

# USB 인터페이스를 이용한 LCD 구동회로의 FPGA 설계

이 승 호<sup>†</sup> · 이 주 현<sup>††</sup>

## 요 약

본 논문에서는 USB(Universal Serial Bus) 인터페이스를 이용하여 Gray Mode Graphic STN LCD를 구동하는 구동회로를 제안한다. 제안된 구동회로는 사용자가 PC상에서 작업한 이미지 데이터를 USB 인터페이스를 이용하여 고속으로 LCD로 전송할 수가 있다. 따라서 기존의 방식과는 달리 마이크로프로세서를 사용하지 않기 때문에 사용자가 손쉽게 LCD를 구동할 수 있는 장점이 있다. 제안한 LCD 구동회로 부분은 VHDL (VHSIC Hardware Description Language)로 설계하여 시뮬레이션은 수행하고 ALTERA 사의 EPF10K10TC144-3 FPGA를 사용하여 구현하였다. USB 인터페이스 부분은 MS-Visual C++ 6.0으로 프로그래밍하였다. 한편, 제안한 구동회로를 테스트 보드를 구성한 후에 하드웨어 동작 검증을 수행하여 그 효율성을 입증하였다. 본 논문에서 제안한 구동회로를 현재 시판중인 마이크로프로세서를 사용하는 타 업체의 구동회로와 비교한 결과는 구동회로의 작동 편의성, 제작시 소요되는 비용 등에서 우수함을 나타내었다.

## FPGA Design of LCD Drive Circuit using USB Interface

Seung-Ho Lee<sup>†</sup> · Joo-Hyun Lee<sup>††</sup>

### ABSTRACT

This paper describes a FPGA Design of Gray Mode Graphic STN LCD drive circuit using USB interface. The drive circuit using USB interface can highly transfer image data created under PC to LCD. Hence, the LCD drive circuit doesn't use microprocessor for the convenience of users. The proposed LCD drive circuit part have been verified by simulation and by ALTERA EPF10K10TC144-3 FPGA implementation in VHDL. The USB interface part have been programmed in MS-Visual C++ 6.0. The validity and efficiency of the proposed LCD drive circuit have been verified by test board. After comparing this LCD drive circuit to specify it was verified that the developed LCD drive circuit showed good performances, such as convenience of users, low cost.

키워드 : USB 인터페이스(USB Interface), LCD 구동회로(LCD drive circuit), 프로토콜(protocol), VHDL, FPGA

### 1. 서 론

오늘날 LCD(Liquid Crystal Display)는 그동안 영상표시 소자의 대표주자인 CRT(Cathode Ray Tube)을 대체할 수 있는 또 하나의 다른 표시소자 이기보다는 21세기의 고도로 진보된 정보화 사회에서 요구하는 새로운 정보제품을 창출하는 필수적인 표시소자로 인식되고 있다[1-3]. 또한 LCD는 소비전력이 적게 들고 구동전압이 낮으며 경박 단소화가 가능한 장점 때문에 1973년 포켓용 전자계산기와 손목시계에 채용되어 양산된 이래로 그 응용범위가 날로 확대되어 왔다. 현재는 LCD가 정보표시소자로서 컴퓨터 게임기, 전자측정장비, 개인용 컴퓨터 등의 화면으로 널리 사용되고 있다. 한편, 전자기기의 정보표시기, 개인 휴대 단말기 등에 사용되는 Gray Mode Graphic STN LCD는 이동 통신 수요의 급성장과 각

종 전자기기의 정보 표시기 장착 추세에 따라 중저가의 보급형 디스플레이 시장에서 폭발적 성장이 예상되고 있다. 그런데 Gray Mode Graphic STN LCD에 숫자, 문자, 도형, 아이콘, 애니메이션 등을 디스플레이 하기 위해서는 Gray Mode Graphic STN LCD를 구동하기 위한 구동회로가 반드시 필요하다[4-8]. 현재 대부분의 Gray Mode Graphic STN LCD 구동회로는 외국에서 수입을 하여 사용하고 있는 실정이며 CDMA, PCS IMT-2000 등에 사용되는 단말기를 위한 Gray Mode Graphic STN LCD 구동회로가 일부 국내 대기업에서 개발되고 있다. 한편, 현재 시판되고 있는 Gray Mode Graphic STN LCD용 구동회로의 경우에 대부분 전용 주제어기와 8086, Z80등과 같은 마이크로프로세서를 함께 사용하는 것이 일반적인 경우이다[9]. 그러므로 사용자가 Gray Mode Graphic STN LCD를 사용하기 위해서는 LCD의 동작특성을 이해해야 함은 물론, 마이크로프로세서의 동작을 이해하고 그에 맞는 어셈블러, C 언어 등을 사용하여 프로그램을 하여야 하는 어려움이 있다. 그리고 이러한 작업은 다분히 하드웨어적인

※ 이 논문은 1999년도 대전산업대학교 교내학술연구비 지원을 받았다.

