

ActiveX 컨트롤을 이용한 단순화된 웹 기반 디지털 논리회로 시뮬레이터

김 동 식* 한 희 진* 서 삼 준** 김 희 숙***

*순천향대학교 공과대학 정보기술공학부

**안양대학교 공과대학 전기전자공학부

***아산정보기능대학 멀티미디어학과

(2003. 1. 3. 접수)

A Simplified Web-based Simulator for Digital Logic Circuits Using ActiveX Control

Dongsik Kim*, Heejin Han*, Samjun Seo**, Heesook Kim***

**Division of Information Technology Engineering, Soonchunhyang University*

***Department of Electrical and Electronic Engineering, Anyang University*

****Department of Multimedia, Asan Information Polytechnic College*

(Received January 3, 2003)

국문요약

본 논문에서는 디지털논리회로의 여러 가지 원리를 이해하고 확인해보기 위해 비주얼베이직의 ActiveX 컨트롤을 이용하여 단순화된 웹 기반 디지털 시뮬레이터를 개발하였다. 개발된 디지털시뮬레이터는 디지털 논리회로 학습에 있어 필수적인 기능만이 구현되었으며, 학습자가 스스로 구성한 디지털 회로를 직접 웹상에서 시뮬레이션 해 봄으로써 디지털시스템에 대한 설계 및 해석이 가능하도록 제작되었다. 개발된 디지털 시뮬레이터가 디지털 논리회로에 대한 멀티미디어 콘텐츠와 함께 웹에 제공된다면 교육현장에서 교육보조도구로써 활용이 가능하고 또한 학습효율의 극대화를 이룰 수 있으리라 생각된다. 제안된 시뮬레이터의 유효성을 입증하기 위해 몇 가지 디지털 논리회로에 대한 시뮬레이션 결과를 제시하였다.

Abstract

This paper presents a simplified web-based simulator for digital logic circuits with which several important principles related to digital logic circuits can be understood and

confirmed. The proposed simulator is implemented to have several simplified functions which are essential to the learning process of digital logic circuits. The learners by themselves simulate several digital logic circuits on the web under specific input conditions and the design/analysis of digital logic circuits can be available.

The proposed simulator, combined with multimedia contents, can be used as an auxiliary educational tool and enhance the improved learning efficiency. The results of this paper can be widely used to improve the efficiency of web-based educations in the cyber space. Several simulation results are illustrated as examples to show the validity of the proposed web-based simulator.

1. 서론

일반적으로 전기, 전자 및 디지털회로 등을 설계할 경우에는 회로특성을 평가할 수 있는 정확한 방법이 필수적으로 필요하다. 이러한 회로를 하드웨어적으로 직접 제작하여 실험을 통해 여러 가지 특성 등을 확인해 볼 수 있지만, 이렇게 할 경우 회로구성 및 특성해석에 많은 시간과 계측장비 및 경비가 필요하기 때문에 실제로 회로를 제작하기 전에 컴퓨터를 이용하여 계산하고, 측정, 평가하는 과정인 컴퓨터 시뮬레이션을 거쳐야만 한다. 현재 이러한 시뮬레이션을 위한 공학용 소프트웨어에는 PSpice, ORCAD, MATLAB 등 전문적인 프로그램이 사용되고 있다. 그러나 이러한 소프트웨어는 가격이 고가이므로 일반적으로 많은 사람들이 사용하기는 경제적으로 매우 부담스러운 실정이다. 그리고 상용의 소프트웨어에는 매우 광범위한 기능들이 총망라되어 있으나 실제 교육현장에서 사용하는 기능은 비교적 간단한 형태의 디지털 논리회로의 입출력 특성파악과 관련된 일부의 기능만을 이용하고 있으므로 고가의 소프트웨어의 사용 효율성은 매우 낮은 형편이다. 따라서 실제 교육현장에서 교육 보조도구로써 사용하기에 적합한 기본적인 기능을 가진 단순화된 저가의 디지털 시뮬레이터가 오히려 교육효율을 높이는 데 더 효과적일 수 있다.

본 논문에서는 이러한 측면에서 디지털논리회로를 이해하는데 필수적인 기능만을 가진 단순화된 디지털 시뮬레이터를 개발하여 인터넷이 가능한

컴퓨터에서는 누구나 쉽게 사용할 수 있도록 하였다. 개발된 디지털시뮬레이터는 ActiveX 컨트롤을 이용하여 구현하였으며 학습자가 스스로 구성된 디지털 회로를 직접 웹상에서 시뮬레이션해 봄으로써 디지털시스템에 대한 여러 가지 중요한 개념이나 원리를 쉽게 이해할 수 있도록 설계되었으며 교육현장에서도 교육보조도구로써 활용한다면 학습효율의 극대화를 이룰 수 있으리라 생각된다.

