T02 Tutorial Slides for Week 11

ENEL 353: Digital Circuits — Fall 2019 Term

Steve Norman, PhD, PEng

Electrical & Computer Engineering
Schulich School of Engineering
University of Calgary

19 November, 2019

Topics for today

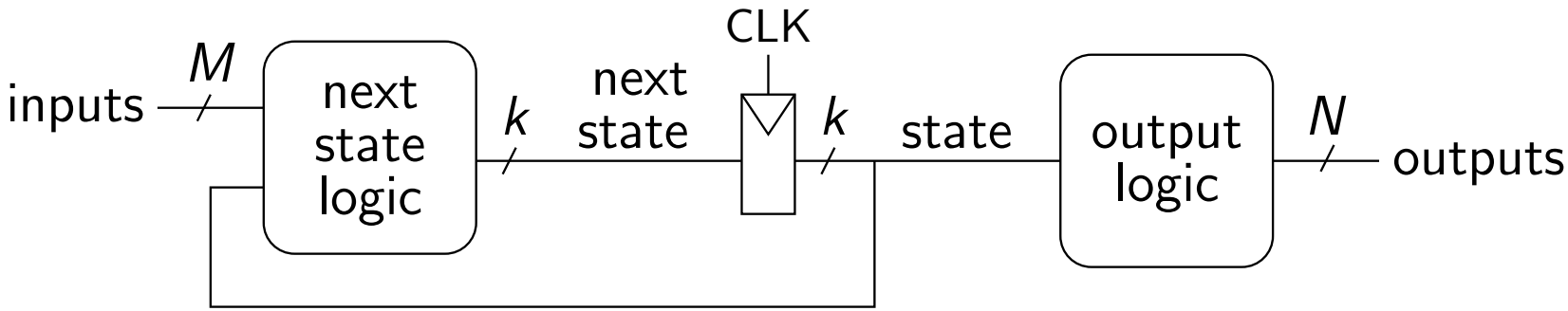
Exercises on FSM structure.

FSM design exercises.

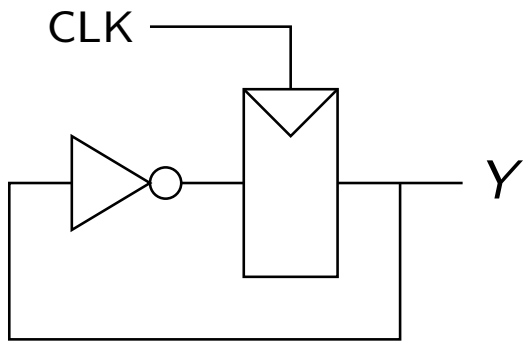
If time permits, an FSM analysis problem.

Exercise 1: Moore FSMs and the clock divider

This is the general structure of a Moore FSM ...



And this is a clock divider circuit ...



The clock divider circuit is just about the simplest possible Moore FSM.

For the clock divider,

- What is M, the number of input bits?
- What is k, the number of state bits?
- What is N, the number of output bits?
- What is correct Boolean algebra for the next-state logic?
- What is correct Boolean algebra for the output logic?

Notes about notation for FSMs (review)

Your ENEL 353 instructors will follow notation used in the course textbook.

S0, S1, S2 and so on are **names of states**. Note that the numbers are the same size as the S, on the same level.

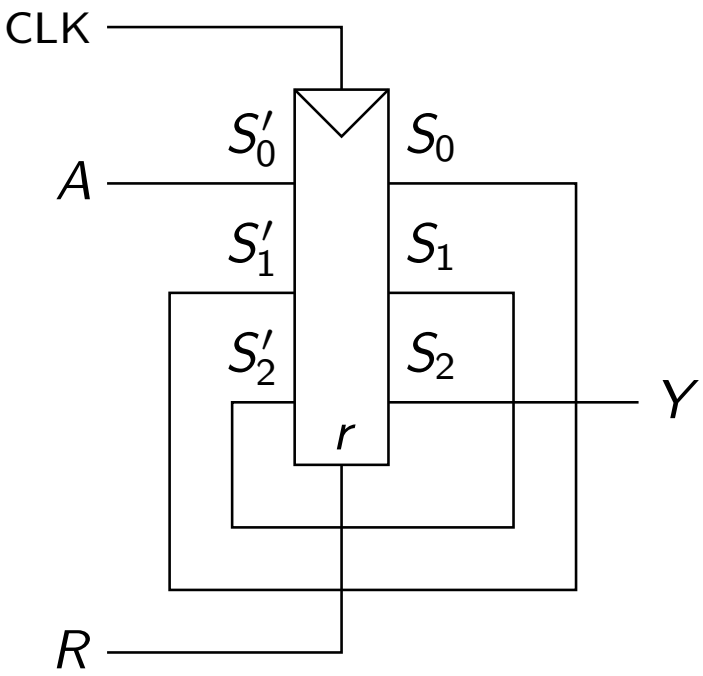
S_0, S_1, S_2 and so on are **state bits**—Q outputs of DFFs. Note that the numbers are **subscripts**.