† 정 회 원 : 국립 한밭대학교 전기전자 제어공학부 교수

†† 정 회 원 : (주)아이뉴정보통신 기술연구소 연구소장

논문접수 : 2001년 11월 2일, 심사완료 : 2002년 2월 5일

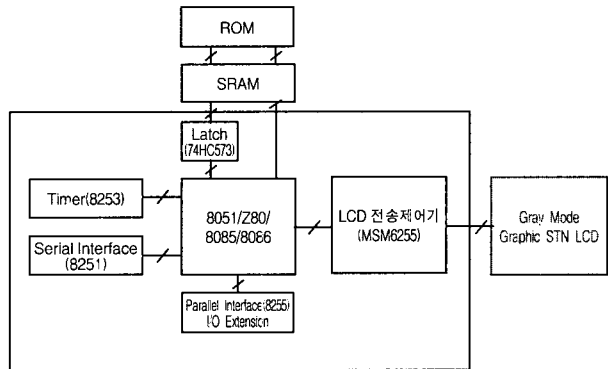
영역에서 다루어져야 하는 일이 되고 있다. 다시 말해서 하드웨어적으로 프로세서에 의해서 구성되어진 후에 소프트웨어에서 이 하드웨어의 모든 특성에 맞게 중간에 드라이버 역할을 할 수 있도록 일일이 프로그램을 하여야 한다. 또한 마이크로프로세서가 바뀔 때마다 각기 다른 방식으로 설계를 하여야 하기 때문에, Gray Mode Graphic STN LCD를 단순히 출력 도구로서 사용하고자 할 경우에는 PCB 점유 면적, 회로의 Artwork 작업시의 소요비용, PCB의 가격적인 면에서 볼 때에도 그다지 유용한 방법이라고 볼 수 없다. 따라서 본 논문에서는 마이크로프로세서를 사용하지 않고 USB 인터페이스[11]를 이용하여 Gray Mode Graphic STN LCD[12]를 구동하는 구동회로를 제안한다. 제안한 구동회로는 사용자가 PC 상에서 작업한 이미지 데이터를 USB 인터페이스를 이용하여 고속으로 LCD로 전송할 수가 있다. 따라서 기존의 방식과는 달리 마이크로프로세서를 사용하지 않기 때문에 사용자가 손쉽게 LCD를 구동할 수 있는 장점이 있다. 즉 PC상의 소프트웨어에서 사용자가 작성한 문자나 이미지를 PC의 USB 포트를 통하여 LCD에 전송하여 손쉽게 고속으로 디스플레이 할 수가 있다. 또한 LCD 구동회로를 1개의 ALTERA 사의 EPF10K10TC 144-3 FPGA를 사용하여 설계하였기 때문에 기존의 LCD 구동방식보다 하드웨어로의 제작시 인쇄회로기판의 크기를 축소할 수가 있어 제작 원가를 감소시키는 장점이 있다.

2. 본 논문에서 제안한 LCD 구동회로의 특징

2.1 기존 LCD 구동회로의 특징

기존 LCD 구동회로의 특징은 (그림 1)에 나타난 바와 같이 마이크로프로세서(8051/Z80/8085/8086)와 타이머(8253), 직렬 인터페이스(8251), 병렬 인터페이스(8255), ROM과 SRAM을 제어하기 위한 래치(74HC573), 이미지 데이터를 LCD에 전송하기 위한 LCD 전송제어기(MSM6255) 등의 주변장치들을 사용하여 Gray Mode Graphic STN LCD를 제어한다. 그

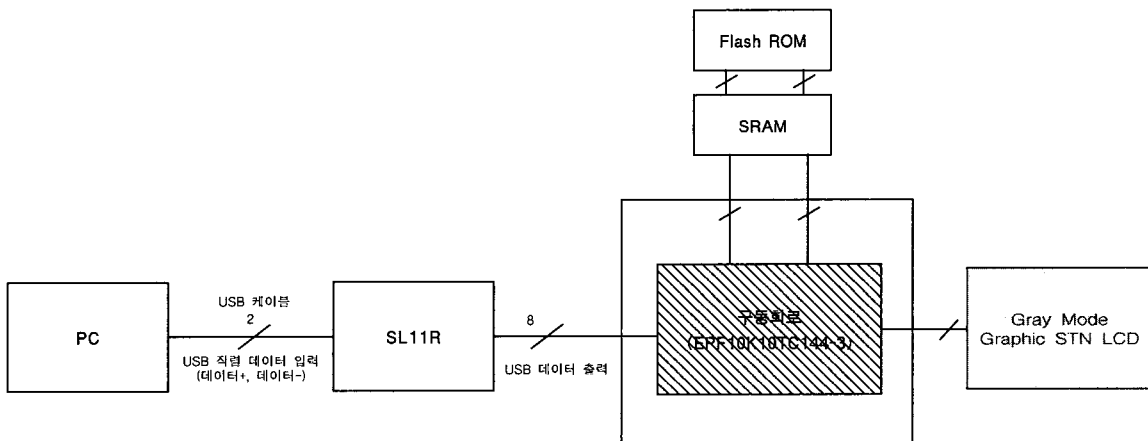
러므로 사용자가 LCD를 제어하기 위해서는 LCD의 동작특성을 이해해야 함은 물론, 마이크로프로세서의 동작을 이해하고 그에 맞는 어셈블러, C 언어 등을 사용하여 프로그램을 하여야 하는 어려움이 있다. 그리고 이러한 작업은 다분히 하드웨어적인 영역에서 다루어져야 하는 일이 되고 있다. 다시 말해서 하드웨어적으로 마이크로프로세서에 의해서 구성되어진 후에 소프트웨어에서 이 하드웨어의 모든 특성에 맞게 중간에 드라이버 역할을 할 수 있도록 일일이 프로그램을 하여야 한다. 또한 마이크로프로세서가 바뀔 때마다 각기 다른 방식으로 주변장치들을 사용하여 설계를 하여야 하기 때문에, Gray Mode Graphic STN LCD를 단순히 출력 도구로서 사용하고자 할 경우에는 PCB 점유 면적, 회로의 artwork 작업시의 소요비용, PCB의 가격적인 면에서 볼 때에도 그다지 유용한 방법이라고 사료되지 않는다.



(그림 1) 기존의 LCD 구동회로의 블록도

2.2 제안한 LCD 구동회로의 특징

본 논문에서 제안한 LCD 구동회로는 (그림 2)와 같이 기존 LCD 구동회로의 마이크로프로세서(8051/Z80/8085/8086)와 타이머(8253), ROM과 SRAM을 제어하기 위한 래치(74HC573), 이미지 데이터를 LCD에 전송하기 위한 LCD 전송제어기(MSM

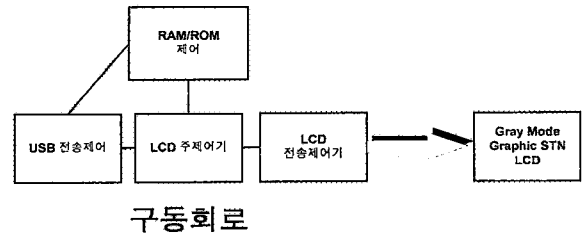


(그림 2) 제안한 LCD 구동회로의 블록도

6255) 등의 주변장치들과 USB 전송제어 부분을 1개의 ALTERA 사의 EPF10K10TC144-3 FPGA를 사용하여 설계하였다. 또한 PC의 USB 포트로부터 직렬로 전송한 8비트의 데이터 및 제어 명령을 주 제어칩인 ALTERA EPF10K10TC144-3에 8비트 병렬로 전송하기 위하여 ScanLogic 사의 SL11R 칩[10]을 사용하였다. 따라서 본 논문에서 제안한 LCD 구동회로는 기존의 방식보다 적은 면적과 저렴한 가격으로 하드웨어를 구현할 수 있다는 장점이 있으며, 메모리 제어를 위한 부가 회로를 하드웨어로 추가하지 않고 FPGA 내부에서 VHDL 코드의 수정으로 메모리의 용량을 확장할 수도 있다. 또한 제안한 LCD 구동회로는 사용자가 PC상에서 작업한 이미지 데이터를 USB 인터페이스를 이용하여 고속으로 LCD로 전송할 수가 있다. 따라서 기존의 방식과는 달리 마이크로프로세서를 사용하지 않기 때문에 사용자가 손쉽게 LCD를 고속으로 구동할 수 있는 장점이 있다. 이밖에 Flash ROM에 이미지 데이터를 프로그래밍 하면 PC상의 USB 포트를 통한 이미지 전송이 없이도 LCD를 구동할 수가 있다.