2. 디지털 시뮬레이터의 전체 구성

2.1 프로그램의 전체 구성도 및 특성

본 논문에서 개발된 디지털 시뮬레이터의 전체적인 구성도는 그림 1에 도시하였으며 이는 시뮬레이터의 프로그램 순서도인 동시에 전체구성에 관한 개략적인 그림이다.

먼저 제공된 Library로부터 Symbol로서 정의된 소자를 Layout Grid에 불러온다. 이는 간단히 마우스의 클릭 이벤트를 이용하여 진행된다. 이때 모든 소자는 각각의 설명과 소자의 Symbol을 이미지로 확인할 수 있도록 제작된 독립된 창인 Part Browser에서 선택된다. 회로에 사용될 소자의 Symbol들이 모두 선택되면 Grid에서 회로를 편집한다. 이때 Symbol 소자들은 마우스의 Move 이벤트로 이동되며 DEL키를 이용한 삭제도 가능하다. 다음으로 Symbol 들간의 회로 연결을 하는데 이는 마우스의 클릭 이벤트와 Move 이

벤트를 사용하여 사용자가 원하는 경로로 그려질 수 있다. 따라서 디지털 논리회로 구성이 간단한 마우스 조작을 통해 매우 간결하고 용이하게 이루어진다. 구성된 논리회로의 시뮬레이션은 회로의 구성시에 함께 생성된 변수들의 연산으로 이루어진다. 각각의 소자들은 입력 펄스의 연산을 독립적으로 수행하는데, 이로 인해 다중시뮬레이션이 가능하다. 이는 회로를 여러 개 작성하여 시뮬레이션 하는 번거로움을 없앨 수 있을 뿐만 아니라 두 개 이상의 회로를 비교 분석할 수 있다는 장점이 있다.

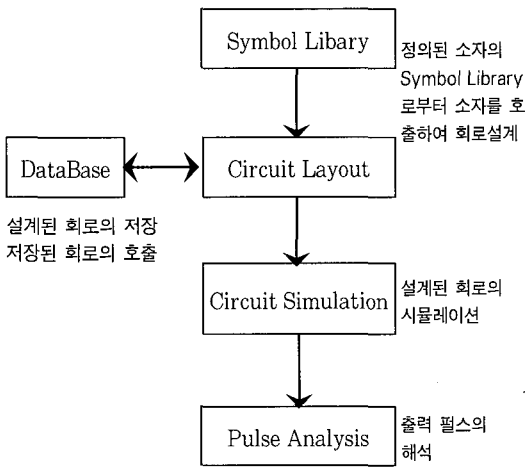


그림 1. 시뮬레이터의 전체 구성도

시뮬레이션이 끝나면 다음이 회로의 해석인데 역시 독립된 창에 사용자가 원하는 부분의 모든 출력 펄스를 나타내도록 하였다. 이는 사용자가 시뮬레이션 이전 단계에서 회로에서 출력 파형을 보기 원하는 부분에 Marker를 생성하도록 하여 원하는 어느 부분이라도 직접 파형을 확인할 수 있다. 또한 논리 회로 구성시 생성되는 여러 변수들을 DataBase에 저장할 수 있도록 하였다. 이는 변수의 연산으로 시뮬레이션이 되기 때문에 가능한데 회로의 저장은 이 변수들을 Data Base에 저장하는 것이다. 또한 저장된 회로의 Open은 DataBase에 저장된 변수들을 불러와 Layout Grid에 다시 그리는 함수를 작성하여 구현하였다.

2.2 초기 실행 창 및 버튼의 구성

ActiveX 컨트롤은 컨테이너 프로그램이 있어야만 한다. 본 논문은 웹 시뮬레이터 개발을 목적으로 했기 때문에 Explorer를 컨테이너로 하였다. 따라서 시뮬레이터의 실행 창은 크게 컨트롤을 담당하는 웹 브라우저 부분과 ActiveX 컨트롤 부분으로 나뉘어진다. ActiveX 컨트롤은 시뮬레이터의 모든 기능이 버튼화 되어있는 도구상자와 회로를 그릴 수 있는 Layout Grid의 두 부분으로 나뉘어진다. 이는 사용자가 별도의 설명 없이도 쉽게 시뮬레이터를 사용할 수 있도록 하기 위함이다. 그림 2에 시뮬레이터의 실행 창을 도시하였다.

도구상자의 버튼 설명은 다음과 같다. 버튼은 각각의 이미지만 보아도 쉽게 용도를 알 수 있도록 제작하였고, 또한 각각 ToolTipText(말풍선) 설정을 하여 마우스 포인터를 버튼위에 위치시켜 간단한 설명을 볼 수 있게 하였다.

