

Embedded Systems 소개

임베디드 시스템

- 하드웨어와 소프트웨어가 조합되어 특정한 목적을 수행하는 시스템
- 특정한 기능을 수행하도록 마이크로 프로세서와 입출력 장치를 내장하며, 이를 제어하기 위한 프로그램이 내장되어 있는 우리의 일상 생활에서 사용되는 각종 전자기기, 가전제품, 제어장치 등

임베디드 시스템 응용분야

- 정보 가전 : 세탁기, 오디오, 인터넷 냉장고, HDTV 등
- 제어분야 : 공장자동화, 가정자동화, 로봇 제어, 공정제어 등
- 정보 단말 : 핸드폰, PDA, 스마트 폰, 네비게이션, MP3, PMP, DivX 플레이어, 디지털 카메라 등
- 네트워크기기 : 교환기, Router, 공유기, 홈 게이트웨이 등
- 게임기기 : 가정용 게임기(PS2, XBox), 지능형 장난감 등
- 항공/군용 : 비행기, 우주선, 로켓, 야전 이동단말(GPS, GIS)
- 물류/금융 : ATM, RFID, 물류단말, 영업단말 등
- 차량/교통 : 자동차, ITS 등
- 사무, 의료 : 전화기, 프린터, Heart pacer, 수술로봇, 증강현실장비

정보단말기

• 정보단말기

- 단순한 통화 중심의 이동 전화기에서 각종 정보검색, 오락, 메시징 등의 복합 기능이 수행되는 디지털 정보단말기기로 발전
- 단말기기 각각의 기능에 맞는 마이크로프로세서, 메모리, 운영체제, 응용 프로그램 등으로 구성
- 앞으로는 다양한 단말기기가 하나의 기기로 통합될 것으로 예상됨
- 핸드폰, PDA, 스마트 폰, MP3 플레이어, PMP, 게임기기 등



게임기

- 고성능 프로세서 탑재
- 마이크로소프트의 Xbox
- 소니의 playstation 2
- 닌텐도 게임보이 어드밴스
(nintendo gameboy advance)
 - 32-Bit ARM 프로세서
 - 2.9인치 TFT 스크린
 - 32,768 색상을 지원하는 휴대형 게임 장치



물류/금융/사무용기기

- 물류/금융
 - 물류 : POS 단말기
 - 금융 : 자동 현금 입출금기 혹은 ATM 단말기
- 사무용기기
 - 프린터, 스캐너, 팩스, 복사기, 이들의 기능을 하나로 모은 복합기 등



공장자동화

- 공장자동화 : FA (Factory Automation)

- 특정 기계나 장비를 통해 생산 과정을 자동적으로 관리하는 시스템
- 센서와 제어 시스템, 로봇 등으로 구성하여 무인시스템을 구축
- 공장 자동화 및 로봇은 실시간 시스템과 임베디드 시스템 발전의 원동력
- 생산성증대: 인건비감소, 오류감소, 품질의 균일화, 생산기간단축
- 로봇, conveyor belt

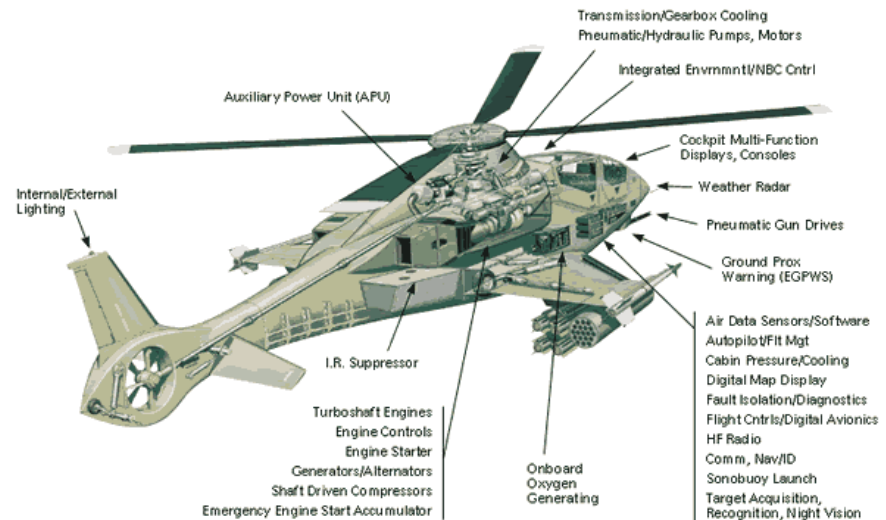


우주/항공

- 항공기
 - 보통 수 백 개의 프로세서 탑재
- 우주왕복선
 - Pathfinder -실시간 운영체제인 VxWorks가 탑재된 것으로 유명
 - 대표적인 실시간 시스템의 하나
 - 영상처리, 통신 등 모든 처리기능을 복합적으로 가짐

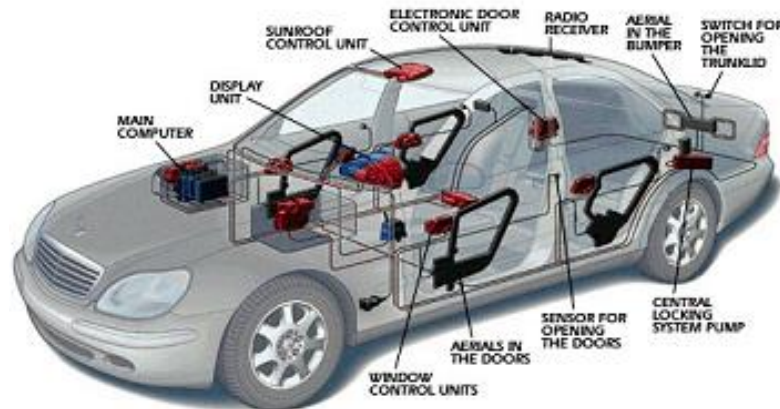


*NASA Pathfinder
(mission to MAR 1997)*



- 교통

- 자동차의 엔진 및 각종 제어 시스템, 무인 자동화 시스템
- 지능형 교통시스템(ITS : Intelligent Transport Systems) 등



지능형 장난감

- 지능형 장난감

- 단순한 장난감의 형태에서 지능성을 갖는 형태로 변화



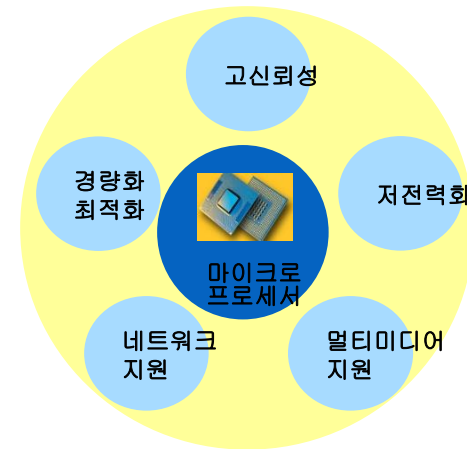
통신기기

- 디지털 교환기, PABX (private automatic branch exchange) 등의 음성 서비스 통신 기기
- 라우터, 게이트웨이, 공유기 등의 유무선 데이터 통신 장비
- Set-top box



임베디드 시스템의 특징

- 마이크로 프로세서/컨트롤러를 비롯한 하드웨어와 소프트웨어를 내장 (embedded)하여 특정한 기능을 수행
- 수행하는 기능은 미리 정해 진다.
- 소형, 경량, 저전력
- 가격에 민감하다
- 안정성이 뛰어나야 한다.
- Real-time 기능을 필요로 하는 시스템이 많다.