S_i' is the **next value** of the state bit S_i.

Many books use the $'$ operator for NOT. (For example, in those books A' means what our textbook would write as $\overline{A}$.) **Watch out for that!**

In discussion of FSMs, some books use Q_i for the ith state bit, and $Q_i^\star$ for the next value of Q_i.

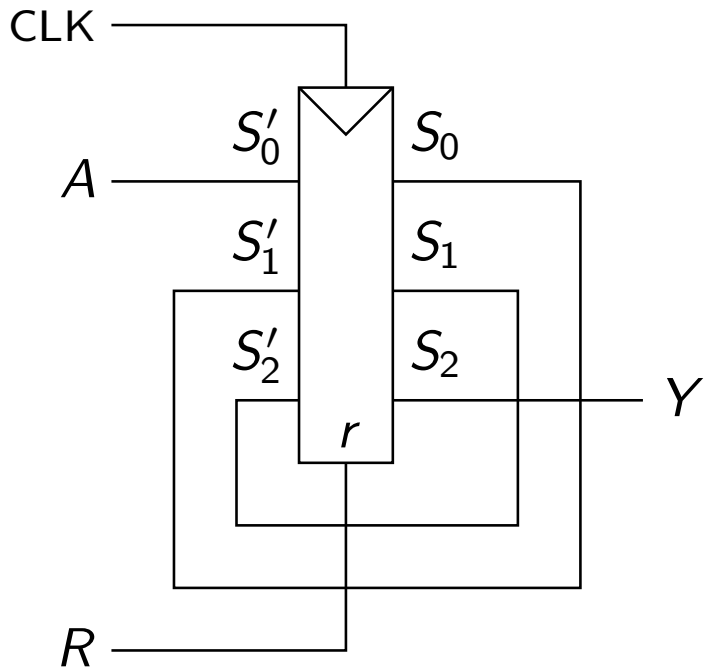
Exercise 2



For this Moore FSM, what are

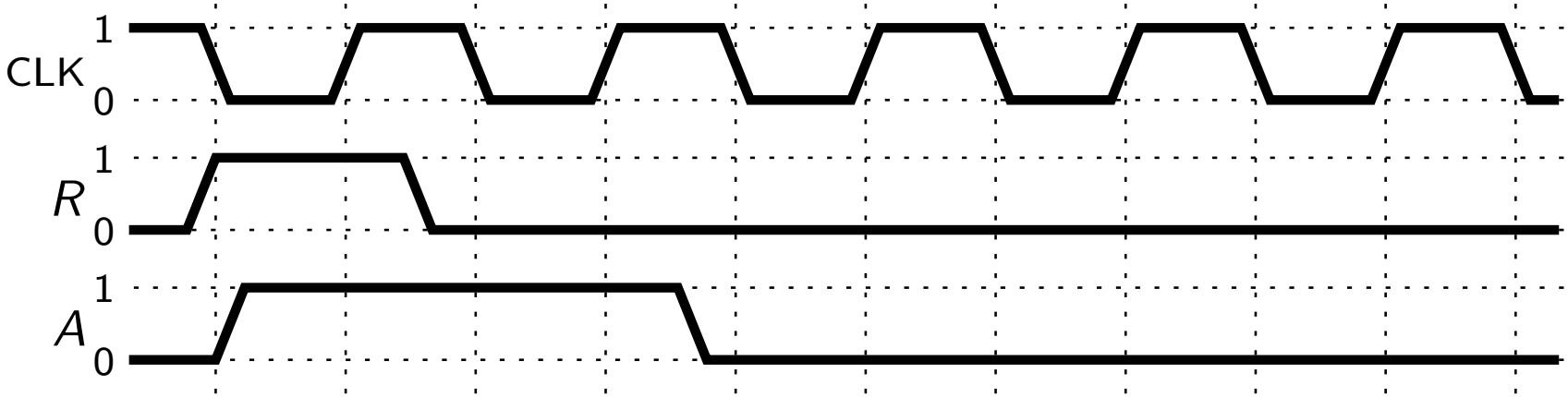
- M, the number of inputs?
- k, the number of state bits?
- the number of possible states of the machine?
- N, the number of output bits?
- next-state equations?
- output equations?

Exercise 2, continued



Let's add waveforms for S_0, S_1 and S_2 to the timing diagram.

Let's assume **synchronous reset** for the three DFFs in the register.



Exercise 3: Simple FSM design

Design an FSM with a single output bit, such that

- on reset the output goes to 0;
- when reset is turned off, the output repeats the sequence of 0 for one clock cycle, 1 for three cycles, 0 for one cycle, 1 for three cycles, and so on.

Exercise 4: Mealy FSM design

Here's a specification for an FSM:

- inputs are CLK, reset, and a 1-bit signal called A;
- output is a 1-bit signal called Y;
- on reset, Y should go to 0 as quickly as possible;
- as soon as there have been at least two post-reset rising edges of CLK, Y should be 1 if the current value of A matches the value of A at the last two rising edges of CLK.

