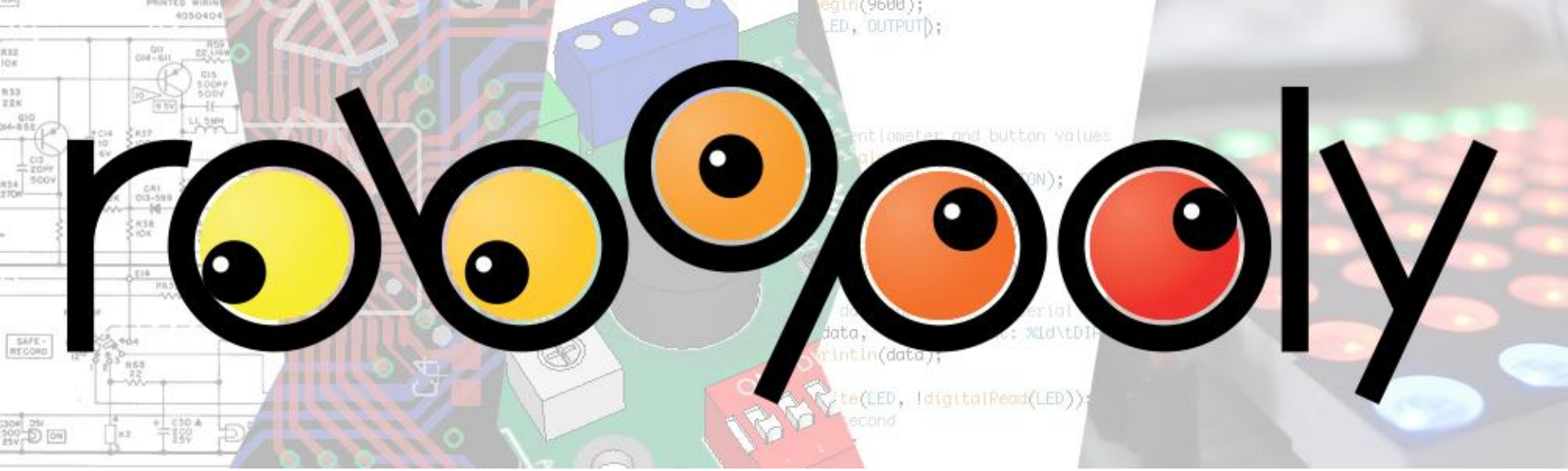


INTERRUPTIONS

LE MICROCONTROLEUR

Interruptions, pourquoi?

Comment faire
plusieurs
choses à la fois?

Interruptions, types

- Timer
  - “Je veux faire quelque chose dans 100*x ms pour N fois”
  - setTimer(fonction, x, N)
- External
  - “Je veux faire quelque chose quand une valeur mesurée change”
  - attachInterrupt(interrupt, fonction, quand)

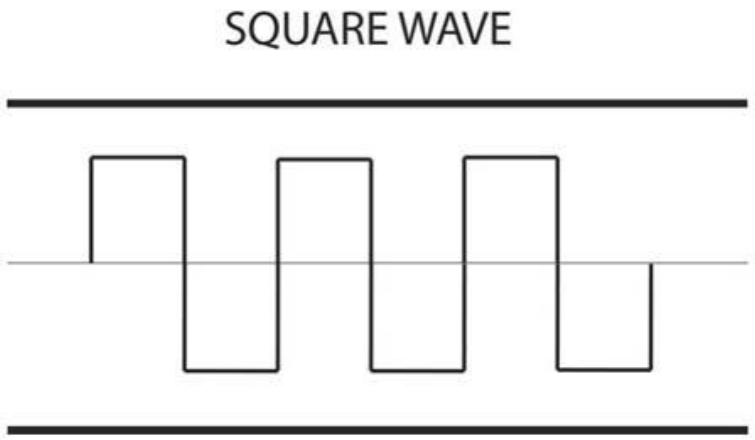
Interruptions, the Arduino way:

attachInterrupt (Prismino ~= Leonardo)

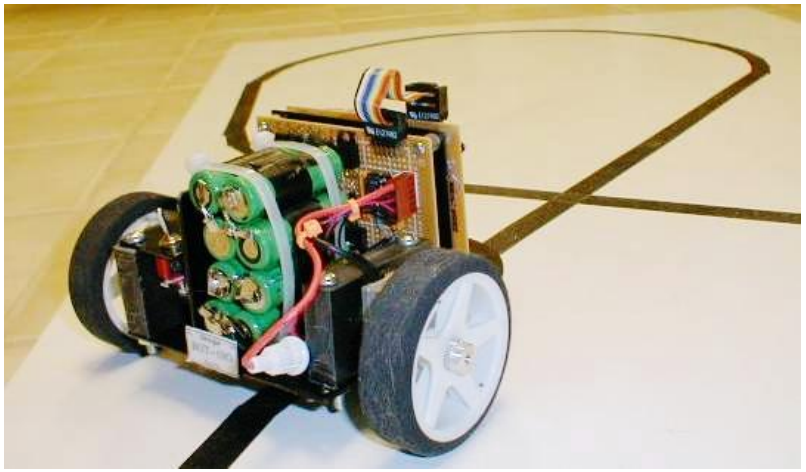
```
#define LED_PIN 13;
volatile int state = LOW;

void setup()
{
  pinMode(LED_PIN, OUTPUT);
  attachInterrupt(0, blink, CHANGE); // LOW, RISING, FALLING, CHANGE
}

void loop()
{
  digitalWrite(LED_PIN, state);
}

void blink()
{
  state = !state;
}
```

Interruptions, demo

- Jouer de la musique

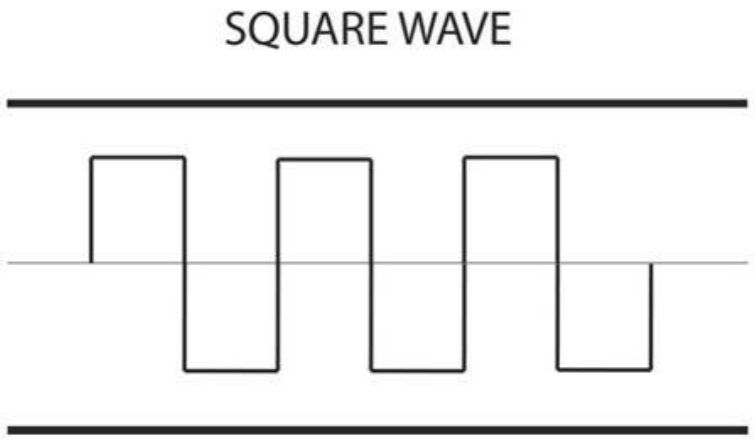


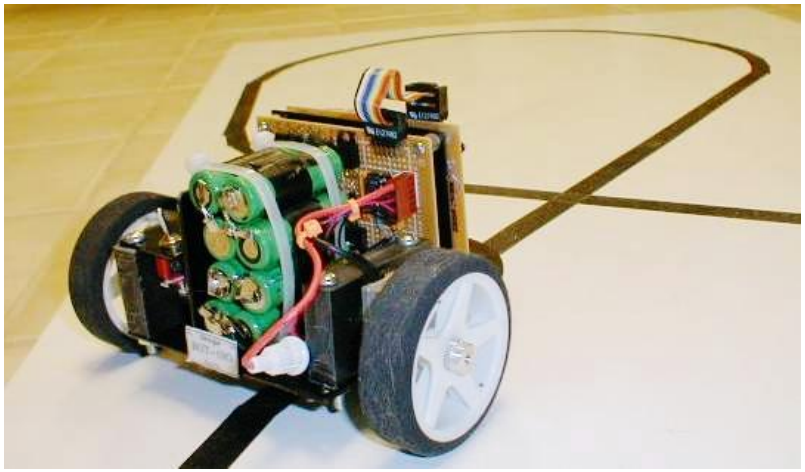
- Suivre une ligne



Interruptions, demo

- Jouer de la musique
  - Interruption à chaque change-ment de la valeur de sortie
- Suivre une ligne
  - Interruption chaque fois que le senseur ne voit plus la ligne





Interruptions, demo

```
void setup()
{
  //Update motor speeds when line is lost/found
  attachInterrupt(IR_PIN, follow_line, CHANGE);
}

void loop()
{
  play_music();
}
```

Interruptions, demo

```
void setup()
{
  //Call the play_note function every 2*100ms
  setTimer(play_note, 2, 0);
}

void loop()
{
  follow_line();
  delay(10);
}

void play_note()
{
      // play_note code
}
```

TIMER: 10ms

Interruptions, demo

```
void setup()
{
  //Call the play_note function every 2*100ms
  setTimer(play_note, 2, 0);
}

void loop()
{
  follow_line();
  delay(10);
}

void play_note()
{
      // play_note code
}
```

TIMER: 20ms

Interruptions, demo

```
void setup()
{
  //Call the play_note function every 2*100ms
  setTimer(play_note, 2, 0);
}

void loop()
{
  follow_line();
  delay(10);
}

void play_note()
{
      // play_note code
}
```

TIMER: 30ms

Interruptions, demo

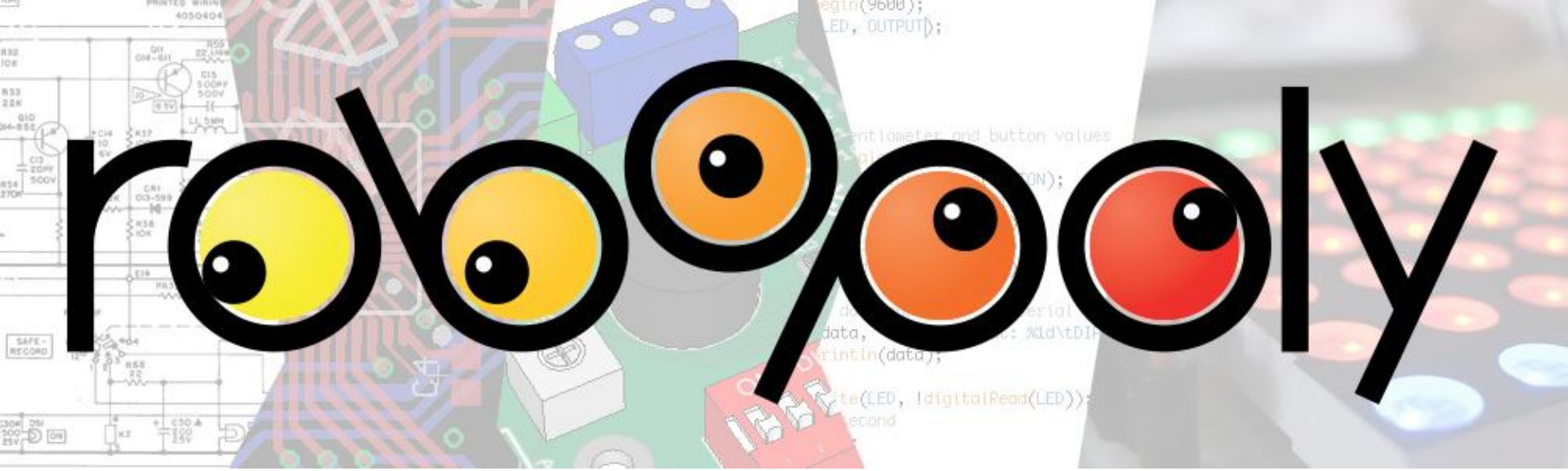
```
void setup()
{
  //Call the play_note function every 2*100ms
  setTimer(play_note, 2, 0);
}

void loop()
{
  follow_line();
  delay(10);
}

void play_note()
{
      // play_note code
}
```

TIMER: 200ms

Interruptions, demo

```
void setup()
{
  //Call the play_note function every 2*100ms
  setTimer(play_note, 2, 0);
}

void loop()
{
  follow_line();
  delay(10);
}

void play_note()
{
      // play_note code
}
```

TIMER: 210ms

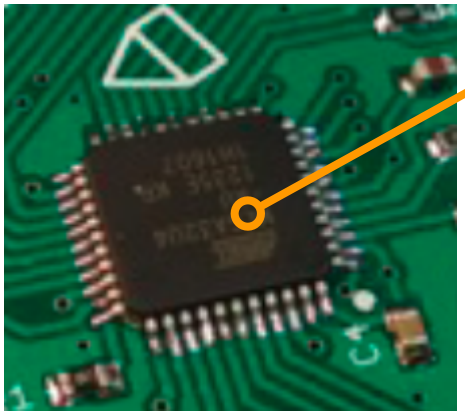


INTERRUPTIONS

LE MICROCONTROLEUR

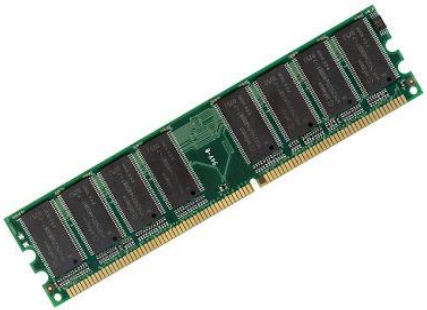
Il y a quoi dedans?

