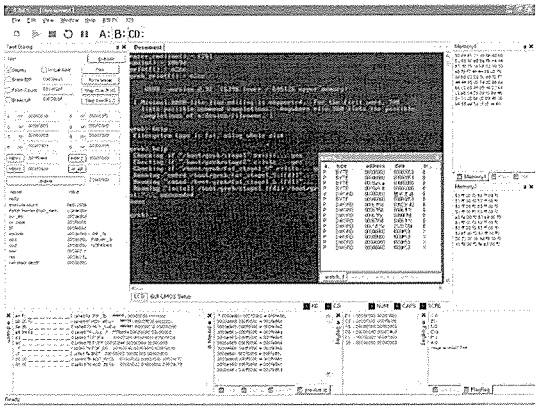
영남대학교 "IA-32 기반 PC Emulator"

System Emulate

System Development, Debugging



영남대학교 전자공학부 예병석, 전경재와 경북대학교 대학원 김현철이 공동 개발한 IA-32 기반 PC Emulator은 System Emulate와 System Debugging을 하기위한 프로그램이다. System Emulate의 기능으로는 Intel 32bit CPU, Monitor, Keyboard, Mouse 등의 I/O, Timer Controller, Interrupt Controller, Floppy Disk Controller, IDE Controller 을 Emulate하였습니다. 두 번째 기능인 System Debugging은 특정 OPCODE체크, 특정 OPERAND, 메모리 R/W 되는 시점, I/O R/W되는 시점에 시스템을 정지시켜 필요한 작업을 할 수 있고 Debugging할 때 필요한 기능인 CPU Register, CPU Status, Watch(Memory, Register), Memory Viewer, Current Code, Step Into, Step Over, Memory Dump, System Break, Call Stack, Disassembler을 지원합니다.

IA-32 기반 PC Emulator 기능들을 통해서 사용자는 편리하게 가상 System을 사용할 수 있고 System 개발자는 커널, 디바이스 드라이버 개발을 할 때 System Debugging을 통하여 손쉽게 System의 정보를 얻을 수 있습니다.

System 개발에 사용되는 툴로는 커널레벨에서 Debugging을 하는 툴인 SoftIce나 WinDbg 등 이 있지만 사용방법이 어렵고, 시스템에 독립적으로 동작 할 수 있는 것이 아니기 때문에 많은 제약이 있습니다. 이에 비해 우리 프로그램은 가상으로 컴퓨터를 Emulate함으로써 정상적인 방법으로 접근 할 수 없는 것을 가상으로 Emulate하여 System의 정보를 확인할 수 있다는 측면에서 큰 의미가 있습니다.

정보통신부장관상

IA-32 기반 PC Emulator

1. 작품명 : IA-32 기반 PC Emulator

2. 제작자 : 영남대학교

대표자 : 예병석
개발참여자 : 전경재, 김현철
주소 : (712-060) 경북 경산시 중방동 광명 APT 1동 501호
전화 : 010 - 9956 -2350
email : ouma02@paran.com

3. S/W 요약설명

IA-32 기반 PC Emulator은 System Emulate와 System Development, Debugging기능을 통해 사용자가 System 개발에 유용하게 사용할 수 있도록 하는데 목적이 있습니다.

3.1 개발 배경

현재 우리나라뿐 아니라 전 세계적으로도 많은 시스템 개발자와 소프트웨어 개발자 등 많은 개발자들이 있습니다. 하지만 아쉽게도 시스템 개발에 사용되는 툴이 많이 개발되어 있지 않습니다. 따라서 System 개발자들은 일반 응용 소프트웨어 개발자보다 많은 어려움이 있습니다. 이런 불편함을 해소하고자 이 프로그램을 만들게 되었습니다.

1) System Emulate
- Host PC와 독립적인 환경을 구축하기 위해서 PC를 Emulate

하여 Software적으로 Computer System을 구축합니다.

2) System Debugging
- 외산 제품을 사용할 때 제약사항인 커널레벨에서의 Debugging 이 아닌 System 전 영역에서 원하는 정보를 얻을 수 있도록 합니다.

3) 손쉬운 User Interface 제공
- 사용자가 쉽게 사용할 수 있도록 WTL을 사용하여 친숙한 개발 GUI환경을 제공합니다.

