임베디드 시스템 플랫폼 개발을 위한 시뮬레이션 환경 구현

*이중희, 오현석, 성광수
영남대학교 전자공학과
e-mail : gomusin08@hotmail.com

A Design of Platform for Embedded System's development

*Joong-Hee Lee, Hyun-Seok Oh, Kwang-Soo Sung
Department of Electronic Engineering Graduate School
Yeungnam University

Abstract

This treatise proposed environment for Embedded system's development. Virtual platform can help to solve problem that hardware designer can experience at design process of hardware. Compose circuit of most suitable that is verified before mix parts by various kinds method and compose circuit by board level because can do simulation with software and software that is optimized to hardware and offer flexibility that can test. Therefore, can shorten expense that is cost in development and time. Embody development platform for 8051 systems for verification of development platform, and compose and verified system in various kinds structure.

I. 서론

임베디드 시스템은 컴퓨터 하드웨어와 소프트웨어가 조합되어 특정 목적을 수행하는 시스템으로 초기의 임베디드 시스템은 그 구성이 매우 단순하였다. 16bit 이하의 마이크로프로세서가 일반화되고 어셈블리 언어를 이용하여 순차적으로 동작하는 응용 프로그램을 이용하여 동작하는 응용 소프트웨어 정도의 사용만으로도 훌륭히 여러 가지 제품에 적용될 수 있었다. 그러나 하드웨어의 고집적화와 저가격화 그리고 고객 요구사항의 증가는 32bit 이상의 계산능력이 강화된 마이크로프로세서와 복잡한 주변회로의 사용을 요구하고 있다. 또한 복잡한 기능을 구현하고 빠른 시장 출시를 위한 제품 개발 시간의 단축을 위하여 네트워킹 기능과 멀티태스킹 기능 등의 지원을 위해, 운영체제를 기반으로 하는 임베디드 시스템 구성이 일반화되어 하드웨어 사양이 높아지고 있다.

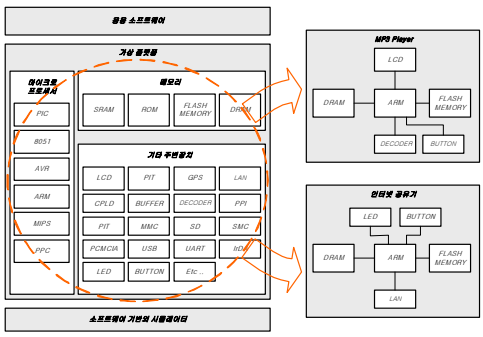
본 논문에서는 복잡한 하드웨어를 효율적으로 구성하고 소프트웨어와 함께 시뮬레이션을 할 수 있는 가상 개발환경을 제공하고자 한다. 이는 하드웨어 구성의 최적화를 이룰 수 있을 뿐만 아니라 소프트웨어를 포함한 검증을 통해 하드웨어를 구현하기 전에 문제점을 미리 발견할 수 있도록 한다. 또한, 여러 장치들을 융통성 있게 추가 또는 삭제할 수 있는 많은 기회를 제공해 줄 수 있다. 잘못된 하드웨어 구성으로 인한 개발기간의 손실과 많은 경험이 필요한 디버깅의 어려움을 소프트웨어의 장점을 통해서 줄여줄 수 있도록 도와준다.

II. 본론

최적의 하드웨어를 구성하기 위해서 회로를 구성하게 되었을 때 하드웨어 상에서 문제점이 발생하게 되면 많은 디버깅 과정에서 어려움을 겪게 되는데, 본 논문에서는 하드웨어 구성에 도움을 줄 수 있는 다음과 같은 개발 과정을 제안하였다.

시스템을 설계하는 과정에서 하드웨어와 소프트웨어를 구현하기 이전에 시뮬레이션 환경을 만들어 설계된 하드웨어를 검증하여 하드웨어의 버그로 인하여 발생할 수 있는 개발과정상의 문제를 보완할 수 있다. 또한 하드웨어와 관련이 있는 소프트웨어의 경우 디버깅을 할 때 하드웨어의 상태를 시뮬레이션을 통해 확인할 수 있으므로, 하드웨어에 관련된 소프트웨어 또한 효율적으로 확인하고, 최적화할 수 있다.

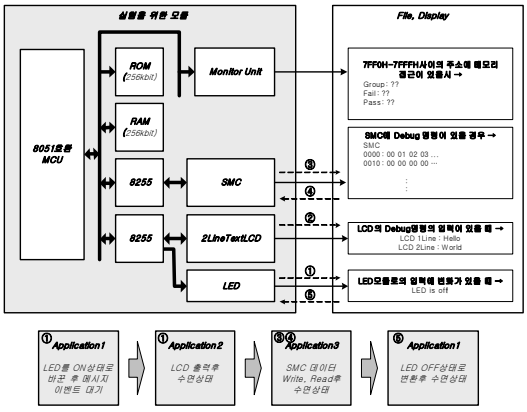
시뮬레이션의 환경에서는 사용하고자 하는 칩들을 구현하고, 설계자가 원하는 조합으로 추가, 삭제가 가능하며, 소프트웨어 상에서 구현이 되어 진다.



