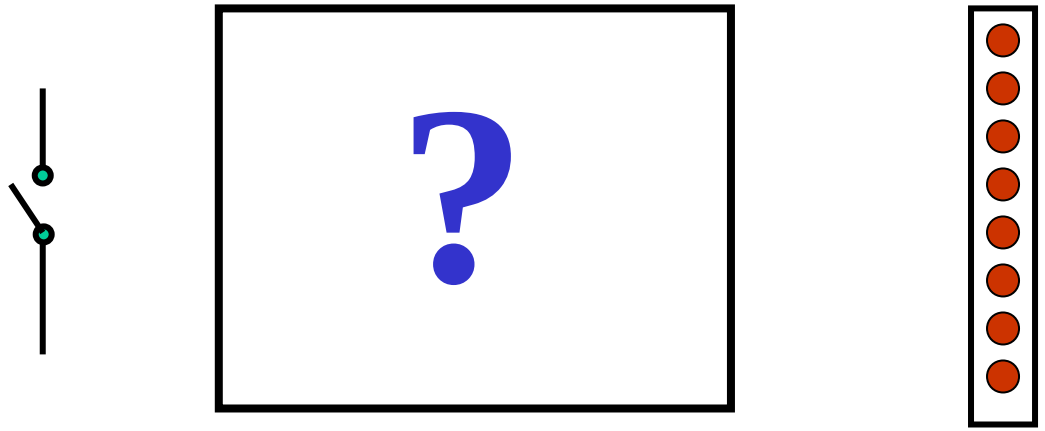
UNIVERSITÉ PARIS 12 VAL de MARNE

FACULTÉ DES SCIENCES ET TECHNOLOGIE

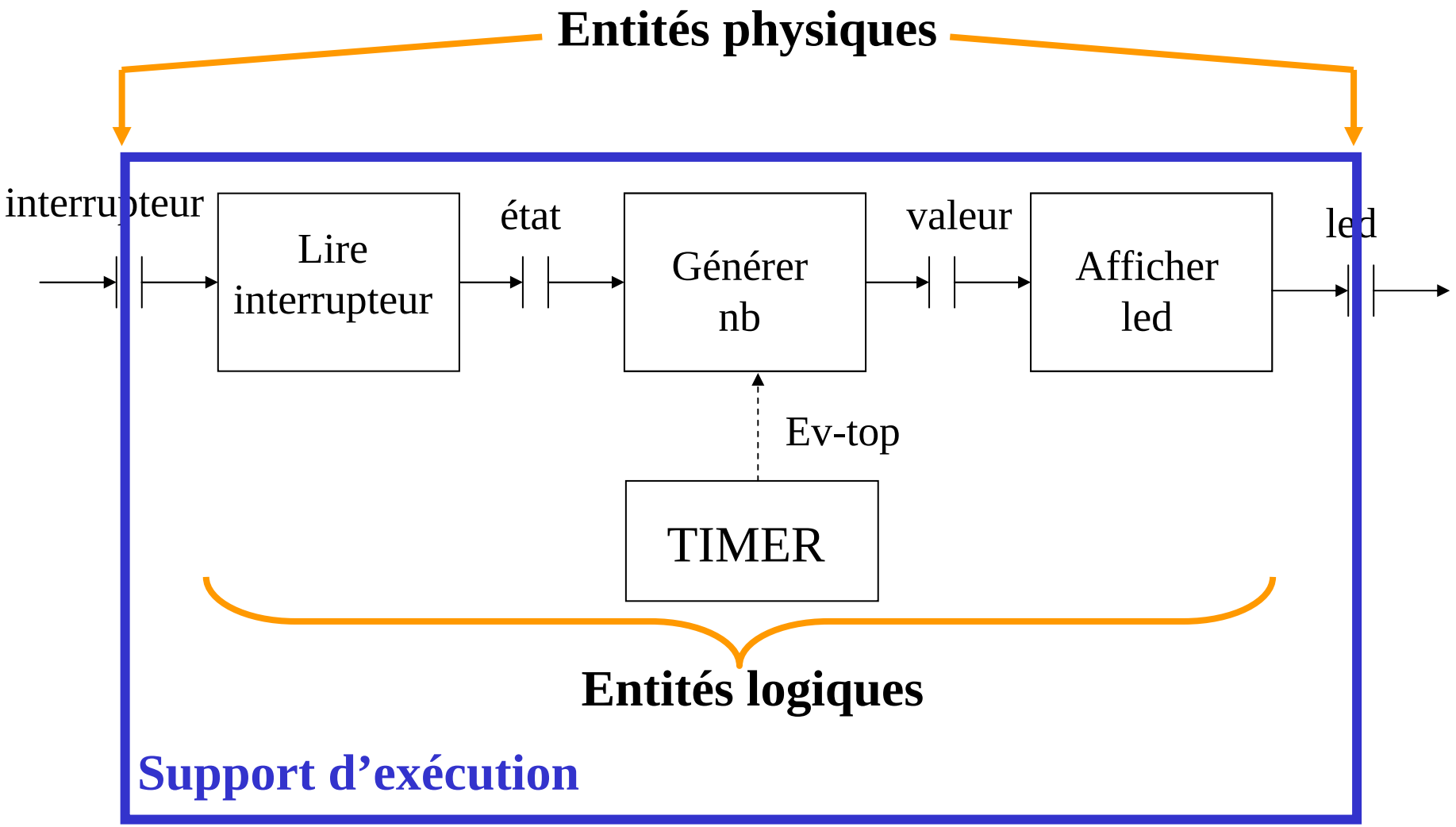
De l’application à la structure d’un microcontrôleur

Ph. Meyne – philippe.meyne@univ-paris12.fr

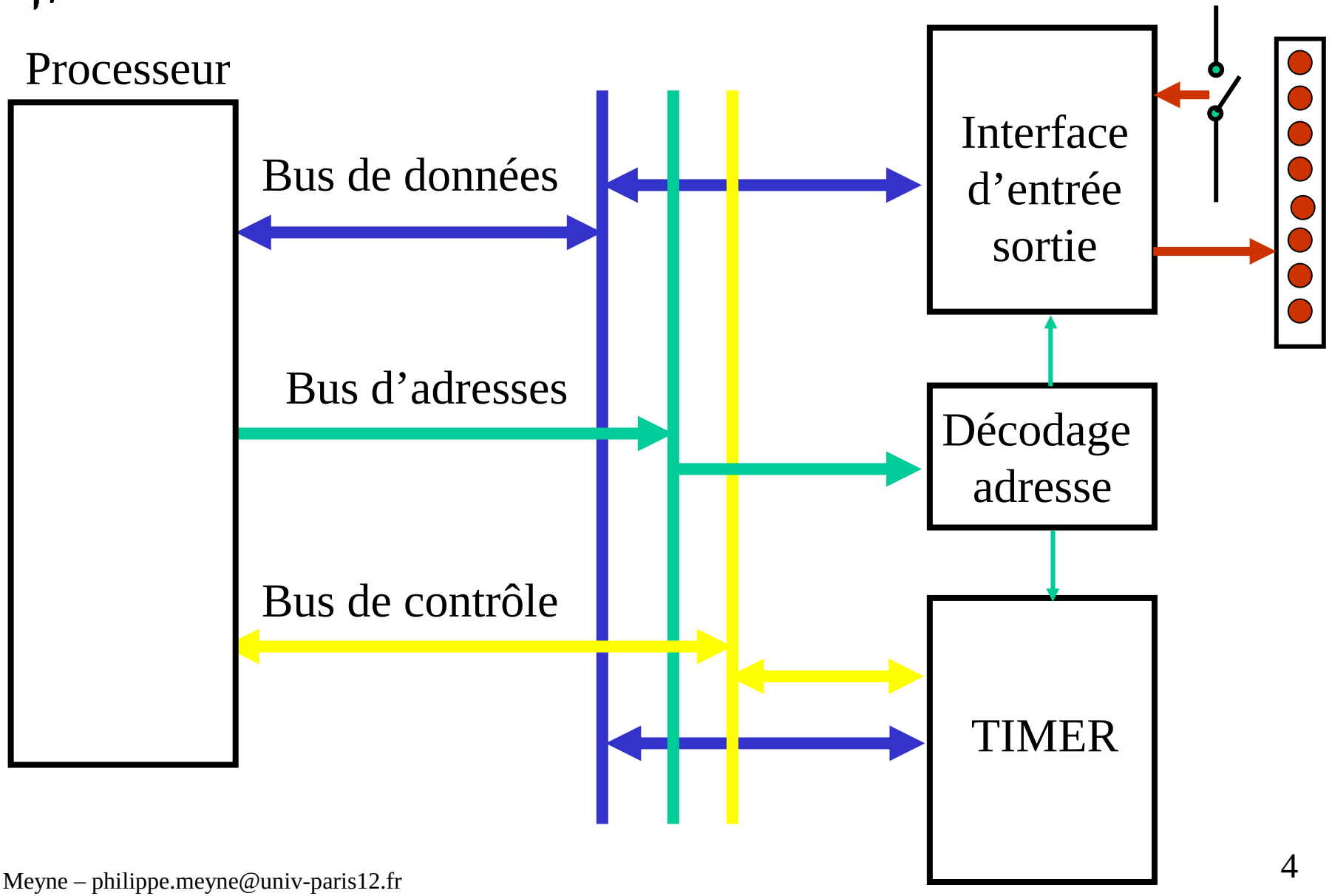
Description de l'application
Cahier des charges : dé électronique