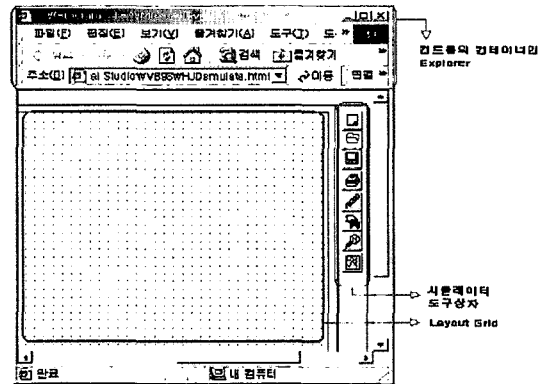


그림 2. 시뮬레이터의 실행 창 설명

2.3 Symbol 소자의 표현

시뮬레이터의 소자 구성은 그림 3과 같으며, 이 소자들의 Symbol 표현은 모두 함수로 처리되는데, 다음과 같은 순서로 그려진다. Library창인 Part Browser를 열면 소자를 선택할 수 있다. 소자를 선택한 후 마우스로 Layout Grid를 클릭하면 원하는 위치에 Symbol 소자를 그릴 수 있다. (그림 4 참조)

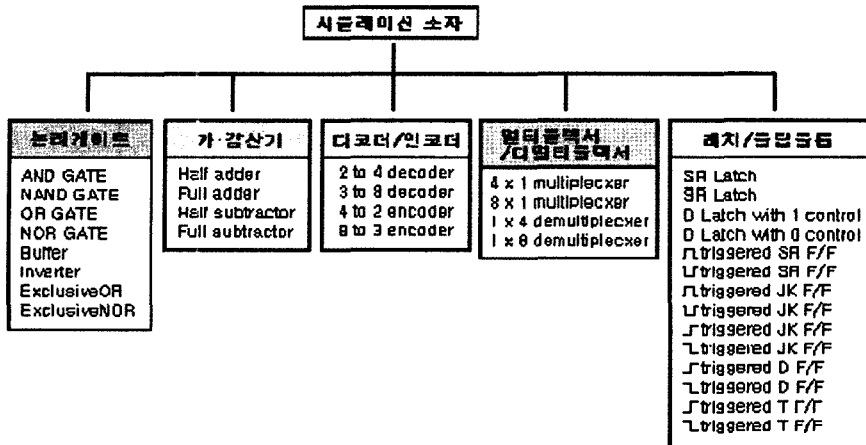


그림 3. 시뮬레이션 소자의 구성도

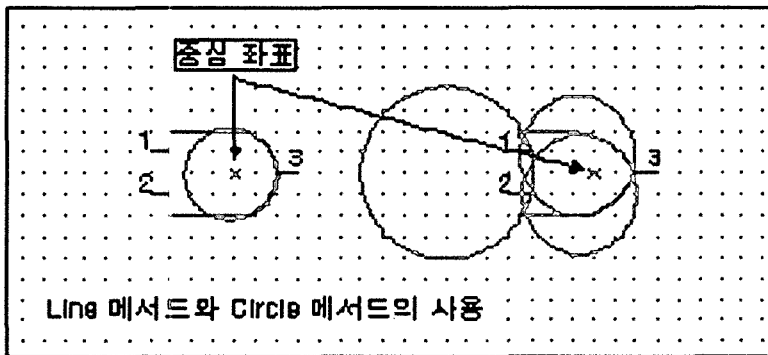


그림 4. 소자 표현의 원리, AND와 OR

Symbol을 그리는 첫 번째 핵심은 중심좌표이다. 이는 Layout Grid상의 마우스 클릭 이벤트의 발생 위치를 찾는 것이다. Layout Grid를 클릭하면 클릭이벤트가 발생하는데, 비주얼 베이직에서는 이벤트가 발생된 지점의 X좌표와 Y좌표를 반환한다. 이 좌표를 Symbol 표현의 중심좌표로 사용하는 것이다. Part Browser를 통해 소자의 선택이 이루어진 상태에서 Layout Grid를 클릭하면 PictureBox 컨트롤(이 컨트롤이 바로 Layout Grid이다.)에서 반환되는 중심좌표 X와 Y로부터 정해진 좌표에 직선부분은 Line 메서드를, 곡선부분은 Circle 메서드를 이용해서 원하는 소자의 Symbol을 그리는 것이다.

2.4 소자간의 연결 관계

회로를 구성하는 가장 중요한 부분이 바로 이 소자간의 연결 관계를 구현하는 것이다. 그림 5와 같이 Layout Grid상의 소자들의 외부적인 연결은 간단히 Line 메서드를 구현하여 가능하지만 컴파일을 위한 소자들의 내부적인 연결관계는 내부적인 알고리즘을 작성해야 한다. 알고리즘은 다음과 같은 원리를 적용해서 구현하였다.

