

Architecture des ordinateurs 2

Cours 9 - Les interruptions AVR

Halim Djerroud <hdd@ai.univ-paris8.fr>

LIASD - Université Paris 8

Janvier 2018

Vider un registre

- En informatique, une interruption est un arrêt temporaire de l'exécution normale d'un programme informatique par le microprocesseur afin d'exécuter un autre programme (appelé service d'interruption). (Wikipedia)
- Lors d'une interruption, le processeur sauve dans la pile l'adresse.
- Une fois le traitement de l'interruption terminé le programme revient à l'endroit où il s'est arrêté pour continuer l'exécution de programme principale.

Qui peut provoquer une interruption

- Compteurs
- Interfaces de communication
- Convertisseur analogique-numérique (CAN)
- Comparateur
- Boches d'entrée-sortie spéciales
- Plusieurs autres fonctions, selon le contrôleur

Vecteur d'interruptions

```
00000000 <__vectors>:
  0: 0c 94 34 00  jmp 0x68 ; 0x68 <__ctors_end>
  4: 0c 94 49 00  jmp 0x92 ; 0x92 <__bad_interrupt>
  8: 0c 94 49 00  jmp 0x92 ; 0x92 <__bad_interrupt>
  c: 0c 94 49 00  jmp 0x92 ; 0x92 <__bad_interrupt>
 10: 0c 94 49 00  jmp 0x92 ; 0x92 <__bad_interrupt>
 14: 0c 94 49 00  jmp 0x92 ; 0x92 <__bad_interrupt>
  ...
 5c: 0c 94 49 00  jmp 0x92 ; 0x92 <__bad_interrupt>
 60: 0c 94 49 00  jmp 0x92 ; 0x92 <__bad_interrupt>
 64: 0c 94 49 00  jmp 0x92 ; 0x92 <__bad_interrupt>
```

Vecteur d'interruptions

```
00000068 <__ctors_end>:  
68: 11 24          eor r1, r1  
6a: 1f be          out 0x3f, r1 ; 63  
6c: cf ef          ldi r28, 0xFF ; 255  
6e: d8 e0          ldi r29, 0x08 ; 8  
70: de bf          out 0x3e, r29 ; 62  
72: cd bf          out 0x3d, r28 ; 61
```

Vecteur d'interruptions

```
00000092 <__bad_interrupt>:  
  92: 0c 94 00 00  jmp 0 ; 0x0 <__vectors>
```

Vecteur d'interruptions

Address	Labels	Code	Comments
0x0000		jmp	RESET ; Reset
0x0002		jmp	INT0 ; IRQ0
0x0004		jmp	INT1 ; IRQ1
0x0006		jmp	PCINT0 ; PCINT0
0x0008		jmp	PCINT1 ; PCINT1
0x000A		jmp	PCINT2 ; PCINT2
0x000C		jmp	WDT ; Watchdog Timeout
0x000E		jmp	TIM2_COMPA ; Timer2 CompareA
0x0010		jmp	TIM2_COMPB ; Timer2 CompareB
0x0012		jmp	TIM2_OVF ; Timer2 Overflow
0x0014		jmp	TIM1_CAPT ; Timer1 Capture
0x0016		jmp	TIM1_COMPA ; Timer1 CompareA
0x0018		jmp	TIM1_COMPB ; Timer1 CompareB
0x001A		jmp	TIM1_OVF ; Timer1 Overflow
0x001C		jmp	TIM0_COMPA ; Timer0 CompareA
0x001E		jmp	TIM0_COMPB ; Timer0 CompareB
0x0020		jmp	TIM0_OVF ; Timer0 Overflow
0x0022		jmp	SPI_STC ; SPI Transfer Complete
0x0024		jmp	USART_RXC ; USART RX Complete
0x0026		jmp	USART_UDRE ; USART UDR Empty
0x0028		jmp	USART_TXC ; USART TX Complete
0x002A		jmp	ADC ; ADC Conversion Complete
0x002C		jmp	EE_RDY ; EEPROM Ready
0x002E		jmp	ANA_COMP ; Analog Comparator
0x0030		jmp	TWI ; 2-wire Serial
0x0032		jmp	SPM_RDY ; SPM Ready
;			
0x0034	RESET:	ldi	r16,high(RAMEND) ; Main program start
0x0035		out	SPH,r16 ; Set Stack Pointer to top of RAM
0x0036		ldi	r16,low(RAMEND)
0x0037		out	SPL,r16
0x0038		sei	; Enable interrupts

FIGURE – Vecteur d'interruption Atmega 328p

Remarque sur les interruptions

- Seuls les vecteurs d'interruption INT0 et INT1 sont indépendants.
- INT0 ne s'applique qu'à la broche PD2.
- INT1 ne s'applique qu'à la broche PD3
- Les 3 vecteurs d'interruption PCIE0, PCIE1 et PCIE2 s'appliquent à un port complet
- l'identification de la broche à l'origine de l'interruption est à la charge du programmeur.
- Quand le Fuse BOOTRSF est programmé, au moment du Reset le micro saute à l'adresse du bootloader.

