

HDL을 이용한 SDRAM Controller의 설계

김 용 국 (金 鏞 國), 오 경 옥 (吳 景 旭), 이 영 희 (李 英 姬)

단국대학교 전자컴퓨터공학부

전화 : (02) 709-2591 / 팩스 : (02) 709-2590

Design of SDRAM Controller in HDL

Yongkook Kim, Kyungwook Oh, Young-Hi Lee

School of Electronics and Computer Engineering Dankook University

E-mail : likepunk@dankook.ac.kr

Abstract

In this research we designed and synthesized an effective Synchronous DRAM controller for Interleaved Column Mode Access with VHDL. When target device was ALTERA CPLD MAX712 105 logic cells were used. The result of the simulation at 66MHz clock operation, the clock-to-output time t_{co} was 4.5ns and the SDRAM controller was in good working order.

I. 서론

컴퓨터등의 마이크로 프로세서 응용시스템의 주기억 장치로 많이 사용되었던 동적메모리(Dynamic RAM)는 집적도에서는 우수하지만 속도가 정적 메모리(Static RAM)에 비하여 빠르지 못하다는 단점이 있다. 이러한 문제를 해결하기 위하여 EDO(Extended Data Output) DRAM, Rambus DRAM 과같은 진보적인 DRAM이 계속해서 제안되었다. 그 중의 하나가 JEDEC(Joint Electron Device Engineering Council)에서 제안한 Synchronous DRAM 이고, 현재의 퍼스널 컴퓨터 등에서 대표적인 대용량 메인 메모리로 사용되고 있다. 이러한 SDRAM을 주 메모리로 사용하려고

할 때에 마이크로 프로세서 등의 버스 마스터(Bus Master)로부터의 메모리 액세스를 인터페이스 하기 위한 장치가 바로 SDRAM Controller이다.[1]

본 논문에서는 SDRAM Controller를 하드웨어 설계 언어인 VHDL(VHSIC Hardware Description Language)를 사용하여 설계하고 Synopsys를 사용하여 합성(Synthesis)하여 ALTERA사의 MAX+Plus 로 시뮬레이션 하였다. 그리고 최종 Target Device로는 CPLD(Complex Programmable Logic Device)[2]를 대상으로 하여 66MHz로 동작 할 수 있도록 Targeting 한다.

II. SDRAM Controller 의 구성 및 동작

대상이 되는 SDRAM으로 삼성전자의 KM432S2020B [3]를 선택하였다. 이것은 1Mbit Address 영역을 갖는 두 개의 뱅크로 되어 있고 32 Bit의 데이터 버스 폭을 가지며, 최대 125MHz까지 동작 가능한 고성능 SDRAM이다. 컨트롤러의 동작 주파수는 일반적인 고성능 마이크로프로세서 시스템의 동작주파수인 66MHz로 작동하도록 하고 Burst Access의 경우 1,4 의 Burst Length를 지원하도록 하였다. Refresh는 Auto Refresh를 사용하며 Refresh 주기를 계산하기 위

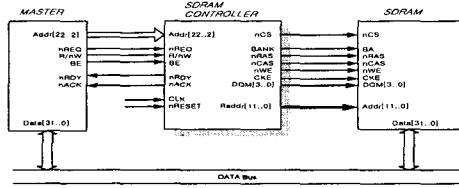


그림 2 SDRAM Controller의 Interface block diagram

한 카운터를 내장한다. 마스터와 SDRAM 사이의 인터페이스 신호 및 연결관계를 그림 1에 나타내었다. 이때 마스터는 마이크로프로세서 등의 메모리를 필요로 하는 장치이고 SDRAM Controller에게 메모리 읽기 쓰기 등의 요청을 하게 된다. 이에 SDRAM Controller는 SDRAM에 필요한 제어 신호를 보내어 데이터의 전송이 준비되면 다시 마스터에게 알려 상호간의 데이터 교류가 이루어지도록 한다.