Ou, les datasheets

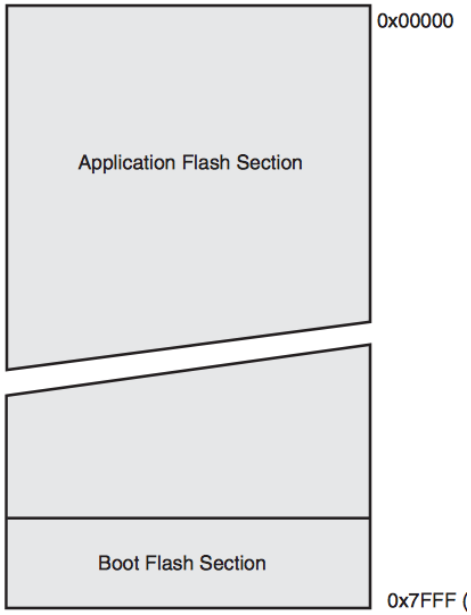


Hard disk

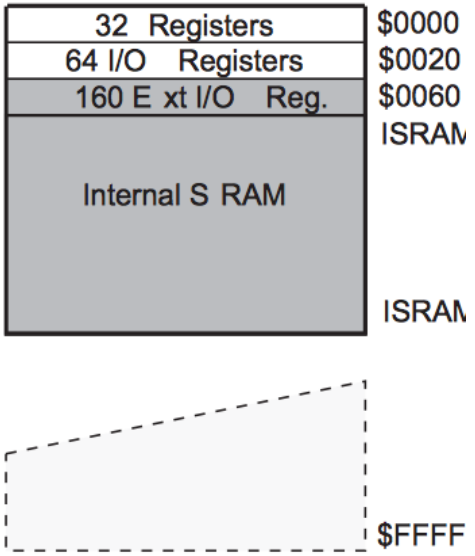
RAM



Program Memory



Data Memory

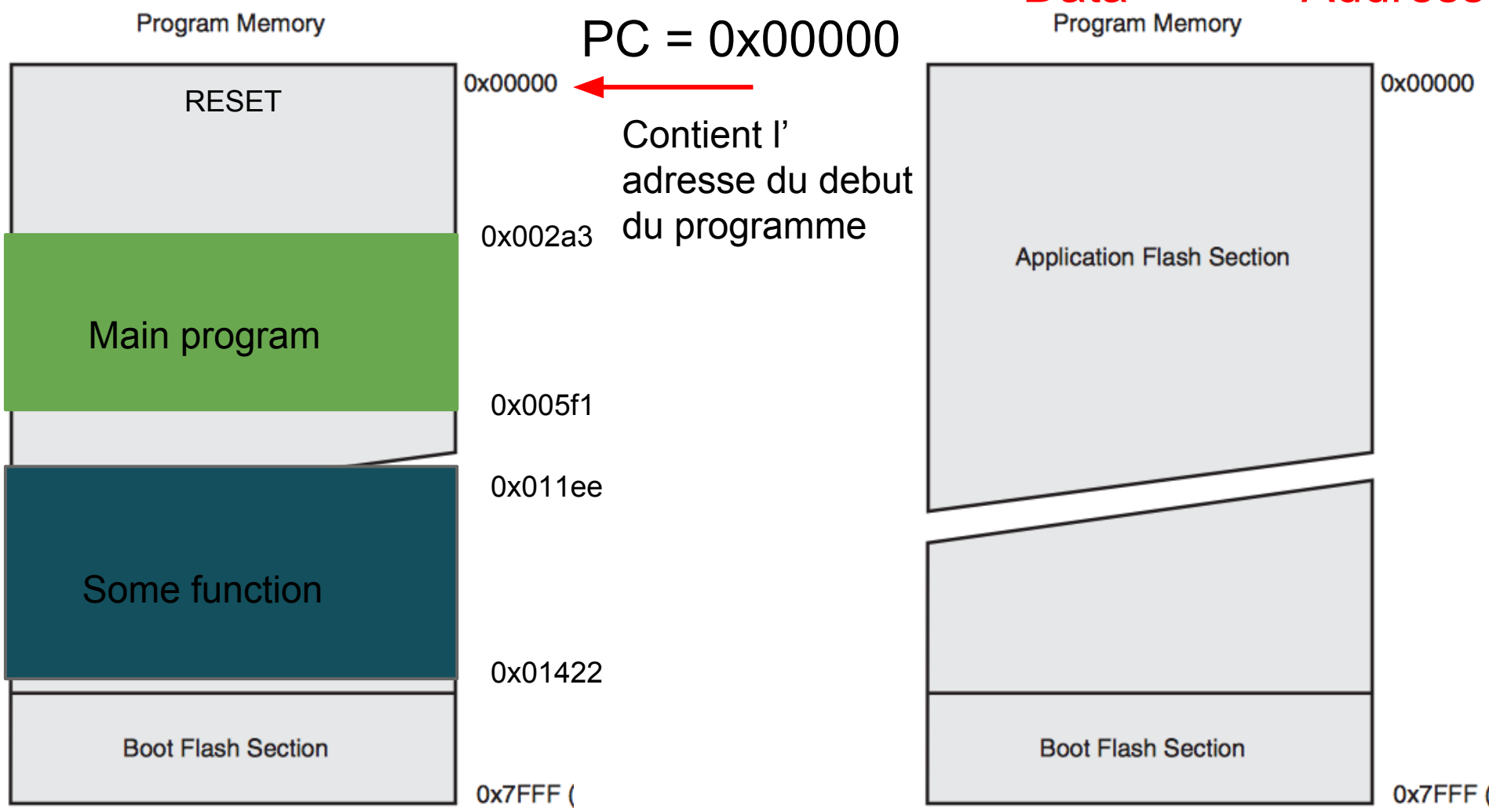


Atmel AtMega32U4 datasheet http://www.atmel.com/Images/doc7766.pdf

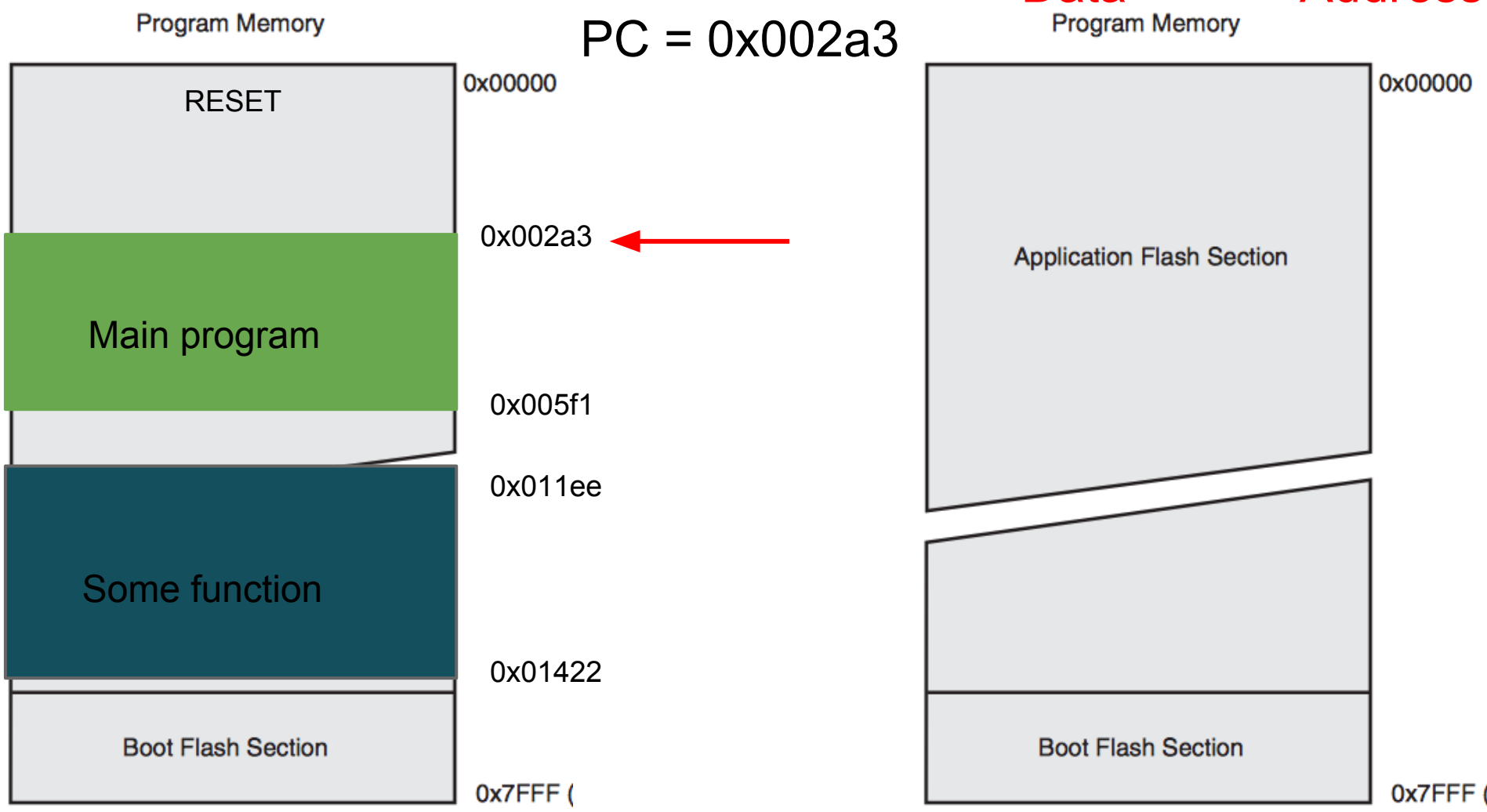
Mon PC

- N’est **pas** un Personal Computer
- Plutôt, un Program Counter
- Marque/compte quelle ligne de code/instruction est à executer
- Au reset, PC = $0x0000
- $ → Adresse, 0x → hexadecimal

