

---

# CF(CompactFlash)메모리 카드를 이용한 부트 시스템 구현에 관한 연구

## 1. CF메모리 카드를 이용한 부트 시스템 구현

이광철\* · 김영길\*\*

The Study of the Implementation of the Boot System Using CF(CompactFlash) memory card

### 1. Implementation of the Boot System Using CF memory card

Kwang-cheol Lee\* · Young-kil Kim\*\*

#### 요 약

본 논문은 CF 메모리카드를 이용한 부트 시스템을 제안하고 실제 구현에 관하여 연구한 것이다. 본 논문에서 제안하는 시스템은 고성능의 마이크로프로세서와 적은 양의 프로그램 메모리, CF 메모리카드를 기본으로 구성된다. 여기에 사용자 인터페이스를 위하여 LCD 모듈 및 터치 패널을 추가된다.

구현된 시스템은 대용량의 Flash 메모리 대신 CF 메모리카드와 DRAM을 이용하여 시스템 단가를 낮출 수 있었으며, 시스템 프로그램이 DRAM에서 실행되기 때문에 시스템 성능이 향상된다.

#### ABSTRACT

In this paper we propose the boot system using CF memory card and study the system implementation method. The system that is proposed in this paper basically consist of high performance microprocessor, small amount of program memory and CF memory card. And added LCD module and touch panel for the user interface.

This system use the CF memory card and DRAM instead of the Flash memory, so it can reduce the system cost. And system performance is increased because of the system program running in the DRAM.

#### 키워드

CF 메모리카드, boot loader

#### 1. 서 론

최근 임베디드 시스템에 대한 관심이 높아지고 있다. 이와 더불어 시장에서는 더 많은 기능을 내장한 임베디드 시스템을 요구하고 있다. 고성능, 고사양의 임베디드 시스템을 구현하기 위해서 고성능의 프로세서와 많은 프로그램 메모리를 필요

로 하고 있다. 고성능의 프로세서는 이미 여러 칩 제조사들이 발표하고 있다. 대표적으로 인텔사, 삼성사 등에서는 저전력 소모, 고기능을 자랑하는 ARM Core를 내장하여 여러 가지 어플리케이션 칩들을 출시하고 있다. 하지만 고용량의 프로그램 메모리를 구성하는 방법에 있어서는 현재 여러 가

---

\* 아주대학교 의용공학과

\*\* 아주대학교 전자공학과

접수일자 : 2003. 9. 17

지 문제점이 있다. 고용량의 프로그램 메모리를 구성하는 대표적인 방법은 인텔사의 StrataFlash 메모리를 이용하는 방법이 있다. 하지만 이 방법은 대용량 메모리 구성에 한계가 있으며 PCB (Printed Circuit Board) 공간 확보의 어려움이 있다. 또한 프로그램 메모리 확장에 따른 시스템 전체의 단가가 상당히 높아진다는 문제가 있다.

이런 문제를 해결하기 위해서는 값이 싸고 쉽게 확장이 가능한 HDD(Hard Disk Drive)나 대용량 메모리카드 등과 같은 대용량 저장 매체를 이용하여, 프로그램 메모리를 더 많이 확보하는 방안을 강구해야 한다.

본 논문에서는 CF(CompactFlash) 메모리카드를 이용하여 시스템 프로그램 메모리와 같은 효과를 낼 수 있는 시스템 설계 방안을 제안하고 이를 구현한다.

## II. 기존의 시스템

프로그램 메모리를 구성하는 방법은 ROM(Read Only Memory)을 사용하는 방법 또는 Flash 메모리를 이용하는 방법이 있다. 여기서 Flash 메모리라고 칭하는 것은 일반적인 ROM 또는 RAM(Random Access Memory)과 동일하게 마이크로프로세서의 주소 영역에 메모리 공간이 일대일로 대응되도록 구성되어 있는 NOR 형태의 Flash 메모리를 말한다.

### II-1. ROM을 사용하는 임베디드시스템

일반적으로 마이크로프로세서를 이용한 시스템을 구성하는 방법은 그림 1과 같이 마이크로프로세서와 이 프로세서를 구동하기 위한 프로그램 저장용 프로그램 메모리와 입력을 설정하거나 출력을 표출하는 입출력 장치와 추가로 필요한 경우 RAM이나 HDD 또는 CF 메모리카드 등의 부가 메모리를 더한다. 이때 어떠한 형태이든 필수 불가결한 것은 마이크로프로세서, 프로그램 메모리, 입출력 장치이다. 물론 시스템을 구동하는 전원은 따로 인가된다고 가정한다.

