

# MC68000 시스템을 위한 모니터 프로그램 설계.

이 범로, 정진현  
광운대학교, 제어계측공학과

The Design of Monitor Program for MC68000 Based System.

B. R. Lee, C. H. Chung  
Kwangwoon Univ., Dept. of Control and Instrumentation Eng.

## Abstract.

The monitor program, which controls overall operation of MC68000 based system and executes some commands and I/O operations, is a system program and consists of functions that executes these operations.

Monitor program can be easily applied to design a digital filter.

This paper approaches to the object, to which refer above, by considering algorithm of monitor program, which operates in M68000 based system, and concerning theory.

This monitor program operate by the serial communication between host IBM-PC and target MC68000 system, with MC6850 and MAX232C.

## 제 I 장 요약.

모니터 프로그램이라 함은 마이크로 프로세서를 기본으로 한 시스템의 총괄적인 동작들을 제어해 주고 입,출력에 관련된 일련의 작업들을 수행해 주는 시스템 프로그램의 일종으로 이러한 일련의 기능들을 수행해 주는 함수들로 구성된다. 모니터 프로그램에는 그것이 사용되는 시스템 특성이 집약되어 나타나게 되므로, 근래에 제어 시스템을 구성하는데 가장 많이 사용되고 있는 디지털 필터와 같은 컨트롤러를 설계하는 데 유용한 도구로써 사용될 수 있다. 본 논문에서는 MC68000을 기본으로 MC68000-Based System에서 동작하도록 설계된 모니터 프로그램의 알고리즘 및 관련 이론들을 고찰해 봄으로써 위에서 언급한 목적들에 접근해 보도록 한다. 본 모니터 프로그램은 MC6850 주변 기기와 MAX232C를 이용해서 IBM-PC와 시리얼 통신을 할 수 있도록 구

성되어 있고 HOST로는 IBM-PC를 사용한다.

## 제 II 장 모니터 프로그램.

### 1. 형식.

모니터 프로그램이란 일종의 시스템 소프트웨어로서 시스템을 사용자가 적절히 사용할 수 있도록 여러가지 편의를 제공하여 주며, 그것이 적용되고 있는 시스템의 상태를 사용자에게 알려주는 등 여러가지 역할을 해준다. 한마디로 모니터 프로그램은 하드웨어 시스템과 사용자를 연결해 주는 프로그램이라고 정의할 수 있다. 본 논문에서 설계된 모니터 프로그램은 모토롤라사의 MC68000을 CPU로 하는 68000-Based System에서 동작하도록 되어 있다. MC68000 어셈블리 언어로 작성되었고 모든 명령은 호스트 역할을 하는 IBM-PC에서 내려지게 되며, 이 명령은 한 바이트의 문자로서 비동기 직렬 포트로 RS-232C를 통해 시스템에 전해지게 된다. 이때 호스트에서 직렬 포트까지의 전송은 범용 통신 프로그램을 사용한다. 이 프로그램은 키보드를 두들기는 순간 해당되는 글자가 전송되게 되어 있다. 또한 시스템으로부터 전송되어 오는 글자를 화면에 나타내어 주는 기능을 포함하고 있어 별도의 디스플레이 프로그램을 필요로 하지 않는다. 직렬 포트를 오가는 것은 ASCII 코드이고 실제로 프로세서가 연산에 사용하는 코드는 이진 코드이기 때문에, 모니터 프로그램에는 이 두 코드간의 전환을 할 수 있는 루틴이 포함되어 있다. 또한 사용자가 모니터 프로그램, 시스템, 사용자 프로그램들의 동작 상태를 알 수 있게 하는 메시지를 호스트의 모니터로 나타내게 하였으며, 입력된 명령어를 직접 화면으로 확인할 수 있게 처리하였다. 특히 실행 코드 영역이 아닌 데이터 영역으로 실행이 넘어가거나, 버스 에러, 주소 에러

등을 원인으로 하여, MC68000 프로세서가 예외처리 에러를 일으켜서 종종 시스템 다운(System Down)을 일으키는 점들을 고려하여서, 그에 대한 처리를 해주는 서비스 루틴을 삽입하였다.

## 2. 구성.

모니터 프로그램이 가지는 전형적인 구성에 준해서 기본적으로 크게 6부분으로 나뉘어진다.

### 2-1. 메인 루틴(Main Routine):

사용자의 명령을 받아들이고 그 명령을 인식하여 명령에 해당되는 서브루틴을 불러주며, 잘못된 명령에 대해 에러 메시지를 출력시켜 준다. 이 루틴은 입출력 함수와 조건문으로 이루어진다.

