# EECS 151/251A: Homework № 8

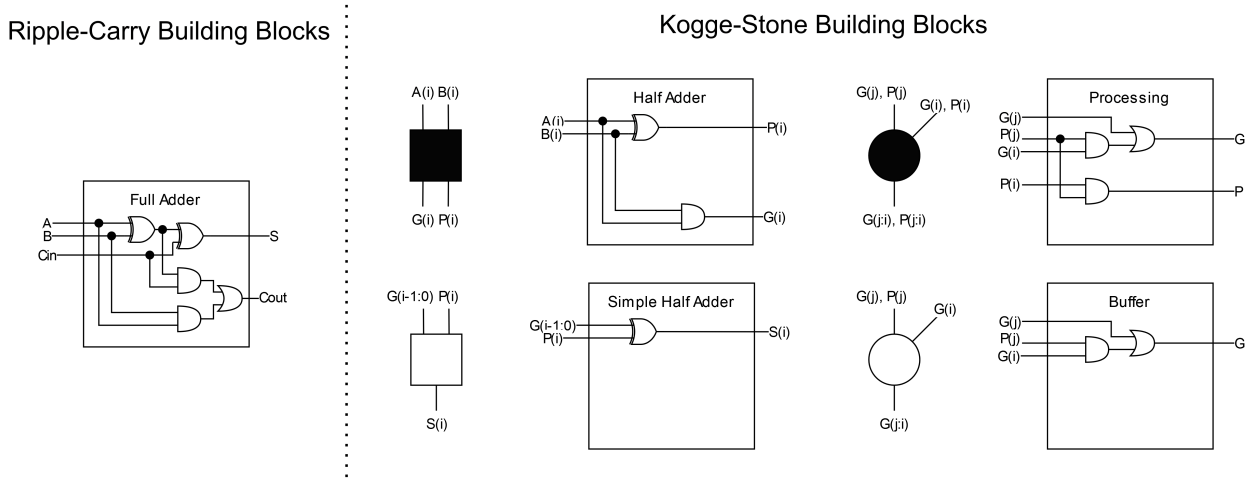
Due Friday, April 22nd

## Problem 1: Energy Adds Up!

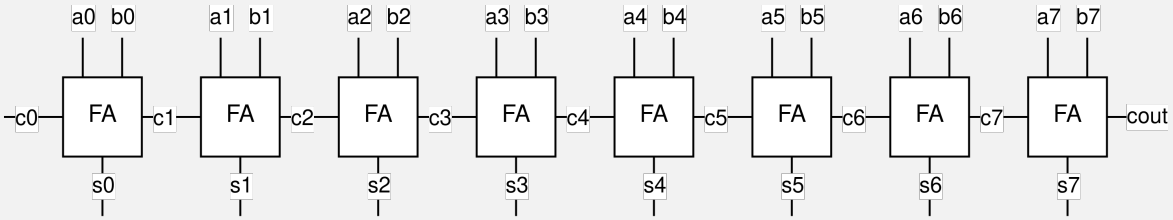
You have been tasked with designing an 8 bit adder to use within an ASIC that needs to meet certain performance and power constraints.

### Part a)

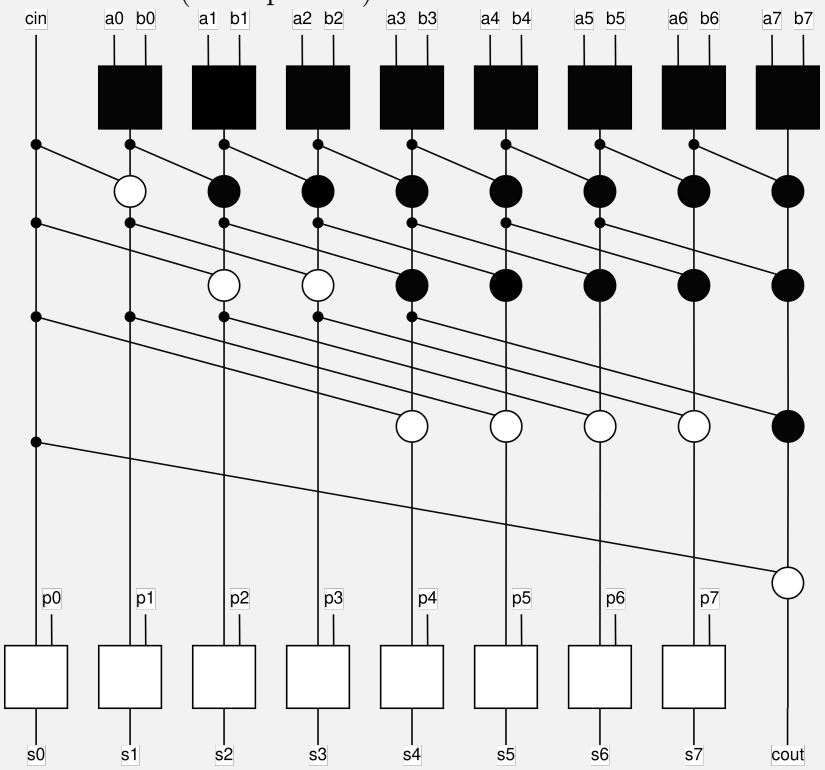
Draw a block diagram for both an 8-bit ripple carry adder as well as an 8 bit radix-2 Kogge-Stone adder. Use the following blocks for the implementation of your designs:



**Ripple Carry Adder:** (`c0` optional)




**Kogge-Stone Adder:** (`cin` optional)



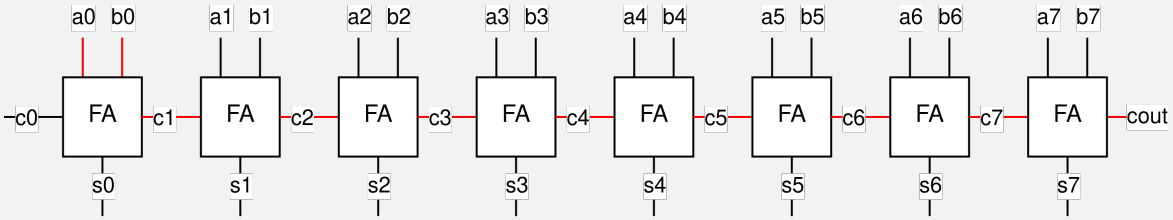

## Part b)

At a voltage of 1.0V, you are given the following propagation delays:

$$t_{\text{AND}} = 5\text{ps}, \quad t_{\text{OR}} = 4\text{ps}, \quad t_{\text{XOR}} = 7\text{ps}$$

For each design in part a), derive the critical path and maximum clock frequency.

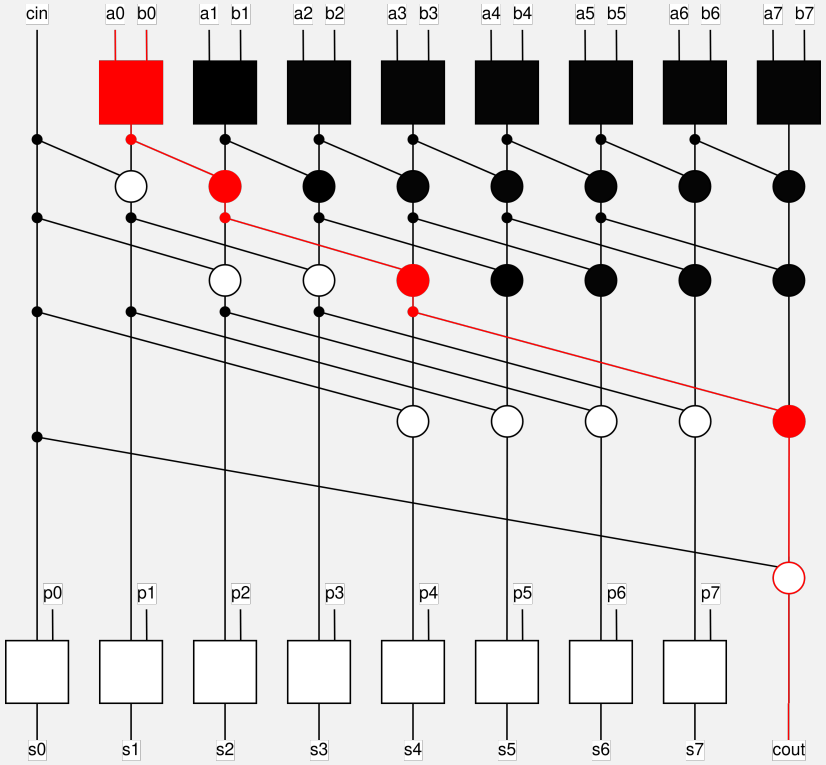
**Ripple Carry Adder:**




$$t_{prop} = t_{AB \to cout} + 7t_{cin \to cout} = (t_{xor} + t_{and} + t_{or}) + 7(t_{and} + t_{or}) = 79ps$$