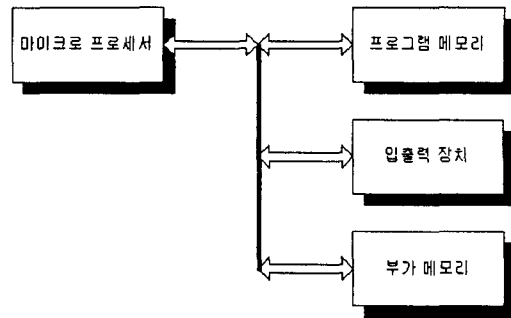


그림 4. 마이크로프로세서를 이용한 일반적인 시스템  
Fig. 1 General system using the microprocessor

ROM을 이용하는 방법은 시스템 제작 시점에 프로그램을 한번 입력하면 수정이 불가능한 형태이다. 따라서 이 경우 시스템 프로그램을 수정하려면 새로운 ROM에 수정된 프로그램을 쓴 후 ROM을 교체해야 한다.

### II-2. Flash메모리를 사용하는 임베디드시스템

Flash 메모리를 이용하는 방법도 ROM과 동일하게 메모리를 교체하는 방법으로 프로그램을 수정할 수 있다. 하지만 이렇게 사용하면 Flash 메모리의 장점인 시스템에 장착된 상태(In-Circuit)에서 Flash 메모리의 내용을 수정하는 기능을 사용하지 않는 것이다.

Flash 메모리는 일반적으로 널리 사용되는 ROM 또는 RAM과 동일한 메모리 영역 액세스 방법을 제공하는 NOR 형태의 Flash 메모리와 전용의 데이터 액세스 방법을 이용해야 하는 NAND나 AND 형태의 Flash 메모리가 있다.

아래 그림 2는 Intel사의 StrataFlash 메모리의 핀 배열을 나타낸다. 이 핀 배열을 보면 알 수 있듯이 이 칩은 ROM과 동일하게 /CE 핀을 LOW로 만들어 준 상태에서 Axx 핀에 원하는 주소를 설정하고 /OE를 LOW로 만들어 주면 해당 주소의 데이터가 Dxx 핀에 나오는 형태의 Flash 메모리이다.

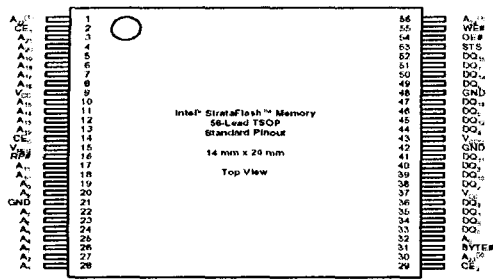


그림 2. Intel사의 StrataFlash 메모리의 핀 배열  
Fig. 2 Pin Assignment of the Intel's StrataFlash Memory

하지만 일반적으로 대용량의 Flash 메모리는 Samsung사나 Toshiba사에서 생산하는 NAND 형태나 Hitachi사에서 생산하는 AND 형태의 Flash 메모리를 사용한다. 이들 Flash 메모리 중 일반적으로 많이 사용되는 것은 NAND 형태의 Flash 메모리이다. 그림 3은 Samsung사에서 생산하는 NAND 형태의 Flash 메모리의 핀 배열과 각 핀의 기능을 간단하게 정리한 표이다. 이 그림을 보면 알 수 있지만 이 메모리 칩은 별도의 주소 설정용 핀이 존재하지 않는다. 즉 메모리의 특정 영역의 데이터를 읽거나 쓰기 위해서는 칩이 정하는 특정 프로토콜을 이용해야 한다. 즉 NAND 형태의 Flash 메모리는 ROM과 같이 주소를 설정하고 해당 주소의 데이터가 바로 출력되는 형태로 구성되어 있지 않다. AND 형태의 Flash 메모리도 NAND 형태의 Flash 메모리와 유사한 프로토콜이 있으며 이를 통해서 데이터의 액세스

