

```

1  public class BitReversedCounter {
2      int counter, reverse, highBit;
3      BitReversedCounter(int initialValue) {
4          counter = initialValue;
5          reverse = 0;
6          highBit = -1;
7      }
8      public int reverseIncrement() {
9          if (counter++ == 0) {
10             reverse = highBit = 1;
11             return reverse;
12         }
13         int bit = highBit >> 1;
14         while (bit != 0) {
15             reverse ^= bit;
16             if ((reverse & bit) != 0) break;
17             bit >>= 1;
18         }
19         if (bit == 0)
20             reverse = highBit <= 1;
21         return reverse;
22     }
23     public int reverseDecrement() {
24         counter--;
25         int bit = highBit >> 1;
26         while (bit != 0) {
27             reverse ^= bit;
28             if ((reverse & bit) == 0) {
29                 break;
30             }
31             bit >>= 1;
32         }
33         if (bit == 0) {
34             reverse = counter;
35             highBit >>= 1;
36         }
37         return reverse;
38     }
39 }

```

FIGURE 15.16 A bit-reversed counter.