

ARM Cortex-M

Interrupt

Embedded Systems with ARM Cortex-M
Microcontrollers in Assembly Language and C

Dr. Yifeng Zhu

Polling vs Interrupt



Copyright © Ron Leishman * <http://ToonClips.com/9845>

Polling:

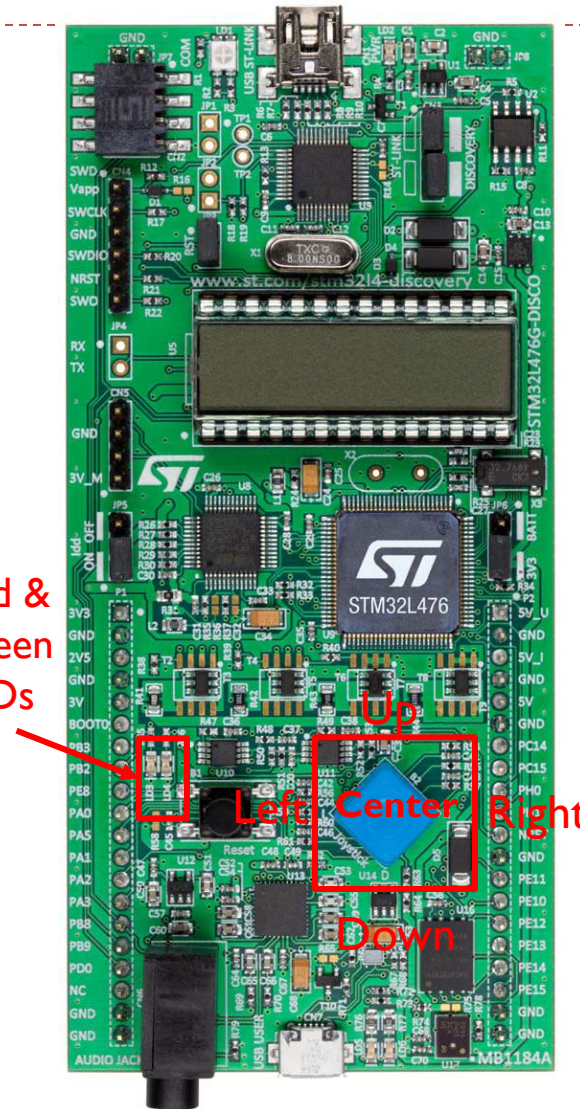
You **pick up the phone every few seconds** to check whether you are getting a call.

Interrupt:

Do whatever you should do and pick up the phone **when it rings**.

STM32L4 Discovery Kit

Red & Green LEDs

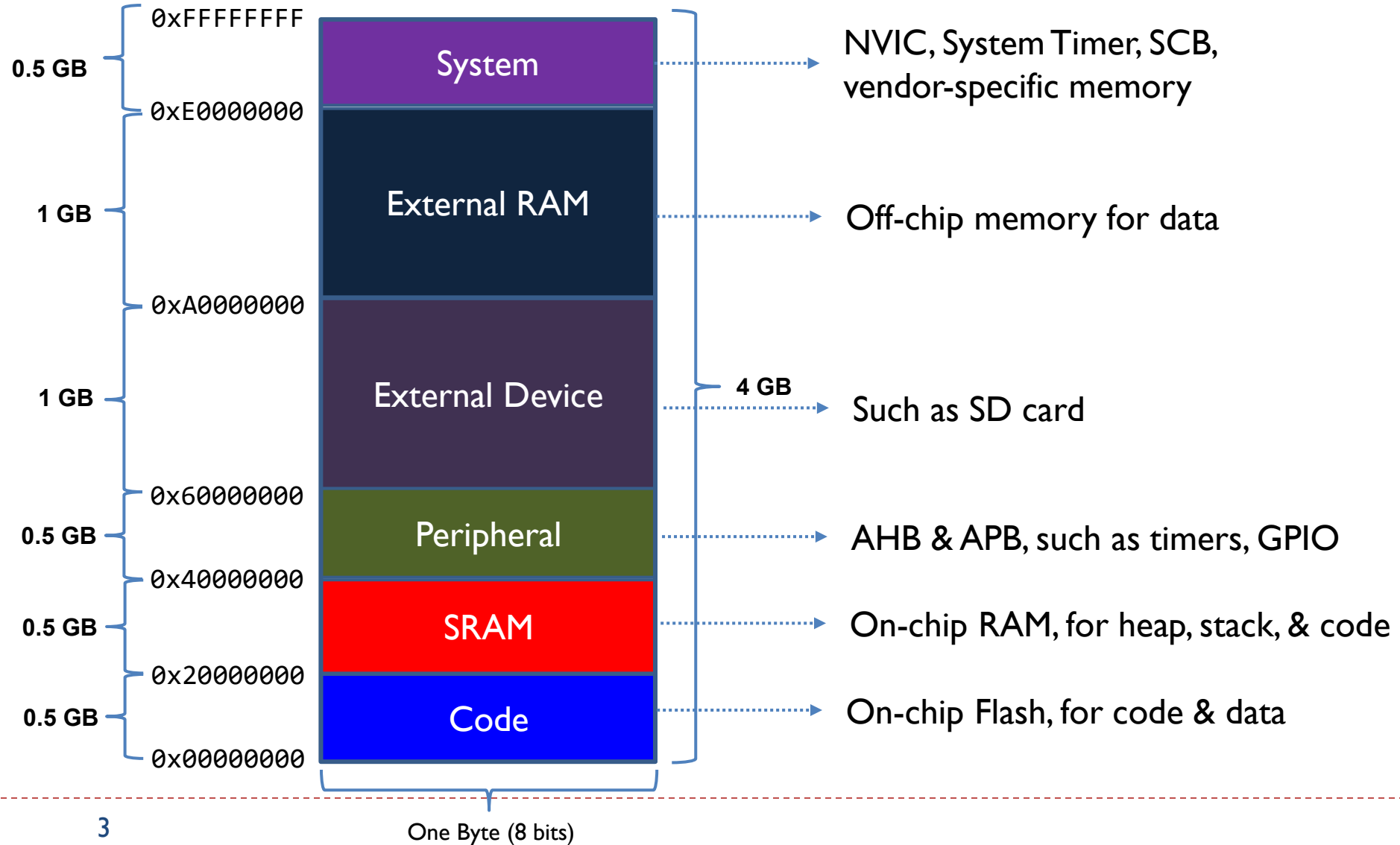


```
// Polling method
while (1) {
    read_button_input;
    if (pushed)
        exit;
}

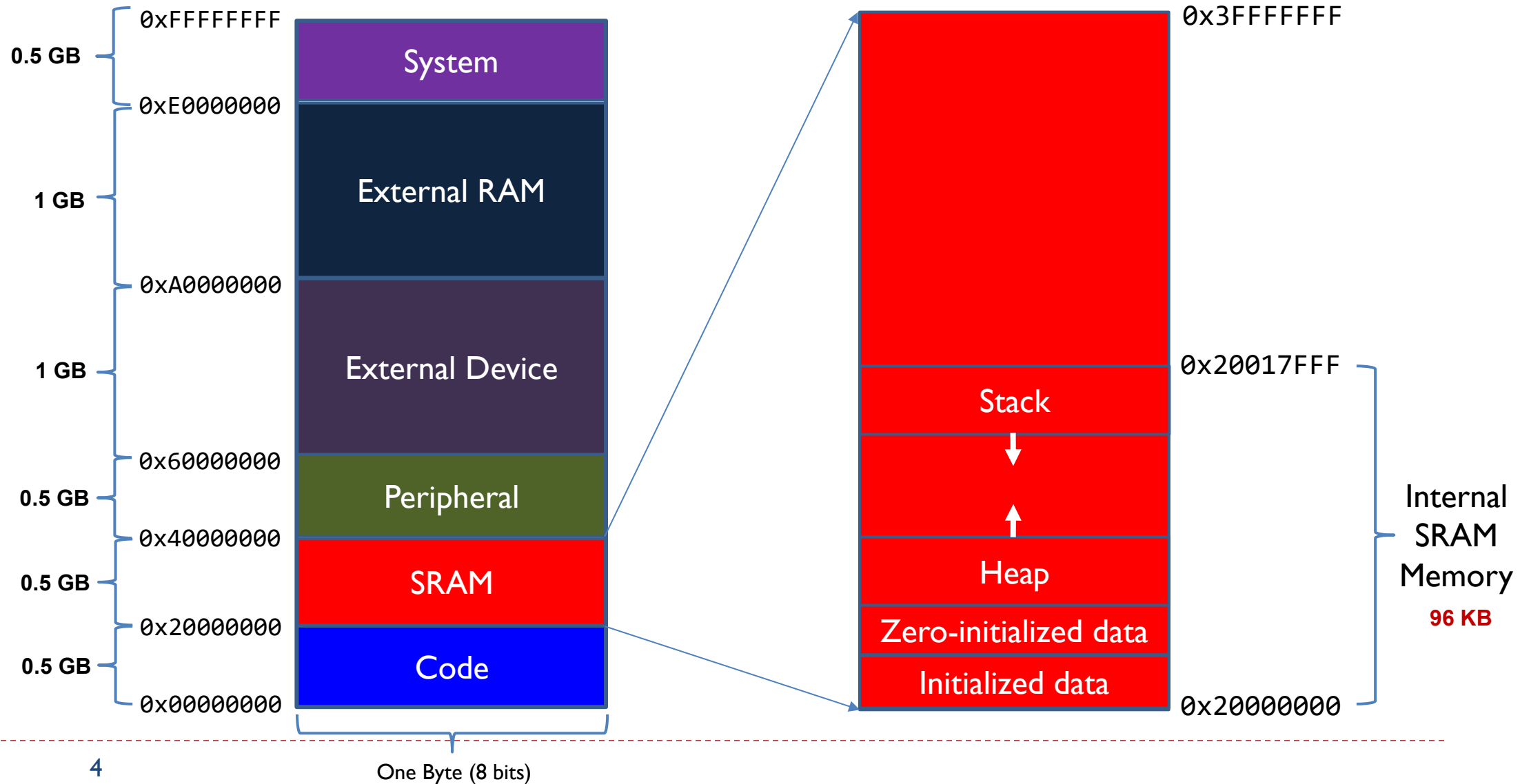
turn_on_LED;
```

```
// Interrupt method
interrupt_handler(){
    turn_on_LED;
    exit;
}
```