- ① 마우스 클릭 이벤트 좌표와 소자의 단자 좌표가 일치할 때 연결이 시작된다.
- ② 연결 시작은 소스인 Pulse, Preset, Clear, Gate의 Output 단자만이 된다. (Start

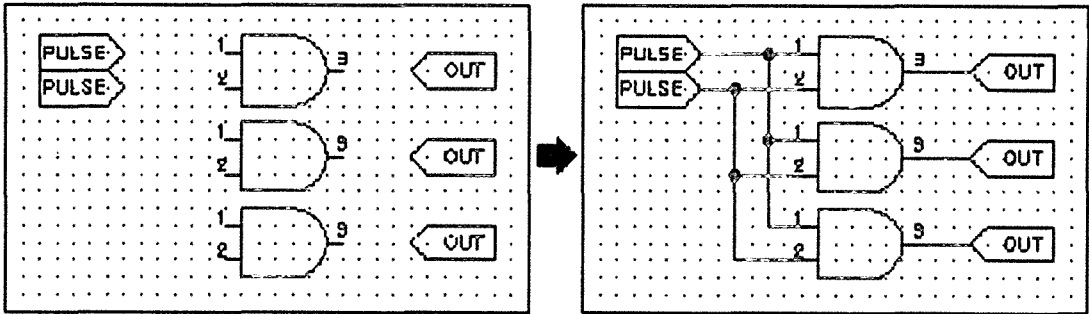


그림 5. Symbol 소자의 외부적인 연결 모습

source)

- ③ 연속된 선의 연결은 바로 앞 배열 Line변수의 소스가 현재 배열 Line변수의 소스이다.
- ④ 연결 종료는 Output, Gate의 Input단자만이 된다. (End source)

이 원리는 소자연결의 기본으로 알고리즘의 핵심이 되며 그림 6에 이 원리를 도식적으로 설명하였다.

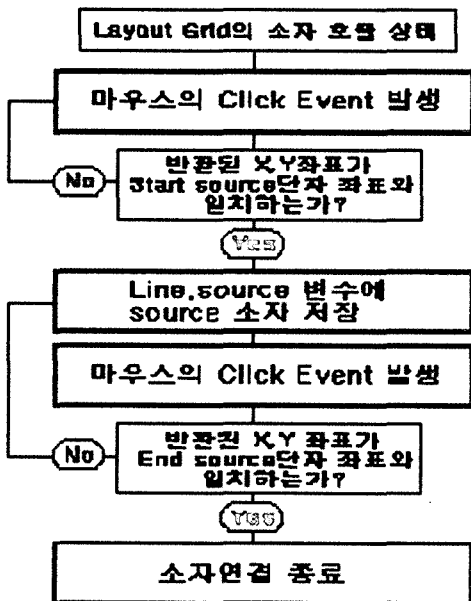


그림 6. 소자연결 구성도

2.5 논리회로 구성 및 파라미터 설정

그림 7에서처럼 소자 찾기 버튼을 누르면 Part Browser 창이 뜨는데 여기서 사용자가 원하는 소자를 선택하면 된다. 이 상태가 Drawing 상태이다. Drawing 상태에서 마우스로 Layout Grid를 클릭하면 클릭 된 좌표에 선택된 소자가 그려진다. 이렇게 원하는 소자들을 Part Browser에서 선택해서 Layout Grid에 불러오면 된다. 그림 8에서처럼 소자의 선택이 끝나면 Drawing 상태는 끝나게 된다.

소자를 호출한 다음 소자들을 원하는 회로의 구성을 위해 연결을 해야 되는데 이때는 연결버튼인 을 누르면 된다. 이 상태는 Connecting 상태