### 2-2. 다운로드 루틴(Download Routine):

호스트에서 어셈블되어 생성된 S-RECORD로 이루어진 \*ABS 화일을 직렬 시리얼 포트로 받아들여서 명시된 주소에 적당히 로드시키는 루틴이 다운로드 서브루틴이다. 이 루틴은 ASCII 코드로 전송된 데이터들을 HEX 코드로 변환시켜 주는 루틴이 포함되어 있고 S-RECORD에 포함되어 있는 정보를 이용하여 정상적인 로드가 이루어졌는지에 대한 검토를 할 수 있는 루틴도 포함되어 있다.

### 2-3. 업로드 루틴(Upload Routine):

사용자 프로그램을 실행시킨 후 결과 값이나 중간 결과를 알아 볼 수 있도록 메모리나 레지스터의 내용을 호스트에 디스플레이 해주는 루틴을 업로드 루틴이라 한다. 또한 메모리의 내용을 보는 것을 메모리 덤프라 하는데 사용자가 지정해 주는 시작번지와 끝번지 사이의 메모리 블록을 덤프하는 데 있어서 HEX 코드를 ASCII 코드로 바꾸어 주는 루틴이 필요하다. 레지스터 덤프는 실행 중이나 실행 후의 레지스터 값을 호스트 화면으로 덤프할 수 있도록 되어 있다. 이 루틴은 사용자 프로그램의 결과치를 유지하기 위해서 트레이스(Trace)나 프로그램 실행 직후에 자동으로 실행되도록 설계되었다. 이 레지스터 덤프를 수동으로 모니터 프로그램 상에서 실행시키면 모니터 프로그램의 레지스터 값이 덤프된다.

### 2-4. 실행 및 트레이싱 루틴(Execution And Tracing Roution):

모니터 프로그램에서 사용자가 원하는 주소의 명령을 임의로 지정하여 실행시킬 수 있는 루틴을 실행 루틴이라 하며, 프로그램의 시작부분과 브레이크 포인트를 잡

아서 그 사이의 블록을 일괄적으로 실행시키며, 트레이싱 루틴은 원하는 시작 번지로부터 한 명령씩 순차적으로 실행시킨다. 이들 루틴은 주로 예외처리 프로세싱을 사용하여 유지 모드, 슈퍼바이저 모드, 트레이스 모드를 넘나들면서 동작하도록 되어 있다. 따라서 이들 루틴은 예외처리가 걸렸을 때 필요한 동작을 수행해 주는 서비스 루틴을 수반하게 된다. 이들은 단순한 실행 뿐만이 아니라 모든 수행을 끝난 후 제어를 모니터 프로그램 영역으로 되돌리는 역할도 하는데 바로 이 부분을 서비스 루틴에서 처리해 준다. 또한 레지스터 덤프도 이 서비스 루틴에서 하게 되어 있다.

### 2-5. 에러 처리 루틴(Error Manipulation Routine):

MC68000에는 몇가지 시스템 에러가 발생 하였을 때 예외처리가 걸리게 되어 각각의 예외처리에 주어진 벡터 어드레스에 저장된 서비스 루틴의 시작번지로 분기되어 에러를 처리해 주는 기능을 가지고 있다. 만약 이러한 에러 처리 루틴이 존재하지 않는다면 약간의 오동작으로도 시스템은 다운되어 버리고 말 것이다. 본 논문에서 설계된 모니터 프로그램에서는 이러한 에러 처리 서비스 루틴을 첨가하여, 좀더 능률적인 프로그래밍을 할 수 있도록 하였다. 첨가된 서비스 루틴은 버스 에러 처리 서비스 루틴, 의미 없는 코드가 저장되어 있는 주소를 액세스 했을 때 발생하는 에러 처리 루틴, 그리고 어드레스 에러 처리 루틴이다. 이러한 서비스 루틴은 에러 메시지를 사용자에게 보이고 시스템의 제어를 다시 모니터로 넘겨서 프로그램의 수정하거나 다시 다운로드 시키는 데 편의를 제공해 준다.

### 2-6. 라이브러리 루틴(Library Routine):

위의 여러 루틴들을 지원해 주는 루틴들이다. 한가지 예를 들자면 메모리에 입력되어 있는 글자들을 포인터를 사용하여 영 문자(Null Character)가 나타날 때까지 계속 비동기 직렬 포트로 내보내어 주는 서브루틴이 있는데 이것은 시스템의 메시지를 사용자에게 알릴 때 편리하다. 전반적으로 반복되어 사용되는 기능은 한 서브루틴으로 묶어서 모듈화하였고 사용자가 명령어에 익숙해 있지 않을 경우 도움말을 출력해주는 루틴도 첨가하였다.

