HW #3

CSEE W3827 - Fundamentals of Computer Systems Spring 2022

Prof. Rubenstein
Due 2/18/22, 5pm

Topics: Standard circuitry (encoder, decoder mux), combinatorial circuit design
Note that this homework has 5 problems and is 2 pages long.

Warmup Problems

These problems do not need to be submitted - they are easier and can help prepare/familiarize the student with the material on the harder problems (that do need to be turned in).

1. Build a MUX from a Decoder and some AND and OR gates
2. A MUX-with-Enable is a MUX with one additional selector input, E. When $E = 1$, the MUX-with-Enable behaves like a traditional MUX. When $E = 0$, the MUX-with-Enable is disabled and outputs 0. Build a MUX-with-Enable from a MUX (without enable) and some AND gates.
3. A 1-to-2^k DEMUX takes one data input I and a k-bit selector S as input and outputs 0 on each of $2^k - 1$ outputs, and outputs I on the jth output where j is the unsigned binary value represented by S. Construct a DEMUX from a decoder and a bunch of AND gates.
4. Use five 2-to-1 MUXs and as many NOT gates as needed, buid a circuit that take a 5-bit value and a selector input S and returns
   - when $S = 0$, the original number is returned
   - when $S = 1$, the 1's complement of the number is returned.
5. Build a circuit using two 2-to-1 MUXs that takes a 2 bit-input B_1B_0 and a 1-bit selector S and returns B_1B_0 when $S = 0$ and returns B_0B_1 (i.e., switches the bit-order) when $S = 1$.
6. Solve problem 3 of the "Harder Problems" using four 16-to-1 MUXs, one MUX for each output NSH, NSL, EWH, EWL. Now solve using four 8-to-1 MUXs
7. Design a circuit that receives a k-bit string $A = A_{k-1}A_{k-2}\cdots A_1A_0$ and, using $k - 2$ 4-to-1 MUXs, outputs k-bit string $B = B_{k-1}B_{k-2}\cdots B_1B_0$ with the following properties
   - $B_0 = A_0, B_{k-1} = A_{k-1}$
   - $B_i = A_i$ whenever $A_{i+1} \neq A_{i-1}, 0 < i < k - 1$
   - $B_i = A_{i-1}$ whenever $A_{i+1} = A_{i-1}, 0 < i < k - 1$

Harder Problems

1. Construct a 4-to-16 line decoder with an enable input using five 2-to-4 line decoders with enable inputs (Hint: Start at the outputs: If all that is being used is decoders, then how many decoders are connected directly to outputs?)
2. A combinatorial circuit is specified by the following three Boolean functions:

$$F = X + \bar{Y} + \bar{X}YZ$$

   Design the circuit with a decoder and external OR gates.

3. A traffic light controller receives a 4-bit input that changes every 5 seconds. The input sequence is a simple counter that counts from binary 0 to binary 15 and then starts again from binary 0. These signals go to 4 outputs, NSH, NSL, EWH, EWL. The first two outputs NSH and NSL respectively represent the high and low bits that are fed to the lamps that face in the North-south direction. The other two outputs EWH and EWL respectively represent the high and low bits that are fed to the lamps that face in the East-West direction. The following table indicates how setting the high and low bits determines the color of the lamp:

High	Low	Lamp Color
0	0	Green
0	1	Yellow
1	0	Red
1	1	Red

Each light should be green for 30 seconds, yellow for 5 seconds, and then red for 45 seconds. There should be two 5 second intervals when both lights are red. Assume that for the interval where the input is 0000, the North-South light has just turned green, and the East-West light has been red for 5 seconds already.

Design the circuitry that feeds from $ABCD$ to the two outputs. You may represent your answer as algebraic equations (make sure to simplify).

4. The 01-swap operation, b, on a binary string S, permutes all occurences of 01 within the original string to 10 (the process is not recursive). For instance:

- $b(0) = 0, b(1) = 1, b(00) = 00, b(01) = 10, b(11) = 11$
- $b(000) = 000, b(001) = 010, b(010) = 100, b(100) = 100$
- $b(0\ 01\ 01\ 1\ 01\ 0) = 0\ 10\ 10\ 1\ 10\ 0$ (spacing added for clarity).

Design a circuit using 2-1 multiplexers that can be used to perform the 01-swap on a k-bit string $S = S_{k-1}S_{k-2}\cdots S_0$. where each S_i is a bit.

(a) First, show the circuit, built using the 2:1 MUX, whose output is the ith bit of $b(S)$ where $0 < i < k - 1$. YOU DO NOT NEED ANY AND, OR, OR NOT GATES, only a single 2:1 MUX. This is somewhat challenging - so think what input information you need.

(b) Use contraction to solve the edge cases when $i = 0, k - 1$. You do not have to simplify the internals of the MUX, just explain why you "contracted" as you did.

