Y86 Function Call and Return

- Topic:
  - CALL and RET instructions
- Learning Outcomes
  - Draw a stack illustrating function calls and return.
  - Use CALL/RET appropriately.

Function Call and Return

CALL	8	0	Dest (a 64-bit address)

RET	9	0

CALL <ADDRESS>

CALL 0x1000
CALL SUM

RET
Encoding: 0x90

```
#Initialize Stack
irmovq  0x2000, %rsp
# Initialize registers
irmovq   0xA0C0, %rdi
irmovq   0x0B0D, %rsi
call      sum
rrmovq   %rax, %rbx
halt
# Compute rsi+rdi and
# return in rax
.pos 0x1000
sum:
xorq      %rax, %rax
addq      %rdi, %rax
addq      %rsi, %rax
ret
```

CALL	8	0	00	10	00	00	00	00	00	00

Function **Call** and Return

CALL:

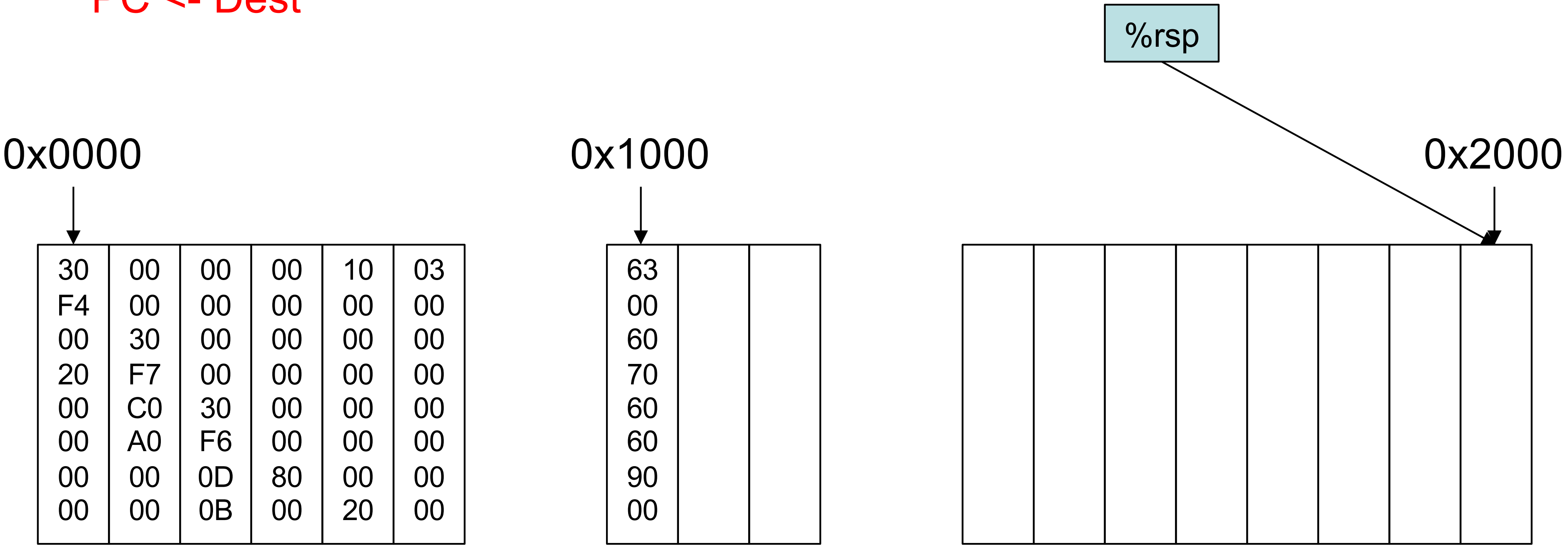
# This part is just PUSHQ PC*

R[%rsp] <- R[%rsp] – 8

M_8[R[%rsp]] <- PC