제안된 개발환경을 이용하게 되면 다음과 같은 장점을 얻을 수 있다.
① 소프트웨어로 만들어진 여러 장치들을 쉽게 조합해 볼 수 있으므로, 여러 가지 구성의 시스템을 구성하여 시뮬레이션 할 수 있다.
② 설계자가 원하는 모듈 내부의 동작 상황을 시뮬레이션 상에서 확인이 가능하다.
③ 보드레벨로 구현하기 전에 하드웨어의 문제점을 미리 발견할 수 있으므로, 하드웨어 제작에 드는 비용을 줄일 수 있다.

Ⅲ. 구현

아래 그림과 같은 가상 플랫폼을 구현하기 위하여 CPU와 RAM, ROM, SMC, 8255, LED, 2Line Text LCD 등의 여러 가지 모듈을 소프트웨어로 구현하였고, 이를 조합하여 8051의 시스템을 구현하였다. 시스템의 정상적인 동작을 확인하기 위한 소프트웨어로는 uC/OSII를 이용하여 테스트를 위한 프로그래밍을 하였다.



Ⅳ. 결론 및 향후 연구 방향

본 논문에서는 임베디드 시스템에서 하드웨어를 개발하는 설계자의 부담을 덜어줄 수 있도록 가상의 개발환경을 구축하였다. 하드웨어의 구성에 사용되는 각각의 모듈은 C언어를 이용하여 구성하였고, 이 모듈은 하드웨어 언어를 이용한 논리회로를 통해 서로 연결시켜서 가상의 하드웨어 개발환경을 구성하였다. 각각의 모듈 내부에 있는 프로그램에서는 외부 출력 파일을 통해서 프로그램이 동작하는 상황을 사용자가 원하는 형태로 출력하여 볼 수 있어서, 하드웨어의 정상적인 동작의 유무나 테스트를 위한 소프트웨어의 정상적인 동작 유무를 판별하는데 도움을 주었다.
각각의 모듈은 하드웨어 언어라는 프로그램 형태의 환경을 사용하여 연결함으로써 CPU 주변의 회로를 설계자가 원하는대로 추가할 수 있고, 여러 모듈 간에 인터페이스 또한 논리회로를 이용하여 구성할 수 있다. 따라서 최적의 성능을 내도록 구성된 회로를 구성하기 위해 많은 테스트가 가능하고, 모듈의 구성에 의한 논리회로 구성 외에도 CPLD를 이용하면 하드웨어 언어로 구성된 논리회로를 실제 하드웨어에서도 그대로 사용할 수 있다.
일반적인 목적으로 제공되는 회로도에서 원하는 기능을 추가, 삭제, 변경을 할 때, 발생할 수 있는 시간과 비용, 배포 후에 발생할 수 있는 손실을 줄여줄 수 있는 개발환경으로써 충분히 가치가 있고, 기존의 개발과정 중 설계된 각 소자는 앞으로도 계속 활용할 수 있다.

참고문헌

[1] 나종화, 강순주, 윤용익, 박윤용, 은성배, 김홍남 공저, “EMBEDDED SYSTEMS PROGRAMMING”, 사이텍미디어, 2005
[2] SAMIR PALNITKAR 지음, 장훈 옮김, “Verilog HDL″, 영한출판사, 2002
[3] 배석희, 한상홍, 전영준 공저, “위피 WIPI 모바일 프로그래밍 기술을 통일한 위피(WIPI)입문서”, 대림, 2004
[4] 여인춘, 김효남, 박희경, 이동훈, 류준상 공저, “임베디드 프로그래밍”, 길벗, 2004
[5] JEAN J. LABROSSE 지음, MicroC/OS-II The Real-Time Kernel 2/E, 에어콘, 2003
[6] 윤덕용, “어셈블러와 C로 익히는 8051 마스터“, Ohm사, 2000년
[7] 송태훈, 남상엽, “알기쉬운 임베디드 시스템 응용”, 홍릉과학출판사, 2005
[8] 최병욱, 고경철, 문전일, 임계영 공저, “임베디드 리눅스 실습 및 활용”, 홍릉과학출판사, 2003
[9] Weng Fook Lee, ″Verilog Coding for Logic Synthesis″, WILEY
[10] T. R. PADMANABHAN, B. BALA TRIPURA SUNDARI, ″Design through verilog HDL″, WILEY-INTERSCIENCE
[11] 송용수, “임베디드 시스템의 설계를 위한)ARM7 Bible : Embedded system design″, 복두출판사, 2004
[12] 송태훈, “Intel PXA255와 임베디드 리눅스 응용“, 사이텍미디어, 2004
[13] 우상철, “임베디드 리눅스 시스템 설계 : Sscale PXA 250 RISC CPU를 이용한” Ohm사, 2004
[14] Sutter, Ed, ″임베디드 펌웨어 분석“, 에어콘출판사, 2002
[15] 서강수, “임베디드 시스템 설계”, 복두출판사, 2004
[16] Kernighan, Brian W., ″C언어 프로그래밍“, 대영사, 2004
[17] 전교화미, “C 언어 포인터 완전제패“, 영진닷컴, 2002