5. The isolated-1-shift-left operation (i1sl for short) applied to a binary string S moves a 1 bit by one position to the left in the solution if the bit is not adjacent to any other 1's (i.e., 0's on both sides, or if it is the least-significant bit, a 0 immediately above it.

For instance, the following show application of i1sl to various strings:

- $010 \to 100$, $011 \to 011$, $01010 \to 10100$
- $011010 \to 011100$, $011100 \to 011100$

(a) Build a simplified circuit (using only AND, OR, NOT gates) whose output is the ith bit of the i1sl, where $1 < i < k - 1$ (there's a hint here about what inputs are needed). You may leave your answer in algebraic (SoP) form in terms of the S_j. Note that the ith bit can be determined by only looking at a few of the S_j.

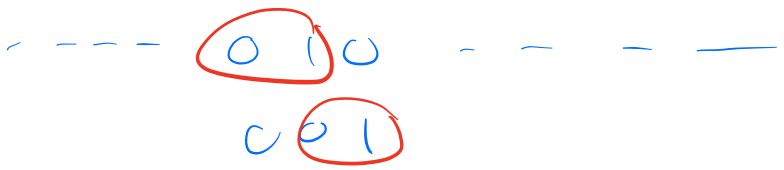
(b) Suppose it is known that the input string will never contain 3 consecutive 1's. Draw the simplified circuit.

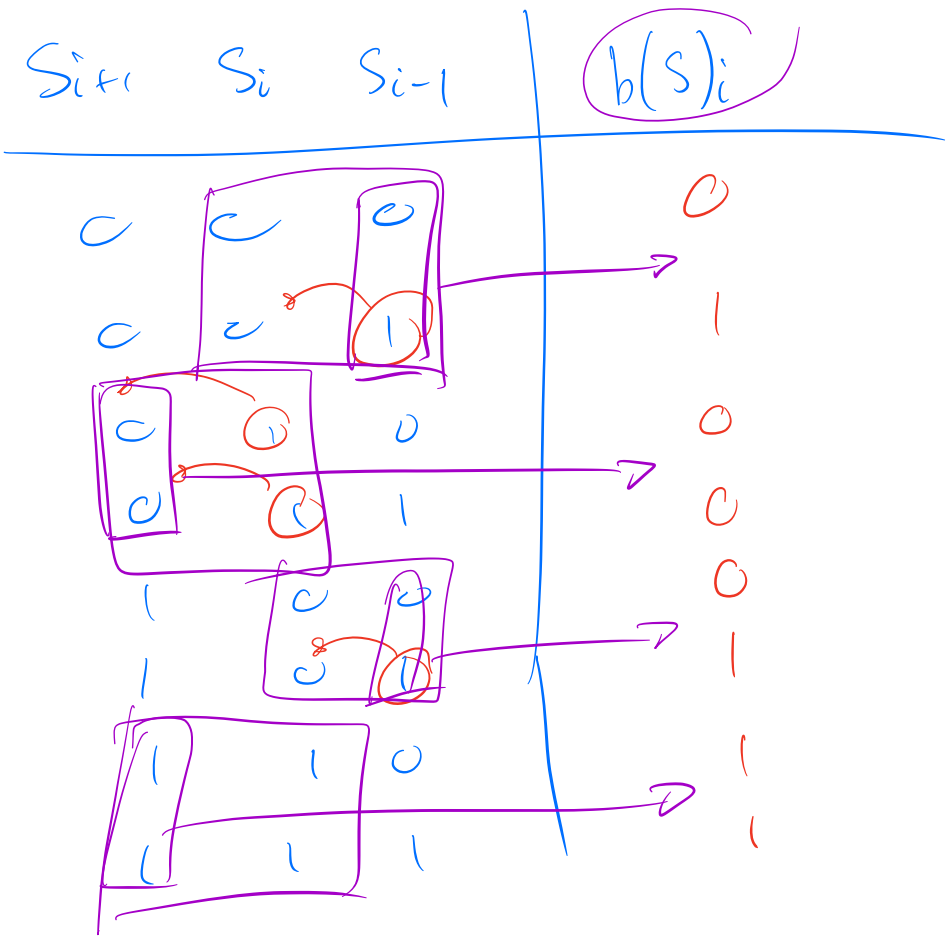
(c) Suppose the input string might contain 3 1's, but that no consecutive 4 bits contain 3 0's (i.e., 0000, 0001, 0010, 0100, or 1000 never appear as a substring), nor does 0101 ever appear (1010 might still appear though, e.g., 111011011010). Draw the simplified circuit.