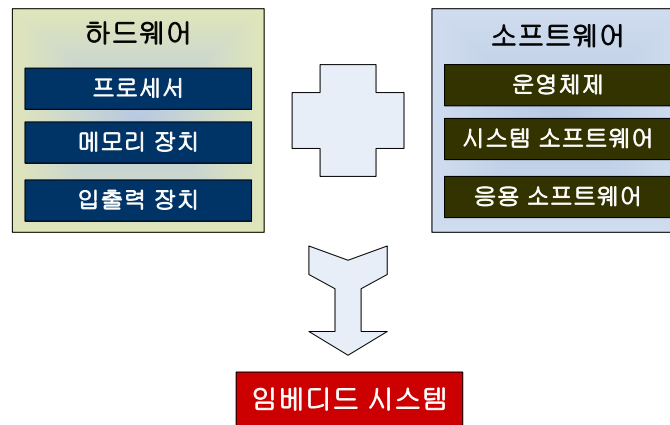


리얼타임 시스템

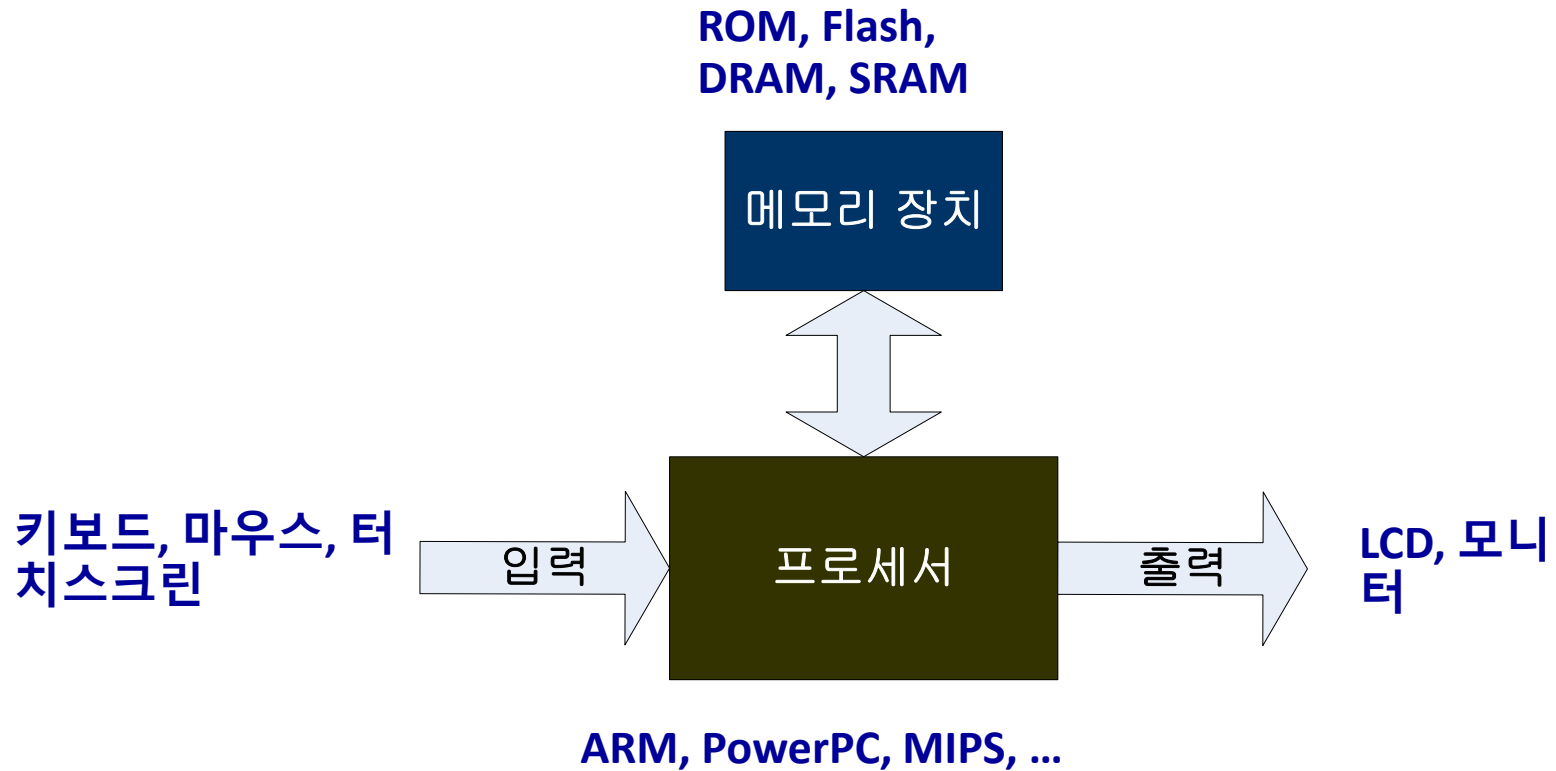
- 주어진 입력 조건을 주어진 시간 내에 처리하는 시스템
- RealTime 시스템의 종류
 - Hard Realtime 시스템
 - 리얼타임이 보장되지 않으면 시스템에 치명적인 오류를 유발
 - 대부분의 제어용 기기
 - 원자력 발전소 제어, 화재 발생 검출 시스템, 항공기, 우주 왕복선, 자동차, 미사일 제어 등
 - Soft Realtime 시스템
 - 주어진 시간 내에 결과를 출력하지 않아도 시스템 전반에 큰 영향이 없는 시스템
 - 예) 네트워크 장비

임베디드 시스템의 구성

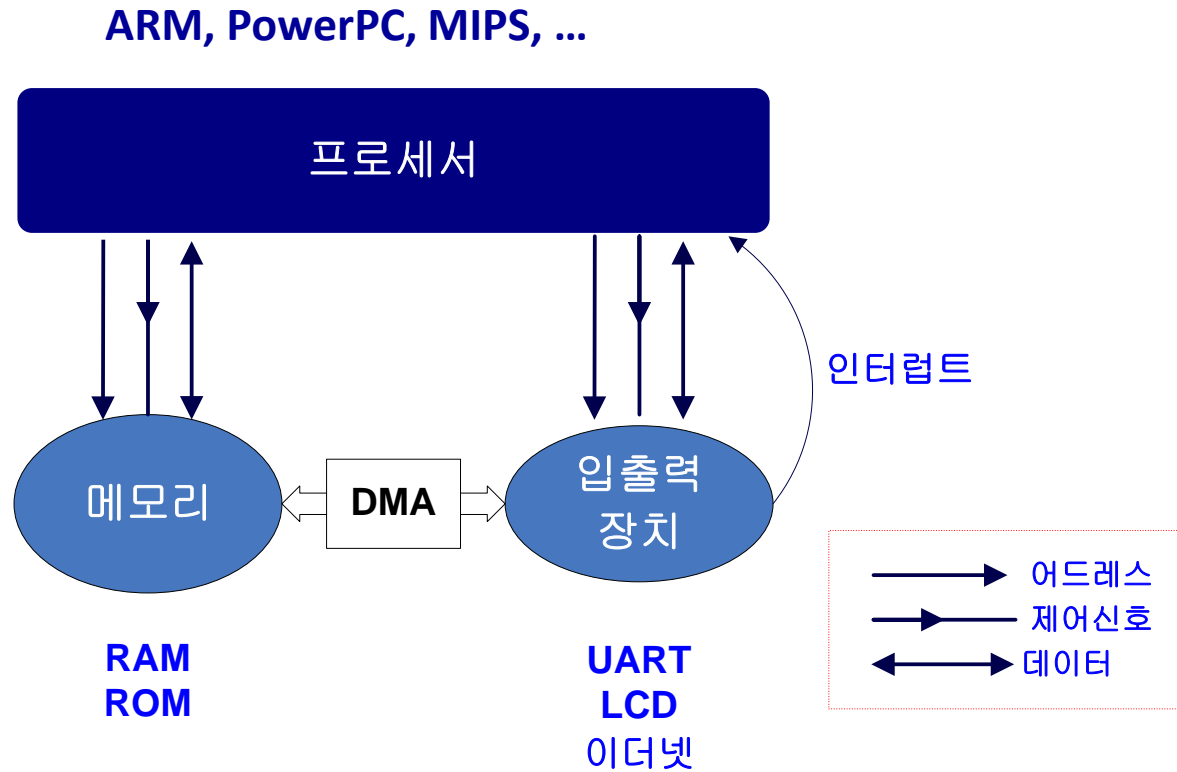
- 하드웨어
 - 프로세서(컨트롤러), 메모리 장치(ROM, RAM), 입출력 장치(네트워크 장치, 센서, 구동기 등)
- 소프트웨어
 - 운영체제(OS), 시스템 s/w, 응용 s/w