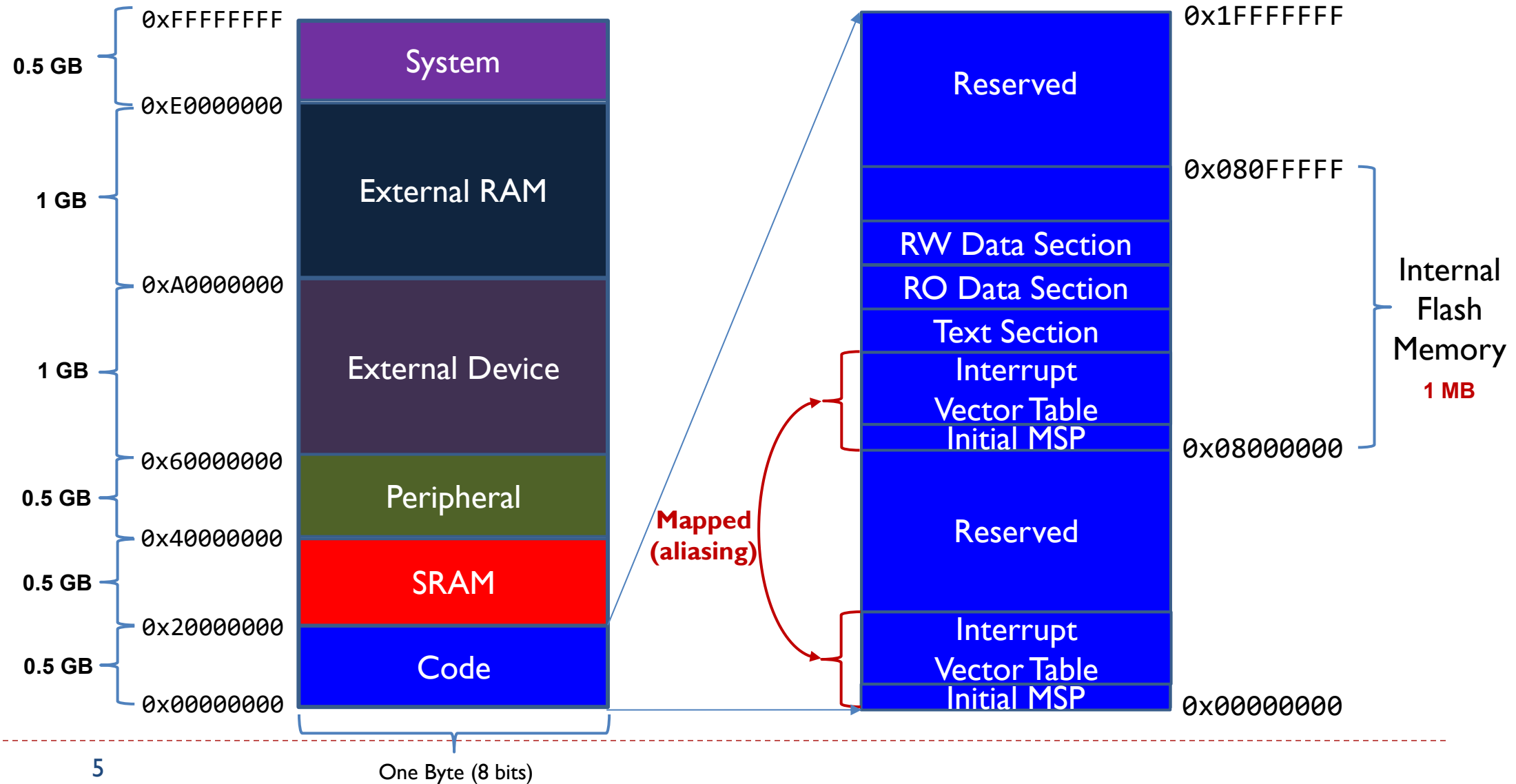
Memory Map of Cortex-M4



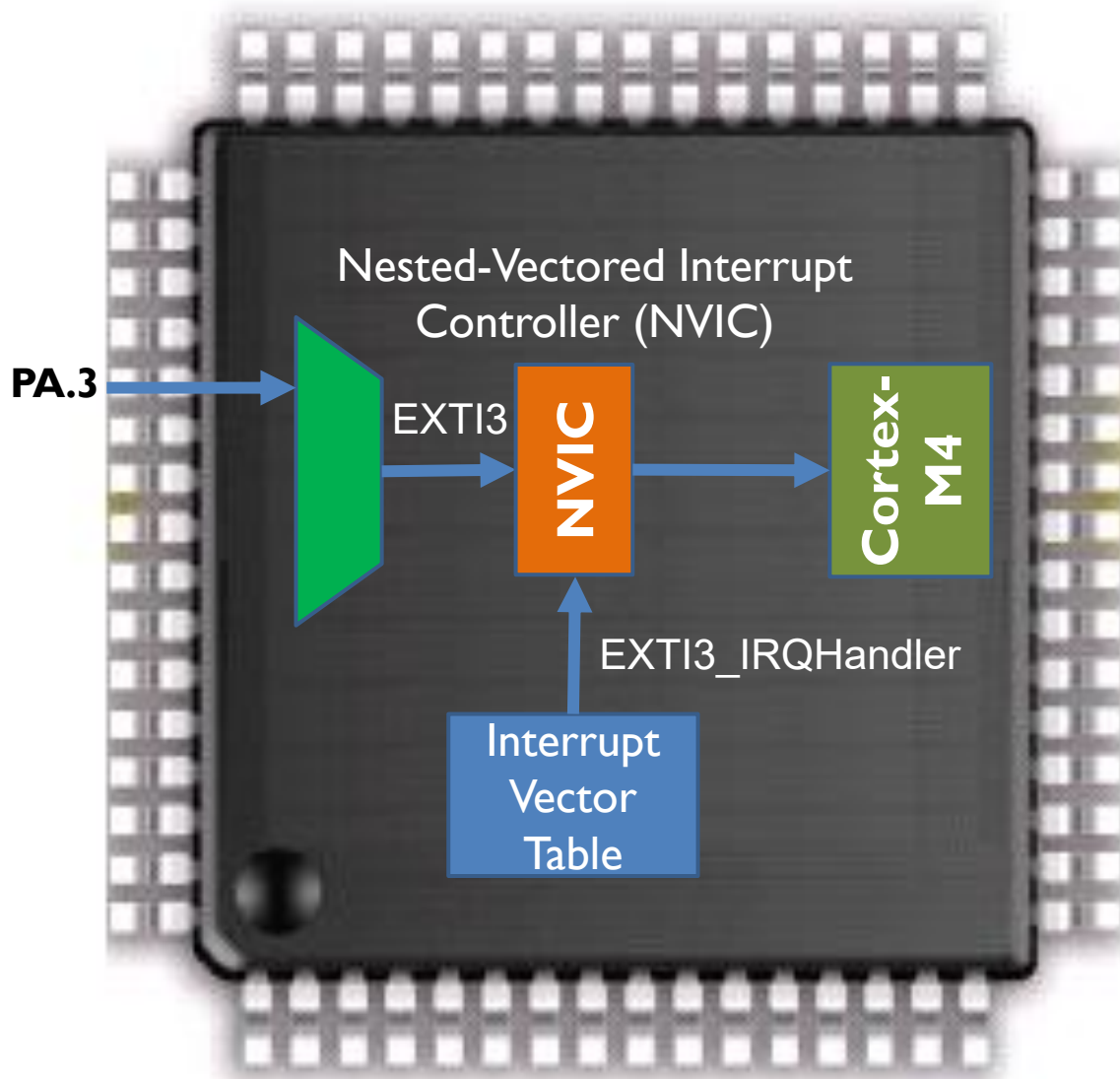
Data Memory



Instruction Memory



Interrupt Vector Table



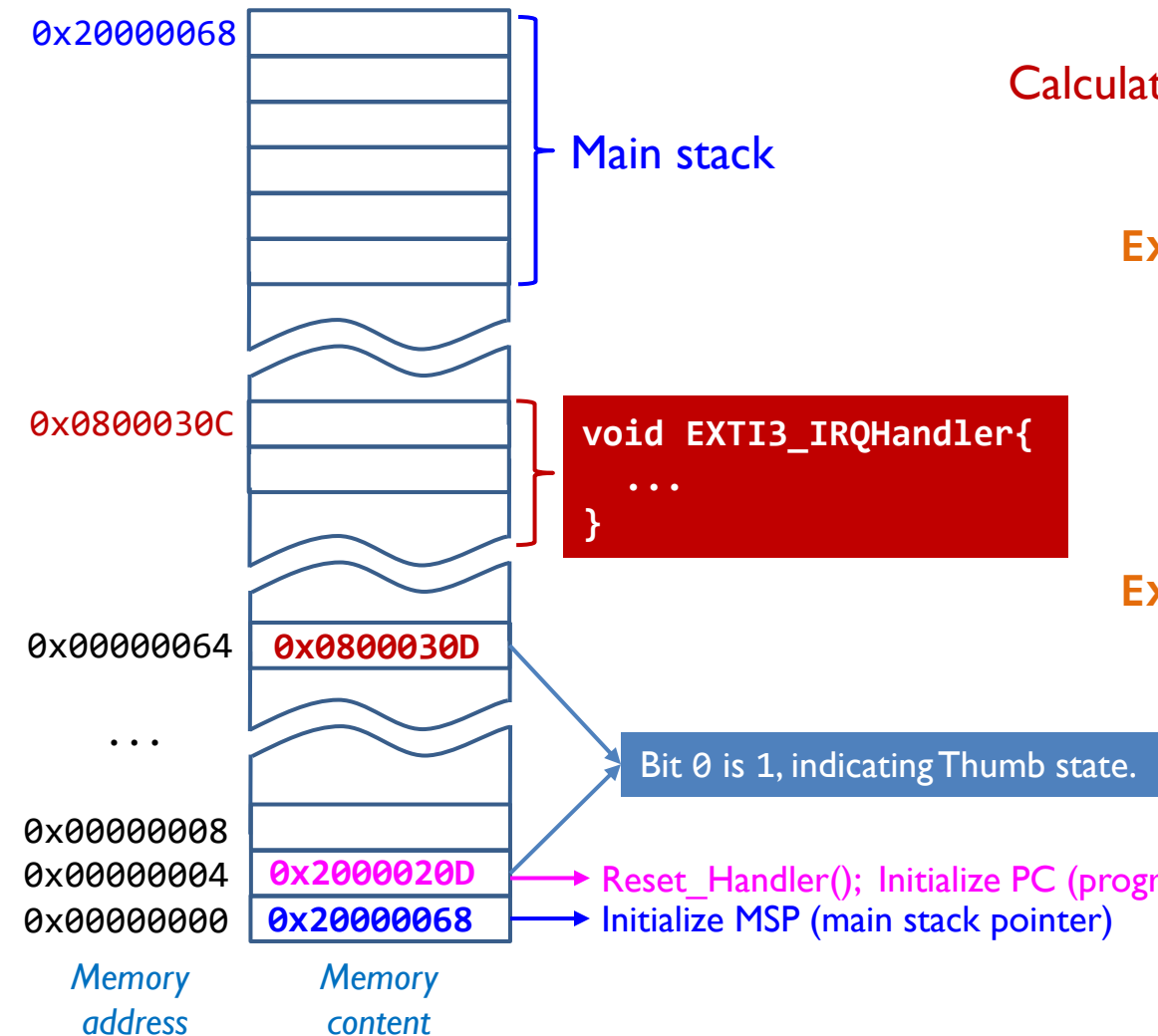
Interrupt Vector Table

Address of ISR 1

Interrupt Number (8 bits)	Memory Address of ISR (32 bits)
1	Interrupt Service Routine for interrupt 1
2	Interrupt Service Routine for interrupt 2
3	Interrupt Service Routine for interrupt 3
4	Interrupt Service Routine for interrupt 4
5	Interrupt Service Routine for interrupt 5
...	...

When interrupt x is triggered, jump to the ISR for interrupt x . ($1 \leq x \leq 255$)

Interrupt Vector Table



Calculate the address which holds the address of the ISR for interrupt n:

$$\text{Address of pointer} = 64 + 4 \times n$$

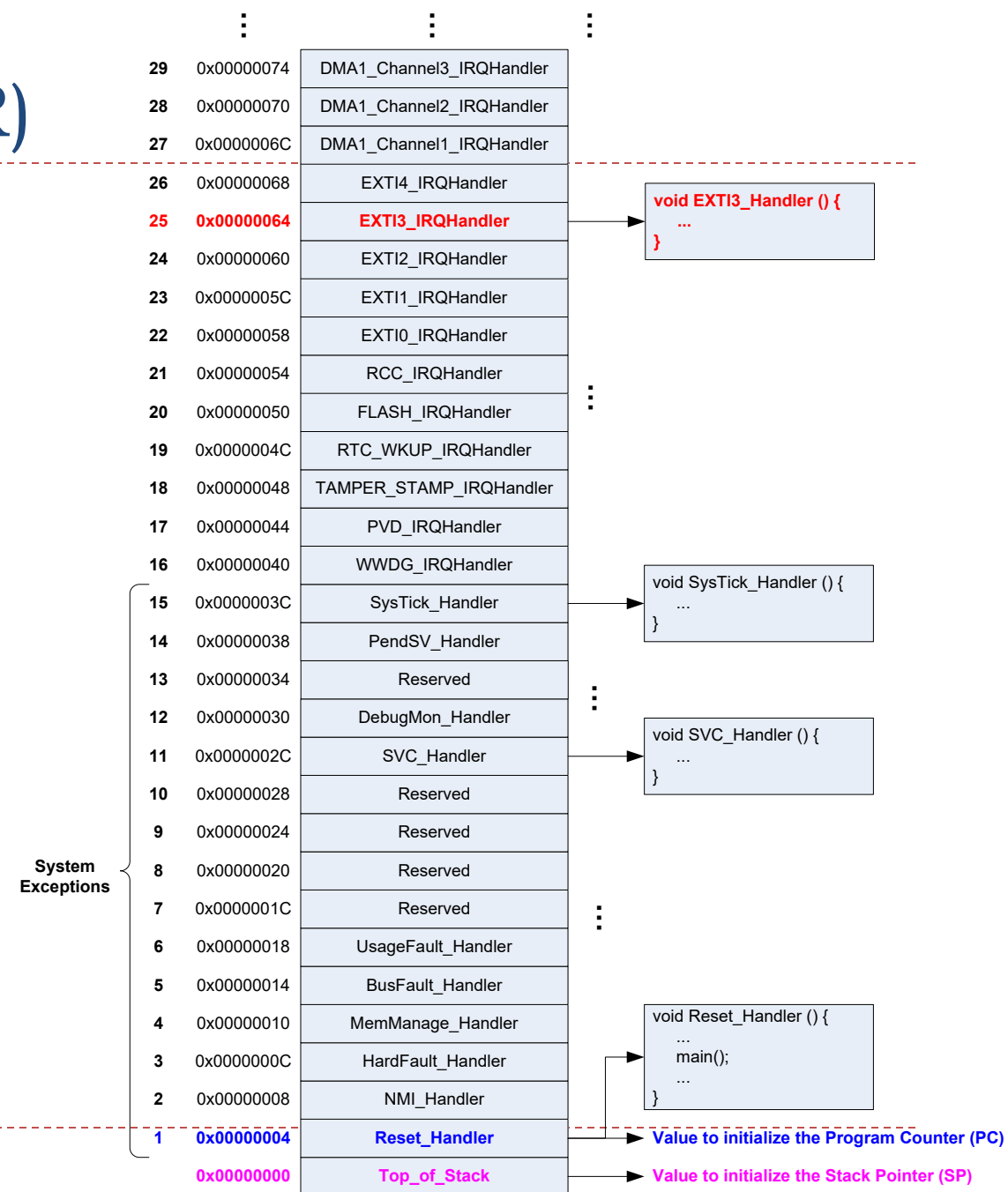
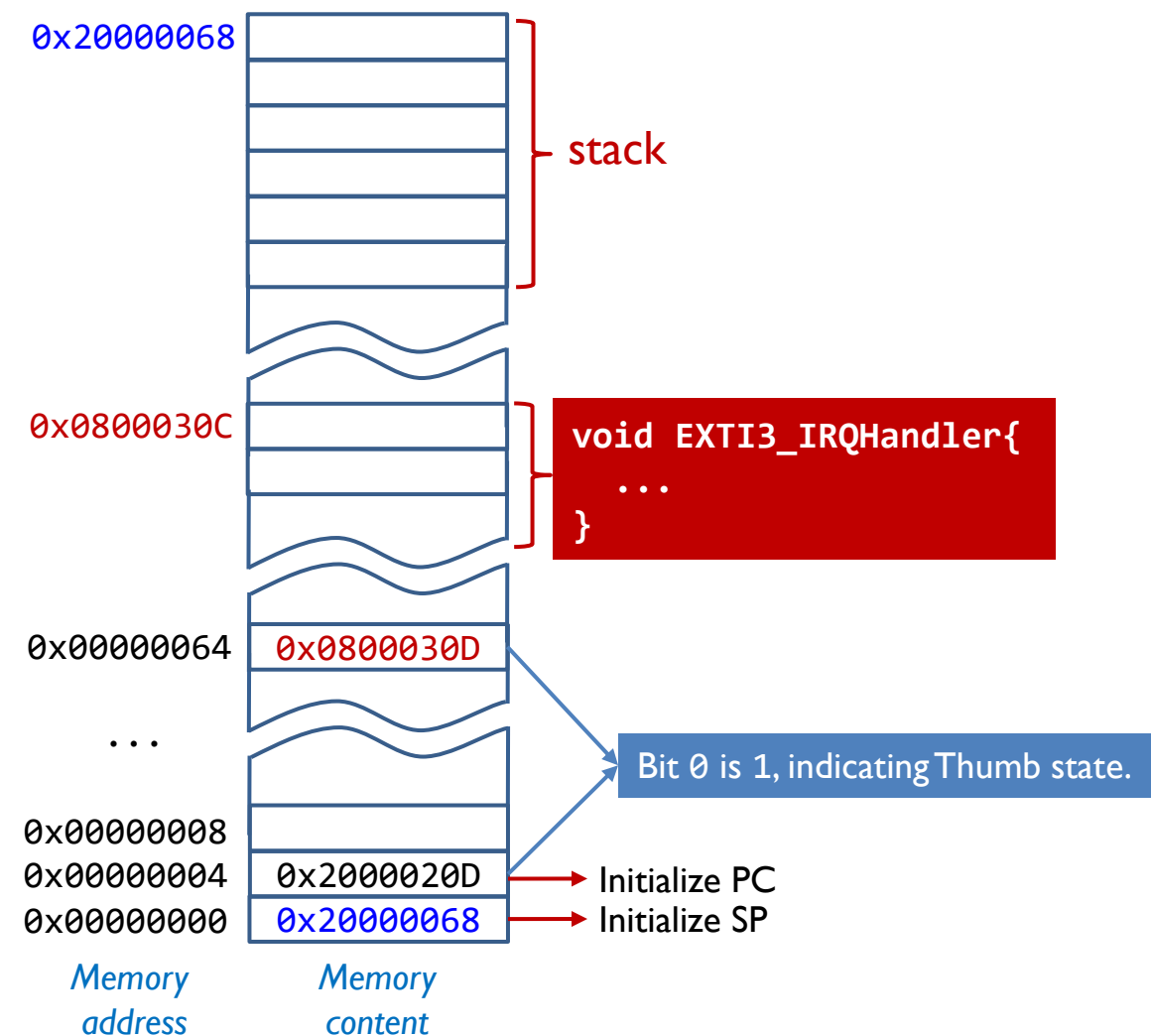
Example 1: EXTI3_IRQn = 9

$$\begin{aligned}\text{Address of pointer to EXTI3 ISR} &= 64 + 4 \times 9 \\ &= 100 \\ &= 0x64\end{aligned}$$

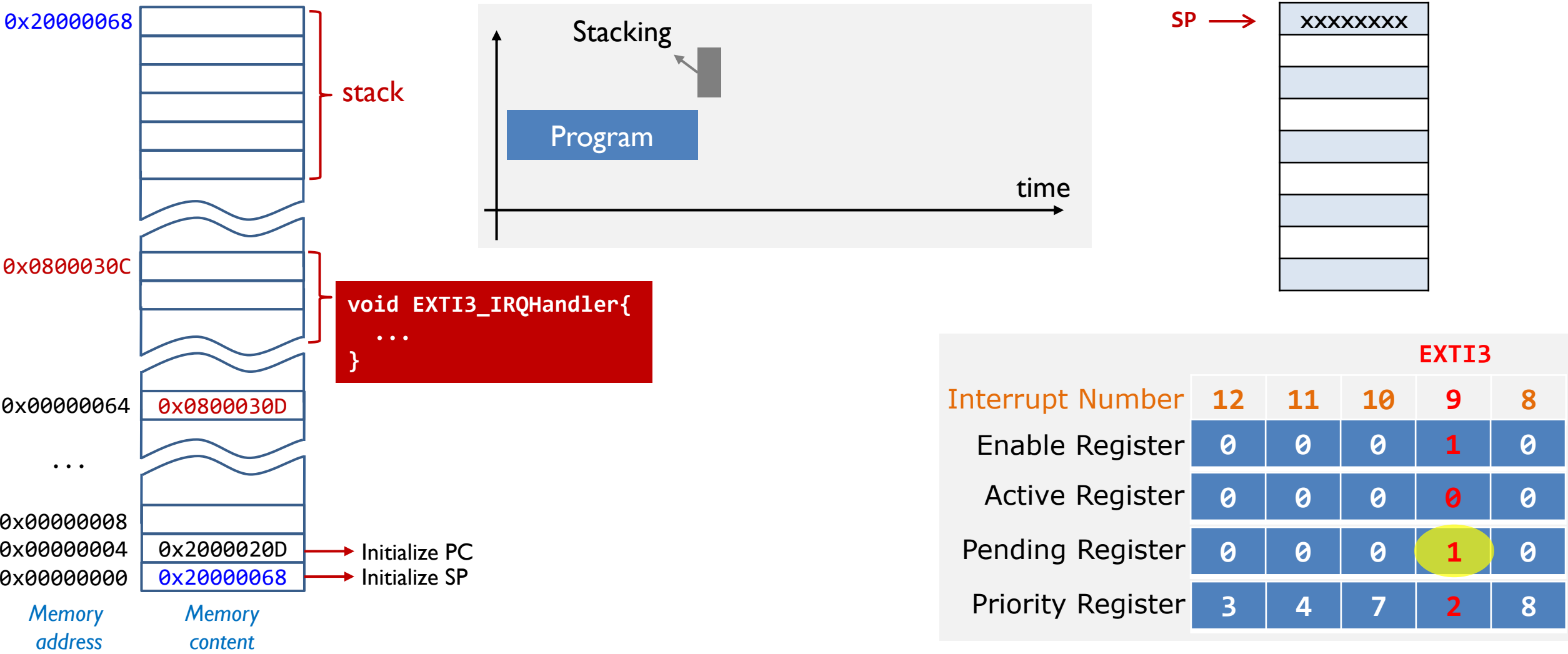
Example 2: SysTick_IRQn = -1

$$\begin{aligned}\text{Address of pointer to SysTick ISR} &= 64 + 4 \times (-1) \\ &= 60 \\ &= 0x3C\end{aligned}$$

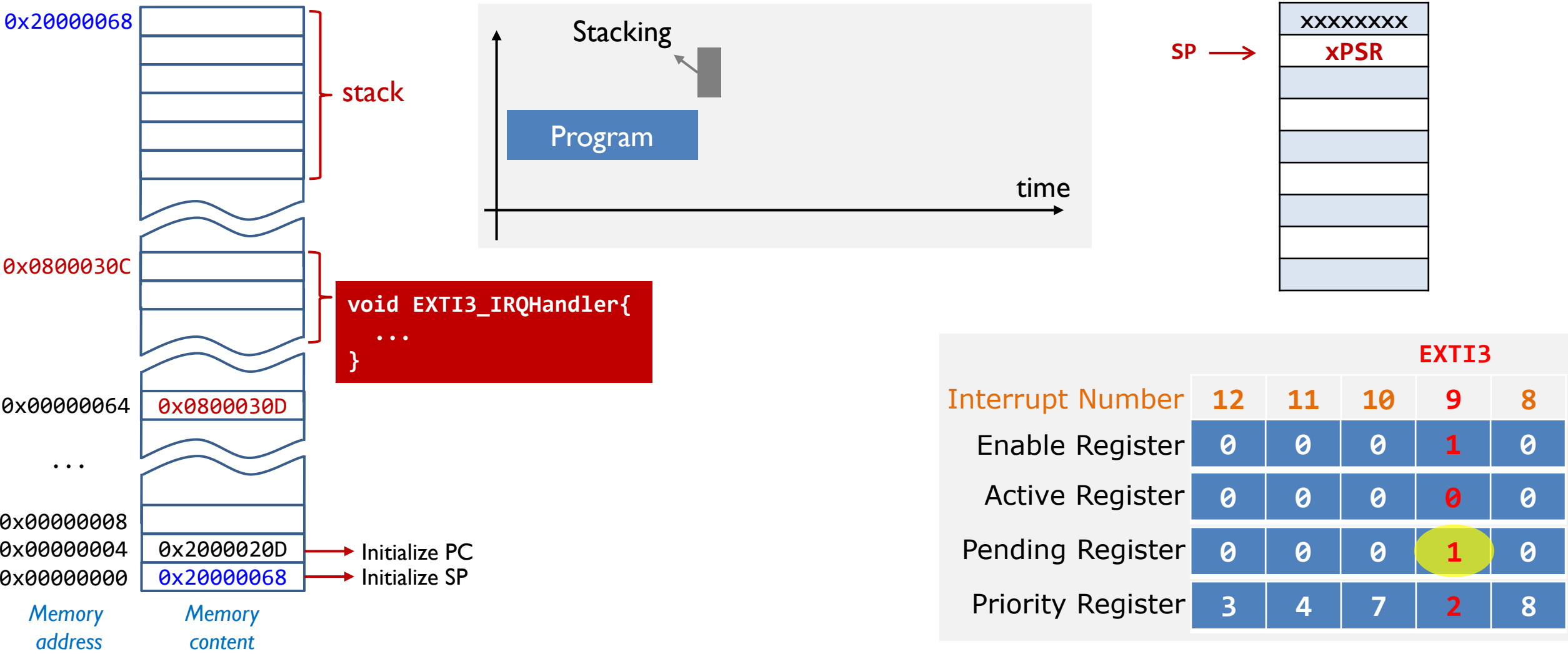
Interrupt Service Routine (ISR)