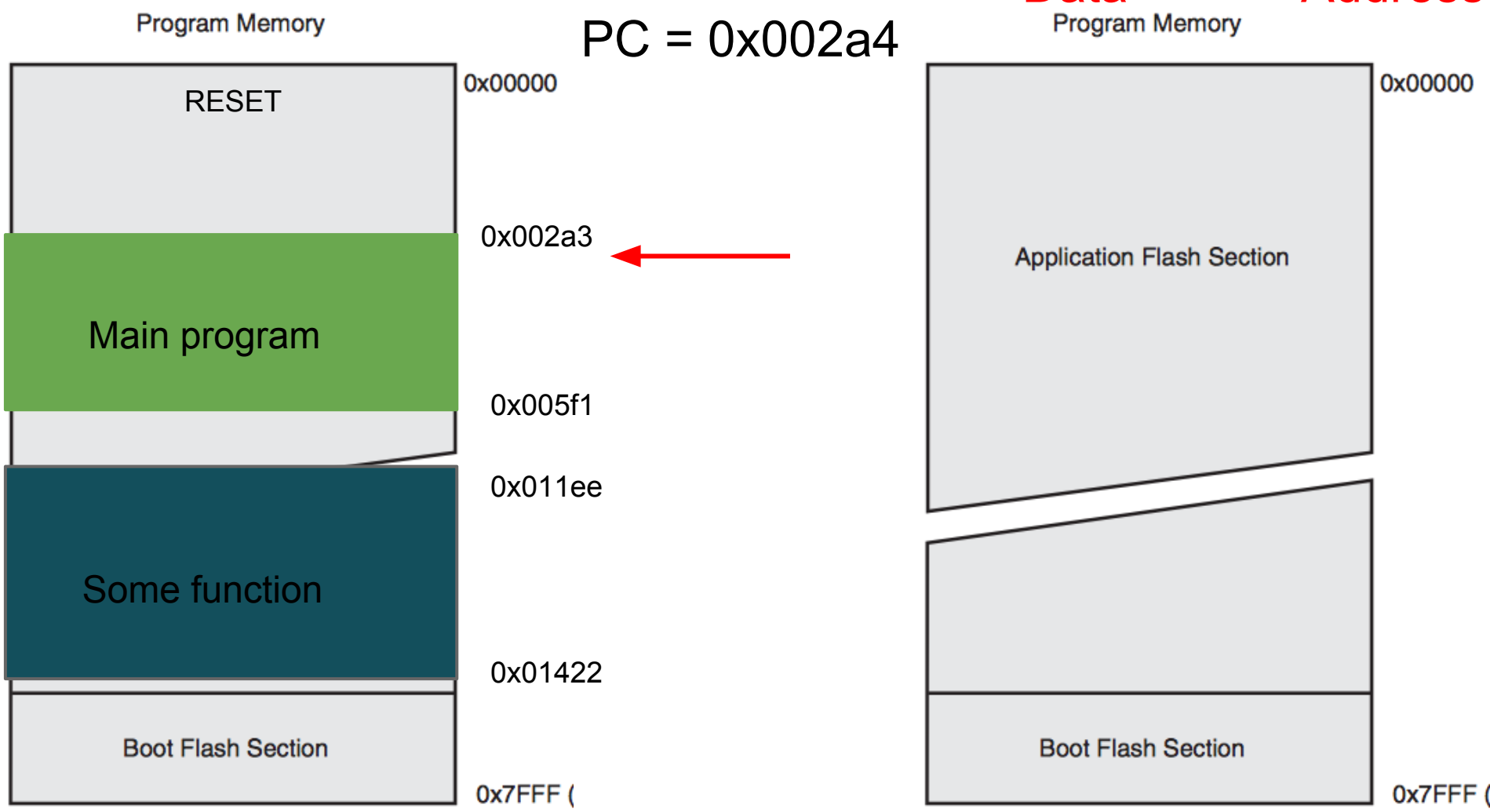
Memoire programme



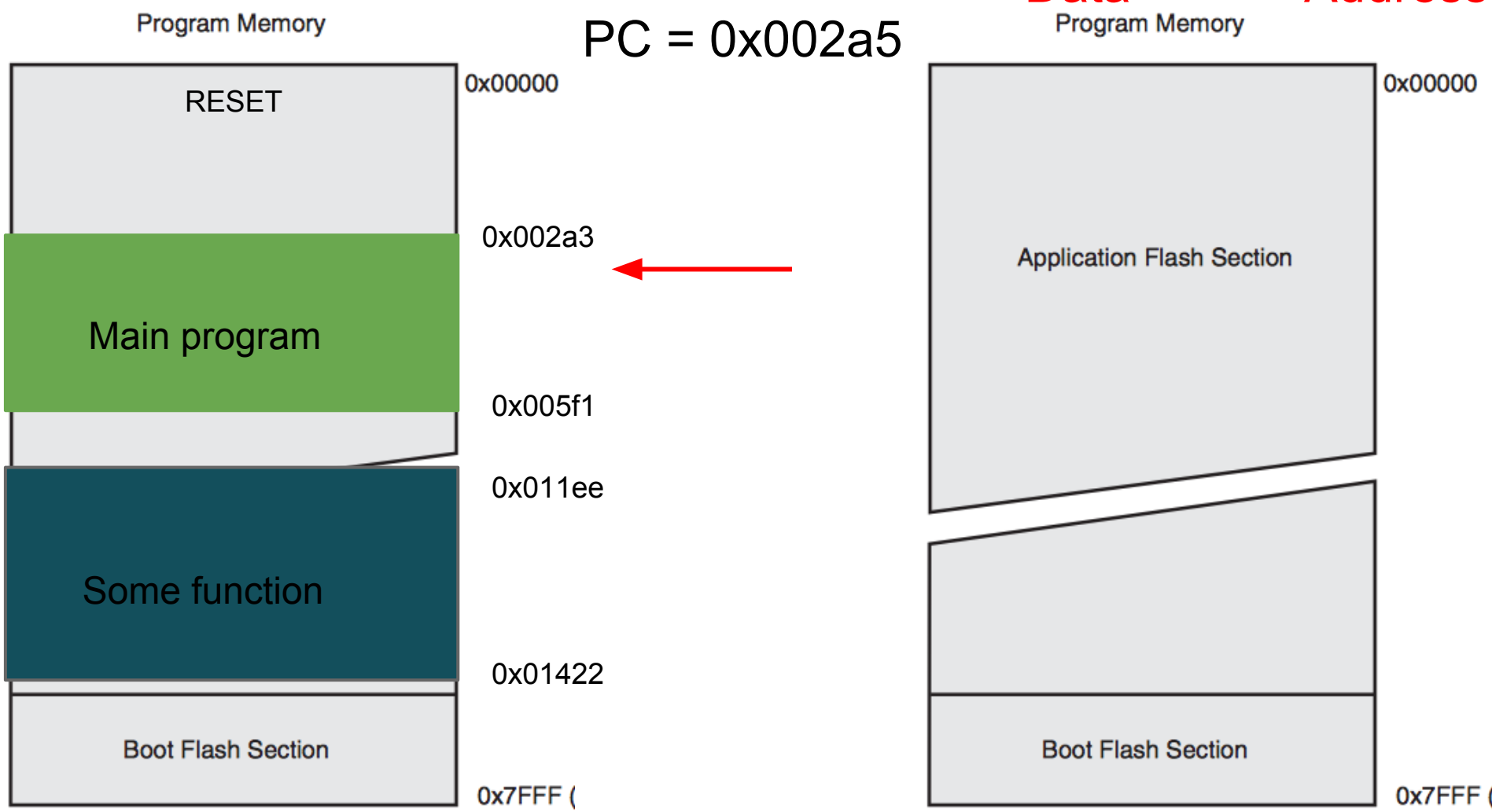
Memoire programme



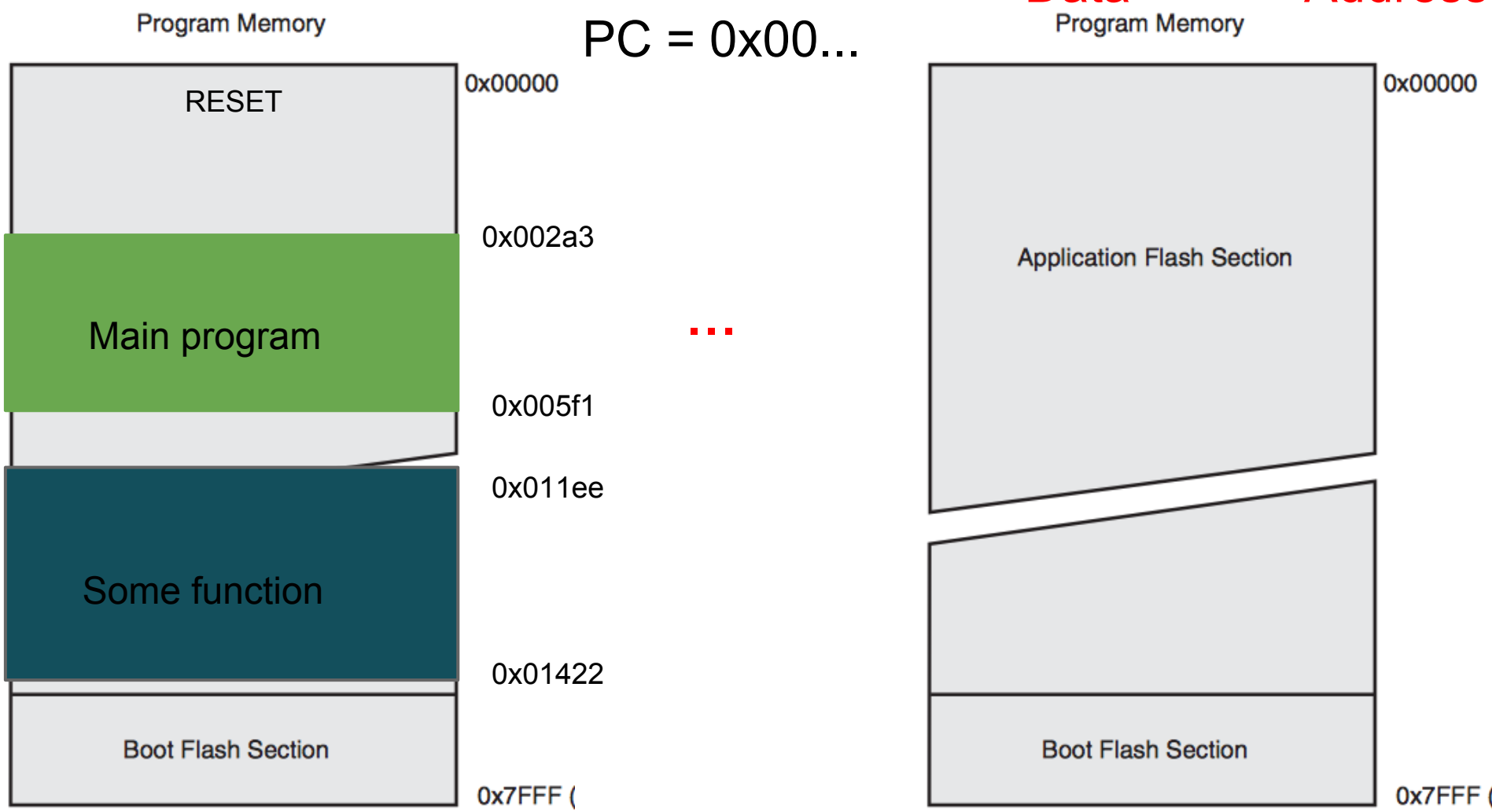
Memoire programme



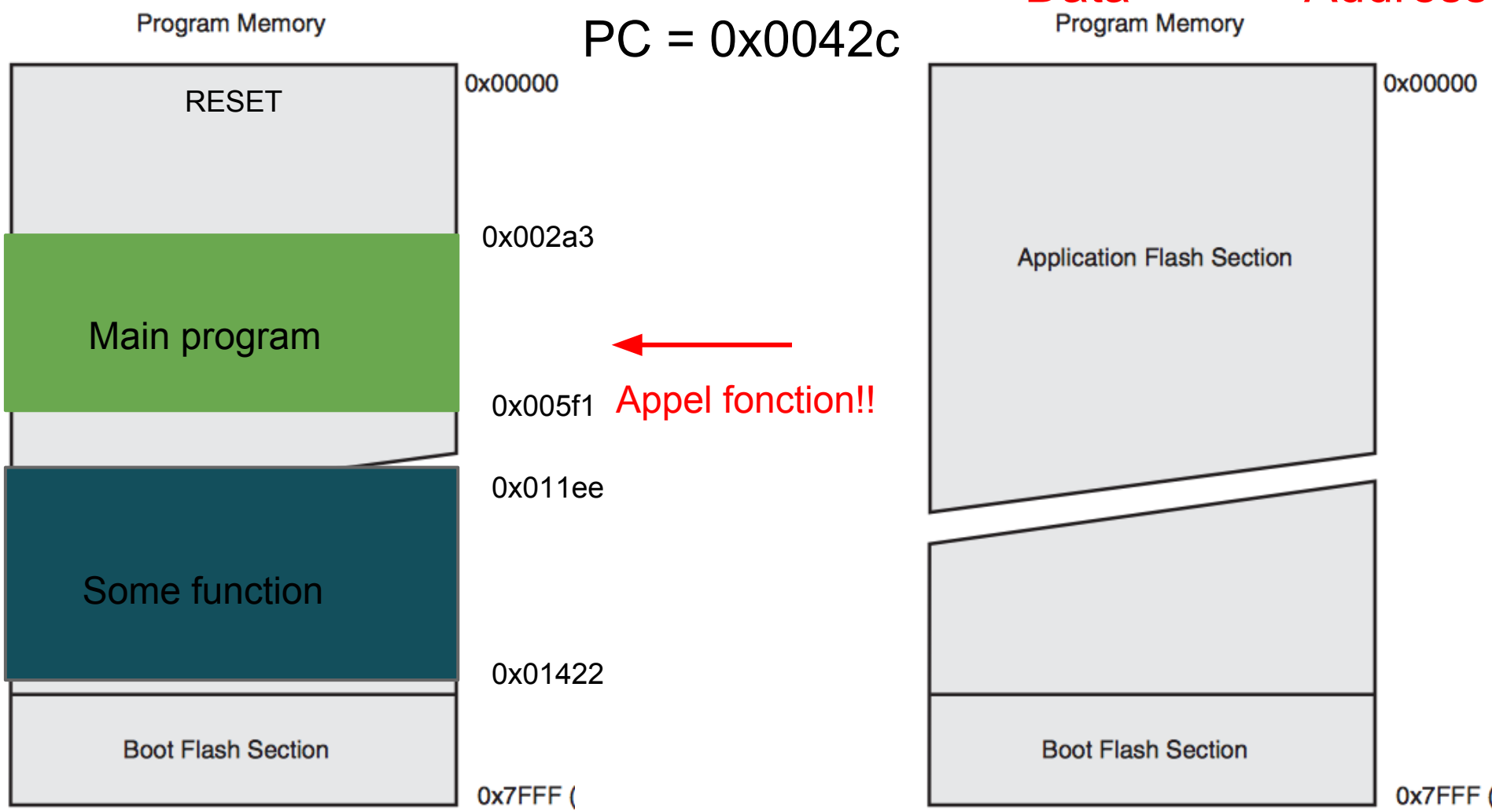
Memoire programme



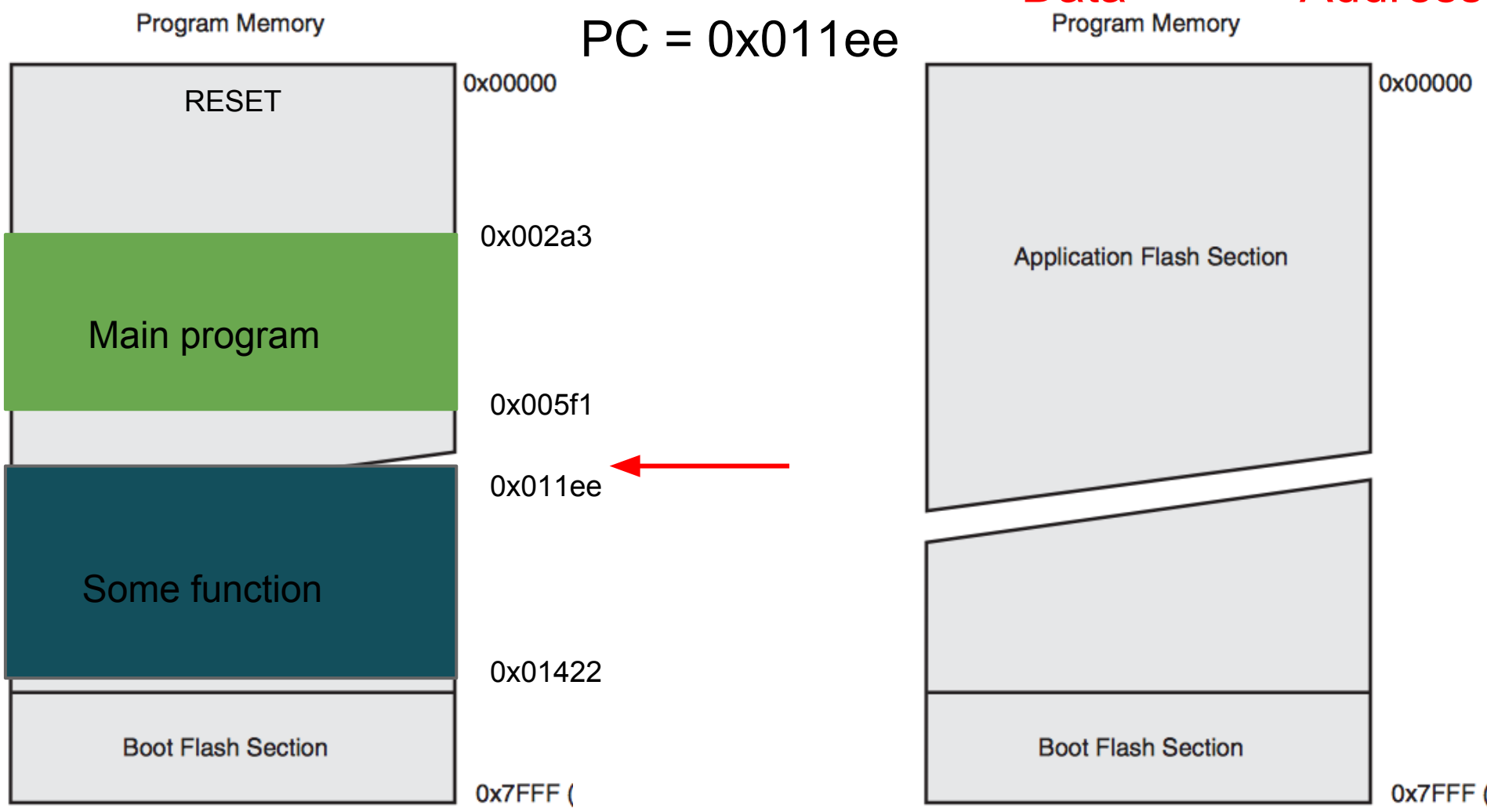
Memoire programme



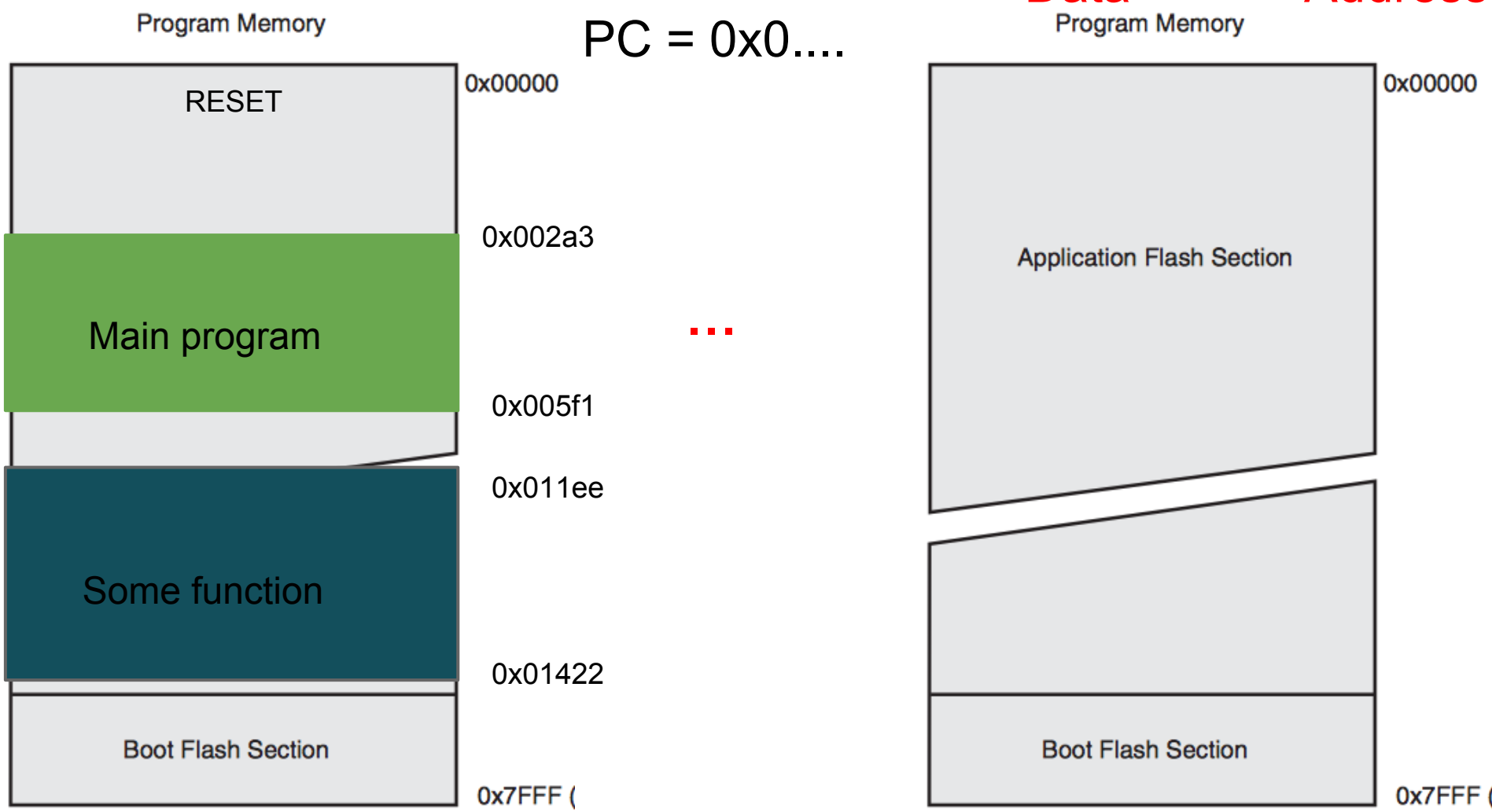
Memoire programme



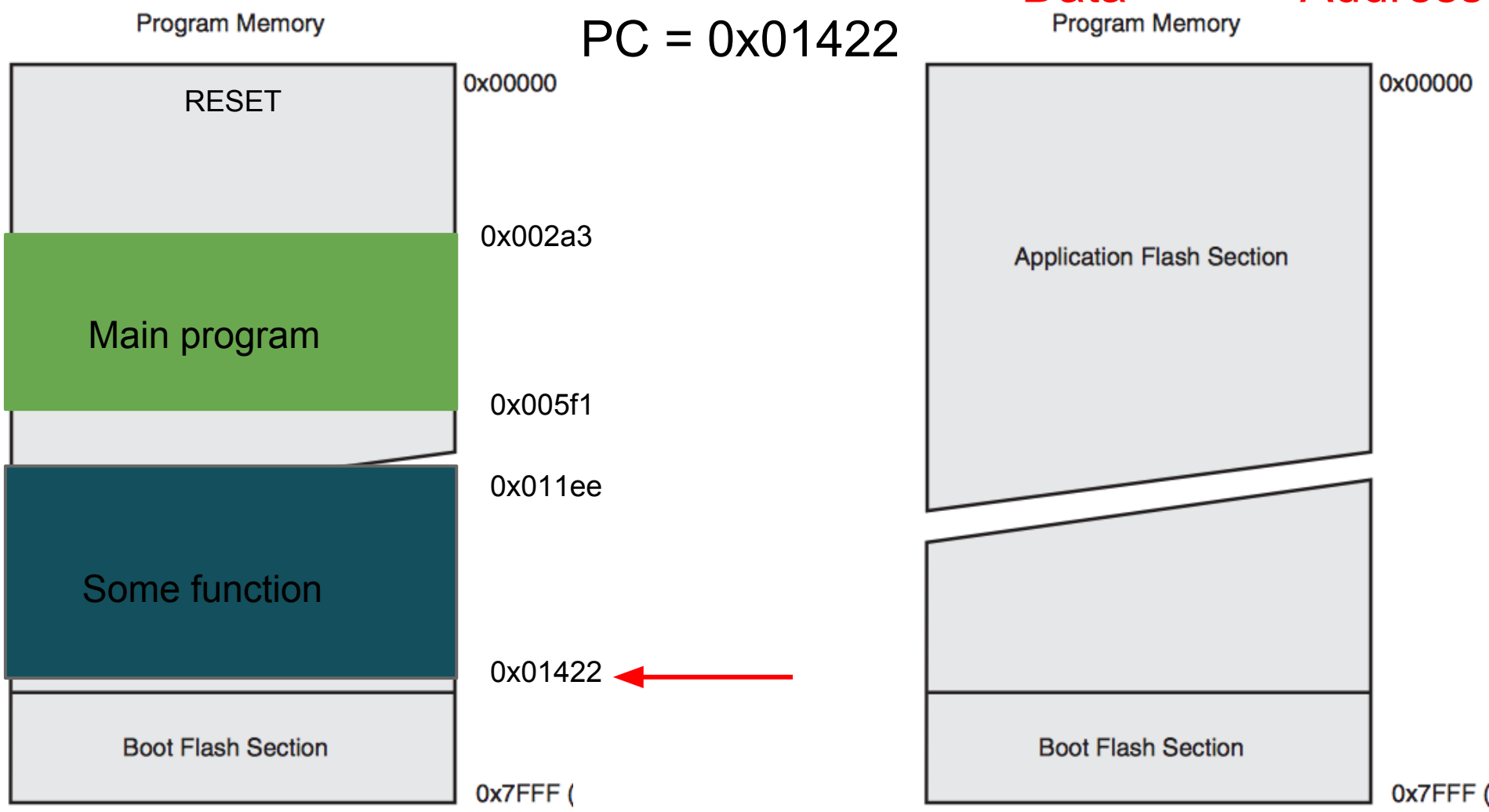
Memoire programme



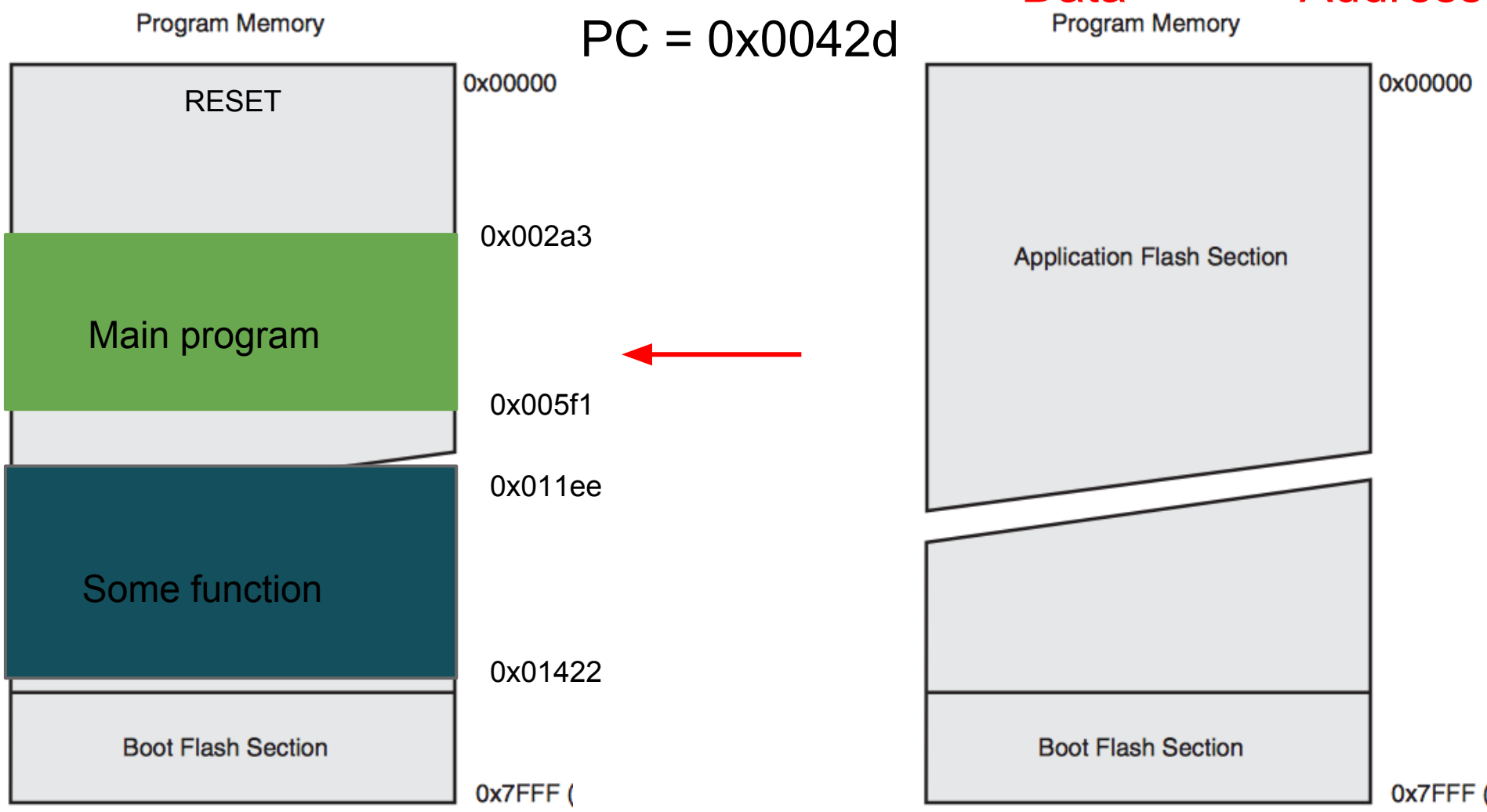
