ECE 501 ASSIGNMENT 7 : STACK

Roshni Uppala

1011735230

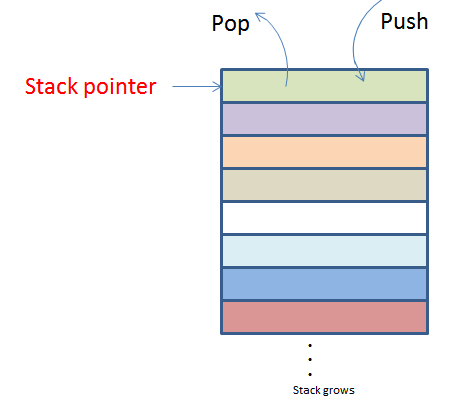
*Objective : To design a stack controller for a stack memory system this in turn uses the memory wizard of Altera to generate the required memory. Also, to observe its performance level.*

**Design Specification :**

* Inputs :
* 8 bit data input
* 1 bit push data
* 1 bit pop data
* Clock
* Output:
* 8 bit data output according to the push/ pop
* 1 bit overflow status
* Functional behavior :
* Control unit with two commands – push and pop
* Push should load the data immediately on top of the stack and the other data moves down one location.
* Pop should remove data from the top of the stack , and moves the data up by one position( if required).
* A structure which supports the modular expansion.
* Memory wizard has to be used to generate the required memory.

**Design Structure:**

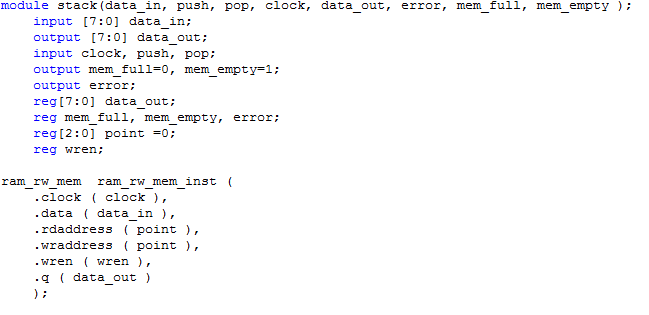
* Inputs :
* 8 bit data input
* 1 bit push data
* 1 bit pop data
* Clock
* Output:
* 8 bit data output according to the push/ pop
* 1 bit overflow status
* Functional behavior :
* Control unit with two commands – push and pop
* Push should load the data immediately on top of the stack and the other data moves down one location.
* Pop should remove data from the top of the stack , and moves the data up by one position( if required).
* A structure which supports the modular expansion.
* Memory wizard has to be used to generate the required memory.

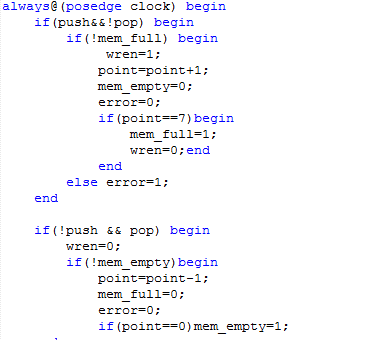


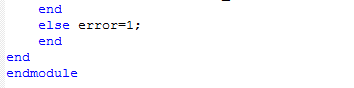
*Fig 7.1 : LIFO ( Last in first out ) STACK*

**Design Entry :**

* Verilog code :

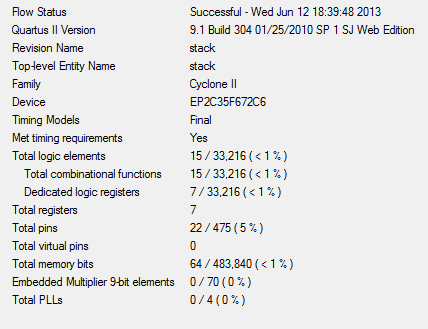






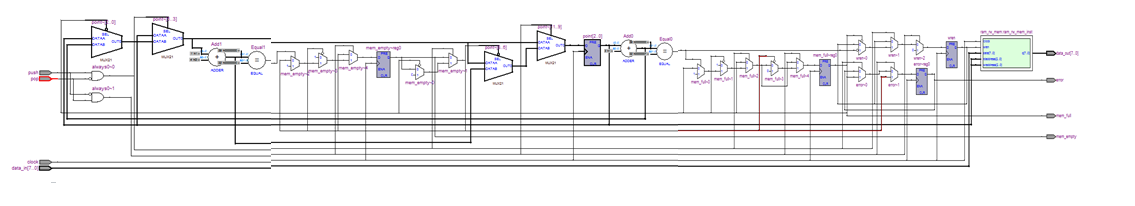
*Fig 7.2 : Verilog code*

* Compilation report :

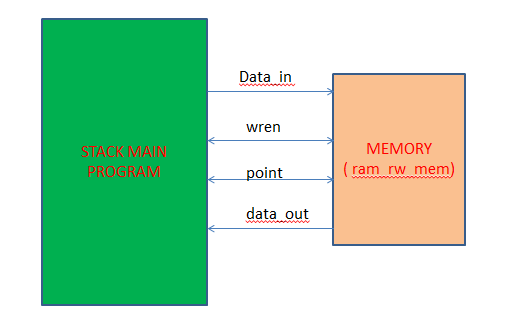


*Fig 7.3 L Compilation report*

* RTL :