$$f_{max} = \frac{1}{79ps} = 12.7GHz$$

**Kogge-Stone Adder:**




$$t_{prop} = t_{AB \to G} + 4(t_{G_i \to G_{j:i}}) + t_{G \to S} = t_{xor} + 4(t_{and} + t_{or}) + t_{xor} = 50ps$$

$$f_{max} = \frac{1}{50ps} = 20GHz$$

## Part c)

You are given the following capacitance for each logic gate.

$$C_{\text{AND}} = 2\text{fF}, \quad C_{\text{OR}} = 3\text{fF}, \quad C_{\text{XOR}} = 4\text{fF}$$

**i)** Running at a voltage of 1.0V and assuming an activity factor of $\alpha = 0.2$ across both designs, what is the energy consumed for one add operation for each of the designs?

**Ripple Carry Adder:** (if using `cin`)

$$E = \alpha CV^2 = 0.2 \cdot 8 \cdot C_{FA} \cdot 1.0 = 0.2 \cdot 8 \cdot (2C_{xor} + 2C_{and} + C_{or}) \cdot 1.0 = 24fJ$$

(if not using `cin`)

$$\begin{aligned} E &= \alpha CV^2 = 0.2 \cdot (7 \cdot C_{FA} + C_{HA}) \cdot 1.0 \\ &= 0.2 \cdot (7 \cdot (2C_{xor} + 2C_{and} + C_{or}) + C_{xor} + C_{and}) \cdot 1.0 = 22.2fJ \end{aligned}$$

**Kogge-Stone Adder:**

$$\begin{aligned} E &= \alpha CV^2 \\ &= \alpha(8C_{HA} + 14C_{Processing} + 8C_{Buffer} + 8_{SimpleHA}) \cdot 1.0 \\ &= 0.2 \cdot (8 \cdot (C_{xor} + C_{and}) + 14 \cdot (2C_{and} + C_{or}) + 8 \cdot (C_{and} + C_{or}) + 8C_{xor}) \\ &= 43.6fJ \end{aligned}$$

**ii)** Assuming both adders are operating at their maximum frequency and that they are processing one addition every cycle. What is the power consumption of each of the designs? Only consider dynamic power.

**Ripple Carry Adder:** (if using `cin`)

$$P = 24fJ \cdot 12.7GHz = 0.305mW$$

(if not using `cin`)

$$P = 22.2fJ \cdot 12.7GHz = 0.282mW$$

**Kogge-Stone Adder:**

$$P = 43.6fJ \cdot 20GHz = 0.872mW$$

## Part d)

You are tasked to find the design that can run at a clock frequency of 10GHz while minimizing power. You perform some simulations on your design and find that the propagation delay of all gates can be reduced to 2/3 of the original value by raising the operating voltage to 1.2V.

In order to minimize power consumption while meeting the 10GHz requirement, which design should you choose (ripple-carry or Kogge-Stone), and at what voltage should it operate (either 1.0V or 1.2V)?

Because the designs already meet the desired clock frequency of 10GHz, it is best to use that uses the least energy per operation, which is the ripple carry adder. Using this,

(if using `cin`)