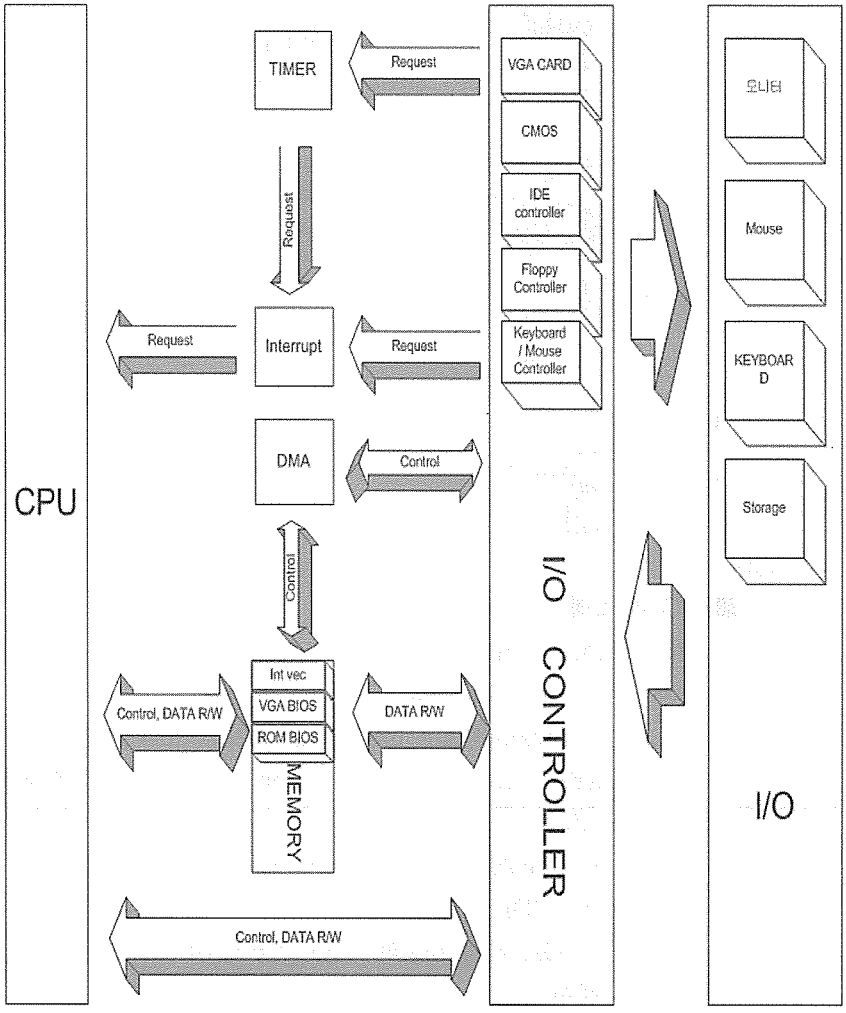
3.2 시스템 개요

IA-32 기반 PC Emulator는 System 개발에 필요한 기능을 가지고 있는 프로그램으로써 Host PC에 독립적으로 작동하게 하기위해서 System Emulate를 하였고, 기존의 방식인 커널레벨에서의 Debugging에서 벗어나 System 레벨에서의 Debugging이 가능하도록 만들어진 프로그램이다. 따라서 이 프로그램을 통해서 커널, 디바이스 드라이브 개발에 도움을 주고자 하는 프로그램입니다.

3.3 시스템 특징

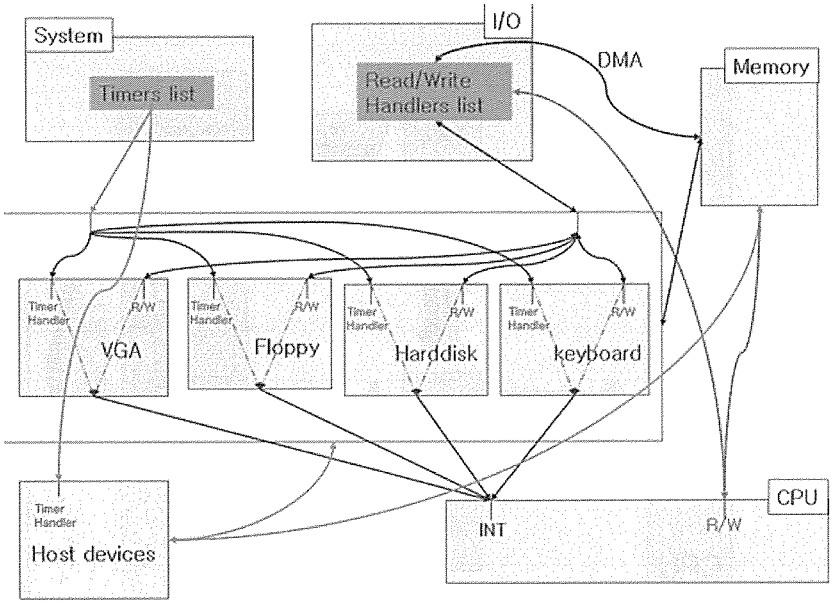
- Computer System 전체를 Emulate하여 Debugging할 때 OS에 무관하게 Debugging 이 가능합니다.
- System 전체 영역에 걸친 Debugging 및 개발이 가능합니다.
- 개발자의 편의를 위해 WTL을 사용하여 개발자에게 친숙한 Visual Studio. NET과 비슷한 환경의 인터페이스를 제공합니다.
- 가상의 System 하에서 독립적인 환경을 구축하고 Debugging을 실시하므로 Host System에 개발도중 영향을 주지 않습니다.
- 가상의 Computer System을 정지하고, 그 상태에서 모든 하드웨어 및 메모리 상태를 Dump 가능합니다.