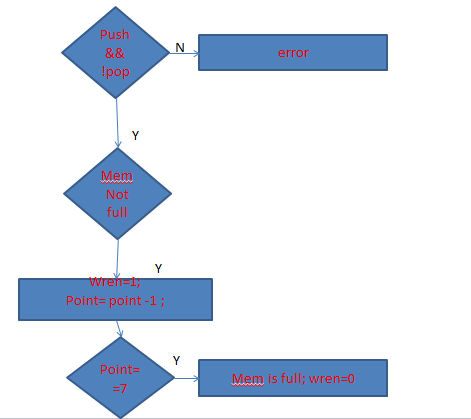
* Stack Block diagram :



*Fig 7. 5: Stack operation with memory*

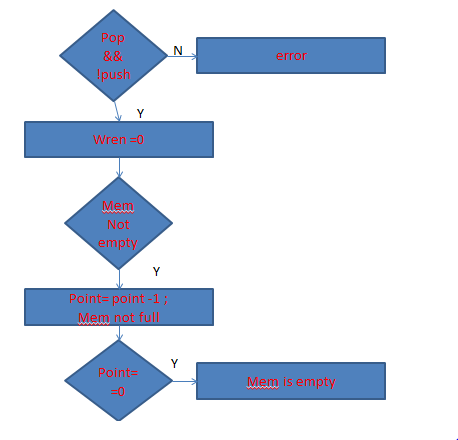
* Push and pop operation :

Push Operation :



*Fig 7. 6: push operation*

POP operation :



*Fig 7.7 : Pop operation*

**Design Verification :**

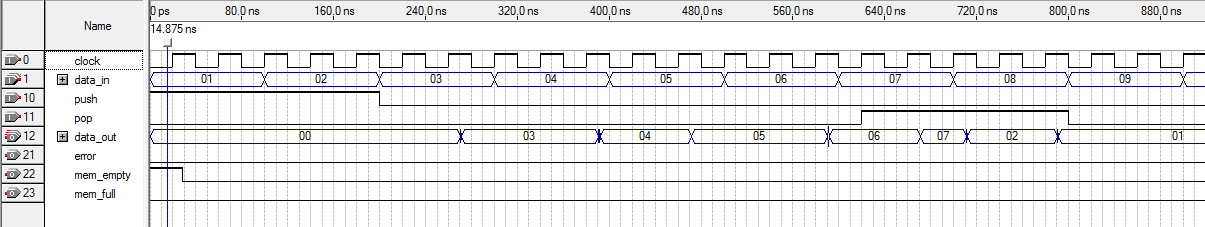
*Test Plan :*

The first test is done to push 01 and 02 and pop it out in order, given in fig 7.8.

The second test is done to push 02, 03 and pop them all out, given in fig 7.9.

Its observed that initially the 00 is stored in the memory , because memory is initially stored with values 00 , so it tends to take the default value first.

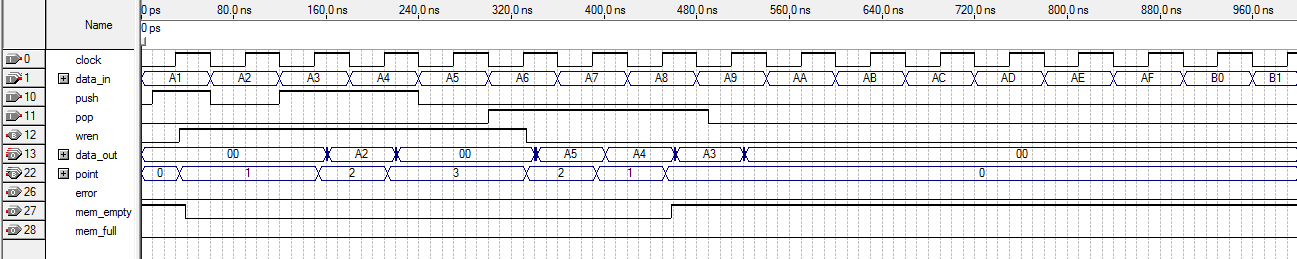
*Simulation results :*



02 and 01 pop out of the stack

01 and 02 push in the stack

*Fig 7.8 : test 1*



A5, A4, A3 pop out of the stack in three pop cycles

*iFig 7.9: Test 2*

**Result :**

* Performance of the system :

|  |  |
| --- | --- |
| Total CPU time | 3 secs |
| Total no. of pins | 5% |
| Failed paths | 0 |
| Clock period | 238 MHz |
| Actual time | 10.646 ns |
| Total logic units used | Less than 1% |

From the above table , its seen that the size this design uses is very less as the logic units totally used is less than 1%. With this it also provides a considerable fast process time of approximately 10 ns.

* Overall Result :

Hence using less space and having a considerable good speed, the stack operation is almost accomplished.