Memoire programme



Memoire programme



Memoire programme



Interruptions, the hard way!

Example pour interruption externe numero 3

Table 9-1. Reset and Interrupt Vectors

Vector No.	Program Address(2)	Source	Interrupt Definition
1	$0000(1)	RESET	External Pin, Power-on Reset, Brown-out Reset, Watchdog Reset, and JTAG AVR Reset
2	$0002	INT0	External Interrupt Request 0
3	$0004	INT1	External Interrupt Request 1
4	$0006	INT2	External Interrupt Request 2
5	$0008	INT3	External Interrupt Request 3
6	$000A	Reserved	Reserved
7	$000C	Reserved	Reserved
8	$000E	INT6	External Interrupt Request 6
9	$0010	Reserved	Reserved

External Interrupt Mask Register – EIMSK

Bit	7	6	5	4	3	2	1	0	
	-	INT6	-	-	INT3	INT2	INT1	IINT0	EIMSK
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

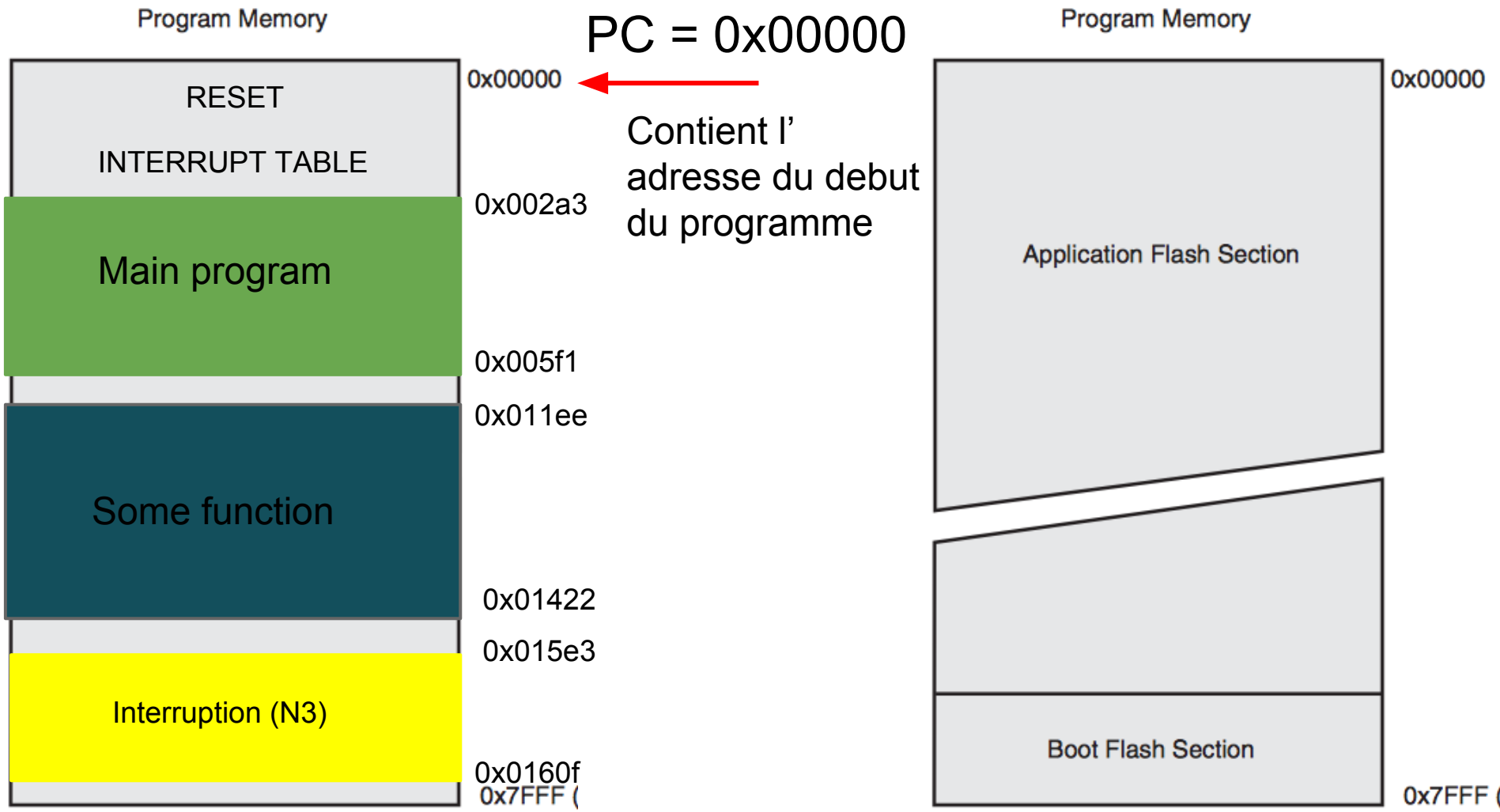
Atmel AtMega32U4 datasheet http://www.atmel.com/Images/doc7766.pdf on page 61 and 87

