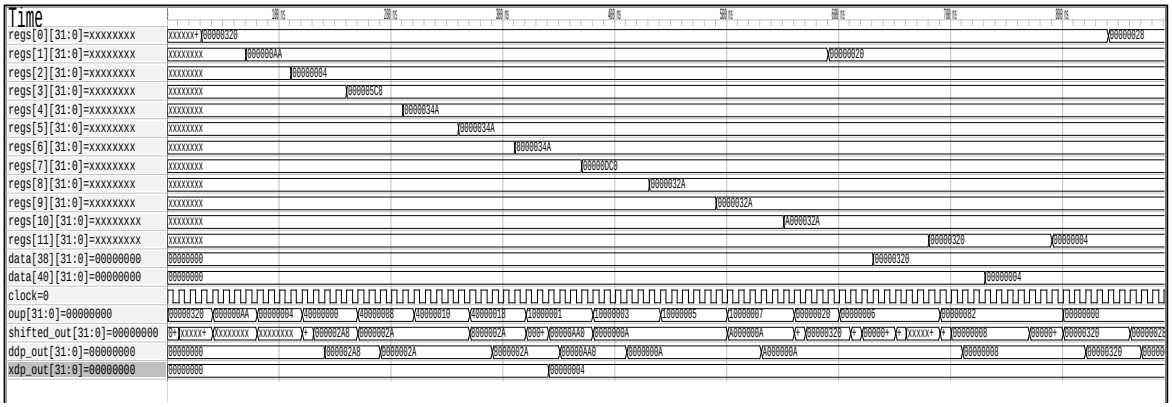
Divyansh Mittal , 2020CS10342 Computer Architecture cs1200342@iitd.ac.in

1. Stage 5

Mentor precision for synthesis and quartus for netlist and used EDA Playground to code. gtkWave for the waves output

a) Implementing shift and rotate

i. gtk Waves



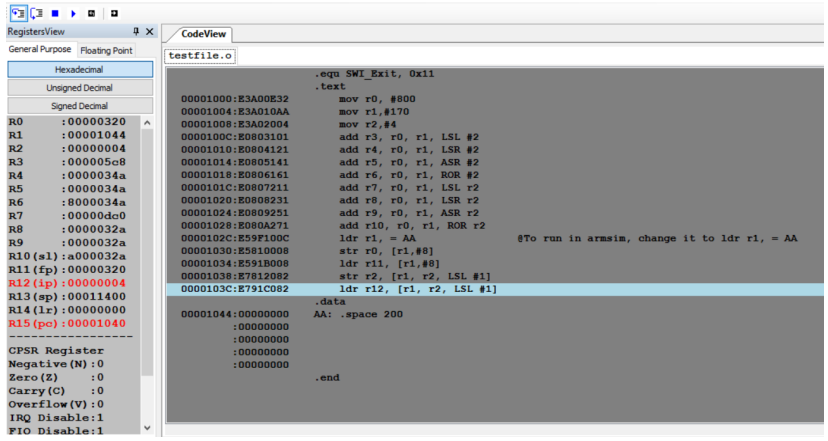


Figure 1: gtkWave along with armsim output to verify working

```
        0 => X"E3A00E32",
        1 => X"E3A010AA",
        2 => X"E3A02004",
        3 => X"E0803101",
        4 => X"E0804121",
        5 => X"E0805141",
        6 => X"E0806161",
        7 => X"E0807211",
        8 => X"E0808231",
        9 => X"E0809251",
        10 => X"E080A271",
        11 => X"E3A01020",
        12 => X"E5810006",
        13 => X"E591B006",
        14 => X"E7812082",
        15 => X"E791B082",
        others => X"00000000"
```

```
    mov r0, #800
    mov r1, #170
    mov r2, #4
    add r3, r0, r1, LSL #2
    add r4, r0, r1, LSR #2
    add r5, r0, r1, ASR #2
    add r6, r0, r1, ROR #2
    add r7, r0, r1, LSL r2
    add r8, r0, r1, LSR r2
    add r9, r0, r1, ASR r2
    add r10, r0, r1, ROR r2
    mov r1, 32              @To run in armsim, change it to ldr r1, = AA and 8 to align word
    str r0, [r1,#6]
    ldr r11, [r1,#6]
    str r2, [r1, r2, LSL #1]
    ldr r12, [r1, r2, LSL #1]
```

b) How to run the code

Simply run the makefile by the command "make" to view the outputs in gtkwave.[1]

c) Basic working

2 modules shifter and rotator are added that implement these operations. 2 new states have been added. First of these new stated is used to load a register X, which stores the third register value which is the shift amount, the other state performs the shift/ rotate as per the insruction.