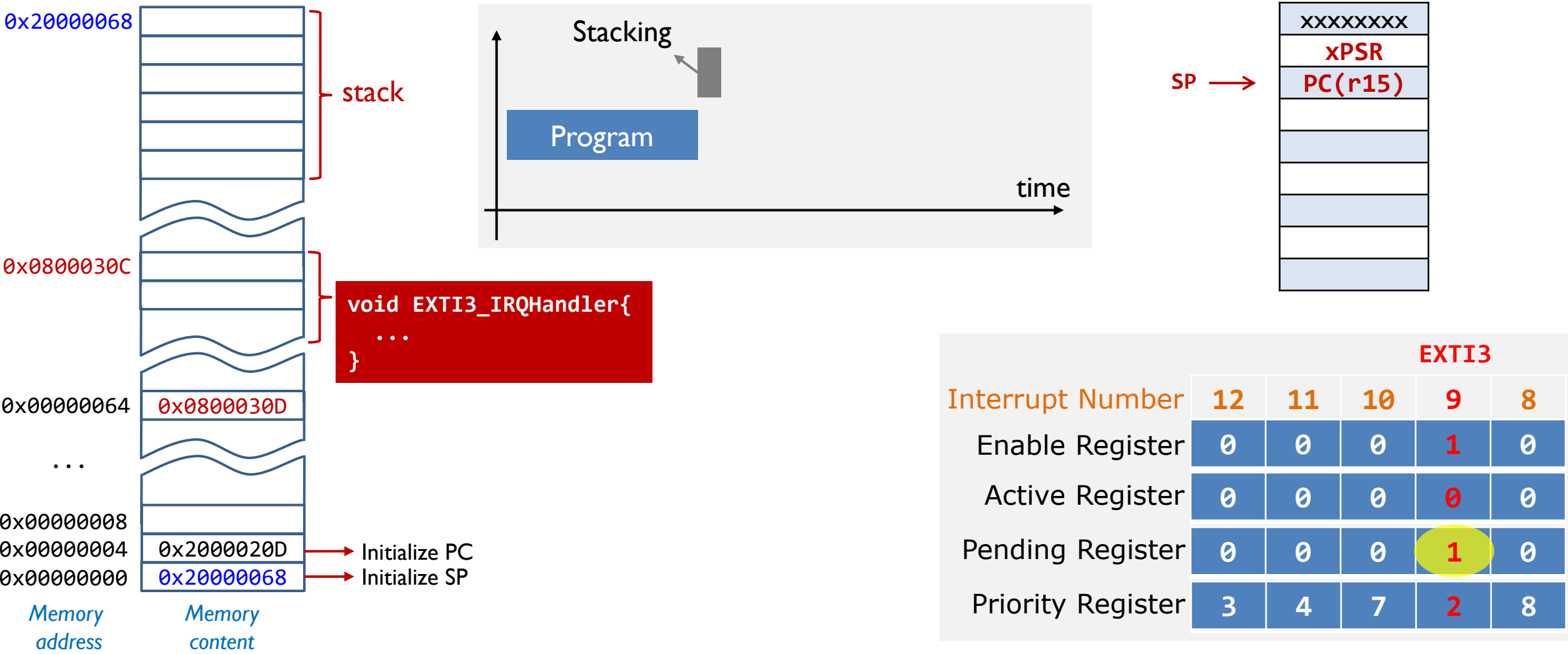
Single Interrupt



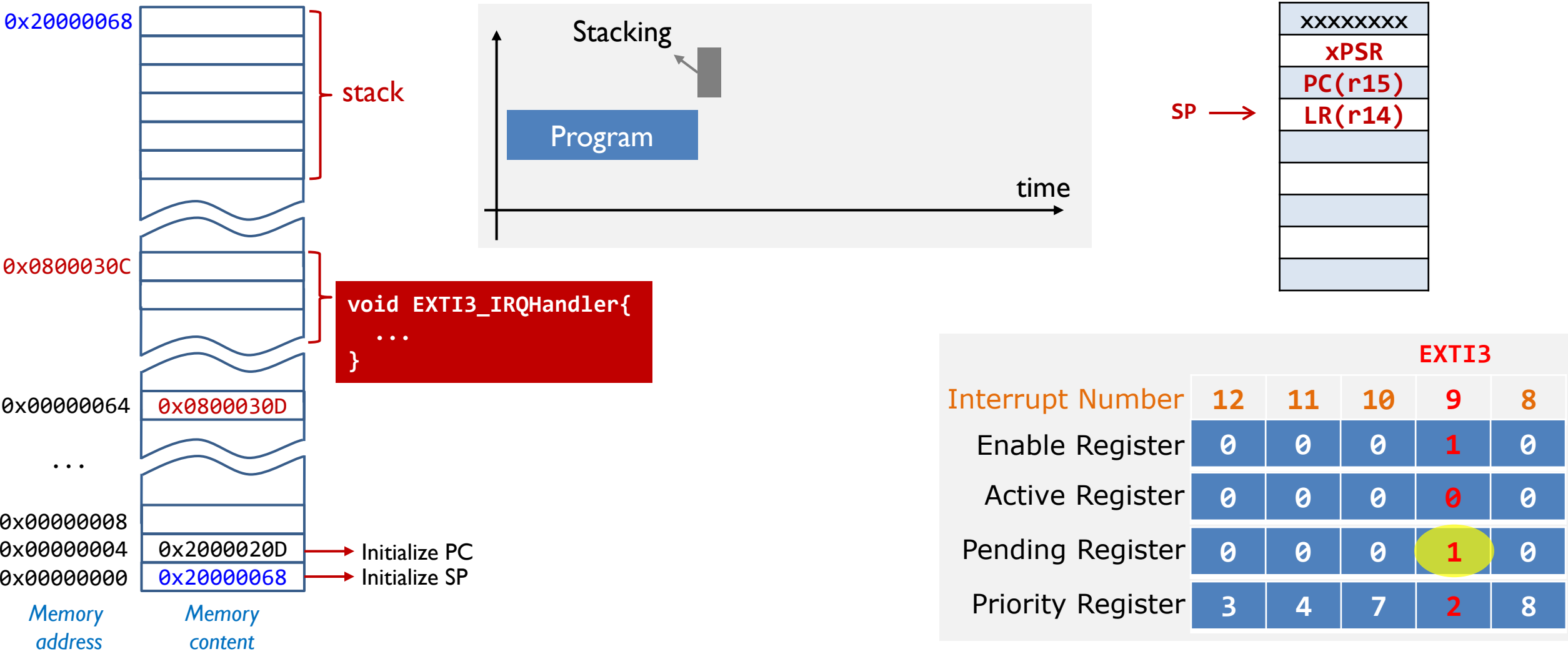
Single Interrupt



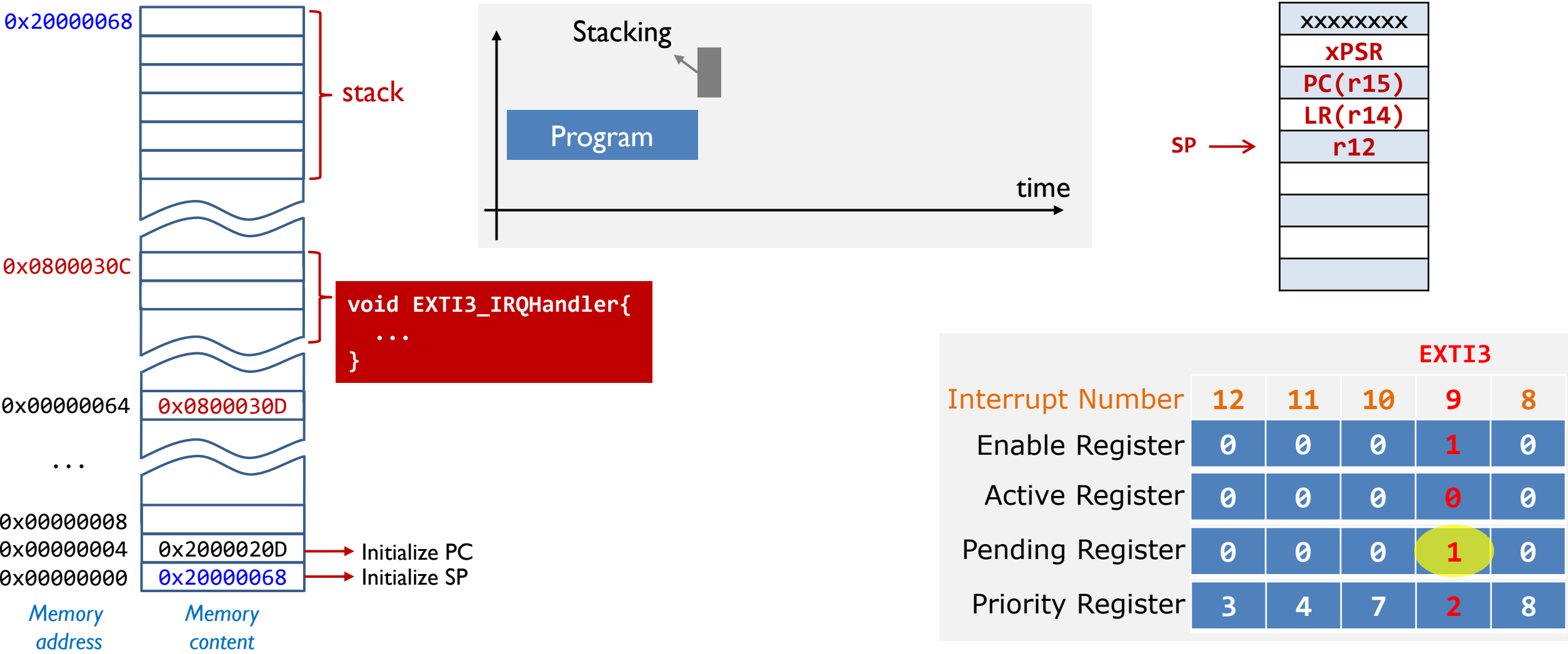
Single Interrupt



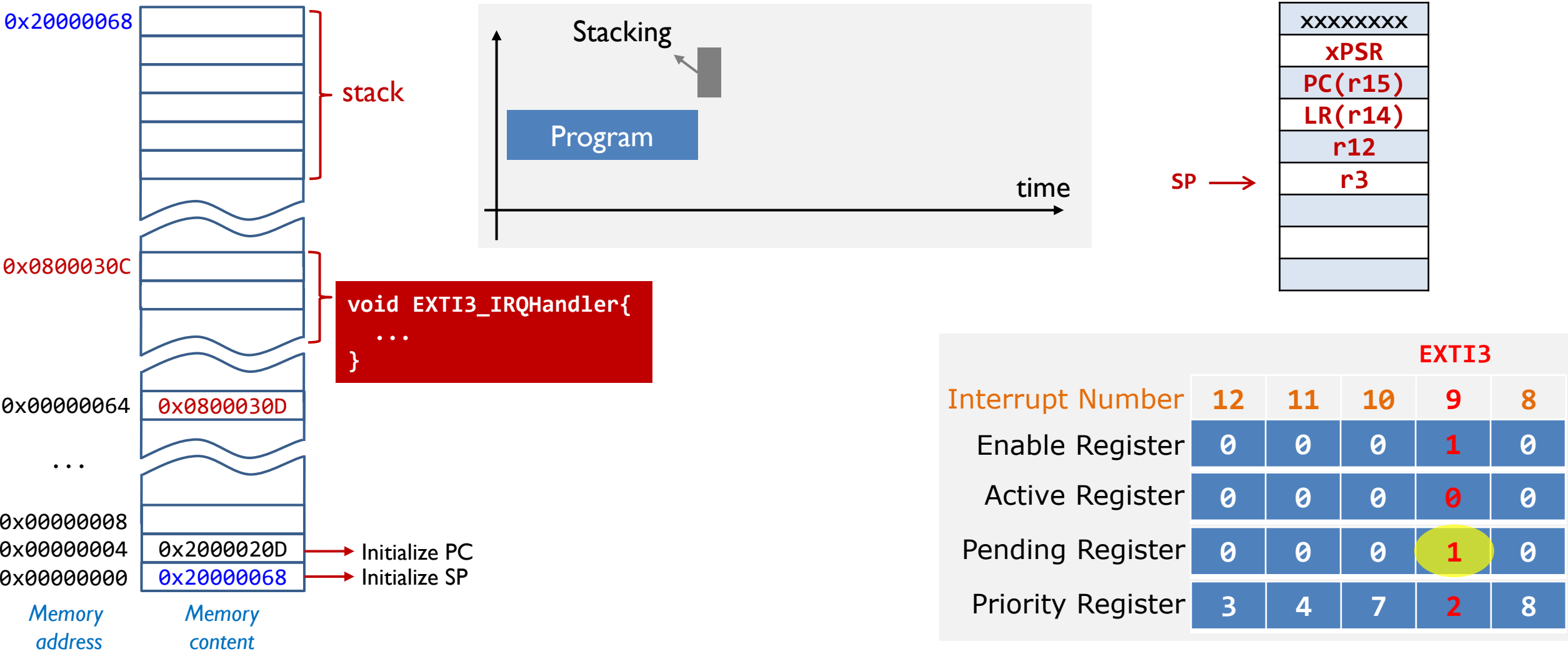
Single Interrupt



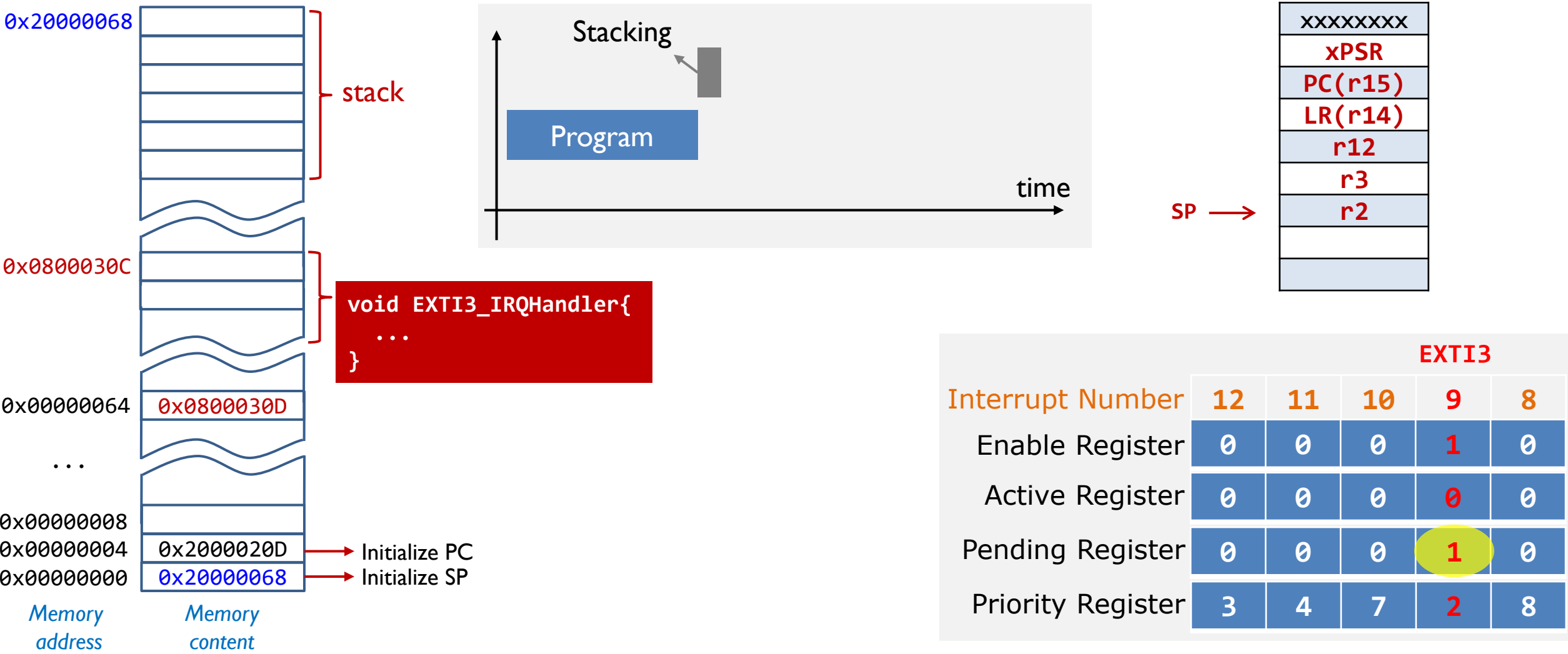
Single Interrupt



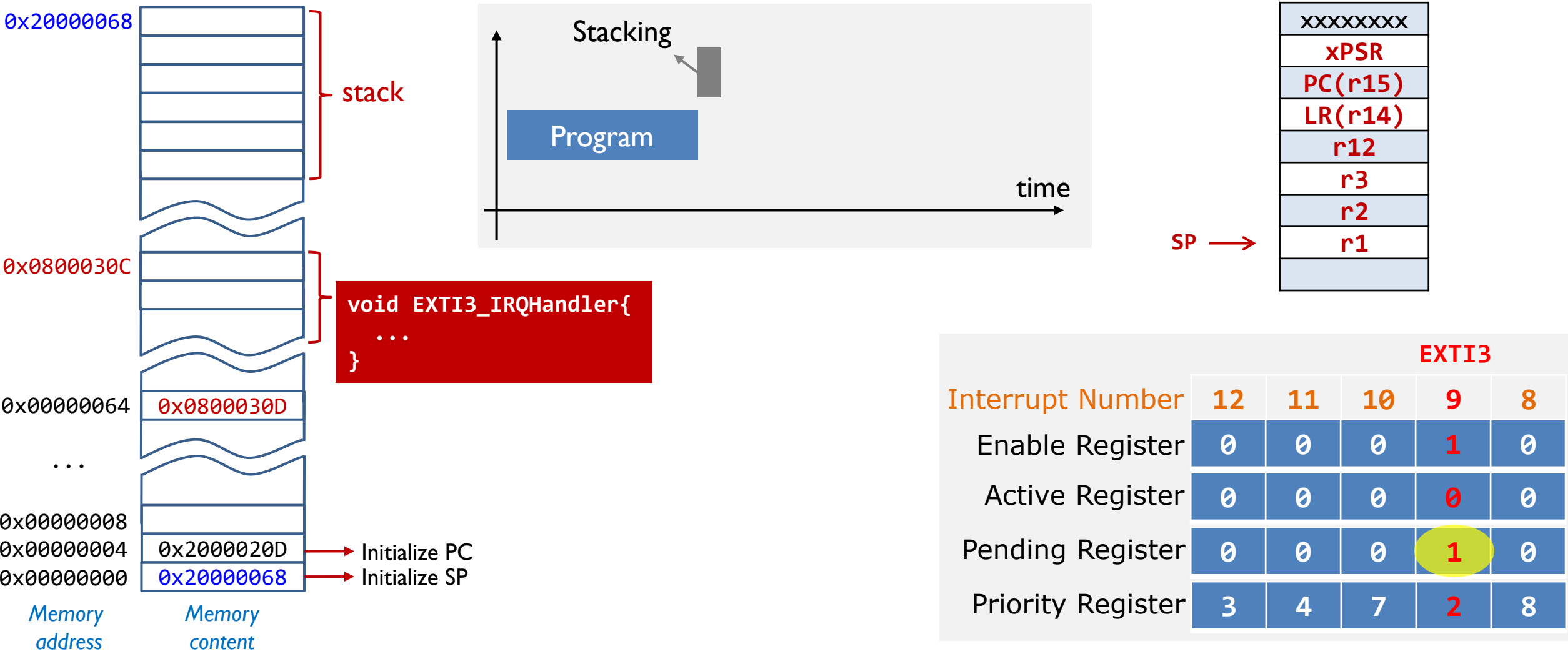