# Now change the PC

PC <- Dest

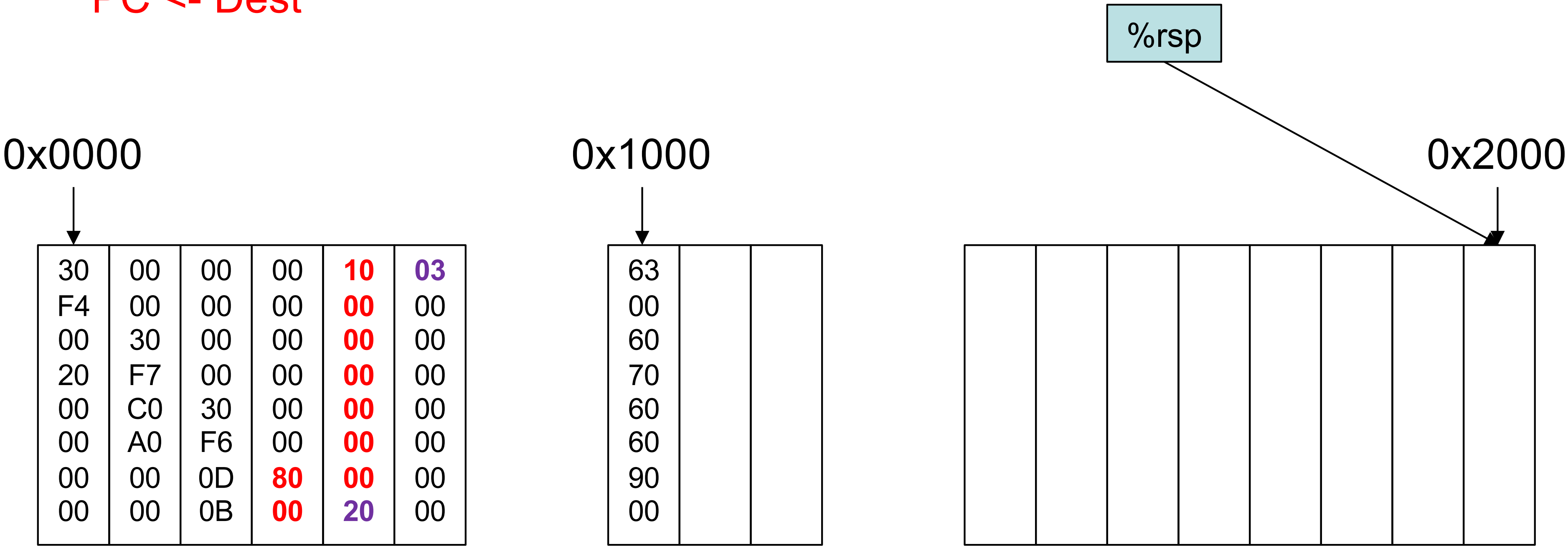


```
0x0000                              0x1000
30  00  00  00  10  03              63
F4  00  00  00  00  00              00
00  30  00  00  00  00              60
20  F7  00  00  00  00              70
00  C0  30  00  00  00              60
00  A0  F6  00  00  00              60
00  00  0D  80  00  00              90
00  00  0B  00  20  00              00
```

Function **Call** and Return

CALL:

```
# This part is just PUSHQ PC*
R[%rsp] <- R[%rsp] – 8
M8[R[%rsp]] <- PC
# Now change the PC
PC <- Dest
```



0x0000

30	00	00	00	**10**	**03**
F4	00	00	00	**00**	00
00	30	00	00	**00**	00
20	F7	00	00	**00**	00
00	C0	30	00	**00**	00
00	A0	F6	00	**00**	00
00	00	0D	**80**	**00**	00
00	00	0B	**00**	**20**	00

