Q1. In binary / decimal system, the radix itself is even. Thus, a number of the form

$$A_n r^n + A_{n-1} r^{n-1} + \dots A_0 r^0$$

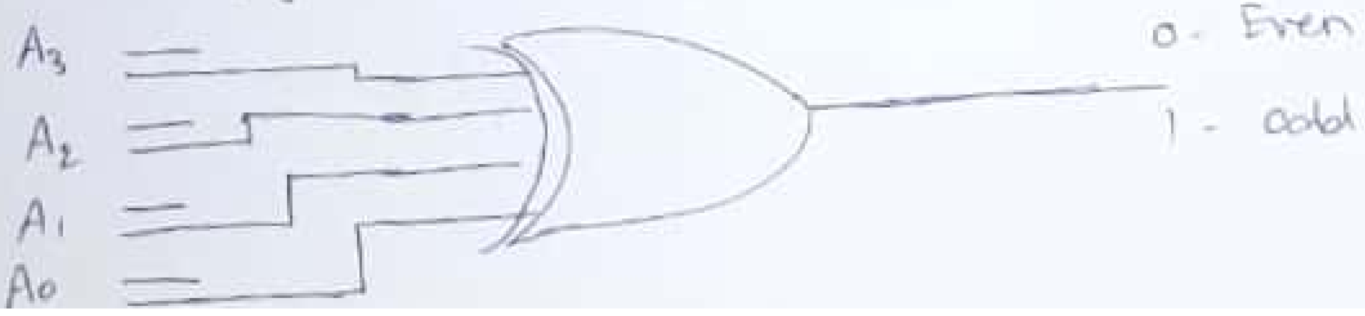
will be even if A_0 is even because the other terms in the numbers are always even because of the presence of r. For $r = 3$, the ~~number~~ radix powers are always odd. Thus, the number can be even if we have a factor of $(r-1)$. Thus,

$$A_n r^n + A_{n-1} r^{n-1} + \dots \quad A_0$$

$$\Rightarrow A_n(r^n - 1) + A_{n-1}(r^{n-1} - 1) + \dots A_1(r-1)$$
$$+ [A_n + A_{n-1} + A_{n-2} \dots + A_1 + A_0]$$

~~The~~ Everything apart from the term in the square bracket is even. Hence, a quick method to determine even/odd number with radix = 3 is to sum all the digits. If the answer is even, the original number is even.

For a 4-digit number, the sum boils down to the sum of the last bit of all the numbers (A_3, A_2, A_1 & A_0). Thus, a simple XOR of the last bits is sufficient to figure out if the 4-digit number is even/odd.



Q2.

A	B	C	D	O/P	
0	0	0	0	0	
0	0	0	1	1	– (iii)
0	0	1	0	0	
0	0	1	1	1	– (iii)
0	1	0	0	1	– (ii)
0	1	0	1	0	
0	1	1	0	0	
0	1	1	1	1	– (iii)
1	0	0	0	0	
1	0	0	1	1	– (iv)
1	0	1	0	0	
1	0	1	1	1	– (iii)
1	1	0	0	1	(i)
1	1	0	1	1	
1	1	1	0	1	
1	1	1	1	1	

$O/P = B'D + CD + AB + BC'D'$

Q3 We need to maintain the count of all Rs 1 & Rs 2 coins inserted in the machine. Because they are added to give a final tally, the same counter can be used to count. For count upto 30, a 5-bit counter is used. The LSB is incremented using the input y & 2nd LSB is incremented using input x. We can design a ripple counter for this