Interruptions, the hard way!

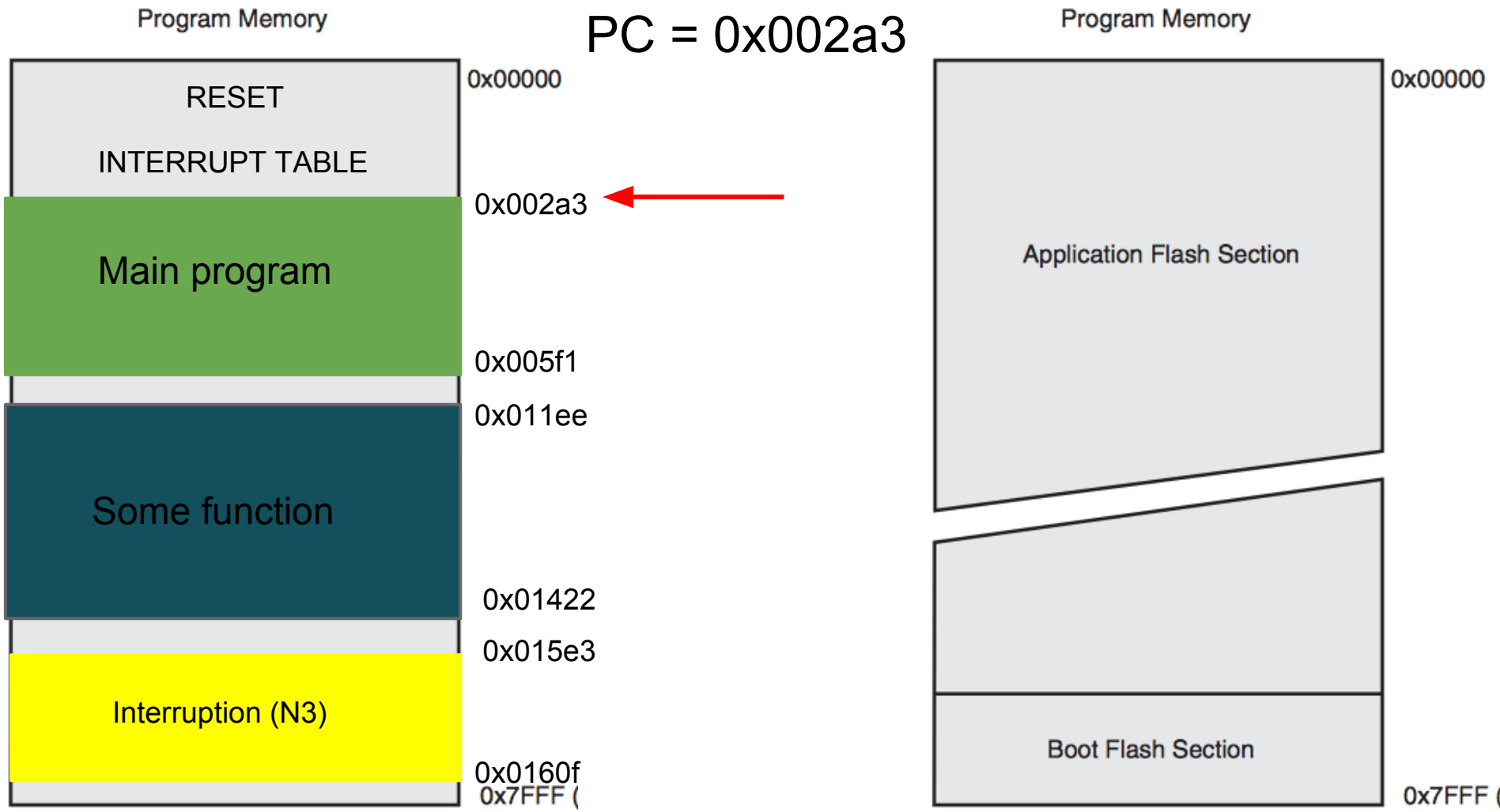
```
int main(void)
{
    // Your init
    asm("cli");       // disable global interrupt
    EIMSK |= (1 << INT3);
    // EIMSK |= 0b00001000; equivalent
    asm("sei");// enable global interrupt
}


ISR(INT3_vect)
{
    // Your interrupt code
}
```

Atmel AtMega32U4 datasheet http://www.atmel.com/Images/doc7766.pdf on page 61 and 87

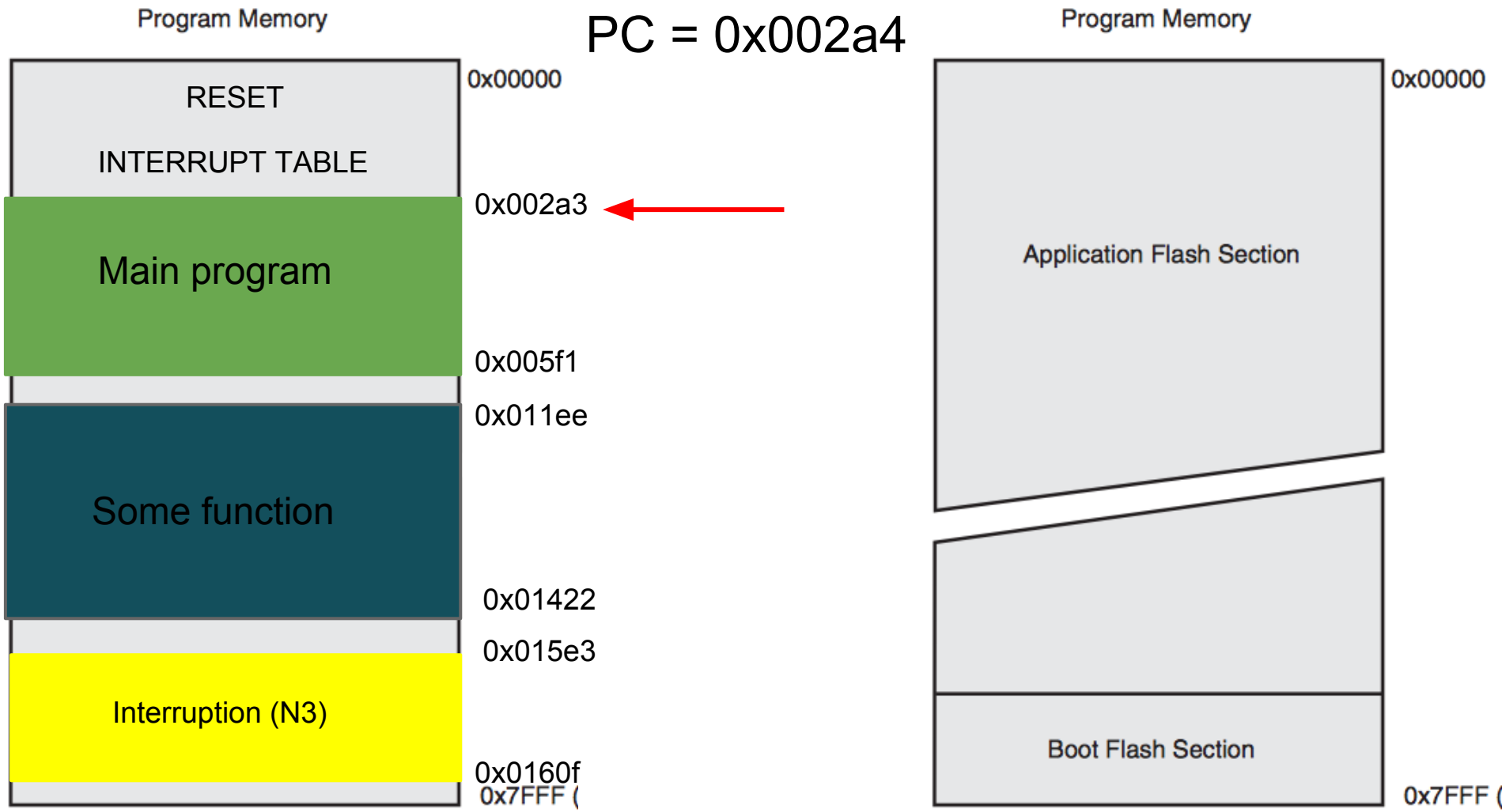
Memoire programme - Interruption!



