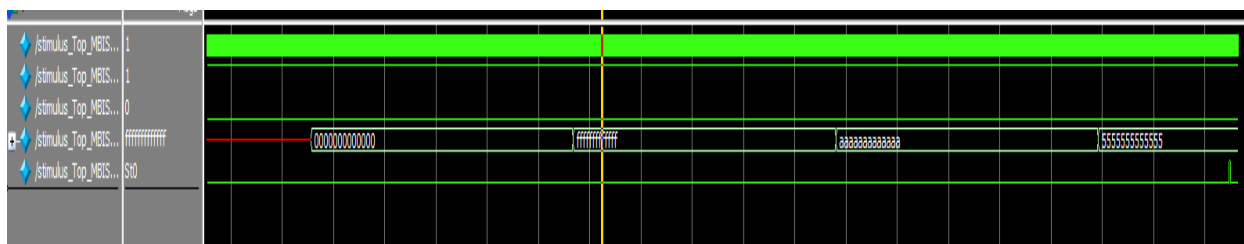


(1) Experimental Goal

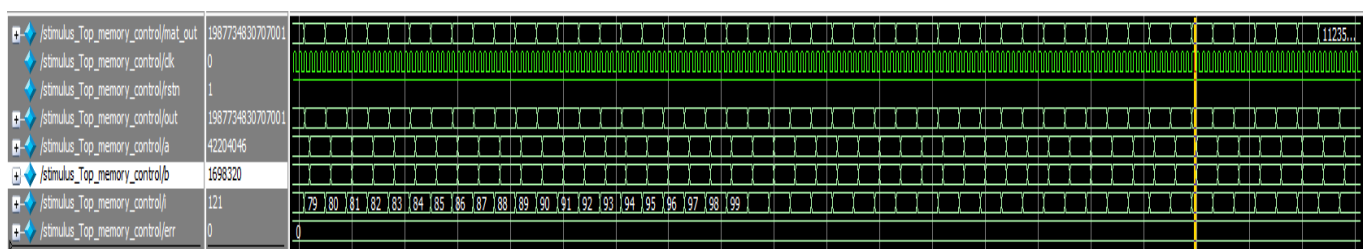
Experiment의 main content는 마치 S-ram 과 같은 구조의 Virtual memory 를 Test 하는 과정이다. 첫 번째 실습은 단순히 메모리에 input을 집어넣고 output을 확인하는 과정이고, 두 번째 process는 Multiplier를 apply하여, memory에 input을 인가하고, 이후 memory에서 output을 multiplier의 input으로 apply한 후, 해당 output value를 memory에 저장하고, 그 저장 값을 잘 알려진 matlab calced value와 비교하여 error를 구하는 것이다.

(2) Result

1. MBIST



2. Mem-control with 26x26 multiplier



(3) Discussion

사실 Synthesis를 하지 않으니 Discussion을 할 내용이 많지는 않다. 이번 실험에서는 MBIST를 설계한 후, Memory의 동작에 이상이 없는지 확인하였다. 또한 지난 RCA를 apply한 26*26bit multiplier 를 더하여 imbedded system을 간단하게 구축하였다.

첫 번째 실험에서는 row와 col이 정해져 있었다. 간단하게, counter가 126bit까지 count한 후, 7~10bit을 memory controlling bit으로 설정하면서 해당 과정을 쉽게 구할 수 있었다. 7bit을 mem-write, 8-9bit을 mem-Data_input_control, 10bit을 NCE로 구하면서 쉽게 이를 구현할 수 있었다. 또한 testbench file로 적절한 시간을 apply하면 결과 값을 얻을 수 있었다. 특히 input data또한 중요하다고 판단하였는데, 0000,1111,1010 , 0101 과 같은 서로 보수관계의 Data를 apply한 것이 적절한 판단이라 여겼다. 최종적으로는 위와 같은 적절한 결과 값을 얻을 수 있었다.

두 번째 실험은 Memory에 52bit input data를 저장하고, 이를 각각 26bit으로 나누어 a ,b 에 apply, 그리고 a,b를 multiplier에 putting in 하여 결과 값을 Memory에 저장한 후, 이를 하나씩 read하면서 theoretical value와 비교하고, 검증하는 과정이었다.

사실 이 과정 역시 proper한 timing과, read timing 등을 잘 조율하여 control하는 것이 관건인 실험이다. 그리고, 실제 메모리의 설계보다는 operation에 focusing한 실험이므로, 어렵지 않게 구현할 수 있었다. 즉, Synthesis의 의미가 없다는 것을 의미한다.

마지막으로 회로의 본질적으로 중요한 점인 Verification을 보자. Test bench는 내가 verification하는 design의 input을 만들어 주어야 하며, Reference model과 동일하게 operation하는지 output을 check하는 부분으로 이루어져 있다. Operating 시나리오를 기반으로 test case의 constraint를 만들고 그 안에서 random한 input을 apply한다. 이를 마지막 score board에서 Wave form으로 check하며, RTL algorithm이 동일하게 구현되었는지 error를 check한다 IP의 모든 부분이 connected 되었는지 check하기 위하여 coverage를 check하는 것도 필요하다. 이제 Verification 결과를 간단하게 확인하자. 우리는 Matlab을 통하여 input과 해당 value 간의 덧셈을 이용한 output 결과를 만들었다. 즉, 앞서 언급한 operating 시나리오를 위한 constraint를 matlab을 통하여 txt로 만든 것이다. 이를 testbench file에 apply하고, 앞에서 본 결과를 살펴보면 먼저 a와 b의 sum이 잘 되었는지 확인하기 위하여 시나리오 상의 output constraint와 실제 회로의 output과 비교를 진행했고, 이후 해당 값과 오차를 error로 출력하였다. 앞의 Testbench의 결과로 clk를 apply한 것의 유무에 관계없이, error가 0을 유지하는 것을 볼 수 있다. 이를 통하여 회로의 logic이 26bit multiplier, 그리고 메모리가 메모리로서의 역할을 잘 수행하고 있음을

알 수 있었다.

(4) Reference

Verilog HDL -joseph Cavanagh

Verilog HDL 이론 -한양대학교 전자전기공학부

Quora- CPU clock speed vs power vs area

ResearchGate – Circuit Area vs frequency

<https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=laonple&logNo=20926179193>

고려대학교 전자전기공학부 – VLSI design practice06