Finite State Machine Design #0

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **000000**

Input for testing: **10010000000110000001**

Finite State Machine Design #2

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **000010**

Input for testing: **10000010000101100000101**

Finite State Machine Design #4

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **000100**

Input for testing: **10000100010011000 1001**

Finite State Machine Design #6

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **000110**

Input for testing: **10001000110001101100011101**

Finite State Machine Design #8

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **001000**

Input for testing: **10010010001000010010001**

Finite State Machine Design #9

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **001001**

Input for testing: **100100010010011100010010**

Finite State Machine Design #10

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **001010**

Input for testing: **100100101001010110010101**

Finite State Machine Design #12

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **001100**

Input for testing: **10010011001100010011001**

Finite State Machine Design #14

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **001110**

Input for testing: **10011001110011101100111101**

Finite State Machine Design #16

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **010000**

Input for testing: **10100010000100000010100001**

Finite State Machine Design #17

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **010001**

Input for testing: **10100001000100011101000010**

Finite State Machine Design #18

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **010010**

Input for testing: **10100010010010110100101**

Finite State Machine Design #20

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **010100**

Input for testing: **10101010010100010101001**

Finite State Machine Design #21

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **010101**

Input for testing: **10101001010101110101010**

Finite State Machine Design #22

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **010110**

Input for testing: **10101011010110010101101**

Finite State Machine Design #24

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **011000**

Input for testing: **10110011000110000010110001**

Finite State Machine Design #25

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **011001**

Input for testing: **10110001100110010101100 10**

Finite State Machine Design #26

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **011010**

Input for testing: **10110110101101001011010 1**

Finite State Machine Design #27

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **011011**

Input for testing: **101101011011011110110110**

Finite State Machine Design #28

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **011100**

Input for testing: **10111011100111000010111001**

Finite State Machine Design #29

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **011101**

Input for testing: **10111001110111010101110 10**

Finite State Machine Design #30

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **011110**

Input for testing: **10111011110111100101111101**

Finite State Machine Design #33

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **100001**

Input for testing: **0100010000100001101000010**

Finite State Machine Design #34

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **100010**

Input for testing: **0100011000100010101000101**

Finite State Machine Design #35

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **100011**

Input for testing: **0100010001100011101000110**

Finite State Machine Design #36

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **100100**

Input for testing: **010010100100100001001001**

Finite State Machine Design #37

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **100101**

Input for testing: **0100100101001011010010 10**

Finite State Machine Design #38

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **100110**

Input for testing: **0100111001100110101001101**

Finite State Machine Design #39

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **100111**

Input for testing: **0100110011100111101001110**

Finite State Machine Design #41

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **101001**

Input for testing: **010101001010011010010010**

Finite State Machine Design #42

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **101010**

Input for testing: **010101101010100010101 01**

Finite State Machine Design #43

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **101011**

Input for testing: **010101011010111010101 10**

Finite State Machine Design #45

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **101101**

Input for testing: **01011101101101001011010**

Finite State Machine Design #46

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **101110**

Input for testing: **01011110111011100010111101**

Finite State Machine Design #47

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **101111**

Input for testing: **0101110111101111101011110**

Finite State Machine Design #49

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **110001**

Input for testing: **011001100011000100110010**

Finite State Machine Design #51

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **110011**

Input for testing: **011011001100111011001l0**

Finite State Machine Design #53

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **110101**

Input for testing: **011011010110101001101010**

Finite State Machine Design #54

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **110110**

Input for testing: **01101110110110001101101**

Finite State Machine Design #55

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **110111**

Input for testing: **01101101110111101101110**

Finite State Machine Design #57

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **111001**

Input for testing: **01110111001110010011110010**

Finite State Machine Design #59

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **111011**

Input for testing: **01111011101100111O110**

Finite State Machine Design #61

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **111101**

Input for testing: **01111101111101001111010**

Finite State Machine Design #63

Develop a finite state machine (FSM) diagram to detect the particular binary pattern shown below. The input to your FSM is a sequence of binary bits that come in series at each clock cycle. When the FSM recognizes the bit pattern, the output is set to '1', otherwise it stays at '0'. Your FSM should be able detect overlapped patterns. Use the following input sequence to test your FSM in the simulation and on the hardware board.

Pattern to be matched: **111111**

Input for testing: **01101111111001111110**