COL216 Lab Assignment-2 (S6)

- Mayank Mangla (2020CS50430)

The aim of this project is to design hardware for implementing a processor that can execute a subset of ARM instructions described below. Starting with a skeleton design, the hardware is built in several stages, adding some functionality at every stage. The designs are to be expressed in VHDL and then simulated and synthesized.

Stage 6: Until previous stage, we have constructed a multi cycle processor for some basic instruction set. We have now introduced some more instructions to our processor to execute. Various other forms of DT instructions have been introduced which contains byte and half word transfers and signed unsigned transfers. For this we have created a new entity that forms a gateway between the register file and memory which modifies the data that is being written according to the instruction. This entity is not clocked. This takes DR as input and modifies it according to the signed extension to be done or not and type of transfer while loading it into the register.
The report for the stage 6 is given below:

1 Program Information

This program has been test on "eda playground" using "-2019 -o" flags.
There were some changes in some of the entities and glue logic previously defined.

1. **FSM**:= One state is updated in load to implement the write back in load instruction in which we were writing the data of the memory in DR register. This states can be used to write in the register file the updated address in case we have to write back.

2. **PM connect**:= A new entity is defined as PMConnect whose purpose is to alter the data inputs to register file and memory accordingly.

The logs for the new entity "PMConnect" is given below

```
# Info: ******************************************************************
# Info: Device Utilization for 7A100TCSG324
# Info: ******************************************************************
# Info: Resource                          Used    Avail   Utilization
# Info: ------------------------------------------------------------------
# Info: IOs                               139     210      66.19%
# Info: Global Buffers                    1       32        3.12%
# Info: LUTs                              91      63400     0.14%
# Info: CLB Slices                        14      15850     0.09%
# Info: Dffs or Latches                   64      126800    0.05%
# Info: Block RAMs                        0       135       0.00%
# Info: DSP48E1s                          0       240       0.00%
# Info: ------------------------------------------------------------------
# Info: ******************************************************************
# Info: Library: work    Cell: PMConnect    View: implement_pmc
# Info: ******************************************************************
# Info:  Number of ports :                          139
# Info:  Number of nets :                           378
# Info:  Number of instances :                      308
# Info:  Number of references to this view :          0
# Info: Total accumulated area :
# Info:  Number of Dffs or Latches :                 64
# Info:  Number of LUTs :                            91
# Info:  Number of Primitive LUTs :                 103
# Info:  Number of LUTs with LUTNM/HLUTNM :          24
# Info:  Number of accumulated instances :          308
# Info: ******************************************************************
```

2 Test Cases

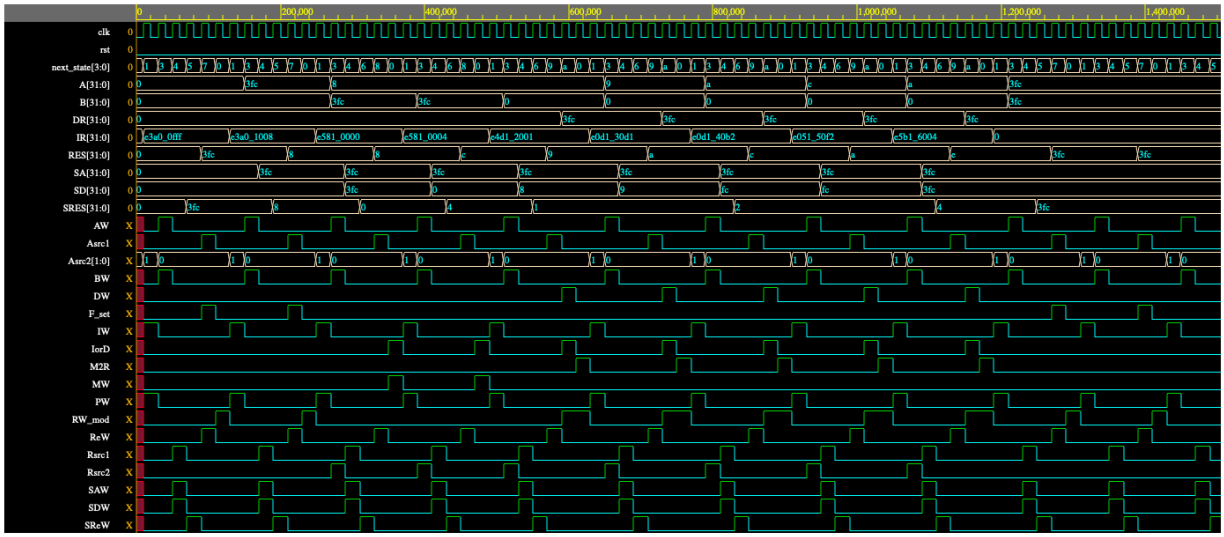
The program has been tested for this test case. This includes the byte, half and full word transfer instructions along with signed and unsigned DT instructions.
Instructions in Hex code

```
0 => x"E3A00FFF",
1 => x"E3A01008",
2 => x"E5810000",
3 => x"E5810004",
4 => x"E4D12001",
5 => x"E0D130D1",
6 => x"E0D140B2",
7 => x"E05150F2",
8 => x"E5B16004",
others => x"00000000"
```

Instructions in ARM code

```
mov R0,#0x03fc
mov R1, #8
str R0, [R1]
str R0, [R1,#4]
ldrb R2,[R1],#1
ldrsb R3,[R1],#1
ldrh R4,[R1],#2
ldrsh R5,[R1],#-2
ldr R6, [R1,#4]!
```

Below are the images of the ep wave for the above test cases:



3 Conclusion

The program has given the correct result for the above test cases. The submission file (.zip) contains 18 files other than this report file.
Eight files each one for one component (e.g. alu_desgin.vhd), five shift units are there which are component of the main Shift Rotate Unit, one contains the glue code (glue_code.vhd), one contains the test bench for the program (testbench.vhd), one contains the Finite state machine (fsm_design.vhd), one contains the data structures (given along with the assignment) (myTypes.vhd) and one is run.do file that has been used to synthesis.