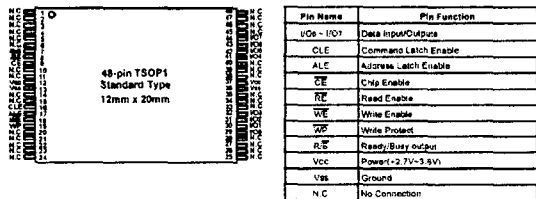


그림 3. NAND Flash 메모리의 핀 배열과 기능  
Fig. 3 NAND Flash Memory Pin Assignment and Functionality

가 가능하다. 따라서 일반적인 대용량의 Flash 메모리는 ROM 형태의 직접 액세스 방식으로 칩 내부에 저장되어 있는 프로그램 또는 데이터를 읽을

수 없으므로 마이크로프로세서에서 직접 Instruction Fetch를 수행할 수 없다. 따라서 NAND나 AND 형태의 Flash 메모리를 시스템 프로그램 저장 용도로 사용하려면 해당 프로그램을 마이크로 프로세서가 직접 Instruction Fetch가 가능한 RAM 영역으로 로드하고 이를 실행시키는 방법을 사용해야 한다. 즉 NAND나 AND 형태의 Flash 메모리를 프로그램 메모리로 사용하기 위해서는 별도의 프로그램 로더 또는 부트로더를 구현해 주어야 한다.

### III. CF 메모리 카드

CF 메모리카드 관련 하드웨어 및 소프트웨어 인터페이스는 그림 4와 같다. CF 메모리카드의 물리적 및 전기적 특성 표준에 맞추어 하드웨어를 제어하기 위하여 카드 인식과 전원 제어를 해주기 위한 블록이 필요하다. CF 카드의 표준에는 CIS(Card Information Structure)의 내용에 의하여 특정 CF 카드의 형태와 카드가 제공하는 기능을 알아낼 수 있는 방법을 제공한다.

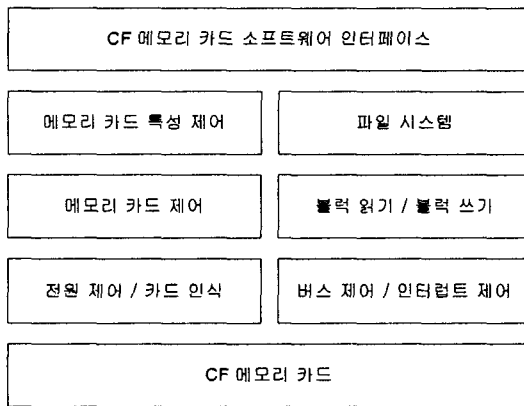


그림 7. CF 메모리카드 인터페이스 블록도  
Fig. 4 CF memory card interface block diagram

예를 들면 우리가 원하는 CF 카드는 ATA (Advanced Technology Attachment) 형태로 액세스가 가능한 메모리카드인데 CF 카드 중에는 네트워크 어댑터나 RAM과 같은 다른 기능을 수

행하는 디바이스들이 많이 존재한다. 이런 경우 원하는 CF 카드임을 인식하기 위해서는 CF 카드의 표준 CIS를 읽어 내어 분석하는 과정이 필요하다. 이와 함께 시스템 버스와 CF 메모리카드의 버스간의 충돌을 방지하고 CF 메모리카드에서 발생하는 인터럽트를 제어하여 CF 메모리카드의 내용을 읽어 내고 필요한 내용을 써넣기 위한 블록이 필요하다. 이 메모리카드 액세스 블록은 CF 메모리카드의 주기능인 데이터 저장 장치로 사용하는 측면에서 가장 중요한 블록이라고 하겠다.

