



EDDI

Electronic Design  
Development Institute

---

# 에디로봇아카데미

## 임베디드 마스터 Lv# 과정

제 # 기

2021. 00. 00

000

main  
stack frame

IF DE EX WB

O x 1018  
O x 1010  
O x 1016  
O x 1018

edx	edx
0x3	0x2

esi	edi
0x2	0x3

[illegible]

## Memory

h p

15p

2/11

한편

작업을 ~~완성~~하는 데  
500 과 2000 원의 차이  
는 이기 남

	oxd1f8
로트 hp 지	
	ox d1f0
Main stock floor	ox d1b8
	ox d1b0
장식물 주	
	ox d1f8
oxoooo ffff      hp	
oxffff d1f0	
d1x3	ox d1b0
ox2	ox d1a4c
	ox d1f8
	ox d1c4

$\text{ox d8fo}$        $\text{ox d8fo}$       push %ebp, %ebp

" " mov %eax, %eax

" " mov %eax, -0x4(%ebp)  
mov %esi, -0x8(%ebp)

i eax, esi는 ~~값이~~ 같음

" "

" " mov -0x4(%ebp), %eax

" " int3 -0x8(%ebp), %eax

:  $0 \times 2 * 0 \times 3 = 6$

eax 값이  $-0 \times 8(\%ebp)$  로 바뀌어

ebx 가 같음

$\text{ox d8fo}$        $\text{ox d8f8}$       pop %ebp

: pp 전의 ~~값과~~ 같을 임의(임의) 바이트가 저장된다.  
(변경이 안 됨)

"       $\text{ox d860}$       ret4

: pp 후의 다른 바이트  
→  
현재 stack 의 끝은 임의식 (바이트) 바이트가 저장된다.  
(바이트) (바이트)

CPV

enx	edx
0x3	0x2
0x6	

es <sub>1</sub>	ed <sub>1</sub>
ox <sub>2</sub>	ox <sub>3</sub>



**EDDI**  
Electronic Design  
Development Institute

## Memory

## Memory

$2\frac{1}{2} \text{ h} \cdot \text{h}^2 \text{ (PWA)}$	$0 \times 0 \times 18$
$0 \times 1 \text{ (max 350K)}$	$0 \times 0 \times 10$
$0 \times 2$	$0 \times 0 \times 10$
$0 \times 3$	$0 \times 0 \times 18$
	$0 \times 0 \times 16$
$0 \times 0 \times 20 \text{ (max 350K)}$	$0 \times 0 \times 10$
$0 \times 3 \times 15 \text{ (max 350K)}$	$0 \times 0 \times 18$

CPV

eax	edx
0x3	0x2
0x6	
0x0	
0x8	
0x0	

plint

esi	edi
0x2	0x3
0x6	0x5555 5555 4134

明



# CONTENTS

## ## 메모리 연산 관점에서 RISC vs CISC ##

### RISC(Load & Store 아키텍처 )

- . 메모리 2 메모리 연산 불가능 ( 레지스터 끼리만 가능 )
- . Load Store 통해서 Register 로 불러와 연산하고 저장함
- . 레지스터 를 많이 두려 하고 있고 레지스터 사이의 연산만해서 불필요한 메모리 접근 줄임

### CISC(Register Memory 아키텍처 )

- . 메모리 2 메모리 연산 가능 ( 레지스터 당연히 가능 )
- . 그렇기에 mov 가 처리할 수 있는 케이스가 여러가지가 됨 (Reg to Mem, Mem to Reg, Reg to Reg, Mem to Mem)