

TI DSP, MCU 및 Xilinx Zynq FPGA 프로그래밍 전문가 과정



2018.05.02

46 일차

강사 - Innova Lee(이상훈)

gcccompil3r@gmail.com

학생 - 신민철

akrn33@naver.com

```
#include<stdio.h>
#include<unistd.h>
#include<errno.h>
#include<stdlib.h>
```

```
int main(void){
```

```
    int i;
    unsigned int test_arr[7] = {0};
```

```
    register unsigned int* r0 asm("r0") = 0;
    register unsigned int r1 asm("r1") = 0;
    register unsigned int r2 asm("r2") = 0;
    register unsigned int r3 asm("r3") = 0;
    register unsigned int r4 asm("r4") = 0;
    register unsigned int r5 asm("r5") = 0;
    register unsigned int r6 asm("r6") = 0;
    register int r7 asm("r7") = 0;
```

```
    r0 = test_arr;
```

```
    asm volatile("mov r1,#0x3\n"
                 "mov r2,r1,lsl #2\n"
                 "mov r4,#0x2\n"
                 "add r3,r1,r2, lsl r4\n"
                 "stmia r0!,{r1,r2,r3}\n"
                 "str r4,[r0]\n"
                 "mov r5, #128\n"
                 "mov r6, r5, lsr #3\n"
                 "stmia r0, {r4,r5,r6}\n"
                 "sub r0,r0, #12\n"
                 "ldmia r0, {r4,r5,r6}\n"
                 "swp r6, r3, [r0]");
```

```

for(i = 0; i < 7; i++)
    printf("test_arr[%d] = %d\n",i,test_arr[i]);

printf("r4 = %u, r5 = %u, r6 = %u\n", r4, r5, r6);

r7 = 2;

asm volatile("swi #0" : "=r" (r0): "r" (r7): "memory");
//r0 에다가 집어넣어라 시스템콜번호 r7 을 , 메모리는
//메모리배리어를 쳐서 인스트럭션스케줄링이 안되게하기위해
//끝날때까지
//소프트웨어인터럽트 0 번에다가 r7 값을 넣으면서 포크하게됨
//포크를 호출하면 밑과 같음
if(r0 > 0)
    printf("r0 = %p, Parent\n", r0);//자식프로세스의 pid
else if(r0 == 0)
    printf("r0 = %p, Child\n",r0);//자식프로세스의 pid = 0
else{
    perror("fork()");
    exit(-1);
}

return 0;
}

```