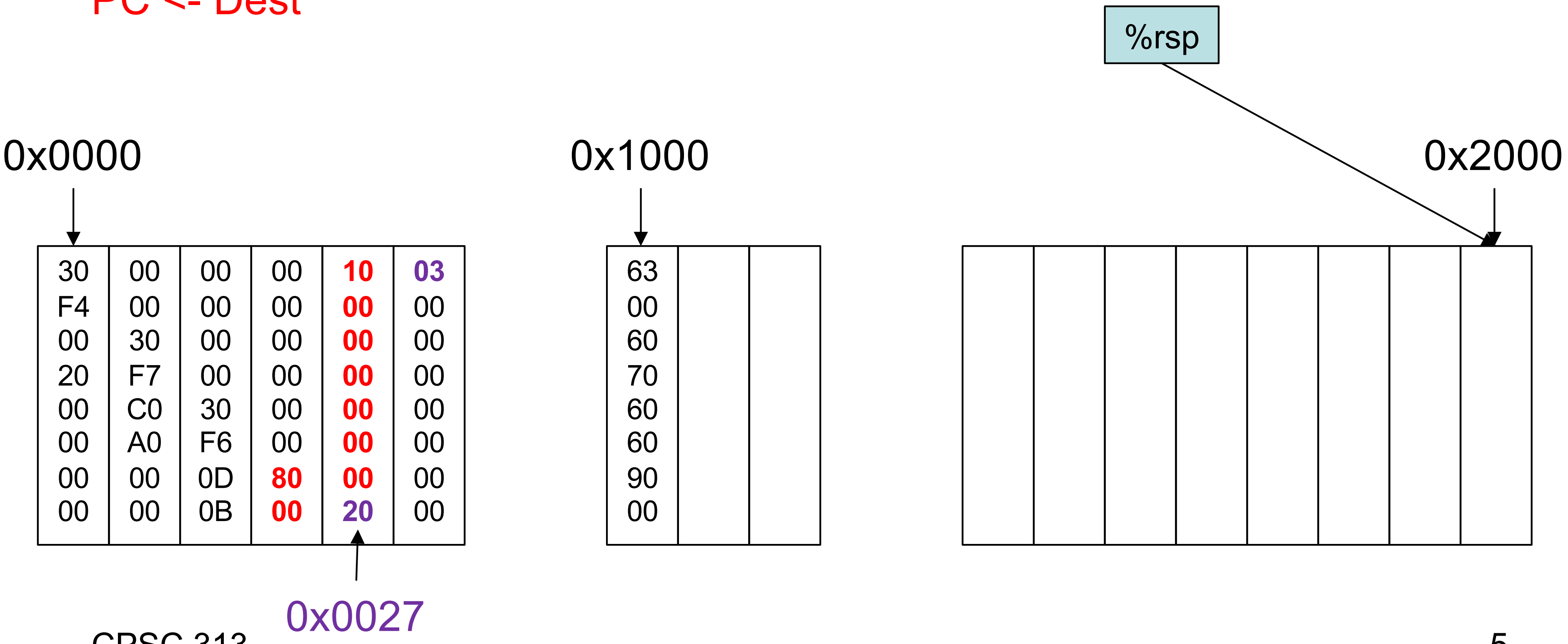
0x1000

63		
00		
60		
70		
60		
60		
90		
00		

Function **Call** and Return

CALL:

```
# This part is just PUSHQ PC*
R[%rsp] <- R[%rsp] – 8
$M_8$[R[%rsp]] <- PC
# Now change the PC
PC <- Dest
```



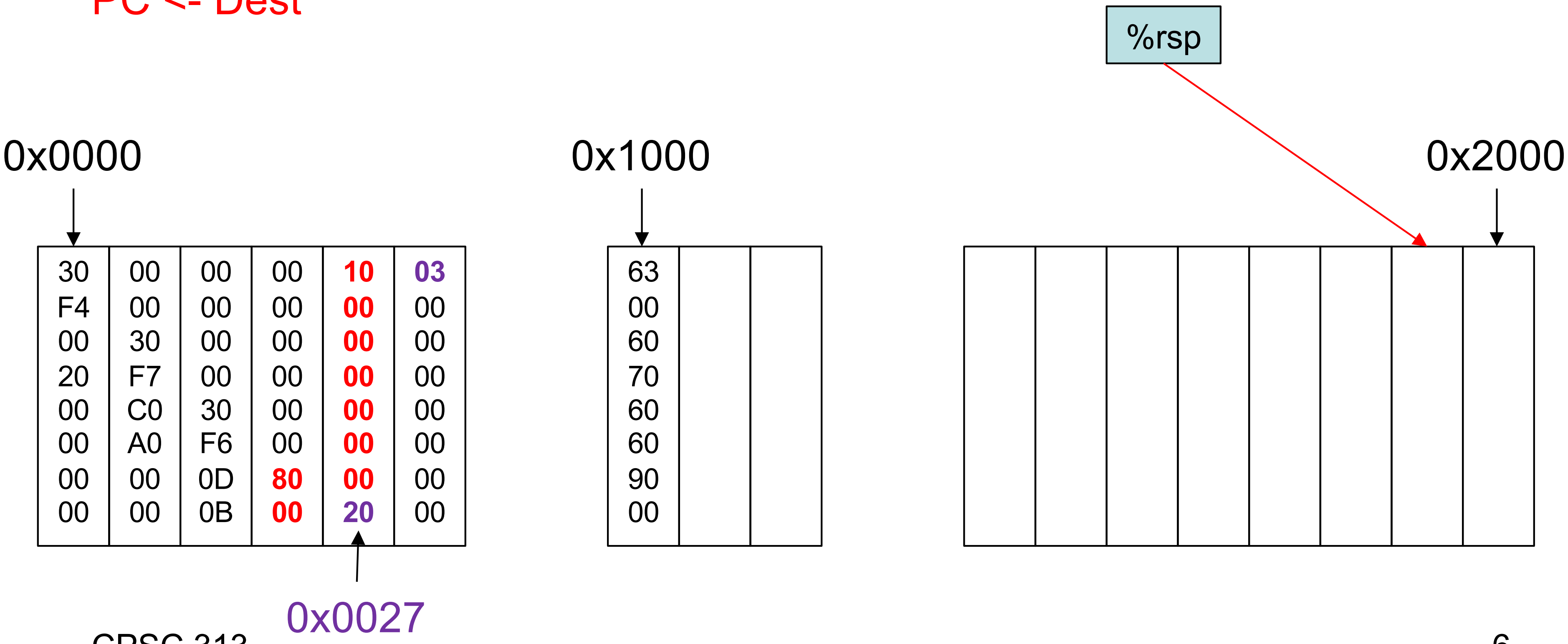
30	00	00	00	**10**	**03**
F4	00	00	00	**00**	00
00	30	00	00	**00**	00
20	F7	00	00	**00**	00
00	C0	30	00	**00**	00
00	A0	F6	00	**00**	00
00	00	0D	**80**	**00**	00
00	00	0B	**00**	**20**	00