3.4 시스템구성 및 주요기능



< 시스템 구성도 >

위의 시스템 구성도는 저희의 프로그램을 통해 Emulate된 가상의 PC의 시스템 구성 형태를 보여줍니다.



< I/O 시스템 >

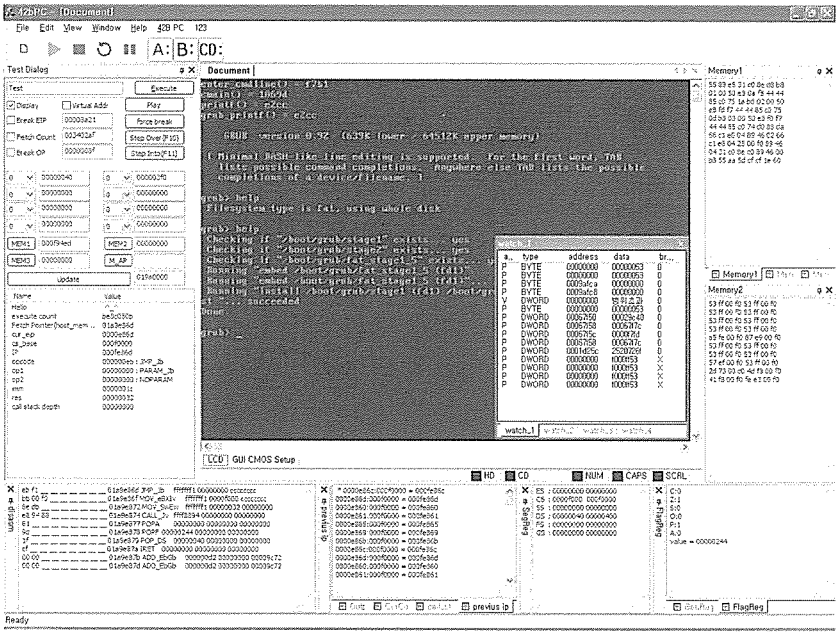
■ I/O 시스템

- PCI - PCI Controller
- DMA - Direct Memory Access Controller
- PIC - Programmable Interrupt Controller(PIC)
- Keyboard - keyboard controller + Host 컴퓨터와의 입출력 인터페이스
- CMOS - CMOS Ram과 입출력 인터페이스
- HardDrv - IDE controller + HardDisk Image
- VGA - VGA Memory와 VGA Controller + Host 컴퓨터 출력 인터페이스
- FloppyDrv - Floppydrv Controller + Floppy Image
- PIT - timer
- BIOS - BIOS Panic, info, debug등을 처리
- PCI to ISA Bridge - PCI인터페이스에 ISA를 처리하기위한 브릿지
- Parallel - 병렬 통신 컨트롤러
- Serial - 직렬통신 컨트롤러