Description de l'application
Analyse de l'application



Description de l'application
Entités physiques



Description de l'application
Pseudo code : fonction Lire_interrupteur

Lecture du mot placé à l'adresse de l'interrupteur

Masque pour isoler le bit qui donne l'état de l'interrupteur

```
Début | état <- @(interrupteur)
      | état <- état&$01
      | retour
Fin
```

Description de l'application
Pseudo code : fonction Afficher_led

Ecriture à l'adresse des led

Début	@(led) <- valeur
	retour
Fin	

Description de l'application
Pseudo code : fonction Générer_valeur

```
Début | tant que(1)
      |   faire | état <- Lire_interrupteur
      |         | état_1 <- état
      |         | tant que (état=état_1)
      |         | faire | valeur <- valeur + 1
      |         |      | si (valeur > 6) alors valeur<- 1
      |         |      | fin si
      |         |      | Afficher_led(valeur)
      |         |      | état <- Lire_interrupteur
      |         | fin tq
      |         |  etat_1 <- etat
      |         | tant que (etat=etat_1)
      |         | faire | état <- Lire_interrupteur
      |         | fin tq
      |   fin tq
Fin   |
```

Tant que l'interrupteur n 'est pas actionner les chiffres défilent

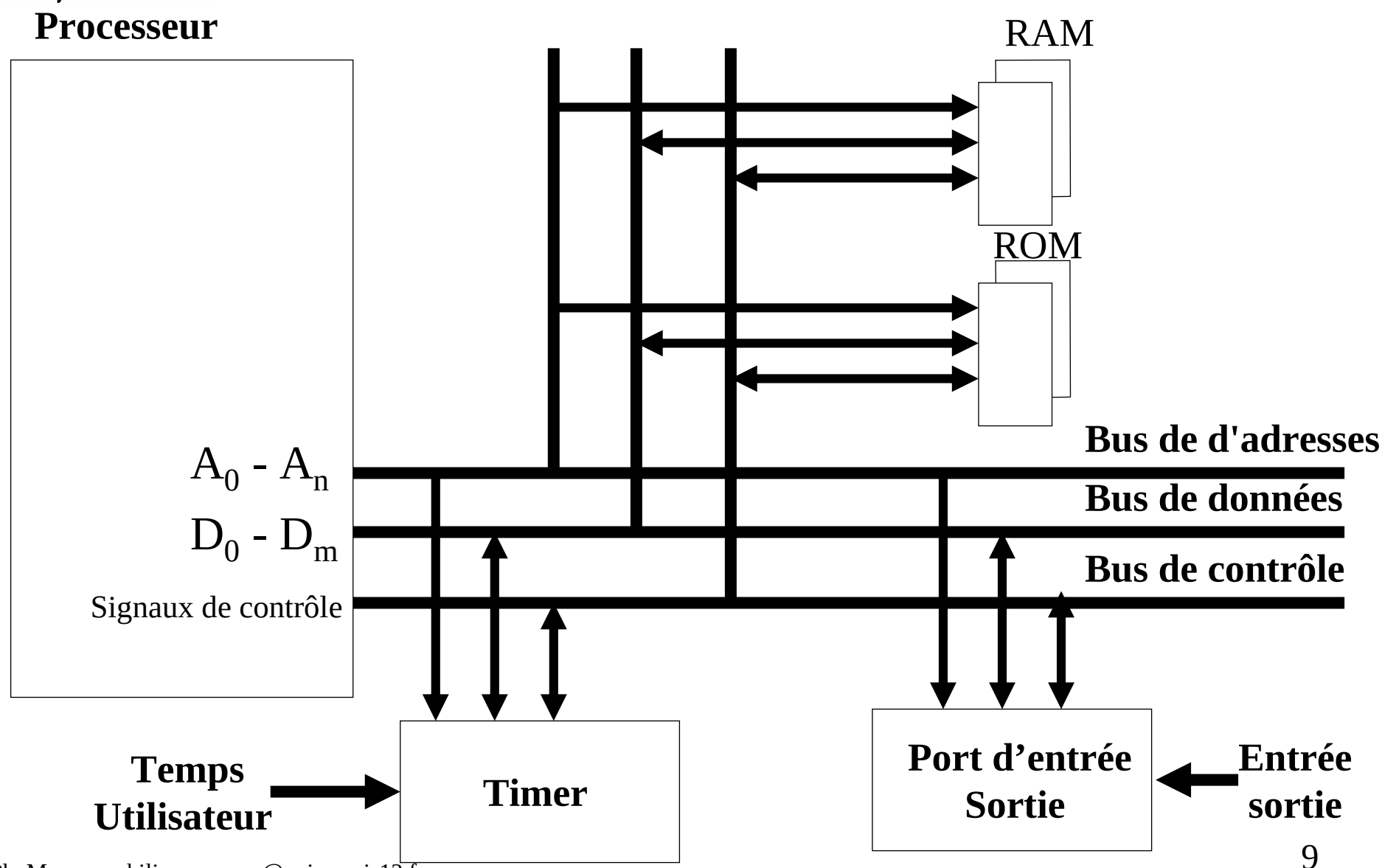
Rafraîchir la valeur de état

Tant que l'interrupteur n 'est pas actionner le programme attend

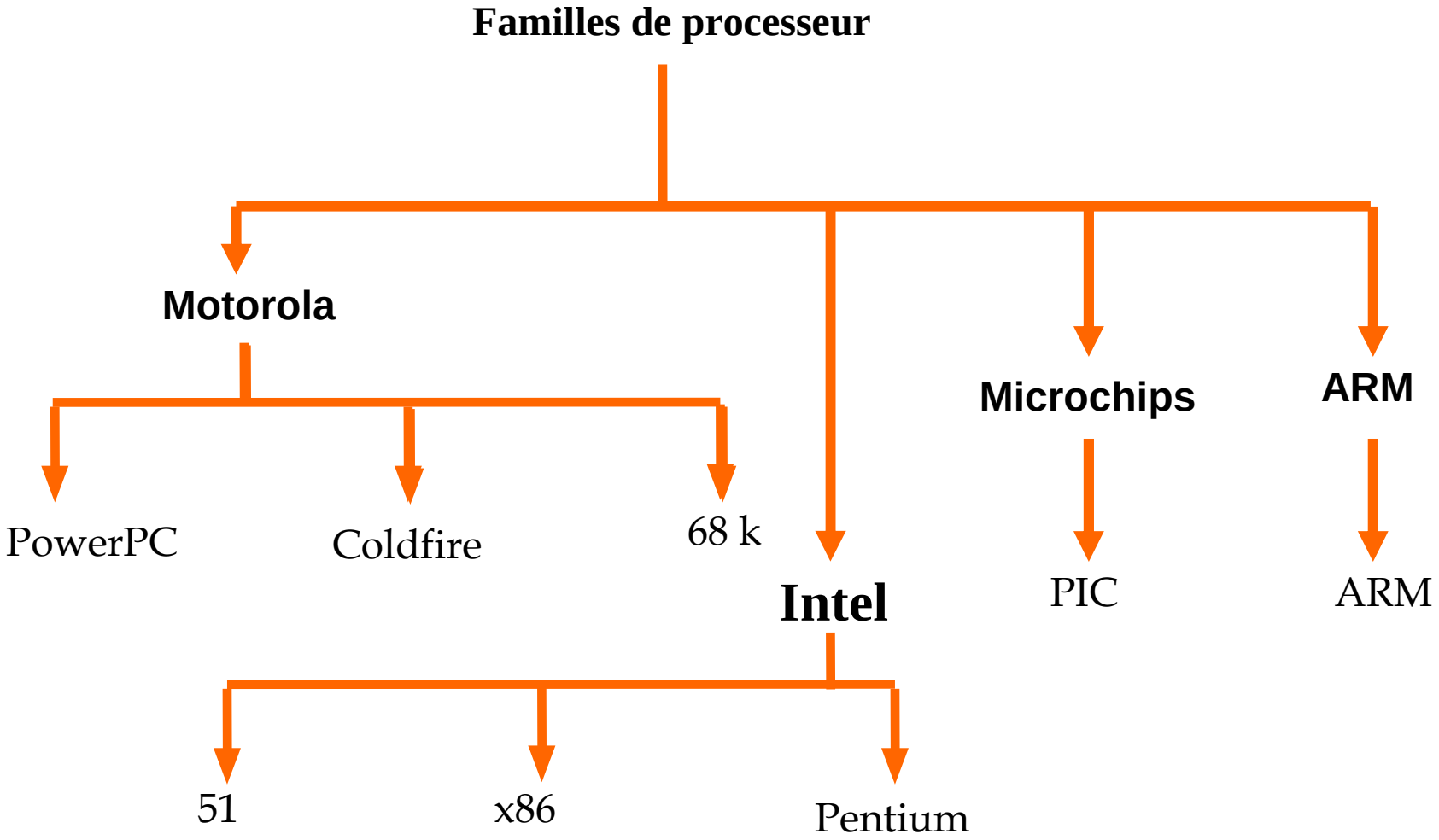
Description de l'application
Table des variables

Nom	Type	Allocation	Nature
Etat	entier		globale
Etat_1	entier		globale
Valeur	entier		globale
Interrupteur	adresse		locale
Led	adresse		locale

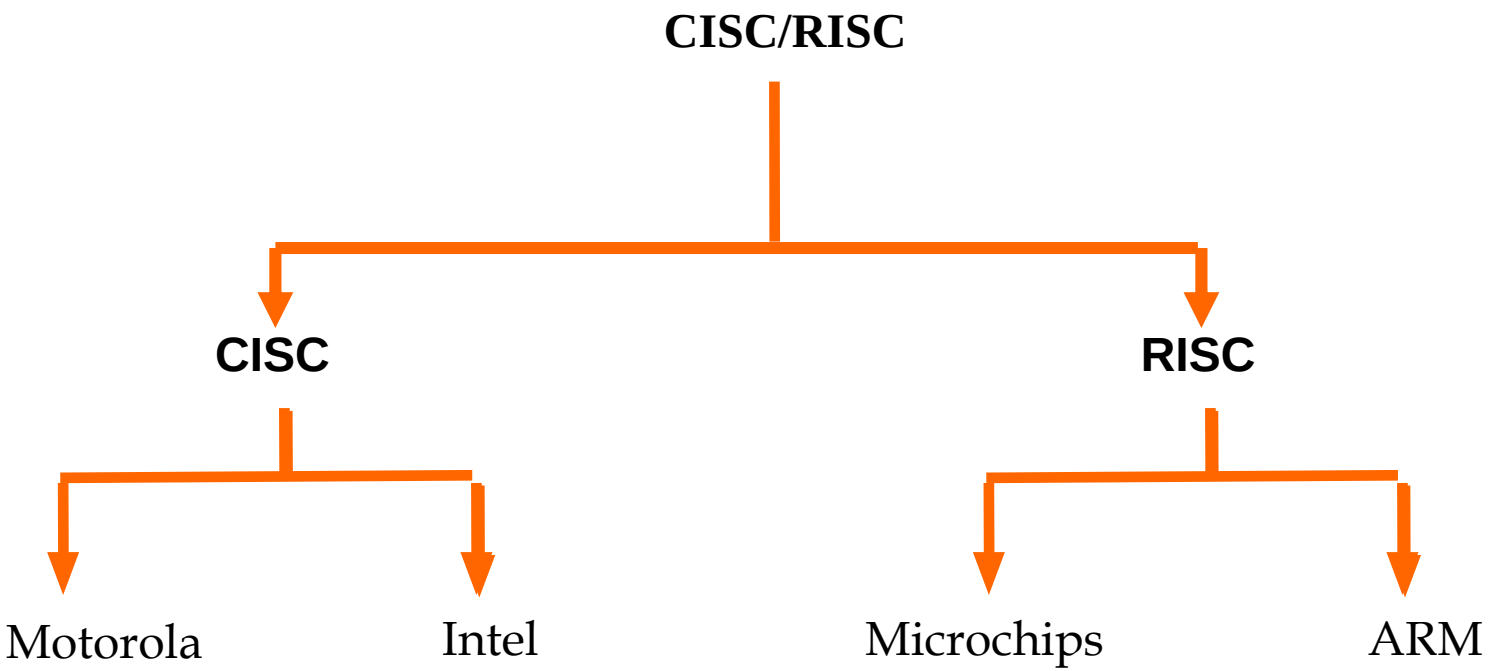
Description de l'application
Structures d'exécution



Structure des microcontrôleurs
Famille de microcontrôleur



Structure des microcontrôleurs
Famille CISC/RISC



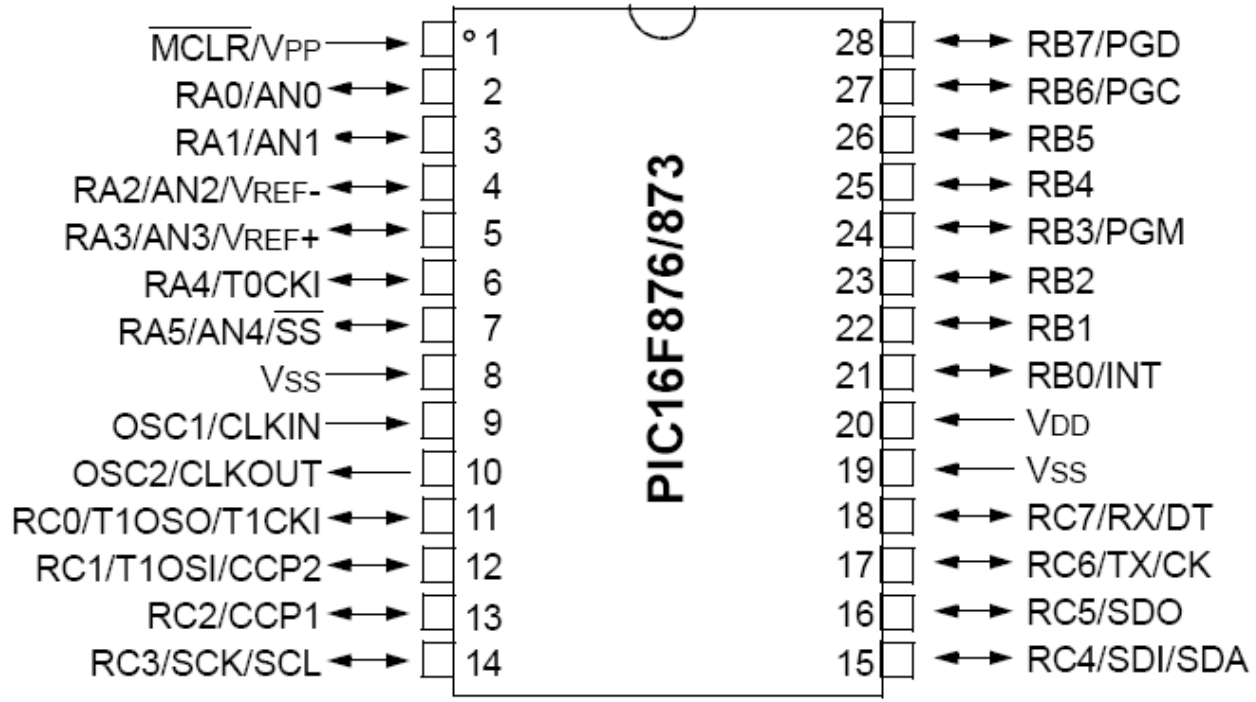
Structure des microcontrôleurs
Choix – Exemple 68HC11