하드웨어 구조



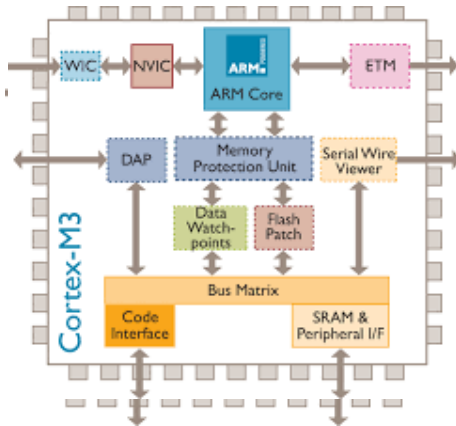
하드웨어 동작



마이크로프로세서 (Microprocessor)

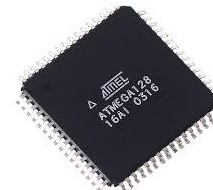
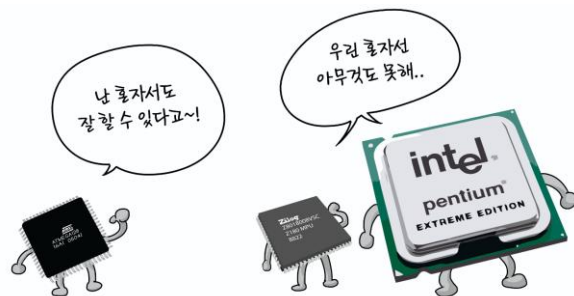
• 정의

- An integrated circuit that contains all the functions of a central processing unit of a computer. (from Wikipedia)
 - 초소형 연산 처리 장치
 - CPU (Central processing unit, 중앙 처리 장치)
 - MPU (Microprocessor Unit)



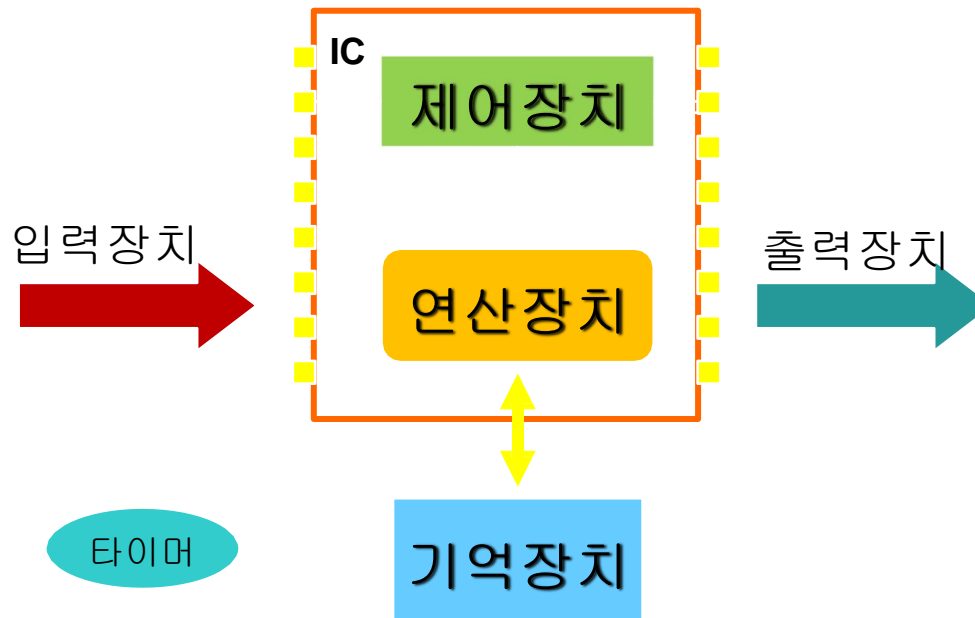
Microprocessor vs. Microcontroller

- Microprocessor
 - MPU (Micro Processor Unit)
 - 컴퓨터의 CPU를 한 개의 chip에 집적한 반도체 소자
- Microcontroller (= Embedded Processor = 내장형 프로세서)
 - MCU (Micro Controller Unit)
 - 별도의 메모리와 입출력 장치를 포함하여 만들어 특수목적의 제어, 통신, 가전제품 등에 사용되는 프로세서
 - MCU 그 자체가 하나의 작은 메인보드+주변장치들의 모임
 - ex) Microchip (Atmel) Atmega128, Intel 8051



