TI DSP, MCU 및 Xilinx Zynq FPGA 프로그래밍 전문가 과정

강사 - Innova Lee(이상훈)
gcccompil3r@gmail.com
학생 - 최대성
c3d4s19@naver.com

#include <unistd.h>

내부 함수

char getopt(int argc, char** argv, char* option_string);

-> 아스키코드 문자 하나 리턴

While 루프에서 돌리면 문자(char) 하나씩 순서대로 리턴되는데 옵션과 일치하는 문자가 있는 경우 옵션문자(char) 반환하고 아닌경우 '?' 반환

#include <dirent.h>

헤더 내부 구조체

DIR -> 대상 디렉토리 자체 정보 보관 구조체 struct dirent -> 디렉토리 내부 파일들의 정보 보관 구조체

내부 함수

opendir(const char* [여는 대상 디렉토리 이름]);
-> return: (DIR *) 반환, 값이 없는경우 NULL 반환 인자에 "."을 넣는 경우 현재 디렉토리를 연다

readdir([읽을 구조체의 포인터 DIR*])

-> return: (struct dirent *)하나씩 반환, 없으면 NULL

While 루프에서 돌리면 포인터(struct dirent *) 하나씩 순서대로 리턴되고 값이 없는 경우 NULL 반환

mkdir([생성할 디렉토리 이름])

-> 디렉토리 생성 함수

closedir([닫을 구조체의 포인터 DIR*])

-> 디렉토리 닫는 함수

#include <sys/stat.h>

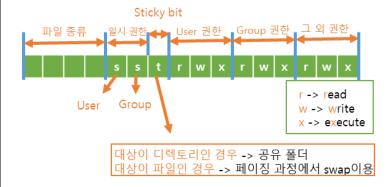
내부 구조체

struct stat -> 파일의 상태를 담은 구조체

```
struct stat{
                   st_dev;
    _dev_t
    _ino_t
                   st_ino;
   unsigned short st_mode;
   short
                   st_nlink;
    short
                   st_uid;
    short
                   st_gid;
    _dev_t
                   st_rdev;
                   st_size;
    _off_t
    time_t
                   st_atime;
    time_t
                   st_mtime;
    time_t
                   st_ctime;
    };
```

위 구조체 내부의 st_mode 구조

struct stat.st_mode



파일 종류 확인 함수들

S_ISDIR([stat 구조체의 st_mode])
-> 디렉토리인지 확인

S_ISFIFO([stat 구조체의 st_mode])
-> 파이프 파일인지 확인

S_ISREG ([stat 구조체의 st_mode])
-> 일반(regular) 파일인지 확인

S_ISLNK([stat 구조체의 st_mode])
-> 바로가기 파일인지 확인

S_ISSOCK([stat 구조체의 st_mode])
-> 소캣 파일(통신용) 인지 확인

S_ISCHR([stat 구조체의 st_mode])
-> 캐릭터 디바이스인지 확인

S_ISBLK([stat 구조체의 st_mode])
-> 캐릭터 디바이스인지 확인

```
리눅스 명령어
```

chmod [8진수 숫자 4개] [파일명]

-> struct stat 내부의 st mode 변경 가능(권한 설정)

리눅스 Is 명령어 기능 구현

#include <unistd.h>

```
#include <stdio.h>
#include <stdlib.h>
#include <dirent.h>
int main(int argc,char **argv)
{
   DIR *dp;
    int i=0;
    int cmd;
   struct dirent *p;
    int flag=0;
   while((cmd=getopt(argc,argv, "alRi"))>0)
        switch(cmd)
        {
            case 'a':
                //flag : bit
                break;
            case 'l':
                 flag |=2;
                 break;
            case 'R':
                flag |=4;
                 break;
            case 'i':
                 flag |=8;
                 break;
        }
    dp=opendir(".");
   while(p=readdir(dp))
        if(!(flag&1))
            if(p->d_name[0]=='.')
                continue;
        printf("%-16s ",p->d_name);
        if((i+1)\%5==0)
            printf("\n");
        j++;
   printf("\n");
   closedir(dp);
    return 0;
```

파일 관련 정보 print 기능 구현

```
#include <stdio.h>
#include <stdlib.h>
#include <dirent.h>
#include <sys/types.h>
#include <unistd.h>
#include <fcntl.h>
#include <svs/stat.h>
#include <pwd.h>
#include <grp.h>
#include <time.h>
int main(int argc,char **argv){
   struct stat buf;
   struct dirent *p;
   struct passwd *pw;
   struct group *gr;
   struct tm *tm;
   char ch;
   char perm[11]="----";
    char rwx[4]="rwx";
    char sst[4]="sst";
    int i:
   stat(argv[1],&buf);
    if(S_ISDIR(buf.st_mode))
                 perm[0]='d';
        if(S_ISREG(buf.st_mode))
                 perm[0]='-';
    if(S ISFIFO(buf.st mode))
        perm[0]='p';
     if(S ISLNK(buf.st mode))
        perm[0]='|';
     if(S_ISSOCK(buf.st_mode))
         perm[0]='s';
     if(S_ISCHR(buf.st_mode))
         perm[0]='c';
     if(S_ISBLK(buf.st_mode))
        perm[0]='b';
     for(i=0;i<9;i++)
         if((buf.st_mode>>(8-i))&1)//파일권한 설정
             perm[i+1]=rwx[i%3];
     for(i=0;i<3;i++)
         if((buf.st_mode>>(11-i))&1)
             if(perm[(i+1)*3]=='-')
        perm[(i+1)*3]=sst[i]^0x20;//대소문자 변경
             else
                 perm[(i+1)*3]=sst[i];
   printf("%s ",perm);
   printf("%ld ",buf.st_nlink);
   pw=getpwuid(buf.st_uid); //개인 id 리턴
   printf("%s ",pw->pw_name);
   gr=getgrgid(buf.st_gid); //그룹 id 리턴
   printf("%s",gr->gr name);
   printf("%d",buf.st_size);
    tm=localtime(&buf.st_mtime);
   printf("%d-%02d-%02d %02d:%02d", tm-
>tm_year+1900, tm->tm_mon+1, tm->tm_mday, tm-
>tm_hour,tm->tm_min);
   printf("\n");
    return 0;
```