__APPROACH FOR ENCRYPTER CIRCUIT(PART1)__

1) COMBINATIONAL CIRCUIT

->The combinational circuit is simple .We have two inputs to the circuit and one output.
->As name indicates output changes whenever input changes .
->Using the given information we can design a f (input is round key and right half) function as combinational and can perform xor operation using xor gates .S boxes are also combinational circuit designed using truthtable as given.
->As the circuit involves 4 similar loops so just design 4 loops.The output of i loop is input to (i+1) loop .
->Shift in round key is generated by just interchanging wires.
->OUR CIRCUIT PROCESS:
We have taken 8 bit plain text and initial key as inputs .Permutation of plain text is generated by interchanging wires and then right half and left half is separated .For loop 1 ,round key-1 and right half of permuted input are inputs to f function .Output of f function is a four bit number and it is xored with left half of permuted input .This output serves as right half of next loop and right half of previous loop serves as left half of next loop and similar process runs for 4 loops.After fourth loop ,the right half and left half of the fourth loop is merged and output is generated by using inverse permutation table which is designed by interchanging wires.This output is encrypted text.
->This is the process for combinational design of encrypter.

2) SEQUENTIAL CIRCUIT

->We have used two digital circuits .One is controller and one is datapath.
CONTROLLER- Controller is basically a circuit which controls the operations to be performed like loading the inputs and outputs and shifting the round key ...etc.This loading inputs and outputs operation involves in every sequential circuit.We basically design controller by drawing its state diagram .State diagram depends on controller outputs which may be mealy or moore outputs ,which acts as signals to the datapath.Drawing the required statediagram we designed controller.

DATAPATH-Datapath is circuit where operations are performed on data .Whenever controller output named shiftregs is high the data stored in registers is circular shifted .Datapath involves both storage elements and also logic deciding function.For example: Here in our circuits , xoring operation and shifting operation involves in datapath.We integrate both datapath and controller to design our main circuits.

OUR CIRCUIT PROCESS:

-> It involves controller which i designed by drawing state diagram .As coordinators of DIGISIM said to do sequentially ,we have almost tried to implement every process by using clock .The outputs of our controller are basically signals ,such as loadregs,shiftregs,inc_cnt,loadnext.Whenever loadregs is high all registers involved in datapath are loaded with data and whenever shiftregs is high ,loaded data in register is shifted .We use shiftregs output to generate roundkeys.Whenever inc_cnt is high counter is incremented ,here counter represents loop number .As the circuit involves 4 loops so we use loadnext to load next input which is present output.At the end of the fourth loop the output is generated by inverse permutation.This is encrypted text.

-> We can also perform round key generation by using mux but used register to implement sequentially by shifting.Round key i is generated for i(th) loop, here round key generation involves circular shifting of bits but number of bits shifted depends on rounds.So output of controller,shiftregs depends on counter whether count is even or odd.

-> This is our sequential encrypter circuit.

_____APPROACH FOR (DECRYPTER AND ENCRYPTER)PART2____

1) COMBINATIONAL CIRCUIT

-> Here in our circuit we have plain text/cipher text ,initial key and mode as input to the circuit .

-> The output is cipher text /plain text.

-> The output changes whenever input is changed .

-> In this circuit the input depends on the mode ,whether circuit is acting as decrypter or encrypter circuit.

-> Shifting bits to produce round key is done by interchanging wires.

-> PROCESS:

We have taken 8 bit input and initial key as inputs .Permutation of

input is generated by interchanging wires and then based on mode right half and left half is separated .For loop 1 , round key-1 / round key-4 (based on mode) and right half /left half (based on mode) of permuted input are inputs to f function .Output of f function is a four bit number and it is xored with left half/right half (based on mode) of permuted input .This output serves as inputs based on mode of next loop similar process runs for 4 loops.After fourth loop ,the right half and left half of the fourth loop is merged and output is generated by using inverse permutation table which is designed by interchanging wires.This output is encrypted text.
-> The round key for loop i depends on mode so we used mux to generate required key to loop i.
-> The actual cost of quad 2:1 mux given is 2 but if we desing it using gates it costs only 1.6 but for the circuit to work efficiently we have used direct mux IC (74LS157).
-> This is our circuit for part 2.

2)SEQUENTIAL CIRCUIT

-> We have designed a controller and datapath for sequential circuit.
-> We then integrate both controller and datapath to specify overall circuit.
-> In this part2 we have designed two controllers.The first controller decide the operation to be performed in datapath and the second controller decide the shift to be performed which depends on input bit mode.We have used register IC(74194),it has two selection bits.This two selection bits indicate 4 kinds of operation to be performed and this selection bits decides the different kinds of shift,load and even hold operation on round keys.
-> This circuit is very similar to part1 sequential with extra input mode bit.We used muxes,whose output depends on mode bit .
-> We have used mux at the round key generation and input loading into register.This mode bit decides the way of shifting (circular left or right shift).All other circuit is similar to encrypter sequential circuit.
->The shift_controller circuit that is being used is a combinational circuit whose inputs are mode and some of the outputs of the controller circuit.
-> Based on the given information about part2 we have to use mux even before output is inverse permuted.
-> The datapath circuit is same as part1 sequential with small changes like adding mux.