Lab Session
Exercises on Arithmetic for Computers

Problem 1

Design a multi-format adder capable of handling several signed additions in parallel at different precisions. The parallelism is programmed by a 2-bit control string **M** as follows:

- **M**=00 : One 64-bit addition;
- **M**=01 : Two 32-bit additions;
- **M**=10 : Four 16-bit additions;
- **M**=11 : Eight 8-bit additions.

1. Design the top-level architecture of the multi-format adder assuming as a base unit an 8-bit Carry-Ripple Adder (CRA), depicted in Figure 1, and indicating all the necessary logic to handle the carries at the different precisions.

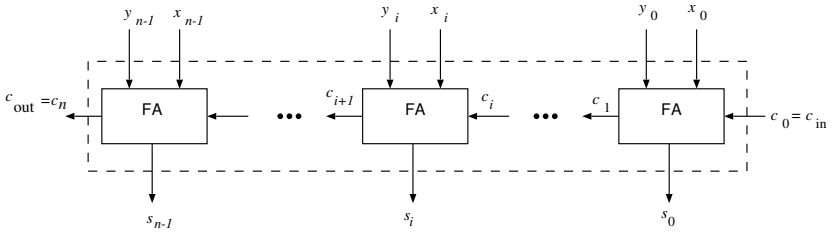
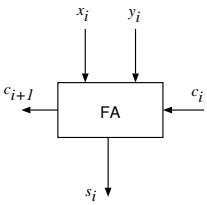


Figure 1: n-bit Carry-Ripple Adder (CRA) architecture.

2. Estimate the delay for the single operation in the four modes assuming the delays in Figure 2 for the full-adder (FA), the basic block of the 8-bit CRA.



$Path$				time
(x_i, y_i)	$\longrightarrow$	s_i	$=$	2
(x_i, y_i)	$\longrightarrow$	c_{i+1}	$=$	2
c_i	$\longrightarrow$	s_i	$=$	1
c_i	$\longrightarrow$	c_{i+1}	$=$	1

Figure 2: Full-Adder (FA): Delays.

Problem 2

Number representation.

1. From the bit vector representation $(x_{n-1}, \ldots, x_1, x_0)$ of a two's complement integer write the expression of its value.
2. With the expression found, convert the bit vector $(1, 1, 0, 1, 0, 0)$ into a signed integer.
3. Write the **binary16** (half precision) floating-point representation of the number 11.625 for the IEEE 754 Standard.

Problem 3 – Floating-Point Addition *(Page numbering for RISC-V 2nd Edition)*

Read the Section on Floating-Point Addition (p. 215–219) and look at Figure 3.15 at page 219.

1. Sketch a drawing describing the hardware necessary to implement the block "Rounding hardware" in Figure 3.15.

2. For correct rounding, an extra control signal must depart from the "Rounding hardware" block. This signal is not drawn in Figure 3.15.

   Describe what this signal is used for, and explain how the rest of the datapath must be modified to accommodate this signal.