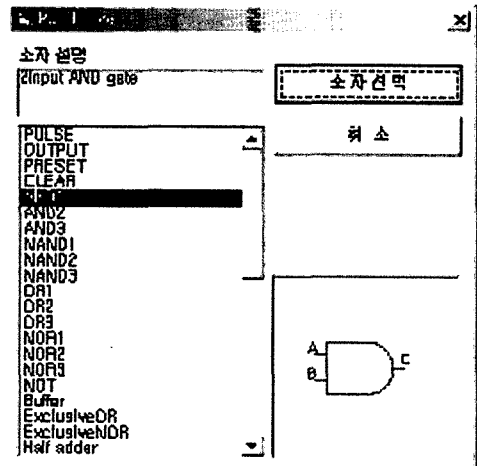


그림 7. Part Browser의 모습

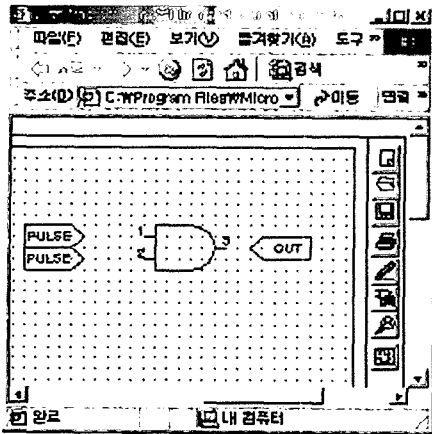


그림 8. 소자들의 호출

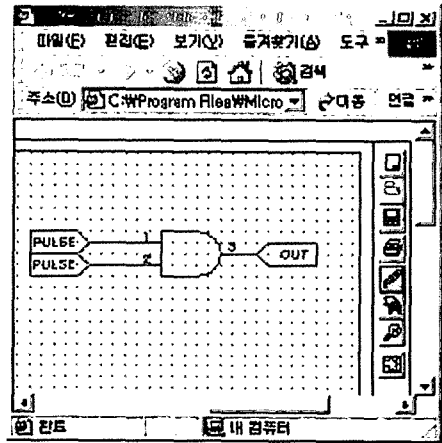


그림 10. 회로의 완성

로 Layout Grid상의 마우스아이콘이 연필모양으로 바뀌게 된다. 이 상태에서 마우스로 소자의 단자를 클릭하면 연결이 시작되고, 연결이 끝난 후 마우스의 오른쪽 버튼을 누르면 Connecting 상태는 종료된다. 이렇게 구성된 회로는 외적인 연결은 물론 내부적인 연결도 이루어지게 된다(그림 9-10 참조).

디지털 시뮬레이터에서 설정해줘야 하는 파라미터는 한 가지 있다. 바로 입력 Pulse를 정해주는 것인데 이 역시 간단히 설정할 수 있다. 펄스를 입력하는 방법은 회로를 구성하고 나서 펄스 소자를 더블클릭하면 그림 11과 같은 창이 뜬다. 그러면 선택된 펄스의 값을 입력할 수 있는데 그 방법

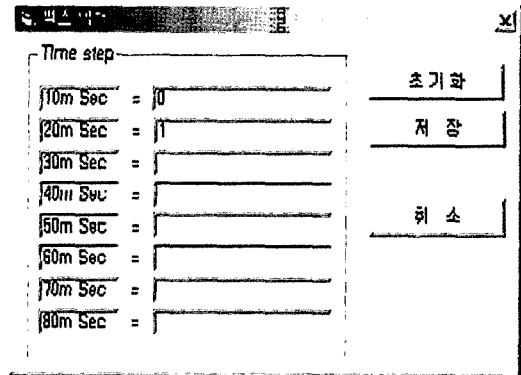


그림 11. 펄스 입력 창

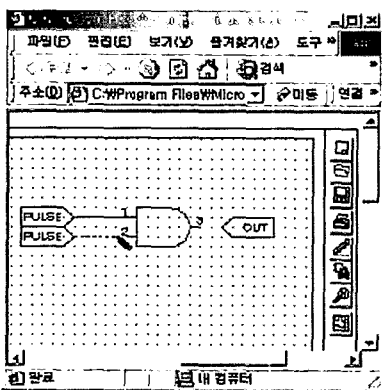


그림 9. 소자의 연결

은 시간별로 값을 정해 주는 것이다. Time Step은 그림 11에 나타나 있는 것과 같이 10 msec 단위로 되어있다. 이는 실제 TTL 소자들의 응답 시간이 μ sec 또는 m_sec 이기 때문에 임의로 Time Step을 정해준 것이다.

2.6 컴파일 알고리즘

시뮬레이터의 컴파일은 회로의 구성이 완료된 상태에서 시뮬레이션 버튼을 누르면 실행된다. 회로가 제작될 때는 다음과 같은 변수들이 생성된다.

UNIT : 소자 개수를 표시

Gateclass.Class : 소자의 종류

Gateclass.compileDon : 소자의 컴파일 유무 표시

Gateclass.Exist : 소자의 존재 유무 표시

이는 컴파일 알고리즘에서 쓰이는 변수들이다.

컴파일이 시작되면 먼저 While문에서 UNIT의 수만큼 컴파일을 실행했는지 판단한다. 이는 모든 소자가 컴파일을 완료할 때까지 유한루프(loop)를 돌리기 위함이다.

다음으로 For 문을 써서 모든 소자를 검색한다.