메모리 액세스 방법들[4] 중 SDRAM은 Burst READ/WRITE를 기본으로 지원한다. 또한 두 개의 뱅크를 자체 내장하고 있기 때문에 각각의 뱅크를 교대로 액세스 하면서 입출력에 필요한 클럭 수를 줄일 수 있는 Interleaved Column Access Mode[5]를 사용하였다. 그림 2에서 보는 바와 같이(a)의 프로그램을 수행

```

for (i=0; i<16; i++)
    t = t + x[i] - y[i]
end for
    
```

(a)

- Row decode
- Column decode
- Precharge

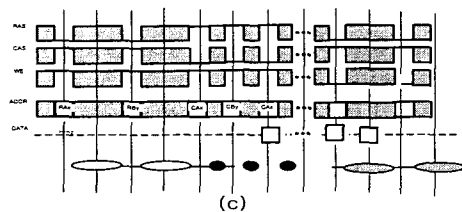
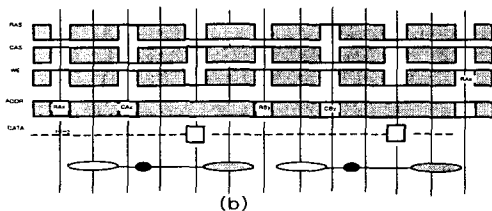


그림 1 (a) Sample program (b) Single read인 경우의 타이밍도 (c) Interleaved column access 인 경우의 타이밍도

하려면 (b)처럼 일반적인 READ/WRITE시에 매번 Precharge를 하는 경우에 총 $16 \times (6 \times 2) = 192$ 회의 클럭 Cycle이 필요하지만 (c)와 같이 Row address가 바뀌어 새로 Row Active를 해야 할 경우에만 Precharge를 해주게 되면 총 $3 + 3 + (16 \times 2) + 2 + 2 = 42$ 회의 클럭 Cycle이 소요하므로 클럭을 보다 효율적으로 사용할 수 있다.

III. SDRAM Controller 의 설계

그림 3은 SDRAM Controller의 블록 다이어그램이다. 이 Controller는 크게 어드레스 멀티플렉서(Address Multiplexor), FSM(Finite State Machine), 어드레스 비교기(Address Comparator), 그리고 리프레시 타이머(Refresh Timer)로 구성되어 있다.

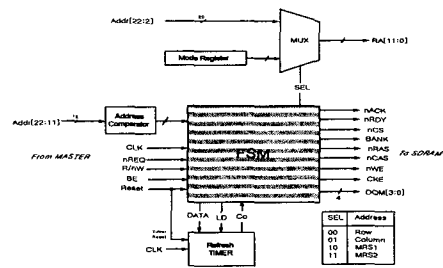


그림 3 SDRAM Controller의 Block diagram

(1) Address Multiplexor 부

마스터가 메모리 액세스를 요구할 때 필요한 메모리 영역의 어드레스를 Row Address와 Column Address로 나누어 차례대로 SDRAM에게 알려주는 역할을 한다. Row Active Timing에서 어드레스의 상위 12비트를 내보내고 Column Active Timing에서 어드레스의 하위 8비트를 내보낸다. 또한 MRS(Mode Register Set) 동작 시에 필요한 OP Code를 Burst Length에 따라서 선택적으로 출력할 수 있게 해준다.

(2) FSM(Finite State Machine) 부

Controller의 핵심이라고 할 수 있는 FSM은 그림 4와 같은 상태를 따라서 VHDL로 구성하였다. VHDL 표현 시에 FSM의 동작을 두 개의 프로세스로 나누어 첫 번째 프로세스는 입력 신호와 현재의 State에 따라서 다음 State를 결정 짓는 역할을 하고, 두 번째 프로세스는 각 State에 따른 출력 값을 정의한다.