Device	ROM (Bytes)	RAM (Bytes)	EPROM/ OTP (Bytes)	EEPROM (Bytes)	Timer[1]	I/O S.C.	I/O EXP	Serial	A/D	PWM	Operating Voltage (V)	Max Bus Frequency (MHz)	Comments
MC68HC11D0	—	192	—	—	16-Bit, 3/4IC, 4/5OC, RTI, pulse accumulator	—	16	SCI SPI	—	—	3.0, 5.0	3	64K external address bus, 3V 2MHz version (MC68L11D0)
MC68HC11D3	4K	192	—	—	16-Bit, 3/4IC, 4/5OC, RTI, pulse accumulator	32	16	SCI SPI	—	—	3.0, 5.0	3	64K external address bus, 3V 2MHz version (MC68L11D3)
MC68HC711D3	—	192	4K	—	16-Bit, 3/4IC, 4/5OC, RTI, pulse accumulator	32	16	SCI SPI	—	—	5	3	64K external address bus, 3MHz available in C temperature range only
MC68HC11E0	—	512	—	—	16-Bit, 3/4IC, 4/5OC, RTI, pulse accumulator	—	22	SCI SPI	8-CH 8-Bit	—	3.0, 5.0	3	3V 2MHz version (MC68L11E0)
MC68HC11E1	—	512	—	512	16-Bit, 3/4IC, 4/5OC, RTI, pulse accumulator	—	22	SCI SPI	8-CH 8-Bit	—	3.0, 5.0	3	3V 2MHz version (MC68L11E1)
MC68HC711KS2	—	1K	32K	640	16-Bit, 3/4IC, 4/5OC, RTI, pulse accumulator	51	26	SCI+ SPI	8-CH 8-Bit	4-CH 8-Bit or 2-CH 16-Bit	5	4	4MHz non-mux bus, slow mode feature, security option available
MC68HC11KW1	—	768	—	640	16-Bit, 3/4IC, 4/5OC, RTI, pulse accumulator	—	55	SCI+ SPI	10-CH 10-Bit	4-CH 8-Bit or 2-CH 16-Bit	5	4	4MHz non-mux bus, 2 extra timers, 4 chip selects extended, memory map up to 1Mbyte
MC68HC11P2	32K	1K	—	640	16-Bit, 3/4IC, 4/5OC, RTI, pulse accumulator	62	37	Triple SCI SPI	8-CH 8-Bit	4-CH 8-Bit or 2-CH 16-Bit	5	4	64K external address bus, MI-bus interface, PLL clock circuitry

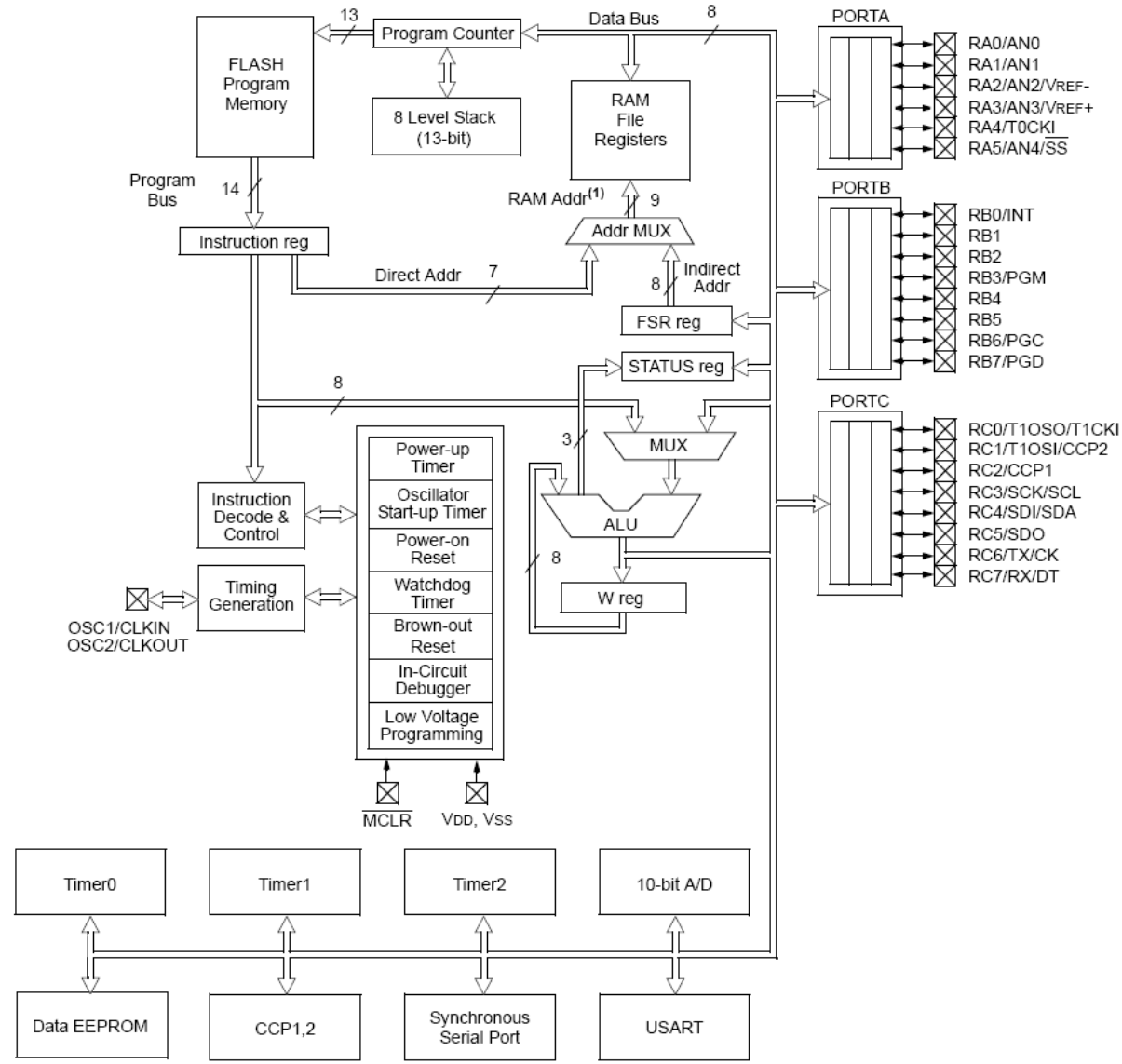
Microcontrôleur 16F876
Famille 16F87X

Key Features PICmicro™ Mid-Range Reference Manual (DS33023)	PIC16F873	PIC16F874	PIC16F876	PIC16F877
Operating Frequency	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz
RESETS (and Delays)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)
FLASH Program Memory (14-bit words)	4K	4K	8K	8K
Data Memory (bytes)	192	192	368	368
EEPROM Data Memory	128	128	256	256
Interrupts	13	14	13	14
I/O Ports	Ports A,B,C	Ports A,B,C,D,E	Ports A,B,C	Ports A,B,C,D,E
Timers	3	3	3	3
Capture/Compare/PWM Modules	2	2	2	2
Serial Communications	MSSP, USART	MSSP, USART	MSSP, USART	MSSP, USART
Parallel Communications	—	PSP	—	PSP
10-bit Analog-to-Digital Module	5 input channels	8 input channels	5 input channels	8 input channels
Instruction Set	35 instructions	35 instructions	35 instructions	35 instructions