63		
00		
60		
70		
60		
60		
90		
00		

Function **Call** and Return

CALL:

```
# This part is just PUSHQ PC*
R[%rsp] <- R[%rsp] – 8
M_8[R[%rsp]] <- PC
# Now change the PC
PC <- Dest
```



0x0000

30	00	00	00	**10**	**03**
F4	00	00	00	**00**	00
00	30	00	00	**00**	00
20	F7	00	00	**00**	00
00	C0	30	00	**00**	00
00	A0	F6	00	**00**	00
00	00	0D	**80**	**00**	00
00	00	0B	**00**	**20**	00

0x1000

63		
00		
60		
70		
60		
60		
90		
00		

Function **Call** and Return

CALL:

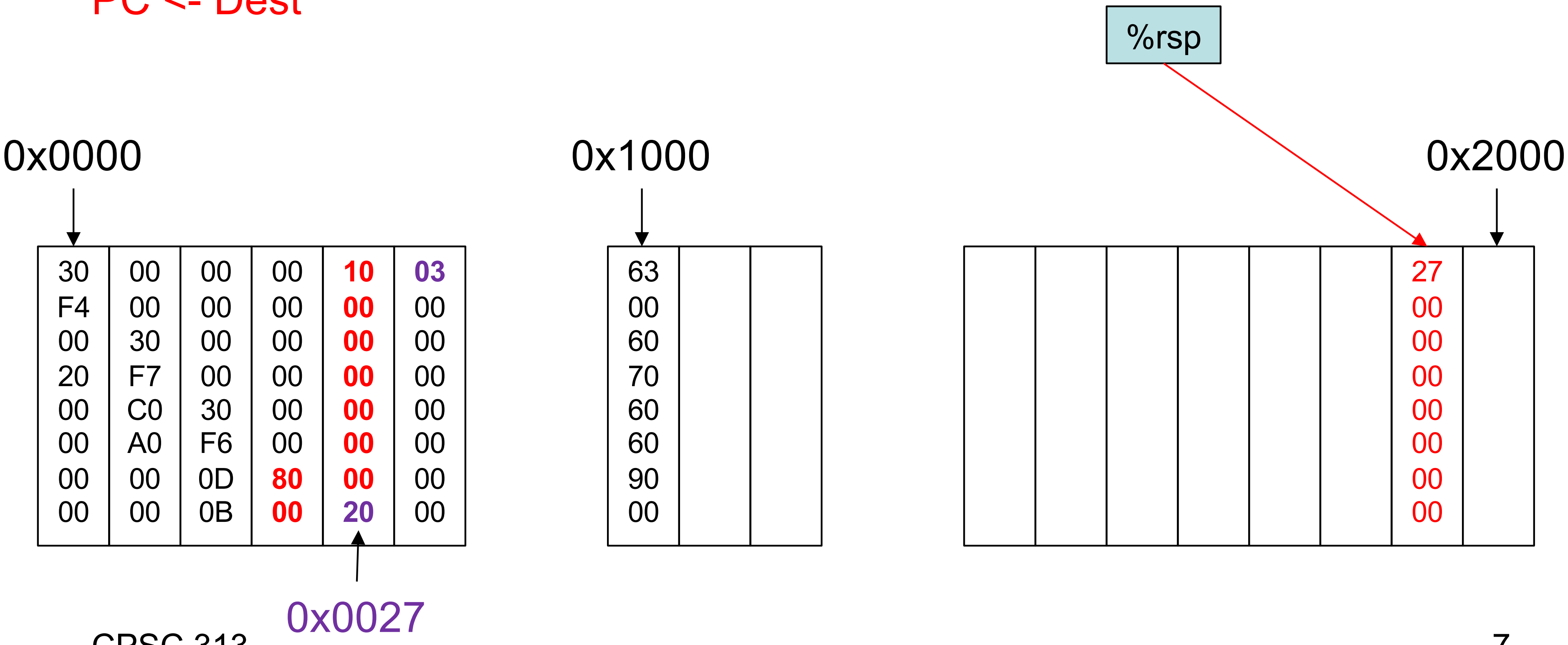
# This part is just PUSHQ PC*

R[%rsp] <- R[%rsp] – 8

M_8[R[%rsp]] <- PC

# Now change the PC

PC <- Dest



30	00	00	00	**10**	**03**
F4	00	00	00	**00**	00
00	30	00	00	**00**	00
20	F7	00	00	**00**	00
00	C0	30	00	**00**	00
00	A0	F6	00	**00**	00
00	00	0D	**80**	**00**	00
00	00	0B	**00**	**20**	00

63		
00		
60		
70		
60		
60		
90		
00		

						27	
						00	
						00	
						00	
						00	
						00	
						00	
						00	

Function **Call** and Return

CALL:

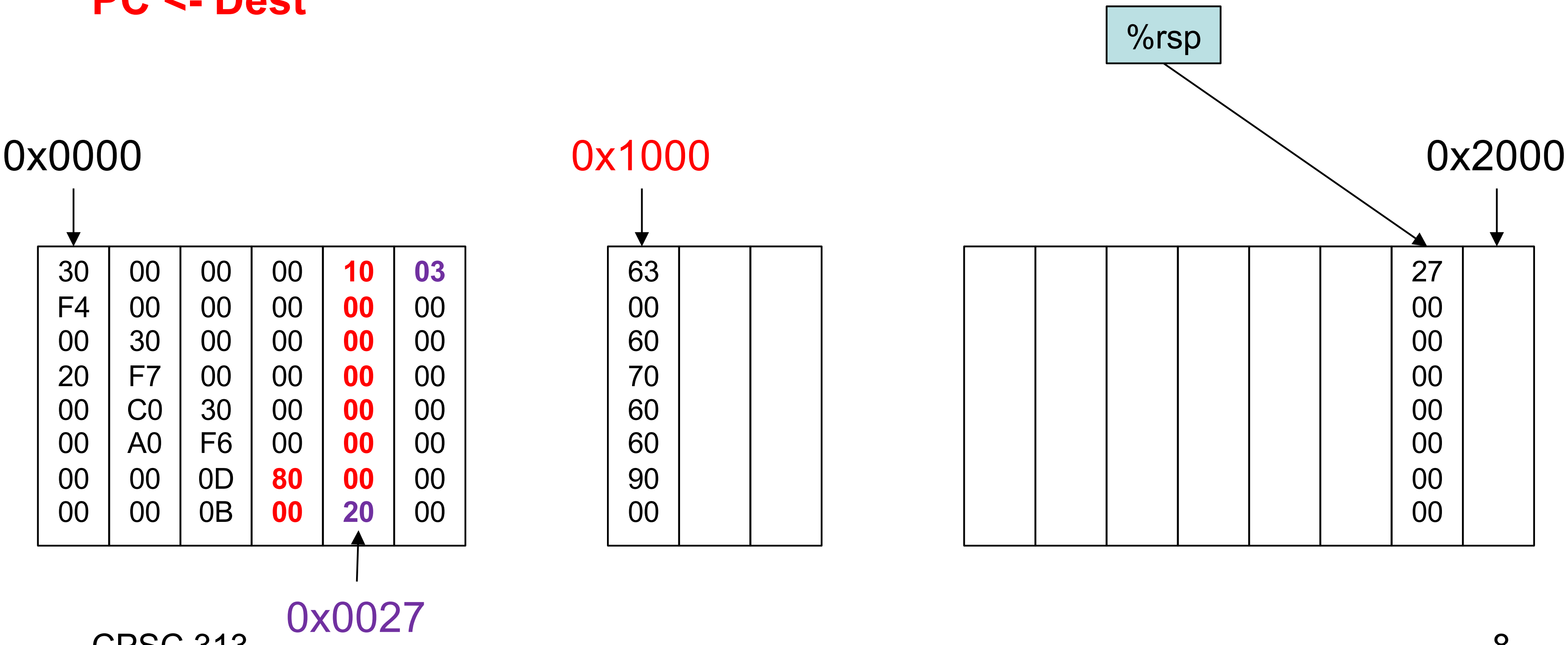
# This part is just PUSHQ PC*

R[%rsp] <- R[%rsp] – 8

M_8[R[%rsp]] <- PC

# Now change the PC

PC <- Dest



30	00	00	00	**10**	**03**
F4	00	00	00	**00**	00
00	30	00	00	**00**	00
20	F7	00	00	**00**	00
00	C0	30	00	**00**	00