d) Netlist

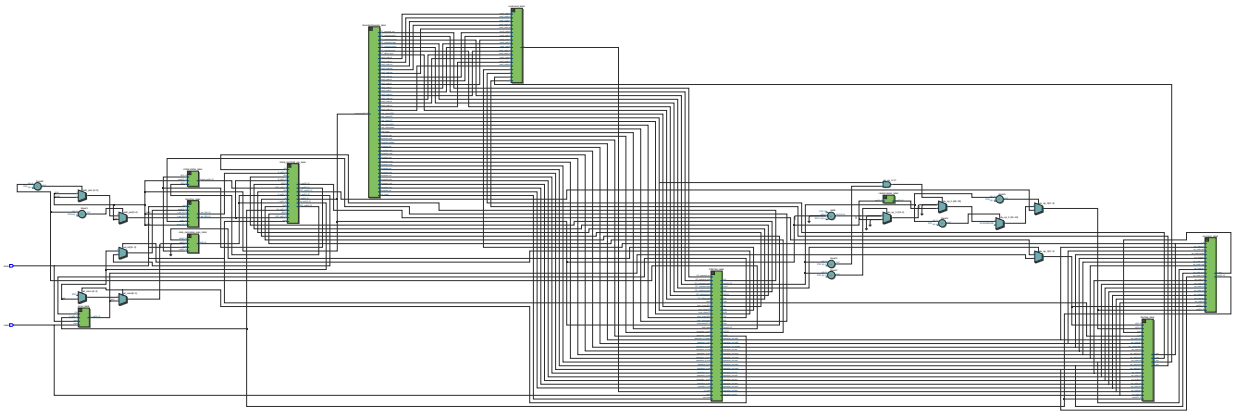


Figure 2: Net List

i. Log Output

```
[2022-03-12 13:58:00 UTC] vlib work && vcom '-2019' '-o' ALU.vhd Cond.vhd data_mem.vhd Flags.vhd FSM.vhd IDAB_Reg.vhd Instr.vhd MyTypes.vhd PC.vhd Register.vhd Rotator.vhd Shifter.vhd design.vhd testbench.vhd && vsim -c -do "vsim TB; vcd file dump.vcd; vcd add -r sim:/*; run -all; exit"
VSIMSA: Configuration file changed: '/home/runner/library.cfg'
ALIB: Library "work" attached.
work = /home/runner/work/work.lib
Aldec, Inc. VHDL Compiler, build 2020.04.130
VLM Initialized with path: "/home/runner/library.cfg".
DAGGEN WARNING DAGGEN_0523: "The source is compiled without the -dbg switch. Line breakpoints and assertion debug will not be available."
COMP96 File: ALU.vhd
COMP96 Compile Entity "ALU"
```