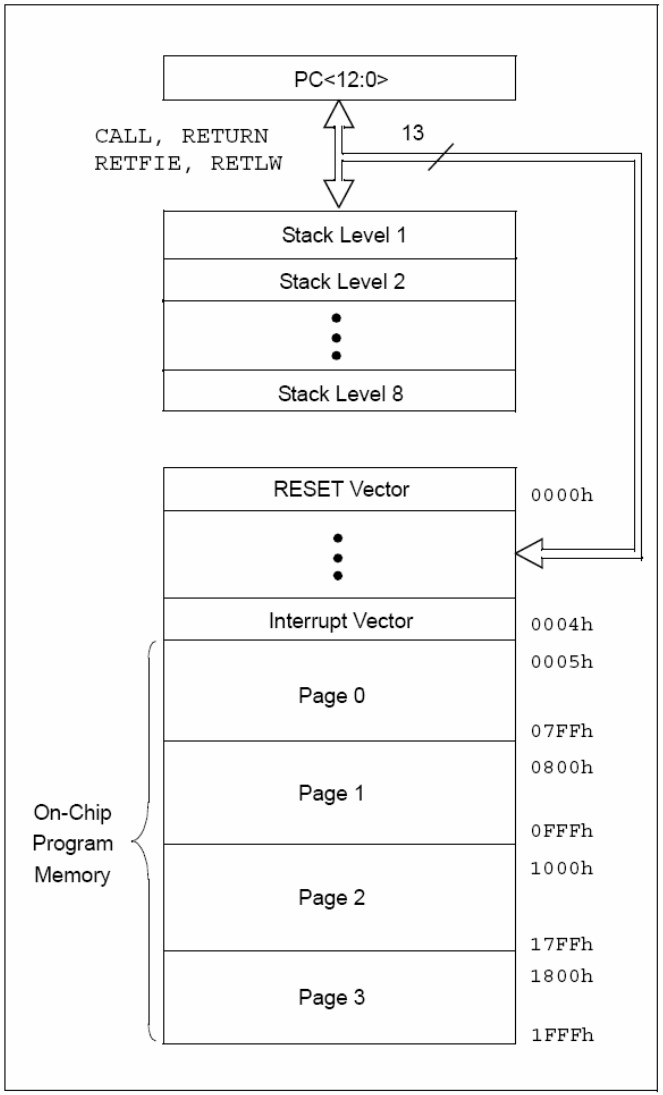
Microcontrôleur 16F876
Brochage



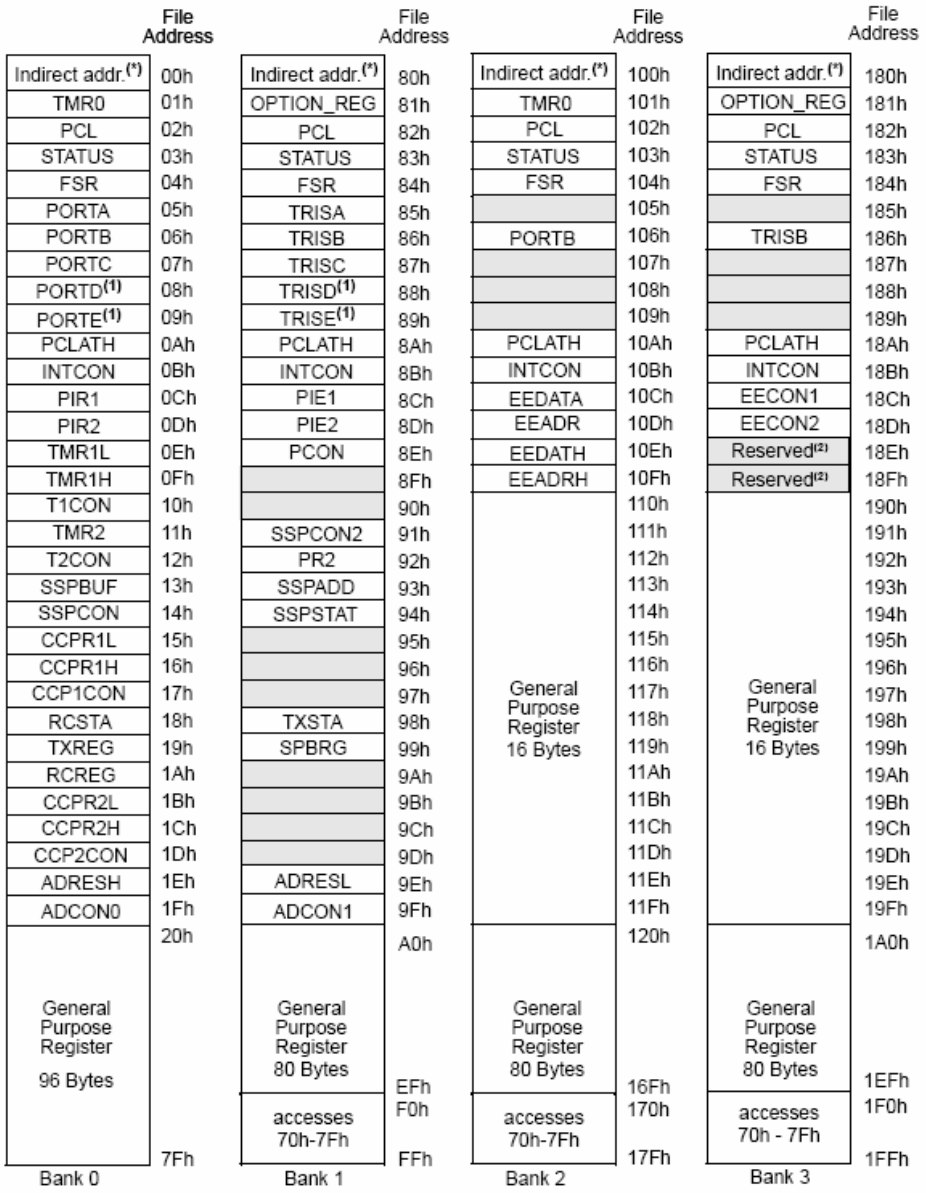
Microcontrôleur 16F876
Structure interne



Microcontrôleur 16F876
Carte mémoire



Microcontrôleur 16F876
Mémoire données



Microcontrôleur 16F876
Contenu des registres

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Value on: POR, BOR	Details on page:
Bank 0											
00h(3)	INDF	Addressing this location uses contents of FSR to address data memory (not a physical register)								xxxx xxxx	27
01h	TMR0	Timer0 Module Register								xxxx xxxx	47
02h(3)	PCL	Program Counter (PC) Least Significant Byte								0000 0000	26
03h(3)	STATUS	IRP	RP1	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	18
04h(3)	FSR	Indirect Data Memory Address Pointer								xxxx xxxx	27
05h	PORTA	—	—	PORTA Data Latch when written: PORTA pins when read						--0x 0000	29
06h	PORTB	PORTB Data Latch when written: PORTB pins when read								xxxx xxxx	31
07h	PORTC	PORTC Data Latch when written: PORTC pins when read								xxxx xxxx	33
08h(4)	PORTD	PORTD Data Latch when written: PORTD pins when read								xxxx xxxx	35
09h(4)	PORTE	—	—	—	—	—	RE2	RE1	RE0	---- -xxx	36
0Ah(1,3)	PCLATH	—	—	—	Write Buffer for the upper 5 bits of the Program Counter					---0 0000	26
0Bh(3)	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	20
0Ch	PIR1	PSPIF(3)	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	22
0Dh	PIR2	—	(5)	—	EEIF	BCLIF	—	—	CCP2IF	-r-0 0--0	24
0Eh	TMR1L	Holding register for the Least Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	52
0Fh	TMR1H	Holding register for the Most Significant Byte of the 16-bit TMR1 Register								xxxx xxxx	52
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	--00 0000	51
11h	TMR2	Timer2 Module Register								0000 0000	55
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	55
13h	SSPBUF	Synchronous Serial Port Receive Buffer/Transmit Register								xxxx xxxx	70, 73
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	67
15h	CCPR1L	Capture/Compare/PWM Register1 (LSB)								xxxx xxxx	57
16h	CCPR1H	Capture/Compare/PWM Register1 (MSB)								xxxx xxxx	57
17h	CCP1CON	—	—	CCP1X	CCP1Y	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	58
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	96
19h	TXREG	USART Transmit Data Register								0000 0000	99
1Ah	RCREG	USART Receive Data Register								0000 0000	101
1Bh	CCPR2L	Capture/Compare/PWM Register2 (LSB)								xxxx xxxx	57
1Ch	CCPR2H	Capture/Compare/PWM Register2 (MSB)								xxxx xxxx	57
1Dh	CCP2CON	—	—	CCP2X	CCP2Y	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	58
1Eh	ADRESH	A/D Result Register High Byte								xxxx xxxx	116
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/$\overline{DONE}$	—	ADON	0000 00-0	111