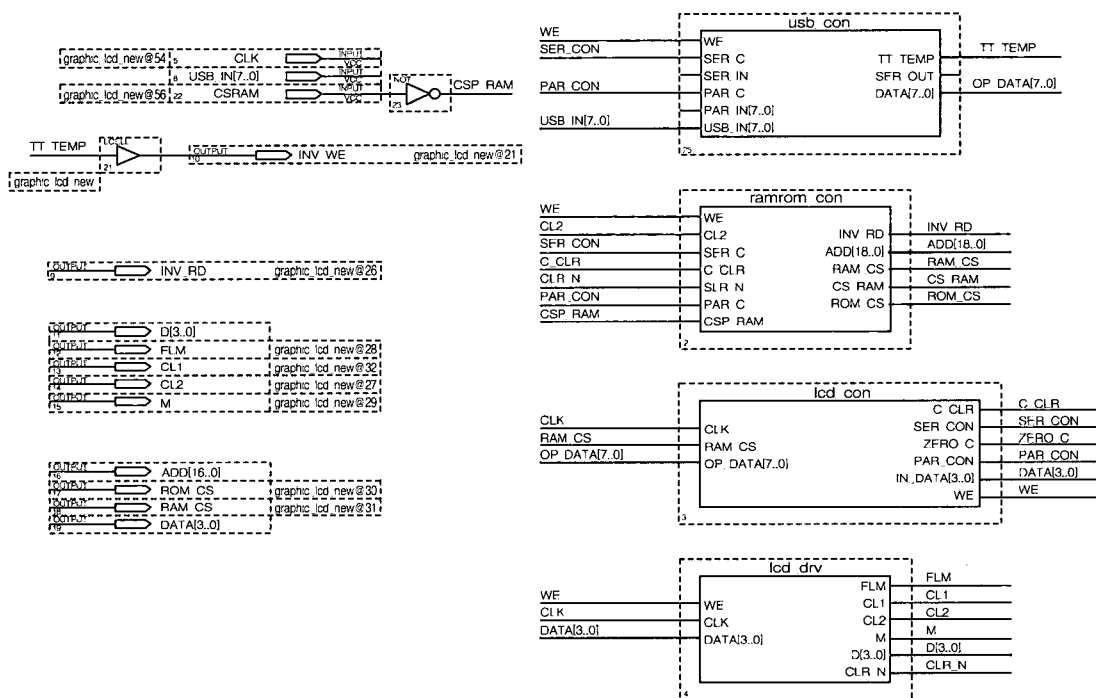
### 2.3 제안한 LCD 구동회로의 사양

(그림 3)은 본 논문에서 제안한 마이크로프로세서를 사용하지 않고 USB 인터페이스를 이용하여 Gray Mode Graphic STN LCD를 구동하는 구동회로의 전체 블록도를 나타내고 있다. 구동회로는 크게 USB 전송제어 블록, LCD 주제어기 블록, LCD 전송제어기 블록, RAM/ROM 제어 블록 등 4부분으로 구성된다.



(그림 3) LCD 구동회로의 블록도

한편, (그림 3)의 블록도에 대한 논리 회로도(그림 4)와 같다. 논리 회로도는 크게 USB 포트를 통해 PC에서 전송하는 데이터 및 제어 신호들을 지정된 패턴의 데이터 및 제어 신호를 구성하게 하는 블록, 구성된 데이터 및 제어신호들을 제어신호에 맞게 메모리에 저장하는 블록, 메모리의 데이터를 LCD의 타이밍 특성에 맞게 LCD에 전송하게 하는 블록으로 구성이 된다. 사용된 메모리는 1Mbits의 SRAM과 1Mbits의 Flash ROM으로 구성이 된다. Flash ROM을 사용하는 경우에는 미리 소프트웨어에 의해서 만들어진 문자나 이미지를 Intel Hex Format 형태의 파일로 구성을 하여 ROM에 프로그래밍 하게 되며, 이 데이터를 LCD 제어 블록에서 제어신호들의 조건에 맞추어 LCD로 전송하게 된다. SRAM을 사용하게 되는 경우는 PC로부터 데이터를 USB 포트를 통하여 받게 되는 경우이다. PC에서 전송되어진 데이터는 제어신호에 의해서 메모리의 동작특성에 맞게 address counting

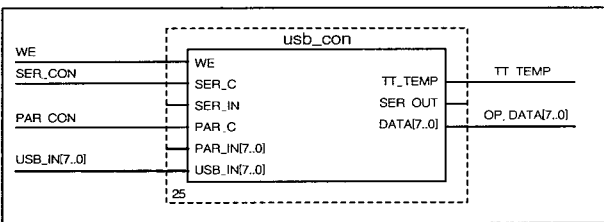


(그림 4) LCD 구동회로의 논리 회로도

을 하게 되어 SRAM에 데이터를 저장하도록 구성이 된다. 한편, 하드웨어로 구성될 Gray Mode Graphic STN LCD는 160×160 Full Dot 이며, 한 화면을 출력하기 위해서는 각 도트의 표시형식이 high 또는 low의 형태, 즉, '1'과 '0'의 상태로 표시되므로 한 화면이 표시되는데 사용되는 메모리의 크기는 25,600bits의 메모리가 필요하게 된다. 따라서 SRAM의 크기가 1Mbits이므로 최대 40프레임의 데이터를 LCD 화면에 출력할 수 있게 된다. 한편, 본 논문에서 제안한 구동회로는 최대 800×600 Full Dot Gray Mode Graphic STN LCD 까지 제어할 수 있으며 SRAM과 Flash ROM을 각각 4Mbytes까지 확장이 가능하도록 설계가 되어 있어 최대 160프레임의 데이터를 LCD 화면에 출력할 수가 있다.

