

Xilinx Zynq FPGA, TI DSP, MCU기반의 프로그래밍 및 회로 설계 전문가 과정

강사 - Innov (이상훈)

gcccompil3r@gmail.com

학생 - 이유성

dbtjd1102@naver.com

chapter 1. 리눅스 소개

리눅스 운영체제를 개발한 리누즈 토발즈와 리처드 스톨만이 설립한 자유 소프트웨어 재단이라는 단체의 GNU(Gnu is Non Unix)프로젝트가 합세하면서 이 단체와 뜻을 같이 하는 전세계프로그래머의 도움으로 현존하는 최고의 운영체제 중 하나인 리눅스가 탄생하게 된 것이다.

이 재단은 이미 운영체제의 기본이 되는 많은 소스코드를 공개했다.

리눅스는 GNU 정신에 따라 완전 공개로 배포되며 모든 프로그램의 소스 또한 공개 되었다.

원하는 사람은 누구나 그 소스를 수정하여 성능을 향상시킬 수 있다.

유닉스와 리눅스

리눅스는 유닉스 계열의 운영체제이다.

유닉스란 1969년 AT&T벨 연구소의 켄 톰슨과 데니스 리치가 만든 운영체제이다.

생명을 제공하는 태스크와 장소를 제공하는 파일이라는 두 가지 객체로 모든 것을 지원한다.

->모든 것은 파일

리눅스의 장점

*사용자 임의대로 재구성이 가능하다

*열악한 환경에서도H/W 자원을 적절히 활용하여 동작한다(똥컴도 잘됨->최적화가 잘 되어있음)

*커널의 크기가 작다

*완벽한 멀티유저, 멀티태스킹 시스템

*뛰어난 안정성 (네트워크가 되고있다.. ex구글)

*빠른 업그레이드

*강력한 네트워크 지원(TCP/IP)

*풍부한 소프트웨어 (스톨만의 GNU)

*사용자를 위한 여러 가지 공개 문서들(몇 십만줄 됨.)

chapter 2.리눅스 커널 구조

리눅스 커널 구조를 설명하기에 앞서 운영체제란 무엇인가?

가장 일반적인 정의는 운영체제는 자원관리자이다.(하드웨어,소프트웨어)

운영체제가 관리해야 할 자원은 크게 물리적 자원과 추상적인 자원으로 구분할 수 있다.

물리적인 자원은 CPU,메모리,디스크,터미널(모니터),네트워크 등 시스템을 구성하고 있는 요소들과 주변 장치 등이 있다.

추상적인 자원은

CPU를 추상화 시킨 (task(thread,process))

메모리를 추상화 시킨 세그먼트(vm)과 page(물리메모리)

디스크를 추상화 시킨 파일

네트워크를 추상화시킨 통신 프로토콜,패킷 등이 있다.

리눅스의 개념적 구조에서

OS가 관리할 5대 요소

Filesystem Manager

Memory Manager

Task Manager

Device Manager

Network Manager

좀 더 구체적으로

Task Manager는 태스크의 생성(fork(),pthread_create),실행(exec),상태 전이(run queue,wait queue),스케줄링(time slice),시그널 처리, 프로세스간 통신(pipe,shared memory,socket..) 등의 서비스를 제공한다

Memory Manager는 물리 메모리 관리(DRAM관리),가상 메모리 관리(Stack,Heap,Data,Text) 세그먼테이션,페이징,페이지 부재 결함 처리(segmentation fault)등의 서비스를 제공한다

Filesystem Manager는 파일의 생성, 접근 제어(systemcall), inode관리(disk block의 위치(실제 데이터의 위치)), 디렉터리 관리, 수퍼 블록 관리(root file 위치 아는 곳) 등의 서비스 제공

Network Manager는 소켓(socket) 인터페이스, TCP/TP같은 통신 프로토콜 등의 서비스 제공

Device Manager는 디스크나 터미널,CD,네트워크 카드 등과 같은 주변 장치를 구동하는 드라이버들로 구성.

chapter 2.리눅스 커널 구조

그럼 운영체제는 무엇을 위하여 자원을 관리 하는 걸까?

: 사용자에게 서비스를 제공하기 위해서이다 - 서비스는 시스템 호출을 의미한다

요약

운영체제는 시스템 호출을 통해 태스크가 자원을 사용할 수 있게 해주는 자원관리자인 것이다.

chapter 3.태스크 관리

리눅스는 태스크를 통해 다양한 생명(fork(),pthread_create)과 변화(exec)를 제공한다

문맥 교환 =Context switching

task의 life cycle = program -> process -> 제어권 -> 구동

task ->프로세스(쓰레드의 리더) -> task_struct

task->쓰레드 ->task_struct

vi -t task_struct ->144 내부에 tgid , pid선언 되어있음.

tgid 와 pid값이 같으면 프로세스

같지 않으면 쓰레드(프로세스 구성원) , 리더인 경우 프로세스

리눅스 파일 포맷 확인 명령어

elf 리눅스 실행 파일 포맷

dwarf 디버깅 파일 포맷

pe 윈도우 실행 파일 포맷

실행 파일 정보.

file [실행 파일명]

readelf -h [실행 파일명]