Why does this require a Mealy FSM? (Why can't it be Moore?)

Let's draw a state transition diagram.

Let's make a combined state transition and output table, using symbols (S0, S1, etc.) to represent the states.

For convenience, here's a repeat of the specification:

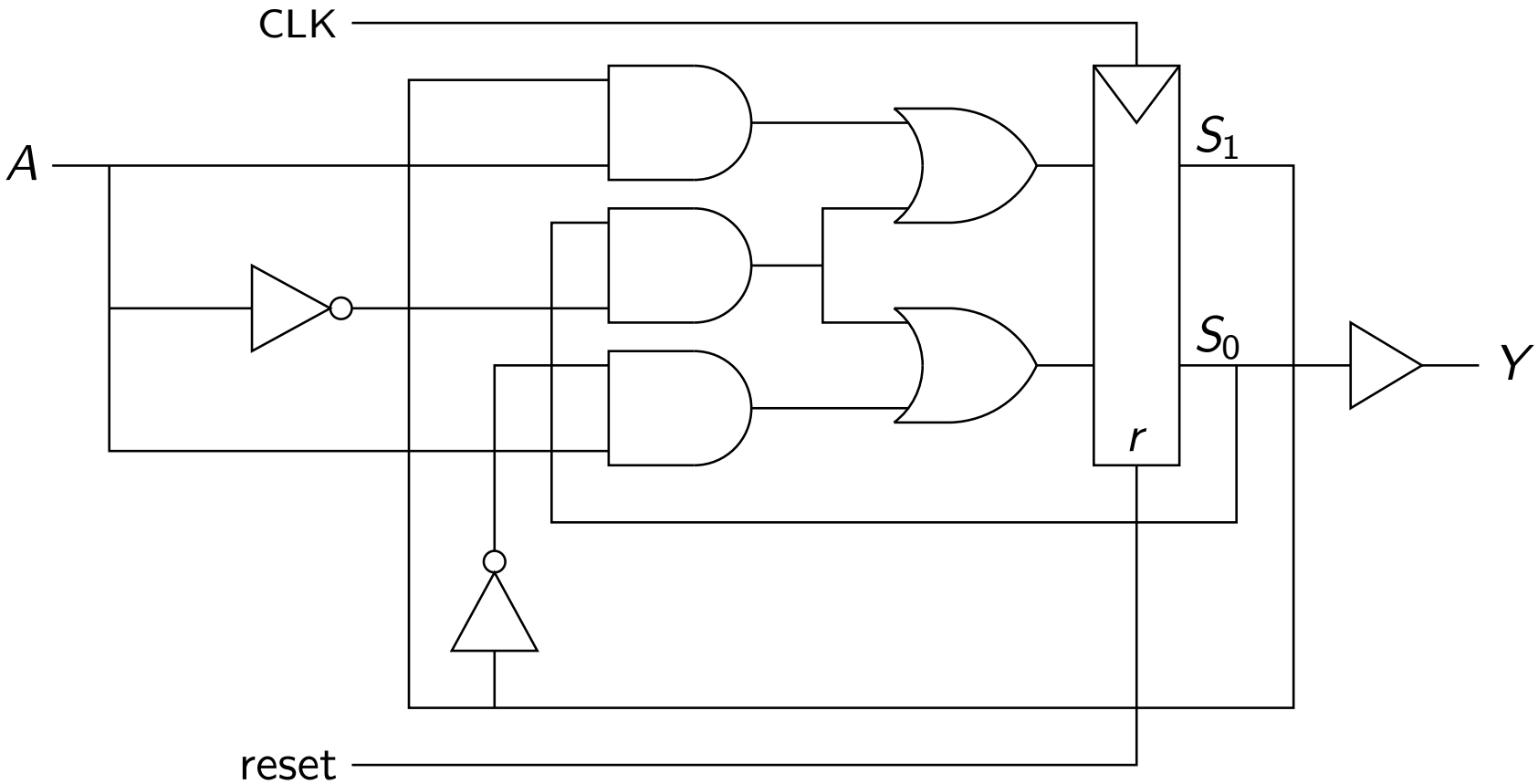
- inputs are CLK, reset, and a 1-bit signal called A;
- output is a 1-bit signal called Y;
- on reset, Y should go to 0 as quickly as possible;
- as soon as there have been at least two post-reset rising edges of CLK, Y should be 1 if the current value of A matches the value of A at the last two rising edges of CLK.

We won't continue all the way to next-state equations and a schematic, but let's think about some of the steps.

What is the minimum number of state bits? If we choose a state encoding with that number of bits, how many rows will the truth table for next-state and output logic have?

Should the state register have synchronous or asynchronous reset?

Exercise 5: Analysis of an FSM



Is this a Moore FSM or a Mealy FSM?

Let's draw a state transition diagram for the circuit.

Exercise 6: FSM analysis, continued

Describe in English what the circuit of Exercise 5 does. Add a timing diagram if that helps with the description.

(This is too fuzzy a question to put on a quiz or exam, but it's useful to think about.)