$$P = 24fJ \cdot 10GHz = 0.24mW$$

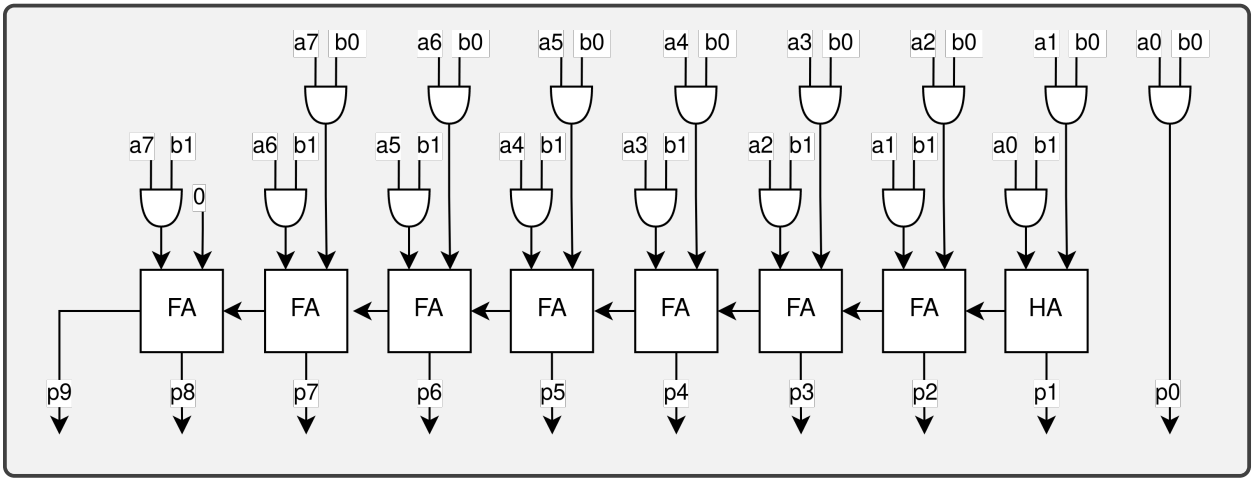
(if not using `cin`)

$$P = 22.2fJ \cdot 10.0GHz = 0.222mW$$

# Problem 2: Product Design

## Part a)

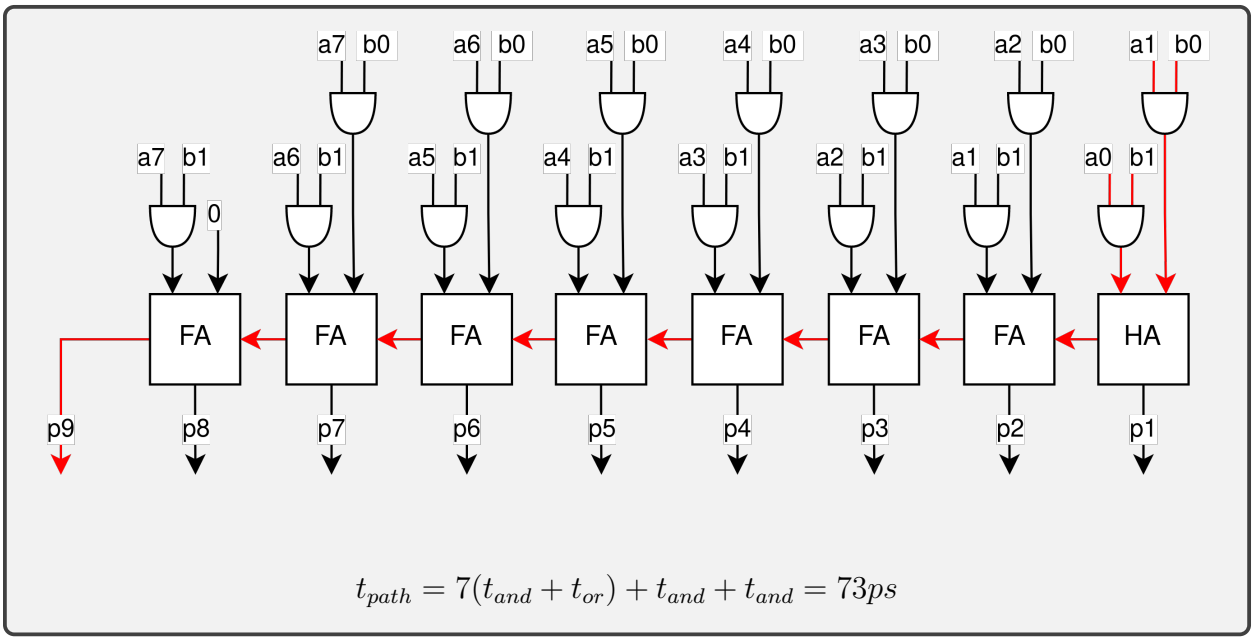
**i)** Draw a block diagram for a parallel (array) multiplier that multiplies an 8-bit integer by a 2-bit integer. Assume you use the same full and half adders as in Problem 1.




**ii)** You are given that the propagation delays are the following:

$$t_{\text{AND}} = 5\text{ps}, \quad t_{\text{OR}} = 4\text{ps}, \quad t_{\text{XOR}} = 7\text{ps}$$

Draw the critical path of the design, and calculate the path delay.

