The problem tries to change the amortized cost from 1 per bit to 2^d for the d-th bit. I've decided to use the aggregated method.

We can consider how often we flip each individual bit, and sum those up to bound the total, rather than individually obtaining a worst case bound for each bit.From the analysis in class, we know that d-th bit toggled every 2^d time, e.g. the second bit toggle every twice time.

If the counter now counts up to n, we can write down how frequent the bits are toggled,

$$\begin{aligned} & n + \lfloor \frac{n}{2} \rfloor + \lfloor \frac{n}{4} \rfloor + \cdots \\ \leq & n + \frac{n}{2} + \frac{n}{4} + \cdots \\ \leq & 2n \end{aligned} \tag{1}$$

so the time to count to n is $2n$.

But the amortized cost is changed from 1 to 2^d to flip a bit, where d is the position of the bit in the counter. For every increment, it needs two flip to perform a toggle, therefore, the price is 2^{d+1}.

We can try to sum up the costs,

$$\begin{aligned} & 2^{1+1} + 2^{1+2} + 2^{1+3} + \cdots + 2^{1+\lceil logn \rceil} \\ \leq & 2^{1+1} + 2^{1+2} + 2^{1+3} + \cdots + 2^{1+logn+1} \\ = & \end{aligned} \tag{2}$$

Divide the cost with the total time and we can know the answer.