2.4 USB 전송제어 블록

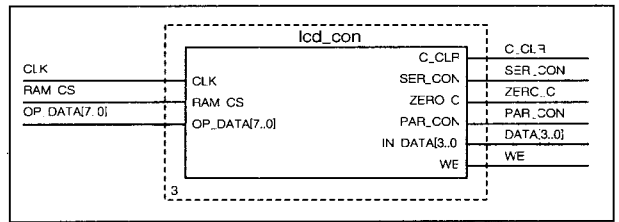
이 블록은 start 제어 신호를 시작으로 데이터를 수신하여 데이터를 SRAM에 저장하며 데이터 전송과 관련 제어신호를 발생한다. 소프트웨어에 의해 만들어진 비트맵 이미지를 LCD의 각 도트에 뿌려질 수 있도록 Intel Hex Format으로 저장을 하고 이 데이터를 Flash ROM 또는 SRAM에 넣어 동작을 시키게 된다. Intel Hex Format의 데이터를 Flash ROM 전용 프로그램을 이용하여 Flash ROM에 프로그래밍 하여 LCD에 출력하고, SRAM을 이용해서 LCD에 출력을 할 경우에는 PC의 USB 포트를 통하여 데이터가 전달되도록 구성되어 있다. 한편, PC의 USB 포트를 통하여 전달된 직렬 데이터는 병렬 형태로 LCD 제어 블록에 의해서 SRAM의 address counting에 의해서 메모리에 순차적으로 저장이 된다. 따라서 USB 방식으로 데이터를 입력받는 경우 PC로부터의 직렬 데이터를 USB 드라이버 칩인 SL11R이 8비트의 병렬 데이터로 변환을 하여 FPGA에 데이터를 공급하게 된다. 한편, 이 블록은 LCD 주제어기 블록에 내장되어 있는 블록으로서 병렬 데이터의 입력이 있는 경우에는 직접 데이터를 받아서 상위 4bits를 제어 코드로, 하위 4bits를 LCD의 화면에 출력하기 위한 데이터로 사용되어 진다. 직렬 데이터의 입력인 경우에는 직렬 데이터를 병렬 데이터로 변환하여 제어코드 부와 데이터 부로 변환하는 기능을 하게 된다. 변환된 제어코드는 소프트웨어의 프로토콜에 맞추어 메모리로의 데이터 입력의 시작과 끝을 알리는 기능을 하게 된다. (그림 5)는 USB 전송제어 블록을 나타내고 있다.



(그림 5) USB 전송제어 블록

2.5 LCD 주제어기 블록

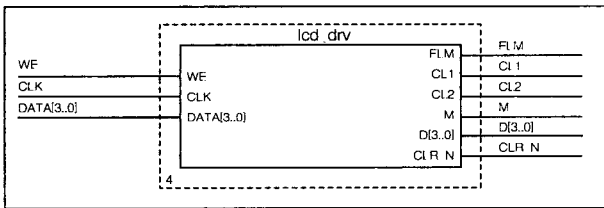
이 블록은 본 논문에서 제안한 LCD 구동회로에서 가장 중요한 부분으로 각 블록들의 제어 및 운영을 총괄하는 기능을 한다. 이 블록의 세부적인 기능은 다음과 같다. 첫 번째는 USB 데이터를 PC로부터 입력받아서 메모리에 저장하도록 한다. 두 번째는 메모리에 저장되어 있는 데이터를 프레임 단위로 LCD 전송제어기로 전송한다. 세 번째는 SRAM에 의한 프레임 이미지 처리와 Flash ROM에 의한 프레임 이미지 처리를 선택하도록 하여 각각의 메모리를 독립적으로 사용할 수 있도록 한다. 이와 같이 제어신호에 의한 화면 출력 데이터를 PC로부터 전송받아 일정한 형식으로 메모리에 저장하고 저장된 메모리의 데이터를 Gray Mode Graphic STN LCD의 동작 특성 및 화면 출력 형식에 맞게 메모리에 저장된 데이터를 변환하여 LCD의 화면에 출력을 할 수 있도록 각 블록에 제어 신호를 보내어 각 블록들이 그에 해당하는 제어옵션을 가지고 각각의 기능을 수행하게 된다. (그림 6)은 LCD 주제어기 블록을 나타내고 있다. 이 블록에서는 CLK에 의한 주파수 입력과 1byte가 전송될 때마다 1로 set되는 RAM\_CS 신호에 의해 전체적인 LCD 주제어기 블록의 역할을 하게된다. RAM\_CS 신호가 '0'인 경우에는 SER\_CON나 C\_CLR과 같은 신호는 '0'값인 low가 되며, RAM\_CS신호가 '1'인 경우에는 USB\_IN [7..0]으로 입력되어온 8bit's의 신호들 중에서 상위 비트인 OP\_DATA[7..4]까지의 신호들 중에서 각각 "0000"인 경우와 "1110"인 경우, "1111"인 경우 그리고 그 외의 조건에 대한 신호의 제어를 나타내고 있다. 여기에서 start와 end 신호의 조건인 "0000"과 "1110"의 조건에서는 USB\_IN [7..0]을 통해 들어온 신호에서 데이터의 시작과 끝을 알리는 신호와 함께 데이터를 입력받아서 메모리에 저장하게 된다. 정확히 표현하면 OP\_DATA는 최초 start bit로 "0000XXXX" 값을 확인한다. RAM\_CS 신호가 1byte 전송되어질 때에 1로 set 되어짐에 따라 제어조건에 의해서 발생된 SER\_CON은 데이터를 저장하기 위한 메모리 address count용 clock으로 사용되어지게 된다. ZERO\_C는 zero address에 최종 데이터가 저장되지 않도록 하기 위한 제어신호이다. LCD를 구동함에 있어서 LCD의 초기 4bits의 데이터가 손실되는 부분을 보완하기 위해 발생하는 신호로서 메모리의 zero address에 최종 데이터가 저장되는 것을 방지하는 기능을 하게 된다. C\_CLR은 데이터가 저장 완료되면 address count를 0으로 reset시킨다.