동시에 If 문을 사용하여 검색된 소자의 컴파일 유무를 판단해서 컴파일이 안된 소자라면 그 소자의 종류를 파악해서 소자의 컴파일을 수행한다.

이런 방식으로 For문을 사용하여 모든 소자의 컴파일 완료여부를 확인하면 While 문의 유한루프를 빠져나올 수 있다. 이것이 간단한 시뮬레이터의 컴파일 알고리즘이며 그림 12에 이러한 프로그램의 간단한 알고리즘의 구성을 도시하였다.

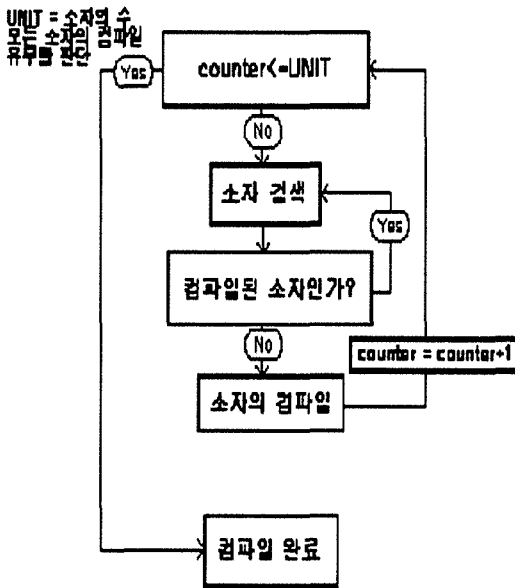


그림 12. 컴파일 알고리즘

2.7 시뮬레이션 결과 창 및 파형 보기

컴파일이 완료되면 시뮬레이션 결과창이 뜨게 된다. 출력창은 Time 축과 Pulse 축이 있으며,

Time 축은 기본단위가 msec 로 사용자가 별도의 시간설정을 할 필요가 없도록 임의로 정하였다.

또한 펄스 축은 디지털 신호인 Low와 High가 출력되는데 그 값은 시뮬레이션의 결과에 따라 나타나게 된다. 출력되는 펄스는 시뮬레이션 전에 사용자가 배치한 marker의 수에 따라 달라진다.

그림 13 은 2 to 4 디코더의 입력펄스와 출력펄스를 보여준다.

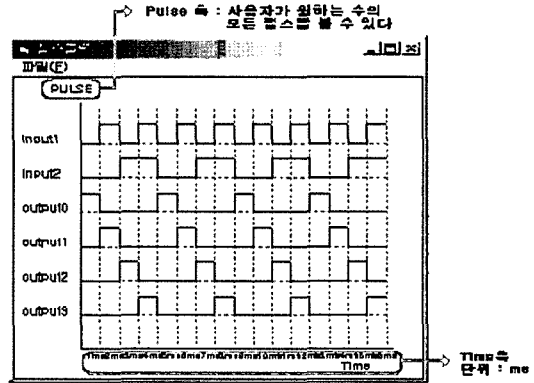


그림 13. 출력 창의 모습

2.8 회로의 저장 및 출력

회로의 구성이 끝나면 시뮬레이션 또는 회로저장이 가능하다. 시뮬레이션이 이루어지는 과정은 앞 절에서 설명한대로 변수의 연산으로 이루어지며 회로의 저장 역시 이 변수들을 이용하여 DataBase에 저장하는 것이다. 본 논문에서는 DB를 Access를 사용하였다. 출력은 Visual Basic에서 제공하는 CommonDialog 컨트롤을 사용하였다. CommonDialog 컨트롤은 파일 열기, 파일 저장, 인쇄 옵션 설정, 색 선택, 글꼴 선택과 같은 작업에 대한 Windows의 표준 대화상자 집합을 제공해주는 컨트롤이다. 본 시뮬레이터에서는 이것을 이용해서 회로 및 출력 파형의 인쇄를 가능하도록 설계 제작하였다.

3. 기본 논리회로에 대한 시뮬레이션 예

3.1 비동기식 카운터 회로에 대한 시뮬레이션

2진 계수의 동작을 하는 T 플립플롭을 사용하여 아래 그림 14와 같이 연결하게 되면 4비트 비동기식 카운터가 된다. 그림에서 CLR 단자를 모두 접속하여 Low 신호를 가하면 4개의 T 플립플롭의 출력은 0이 된다. 이 상태에서 첫 번째 단의 클럭입력에 클럭펄스가 공급되어 클럭펄스가 1에서 0으로 변환되는 순간 첫 번째 단의 출력은 0에서 1로 바뀌고 두 번째 클럭펄스가 공급될 때까지 그 상태를 유지하게 된다. 두 번째 클럭펄스가 공급되면, 첫 번째 단의 출력은 1에서 0으로 바뀌며 따라서 두 번째 단의 출력을 0에서 1로 바꾸게 된다. 이와 같이 클럭펄스가 공급될 때마다 계속하여 카운터를 하게 된다.