Pins

Pin-out

Figure 5-1. 28-pin PDIP

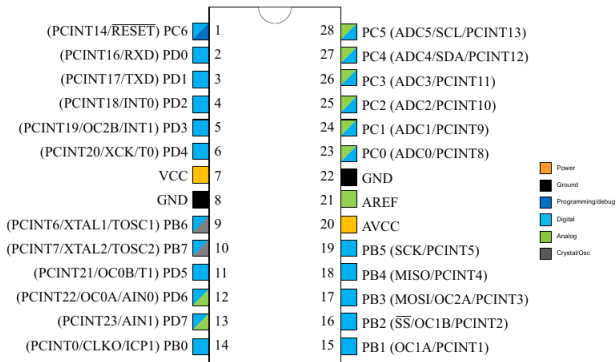


FIGURE – Pins Atmega 328p

Généralités sur les interruptions

- Les interruptions « Externes » sont déclenchées par les pins INT0, INT1 ou une quelconque des pins PCINT 0 à 23.
- Les interruptions restent fonctionnelles même si les pins INT0, INT1 ou PCINT 0 à 23 sont configurées en sortie.
- L'interruption PCI2 se déclenche si une quelconque des pins PCINT23 à 16 change d'état.
- L'interruption PCI1 se déclenche si une quelconque des pins PCINT14 à 8 change d'état.
- L'interruption PCI0 se déclenche si une quelconque des pins PCINT7 à 0 change d'état.

Interruptions INT0 et INT1

- Les interruptions INT0 et INT1 disposent chacune de leur propre vecteur d'interruption et peuvent être déclenchées sur un front montant, descendant, un niveau bas ou un basculement de niveau.
- La configuration s'effectue dans le registre EICRA.
- Choix niveau bas : l'interruption dure aussi longtemps que le niveau est maintenu à l'état bas.
-

EICRA

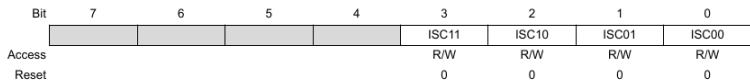


FIGURE – EICRA 328p

- ISC1n : Interrupt Sense Control 1 [n = 1 :0]
- ISC0n : Interrupt Sense Control 0 [n = 1 :0]

Configuration EICRA

Value	Description
00	The low level of INT1 generates an interrupt request.
01	Any logical change on INT1 generates an interrupt request.
10	The falling edge of INT1 generates an interrupt request.
11	The rising edge of INT1 generates an interrupt request.

FIGURE – Configuration EICRA 328p

SREG

Name: SREG**Offset:** 0x5F**Reset:** 0x00**Property:** When addressing as I/O Register: address offset is 0x3F

Bit	7	6	5	4	3	2	1	0
	I	T	H	S	V	N	Z	C
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

FIGURE – Configuration SREG 328p

- $I = 0$ → interdit toutes les interruptions de manière globale.
- $I = 1$ → autorise les interruptions de manière globale.
- Chaque interruption doit être individuellement autorisée et configurée dans son registre spécialisé.

Configuration EIMSK

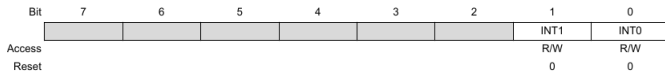


FIGURE – Configuration EIMSK 328p

- INT0 Activer ou désactiver les interruption
- INT1 Activer ou désactiver les interruption

Configuration EIFR

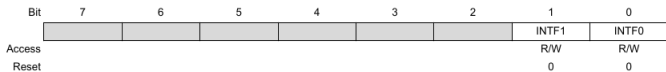


FIGURE – Configuration EIFR 328p

- INT0 mis à 1 si interruption
- INT1 mis à 1 si interruption

Configuration PCICR

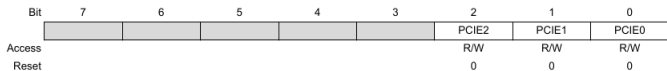


FIGURE – Configuration PCICR 328p

- PCIE2 Activer ou désactiver les interruption (pins : 23-16)
- PCIE1 Activer ou désactiver les interruption (pins : 14-8)
- PCIE0 Activer ou désactiver les interruption (pins : 7-0)

Configuration PCIFR



FIGURE – Configuration PCIFR 328p

- PCIE2 Mise a 1 en cas d'interruption (pins : 23-16)
- PCIE1 Mise a 1 en cas d'interruption (pins : 14-8)
- PCIE0 Mise a 1 en cas d'interruption (pins : 7-0)

Configuration PCMSK2

Bit	7	6	5	4	3	2	1	0
	PCINT23	PCINT22	PCINT21	PCINT20	PCINT19	PCINT18	PCINT17	PCINT16
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

FIGURE – Configuration PCMSK2 328p

- PCIE2 Masque pour activer les interruptions (pins : 23-16)

Configuration PCMSK1

Bit	7	6	5	4	3	2	1	0
		PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8
Access		R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0

FIGURE – Configuration PCMSK1 328p

- PCIE1 Masque pour activer les interruptions (pins : 14-8)

Configuration PCMSK0

Bit	7	6	5	4	3	2	1	0
Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

FIGURE – Configuration PCMSK0 328p

- PCIE0 Masque pour activer les interruptions (pins : 0-7)

Inclure le fichier Header

```
#include <avr/interrupt.h>

ISR(XXX_vect)
{
    // code dans l'interruption
}
```

- Pour utiliser des interruptions avec la librairie LibC de l'AVR, ajouter la fonction interrupt.h
- Le texte entre parenthèses après "ISR" est le nom de l'interruption.

Activer / désactiver les interruption Globales

```
...  
cli(); // Désactiver les interruptions  
...  
sei(); // Activer les interruptions  
...
```

Exemple

```
#include <avr/interrupt.h>
ISR(INT7_vect){
    // faire qqch
}
int main(){
    DDRE &= ~(1 << PIN7);
    PORTE |= (1 << PIN7);
    EICRB = (1 << ISC70);
    EIMSK |= (1 << INT7);
    sei();
    while (1) continue;
}
```