앞에서 기술한 과정을 통하여 CF 메모리카드의 CIS를 읽어내어 해당 정보에 맞게 CF 메모리카드 인터페이스를 재설정하고, CF 메모리카드에 의한 인터럽트 제어 부분을 초기화 한 후 우리가 관심 있는 CF 메모리카드 내의 사용자 데이터 영역에 있는 실제 데이터를 읽어 낸다. 읽어낸 내용을 조사하여 원하는 파일 시스템인지를 확인한다. 이 경우 Microsoft사의 MS-DOS 시스템 및 Windows 시스템을 비롯하여 여러 시스템의 표준으로 쓰이는 FAT12/FAT16/FAT32 (File Allocation Table 12/16/32)에 해당하는 부분을 확인한다. 파일 시스템 블록은 FAT12/16/32 이외의 다른 파일 시스템을 고려하여 원하지 않는 파일 시스템 처리를 수행한다. CF 메모리카드 소프트웨어 인터페이스는 파일 시스템 종류에 상관없이 파일을 읽고 쓰기 위한 사용자 인터페이스이다. 파일 시스템 블록을 구현할 때, 파일의 내용을 시스템에서 수정할 필요가 없는 경우에는 읽기 전용 파일 시스템을 구현한다. 일반적으로 파일 시스템 구현 시 쓰기를 고려하는 경우 기능 구현이 더욱 어렵기 때문이다.

#### IV. 제안하는 시스템

##### IV-1. 구현 방법

본 논문에서 제안하는 시스템은 그림 5와 같이 CF 메모리카드를 이용하여 대용량 저장 매체를 구성하며, 이 CF 메모리카드를 시스템 프로그램을 저장할 목적으로 사용한다. 물론 시스템 프로그램 이외의 일반 데이터 저장 용도로도 사용된

다.

일반적으로 널리 사용되는 검증된 시스템을 이용하여 CF 메모리카드에 시스템 프로그램을 저장하는 방법을 제공한다.

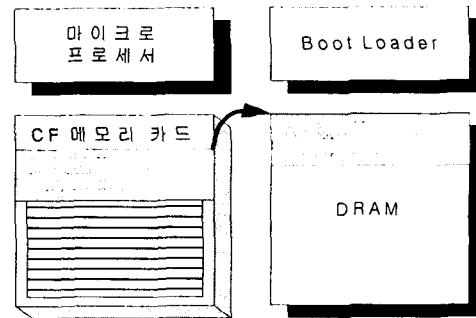


그림 8. 제안하는 시스템의 구현 방법  
Fig. 5 Implementation method

또한 CF 메모리카드에 저장된 시스템 프로그램 이미지를 시스템 부팅 시점에 시스템 내의 RAM으로 로드하는 방법과 로드된 시스템 프로그램을 마이크로프로세서가 구동하는 방법을 제공하기 위한 부트로더를 구성한다.

##### IV-2. 시스템 구성

그림 6은 구현한 시스템의 블록도를 나타낸다. 시스템 동작의 중심인 StrongARM 마이크로프로세서와 이를 제어하기 위한 프로그램을 내장하고 있는 NOR 형태의 프로그램 메모리는 마이크로프로세서를 이용한 시스템의 기본 사양이다. 시스템 구성 예로 Microsoft사의 WindowsCE OS(Operating System)를 기반으로 하는 CNS (Car Navigation System)를 구현하기 위해서 StrongARM 프로세서와 대용량의 메모리를 구성한다.

프로그램 메모리는 오류의 발생 등에 의한 프로그램 업데이트를 위하여 내용을 수정하기 용이한 Flash 메모리를 사용하는 것이 좋다. 하지만 부트로더의 수정이 필요없는 경우 ROM을 사용해도 된다. 프로그램 메모리의 용량은 시스템 프로그램의 용량에 따라서 다르겠지만 시스템 프로그램 용량이 많다는 가정을 하면 이에 맞춰 Flash 메모리 또는 ROM의 용량이 커져야 한다. 보통 PDA (Personal Data Assistant) 등의 목적으로

사용하는 경우 기본적으로 16MB ~ 32MB 정도 된다. 본 논문에서 제안하는 시스템은 CF 메모리 카드에 대부분의 시스템 프로그램을 저장하고 Flash 메모리 또는 ROM은 128kB 정도의 용량으로도 충분히 시스템을 운용할 수 있다. 실제로 구현된 시스템은 SST39VF100이라는 NOR 형태의 Flash 메모리를 사용하였다.