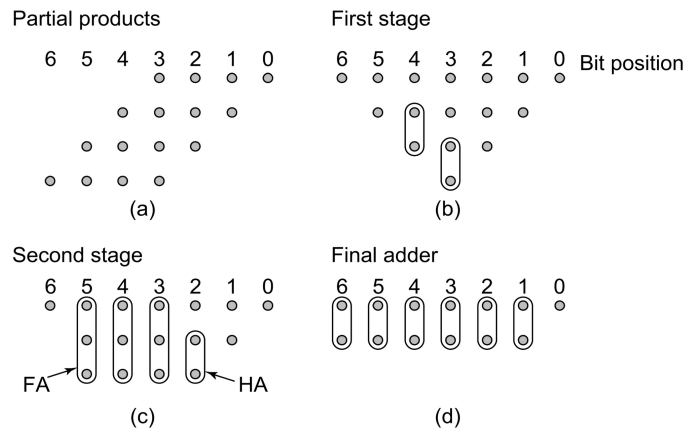


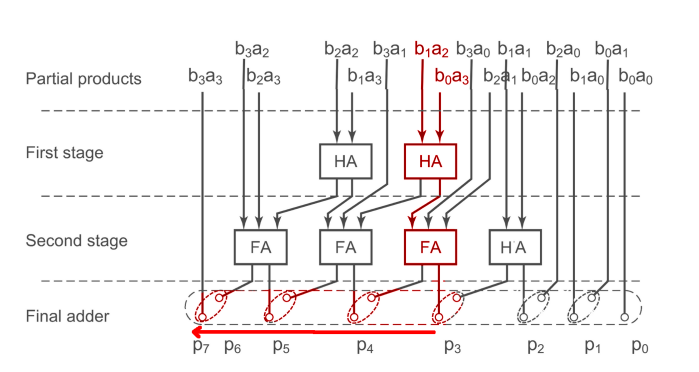

$$t_{path} = 7(t_{and} + t_{or}) + t_{and} + t_{and} = 73ps$$

## Part b)

**i)** You are then tasked to design a 4x4 multiplier. Draw a block diagram for a Wallace-Tree multiplier, and highlight the critical path. Calculate the propagation delay for the design.

Multiple solutions are valid, but an example solutions. Solutions should have some form of multiplexing, and the propagation delay should be through the layers of multiplexers used.




$$
\begin{aligned}
t_{path} &= t_{HA,sum} + t_{FA,sum} + t_{FA,b,carry} + 3t_{FA,carry} \\
&= t_{XOR} + 2t_{XOR} + (t_{XOR} + t_{AND} + t_{OR}) + 4(t_{AND} + t_{OR}) \\
&= 3 \cdot 7ps + 7ps + 4ps + 5ps + 4 \cdot 5ps + 4 \cdot 4ps \\
&= 73ps
\end{aligned}
$$

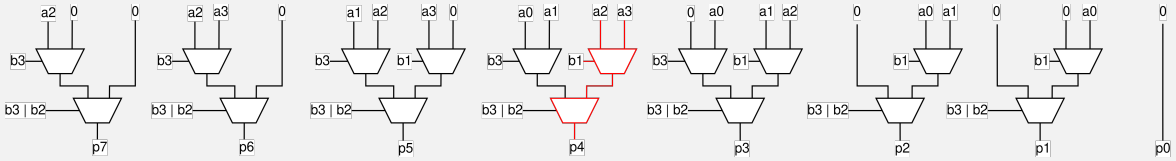
**ii)** You then discover that in your application, exactly one bit of the multiplicand is equal to 1. What kind of operation is this, and how would this simplify your multiplier design?

Because exactly one of the bits is high, we actually know that this is a bit shift operation, rather than a full multiply operation. Because of this, we don't need to perform any addition, and only need to select between the number of bits we need to shift.

**iii)** Draw a block diagram for your design in part b), highlight the critical path, and calculate the propagation delay. Hint: the propagation delay for a 2-input multiplexer is given as

$$t_{\text{MUX}} = 6ps$$

Multiple solutions are valid, but an example solutions. Solutions should have some form of multiplexing, and the propagation delay should be through the layers of multiplexers used.




$$t_{path} = 2t_{\text{MUX}} = 12ps$$