Single Interrupt



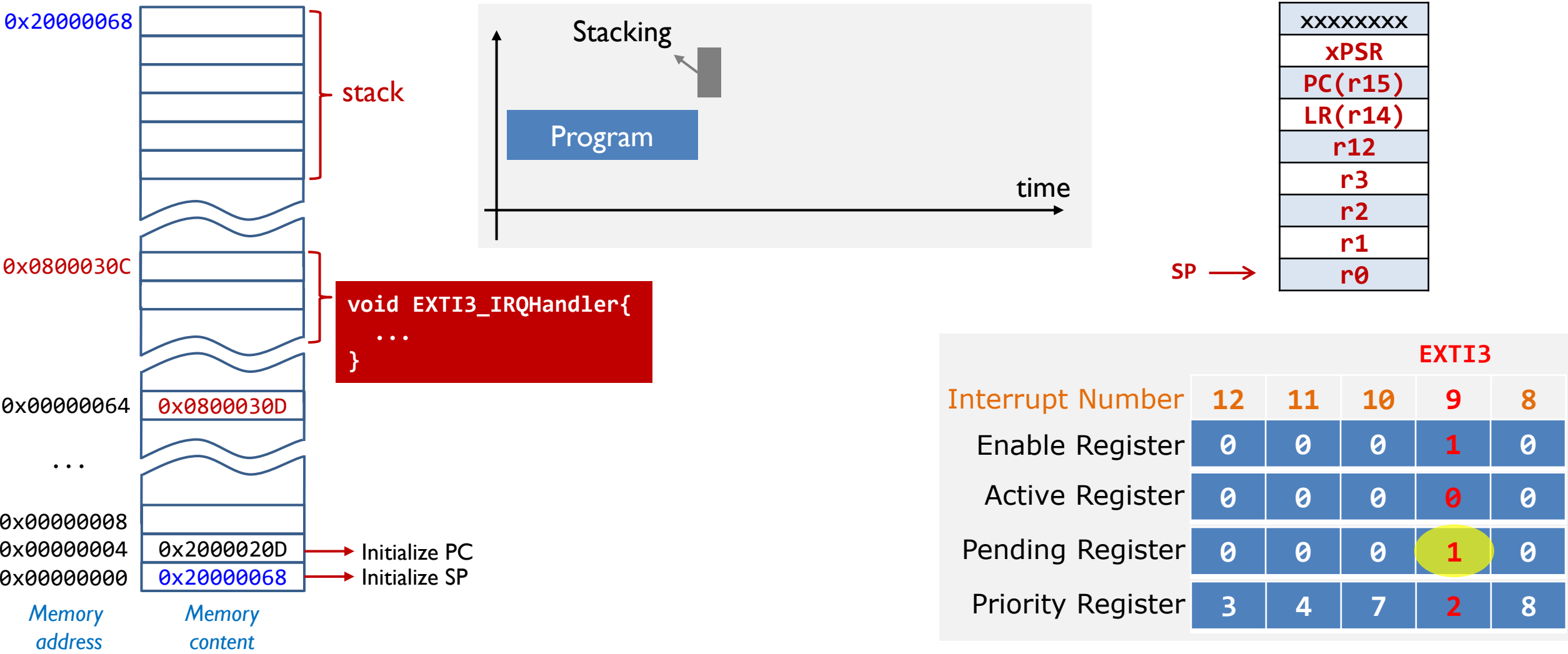
Single Interrupt



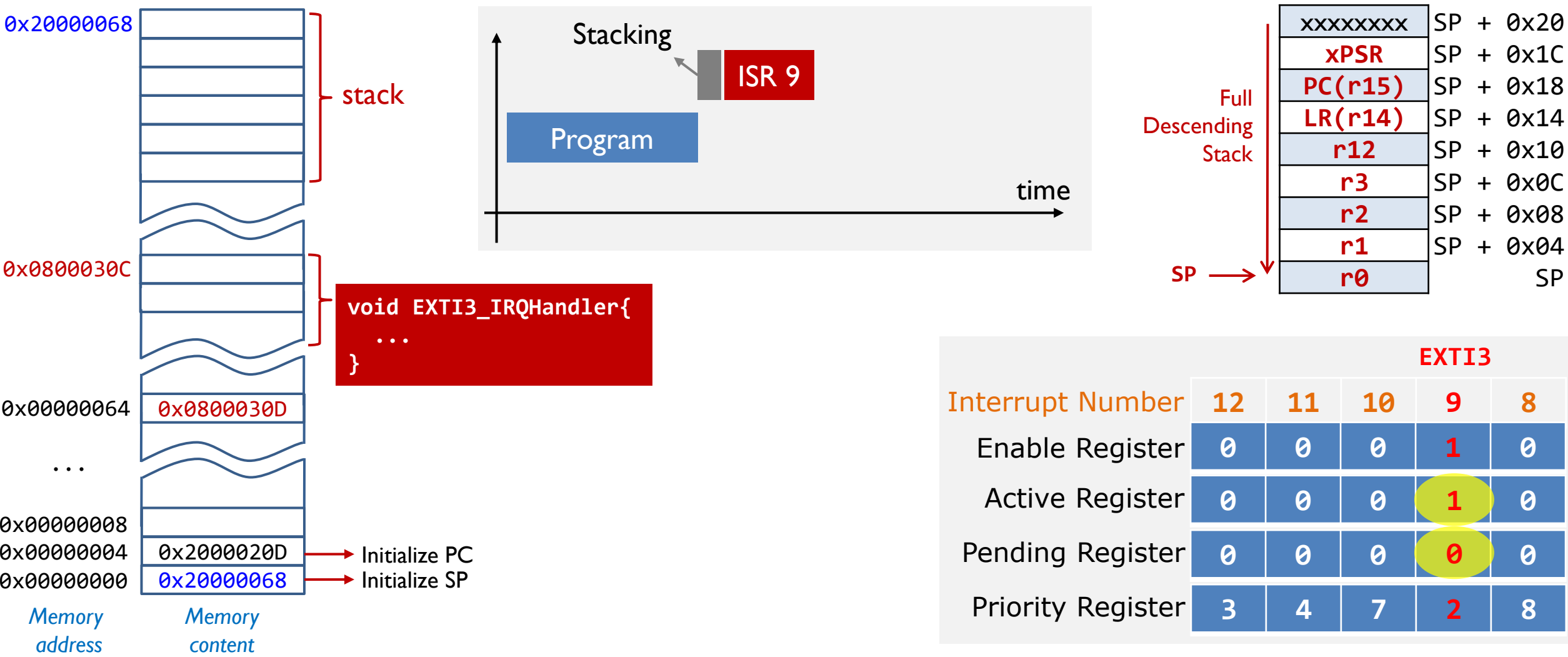
Single Interrupt



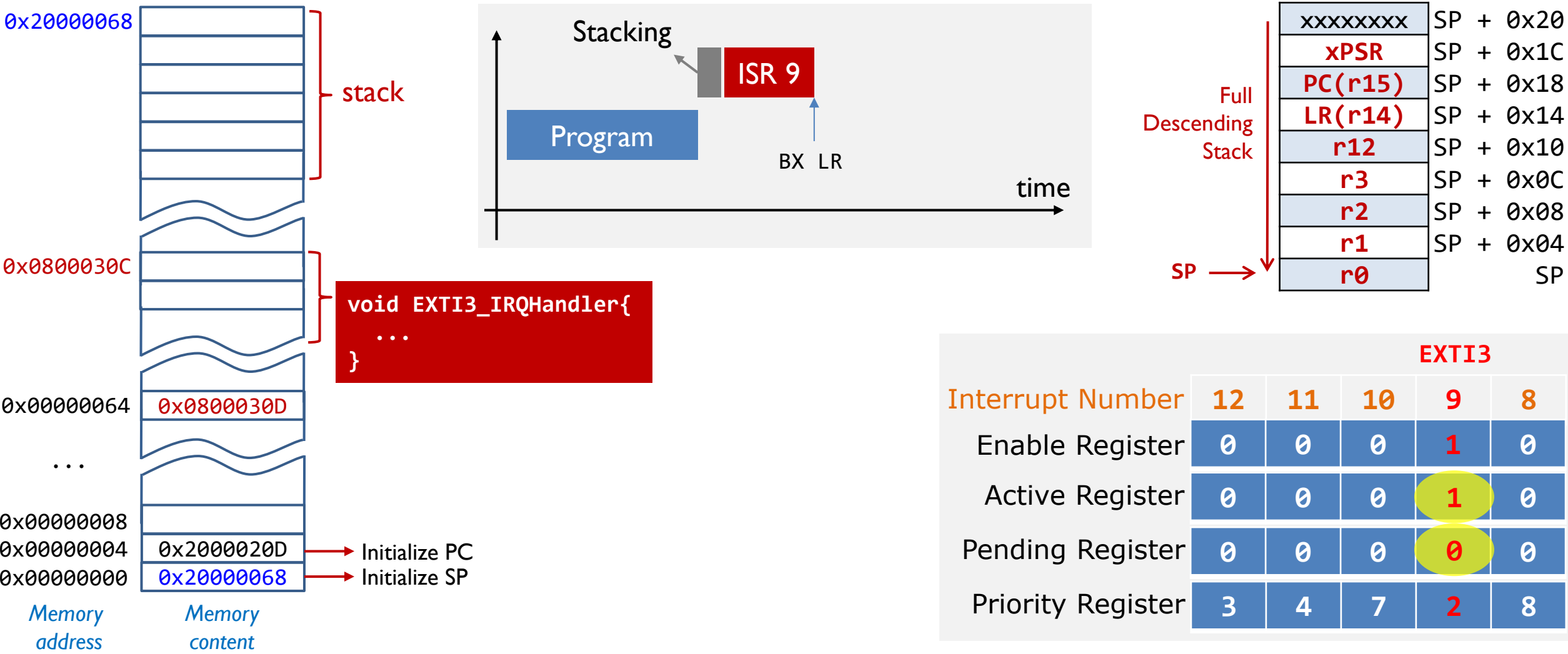
Single Interrupt



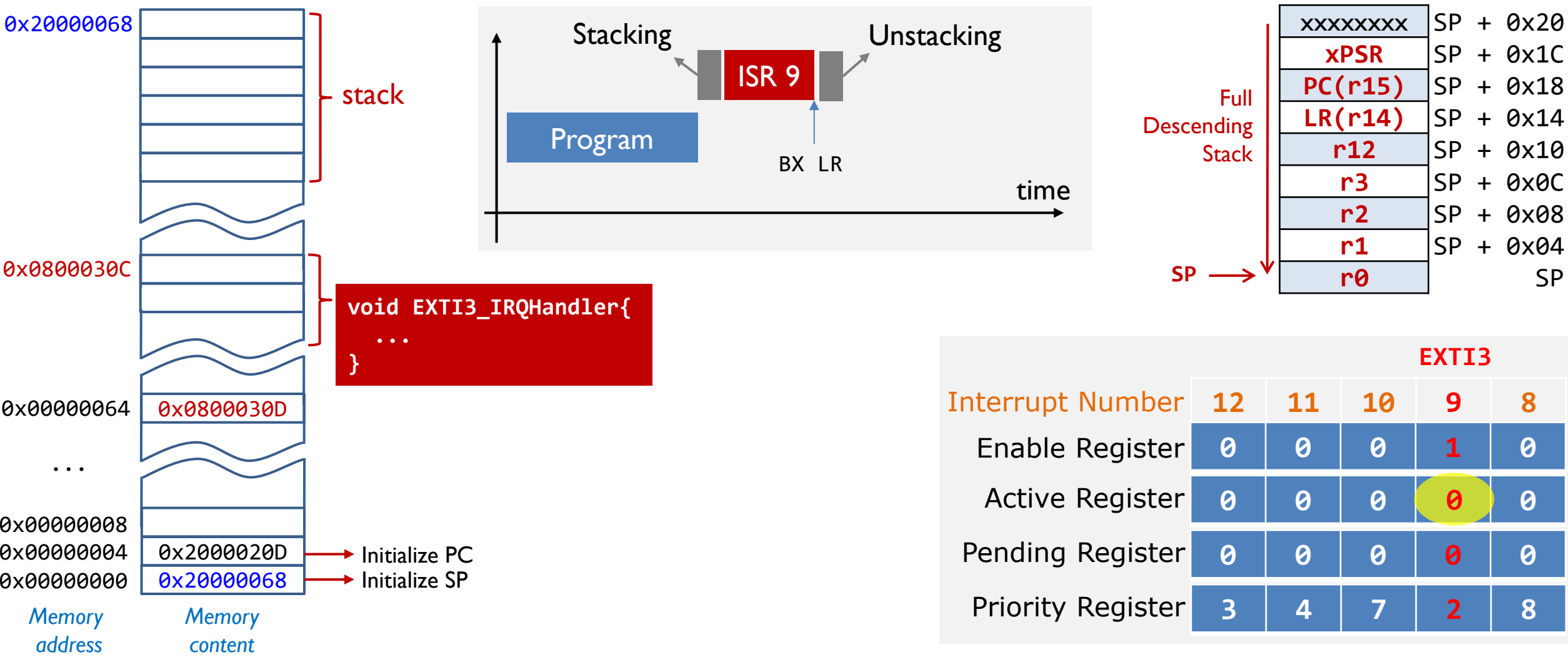
Single Interrupt



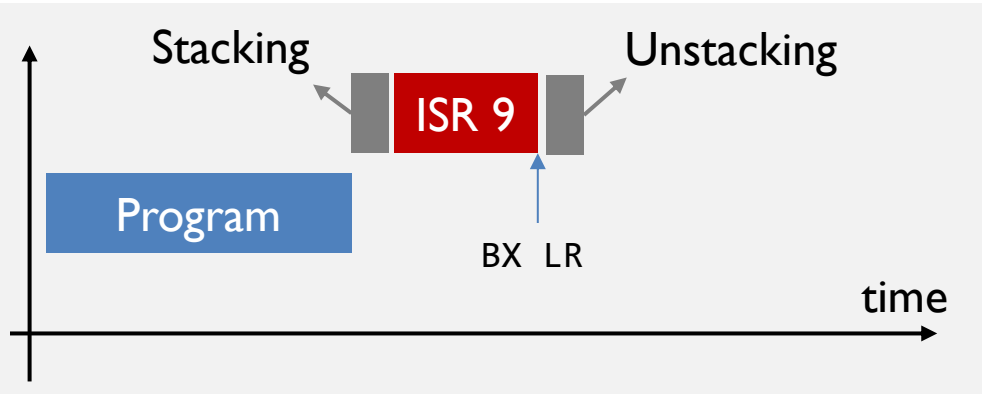
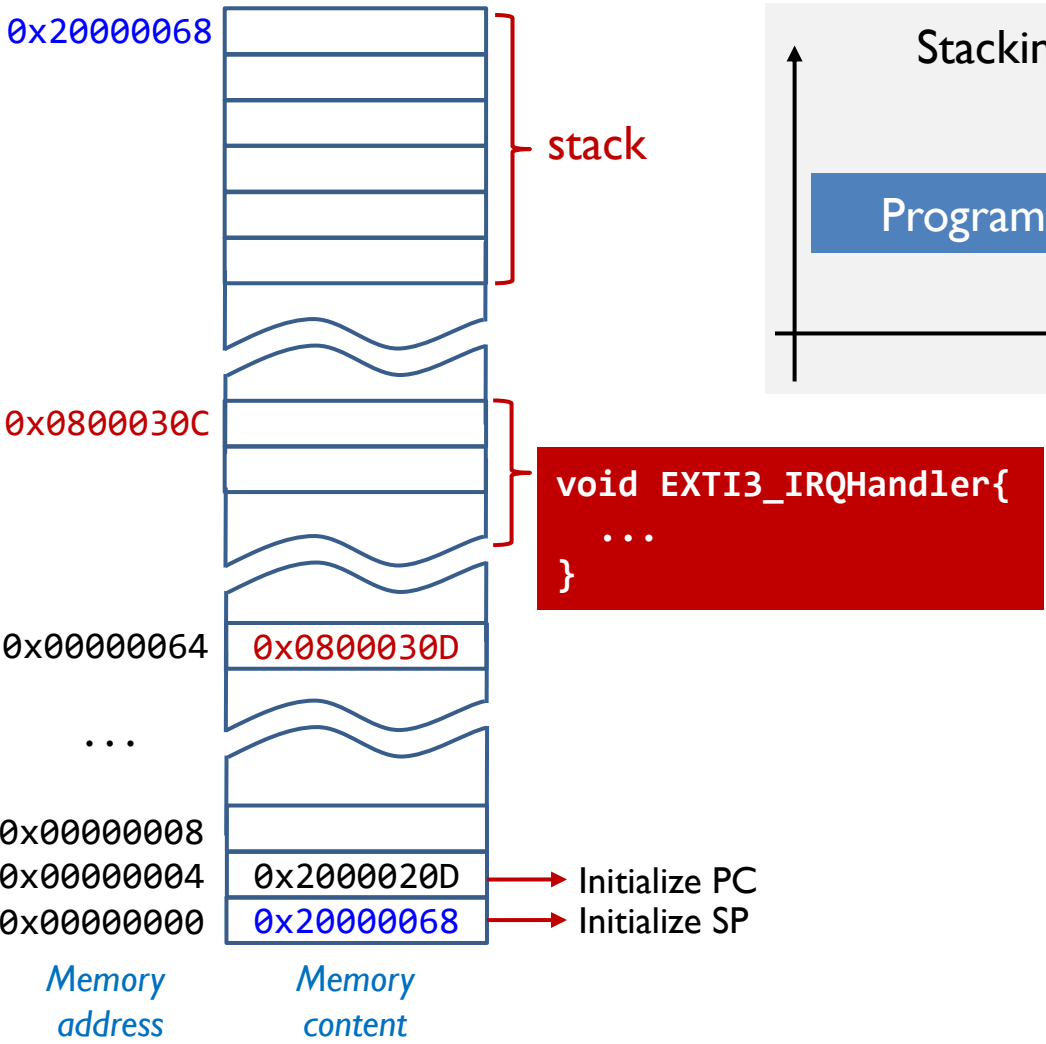
Single Interrupt



