EE 533 Lab #3
Mini-Intrusion Detection Engine Design
Instructor: Young Cho

NOTE: You must use GitHub to store all source codes and reports associated with the lab assignments. You must commit/push all the modifications for the assignments with detailed descriptions at least once daily when working on the lab assignment. Please submit the GitHub link and record of your commits and detailed commit description to Brightspace before the due date.

For this lab, you will create an intrusion detection system (mini-IDS) using schematics, IP Cores, and verilog. This is an individual assignment. You may help each other understand, but please do not copy.

1 Schematic Capture

- Enter the following directly into a file called `busmerge.v`

```
module busmerge(da, db, q);
    input [47:0] da;
    input [63:0] db;
    output [111:0] q;
    assign q = {da,db}
endmodule
```

- Enter the following schematics into the Xilinx ISE schematic capture tool. Give each schematic the indicated name. The 8-bit comparator (`comp8.sch`) is a built-in Xilinx component and is included for reference, you do not need to re-implement it.

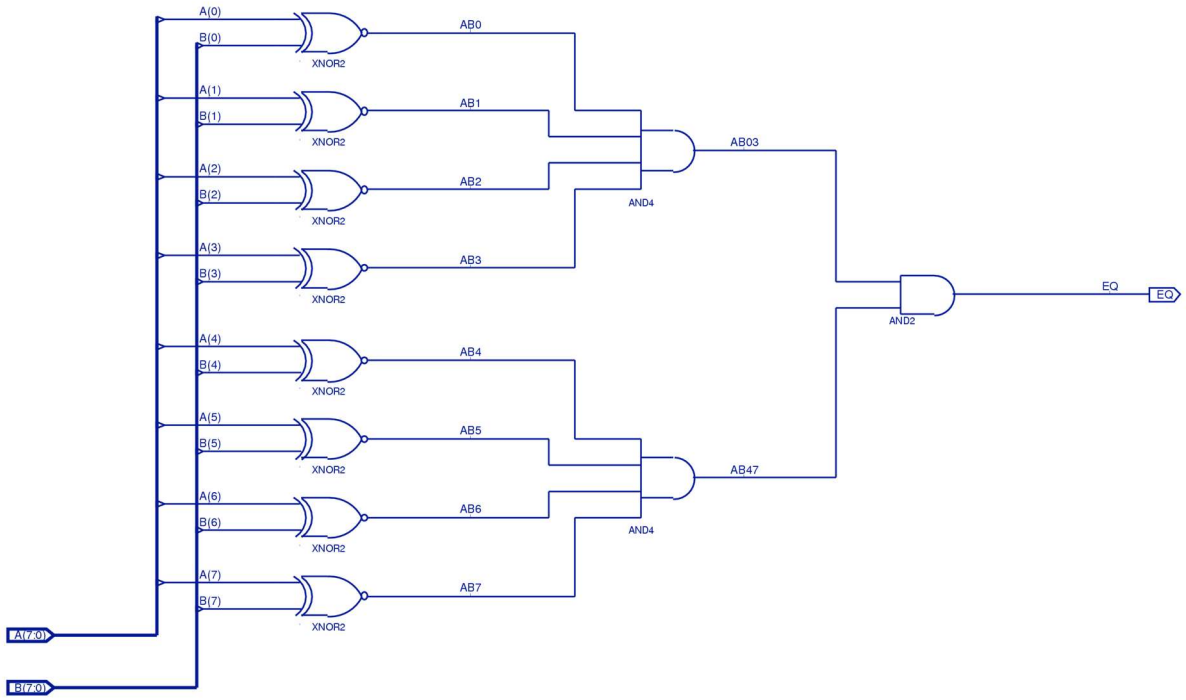