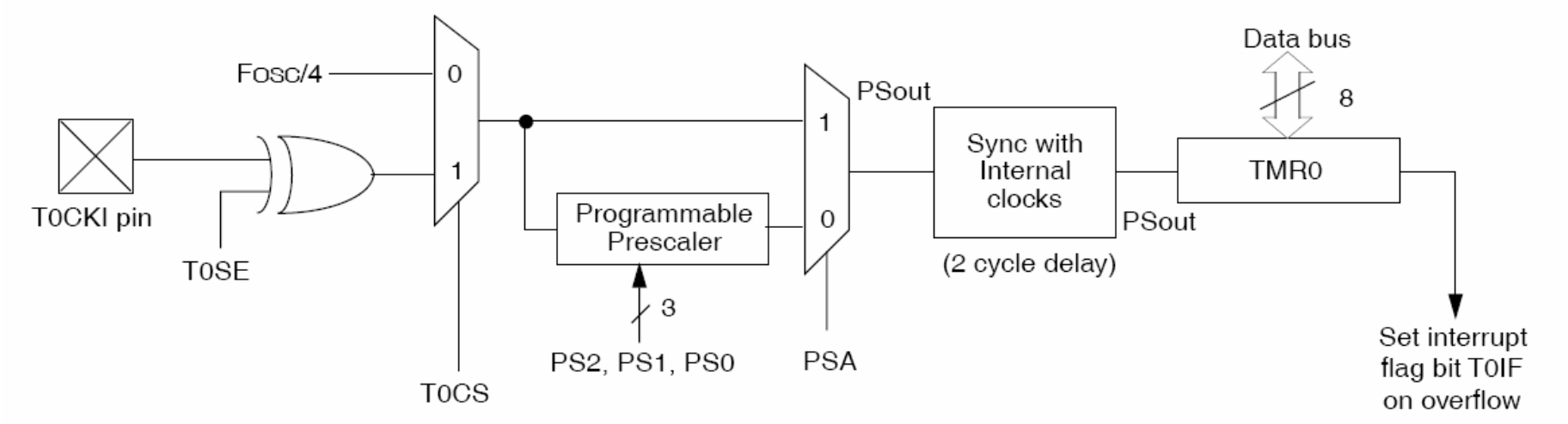
Legend: x = unknown, u = unchanged, q = value depends on condition, - = unimplemented, read as '0', r = reserved. Shaded locations are unimplemented, read as '0'.

Note 1: The upper byte of the program counter is not directly accessible. PCLATH is a holding register for the PC<12:8> whose contents are transferred to the upper byte of the program counter.
2: Bits PSPIE and PSPIF are reserved on PIC16F873/876 devices; always maintain these bits clear.
3: These registers can be addressed from any bank.
4: PORTD, PORTE, TRISD, and TRISE are not physically implemented on PIC16F873/876 devices; read as '0'.
5: PIR2<6> and PIE2<6> are reserved on these devices; always maintain these bits clear.

Microcontrôleur 16F876
Instructions

Mnemonic, Operands		Description	Cycles	14-Bit Opcode MSb LSb	Status Affected
BYTE-ORIENTED FILE REGISTER OPERATIONS					
ADDWF	f, d	Add W and f	1	00 0111 dfff ffff	C,DC,Z
ANDWF	f, d	AND W with f	1	00 0101 dfff ffff	Z
CLRF	f	Clear f	1	00 0001 lfff ffff	Z
CLRW	-	Clear W	1	00 0001 0xxx xxxx	Z
COMF	f, d	Complement f	1	00 1001 dfff ffff	Z
DECF	f, d	Decrement f	1	00 0011 dfff ffff	Z
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00 1011 dfff ffff	
INCF	f, d	Increment f	1	00 1010 dfff ffff	Z
INCFSZ	f, d	Increment f, Skip if 0	1(2)	00 1111 dfff ffff	
IORWF	f, d	Inclusive OR W with f	1	00 0100 dfff ffff	Z
MOVF	f, d	Move f	1	00 1000 dfff ffff	Z
MOVWF	f	Move W to f	1	00 0000 lfff ffff	
NOP	-	No Operation	1	00 0000 0xx0 0000	
RLF	f, d	Rotate Left f through Carry	1	00 1101 dfff ffff	C
RRF	f, d	Rotate Right f through Carry	1	00 1100 dfff ffff	C
SUBWF	f, d	Subtract W from f	1	00 0010 dfff ffff	C,DC,Z
SWAPF	f, d	Swap nibbles in f	1	00 1110 dfff ffff	
XORWF	f, d	Exclusive OR W with f	1	00 0110 dfff ffff	Z
BIT-ORIENTED FILE REGISTER OPERATIONS					
BCF	f, b	Bit Clear f	1	01 00bb bfff ffff	
BSF	f, b	Bit Set f	1	01 01bb bfff ffff	
BTFSC	f, b	Bit Test f, Skip if Clear	1 (2)	01 10bb bfff ffff	
BTFSS	f, b	Bit Test f, Skip if Set	1 (2)	01 11bb bfff ffff	
LITERAL AND CONTROL OPERATIONS					
ADDLW	k	Add literal and W	1	11 111x kkkk kkkk	C,DC,Z
ANDLW	k	AND literal with W	1	11 1001 kkkk kkkk	Z
CALL	k	Call subroutine	2	10 0kkk kkkk kkkk	
CLRWDT	-	Clear Watchdog Timer	1	00 0000 0110 0100	$\overline{TO}$,$\overline{PD}$
GOTO	k	Go to address	2	10 1kkk kkkk kkkk	
IORLW	k	Inclusive OR literal with W	1	11 1000 kkkk kkkk	Z
MOVLW	k	Move literal to W	1	11 00xx kkkk kkkk	
RETFIE	-	Return from interrupt	2	00 0000 0000 1001	
RETLW	k	Return with literal in W	2	11 01xx kkkk kkkk	
RETURN	-	Return from Subroutine	2	00 0000 0000 1000	
SLEEP	-	Go into standby mode	1	00 0000 0110 0011	$\overline{TO}$,$\overline{PD}$
SUBLW	k	Subtract W from literal	1	11 110x kkkk kkkk	C,DC,Z
XORLW	k	Exclusive OR literal with W	1	11 1010 kkkk kkkk	Z