Single Interrupt



Single Interrupt

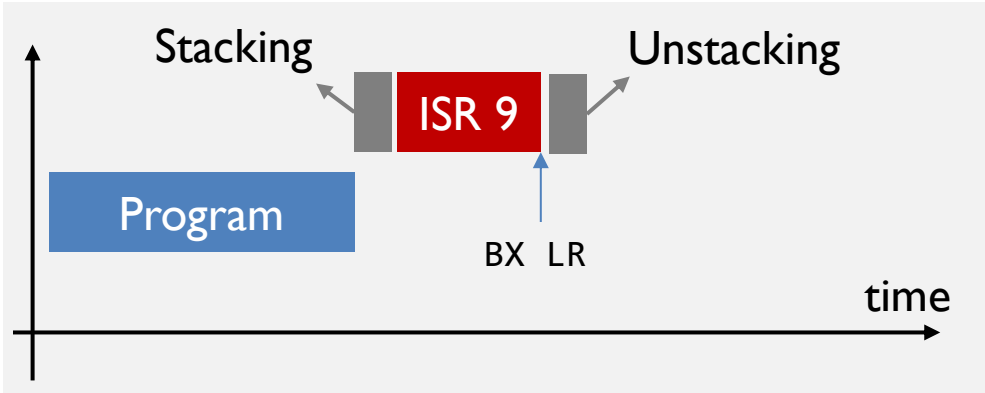
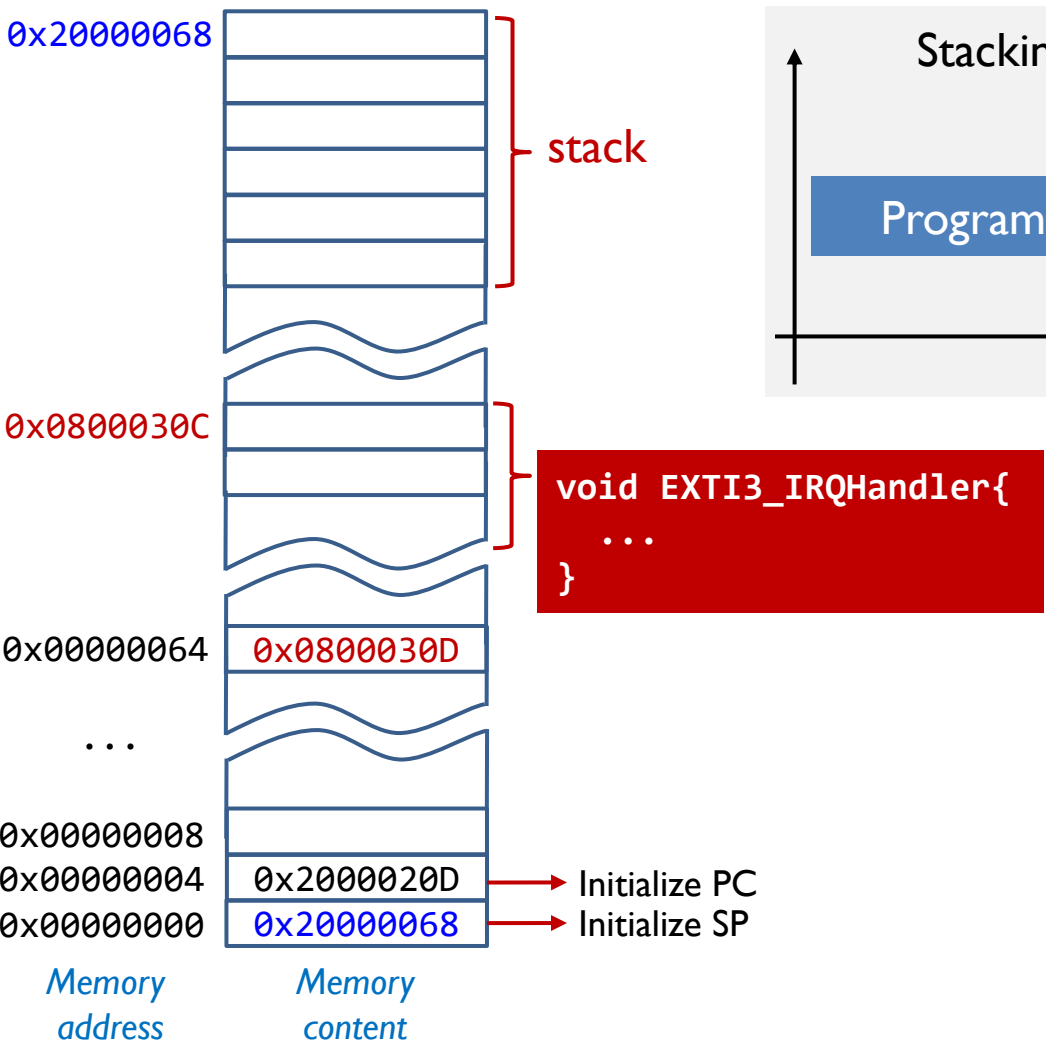


SP →

xxxxxxxx
xPSR
PC(r15)
LR(r14)
r12
r3
r2
r1
r0

	EXTI3				
Interrupt Number	12	11	10	9	8
Enable Register	0	0	0	1	0
Active Register	0	0	0	0	0
Pending Register	0	0	0	0	0
Priority Register	3	4	7	2	8

Single Interrupt

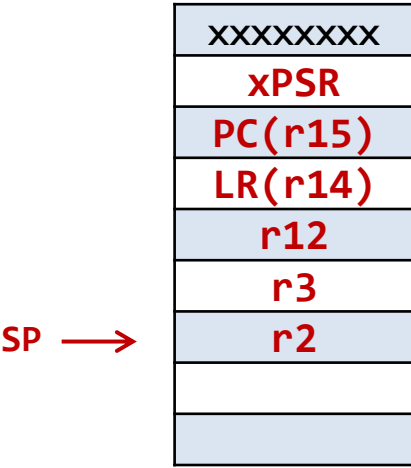
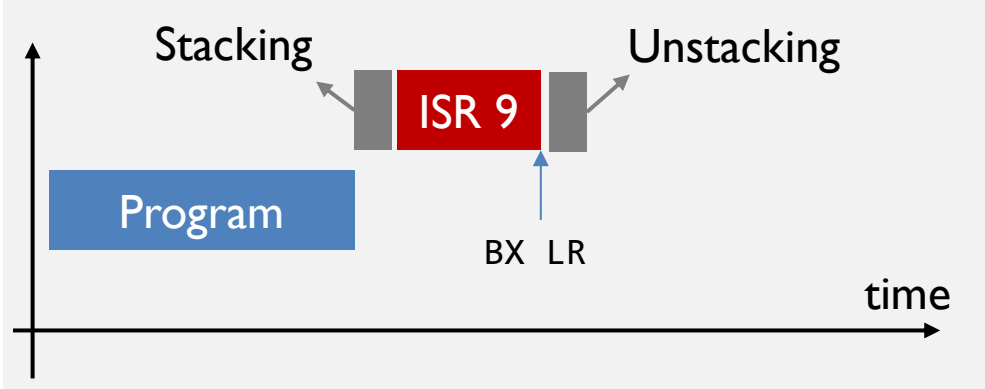
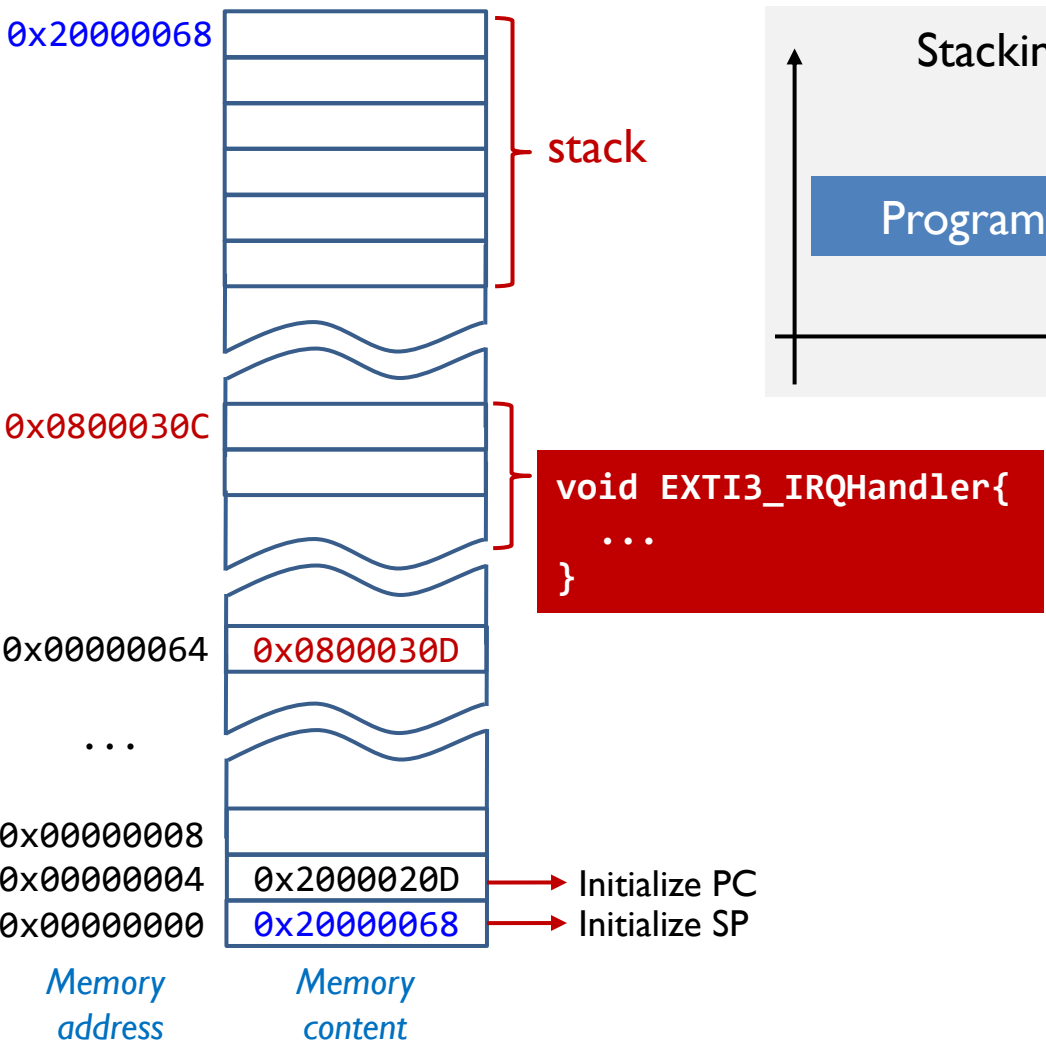


SP →

xxxxxxxx
xPSR
PC(r15)
LR(r14)
r12
r3
r2
r1

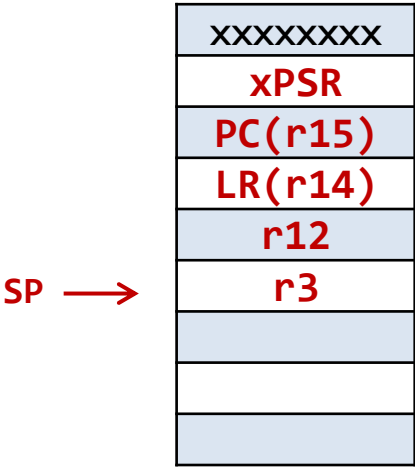
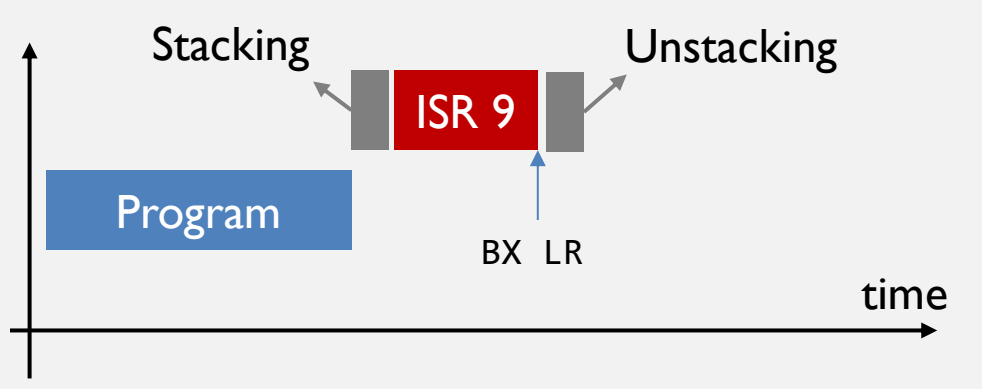
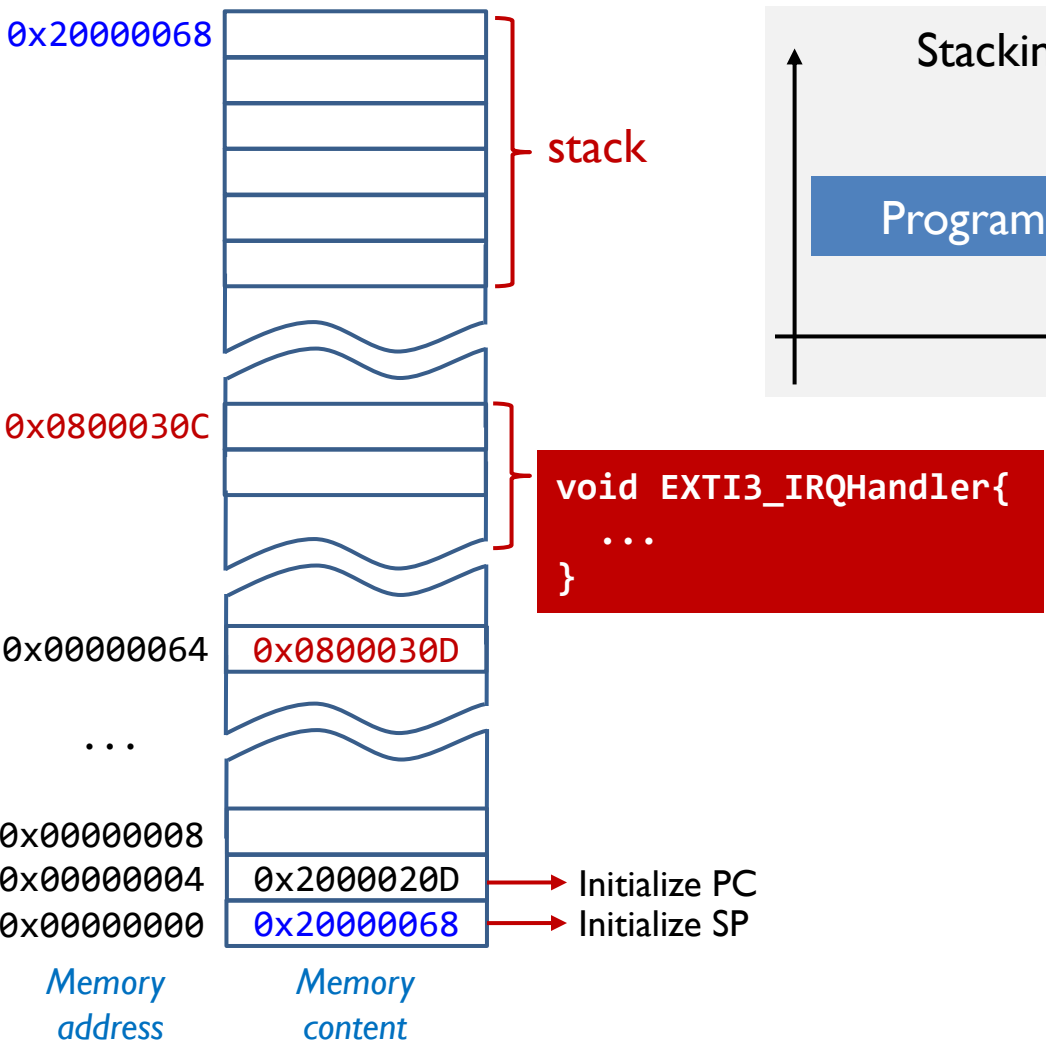
	EXTI3				
Interrupt Number	12	11	10	9	8
Enable Register	0	0	0	1	0
Active Register	0	0	0	0	0
Pending Register	0	0	0	0	0
Priority Register	3	4	7	2	8

Single Interrupt



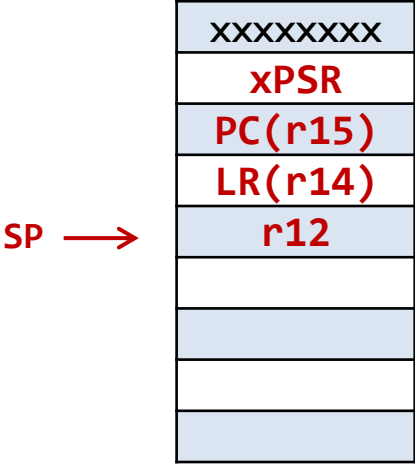
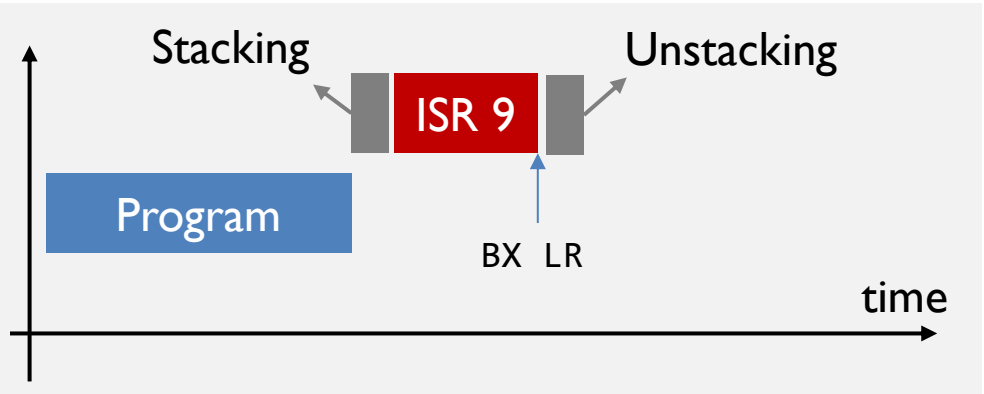
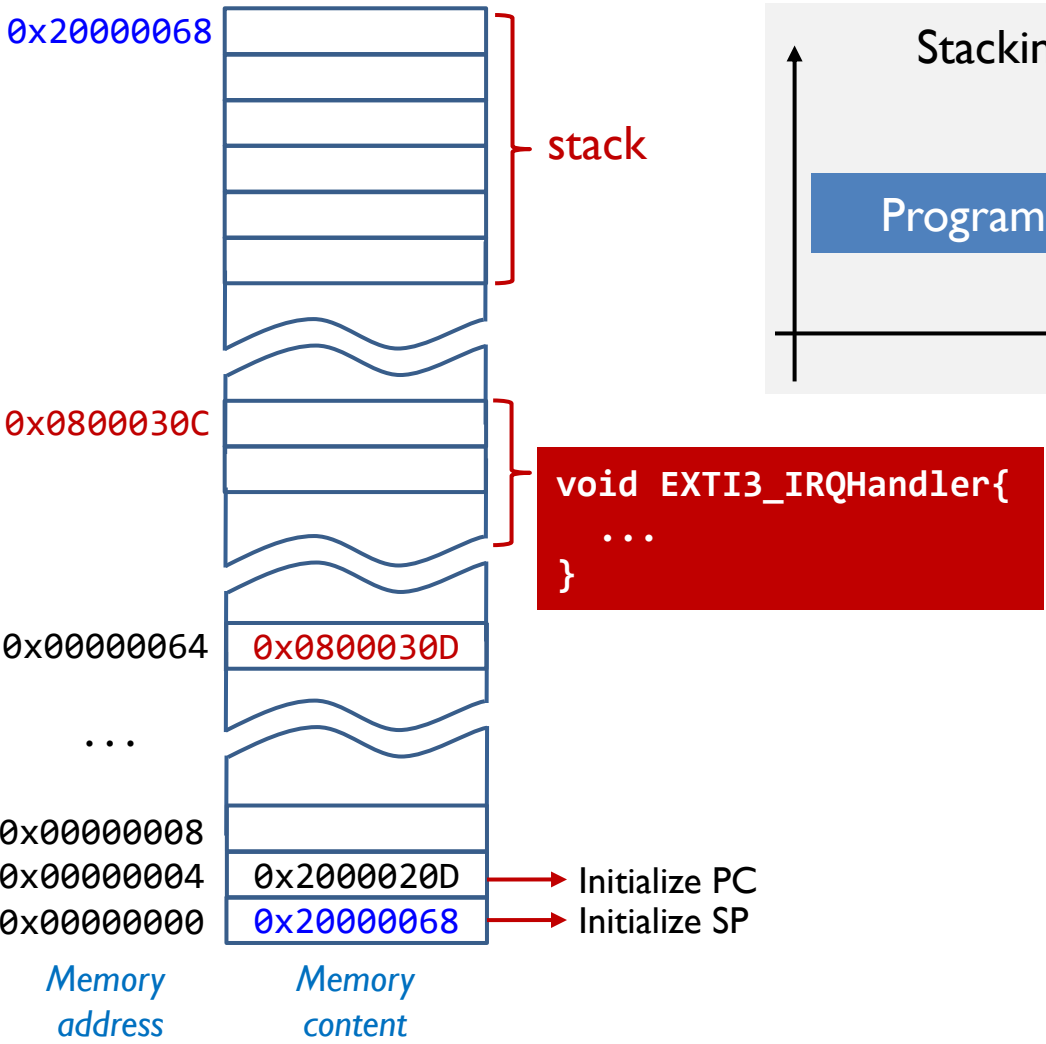
	EXTI3				
Interrupt Number	12	11	10	9	8
Enable Register	0	0	0	1	0
Active Register	0	0	0	0	0
Pending Register	0	0	0	0	0
Priority Register	3	4	7	2	8