[1]For this we do need gtkWave, which can easily be installed on a linux system with the command "sudo apt install gtkwave"

```
COMP96 Compile Architecture "alu_arch" of Entity "ALU"
COMP96 File: Cond.vhd
COMP96 Compile Entity "cond"
COMP96 Compile Architecture "arch" of Entity "cond"
COMP96 File: data_mem.vhd
COMP96 Compile Entity "data_mem"
COMP96 Compile Architecture "arch" of Entity "data_mem"
COMP96 File: Flags.vhd
COMP96 Compile Entity "flag"
COMP96 Compile Architecture "arch" of Entity "flag"
COMP96 File: FSM.vhd
COMP96 Compile Entity "FSM"
COMP96 Compile Architecture "behaviour" of Entity "FSM"
COMP96 File: IDAB_Reg.vhd
COMP96 Compile Entity "IDAB_reg"
COMP96 Compile Architecture "IDAB_reg_arch" of Entity "IDAB_reg"
COMP96 File: Instr.vhd
COMP96 Compile Entity "Decoder"
COMP96 Compile Architecture "Behavioral" of Entity "Decoder"
COMP96 File: MyTypes.vhd
COMP96 Compile Package "MyTypes"
COMP96 Compile Package Body "MyTypes"
COMP96 File: PC.vhd
COMP96 Compile Entity "pc"
COMP96 Compile Architecture "arch" of Entity "pc"
COMP96 File: Register.vhd
COMP96 Compile Entity "Reg"
COMP96 Compile Architecture "reg_arch" of Entity "Reg"
COMP96 File: Rotator.vhd
COMP96 Compile Entity "rotator"
COMP96 Compile Architecture "arch" of Entity "rotator"
COMP96 File: Shifter.vhd
COMP96 Compile Entity "shifter"
COMP96 Compile Architecture "arch" of Entity "shifter"
COMP96 File: design.vhd
COMP96 Compile Entity "processor"
COMP96 Compile Architecture "arch" of Entity "processor"
COMP96 File: testbench.vhd
COMP96 Compile Entity "TB"
COMP96 Compile Architecture "behavior" of Entity "TB"
COMP96 Incorrect order of units detected.
COMP96 Automatic reorder and incremental recompilation of required units in progress.
COMP96 File: ALU.vhd
COMP96 Compile Entity "ALU"
COMP96 Compile Architecture "alu_arch" of Entity "ALU"
COMP96 File: Cond.vhd
COMP96 Compile Entity "cond"
COMP96 Compile Architecture "arch" of Entity "cond"
COMP96 File: FSM.vhd
COMP96 Compile Entity "FSM"
COMP96 Compile Architecture "behaviour" of Entity "FSM"
COMP96 File: Instr.vhd
COMP96 Compile Entity "Decoder"
COMP96 Compile Architecture "Behavioral" of Entity "Decoder"
COMP96 File: Flags.vhd
```

```
COMP96 Compile Entity "flag"
COMP96 Compile Architecture "arch" of Entity "flag"
COMP96 File: design.vhd
COMP96 Compile Architecture "arch" of Entity "processor"
COMP96 Top-level unit(s) detected:
COMP96 Entity => TB
COMP96 Compile success 0 Errors 0 Warnings Analysis time : 0.1 [s]
dmesg: read kernel buffer failed: Operation not permitted
dmesg: read kernel buffer failed: Operation not permitted
# Aldec, Inc. Riviera-PRO version 2020.04.130.7729 built for Linux64 on June 10, 2020.
# HDL, SystemC, and Assertions simulator, debugger, and design environment.
# (c) 1999-2020 Aldec, Inc. All rights reserved.
# ELBREAD: Elaboration process.
# ELBREAD: Elaboration time 0.0 [s].
# KERNEL: Main thread initiated.
# KERNEL: Kernel process initialization phase.
# ELAB2: Elaboration final pass...
# ELAB2: Create instances ...
# KERNEL: Time resolution set to 1ps.
# ELAB2: Create instances complete.
# SLP: Started
# SLP: Elaboration phase ...
# SLP: Elaboration phase ... skipped, nothing to simulate in SLP mode : 0.0 [s]
# SLP: Finished : 0.0 [s]
# ELAB2: You do not have a license to run VHDL performance optimized simulation. Contact Aldec for
    ordering information - sales@aldec.com.
# ELAB2: Elaboration final pass complete - time: 0.0 [s].
# KERNEL: Warning: You are using the Riviera-PRO EDU Edition. The performance of simulation is reduced
    .
# KERNEL: Warning: Contact Aldec for available upgrade options - sales@aldec.com.
# KERNEL: Kernel process initialization done.
# Allocation: Simulator allocated 8080 kB (elbread=427 elab2=7508 kernel=144 sdf=0)
# KERNEL: ASDB file was created in location /home/runner/dataset.asdb
# KERNEL: PLI/VHPI kernel's engine initialization done.
# PLI: Loading library '/usr/share/Riviera-PRO/bin/libsystf.so'
# KERNEL: WARNING: NUMERIC_STD.TO_INTEGER: metavalue detected, returning 0
# KERNEL: Time: 0 ps, Iteration: 0, Instance: /TB/uut/Decoder_label, Process: line__34.
# KERNEL: WARNING: NUMERIC_STD.TO_INTEGER: metavalue detected, returning 0
# KERNEL: Time: 0 ps, Iteration: 0, Instance: /TB/uut/Decoder_label, Process: line__48.
# KERNEL: WARNING: NUMERIC_STD.TO_INTEGER: metavalue detected, returning 0
# KERNEL: Time: 0 ps, Iteration: 0, Instance: /TB/uut/data_mem_label, Process: line__34.
# KERNEL: WARNING: NUMERIC_STD.TO_INTEGER: metavalue detected, returning 0
# KERNEL: Time: 0 ps, Iteration: 0, Instance: /TB/uut/Reg_label, Process: line__23.
# KERNEL: WARNING: NUMERIC_STD.TO_INTEGER: metavalue detected, returning 0
# KERNEL: Time: 0 ps, Iteration: 0, Instance: /TB/uut/Reg_label, Process: line__24.
# KERNEL: WARNING: NUMERIC_STD.TO_INTEGER: metavalue detected, returning 0
# KERNEL: Time: 0 ps, Iteration: 0, Instance: /TB/uut/rotator_label, Process: line__15.
# KERNEL: WARNING: NUMERIC_STD.TO_INTEGER: metavalue detected, returning 0
# KERNEL: Time: 0 ps, Iteration: 0, Instance: /TB/uut/shifter_label, Process: shift_by_set.
# KERNEL: Simulation has finished. There are no more test vectors to simulate.
# VSIM: Simulation has finished.
Finding VCD file...
./dump.vcd
[2022-03-12 13:58:03 UTC] Opening EPWave...
Done
```