Lab Report 6

Ashish Prasad, 21D180009

October 9, 2022

1 Overview of the experiment

The objective was to detect a given sequence of string among given sequence of string using Behavioral-Dataflow modelling.

2 Experiment setup

2.1 Digital Design for Sequence detector

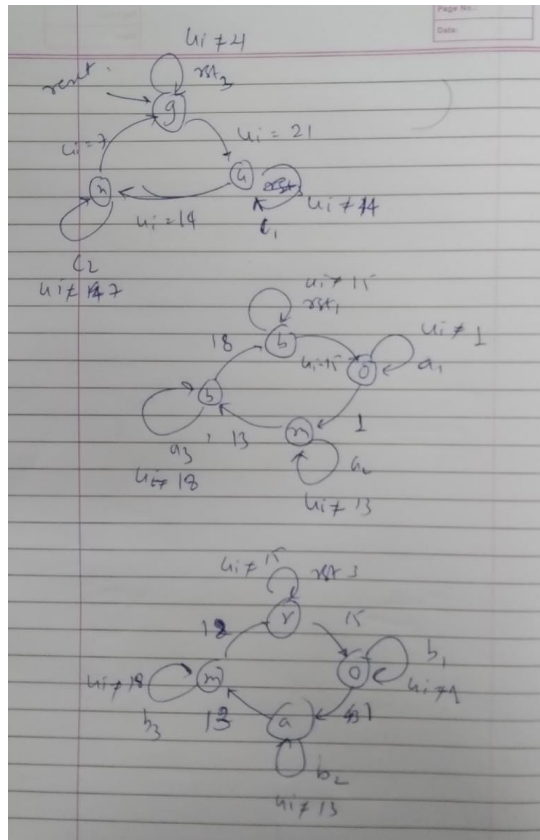The State diagram of the sequence detector is shown as follows:

Figure 1: State diagram

The code for the same Universal shifter design is shown :

2.1.1 Sequence: ROAM

```
state_transition_proc_roam:process(inp,y_present2)
begin
case y_present2 is
when rst2=>
if(unsigned(inp)= 18) then --r has been detected
y_next2<= b1; -- Fill the code here
else
y_next2 <= y_present2;
end if;
when b1=>
if(unsigned(inp)= 15) then --o has been detected
y_next2<= b2; -- Fill the code here
else
y_next2<= y_present2;
end if;
when b2=>
if(unsigned(inp)= 1) then --a has been detected
y_next2<= b3; -- Fill the code here
else
y_next2<= y_present2;
end if;
when b3=>
if(unsigned(inp)= 13) then --m has been detected
y_next2<= rst2; -- Fill the code here
else
y_next2 <= y_present2;
end if;
end case;
end process;
```

2.1.2 Sequence: BOMB

```
state_transition_proc_bomb:process(inp,y_present1)
begin
case y_present1 is
when rst1=>
if(unsigned(inp)= 2) then --b has been detected
y_next1<= a1; -- Fill the code here
else
y_next1 <= y_present1;
end if;
when a1=>
if(unsigned(inp)= 15) then --0 has been detected
y_next1<= a2; -- Fill the code here
```

```
else
y_next1<= y_present1;
end if;
when a2=>
if(unsigned(inp)= 13) then --m has been detected
y_next1<= a3; -- Fill the code here
else
y_next1 <= y_present1;
end if;
when a3=>
if(unsigned(inp)= 2) then --b has been detected
y_next1<= rst1; -- Fill the code here
else
y_next1 <= y_present1;
end if;
end case;
end process;
```

2.1.3 Sequence: GUN

```
state_transition_proc_gun:process(inp,y_present3)
begin
case y_present3 is
when rst3=>
if(unsigned(inp)= 7) then --g has been detected
y_next3<= c1; -- Fill the code here
else
y_next3 <= y_present3;
end if;
when c1=>
if(unsigned(inp)= 21) then --u has been detected
y_next3<= c2; -- Fill the code here
else
y_next3<= y_present3;
end if;
when c2=>
if(unsigned(inp)= 14) then --n has been detected
y_next3<= rst3; -- Fill the code here
else
y_next3 <= y_present3;
end if;
end case;
end process;
```