## 3. 설계 과정.

본 논문에서 설계된 모니터 프로그램은 탑 다운(TOP·DOWN)방식에 의해서 설계되었다. 설계 과정이나 코딩 과정에서 가장 어려웠던 점은 바로 스택 포인터를 취급함에 있어서 모드를 정확히 고려하는 것이었다. 모니터 프로그램의 특성상 유지 모드와 슈퍼바이저 모드를 오가

야 하는 부분들이 나오게 되는데 이런 과정 중에서 스택 포인터가 두 개로 구성되어 있다는 사실을 명심하여야 한다. 또한 모니터 프로그램이 ROM 영역에 깔릴 프로그램이므로 버퍼, 카운터 변수 등은 레지스터로 주로 지정해서 코딩하게 되는데, 각종 서브루틴이 불리지고 서브루틴 안에서 또 다른 서브루틴들이 불리워지면서 이러한 레지스터들이 중첩되어져서 정확한 수행을 방해하는 경우도 있었다. 서브루틴을 취급하는 과정 중에 많이 사용되는 스택의 메모리 맵을 잘 고려해야 한다. 서브루틴 중에 스택을 사용하는 명령이 들어가 있는 경우에는 리턴 하기 전에 반드시 스택을 해제하여 주어야 한다. 레지스터 덤프의 경우 A7 레지스터를 PUSH해서 SP(Stack Pointer) 값을 출력하면 SSP(Supervisor Stack Pointer) 값이 출력되므로 반드시 USP(User Stack Pointer) 값을 따로 출력하는 과정을 거쳐야 하고 이때 USP는 어드레스 레지스터로만 MOVE된다. SR(Status Register) 값은 트레이스나 실행 서브루틴이 수행될 때 스택에 저장되는 SR 값을 RAM 영역에 저장하였다가 그 절대번지를 액세스 함으로써 정확한 값을 취하게 하였다.

### 제 III 장 결 론.

서론에서도 언급한 것처럼 모니터 프로그램은 그것이 사용되는 시스템의 하드웨어와 밀접한 관계가 있을 뿐 아니라, 디지털에 연관된 응용프로그램 개발에도 유용한 도구로서 쓰인다. 이러한 모니터 프로그램의 알고리즘은 프로세서를 이용하여 플랜트를 제어하는 디지털 콘트롤 시스템에서의 유저 인터페이스(User Interface)의 역할을 하는 I/O 부분에도 응용될 수 있으며, 한 모듈의 주변기기 및 시스템의 버스 인터페이스(Bus Interface)의 이상 유무를 프로세서 자신이 알아서 체크하는 기능이 내장되어 있으므로 시스템의 자기 고장 진단의 기능의 응용에도 사용될 수 있다. 이와같이 모니터 프로그램은 그 이름에서도 알 수 있듯이 시스템 프로그램의 일종이지만, 여

기에 이용된 알고리즘들은 위에서 말한 바와같이 여러 분야에 응용되어 사용될 수 있음을 밝혀둔다.

본 논문에서 설계된 모니터 프로그램은 처음 시스템에 전원이 인가 되면 자동적으로 모니터 영역으로 분기해 수퍼바이저 모드에서 동작을 하게 설계되어 있다.

이 모니터 프로그램의 가장 큰 특성은 무엇보다도 사용자 프로그램이나 사용자의 잘못된 번지 입력에도 시스템 다운을 막아주는 루틴에 있다. 그러므로 사용자는 에러가 발생함으로 사용자 프로그램의 어딘가에 이상이 있거나 잘못된 번지 입력을 한 것을 체크할 수 있다는 것이다.

본 논문의 모니터 프로그램 설계는 주변기기 인터페이스와 마이크로 프로세서의 응용기술 개발과 그의 소프트웨어 개발 기술도구로서 매우 유용하게 사용되어 질 수 있다.

### 제 IV 장 참 고 문 헌.

1. YU-CHENG LIU, "THE MC68000 MICROPROCESSOR FAMILY".(1991 By Prentice-Hall)
2. WILLIS J.TOMPKINS, JOHN G.WEBSTER, "INTERFACING SENSORS TO THE IBM® PC. (1988 By Prentice-Hall)
3. LANCE A.LEVENTHAL,DOUG HAWKINS, "68000 ASSEMBLY LANGUAGE PROGRAMMING 2ND EDITION". (1986 By McGraw-Hill)
4. 안수길,최승철,배명진, "M68000의 원리와 프로그램", (大英社, 1990).
5. "16 Bit MicroProcessor", (Ohm 社,1990).
6. 이범로 외 5인, "MC68000-Based System을 위한 모니터 프로그램 설계", (ICIS-TR-RAL93-1, 제어계측시스템연구소, 광운대학교,1993).