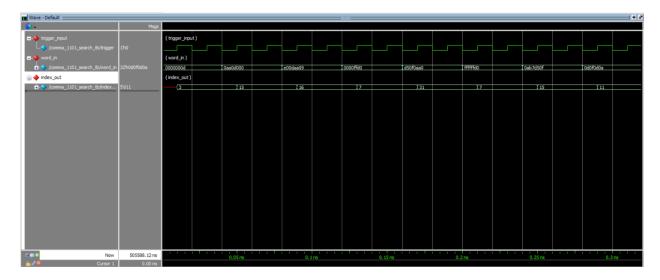
Bit wave



Output Table

```
0 word in = 0000000000000000000000000001101: index out = x
10 word in = 000000000000000000000000001101: index out = 3
40 word in = 00001010101000000110100000000000: index out = 3
50 word in = 00001010101000000110100000000000: index out = 15
80 word in = 11100000000011011010101000001001: index out = 15
90 word in = 11100000000011011010101000001001: index out = 16
120 word in = 00000000000000001111111111111010000: index out = 16
130 word_in = 0000000000000000111111111111010000: index_out = 7
160 word_in = 11010101000011110000101010100000: index_out = 7
170 word_in = 11010101000011110000101010100000: index_out = 31
200 word in = 1111111111111111111111111111111010000: index out = 31
210 word in = 11111111111111111111111111111010000: index out = 7
240 word in = 00001010101101111101010100001111: index out = 7
250 word in = 00001010101101111101010100001111: index out = 15
280 word in = 00001101000011110000110100001010: index out = 15
290 word in = 00001101000011110000110100001010: index out = 11
```

Summary

- The design takes in a 32 bits input as word_in
- Checks for a 4-bit set of 1101 and marks the index of the MSB of the set
- If the set is not found in the first 4 bits the index shifts one bit to the right and looks at the next 4 bits
- Once the set is found the index of the MSB is reported as index_out
- If all 32 bits are checked OR a set of 1101 is not found index out becomes zero