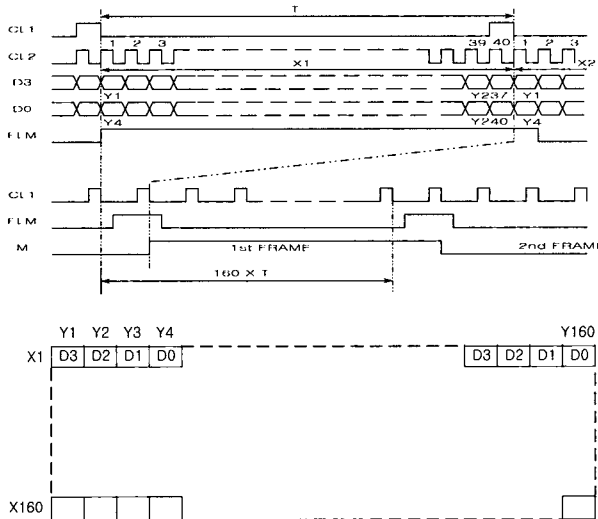
(그림 6) LCD 주제어기 블록

2.6 LCD 전송제어기 블록

(그림 7)은 LCD 전송제어기 블록을 나타내고 있다. LCD 화면에 출력되는 이미지나 문자들은 PC로부터 USB 포트를 통해서 전송되어진 데이터를 USB 전송제어 블록과 RAM/ROM 제어 블록에 의해 메모리에 저장하게 된다. 다음에 LCD 주제어기 블록의 제어에 의해서 USB 포트를 통한 데이터의 전송을 중지하고 모드를 변환하여 LCD 전송제어기 블록을 동작시키며, 메모리에 있는 데이터를 LCD의 액정화면에 출력하게 된다. 이때 출력되는 이미지는 RAM/ROM 제어 블록에 의해 SRAM이나 Flash ROM의 크기에 의해 출력될 수 있는 이미지의 수가 제한이 되어지며, 메모리의 크기만큼 LCD의 도트 수에 맞는 프레임 이미지를 출력하게 되며, 일정 시간동안 화면의 유지를 할 수 있도록 상태유지 카운터를 동작시키도록 하여 일정시간 간격으로 화면이 나타나도록 한다.



(그림 7) LCD 전송제어기 블록



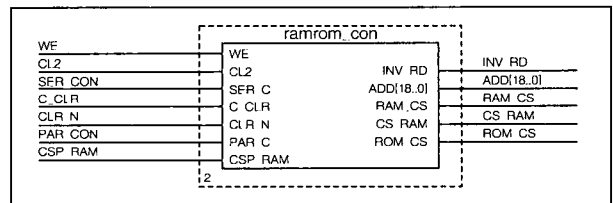
(그림 8) LCD 전송제어기 블록의 타이밍 도

한편, 이 블록은 LCD를 구동시키는 기능을 하며 (그림 8)과 같은 타이밍 도를 가지고 동작되어진다. CL2는 4bits의 데이터를 column의 4bits 병렬 레지스터에 시프트를 시키며 저장되어진다. 초기에 row의 첫 번째 라인을 인식하게 하는 FLM이 160번의 CL1에 한번씩 low에서 high로 1 : 159의 duty 비율로 주기를 이루어 신호를 발생하게 된다. FLM의 기능은 row의 순차적인 active를 이루도록 하며, 각 row의 순차적인 active에 column의 160bits의 데이터가 active된다.

프레임의 반전을 위해 M신호가 low에서 high, high에서 low로 주기적으로 반복된다. 따라서 CL1, CL2, FLM, M의 제어에 의해서 LCD가 구동되며, LCD의 화면에 디스플레이 되는 데이터는 CL2에 맞추어 4bits의 데이터가 입력되어지게 된다.

2.7 RAM/ROM 제어 블록

(그림 9)는 LCD 주제어기 블록에 의해 USB 포트를 통해 입력받은 병렬 데이터를 address count를 통해서 메모리에 저장하도록 하는 RAM/ROM 제어 블록을 나타낸다. 사용되는 LCD의 해상도에 따라 하나의 프레임에 사용되는 이미지의 크기가 달라지므로 하나의 이미지 데이터의 양만큼 메모리의 영역이 할당되도록 제어한다. 테스트를 위해 제작되는 보드에 사용되어진 LCD는 해상도가 160×160 Full Dot 이므로 필요한 데이터의 크기는 25,600bits이다. 따라서 1Mbits의 메모리를 사용할 경우에 40개의 프레임 이미지를 저장하여 사용할 수 있다. 또한 SRAM, Flash ROM에 저장된 데이터를 LCD 주제어기 블록의 명령에 의해 LCD 전송제어기 블록으로 프레임 이미지를 전송하기도 한다. 한편, LCD 주제어기 블록으로부터 발생한 WE 제어신호가 high로 active 되면 PC로부터 전송되어온 데이터를 수신하여 SRAM에 저장하고 low가 되면 SRAM에 저장된 데이터를 LCD에 디스플레이 되도록 메모리의 동작을 제어한다. SER\_CON은 CSP\_RAM에 동기되어 들어오는 데이터를 저장하기 위해 address count를 위한 clock 역할을 하게된다. 이 신호는 WE의 신호가 high로 active가 되면서 address count를 하지 않게 되며, CSP\_RAM 신호가 high가 될 때에 다시 count를 시작하게 된다. CL2는 RAM/ROM의 데이터를 LCD에 디스플레이 시킬 때에 RAM/ROM에 저장된 값을 읽기 위한 address count 역할을 하게 되며, INV\_RD는 CL2 신호에 동기가 되어 데이터를 memory로부터 읽어들이는 때에 사용된다.