Microprocessor vs. Microcontroller

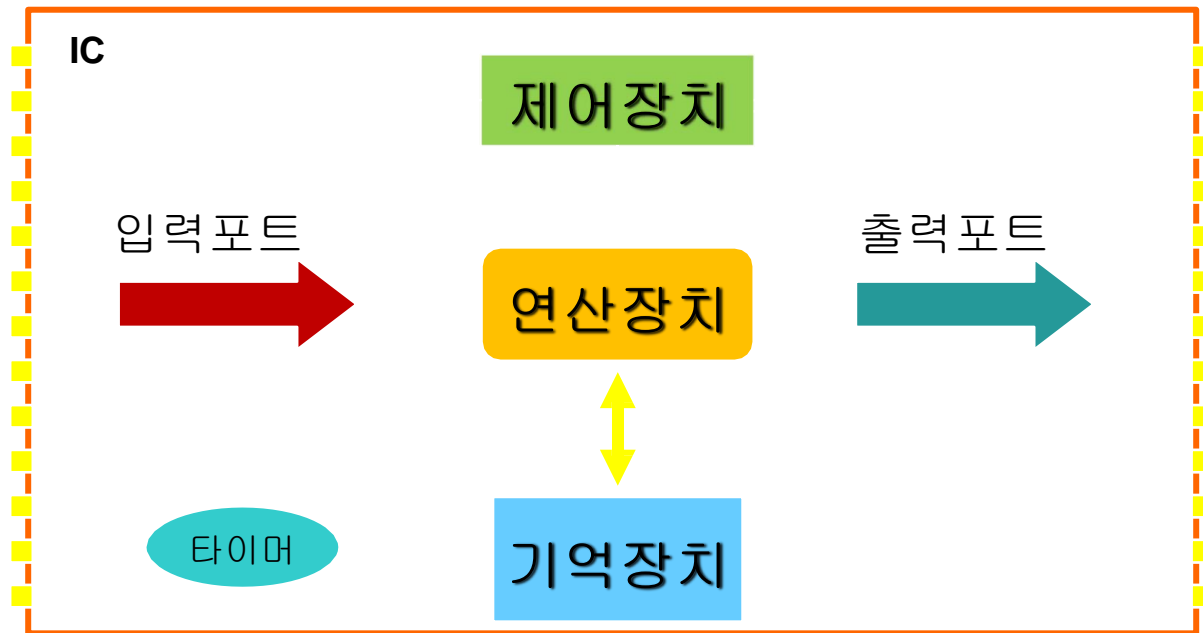
- Microprocessor



Ref) 창원대 조철우 교수님 자료

Microprocessor vs. Microcontroller

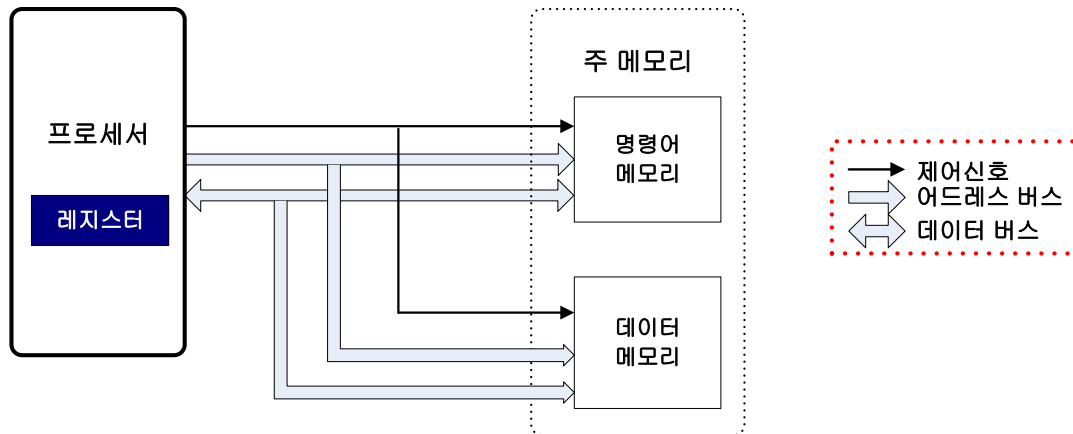
- Microcontroller



Ref) 창원대 조철우 교수님 자료

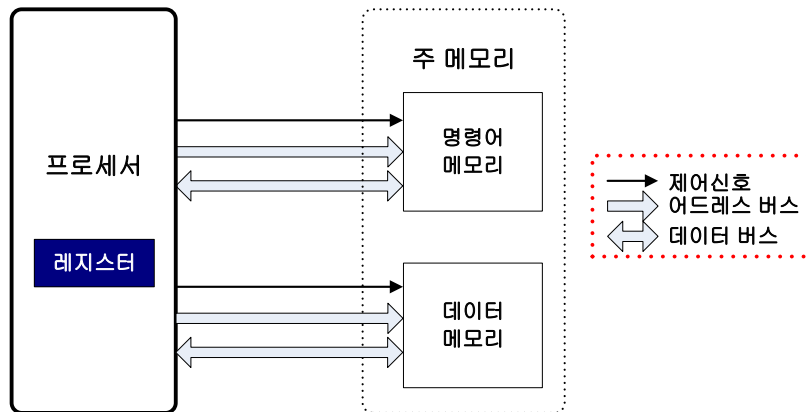
폰 노이만 아키텍처

- 폰 노이만(Von-Neumann) 아키텍처
 - 명령어와 데이터를 위한 메모리 인터페이스가 하나이다.
 - 명령어를 읽을 때 데이터를 읽거나 쓸 수 없다.
 - IBM 계열 PC(개인용 PC), ARM7 등



하버드 아키텍처

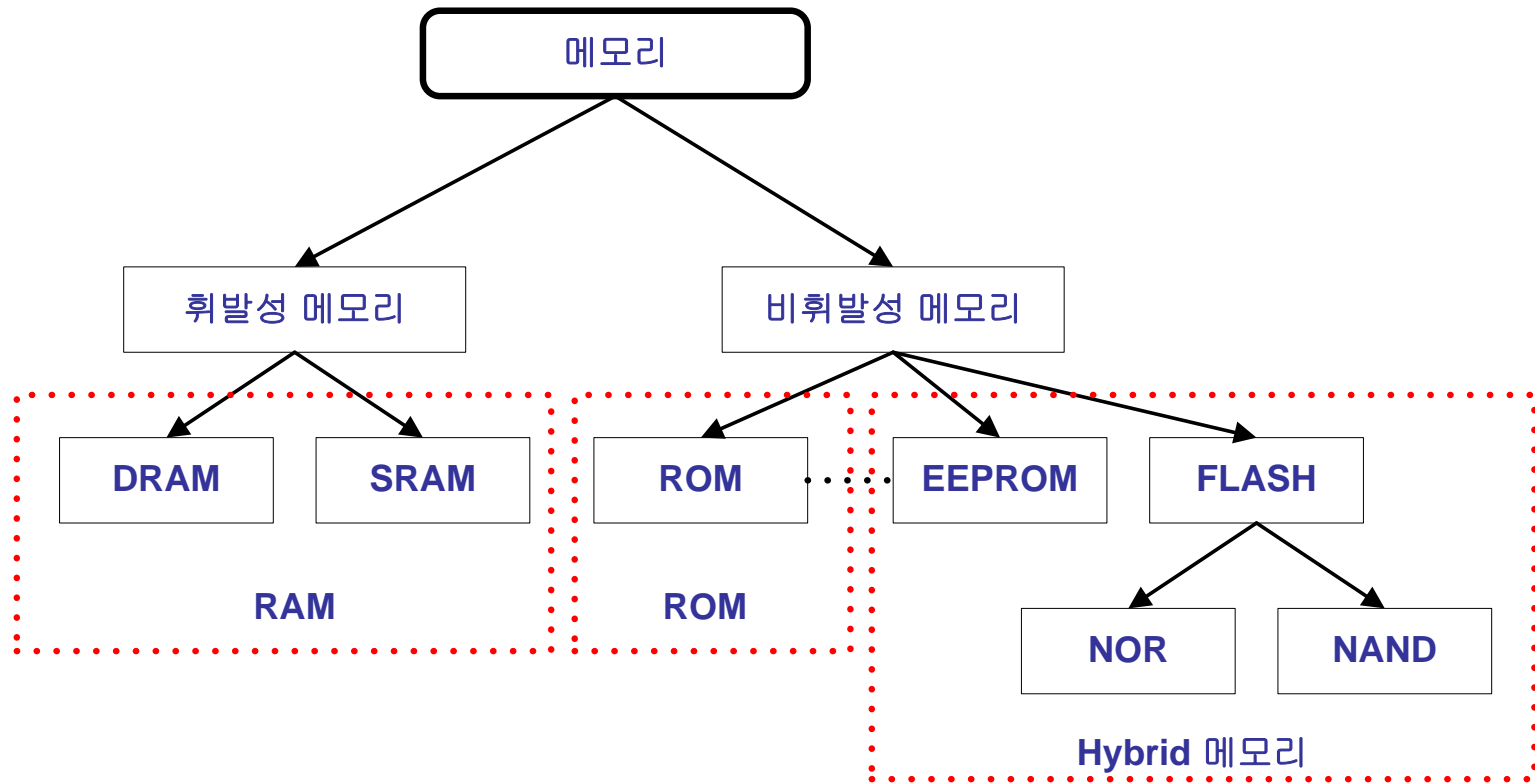
- 하버드(Havard) 아키텍처
 - 명령어를 위한 메모리 인터페이스와 데이터를 위한 메모리 인터페이스가 분리되어 있다.
 - 명령어를 읽을 때 데이터를 읽거나 쓸 수 있어 성능이 우수하다.
 - 버스 시스템이 복잡하여 설계가 복잡하다
 - ARM9, ARM10, XScale 등



메모리 장치

- 메모리 장치의 용도
 - 프로그램과 데이터를 저장하기 위한 공간
- 메모리 장치의 종류
 - 주 기억 장치 (main memory)
 - 프로그램이 실행되는 동안 프로그램과 데이터 저장
 - DRAM이 많이 사용된다
 - 보조 기억 장치(secondary memory)
 - 주 기억장치보다 빈번하게 사용하지 않는 프로그램과 데이터 저장
 - HDD, SD, MMC 등이 사용된다
 - 캐시 (cache)
 - 주 기억장치의 접근 속도를 빠르게 하기 위해서 프로세서 주변에 배치된 소 용량의 메모리
 - SRAM이 사용된다.