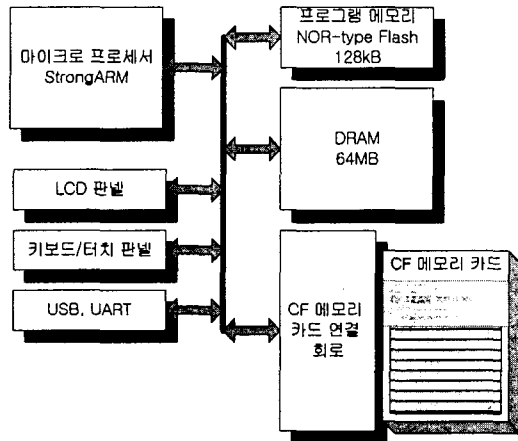


그림 9. 구현한 시스템 블록도  
Fig. 6 Implemented system block diagram

DRAM은 SRAM에 비해 동일 메모리 용량을 구성할 경우 구성비용이 적게 소요된다. 특히 데이터 처리가 많이 필요한 시스템에 사용된다. 일반적으로 시스템 내에서 DRAM의 용량은 실행되는 어플리케이션 프로그램에서 요구하는 용량에 따라 정해진다. 하지만 본 논문에서 제안하는 시스템의 경우 DRAM의 용량은 어플리케이션 프로그램에서 요구하는 용량에 시스템 프로그램의 크기만큼의 용량이 더 필요하다.

CF 메모리카드 인터페이스 부는 CF 메모리카드를 쉽게 꽂고 뺄 수 있도록 소켓을 연결하고 프로그램 메모리 및 DRAM 등과 함께 마이크로프로세서의 데이터 버스를 공용으로 사용하기 위하여 충돌을 방지하는 회로와 CF 메모리카드의 전원을 제어하기 위한 회로가 필요하다. 물론 CF 메모리카드의 유무를 판별할 수 있는 회로도 구성되어 있어야 한다. 이상의 CF 메모리카드 인터페이스 부는 CF 메모리카드를 동작 중에도 동적으로

착탈할 수 있도록 구현하여야 하기 때문에 반드시 구현되어야 한다.

이외에 출력 화면 구성을 위한 LCD(Liquid Crystal Display) 패널, 사용자 입력을 받아들이기 위한 키보드/터치 패널, 시스템 외부와의 데이터 통신을 위한 USB(Universal Serial Bus), UART(Universal Asynchronous Receiver Transmitter) 인터페이스를 추가한다. 물론 이 부분은 시스템의 필요에 의하여 추가 또는 삭제될 수 있는 부분이다.

### V. 실험 및 고찰

본 논문에서 제안한 구성을 적용하여 실제 구현된 시스템은 그림 7의 사진에서 확인할 수 있다. 터치 패널이 달려 있는 STN LCD가 시스템의 중심에 있고 위쪽에 CF 메모리카드를 배치하고 오른쪽 위에 음을 재생하기 위한 스피커를 연결하였다.

그림 8은 구현된 시스템에 전원을 연결하고 CF 메모리카드에 미리 써놓은 WindowsCE 부팅 이미지 파일인 NK.bin을 이용하여 정상적으로 부팅이 되고 작동하고 있는 화면이다. LCD 화면에 WindowsCE 바탕화면과 아이콘 및 TaskBar가 보인다.

본 논문에서 제안하는 시스템 구조를 바탕으로 구현한 시스템은 StrataFlash 메모리를 이용하여 프로그램 메모리를 구현한 시스템에 비해 부팅 속도가 느리다. 약 10MB 정도의 시스템 프로그램을 로드하는 데에 약 10초 정도 걸린다. StrataFlash 메모리를 이용한 시스템의 경우 2초 이내로 부팅이 된다. 이렇게 부팅 시간에서 차이가 나는 주원인은 CF 메모리카드에 들어있는 프로그램을 DRAM으로 로드하는 데에 시간이 많이 걸리기 때문이다.

WindowsCE의 응용 프로그램 중 CNS 어플리케이션을 실행시켜보는 것은 그림 9의 화면에서 확인할 수 있다. 왼쪽 그림이 미리 설정해 놓은 주행 경로를 읽어온 화면이고 오른쪽 그림이 모의 주행을 실행한 화면이다. 이때 스피커에서는 상황

