

FUNKIT SUBMISSION.

Components used:

1. 4 bit comparator (74HC85) - 1
2. 8 bit ROM (2732) - 2
3. 4 bit counter (74LS161) - 2
4. clock (10000Hz)

LOGIC:-

For the pastry to fall on floor with minimum number of cuts, we need to cut all the ropes having length less than the peg they are attached to. Thus, we have to count the number of cases in which length of rope is less than height of peg.

EXPLANATION OF THE SCHEMATIC:

In schematic the length of rope and height of peg is stored in 2 separate ROMs. As we need to go to different addresses of a ROM a counter is there which counts from 0 to n ($n < 15$ and represents number of pegs), as soon as the counter reaches value $n+1$ the counter stops counting because of the or gate connected to the ENP input of 74LS161 (as at $n+1$ address the ROM produces value 0000). Now, as counter counts, different addresses of ROMs are called and produce the values stored in them as an output. These outputs are then compared using 74HC85 comparator. If the output from ROM(Rope Length) is smaller than the output from ROM(Peg Height), it enables the counter and counter adds one to its previous value. In this way all the cases (length of rope < height of peg) are counted and we get the minimum no of cuts required.

PYTHON CODE FOR ROM:

```
1 array = [15,14,13,12,11,10,9,8,7,6,5,4,3,2,1]
2 # Convert the list of numbers to bytes
3 binary_format = bytes(array)
4 # Open the file in 'w+b' mode
5 with open('yay_binary_file_made2.bin', 'w+b') as f:
6     f.write(binary_format)
```

TEAM MEMBERS (Team Slytherin):-

Abhay Partap Singh Gill

Vivek Premi

Punit Makwana