(그림 9) RAM/ROM 제어 블록

3. 프로토콜

LCD 구동을 위한 프로토콜은 (그림 10)과 같이 크게 2개의 블록으로 구성이 된다. 첫 번째 블록은 USB 인터페이스를 위한 데이터 전송 프로토콜로서 PC로부터의 USB cable을 통해 직렬 데이터가 전송이 되었을 때에 직렬 데이터가 병렬 데이터로 변환이 이루어지게 한다. 이 블록은 ScanLogic사의 SL11R 칩을 사용하여 구현되었다. 두 번째 블록은 LCD

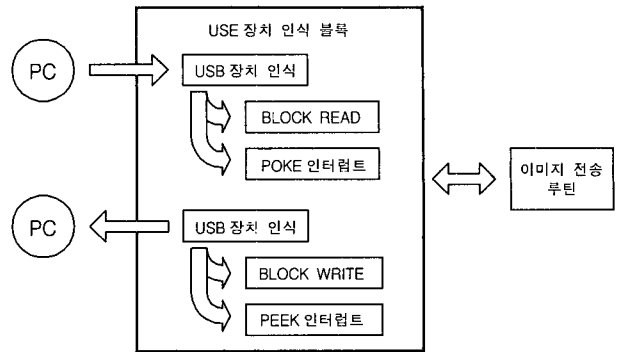
에 데이터를 전송하기 위한 제어 프로토콜로서 첫 번째 블록에 의하여 출력된 병렬 데이터를 입력받아 상위 4bits를 제어에 위한 명령으로 하위 4bits를 이미지 전송을 위한 데이터로 활용을 하게 된다. 이 블록은 ALTERA 사의 EPF10K10TC144-3 FPGA를 사용하여 구현되었다.



(그림 10) LCD 구동을 위한 프로토콜

본 논문에서 적용한 USB 규격은 최대 12Mbps의 전송속도를 나타내는 1.1 버전이며, USB 인터페이스를 위한 데이터 전송 프로토콜은 ScanLogic 사의 SL11R 칩에 어셈블리언어를 사용하여 프로그래밍 하였다. 이 프로토콜은 먼저 데이터를 PC로부터 전송 받을 것인지 전송할 것인지를 판별하기 위한 USB 장치 인식 블록과 한번에 8bits의 데이터를 LCD에 데이터를 전송하기 위한 제어 프로토콜에 전송하는 이미지 전송 루틴 블록으로 구성된다. (그림 11)은 USB 장치 인식 블록을 나타내고 있다. USB 주제어기인 SL11R(이하 SLAVE라 칭함)의 백터 인터럽트 중에는 PEEK 인터럽트와 POKE 인터럽트가 있다. 이들 2개의 인터럽트는 모두 End Point 0을 사용한다. PEEK 인터럽트는 SLAVE측에서 PC로 데이터를 전송하는 인터럽트이고, POKE 인터럽트는 반대로 PC로부터 SLAVE로 데이터를 전송하는 인터럽트로서 처음 4bits를 먼저 전송하고 나머지 4bits를 shift right하여 다시 4bits를 전송한 후에 이미지 전송 루틴으로 이동하게 된다. 소프트웨어에서 이들 인터럽트를 호출시 PEEK일 때에는 0x48, POKE일 때에는 0x49를 SLAVE 쪽으로 넘겨주고 SLAVE는 이를 통해 각 인터럽트를 식별하고 해당 인터럽트를 호출한다. PC로부터 LCD 모듈로 이미지 데이터를 전송하는 것은 POKE에 해당하므로 PC는 우선 0x49를 SLAVE로 전송한 후에 이

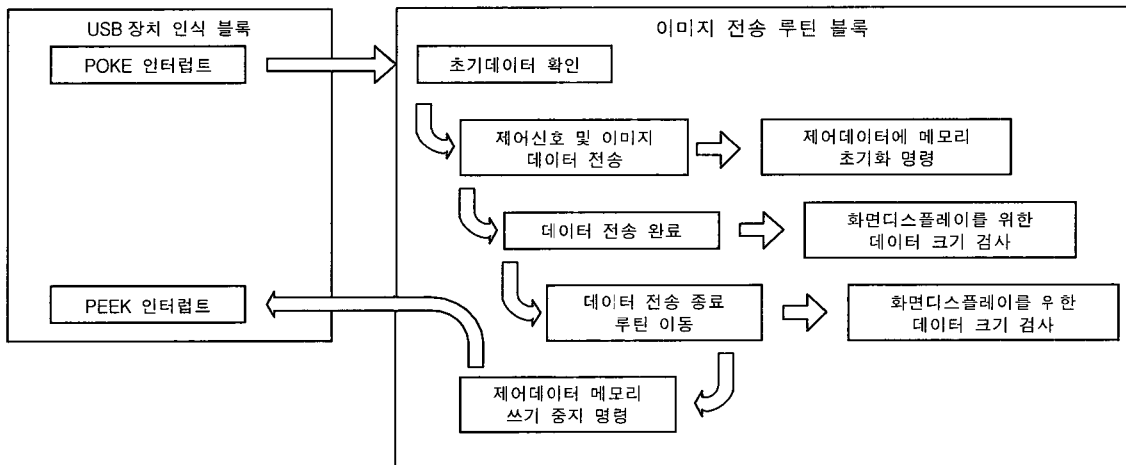
미지 데이터를 전송한다.



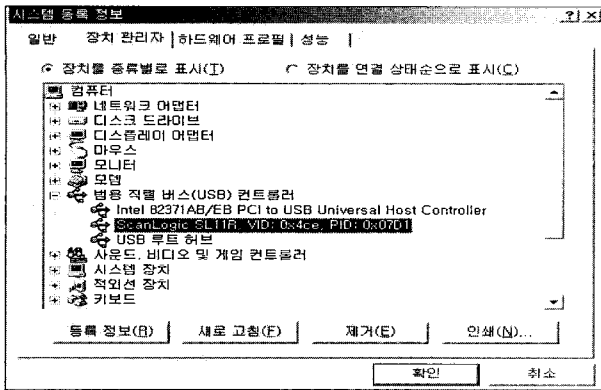
(그림 11) USB 장치 인식 블록

(그림 12)는 이미지 전송 루틴 블록을 나타내고 있다. 한번에 전송하는 데이터는 8bits로 설정하였다. SLAVE는 이 8bits 데이터를 받아 4bits씩 나누어 상위 4bits는 LCD에 데이터를 전송하기 위한 제어 데이터로, 하위 4bits LCD에 이미지를 전송하기 위한 이미지 데이터로서 사용한다. POKE 인터럽트에 의해 이미지 전송루틴이 시작된다. 이미지 전송루틴 블록은 먼저 초기 데이터 확인을 한 후에 PC로부터의 데이터 전송이 시작된다. 제어 신호는 메모리를 초기화하고 메모리에 데이터를 쓰기 위한 주소 카운팅이 시작된다. 이미지에 맞는 데이터의 전송이 완료되면 데이터 전송완료 신호와 함께 데이터 전송을 처리하는 데이터 전송 종료루틴이 가동된다. 데이터가 전송되면서 LCD에 화면 디스플레이를 위한 데이터 크기를 수시로 검사하게 된다. 데이터 전송 종료 루틴은 메모리 쓰기 명령과 어드레스 초기화 명령을 전송하고 PEEK 인터럽트를 호출하여 PC로부터 데이터의 전송을 중지하도록 명령한다.