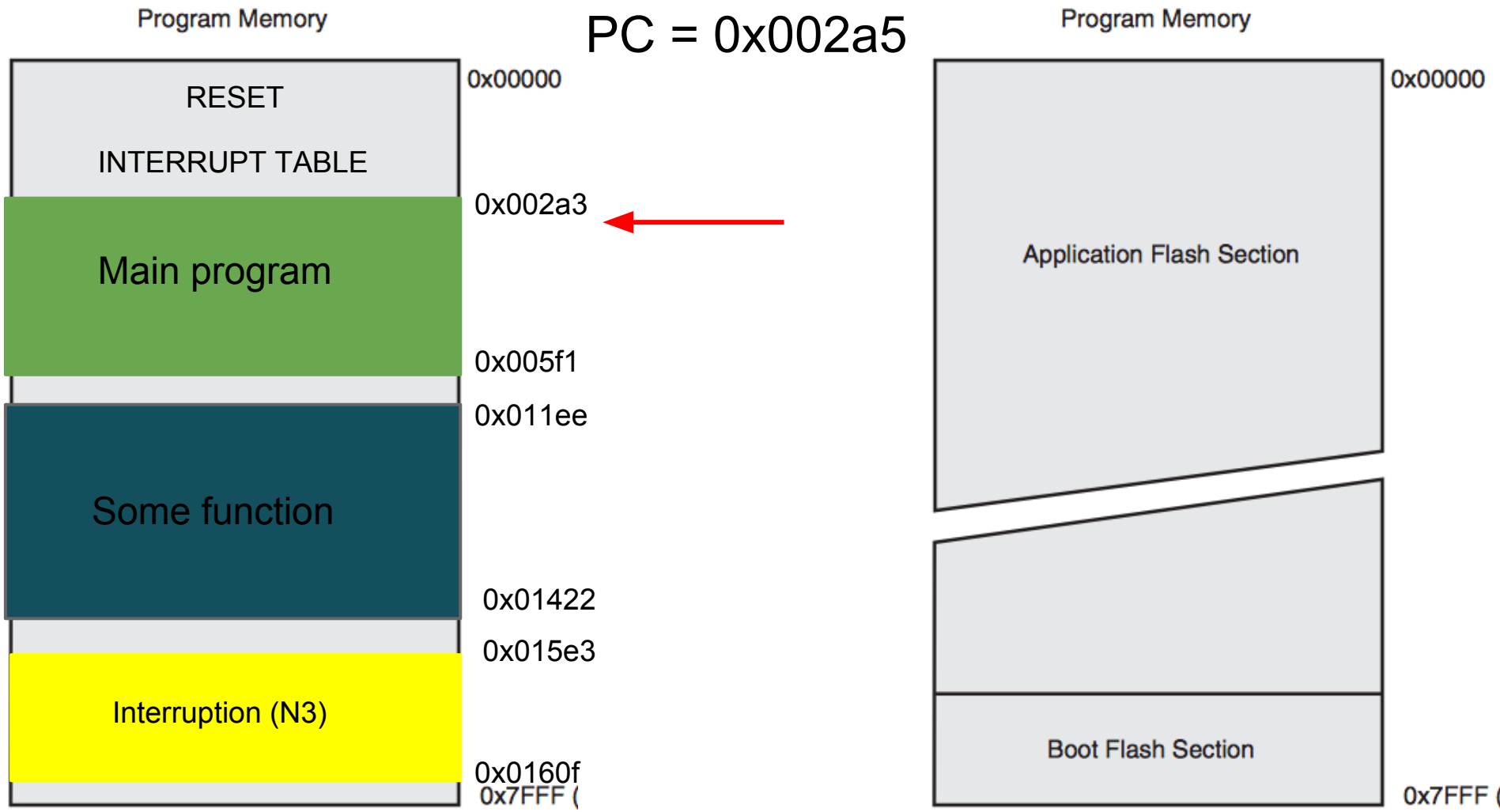
Memoire programme - Interruption!



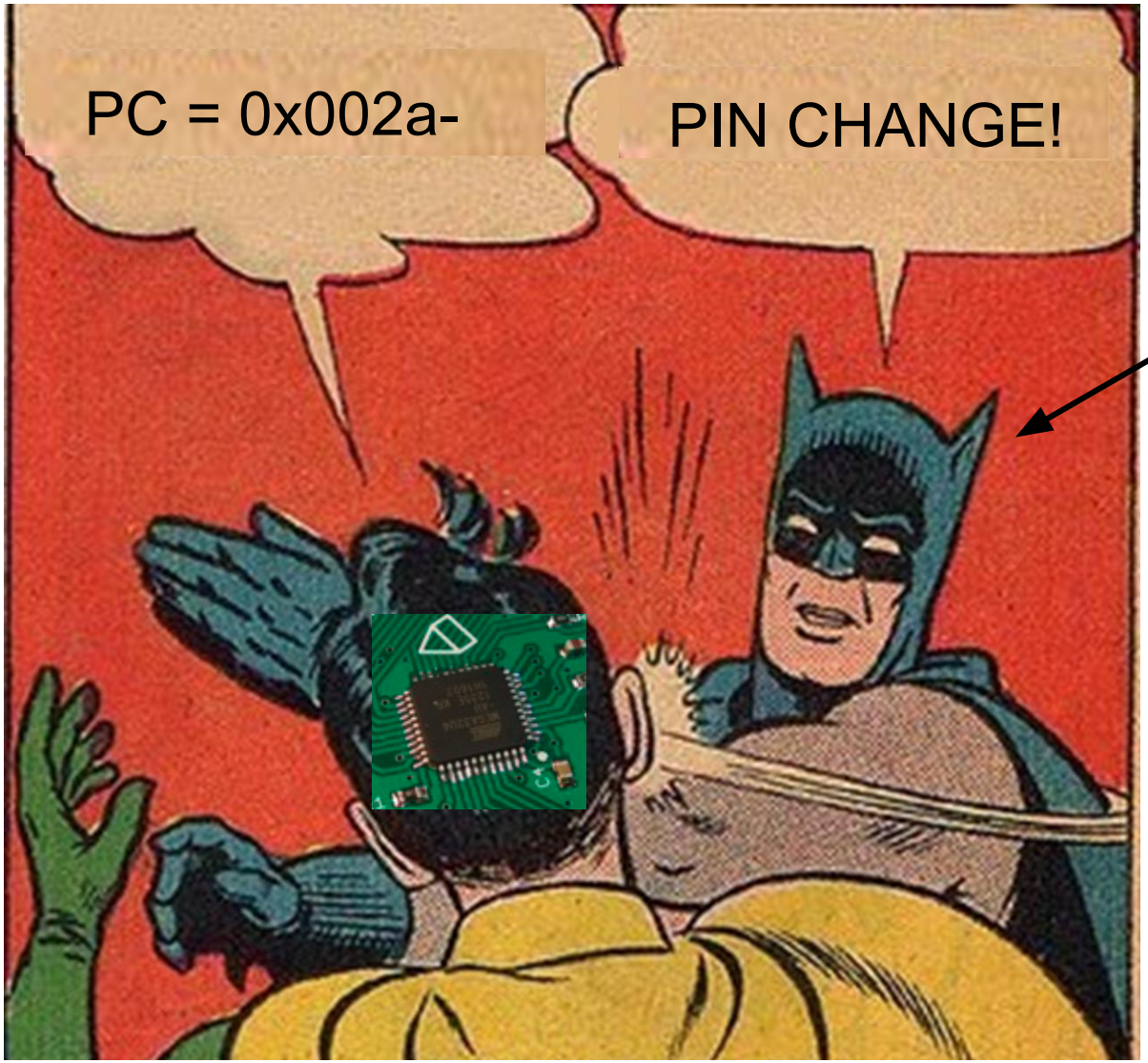
Memoire programme - Interruption!



Memoire programme - Interruption!



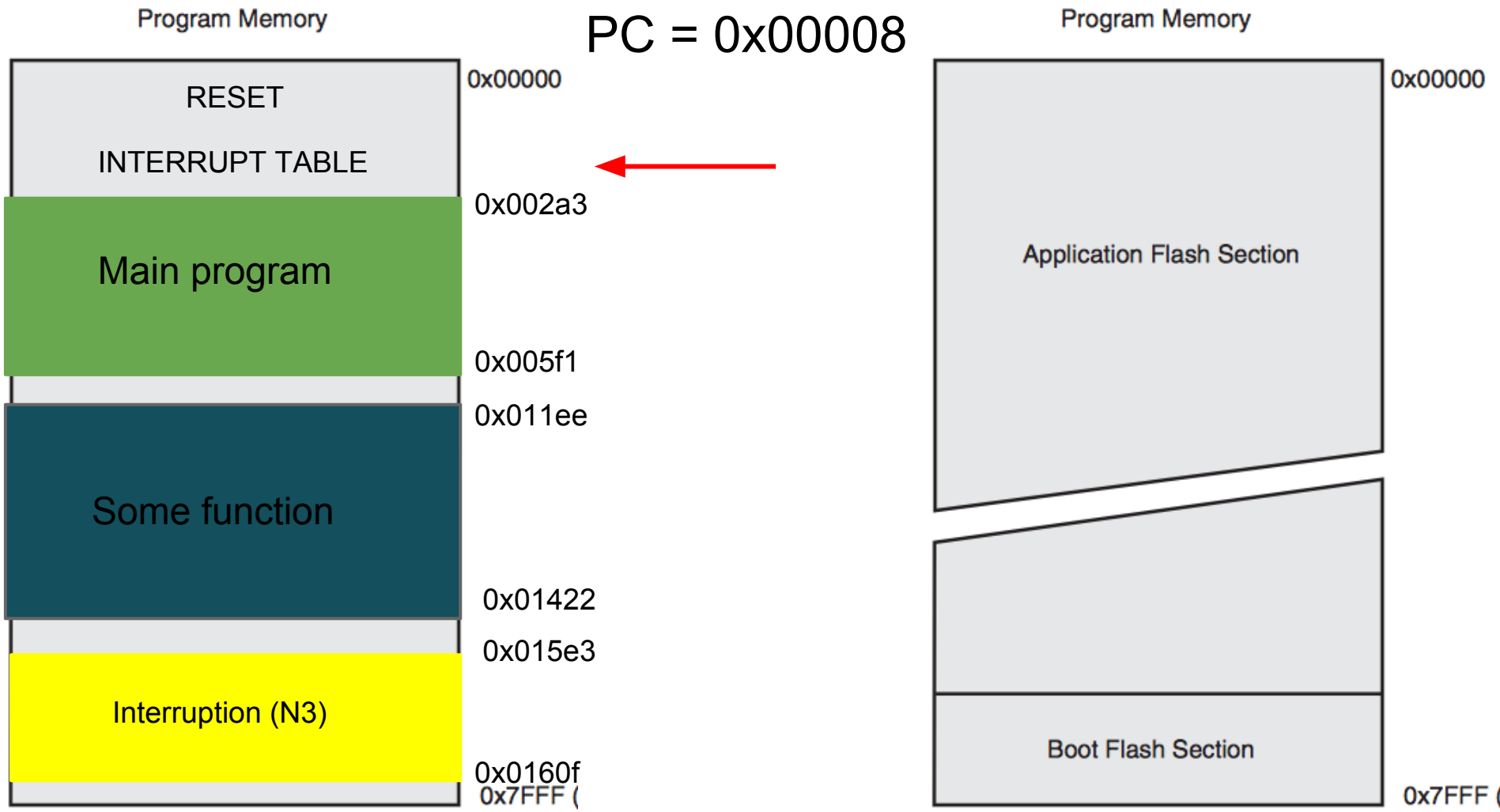
Memoire programme - Interruption!