Figure 1: comp8.sch

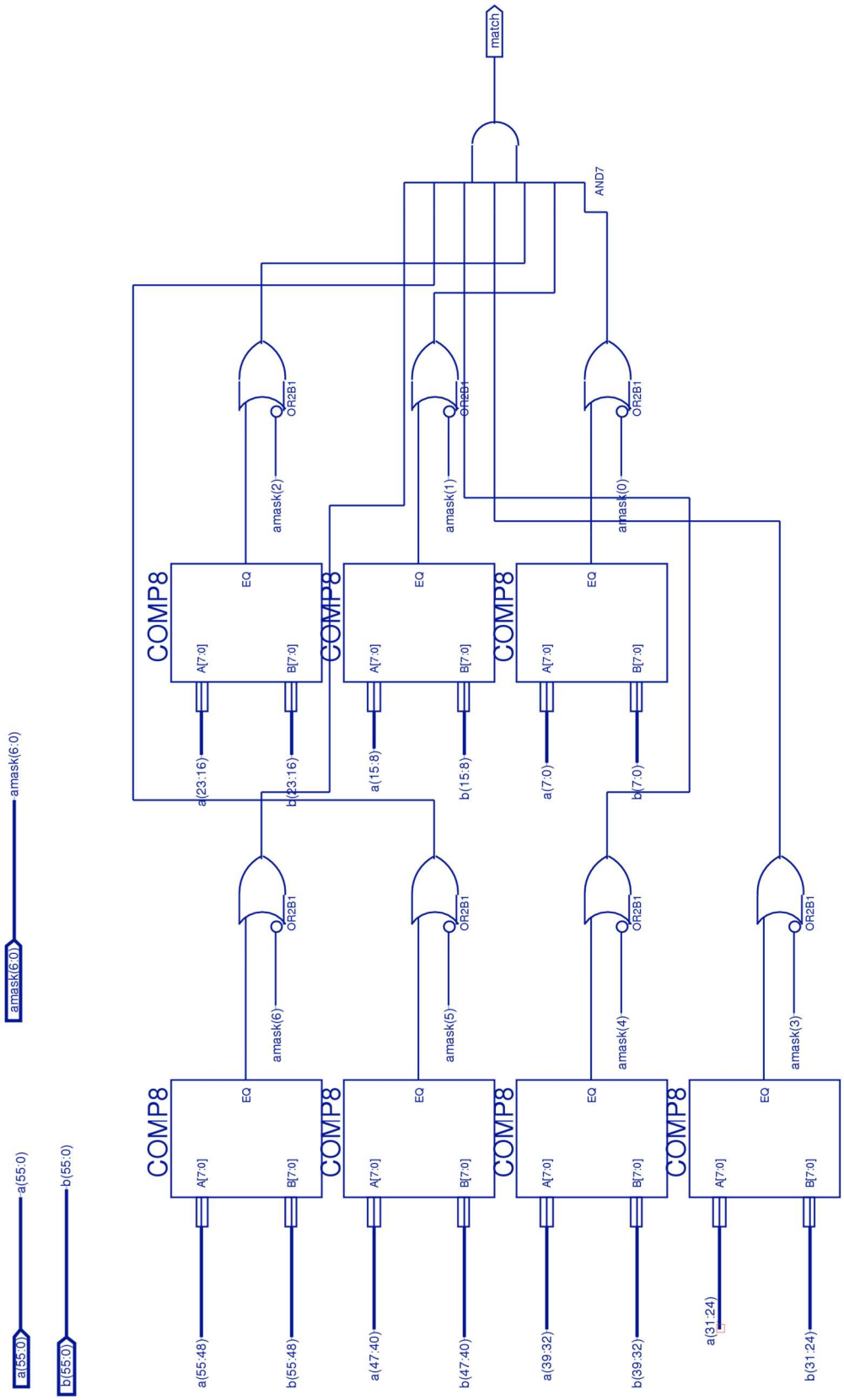

Figure 2: comparator.sch

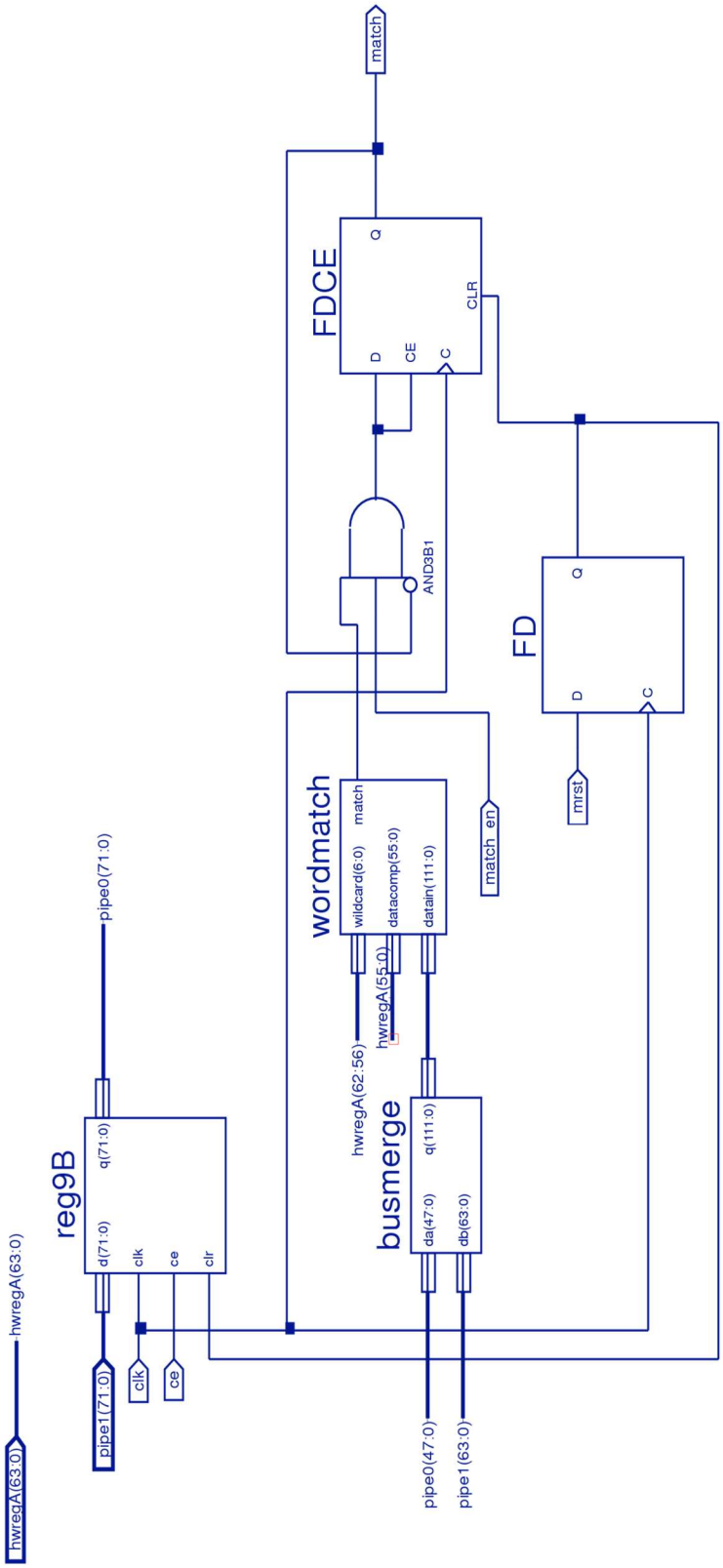

Figure 3: detect7B.sch

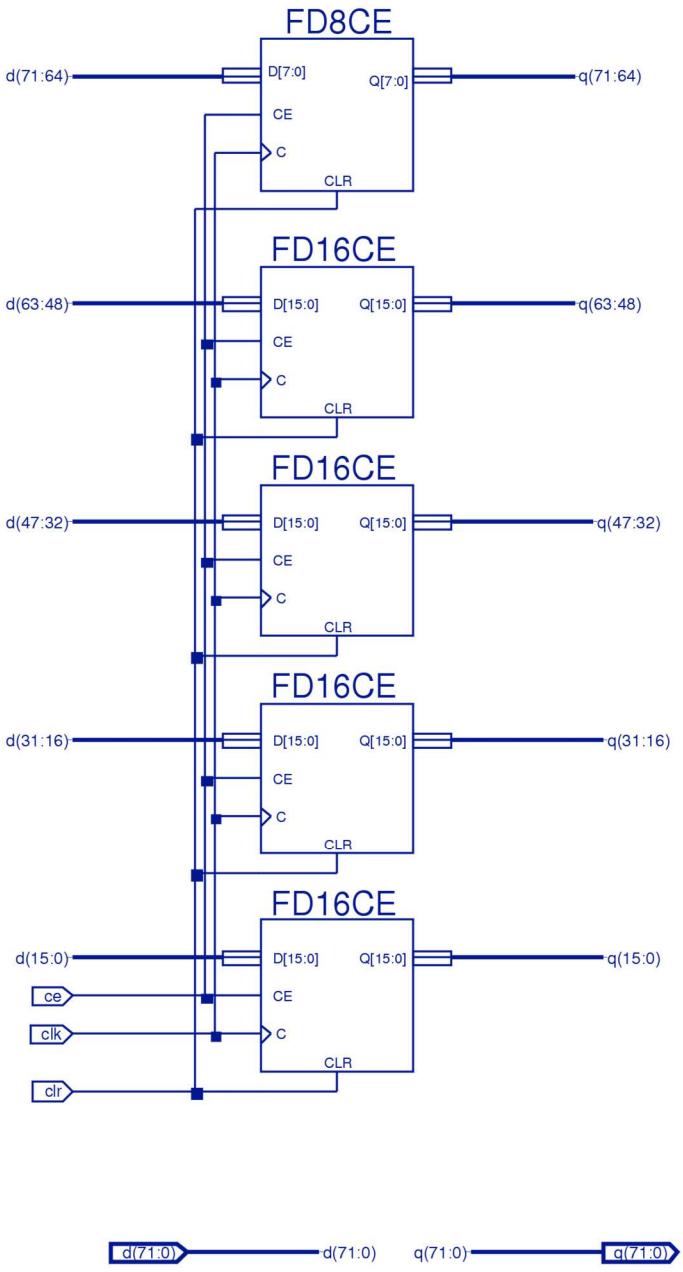

Figure 4: reg9B.sch

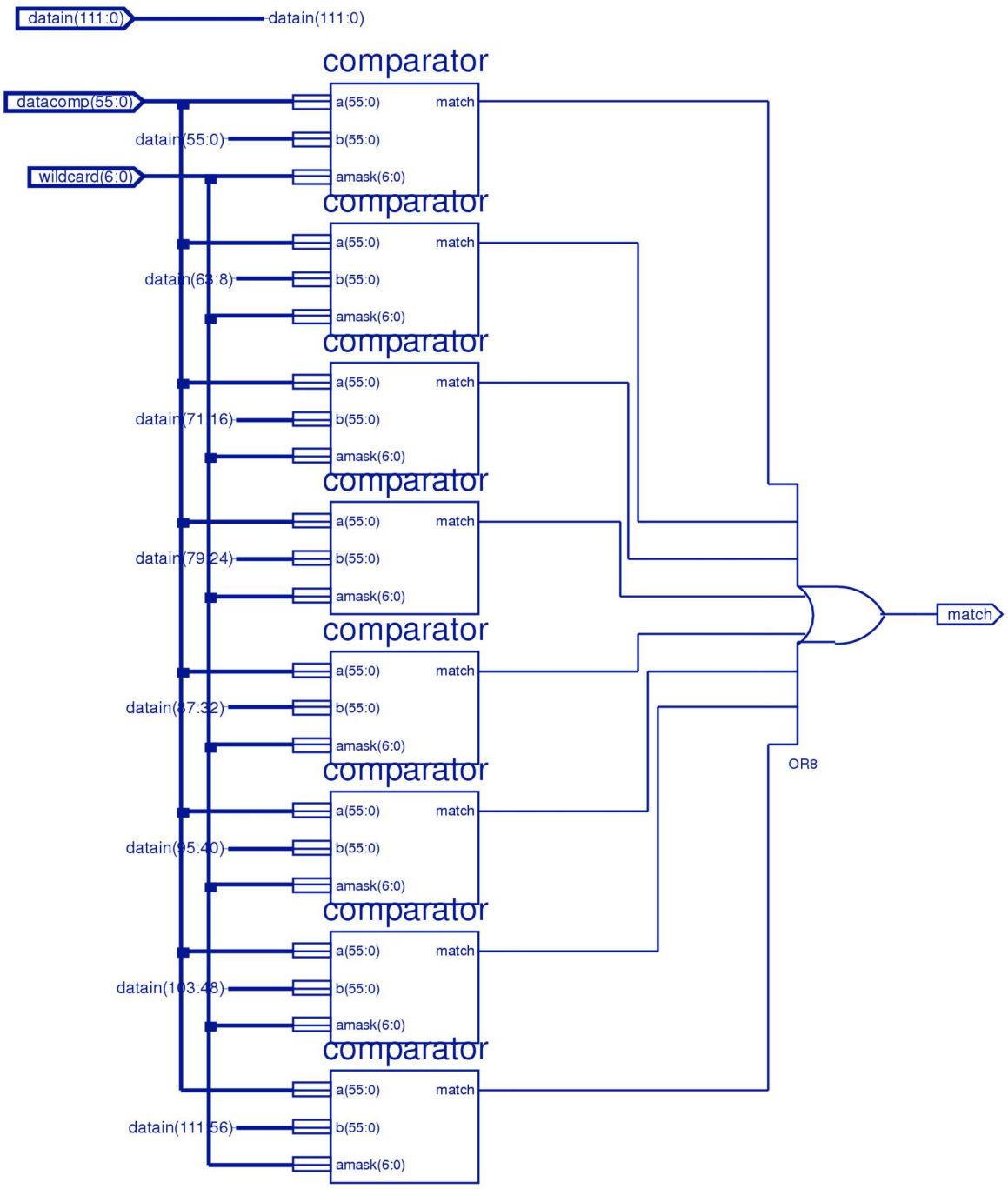

Figure 5: wordmatch.sch

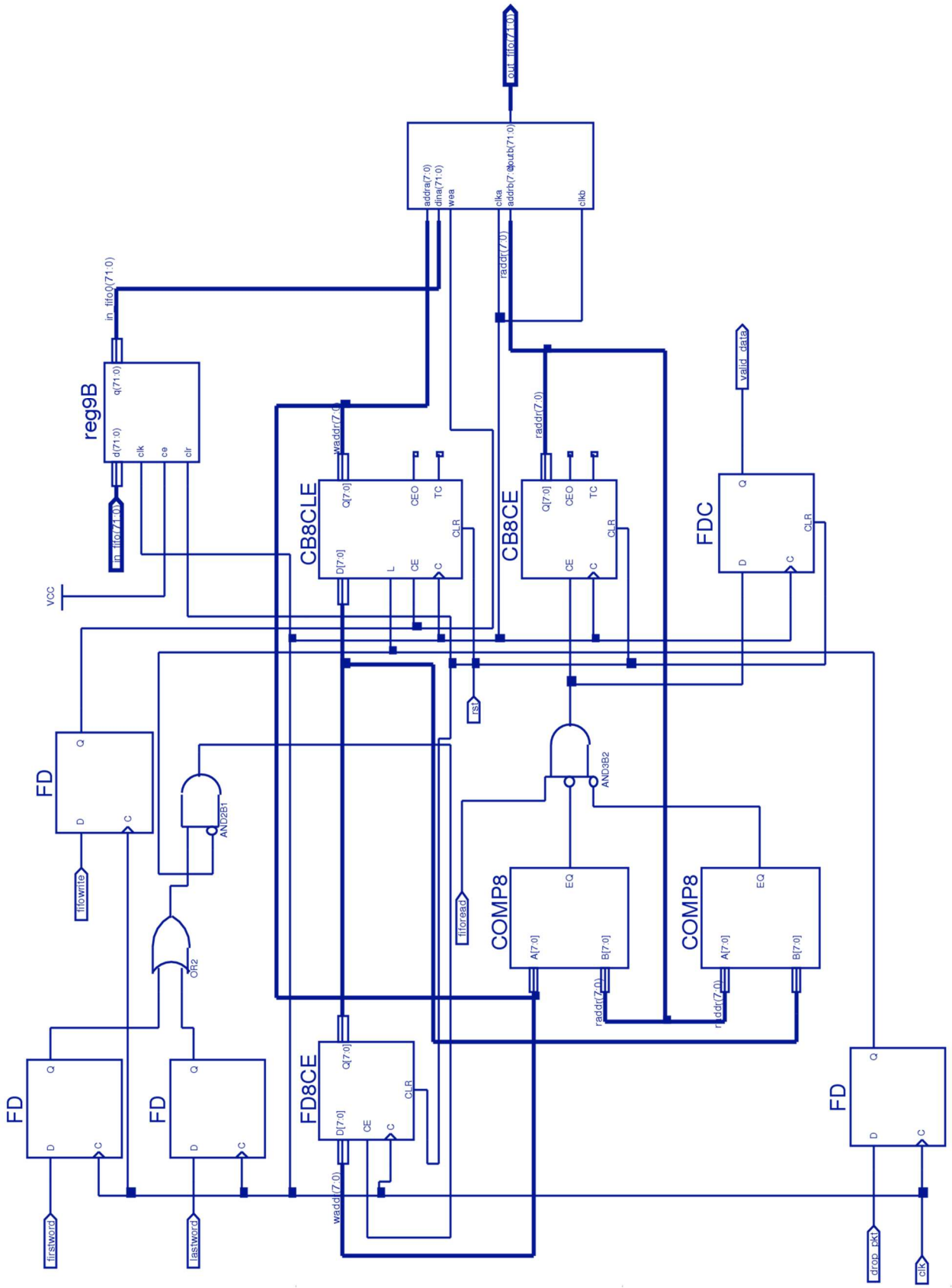

Figure 6: dropfifo.sch

- Add a new IP Core component for a 9-byte (72 bit) wide synchronous dual port memory (1 read, 1 write port).
- Convert all of your schematics to Verilog (ISE provides this feature). The following video demonstrates how to do this:

  http://oasysresearch.com/ise_sch2ver.mkv

- Take a look at the created Verilog. Do they make sense? Which do you think easier: entering the schematics or writing Verilog? Why? In which cases might you do the other?
- Download the `lab5_mini_ids_src.zip` file.
  - In this file you will find two directories.
  - Extract `ids_sim` directory to your ISE project. You should now be able to simulate the mini-IDS using the `ids_tb.tbw` testbench. The other files are needed to emulate the pieces of the NetFPGA that are around your design. Run the testbench and take a screen shot. Describe what the testbench does and how it shows that the mini-IDS is functioning.
  - `ids_hw` directory is for the next lab in hardware synthesis

2 Submission and Demonstration

1) Write and submit your lab report
   a) Explain the pattern matching algorithm in the report
   b) Include the answers to your lab in this report.
      i) What is the purpose of `AMASK[6:0]`?
      ii) What exactly does `busmerge.v` do?
      iii) What do the `comp8` modules do in this schematic?
      iv) What is the purpose of `dual9Bmem` in `dropfifo.sch`?
   c) Please also include all of the screen capture of Schematics as well as generated Verilog
2) Use a screen video capture tool to demonstrate your design.
   a) Capture the video of the screen (using free tools like OBS studio) during this process for edited demo video that will be uploaded on youtube

      https://obsproject.com/

   b) Add your voice narration to the video to succinctly describe your lab results
   c) Upload your video to YouTube for your demo submission.
   d) Please make sure you start this process long before the deadline since this can take a some time to get correct.