■ 회로해석 및 시뮬레이션 결과

그림 14 에서와 같이 4개의 JK 플립플롭을 J와 K에 계속 1 상태를 유지시켜서 항상 토글(보수)상태가 되도록 하여 클럭펄스의 입력 및 각 플립플롭의 출력에 따라 순차적으로 업 카운트가 되는 회로이다.

3.2 동기식 카운터 회로에 대한 시뮬레이션

■ 회로해석

동기식 카운터회로 및 그에 대한 시뮬레이션 결과를 그림 16과 17에 각각 도시하였다.

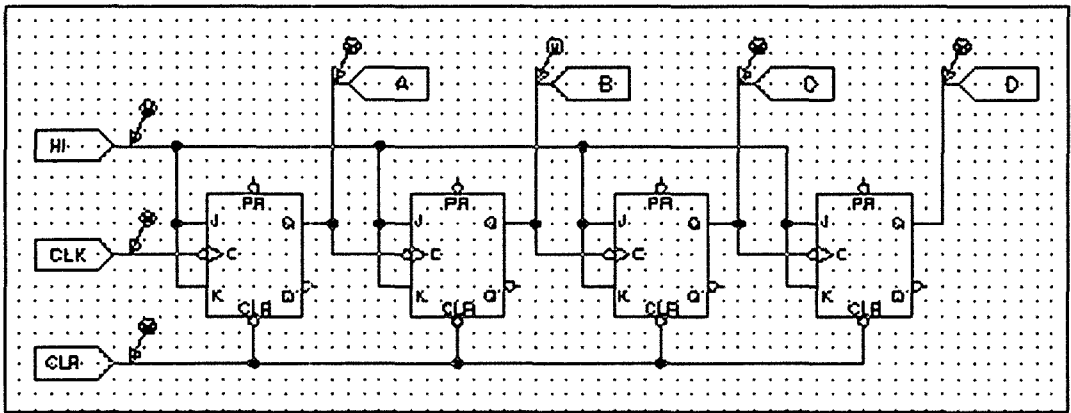


그림 14. 비동기식 카운터 회로

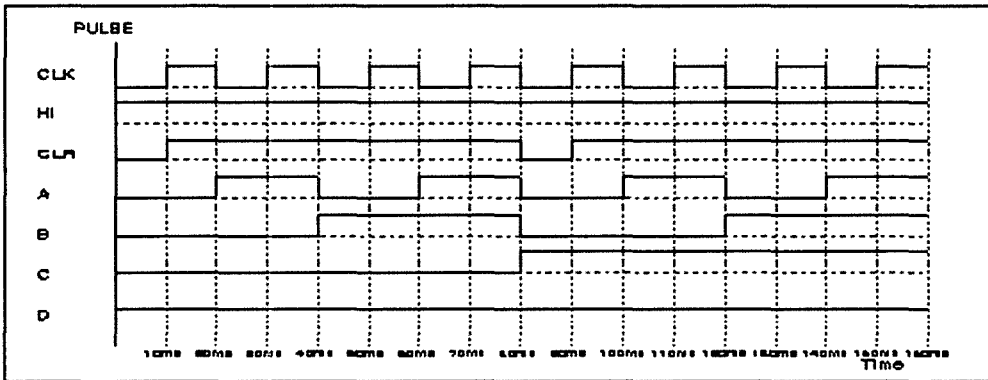


그림 15. 비동기식 카운터의 시뮬레이션 결과

■ 시뮬레이션 결과

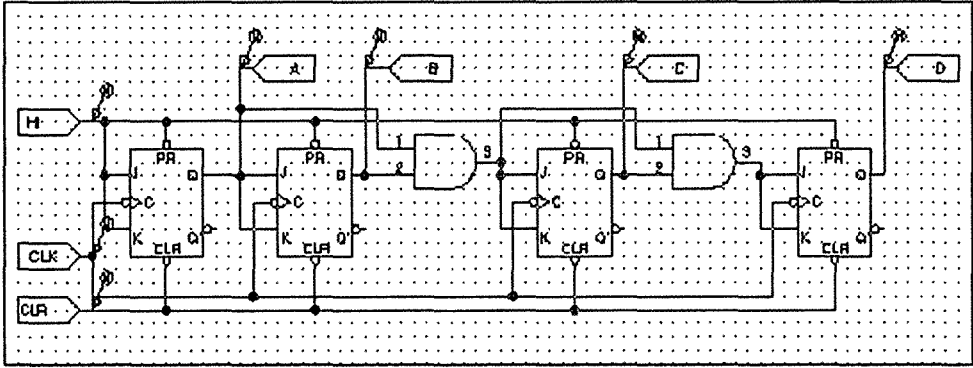


그림 16. 동기식 카운터

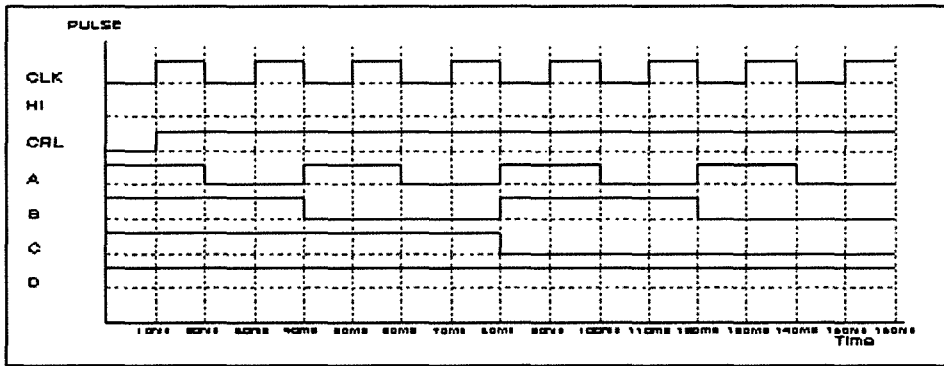


그림 17. 동기식 카운터의 시뮬레이션 결과

4. 결론

본 논문에서 개발된 디지털 시뮬레이터는 ActiveX 컨트롤로 제작되었으며, ActiveX 컨트롤은 윈도우 시스템 리소스(Resources)로서 윈도우에서 실행되는 모든 프로그램이 서로의 리소스를 공유할 수 있을 뿐만 아니라 Client/Server 환경에서도 액세스 할 수 있다. 즉, 시뮬레이터의 컨테이너로 웹 브라우저를 선택해서 사용자가 굳이 시뮬레이터를 설치하지 않아도 인터넷을 통하여 쉽게 시뮬레이터를 사용할 수 있도록 한 것이다. 이는 일반적으로 사용되어지는 시뮬레이션 툴들의 사용의 제한성을 없앤 것이다. 개발된 웹 기반 디지털 시뮬레이터는 다음과 같은 특징을 가지고 있다.

- ① 간단한 마우스 이벤트에 의해 동작하며, 두 개 이상의 논리회로에 대해 동시에 입력력관계를 제시해 주는 다중 시뮬레이션이 가능하다.
- ② 사용자가 원하는 모든 논리회로의 구성이 가능하며 사용자 편의를 고려하여 화면설계를 하였기 때문에 단시간 내에 시뮬레이터 사용법 습득이 가능하다.