Single Interrupt



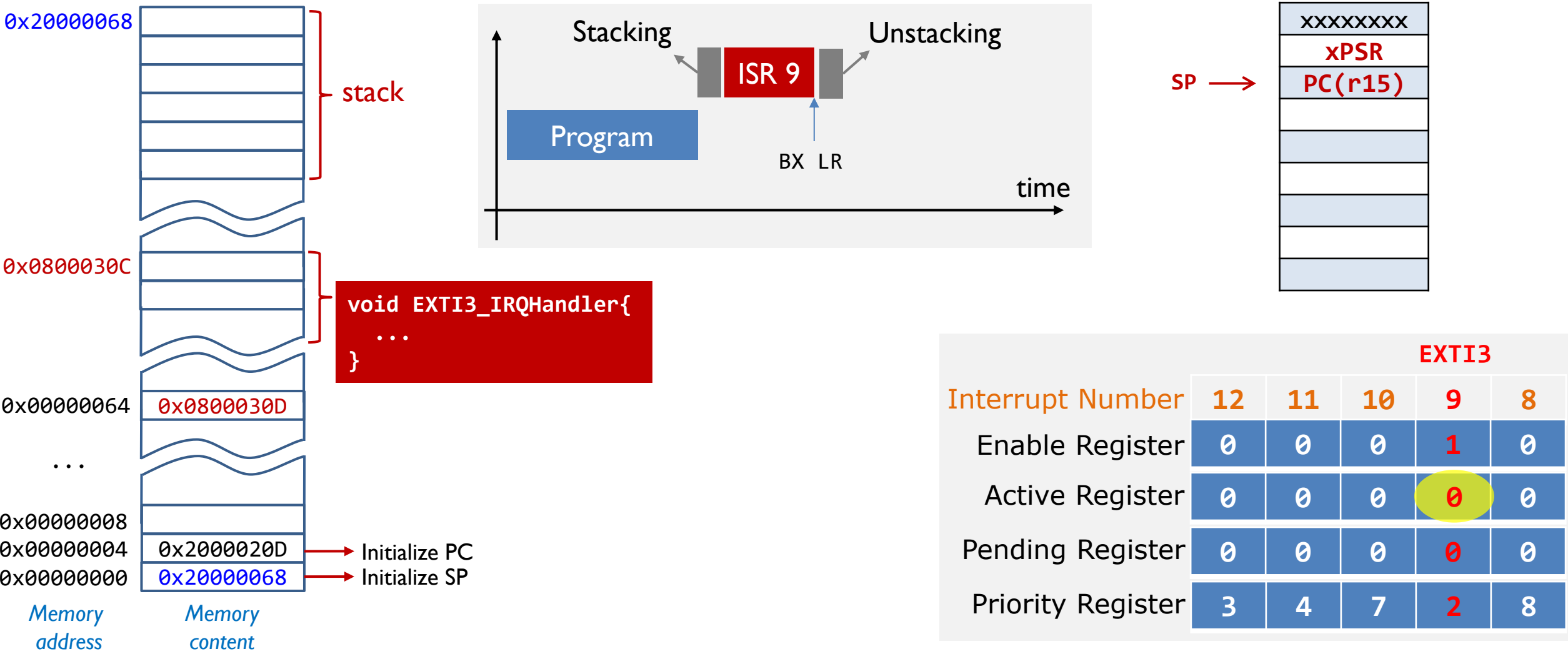
	EXTI3				
Interrupt Number	12	11	10	9	8
Enable Register	0	0	0	1	0
Active Register	0	0	0	0	0
Pending Register	0	0	0	0	0
Priority Register	3	4	7	2	8

Single Interrupt

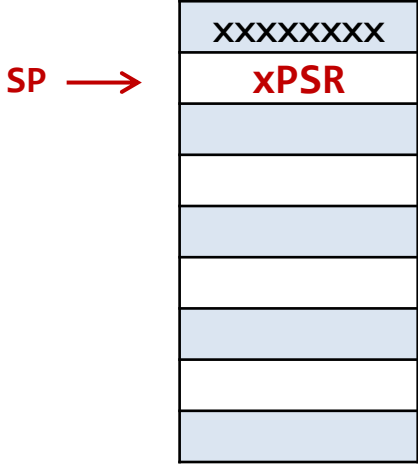
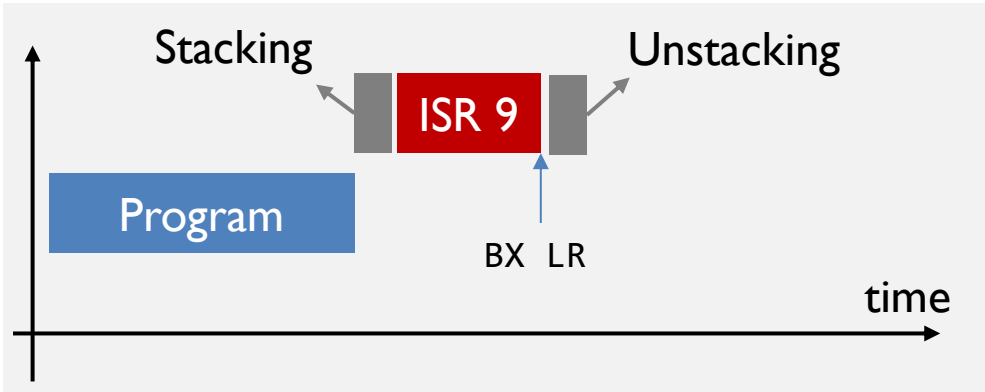
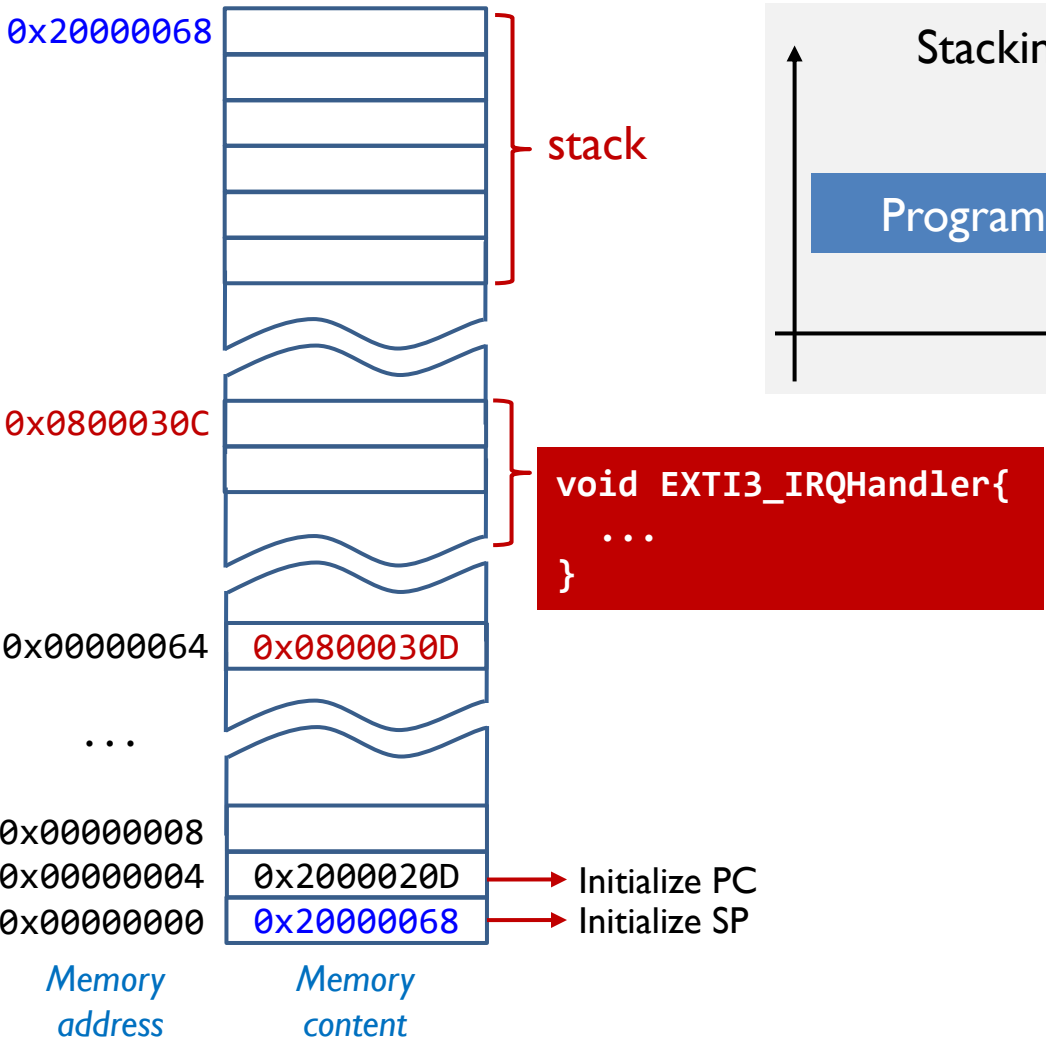


	EXTI3				
Interrupt Number	12	11	10	9	8
Enable Register	0	0	0	1	0
Active Register	0	0	0	0	0
Pending Register	0	0	0	0	0
Priority Register	3	4	7	2	8

Single Interrupt

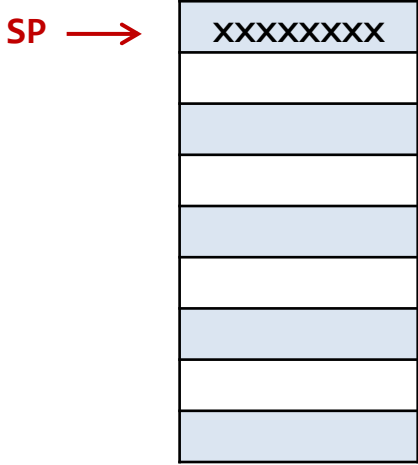
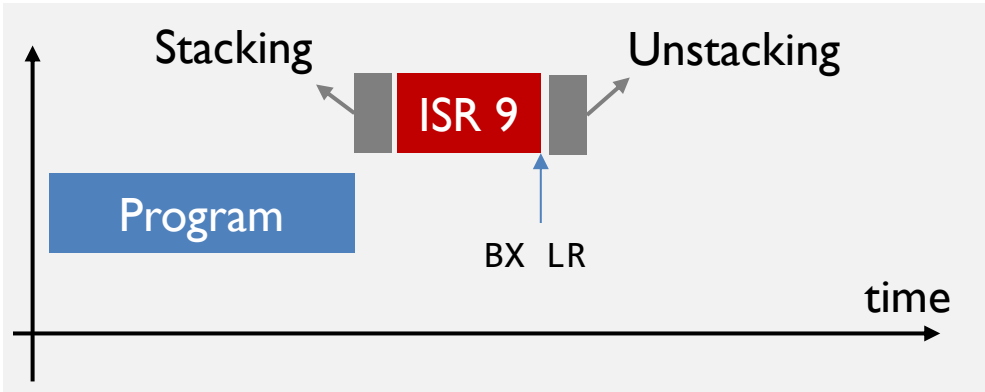
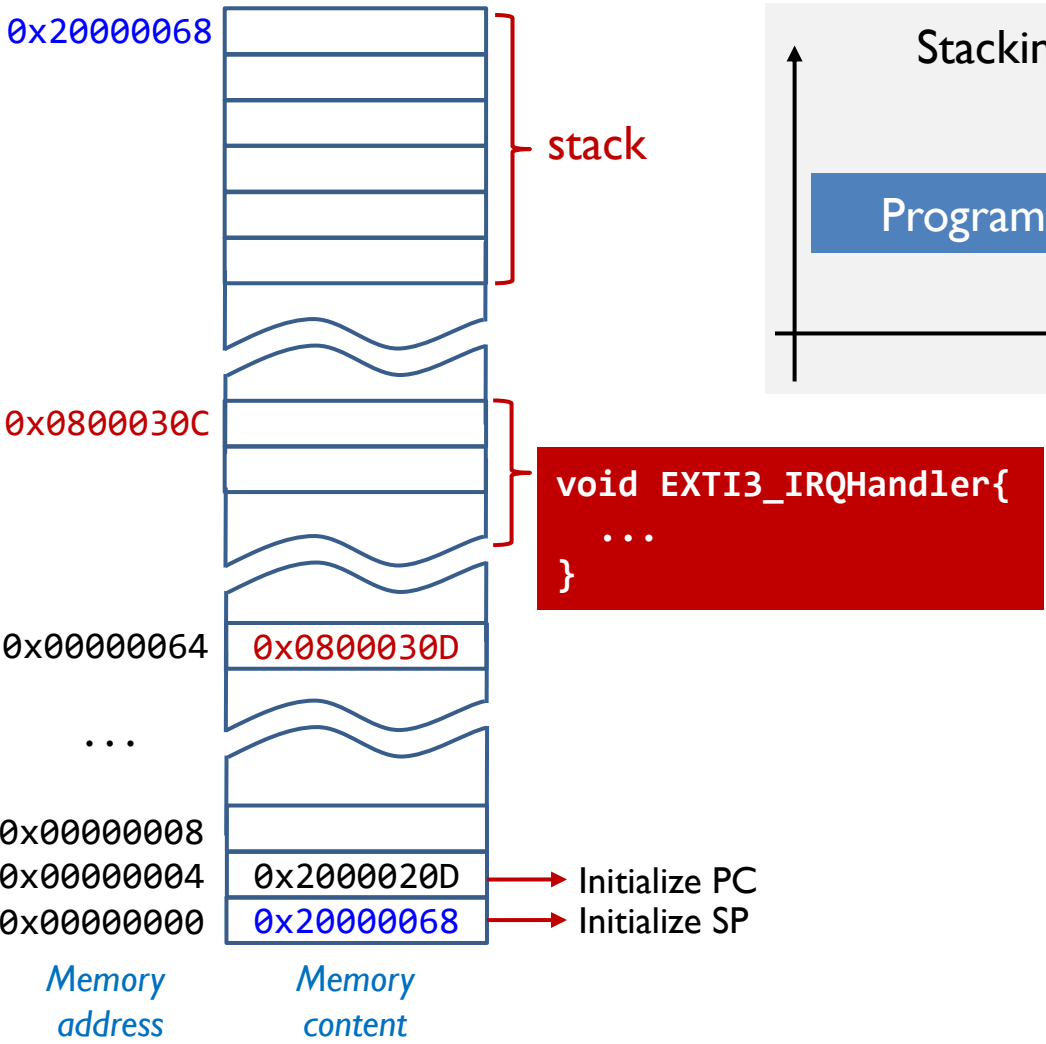