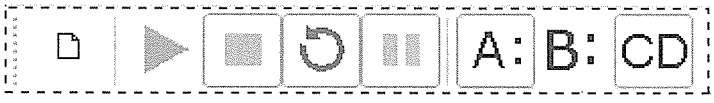
3.5 프로그램 주요 화면 구성

■ Emulate 모드

(1) 전체 화면 구성



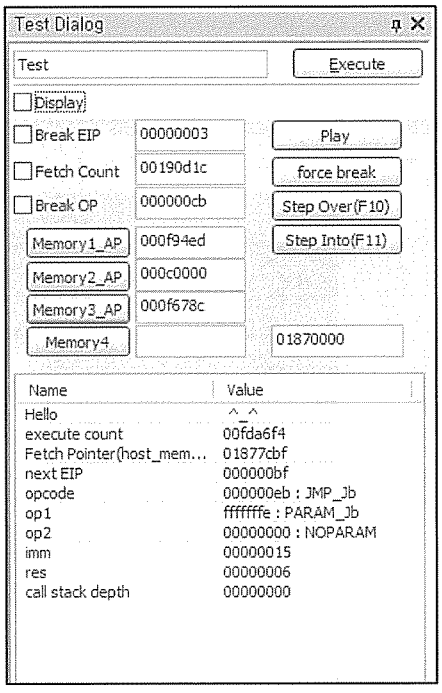
(2) 메뉴 화면



- Start : Virtual Machine Power On
- Reset : Virtual Machine Reset
- Stop : Virtual Machine Power Off
- A, B, CD : 장치디스크를 시스템에 삽입, 제거한다.

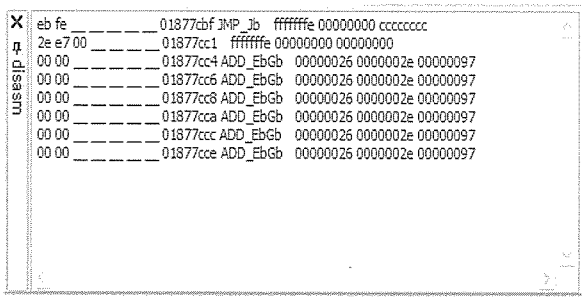
■ System Viewer

(1) CPU Decoding info & Control

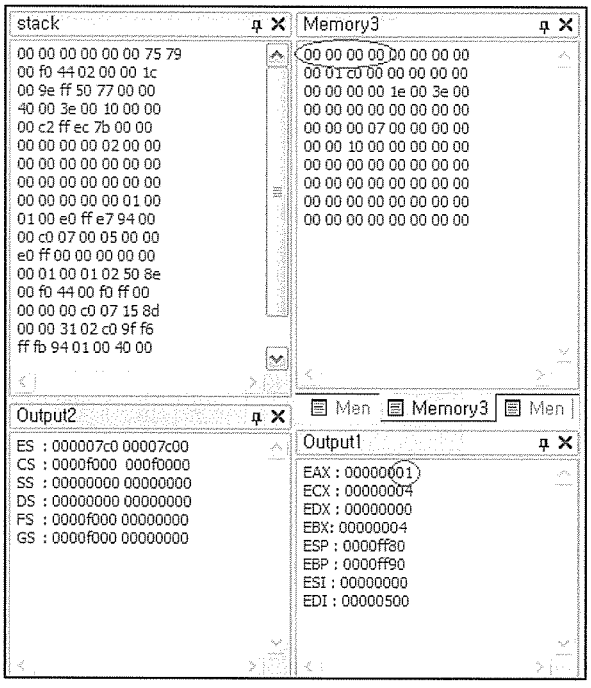


- Play : Virtual Machine 동작을 실행 시킨다.
- Force break : Virtual Machine 동작을 강제 멈추게 함.
- Step over : Virtual Machine에서 함수 단위로 명령어 실행
- Step into : Virtual Machine에서 다음 명령을 실행
- Memory : Virtual Machine의 메모리 값을 확인
- DisPlay : 명령어 실행중인 내용을 화면에 나타낼지를 체크한다.
- Break EIP : 명령어 실행 멈춤을 할 EIP 설정
- Fetch Count : 명령어 실행 멈춤을 할 Fetch Count 설정
- Break OP : 명령어 실행 멈춤을 할 Opcode 설정