한편, (그림 13)은 USB 인터페이스를 위한 데이터 전송 프로토콜을 적용하여 작성된 USB 디바이스 드라이버가 PC의 MS-Windows에 설치된 상태를 나타내고 있다.



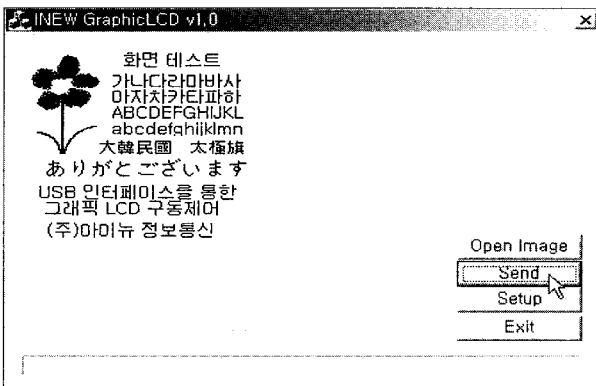
(그림 12) USB 이미지 전송 루틴 블록



(그림 13) PC의 MS-Windows에 USB 드라이버가 설치된 상태

#### 4. 전송 소프트웨어

(그림 14)는 PC상의 소프트웨어에서 사용자가 작성한 문자나 이미지를 PC의 USB 포트를 통하여 LCD에 전송하여 손쉽게 디스플레이하기 위하여 MS-Visual C++ 6.0으로 프로그래밍 된 그래픽 사용자 인터페이스 방식의 전송 소프트웨어를 나타내고 있다. 전송 소프트웨어는 흑백 scale의 BMP 파일과 애니메이션을 위한 프레임용 USB 포트를 이용하여 Gray Mode Graphic STN LCD에 전송할 수 있다.



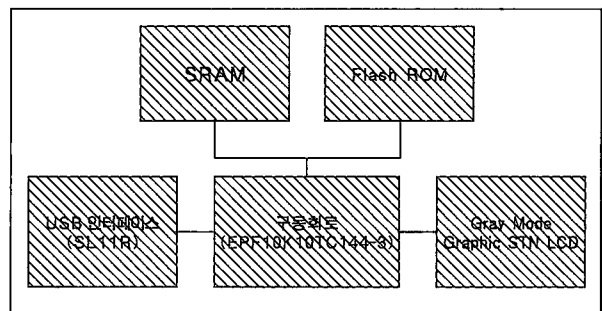
(그림 14) LCD에 데이터를 전송하기 위한 그래픽 사용자 인터페이스 방식의 전송 소프트웨어

#### 5. 시스템의 구현 결과 및 고찰

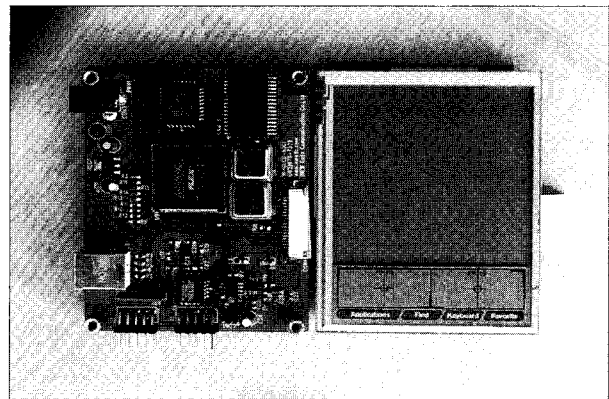
##### 5.1 테스트 보드

본 논문에서 제안한 LCD 구동회로를 VHDL로 설계한 후에, ALTERA MAX+PLUS II를 사용하여 시뮬레이션을 수행한 결과 설계된 구동회로의 논리적 기능이 오류가 없음을 검증하였다. 따라서 설계된 논리회로를 실제 하드웨어로 구현하여 동작검증을 수행하기 위하여 테스트 보드를 제작하였다. (그림 15)는 테스트보드의 블록도를 나타내고 있다. 테스트 보드는 1Mbits의 SRAM, 1Mbits의 Flash ROM, 160×160 Full Dot Matrix Gray Mode Graphic STN LCD, ALTERA

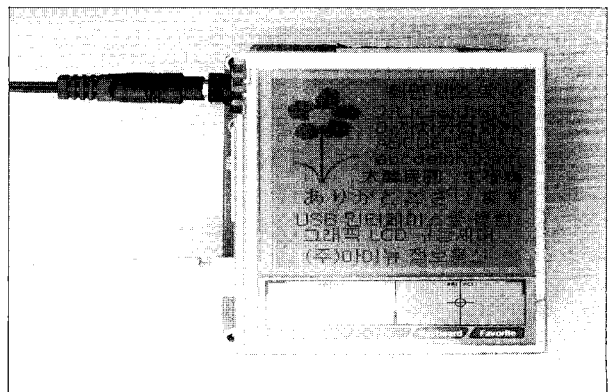
사의 EPF10K10TC144-3 FPGA, ScanLogic 사의 SL11R 칩 등을 사용하여 구성하였다. 특히 본 논문에서 제안한 LCD 구동회로는 166개의 로직 셀(약 2,000게이트)과 46.72MHz 주파수를 사용하여 EPF10K10TC144-3 FPGA에 구현되었다. 한편, (그림 16)은 제작된 테스트 보드의 실제 모습을, (그림 17)은 제작된 테스트 보드의 동작 모습을 나타내고 있다. (그림 20)에서 나타난 바와 같이 제안된 구동회로는 PC상의 소프트웨어에서 사용자가 작성한 문자나 이미지를 PC의 USB 포트를 통하여 LCD에 전송하여 손쉽게 디스플레이하였다. 따라서 본 논문에서 제안한 마이크로프로세서를 사용하지 않고 USB 인터페이스를 이용하여 Gray Mode Graphic STN LCD를 구동하는 구동회로의 효율성이 입증되었다.