Single Interrupt



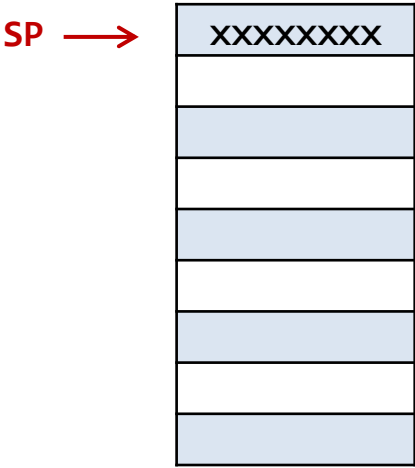
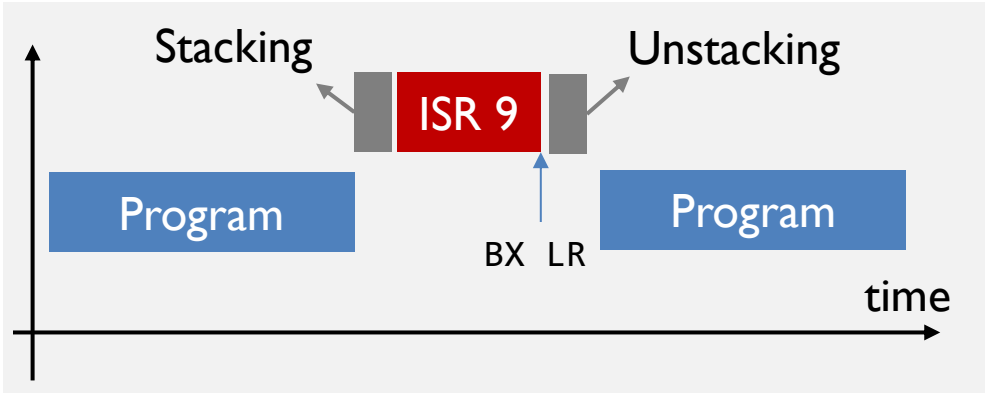
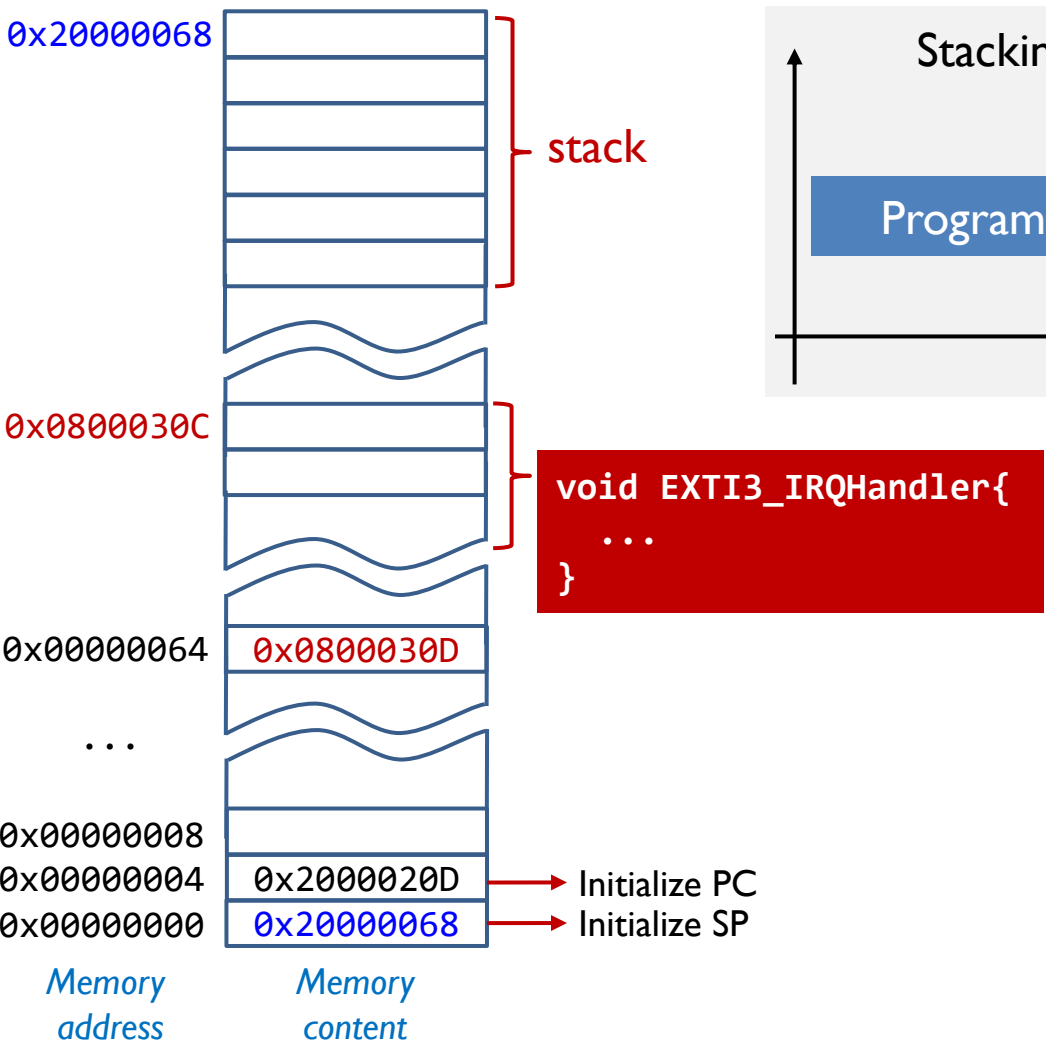
	EXTI3				
Interrupt Number	12	11	10	9	8
Enable Register	0	0	0	1	0
Active Register	0	0	0	0	0
Pending Register	0	0	0	0	0
Priority Register	3	4	7	2	8

Single Interrupt



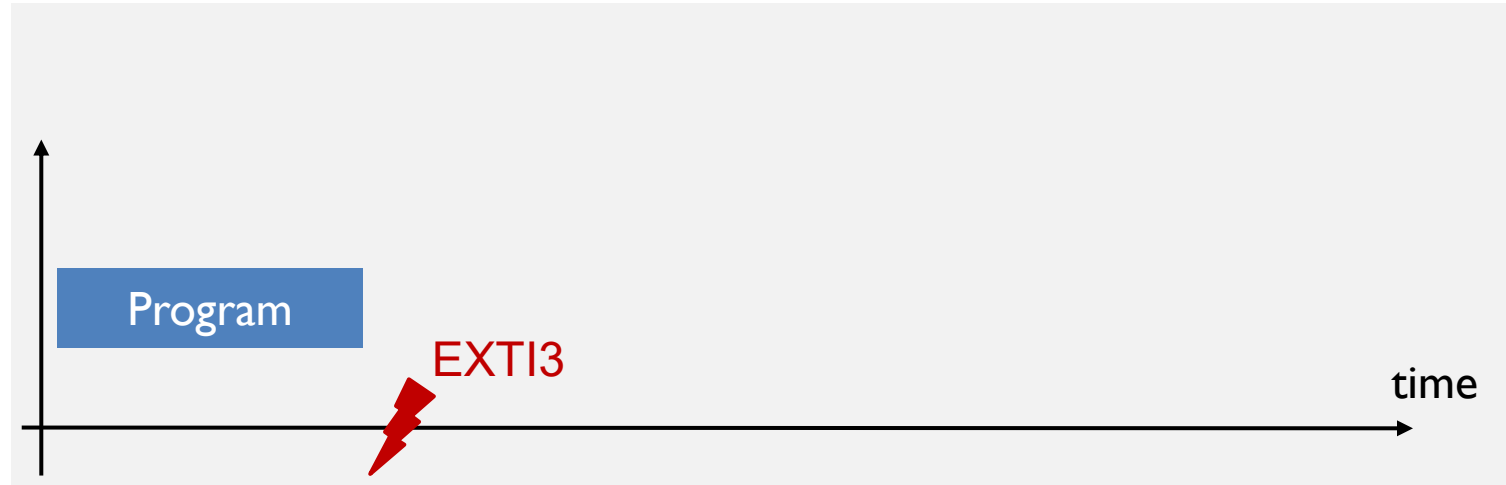
	EXTI3				
Interrupt Number	12	11	10	9	8
Enable Register	0	0	0	1	0
Active Register	0	0	0	0	0
Pending Register	0	0	0	0	0
Priority Register	3	4	7	2	8

Single Interrupt

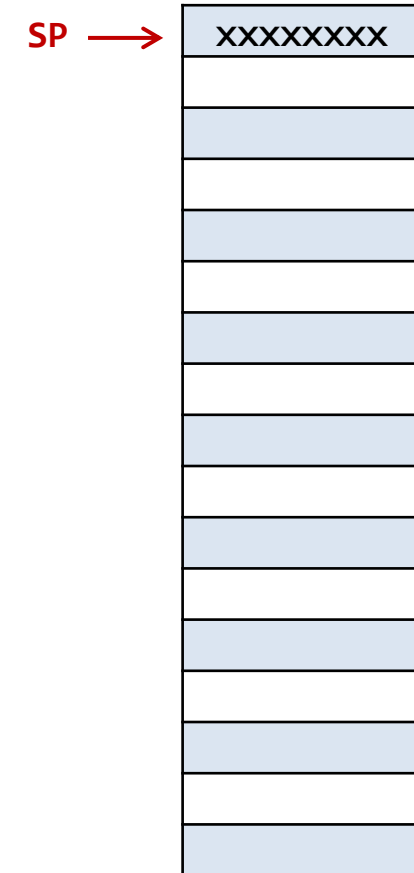


EXTI3					
Interrupt Number	12	11	10	9	8
Enable Register	0	0	0	1	0
Active Register	0	0	0	0	0
Pending Register	0	0	0	0	0
Priority Register	3	4	7	2	8

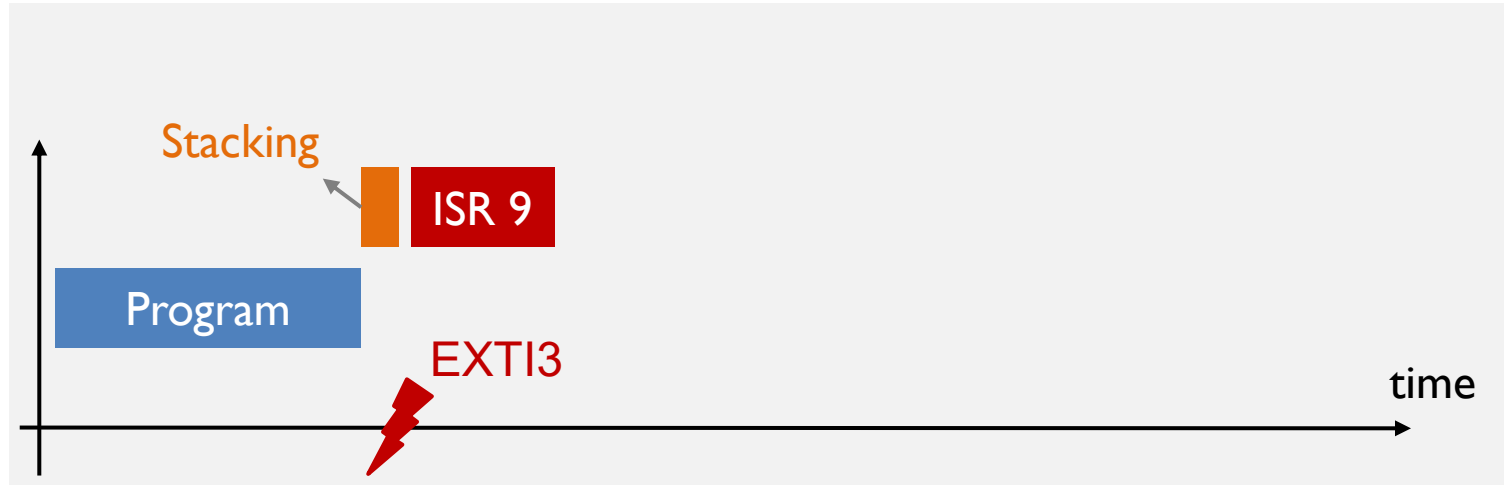
Nested Interrupts: Example of Preemption



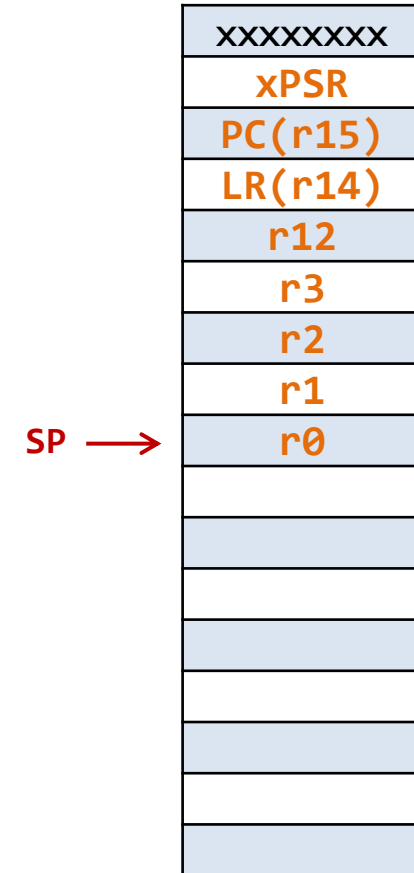
	DMA1_Channel2			EXTI3	
Interrupt Number	12	11	10	9	8
Enable Register	1	0	0	1	0
Active Register	0	0	0	0	0
Pending Register	0	0	0	0	0
Priority Register	3	4	7	5	3



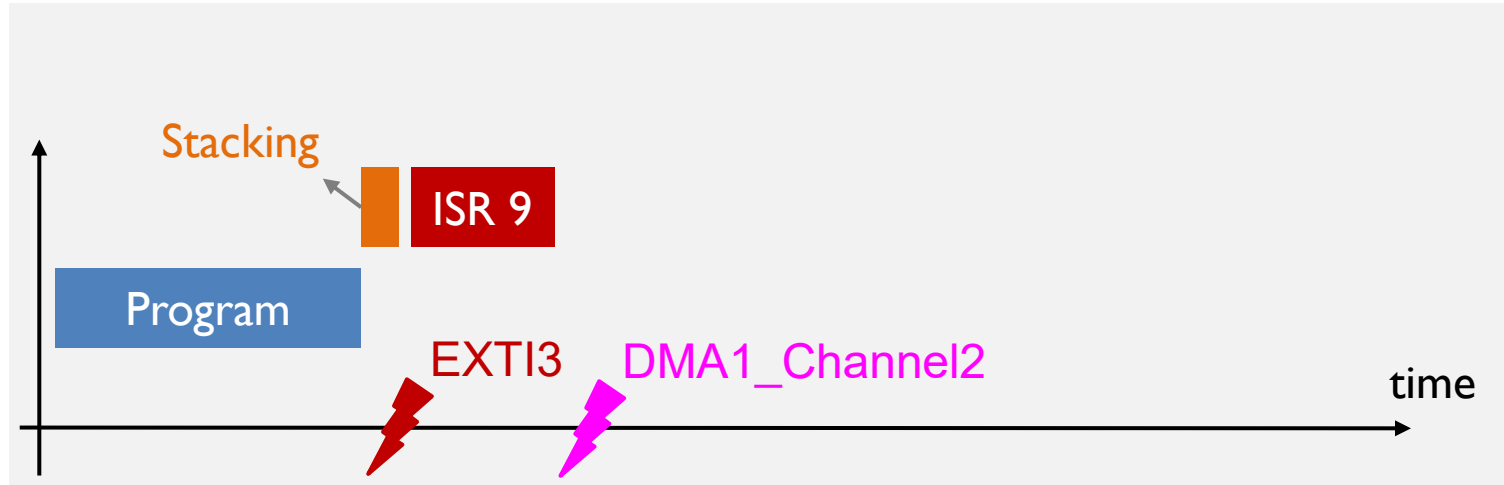
Nested Interrupts: Example of Preemption



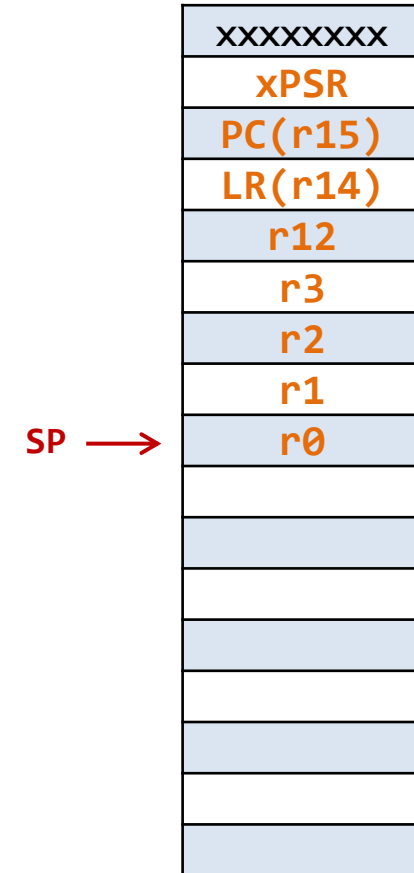
	DMA1_Channel2			EXTI3	
Interrupt Number	12	11	10	9	8
Enable Register	1	0	0	1	0
Active Register	0	0	0	1	0
Pending Register	0	0	0	0	0
Priority Register	3	4	7	5	3



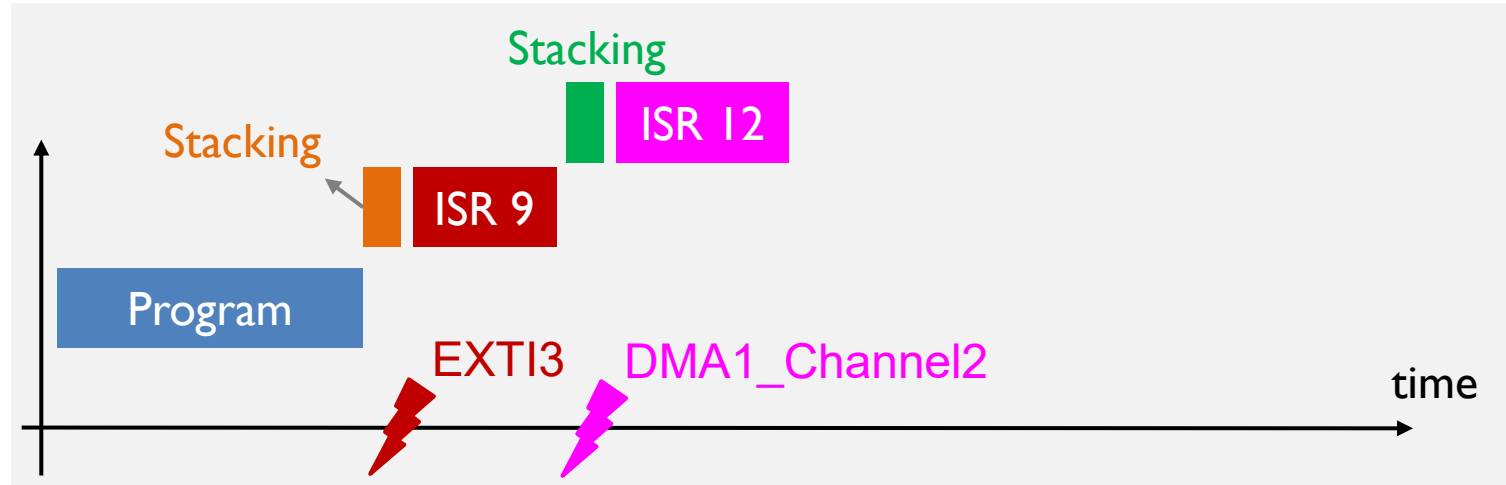
Nested Interrupts: Example of Preemption



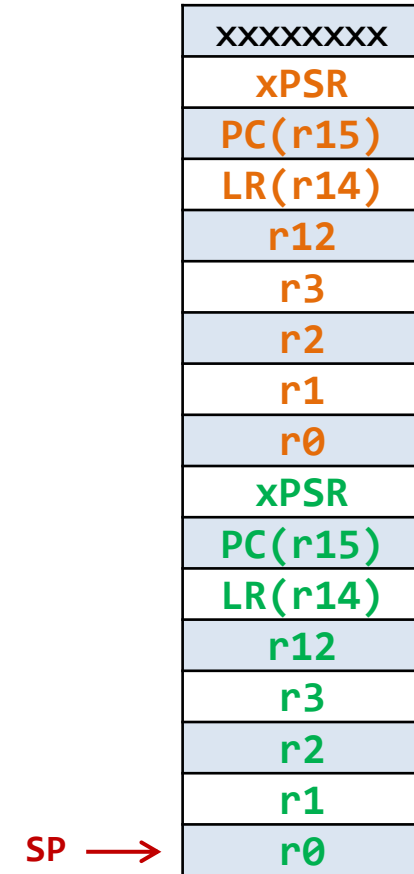
	DMA1_Channel2			EXTI3	
Interrupt Number	12	11	10	9	8
Enable Register	1	0	0	1	0
Active Register	0	0	0	1	0
Pending Register	1	0	0	0	0
Priority Register	3	4	7	5	3