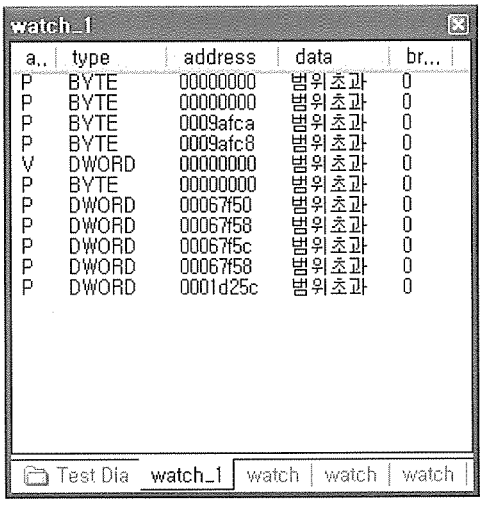
(2) System Information 화면



- 메모리에 있는 Opcode, Operand를 읽어서 명령어 실행을 DisAssembler 으로 나타냄

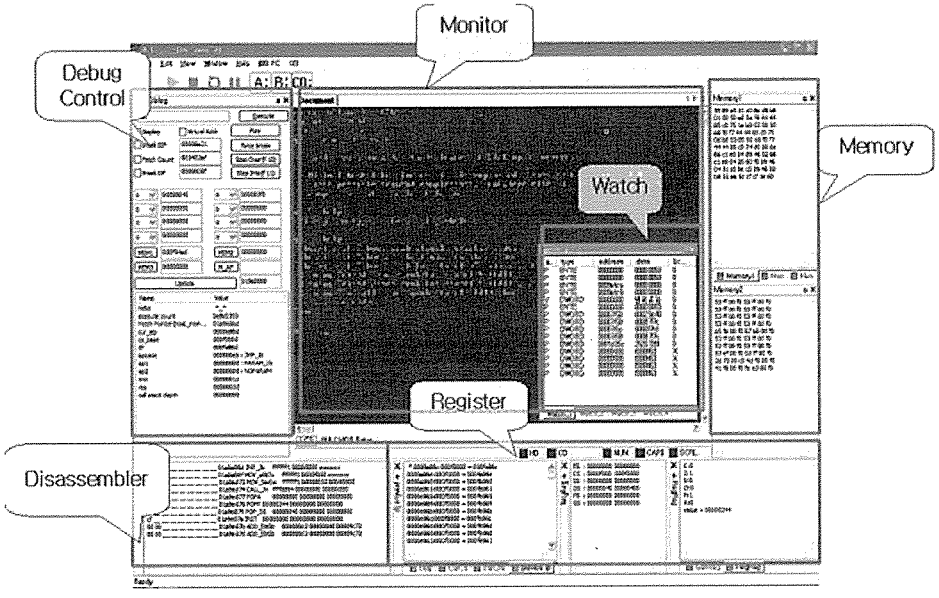


- Virtual Machine에서 동작 실행될 때 Stack, Register, Segment, Memory 상태를 화면에 나타냄



- 메모리에 있는 정보를 실시간으로 Display

(3) Debug 모드 화면



- System을 Emulate하고 Debugging을 통해 System의 값을 확인가능

4. 프로그램 개발 효과

이 프로그램은 System Developer를 대상으로 만들었다. 시스템 개발자에게 가장 필요한 것 중 하나가 프로그램을 개발하면서 현재 메모리 상황, 그리고 레지스터 상황 그리고 동작 시나리오 등을 보면서 Read / Write 통신 등의 작업을 하는 것을 구현하고, 이 같은 것을 확실히 이해하고 있지 않으면 시스템에 치명적인 오류가 생길 수 있기 때문에 이런 정보를 보여 주는 것은 매우 중요한 일이다. 하지만 Windows 환경에서 현재 응용 프로그램에서는 이런 기능이 가능하지만 커널레벨에서 이것을 할 수 있는 툴은 SoftIce나 WinDbg 등 이 있지만 사용방법이 어렵고, 시스템에 독립적으로 동작 할 수 있는 것이 아니기 때문에 많은 제약이 있다. 우리 프로그램은 가상으로 컴퓨터를 시뮬레이션 함으로써 정상적인 방법으로 접근 할 수 없는 것을 가상으로 시뮬레이션 하여 내부 시스템 값들을 확인할 수 있다는 측면에서 큰 의미가 있다.

이를 응용하여 Embedded 보드나 다른 하드웨어 장비를 시뮬레이션 한다면 비싼 장비를 직접 하지 않고서도 시스템을 개발 할 수 있고, 테스트하기도 더 쉬울 것이다.

앞으로 한국 IT산업은 새로운 Platform 대한 표준선도 및 국제적 입지 확보를 하는데 중점을 두어 발전하여야 할 것이다. 이에 따라 시스템개발에서 base가 되는 부분에 대한 쉬운 개발환경과 Debugging환경 및 System Information Viewer를 제공하여야 한다. 따라서 OS개발자나 Device Driver 및 Firmware 개발자를 포함한 우수 IT인력 양성 및 우리나라의 IT산업 발전에 큰 도움이 될 것으로 예상된다. 그리고 앞으로 더 많은 기능을 추가하고, 속도를 개선하여 더 많은 개발자들이 쉽고 편하게 이용할 수 있도록 할 것이다.

5. 사용 또는 개발언어, TOOL

Visual Studio. Net (Visual AssistX), GCC, AS86, NASM

6. 사용시스템

사용OS	Microsoft Windows 2000/XP
CPU	펜티엄4이상
모니터	15인치이상
메모리	128MB이상
FDD	1.44MB
HDD	10GB이상
VGA	SVGA 이상