를 실행한 뒤에 계속해서 Burst READ 동작을 하는 것을 확인할 수 있다. 그림 8은 일정 주기마다 Refresh Command Signal을 출력해주는 것을 확인하기 위해 Auto Refresh Command Simulation을 나타내었다. 약 15.5us 간격으로 Auto refresh command 가 출력되는 것을 볼 수 있다.

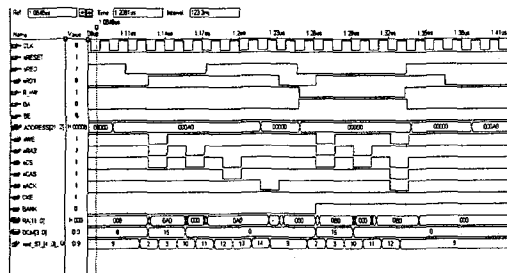


그림 6 Single READ/WRITE의 Simulation

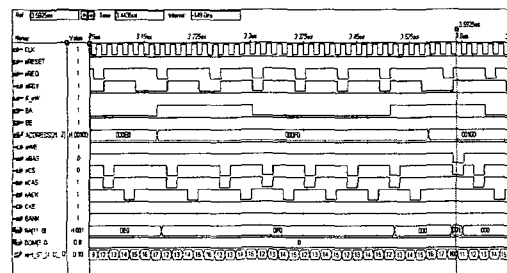


그림 8 Interleaved column access의 Simulation

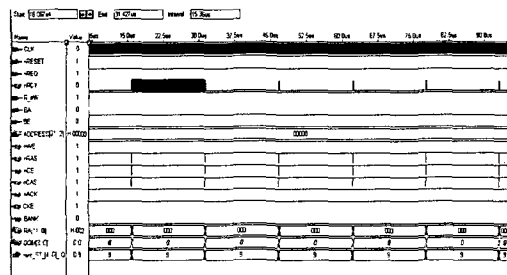


그림 7 Auto Refresh의 Simulation

V. 결론

본 논문은 Synchronous DRAM을 제어하기 위한 회로를 VHDL을 사용하여 설계하고 합성하였다. SDRAM의 특성을 이용하여 Burst READ/WRITE를 할 수 있도록 하였고, Interleaved Column Access를 적용할 수 있는 Controller Architecture를 구현하여 고속/대용량의 메모리 Access가 필요한 시스템의 메모리 컨트롤러로 사용할 수 있다.

Target Device로는 ALTERA사의 CPLD인 MAX7128을 사용하여 합성 결과 총 105개의 LC가 사용되었다. 66MHz Clock으로 Simulation한 결과 Clock-to-output time $t_{co}=4.5ns$ 로 각 동작이 만족함을 확인할 수 있었다.

이 SDRAM Controller는 VHDL로 구현하였기 때문에 설계의 변형이 용이하여 차후에 System LSI를 구성 할 때 Memory Interface Building Block으로 사용할 수 있으며 이 설계 방법을 Rambus DRAM등과 같은 또 다른 고속의 메모리를 제어하기 위한 설계에 적용할 수 있을 것이다.

참고문헌

- [1] Christian Green, "Analyzing and implementing SDRAM and SGRAM controllers", EDN Mag, Feb. 1998.
- [2] Altera Corporation, ALTERA DATA BOOK, 1998.
- [3] Samsung Semiconductors, <http://www.intl.samsungsemi.com/products/>. "Datasheets for synchronous DRAM chip KM432S2020BT", July 1998.
- [4] Steve Gumm, Carl T Dreher, "Unraveling the intricacies of dynamic RAMs", EDN Mag., March 1989.
- [5] A. Khare, P. R. Panda, N. D. Dutt, and A. Nicolau, "High level synthesis with SDRAMs and RAMBUS DRAMs.", Technical Report 98-28, ICS department, University of California, Irvine, May 1998.
- [6] 이영희, 오경욱, 김용국, "CPLD를 이용한 SDRAM Controller의 설계", 단국대학교 논문집 제 34집, 1999.