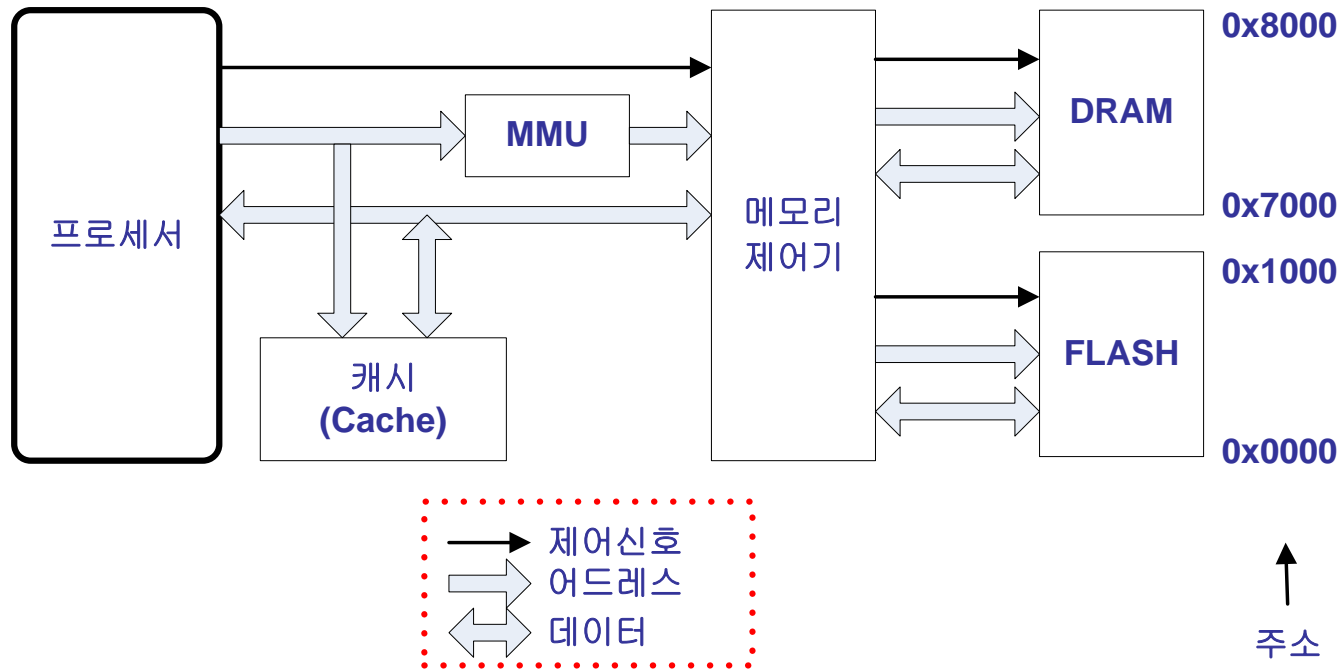
임베디드 시스템에서 사용되는 메모리



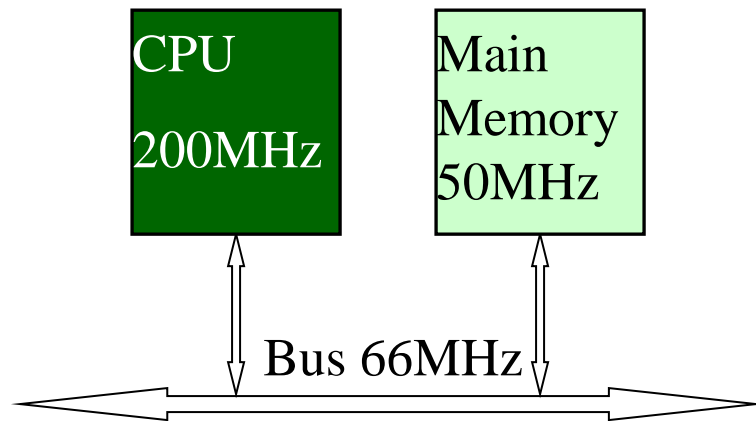
메모리장치의 종류

구 분			속 도	가 격	용 도	특 징
휘발성 (Volatile Memory)	SRAM (Static)		수ns,고속	비싸다	캐시 등	
	DRAM (Dynamic)		수십 ns	저렴	주기억 장치	
비휘발성(Non- Volatile Memory)	EEPROM (Electrically Erasable)		수십 ns	비싸다	소용량 데이터나 프로그램 저장용	
	Flash	NAND	수십 ns	저렴	대용량 데이터 저장	블록 단위 읽기 쓰기
		NOR	수십 ns	비싸다	프로그램 저장 데이터 저장	

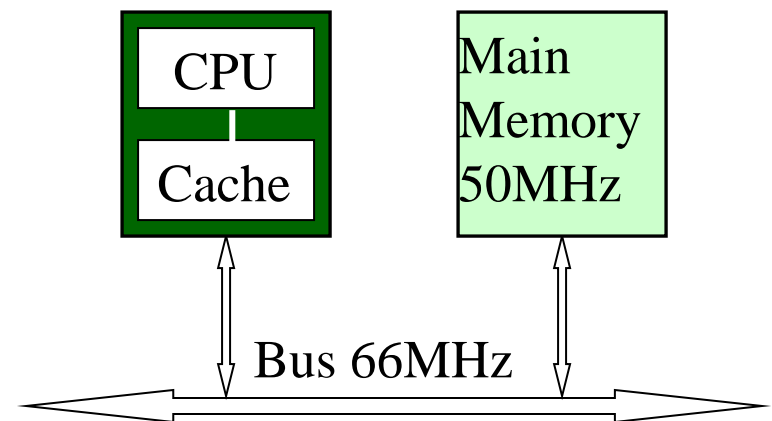
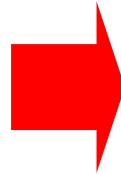
메모리 시스템 구조