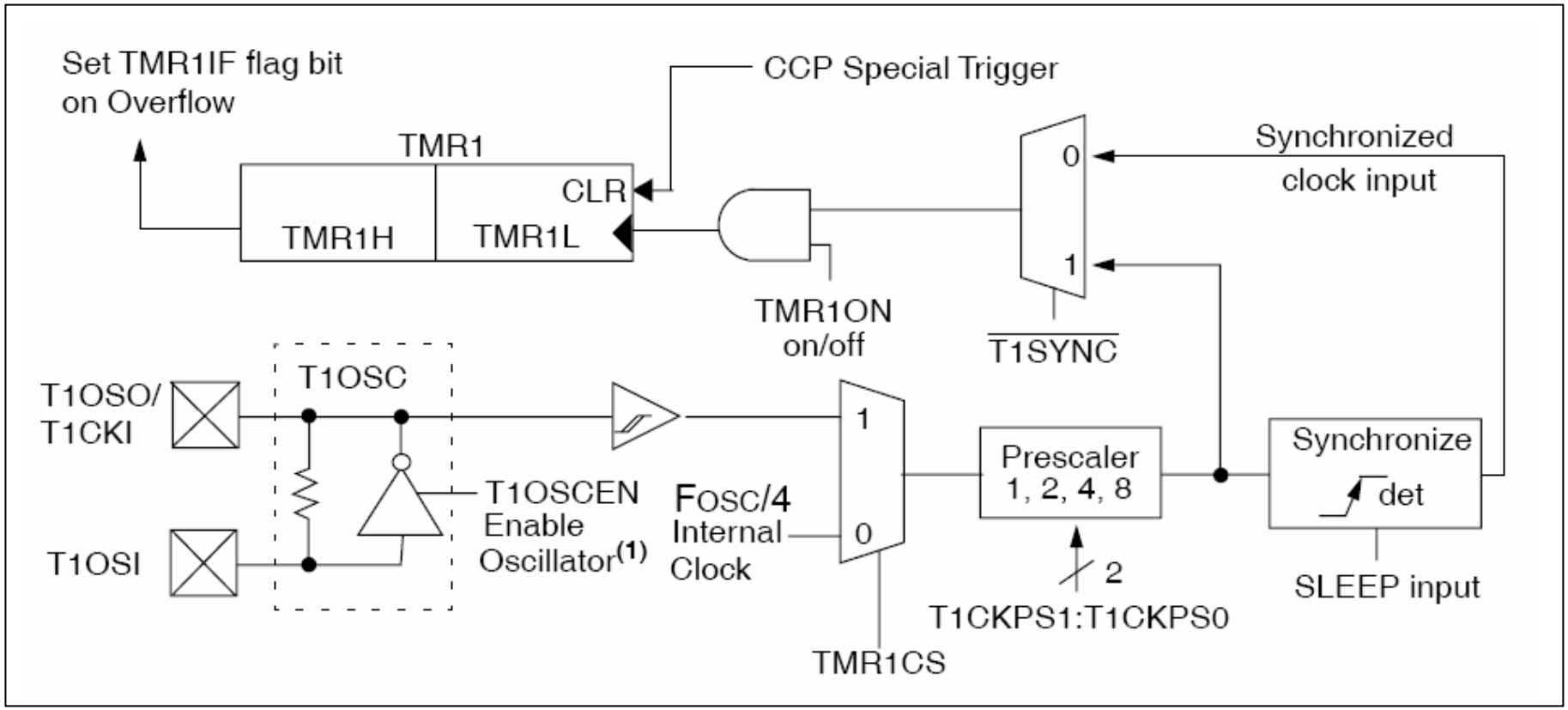
Microcontrôleur 16F876
TIMER 0 - Schéma



Microcontrôleur 16F876
TIMER 0 – Registre de programmation

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

Microcontrôleur 16F876
TIMER 1



U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	TMR1CS	TMR1ON
bit 7							bit 0

Ph. Meyne – philippe.meyne@univ-paris12.fr

Interruptions
Principes

IRQ

Composant périphérique

Requête d'interruption

```
// Tache principal

main()
{
        .
        .
        .
        .
        .
        exit(0)
}
```

Recherche de l'adresse de la fonction d'IT

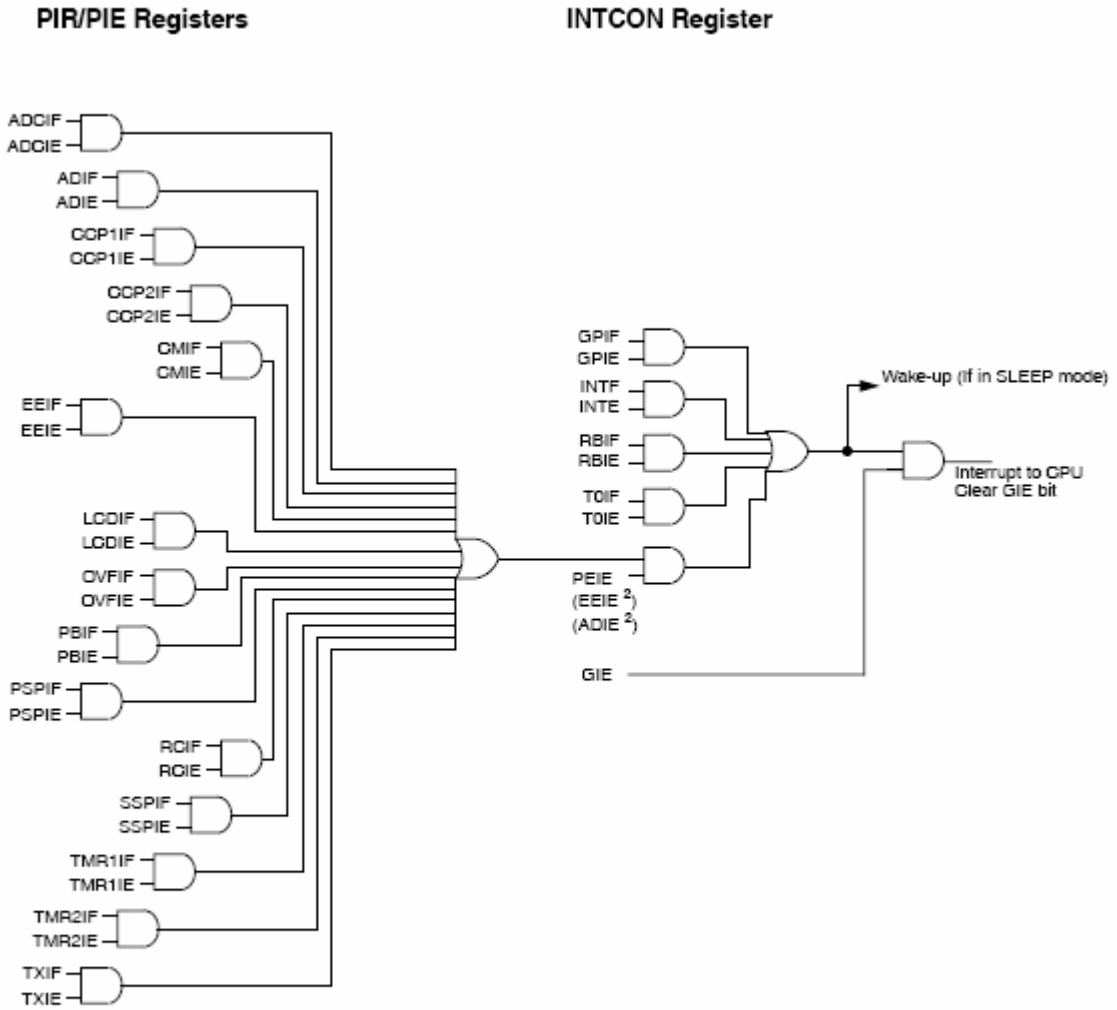
Branchement à la fonction d'interruption

*** fonction d'interruption**

*** sauvegarde du contexte**
*** traitement de l'interruption**

Retour d'IT *** retour au programme principal**

Interruptions
Cas du PIC 18F876 – Source des IT



Interruptions
Cas du PIC 18F876 – Registre INTCON

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
GIE	PEIE (3)	T0IE	INTE (2)	RBIE (1, 2)	T0IF	INTF (2)	RBIF (1, 2)
bit 7							bit 0

Interruptions
Cas du PIC 18F876 – Fonction d'IT typique

```
ISR_ADDR        CLRF STATUS ; Bank0
                BTFSC PIR1, TMR1IF ; Timer1 overflow interrupt?
                GOTO T1_INT ; YES
                BTFSC PIR1, ADIF ; NO, A/D interrupt?
                GOTO AD_INT ; YES, do A/D thing
                …
                BTFSC INTCON, RBIF ; NO, Change on PORTB interrupt?
                GOTO PORTB_INT ; YES, Do PortB Change thing
                …
PORTB_INT ; Routine when PortB has a change
…

RETFIE ; Return and enable interrupts
```

Interruptions
Cas du PIC 18F876 – Exemple IT externe

```
            LIST    p=16F876            ; PIC16F876 est le processeur cible
            #include "P16F84.INC"       ; fichier include de déclaration
            ORG   0
            goto  debut             ; branchement au reset
            ORG 4
            goto F_IT               ;branchement si IT
; memoire code
            ORG   5
debut       ; interdiction des interruption
                        clrf INTCON                 ; interdiction de toutes les IT
            ; port B en sortie
                        ; selection du bank 1 pour acces au registre de config du port B
                        bcf STATUS,RP1              ; RP1 <- 0
                        bsf STATUS,RP0              ; RP0 <- 1
                        ; direction du port B
                        movlw 0x01                  ;RB0 en entrée, les autres en sortie
                        movwf TRISB                 ; bit du port B en sortie
```