2.1.4 Output Process

```
output_proc_bomb:process(y_present1, inp)
begin
```

```vhdl
case y_present1 is
when rst1=>
outp1<='0';
when a1=>
outp1<='0';
when a2=>
outp1<='0';
when a3 =>
if ( unsigned(inp)=2) then
outp1<='1';
else
outp1<='0';
end if ;
end case;
end process;

output_proc_roam:process(y_present2, inp)
begin
case y_present2 is
when rst2=>
outp2<='0';
when b1 =>
outp2<='0';
when b2 =>
outp2<='0';
when b3 =>
if ( unsigned(inp)=13) then
outp2<='1';
else
outp2<='0';
end if;
end case;
end process;

output_proc_gun:process(y_present3, inp)
case y_present3 is
when rst3=>
outp3<='0';
when c1=>
outp3<='0';
when c2 =>
if ( unsigned(inp)=14) then
outp3<='1';
else
outp3<='0';
end if;
end case;
```

```
end process;

outp<= outp1 or outp2 or outp3;
```

3   Observations

3.1   Digital Design

The Design passed all the test cases as given in TRACEFILE the RTL Simulation for the same is shown below.
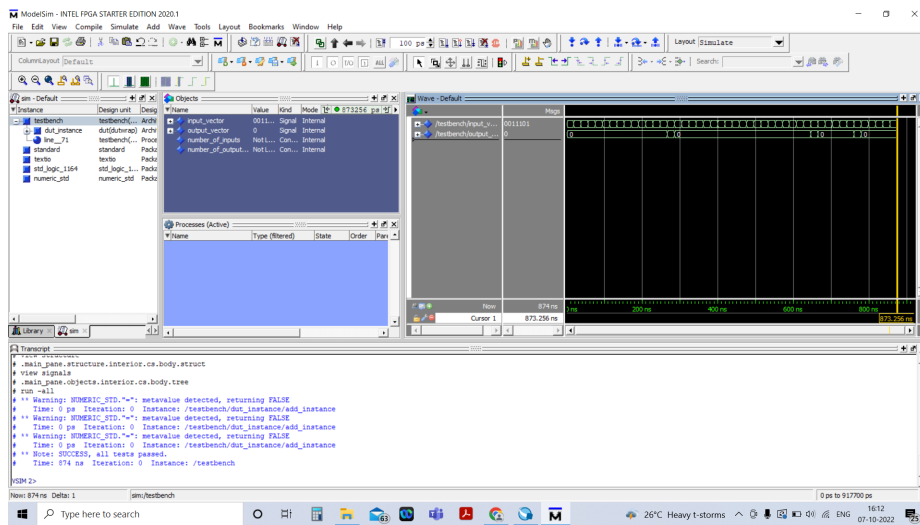
3.2   RTL Simulation

Figure 2: RTL Simulation

3.3   Final Observation

```
out - Notepad
File  Edit  Format  View  Help
0000010 0 Success
0000011 0 Success
0001000 0 Success
0001001 0 Success
1001000 0 Success
1001001 0 Success
0100100 0 Success
0100101 0 Success
0111000 0 Success
0111001 0 Success
0011100 0 Success
0011101 0 Success
1010100 0 Success
1010101 0 Success
0111000 1 Success
0111001 0 Success
0111100 0 Success
0111101 0 Success
0001100 0 Success
0001101 0 Success
0000100 0 Success
0000101 0 Success
1001000 0 Success
1001001 0 Success
0010000 0 Success
0010001 0 Success
1001100 0 Success
1001101 0 Success
0011000 0 Success
0011001 0 Success
1001000 0 Success
1001001 0 Success
0111100 0 Success
0111101 0 Success
0110100 1 Success
0110101 0 Success
0110100 0 Success
0110101 0 Success
1100100 0 Success
1100101 0 Success
0001000 1 Success
0001001 0 Success
```