Cache 메모리 시스템



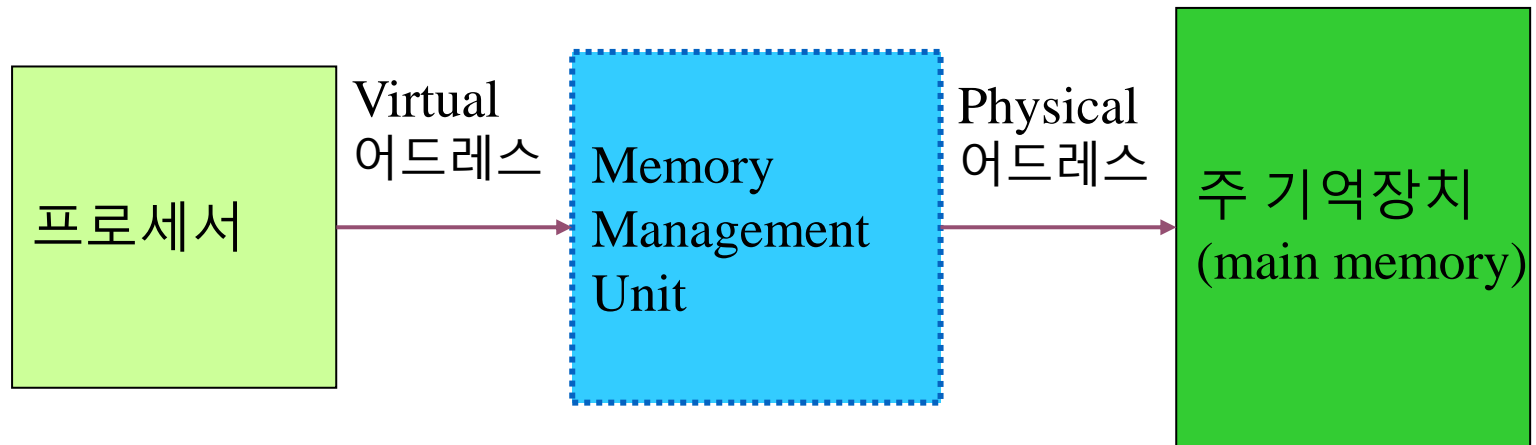
고속의 CPU가 버스 및 메모리 속도에 의존적이며 느다



CPU 주변에 고속의 메모리를 두고 자주 사용되는 명령과 데이터를 저장하여 시스템 성능을 개선

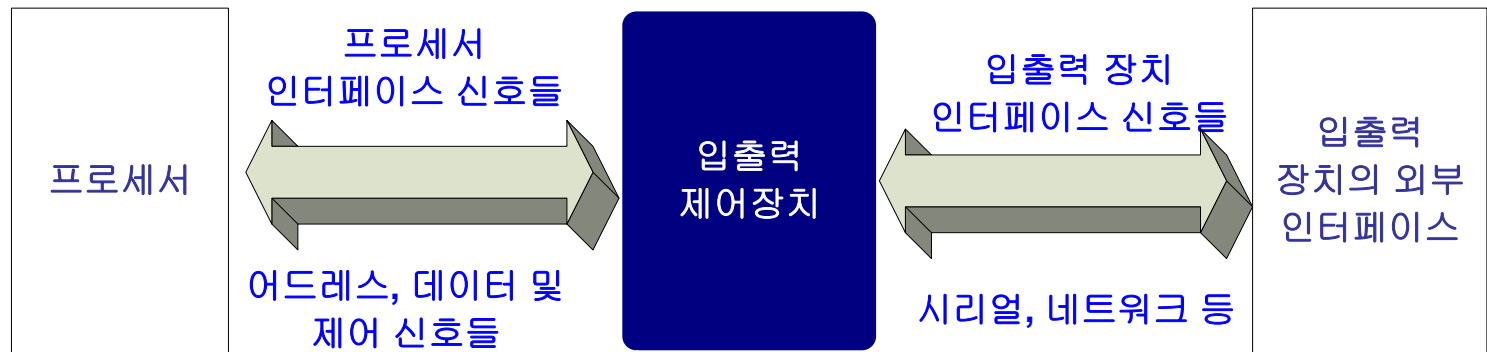
MMU (Memory Management Units)

- 어드레스 변환(translation) 기능
 - CPU에서 사용되는 logical 한 Virtual 어드레스를 physical 어드레스로 변환
- 메모리 보호(protection) 기능



입출력 장치

- 프로세서와 정보를 교환하는 장치
- 디지털 신호 또는 아날로그 신호를 포함 한다.
- 프로세서와는 메모리 장치와 같이 디지털 신호인 어드레스, 데이터 및 제어 신호를 통해서 연결된다.



입출력 장치 제어

- 입출력 장치를 제어하기 위해서는 어드레스 할당이 필요하고 데이터를 교환 하기 위한 데이터 버스와 제어 신호 사용
- 표준 I/O 맵 방식(I/O-mapped peripheral)
 - 전용의 입출력 장치 주소 공간을 할당하여 사용
 - 인텔의 x86 CPU 계열이 대표적
- 메모리 맵 방식(Memory mapped peripheral)
 - 메모리 주소 공간의 일부를 활용하여 사용
 - 대부분의 임베디드 프로세서에서 사용 됨

메모리 맵 방식과 I/O 맵 방식

구분	메모리 맵 방식	I/O 맵 방식
대표적인 프로세서	ARM, MIPS, PowerPC, M68K ...	x86 계열
입출력 장치의 영역	메모리의 일부를 I/O 장치로 사용	메모리 영역과는 별도의 I/O 번지 영역이 존재
명령어	메모리와 I/O 장치 모두 메모리 동작 명령으로 액세스 하며, 각 영역의 구분은 어드레스로 한다.	메모리 액세스 명령과 I/O 액세스 명령(in/out)이 구분
하드웨어	어드레스를 해석하는 디코더 회로에 따라 메모리 혹은 I/O 장치가 선택	메모리 번지와 I/O 번지를 구분하는 신호가 존재.
주의 사항	<ul style="list-style-type: none">- I/O 영역은 Non-cacheable로 설정해야 한다- I/O 영역 변수는 volatile type으로 선언해야 한다.	

입출력 장치의 자원 관리

- 폴링 방식

- 한 프로그램이나 장치에서 다른 프로그램이나 장치들이 어떤 상태에 있는지를 지속적으로 검사하는 전송 제어 방식
- 입출력 장치의 접속 여부 및 데이터 전송의 요청과 종료를 검사한다.

- 인터럽트

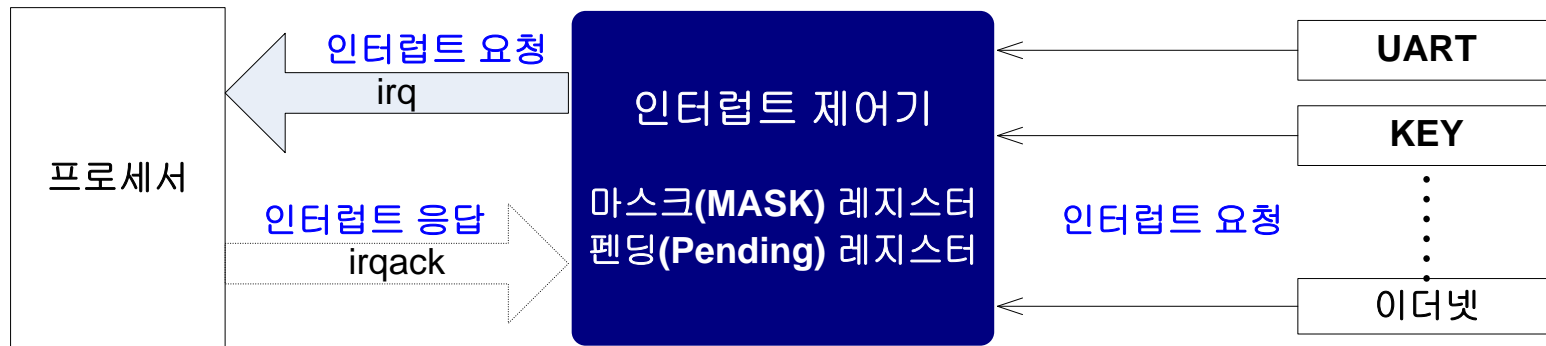
- 프로세서는 일련의 처리를 수행하고, 주변장치에서 입출력 처리 동작이 필요한 경우 프로세서에게 진행 중 이던 명령을 멈추고 새로운 동작을 할 수 있도록 한다.
- 프로세서는 한번에 한 개의 명령만을 수행할 수 있다.
- 인터럽트를 이용하면 멀티태스킹을 지원할 수 있도록 한다.
- 사용자는 모든 작업이 동시에 수행되는 것처럼 보이게 동작한다.

- DMA 방식

- DMA(Direct Memory Access) 방식은 CPU의 개입 없이 입출력 장치와 기억장치 사이에 데이터를 전송하는 방식

인터럽트 인터페이스

- 인터럽트 제어기
 - 입출력 장치에서 발생하는 인터럽트의 요청을 제어 한다.
 - 하드웨어에 따라 인터럽트 응답을 위한 신호도 제공된다.