00	A0	F6	00	**00**	00
00	00	0D	**80**	**00**	00
00	00	0B	**00**	**20**	00

63		
00		
60		
70		
60		
60		
90		
00		

						27	
						00	
						00	
						00	
						00	
						00	
						00	
						00	

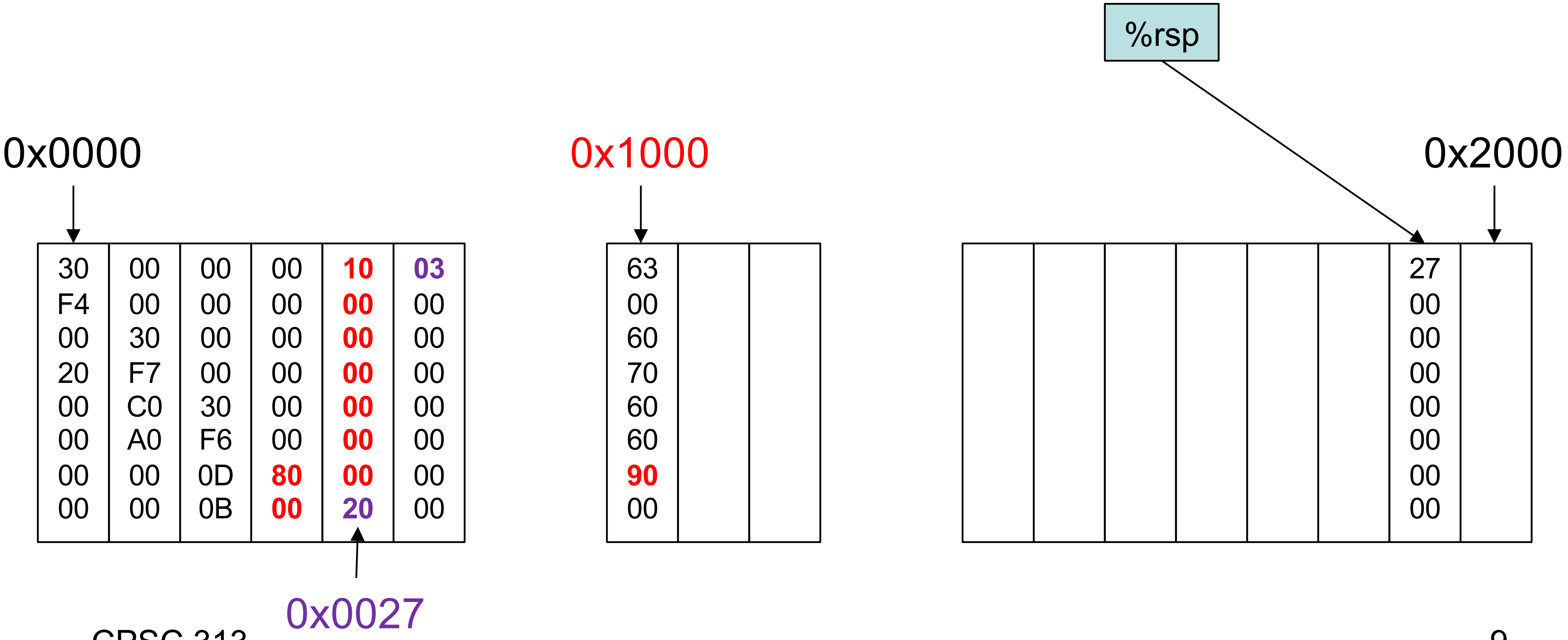
Function Call and **Return**

RET:

# This part is just POPQ PC*

PC <- M_8[R[%rsp]]

R[%rsp] <- R[%rsp] + 8



0x0000

30	00	00	00	**10**	**03**
F4	00	00	00	**00**	00
00	30	00	00	**00**	00
20	F7	00	00	**00**	00
00	C0	30	00	**00**	00
00	A0	F6	00	**00**	00
00	00	0D	**80**	**00**	00
00	00	0B	**00**	**20**	00

0x1000

63		
00		
60		
70		
60		
60		
90		
00		

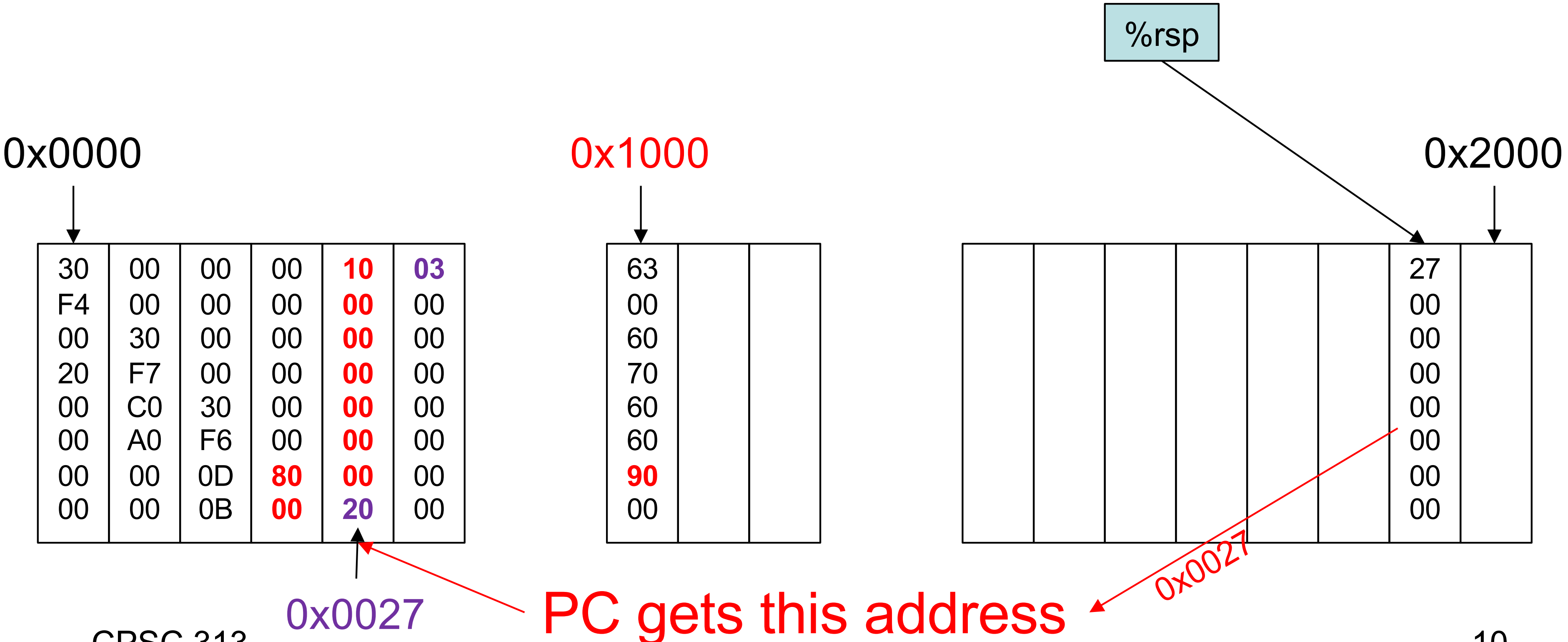
0x2000

						27	
						00	
						00	
						00	
						00	
						00	
						00	
						00	

Function Call and **Return**

RET:

```
# This part is just POPQ PC*
PC <- $M_8$[R[%rsp]]
R[%rsp] <- R[%rsp] + 8
```



