

TI DSP, MCU 및 Xilinx Zynq FPGA 프로그래밍 전문가 과정



2018.03.27

24 일차

강사 - Innova Lee(이상훈)

gcccompil3r@gmail.com

학생 - 신민철

akrn33@naver.com

goto.c

```
#include<stdio.h>
#include<unistd.h>
#include<stdlib.h>
#include<fcntl.h>
#include<signal.h>
```

```
int main(void)
{
    int ret;
    char buf[1024];
    if((ret = read(0, buf, sizeof(buf))) > 0)read 의 리턴이 0 보다크면
        goto err;//err 로 gogo
    return 0;

err:
    perror("read()");//success 가 출력될것임.
    exit(-1);
}
```

goto2.c

```
#include<stdio.h>
#include<unistd.h>
#include<stdlib.h>
#include<fcntl.h>
#include<signal.h>
void name(){
    goto rara;//goto 는 스택을 해제할 수 없다.
}
int main(void)
{
    int ret;
    char buf[1024];
```

```

    name();
    if((ret = read(0, buf, sizeof(buf))) > 0)
        rara:
        goto err;

    return 0;
err:
    perror("read()");
    exit(-1);
}

```

```

setjmp.c
#include<stdio.h>
#include<stdlib.h>
#include<setjmp.h>
#include<fcntl.h>

```

jmp_buf env;//:을 대신해주는놈이 셋점프인데 셋점프는 이엔브이를 설정해주는거임

```

void test(void)
{
    longjmp(env, 1);//밑에있는구간에있는 이엔브이로 가는거 셋점프에
리턴값을 1 로주라는 뜻
}

```

```

int main(void)
{
    int ret;
    if((ret = setjmp(env)) == 0){//goto of lable 아무것도 세팅안했을
땐 셋점프가 무조건 0 이 나온다.
        printf("test\n");
        test();
    }
}

```

```
        else if(ret > 0)
            printf("error\n");
        return 0;
    }
```

setjmp2.c

```
#include<stdio.h>
#include<stdlib.h>
#include<setjmp.h>
#include<fcntl.h>
```

```
jmp_buf env;
```

```
void test(void)
{
    int flag = -1;
    if(flag < 0)
        longjmp(env, 1);
    printf("call test\n");
}
```

```
int main(void)
{
    int ret;
    if((ret = setjmp(env)) == 0)
        test();
    else if(ret > 0)
        printf("error\n");
    return 0;
}
```

signal.c

```
#include<stdio.h>
```

```
#include<signal.h>
```

```
void my_sig(int signo)
```

```
{
```

```
    printf("my_sig called\n");
```

```
}
```

```
void my_sig2(int signo)
```

```
{
```

```
    printf("my_sig2 called\n");
```

```
}
```

```
int main(void)
```

```
{
```

```
    void(*old_p)(int);
```

```
    void(*old_p2)(int);
```

```
    old_p = signal(SIGINT, my_sig); //sigint 발생하면 mysig 동작  
    pause();
```

```
    old_p = signal(SIGINT, my_sig2); //한단계 전에 등록했던것을 리  
    턴해준다.
```

```
    pause();
```

```
    old_p = signal(SIGINT, old_p);
```

```
    pause();
```

```
    for(;;)
```

```
        pause();
```

```
    return 0;
```

```
}
```

signal5.c

```
#include<stdio.h>
```

```
#include<signal.h>
```

```
int main(void)
```

```
{
```

```
    signal(SIGINT, SIG_IGN);
```

```
    pause();
```

```
    return 0;
```

```
}
```

setjmpdsp.c

```
#include <fcntl.h>
```

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <setjmp.h>
```

```
jmp_buf env1;
```

```
jmp_buf env2;
```

```
void test1(void)
```

```
{
```

```
    longjmp(env1, 1); //env1 로 이동
```

```
}
```

```
void test2(void)
```

```
{
```

```
    longjmp(env1, 2);
```

```
}
```

```
void test3(void)
```

```
{
```

```
    longjmp(env2, 1);
```

```
}
```

```

int main(void)
{
    int ret;
    if((ret = setjmp(env1)) == 0)//goto 레이블과 비슷함 :같은놈
    {
        printf("this\n");
        test1();
    }
    else if(ret == 1)
    {
        printf("1\n");
        test2();
    }
    else if(ret == 2)
        printf("2\n");
    else
    {
        printf("goto letsgo label\n");
        goto letsgo;
    }

    if((ret = setjmp(env2)) == 0)
    {
        printf("second label\n");
        test3();
    }
    else
        longjmp(env1, 3);
letsgo:
    goto err;

    return 0;

err:

```

```
    printf("Error!!!\n");
    exit(-1);
}
```

updownnt.c

```
#include <time.h>
#include <stdio.h>
#include <fcntl.h>
#include <unistd.h>
#include <signal.h>
#include <stdlib.h>
#include <stdbool.h>
```

```
void sig_handler(int signo)
{
    printf("You lose! Input should be within 1 second!\n");
    exit(0);
}
```

```
void make_game(int *data)//main 에있는 data 의 주소를 인자로
{
    *data = rand() % 100 + 1;
}
```

```
bool check_correct(int data, int cmp)//데이타 확인용 함수
{
    if(data == cmp)
        return true;
    else
        return false;
}
```

```
void start_game(int data)//게임시작 함수
```



```

{
    char buf[32] = {0};
    bool fin;
    int i, cmp;

    for(i = 0; i < 10; i++)
    {
        signal(SIGALRM, sig_handler);
        printf("숫자를 맞춰봐!\n");
        alarm(1); // 1 초 후 알람 울림
        read(0, buf, sizeof(buf));
        alarm(0); // 눌렀는데 다음께 0.99 될수있으니까 알람 초기화 같은거
        cmp = atoi(buf); // 아스키 투 인티저 ㅋㅋ 버프를

        fin = check_correct(data, cmp);

        if(fin)
        {
            printf("You Win!!!\n");
            exit(0);
        }
        else
        {
            if(data > cmp)
                printf("%d 보다 크다\n", cmp);
            else
                printf("%d 보다 작다\n", cmp);
        }
    }

    printf("You Lose!!! You Babo\n");
}

```

```
int main(void)
{
    int data;

    srand(time(NULL));
    make_game(&data);

    start_game(data);

    return 0;
}
```