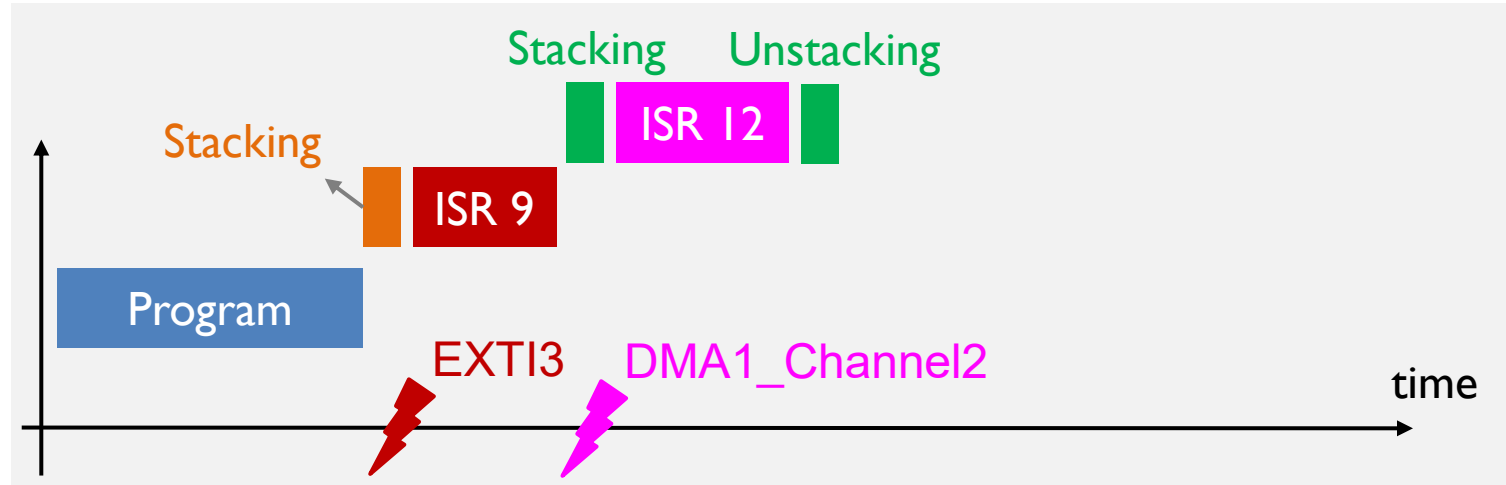
Nested Interrupts: Example of Preemption



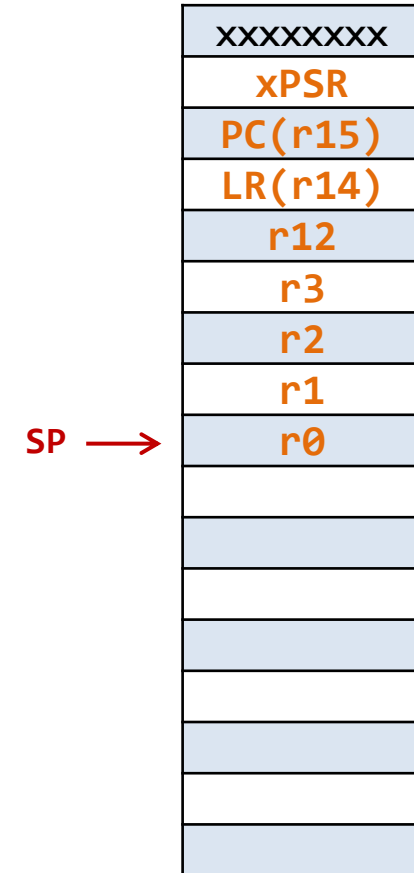
	DMA1_Channel2			EXTI3	
Interrupt Number	12	11	10	9	8
Enable Register	1	0	0	1	0
Active Register	1	0	0	1	0
Pending Register	0	0	0	0	0
Priority Register	3	4	7	5	3



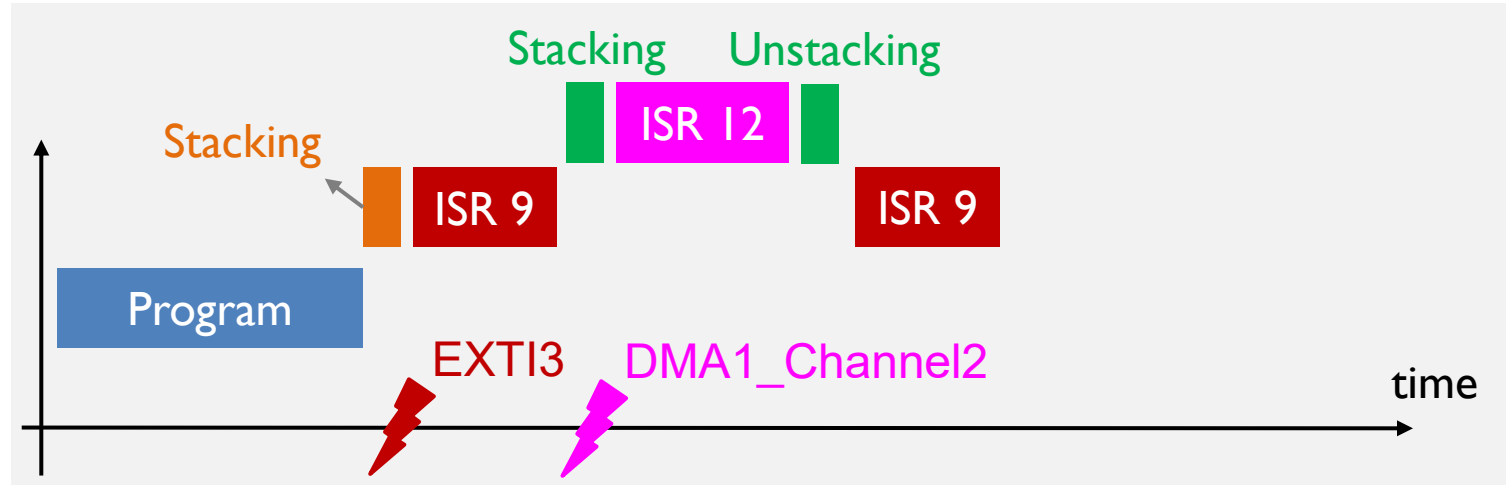
Nested Interrupts: Example of Preemption



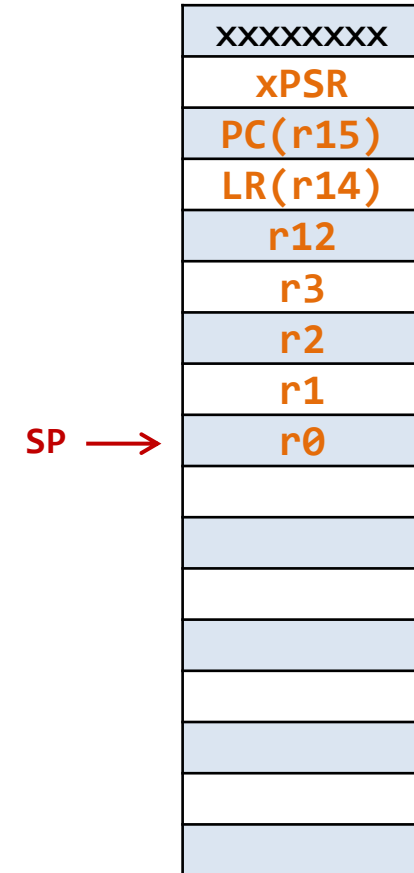
	DMA1_Channel2			EXTI3	
Interrupt Number	12	11	10	9	8
Enable Register	1	0	0	1	0
Active Register	0	0	0	1	0
Pending Register	0	0	0	0	0
Priority Register	3	4	7	5	3



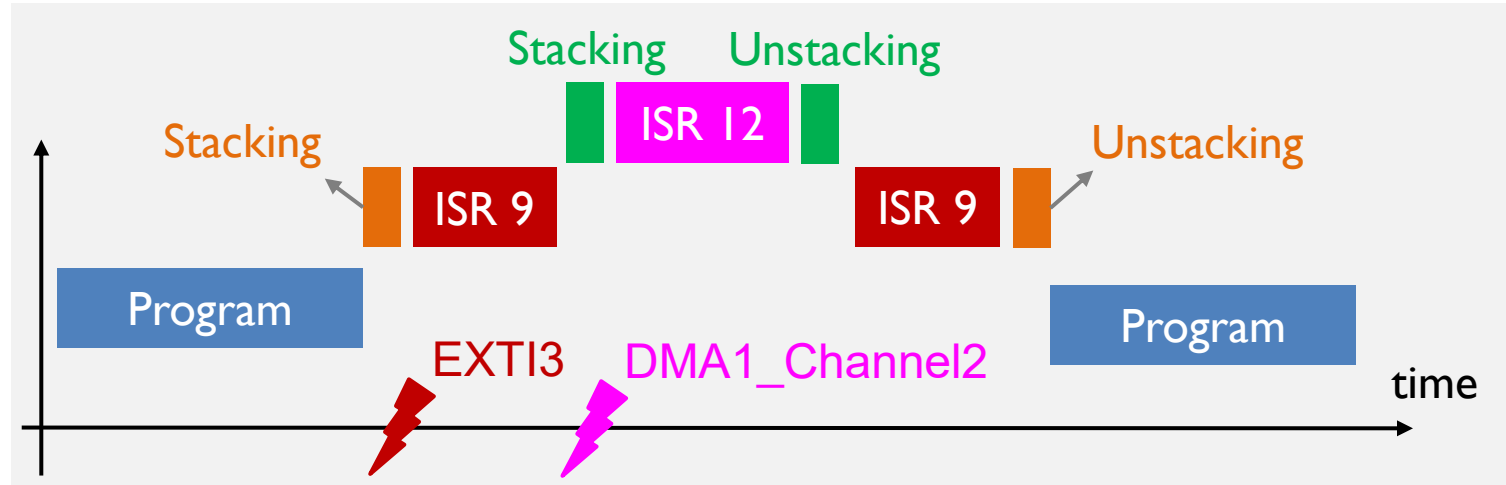
Nested Interrupts: Example of Preemption



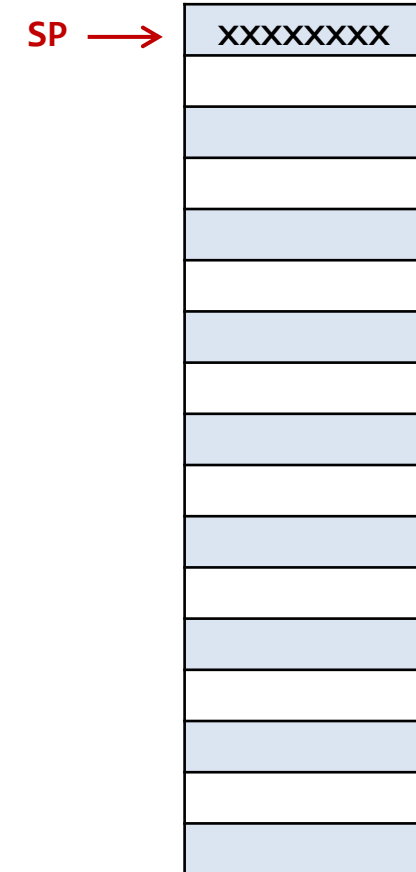
	DMA1_Channel2			EXTI3	
Interrupt Number	12	11	10	9	8
Enable Register	1	0	0	1	0
Active Register	0	0	0	1	0
Pending Register	0	0	0	0	0
Priority Register	3	4	7	5	3



Nested Interrupts: Example of Preemption

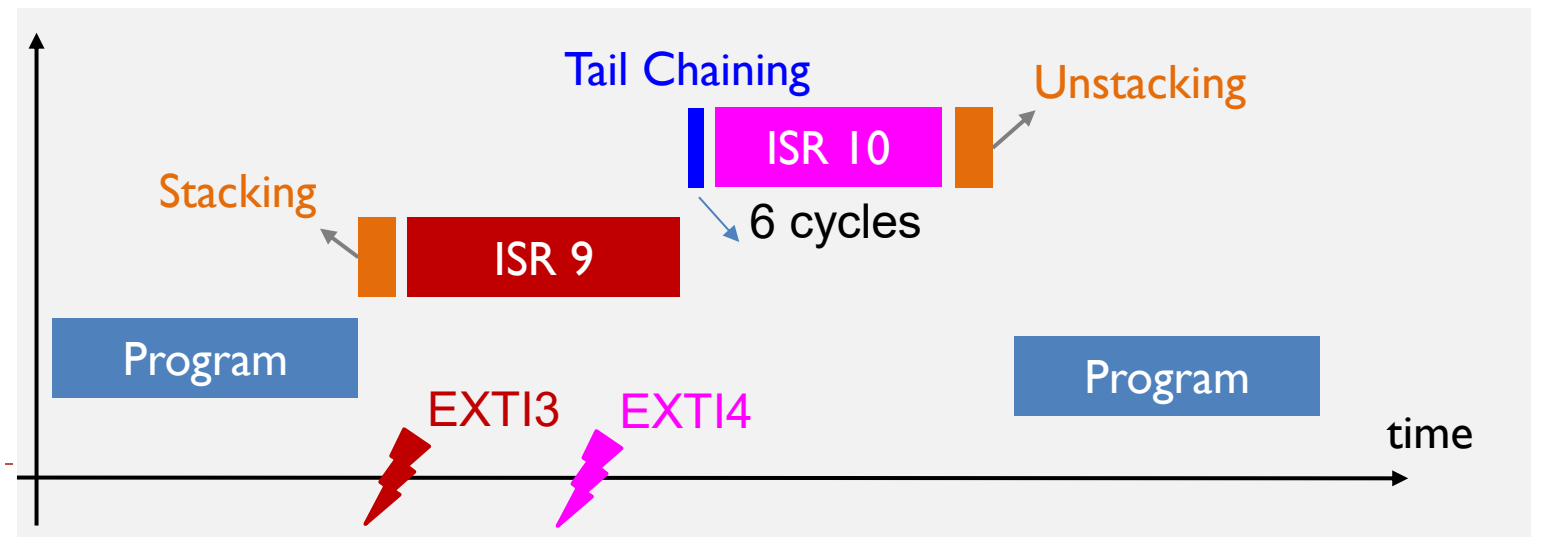
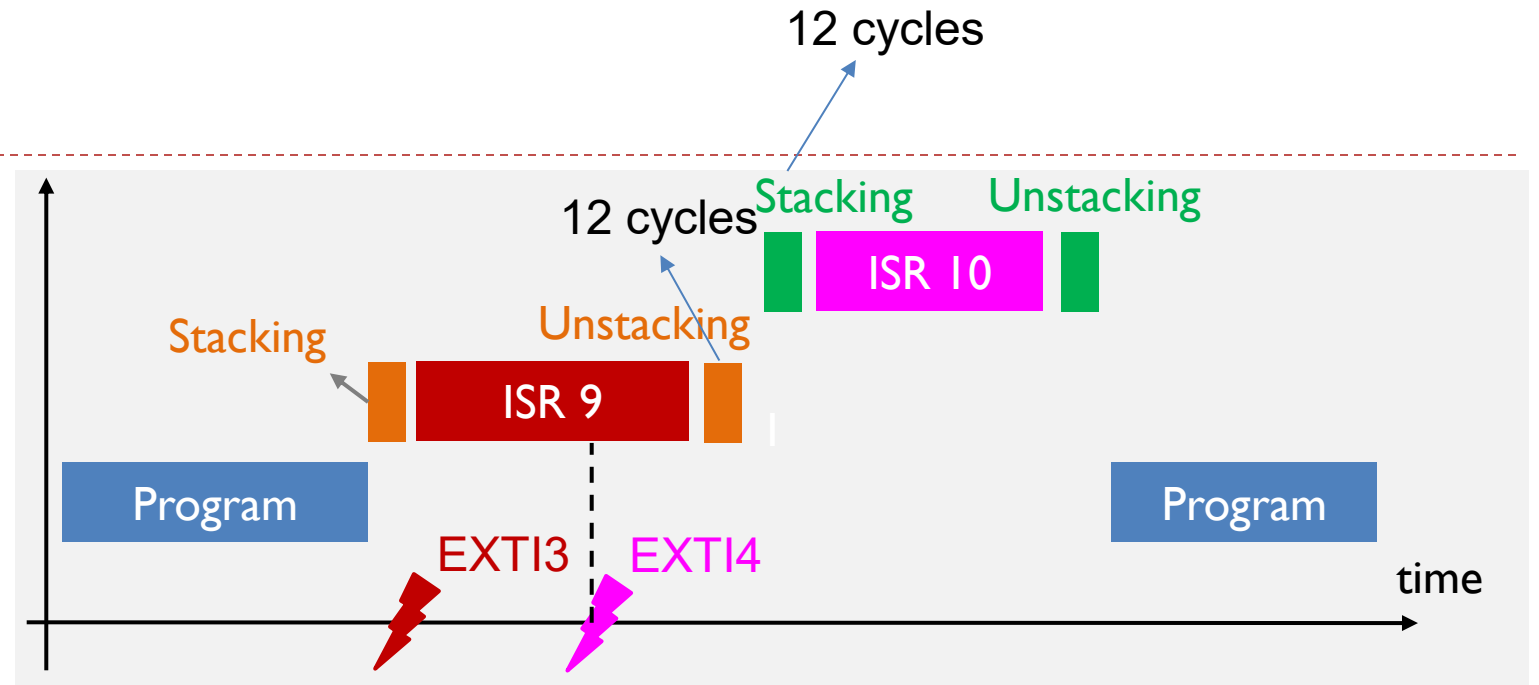


	DMA1_Channel2			EXTI3	
Interrupt Number	12	11	10	9	8
Enable Register	1	0	0	1	0
Active Register	0	0	0	0	0
Pending Register	0	0	0	0	0
Priority Register	3	4	7	5	3



Nested Interrupts: Tail Chaining

- ▶ EXTI3 → ISR 9
- ▶ EXTI4 → ISR 10
- ▶ Suppose EXTI4 has less urgency than EXTI3.
 - ▶ EXTI4 has a higher numeric priority value than EXTI3.



References

- ▶ Lecture 9: Interrupts

- ▶ <https://www.youtube.com/watch?v=uFBNf7F3I60&list=PLRJhV4hUhlymmp5CCelFPyxbknsdcXCc8&index=9>