

Solutions to Problem Set #5

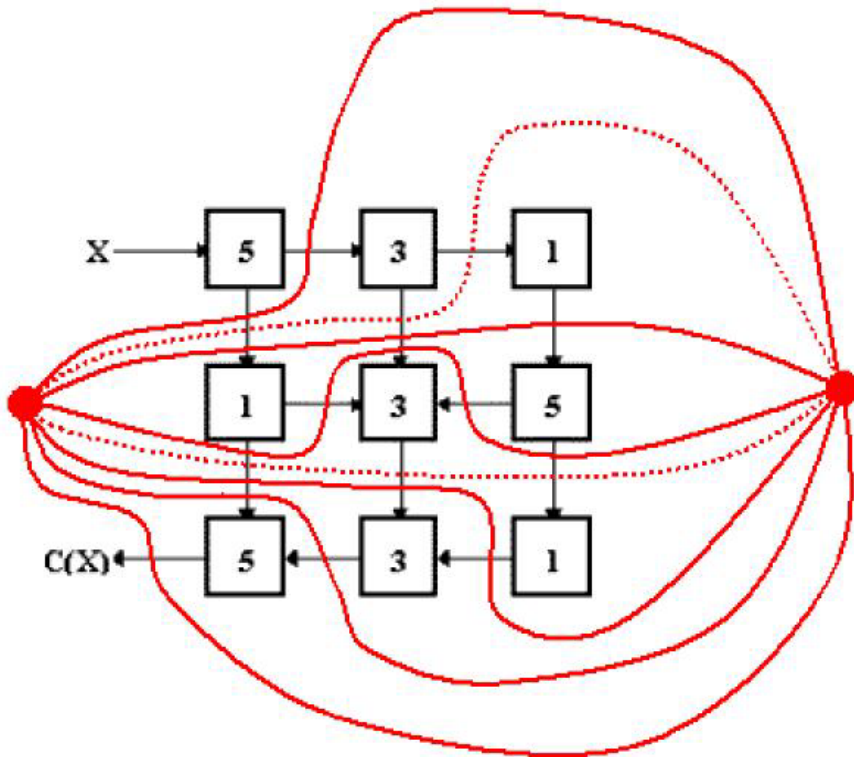
Problem 1. "Peculiar Pipelines" [30 points]

(A) What is the latency and throughput of the combinational encryption device?
3/4/2002 Page 7 of 15

$$\text{Latency} = 5 + 3 + 1 + 5 + 3 + 3 + 5 = 25\text{us.}$$

$$\text{Throughput} = 1/25\text{us.}$$

(B) We can remove some of the registers implied by contours (eg, those shown with dotted lines) without decreasing the throughput. There are several equivalent variations of this diagram.



(C) Again assume ideal registers. There are six registers in each input-output path and the clock period is 5, so latency = 30 and throughput = 1/5.

Problem 2. Turing Machine [40 points]

(A) There are $n+1$ states (including S_0), which will take $\lceil \log_2(n+1) \rceil$ bits to encode. Adding the "Tape Value" input, there are $\lceil \log_2(n+1) \rceil + 1$ inputs total, thus $2^{\lceil \log_2(n+1) \rceil + 1} = 2(n+1)$ rows in the ROM. Each row encodes a bit for "Write Tape", a bit for "Move", and $\lceil \log_2(n+1) \rceil$ bits (counting S_0) for the "Next State" for a total of $\lceil \log_2(n+1) \rceil + 2$ bits per row. Thus, the size of the ROM is $2(n+1) \times (\lceil \log_2(n+1) \rceil + 2)$ bits.

(B) The highlighted square represents the current position of the head.

| Time | Sequence |
|---------|------------------|
| 0 (S1) | 0000000000000000 |
| 1 (S2) | 0000000100000000 |
| 2 (S1) | 0000001100000000 |
| 3 (S3) | 0000001100000000 |
| 4 (S2) | 0000001110000000 |
| 5 (S1) | 0000001111000000 |
| 6 (S2) | 0000001111100000 |
| 7 (S2) | 0000001111100000 |
| 8 (S2) | 0000001111100000 |
| 9 (S2) | 0000001111100000 |
| 10 (S2) | 0000001111100000 |
| 11 (S1) | 0000011111100000 |
| 12 (S3) | 0000011111100000 |
| 13 (S0) | 0000011111100000 |

(C) The following 2-state FSM can write 4 ones:

| Current State | Tape Value | Write Tape | Move | Next State |
|---------------|------------|------------|------|------------|
| S1 | 0 | 1 | R | S2 |
| S1 | 1 | 1 | L | S2 |
| S2 | 0 | 1 | L | S1 |
| S2 | 1 | 1 | X | S0 |

(D) If we don't really care which direction we're writing in, and we want to write the largest (but finite) number of ones, an n state FSM can succeed only in writing n ones. The states essentially amount to a counter.

An n state FSM can write an infinite number of ones (even if n=1), provided it wasn't required to halt.

Problem 3. "Beating a Path" [30 points]

Note: For this problem, we will present only the strategies used. The truth tables can be derived in an obvious manner.

(A)

The strategy employed is the following:

1. Check if y is 0 (this can be done by checking if $y-0 == 0$)
2. If $y==0$ then stop (the result is in x)
3. If not, compute $x=x + x$
4. Decrement y
5. Go back to step 1

(B) Note: x^2 can be rewritten as: $x^2 = -x + 2\sum_{i=1}^n i$

The strategy employed is the following:

1. Assign y to x (the value of y will represent the counter to calculate the summation part)
2. Assign $x=-x$ (x will contain the final value)
3. Check if y equals 0
4. If $y==0$ then stop
5. If not, compute $y=x + y$ twice
6. Decrement y
7. Go back to step 3

(C) This can be done in multiple ways.

One of the possible strategies is the following;

1. Assign $x=x + y$

2. Assign $y = x + y$
3. Assign $x = x - y$
4. Assign $y = x + y$
5. Assign $y = x + y$
6. Assign $x = 0 - x$