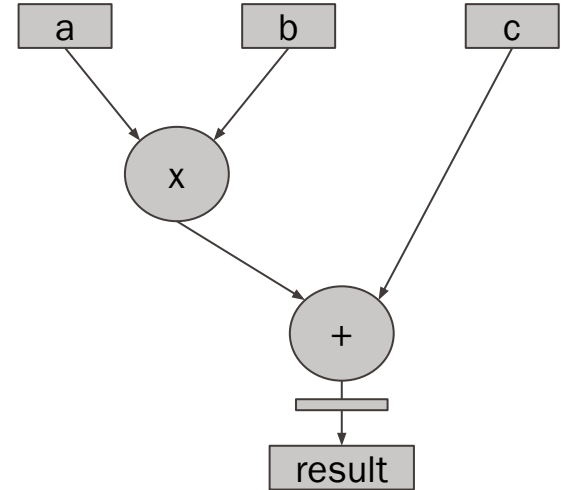


# DSO Project

- Project to understand DSO
- Reference design a MAC unit which does:
  - $a*b + c$
  - $a, b,$  and  $c$  are three signed 32b integers
  - result is a signed 32b output
    - yes, it can overflow and for the purpose of this exercise we are ok with it
- Final project requires a PDF answerings the questions that will follow
- Get the project folder at:



`cp -r /education/classes/2025-2026/CS472/DSO_project`

`/home/YOURNAME/FC_Labs/FC_Labs/labs/dso_project`



# Instructions

In *dso\_project/scripts/fc\_compile.tcl*, *compile.tcl* and *dso.tcl*:

- set the variable `FC_PATH` to the root of **lab**
  - I set it to `"/scrap/users/schiavon/FC_Labs/FC_Labs/"`
- Run the scripts in the *work* folder as you did for FusionCompiler

First, you need to run the pre-compile design with

- a. `make run_fc`



# Q1

First, you need to run the pre-compile design with

a. `make run_fc`

**What does this script do? What is the area (floorplan) size of the mac\_unit?**

**Write a short description (max 3 sentences) about what the script does and what is the scope of it.**



## Q2

Now it's time for DSO

a. `make run_dso`

**What does this script do?**

**What is the area (floorplan) size of the mac\_unit?**

**How many launchers does it use?**

**What is the QoR strategy used?**



# Q3

Analyze the results of the `make run_dso`

**How many runs did DSO do?**

**How many have been saved? Why?**

**Compared to the baseline, how smaller is the smallest design? Give a percentage and report the values of the baseline and the smallest. How much faster/slower it is (WNS means worst negative slack)? In which folder can I find its database?**

**Compared to the baseline, how faster is the fastest design? Give a percentage and report the values of the baseline and the fastest. How much bigger/smaller it is? In which folder can I find its database?**



## Q4 [optional]

You can modify the SystemVerilog file with another multiplier (keep input and output the same). Try to first pre-compile it, and then compile it with FusionCompiler (and remember to increase the FloorPlan Area in case it fails).

Then, remove the compilation step from the FusionCompiler script and re-run it so that you can create a new database where DSO can start exploring from. Then re-run the DSO flow and analyze the results compared to your baseline.

Run **make clean** to clean up the previous runs