(그림 15) 테스트 보드의 블록도



(그림 16) 테스트 보드의 실제 모습



(그림 17) 테스트 보드의 동작 모습

5.2 타 업체의 구동회로와 비교분석

본 논문에서 제안한 LCD 구동회로를 현재 시판중인 타 업체의 구동회로[9]와 비교한 결과가 <표 1>에 나타나 있다. 본 논문에서 제안한 LCD 구동회로는 마이크로프로세서를 사용하지 않기 때문에 구동회로의 작동 편의성, 인쇄회로기판의 크기감소로 보드로 제작시 소요되는 비용 등에서 현재 시판중인 타 업체의 구동회로보다 우수함을 나타내었다. 따라서 사용자가 손쉽게 사용할 수 있는 LCD 구동회로를 적은 비용과 작은 크기로 제작할 수 있을 것으로 기대된다.

<표 1> LCD 구동회로의 성능비교

구동회로의 종류 비교항목	본 논문에서 제안한 LCD 구동회로	현재 시판중인 타 업체의 LCD 구동회로
구동회로 작동방식	사용자가 PC상에서 작업한 데이터를 USB 인터페이스를 이용하여 고속으로 LCD로 전송하기 때문에 LCD 작동이 편리함	마이크로프로세서를 사용하여 어셈블리 언어로 프로그래밍 하여 LCD에 문자나 이미지를 디스플레이 하기 때문에 LCD 작동이 어려움
구동회로 구성	1개의 ALTERA 사의 EPF10K10TC144-3 FPGA + ScanLogic 사의 SL11R 칩	마이크로프로세서 + 전용 주제어기 + 주변장치들 (타이머, 직렬 인터페이스, 병렬 인터페이스, 래치)
인터페이스 방식	USB 인터페이스	직렬/병렬 인터페이스
제작시 소요되는 비용	50,000원	100,000원

6. 결 론

본 논문에서는 마이크로프로세서를 사용하지 않고 USB 인터페이스를 이용하여 Gray Mode Graphic STN LCD를 구동하는 구동회로를 제안하였다. 제안된 LCD 구동회로는 마이크로프로세서를 사용하지 않고 PC상의 소프트웨어에서 사용자가 작성한 문자나 이미지를 PC의 USB 포트를 통하여 LCD에 전송하여 손쉽게 고속으로 디스플레이 할 수가 있는 장점이 있다. 또한 LCD 구동회로를 1개의 ALTERA 사의 EPF10K10TC144-3 FPGA를 사용하여 설계하였기 때문에 기존의 LCD 구동방식보다 하드웨어로의 제작시 인쇄회로기판의 크기를 축소할 수가 있어 제작 원가를 감소시키는 장점이 있다. 제안한 LCD 구동회로를 테스트 보드를 구성하여 하드웨어 동작검증을 수행하여 그 효용성을 입증하였다. 또한 타 업체의 구동회로와 비교 분석한 결과, 본 논문에서 제안한 구동회로가 구동회로의 작동 편의성, 인쇄회로기판의 크기감소로 보드로 제작시 소요되는 비용 등에서 우수함을 나타내었다. 따라서 본 논문에서 제안한 구동회로를 사용할 경우에 사용자가 손쉽게 Gray Mode Graphic STN LCD를 구동할 수 있으며 하드웨어로의 제작시 인쇄회로기판의 크기를 축소할 수가 있어 제작 원가를 감소시킬 수가 있으리라 기대된다.

참 고 문 헌

- [1] 이승수, "STN 액정표시소자의 기술과 전망", 대한전자공학회지, 제22권 제3호, pp.82-91, 1995.
- [2] 이체우, "전자 DISPLAY 기술동향", 대한전자공학회지, 제22권 제3호, pp.23-30, 1995.
- [3] 장진, "박막트랜지스터 액정디스플레이", 대한전자공학회지, 제26권 제2호, pp.44-61, 1999.
- [4] De Rycke I ; Van Calster, A ; VanFleteren, J ; De Baets, J ; Doutrelaigne, J ; De Smet, H ; Vetter, P., "2-MHz clocked LCD drivers on glass," Solid-State Circuits, IEEE Journal of, Volume : 25 Issue : 2, pp.531-538, April, 1990.
- [5] Valencic, V ; Ballan, H ; Deval, P ; Hochet, B ; Declercq, M., "50-V LCD driver integrated in standard 5-V CMOS process Custom Integrated Circuits Conference," 1994., Proceedings of the IEEE 1994, pp.578-581, 1994.
- [6] 권오경, "Flat Panel Display를 위한 구동방식 및 구동회로", 전자공학회지, 제22권 제3호, pp.92-105, 1995.
- [7] 김준배, 정병후, 배병성, 한민구, 권오경, "Cycling Shift Register를 이용한 새로운 LCD 구동회로의 설계", 대한전자공학회 추계종합학술대회논문집(B), 제19권 제2호, pp.841-844, 1996.
- [8] 이수범, 남상길, 조정연, 김종진, "광시각용 LED 전광판 제어 시스템 설계", 한국정보처리학회논문지, 제6권 제5호, pp.1351-1361, 1999.
- [9] MSM6255 DOT MATRIX LCD CONTROLLER, OKI semiconductor Data Sheet.
- [10] SL11R Hardware Specification Technical Reference, ScanLogic Corporation DataSheet.
- [11] 전세일, 이두복, "USB 인터페이스를 이용한 데이터 전송프로그램 개발", 한국정보처리학회논문지, 제7권 제5호, pp.1553-1558, 2000.
- [12] Specifications for Approval(UG-24D01-GEBT5-E), SAMSUNG DISPLAY DEVICES CO., LTD.



이 승 호

e-mail : shlee@cad.hanbat.ac.kr

1986년 한양대학교 전자공학과 졸업(공학사)  
 1989년 한양대학교 대학원 전자공학과 졸업(공학석사)  
 1994년 한양대학교 대학원 전자공학과 졸업(공학박사)

1994년~현재 국립 한밭대학교 전기·전자·제어공학부 부교수  
 관심분야 : 집적회로설계(CAD for VLSI), 시스템온칩설계, 컴퓨터통신, 인터넷



이 주 현

e-mail : jhlee@inewdc.com

1997년 대전산업대학교 전자공학과 졸업(공학사)  
 2000년 대전산업대학교 대학원 전자공학과 졸업(공학석사)  
 1998년~1999년 (주)한백전자 기술연구소 연구소장

2000년~현재 (주)아이뉴정보통신 기술연구소 연구소장  
 관심분야 : 집적회로설계(CAD for VLSI), 시스템온칩설계, Embedded System