개발된 시뮬레이터는 학습자가 원하는 논리회로를 구성하고 임의의 입력값을 인가하여 웹상에서 시뮬레이션을 수행한 후 출력 파형을 확인할 수 있도록 설계되었다. 이는 수업시간에 학습한 내용을 학습자가 직접 확인하고 활용할 수 있어

학습효과를 극대화 할 수 있다는 장점을 가진다.

위와 같은 장점으로 인해 공학교육에 보조도구로써 디지털 시뮬레이터를 활용한다면 웹 기반 교육에 있어 새로운 접근방법중의 하나가 될 것으로 생각된다.

향후 연구과제로는 결선과정의 유연성을 가지도록 시뮬레이터를 구현하는 것이 필요하다. 또한 학습자가 시뮬레이션 한 결과를 서버측의 데이터 베이스에 저장한 후 결과보고서를 작성하여 제출할 수 있도록 하는 연구를 진행할 예정이다.

[참 고 문 헌]

- [1] 김 동식, "인터넷을 이용한 효율적인 공학 실험실습교육을 이용한 가상실험의 개발," 공학교육과 기술 논문지, Vol. 3, No.2,

2000.11

- [2] Jerry Ablan 외 11명저, 류광 역, *Visual Basic 6 Web*, 정보문화사, 2000
- [3] 최평, 조용범외 3인, PSpice 기초와 활용, 복두출판사, 1997
- [4] Siegel M. & KirKley, S., *Moving Toward the Digital Learning Environment :The Future of Web-Based Instruction*, Education Tech. Publications, 1994.
- [5] Dongsik Kim et al, "Practical Implementation of A Web-based Virtual Laboratory in the Area of Electrical Engineering," IASTED International Conference on Computers Advanced Technology in Education, 2001.

감사의 글

본 연구과제는 2003학년도 순천향대학교 산업기술연구소 학술연구조성비 일반연구과제로 지원을 받아 수행하였음.