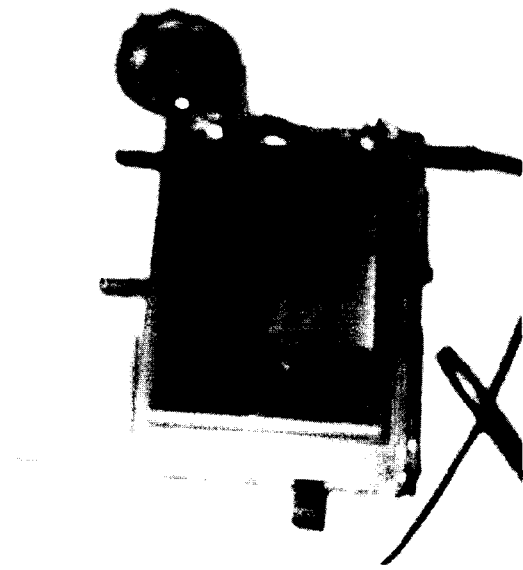


그림 8. WindowsCE를 실행시킨 화면  
Fig. 8 WindowsCE Execution

에 맞는 주행안내를 들을 수 있다. 이 어플리케이션은 많은 양의 DRAM을 할당하여 고속 연산을 수행하는 어플리케이션이다. 이 어플리케이션의 실행 성능을 통하여 구현한 시스템의 전체 성능을 가늠해볼 수 있다.

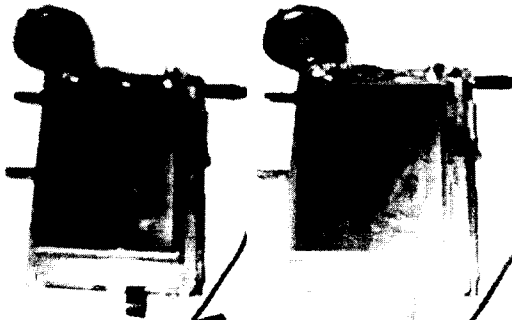


그림 12. CNS 어플리케이션 실행 모습  
Fig. 9 CNS Application Execution

## VI. 결 론

본 시스템은 CF 메모리카드를 사용해서 부트

Flash 메모리의 용량을 줄이고 시스템 부트 이미지를 CF 메모리카드에 옮겨놓고 시스템 시작시에 이를 로드하는 기능을 갖는 시스템이다.

구현한 시스템의 장점으로는 기존의 대용량 Flash 메모리 또는 ROM으로 구성할 수밖에 없었던 대용량 프로그램 메모리 공간을 적은 양의 Flash 메모리 또는 ROM과 CF 메모리카드를 이용하는 시스템으로 설계하여 시스템 단가를 낮출 수 있다는 것이다.

시스템 동작 성능 면에서는 프로그램을 DRAM으로 복사하여 DRAM에서 프로그램이 수행되기 때문에 기존의 Flash 메모리에서 수행되는 구조에 비해서 훨씬 높은 프로그램 동작 성능을 발휘한다. 일반적으로 DRAM은 버스트 모드로 액세스가 가능하기 때문에 많은 성능 향상을 확인할 수 있다.

향후 연구방향으로는 CF 메모리 카드 이외의 대용량 저장 장치도 이용할 수 있는 일반적인 방법을 연구해야 할 것이다.

## 참고 문헌

- [1] Building Powerful Platforms with WindowsCE, James Y. Wilson, Aspi Havewala, 2001
- [2] CompactFlash™ Memory Card Product Manual, SanDisk, 2001
- [3] Microsoft Windows CE 3.0 Software Developer Documentation, Microsoft Corporation, 2000
- [4] PCMCIA PRIMER, Larry Levine, 1995
- [5] The PCMCIA Developer's Guide Second Edition, Michael T. Mori, W. Dean Welder, Sycard Technology, 1995

저자 소개



**이광철(Kwang-cheol Lee)**

2001년 8월 아주대학교 전자공학과 졸업(공학사)

2003년 8월 아주대학교 산업대학원 정보전자공학과 졸업(공학석사)

2003년 9월 ~ 현재 아주대학교 대학원 의용공학과 박사과정

※ 관심분야 : 리얼타임 시스템, 임베디드시스템 설계, 네트워크 시스템



**김영길(Young-kil Kim)**

1978년 고려대학교 전자공학과 졸업(공학사)

1980년 한국 과학원 산업전자공학과 졸업(공학석사)

1984년 ENST(France) 졸업(공학박사)

1978년 제 12회 기술고시 통신직: 체신부 정책국 전자 통신 전담반 통신기과

1978년 9월 ~ 현재 아주대학교 전자공학과 정교수

※ 관심분야 : 초음파 의료시스템, 의료용 통신 시스템, 선박 전자