Julian Andrews ID 94841201
Marcos Avila ID 58548681
Noah Djenguerian ID 26957883
Ryan Gonzales ID 57555019

CSE 132L FINAL PROJECT

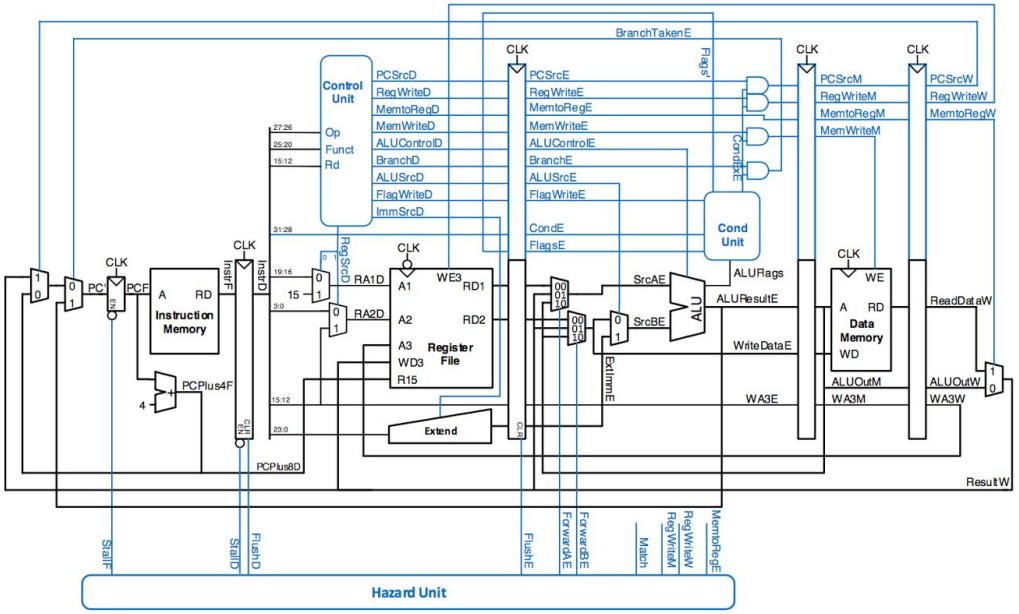
1. Architecture Design Description

The final project involved converting our fundamental single-cycle processor design into a multi-cycle processor. The primary challenge of this project was to make sure that all of the stages were correct synchronized and able to communicate with each other. Various small errors, such as not instantiating particular inputs into selectors or sending incorrect port/instruction values in between stages, made this project difficult.

Through rigorous debugging, we were able to successfully implement all of the operations and pass the core tests from Project 2. We followed the hazard unit methods described in the Chapter 7 slides (pgs. 160-170). The fundamental processor layout was primarily drawn from the schematic provided from the notes.

At a verilog design perspective, we decided to instantiate all of the pipe stages in the datapath module. All intermediate wires between the stages were created locally in the datapath module. Some of the stages have the pipeline registers embedded inside of them (like the decode and execute module). Others have the pipeline registers located externally of the stage modules.

The following is a picture of the core datapath of our project, provided from the notes:



2. TestBench Evaluation

We reused Project 2's testbench. We commented and uncommented the IMEM instructions for each test. The IMEM data is located in verif.

The sequence of running the instructions for our particular processor is the following:

1. Enter the sim folder of our project.
2. Run: source setup.csh
3. Run: source pre_compile.csh
4. Run: vlog -64 -sv -f ../design/rtl.cfg
5. Run: vlog -64 -sv -f ../verif/tb.cfg -work work
6. Run: vopt -64 testbench -o testbench_opt +acc -work work
7. Run: vsim -64 -c testbench_opt -do sim.do

3. Synthesis Results

After running multiple versions of the synthesis by slightly decrementing/incrementing the clock period of our processor, we were able to get it the following numbers.

These are just some of the information specified by the Synopsys compilation. More is in the syn folder.

For total area (in arm.area.rpt, units are micro-meters squared):

```
Combinational area:          6717.025986
Buf/Inv area:              386.044737
Noncombinational area:       3433.485546
Macro/Black Box area:          0.000000
Net Interconnect area:       2939.864814

Total cell area:           10150.511532
Total area:                13090.376347
```

The total area of the design is approximately 13090 micro-meters squared. About 3000 micro-meters squared is just spent connecting the wires together, the remaining 10000 micro-meters is on combinatorial and non-combinatorial logic.

For Total Power (in arm.power_hier.rpt, voltage is in micro-watts.):

```
Global Operating Voltage = 1.05
Power-specific unit information :
   Voltage Units = 1V
   Capacitance Units = 1.000000ff
   Time Units = 1ns
   Dynamic Power Units = 1uW    (derived from V,C,T units)
   Leakage Power Units = 1pW
```

```
--------------------------------------------------------------------------------
                                Switch   Int      Leak     Total
Hierarchy                       Power    Power    Power    Power    %
--------------------------------------------------------------------------------
arm                             250.093 4.60e+03 7.55e+09 1.24e+04 100.0
```

The total power is 7.55 micro-watts.

For Frequency (in synth.tcl ,clock period is 0.92 nano-seconds)

```
#                  D E S I G N   P A R A M E T E R S
#================================================================================
=
set PROJECT_NAME              "112L-ARM"
set TOP                       "arm"
set FILES                     "files_arm"
set clock_period              0.92
```

Our clock period correctly operates at 0.92 nano-seconds per clock cycle, approximately 1.08 Giga-hert per second.