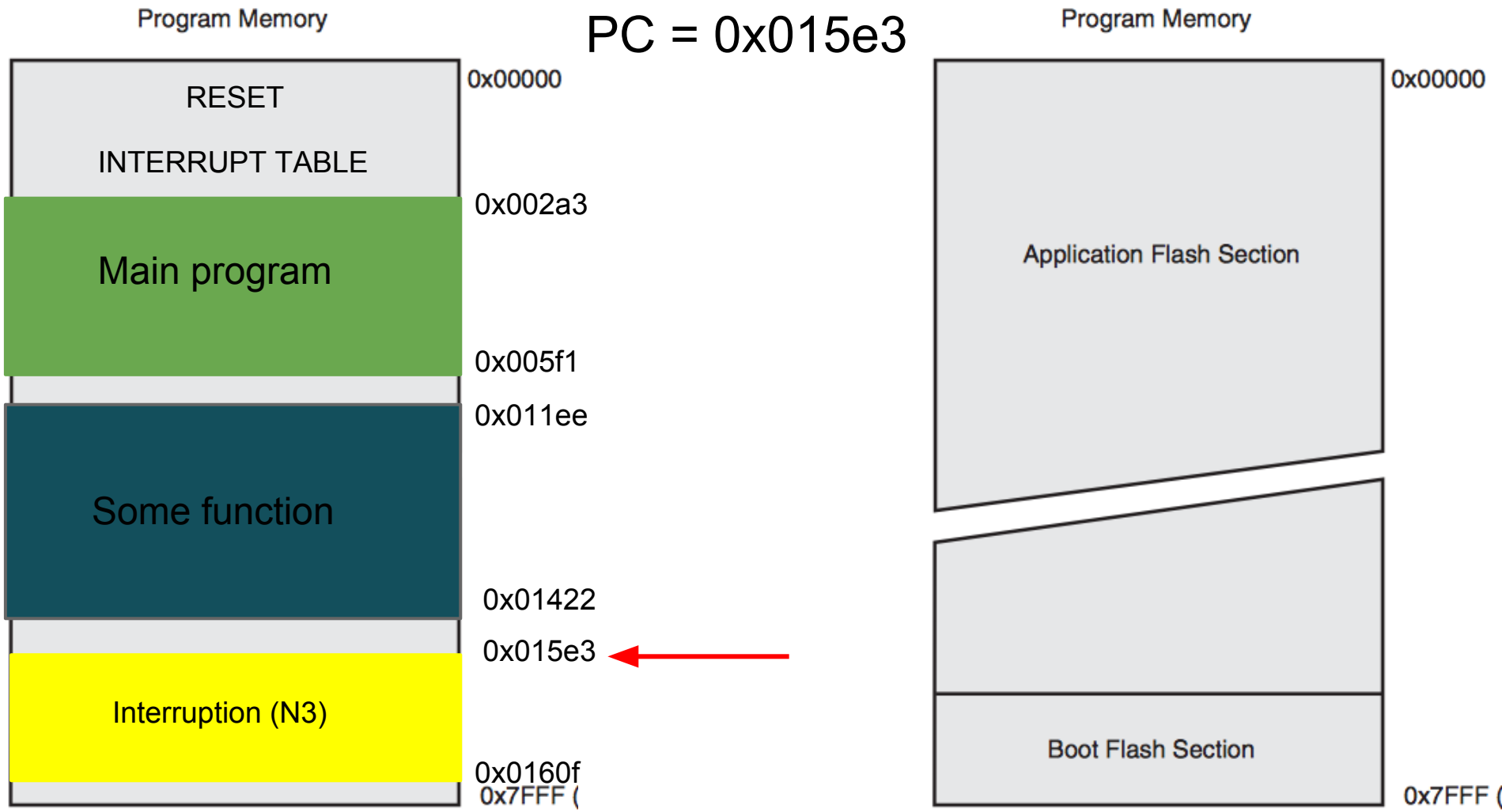
External World

Dans ce cas, on admet c'est l'interruption numero 3

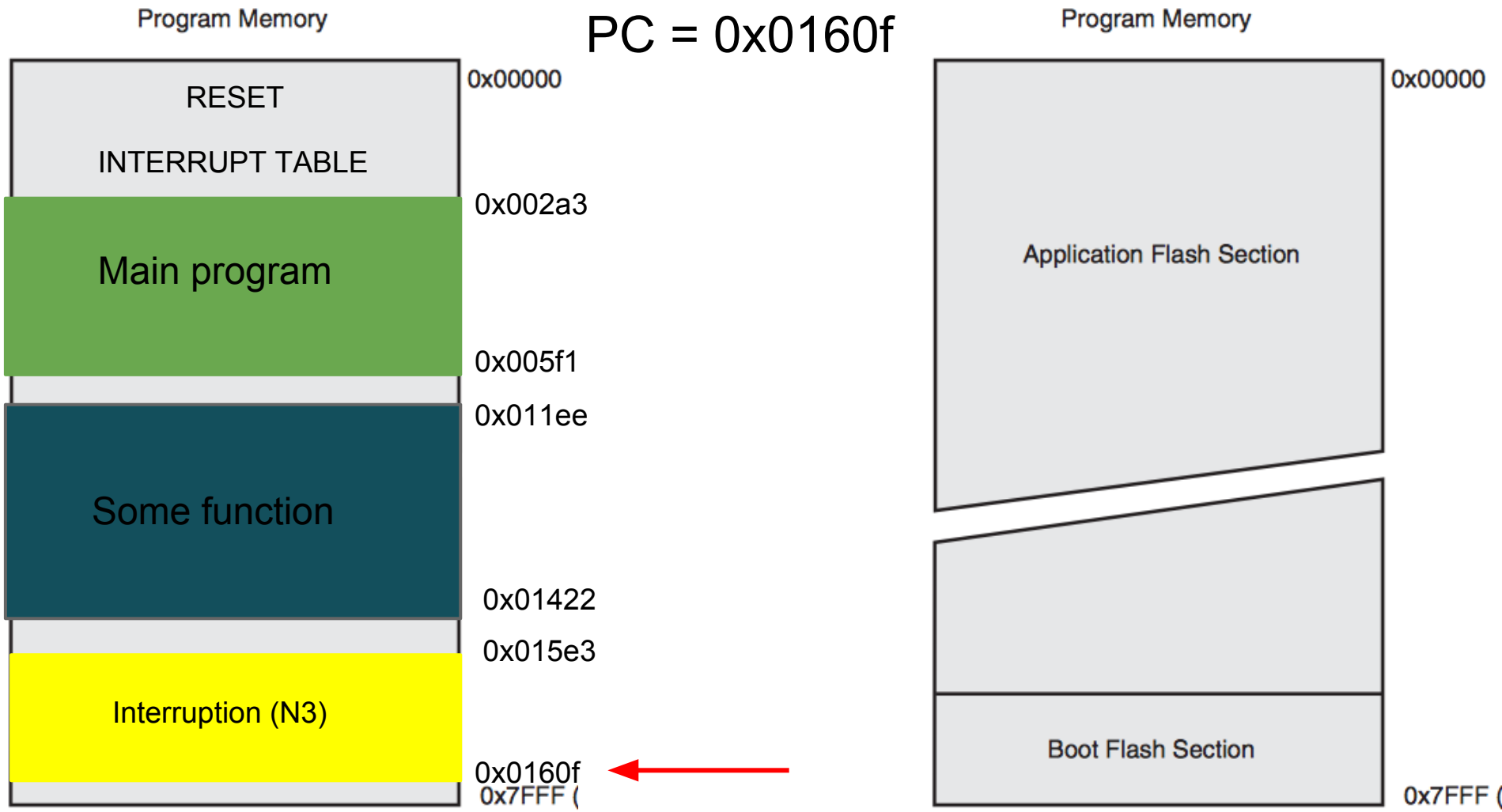
Memoire programme - Interruption!



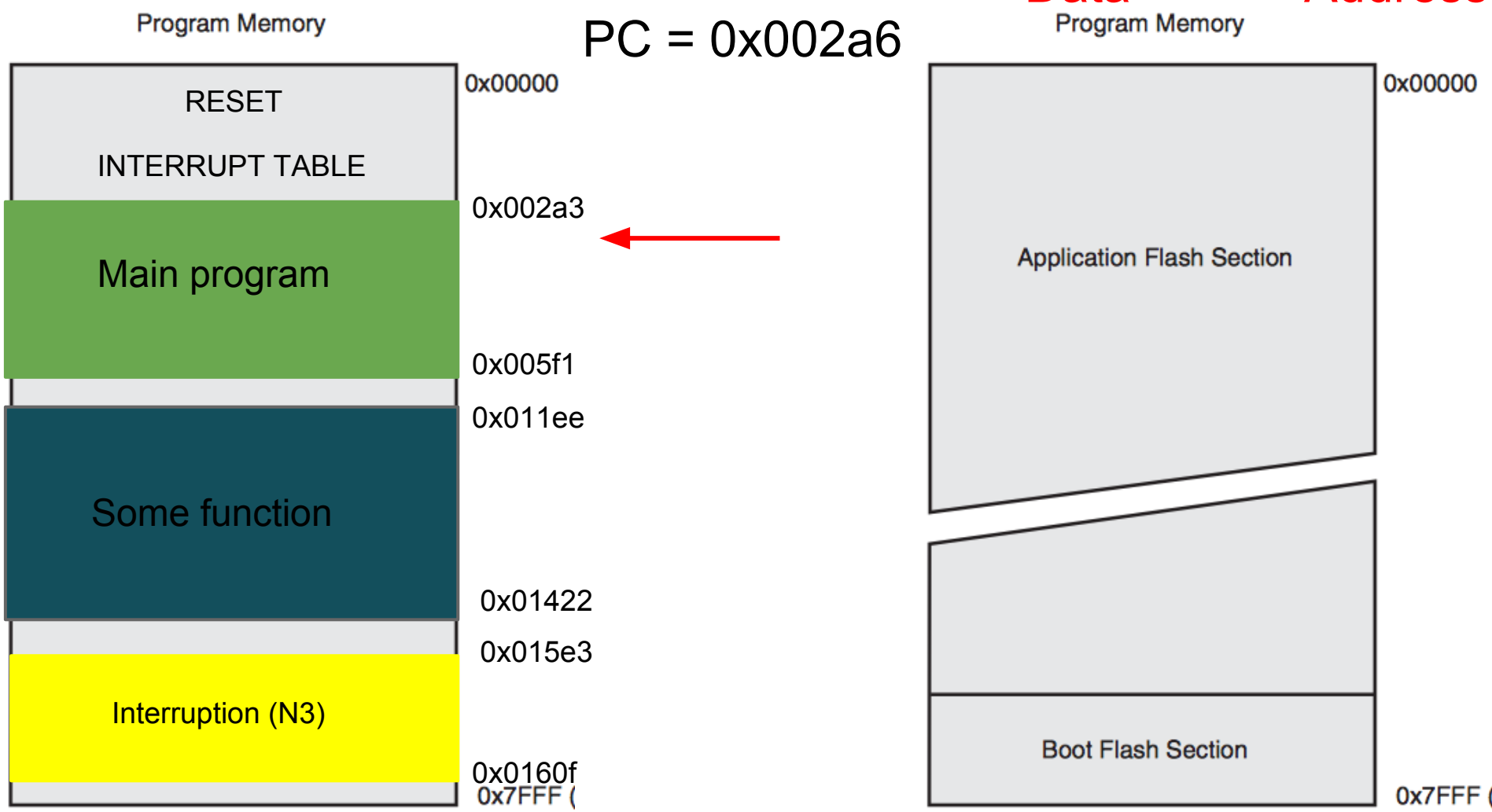
Memoire programme - Interruption!



Memoire programme - Interruption!



Memoire programme - Interruption!



Interruptions

- En C: Ne pas oublier de les activer!
- Ligne en C: asm(“sei”);
- Activer les interruptions APRES les avoir attachés/declarés

- Pour desactiver?
- Ligne en C: asm(“cli”);

- Arduino: attachInterrupt & detachInterrupt

Remarque moteurs

```
void loop()
{
  setSpeed(100,100);
}
```

NE marche PAS

```
void loop()
{
  setSpeed(100,100);
  delay(1);
}
```

Fonctionne

Prochains évenements

Lundi 25 novembre
Invité: Steven, avec des QUADCOPTERS!

???


HEVRON

PROFIT!

Lundi 25 novembre
Invité: Steven, avec des QUADCOPTERS!

Lundi 2 decembre
Presentation reglès grand concours

Dimanche 6 avril 2014
Grand concours! Pendant l'inauguration du convention center

FIN

Questions?