인터럽트의 발생

- 인터럽트 요청 (Interrupt Request)

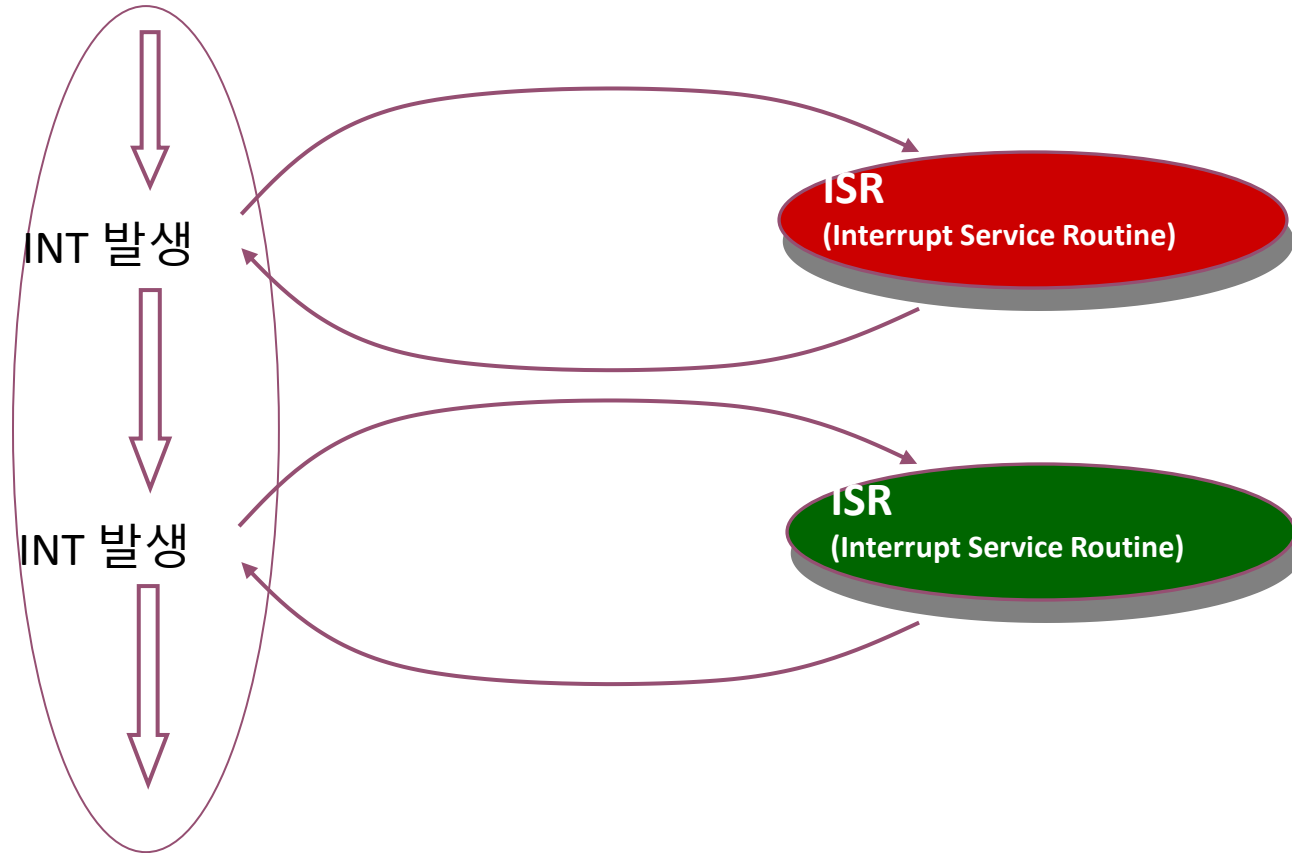
- 외부 장치에서 입출력 동작에 대한 처리를 프로세서에 요청
- 인터럽트의 발생은 하드웨어적으로 이루어 진다. 따라서 인터럽트가 발생하면 프로세서가 스스로 프로그램의 개입 없이 일련의 동작을 수행해야 한다.
- 인터럽트 Vector
 - 인터럽트 서비스 루틴을 처리하기위한 명령 또는 위치가 저장된 메모리 공간
- 인터럽트의 요청에 따라 프로세서는 정해진 절차에 의하여 발생 된 인터럽트의 처리 여부를 결정하고 인터럽트를 서비스하는 절차(ISR : Interrupt Service Routine)를 수행한다.

- 인터럽트 발생의 예

- 시리얼로 데이터 입력 완료
- 시리얼 데이터 전송 준비 완료, 또는 전송 에러 발생
- 이더넷 데이터 수신 완료, 이더넷 전송에러 등

인터럽트 발생에 의한 프로세서 흐름제어

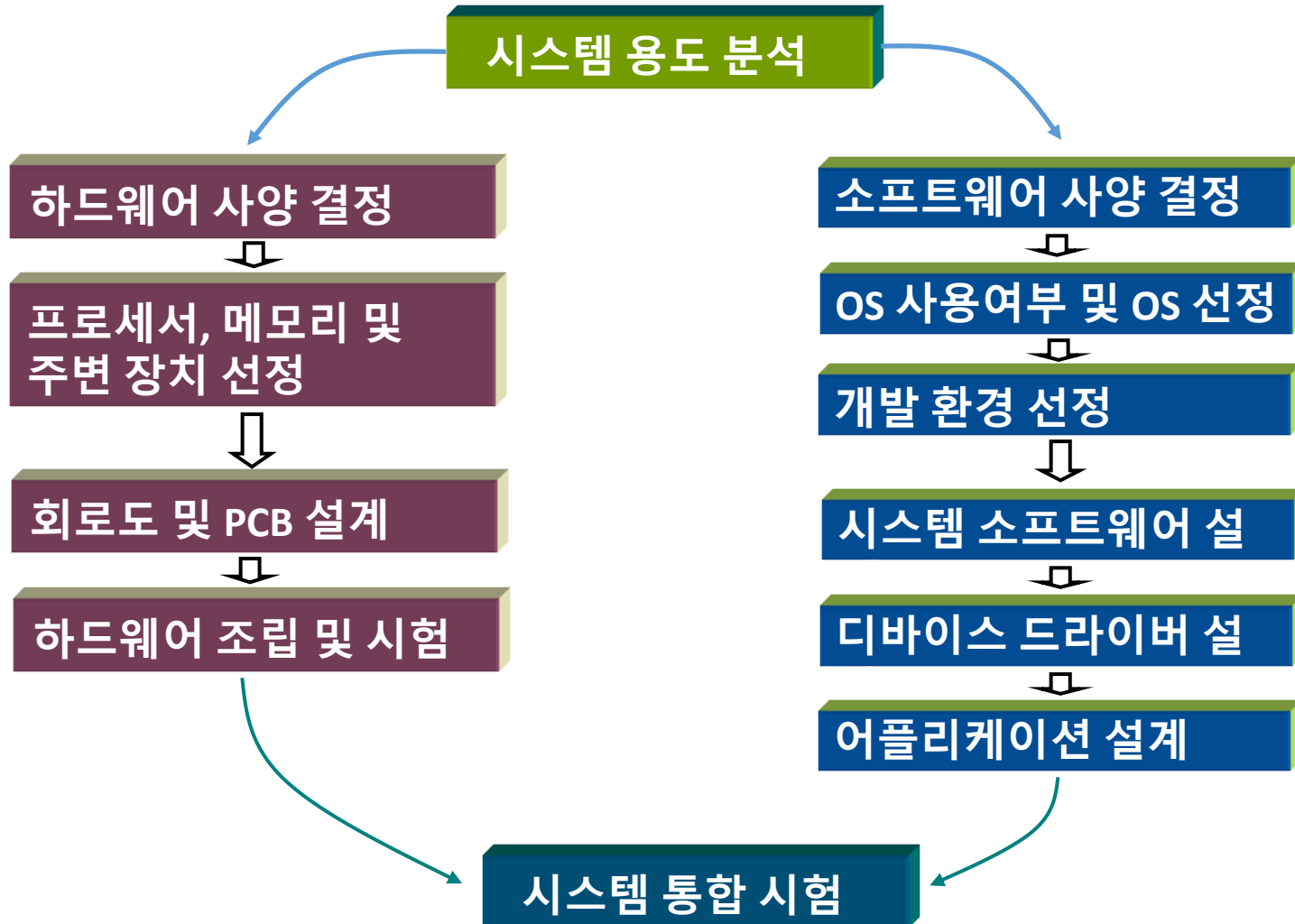
MAIN 프로그램 루틴



인터럽트 벡터(Vector)