30	00	00	00	**10**	**03**
F4	00	00	00	**00**	00
00	30	00	00	**00**	00
20	F7	00	00	**00**	00
00	C0	30	00	**00**	00
00	A0	F6	00	**00**	00
00	00	0D	**80**	**00**	00
00	00	0B	**00**	**20**	00

63		
00		
60		
70		
60		
60		
90		
00		

						27	
						00	
						00	
						00	
						00	
						00	
						00	
						00	

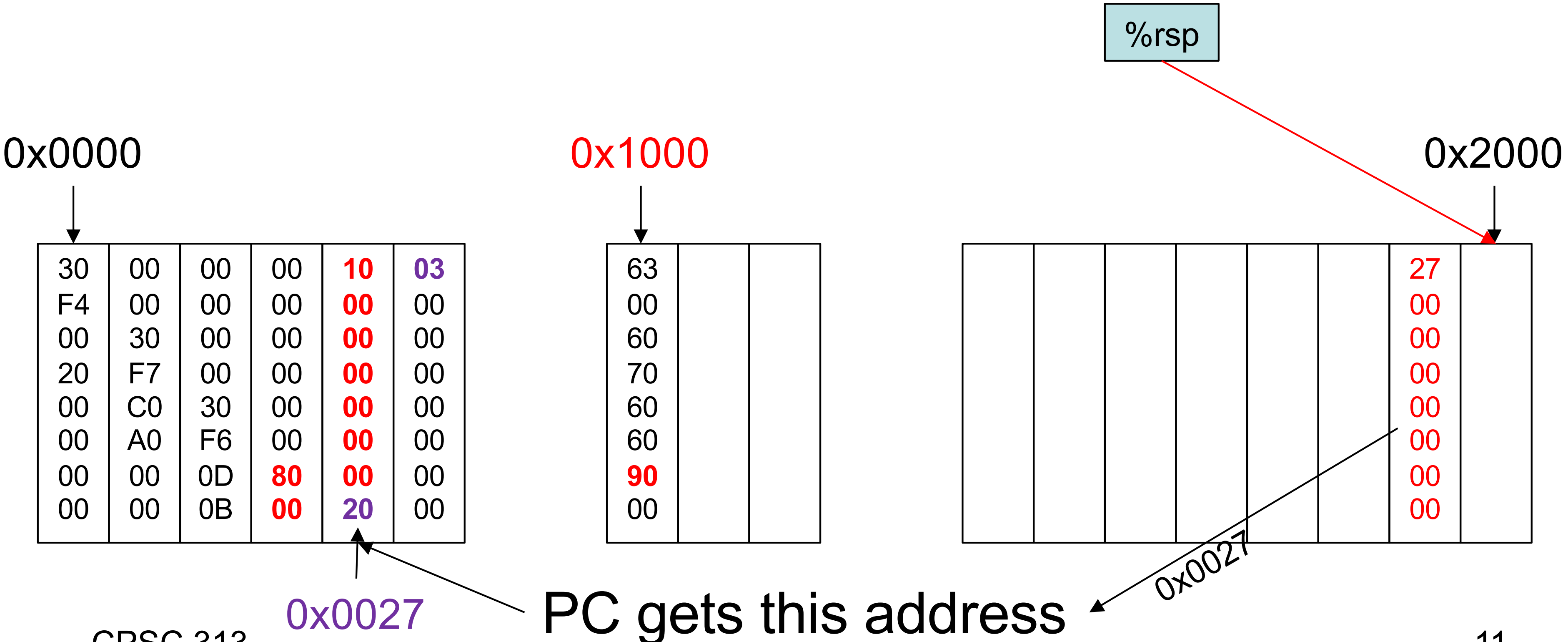
Function Call and **Return**

RET:

# This part is just POPQ PC*

PC <- M_8[R[%rsp]]

R[%rsp] <- R[%rsp] + 8



30	00	00	00	**10**	**03**
F4	00	00	00	**00**	00
00	30	00	00	**00**	00
20	F7	00	00	**00**	00
00	C0	30	00	**00**	00
00	A0	F6	00	**00**	00
00	00	0D	**80**	**00**	00
00	00	0B	**00**	**20**	00

63		
00		
60		
70		
60		
60		
90		
00		

						27	
						00	
						00	
						00	
						00	
						00	
						00	
						00	

PC gets this address