condition Code = 1
O R6,FIELD4  R6 = X'123456FC'  Condition Code = 1