- 인터럽트 벡터
 - 인터럽트 서비스 루틴을 처리하기 위한 명령 또는 위치가 저장된 메모리 공간
- 인터럽트 벡터 주소 지정 방식
 - 고정 인터럽트(Fixed interrupt)
 - 인터럽트가 발생시 처리할 어드레스가 지정되어 변경이 않 된다.
 - 지정된 어드레스에 인터럽트를 처리하기위한 명령 또는 위치가 저장되어 있다.
 - 벡터 인터럽트(Vectored interrupt)
 - 일반적인 마이크로프로세서 장치에서 여러 개의 주변 장치가 시스템 버스에 연결되어 사용되는 경우
 - 주변장치가 인터럽트를 처리할 주소를 제공

임베디드 시스템 설계 절차



소프트웨어 사양 결정

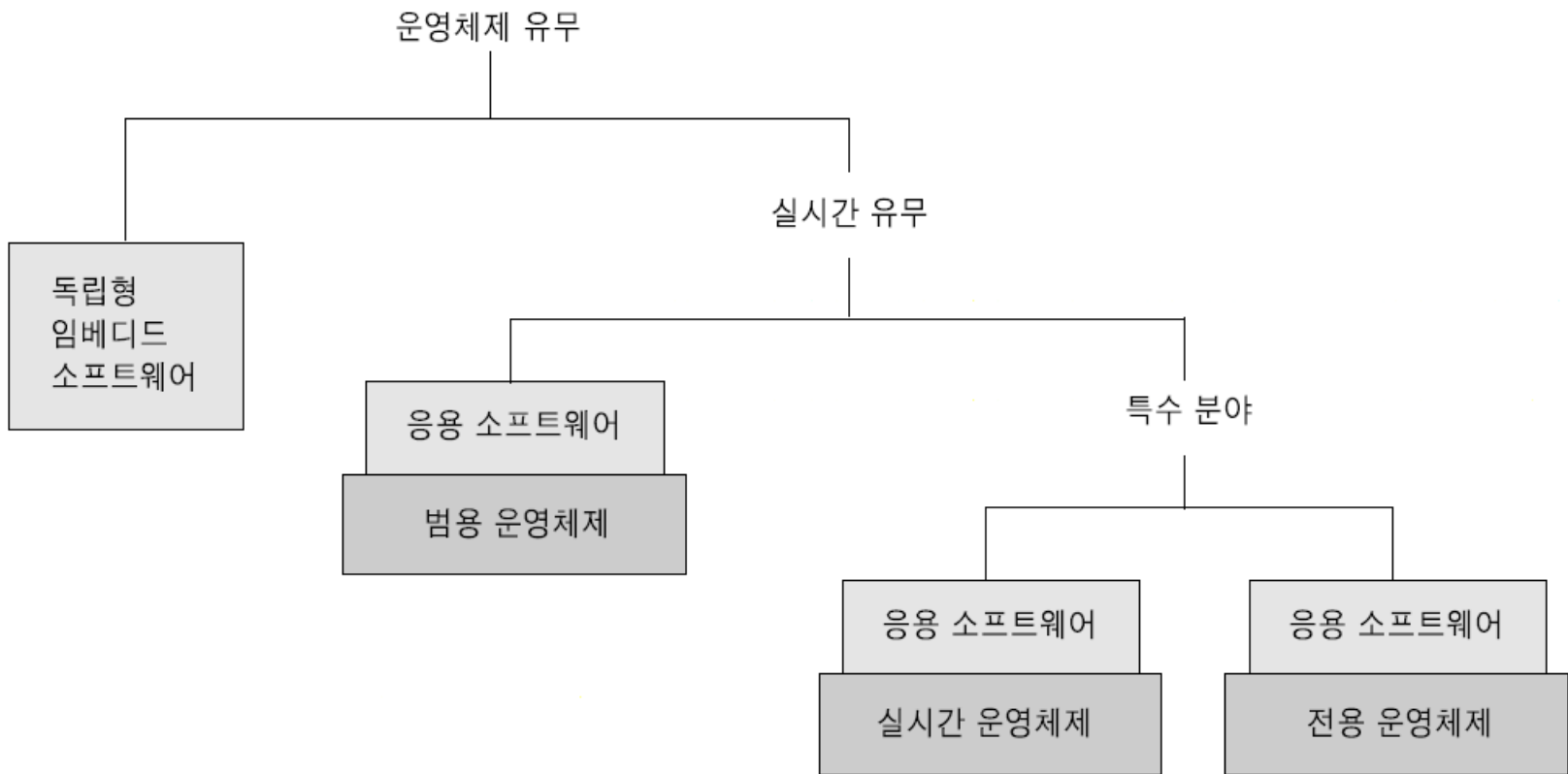
- 시스템의 사양 및 성능에 따른 소프트웨어 선정
- OS 사용 여부 결정
 - OS의 필요성 결정
 - Real-time의 필요성 및 시스템 메모리의 크기 등에 따른 OS 선정
- 소프트웨어 개발
 - 개발 시간, 난이도 및 비용에 따라 자체 개발 또는 외주 개발 결정
- 라이선스(License) 조건
 - 소프트웨어의 사용 권한 및 제한 사항 확인
 - MPEG, MP3 등

OS 사용 여부 결정

- 임베디드 시스템에서의 운영체제
 - 시스템의 규모가 커짐에 따른 멀티 태스킹(Multi Tasking) 기능 요구
 - 네트워크나 멀티미디어 기능이 시스템의 기본 요소가 됨
 - Networking, GUI, Audio, Video
 - 리얼타임의 필요성이 부각됨
 - 지능성이 부가되고, 기능이 많아지고, 복잡해짐
 - 순차적인 프로그램 작성이 불가능하여 운영체제가 도입됨
- 임베디드 운영체제
 - 상용 RTOS(Real-Time OS)
 - 윈도우 CE
 - 임베디드 Linux
 - 임베디드 JAVA

임베디드 OS 선정

- 상용 RTOS : Hard RealTime/Multi-thread/ Preemptive
 - pSOS, VxWorks, VRTX 등 다수
 - 일반 운영체제와 거의 같은 기능을 수행
 - 시간 제약성, 신뢰성 등을 일반 운영체제 보다 중요시 함
 - 일반적으로 한가지 목적에 최적화 되어있음
- 임베디드 OS : Soft RealTime/Multi-process/ non preemptive
 - Windows CE
 - 임베디드 리눅스
 - 임베디드 자바
- 최근 동향
 - 임베디드 OS 세계시장 :
 - WinCE, 임베디드 리눅스가 기존의 RTOS 보다 시장 점유율이 높아지는 추세
- OS 선정
 - 시스템의 특성에 적합한 OS 선정



[그림 1-4] 운영체제에 따른 임베디드 소프트웨어