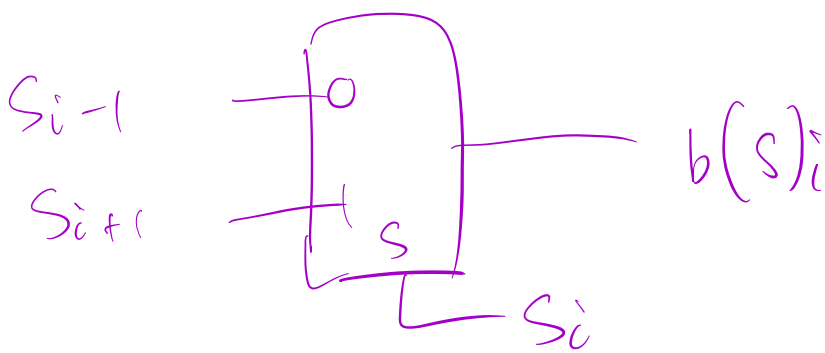
(d) Use contraction to design the circuits for the outputs of the $k - 1$st, 1st and 0th bits.

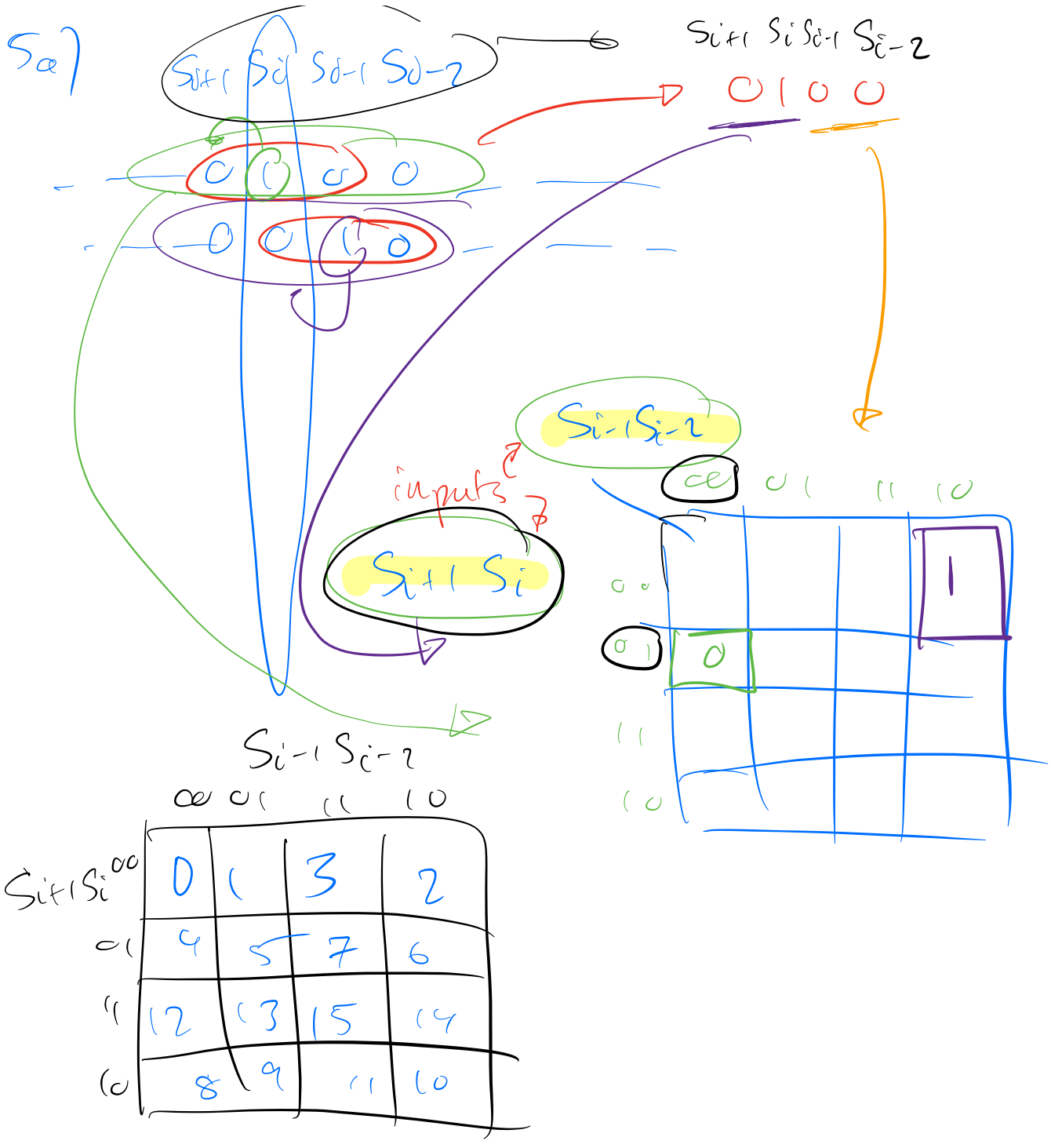
4a) $b(s)_i$ $\quad S = S_{k-1} \cdots S_0$

$S_{i+1} S_i S_{i-1}$








Sa)
Si+1 Si Si-1 Si-2
Si+1 Si Si-1 Si-2
0 1 0 0
Si-1 Si-2
inputs
Si+1 Si
1
0
Si-1 Si-2
Si+1 Si
0
1
3
2
4
5
7
6
12
13
15
14
8
9
11
10