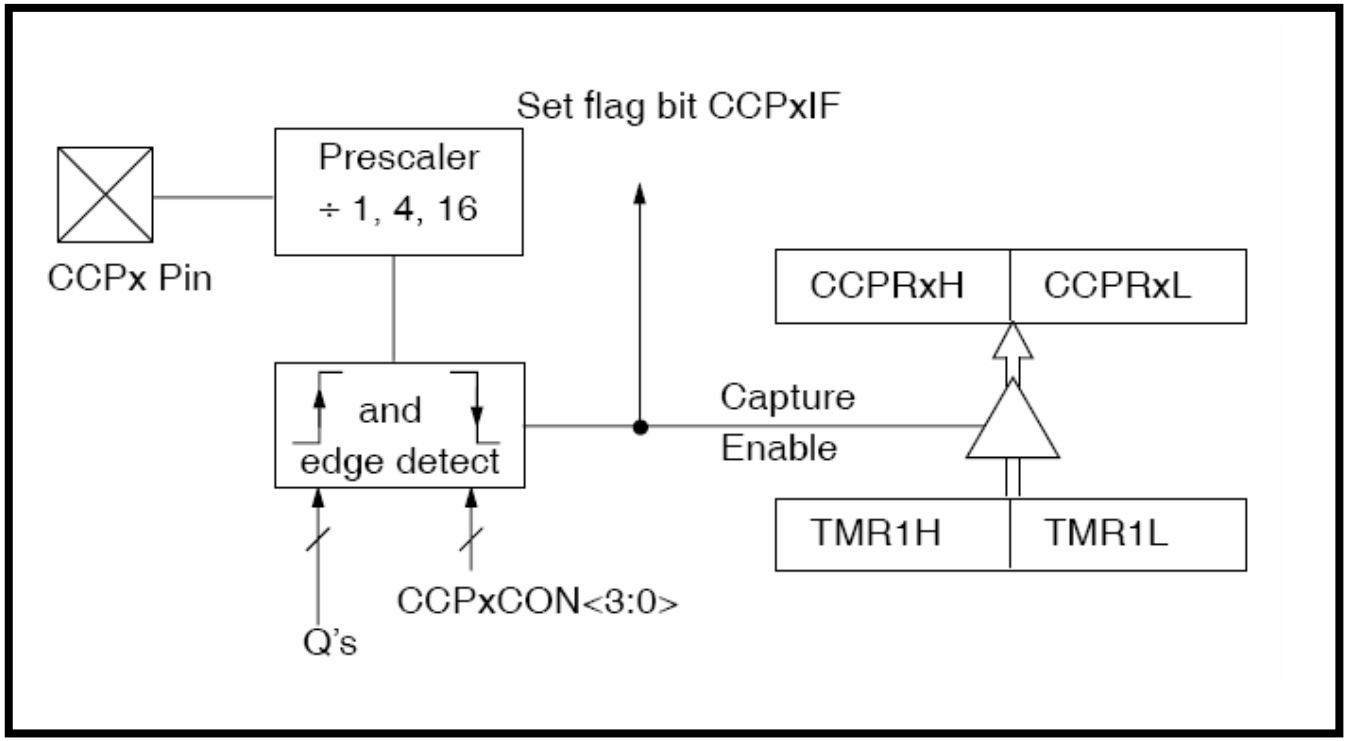
Interruptions
Cas du PIC 18F876

```
                    ; selection du bank 0 pour acces au registre E/S du port B
                    bcf STATUS,RP0          ; RP0 <- 0
            ; autorisation des interruptions
                    bsf INTCON,GIE          ; autorisation globale
                    bsf INTCON,INTE         ; interruption entree d'IT
                    bcf INTCON,INTF         ; pas d'IT externe produite

boucle              goto boucle             ; aller a boucle

; fonction d'interruption
F_IT                bcf STATUS,RP1          ; acces bank 0
                    bcf STATUS,RP0          ; acces bank 0
                    bsf PORTB,2             ; bit 1 a 0
                    bcf INTCON,INTF         ; remise a zero pour IT suivante
                    RETFIE
                    END
```

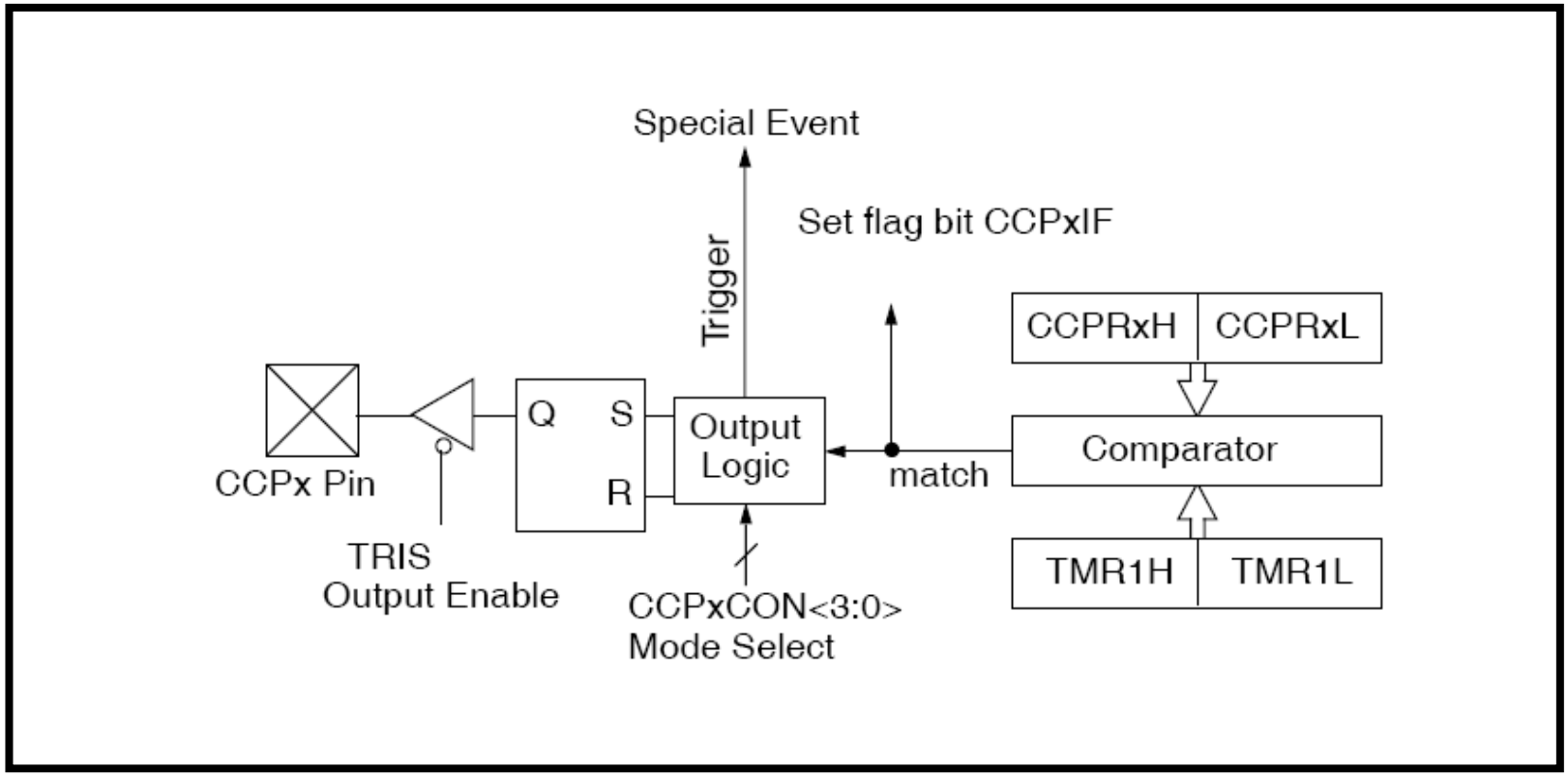
Périphériques
Compare Capture PWM – Registre de configuration

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	CCPxX	CCPxY	CCPxM3	CCPxM2	CCPxM1	CCPxM0
bit 7							bit 0

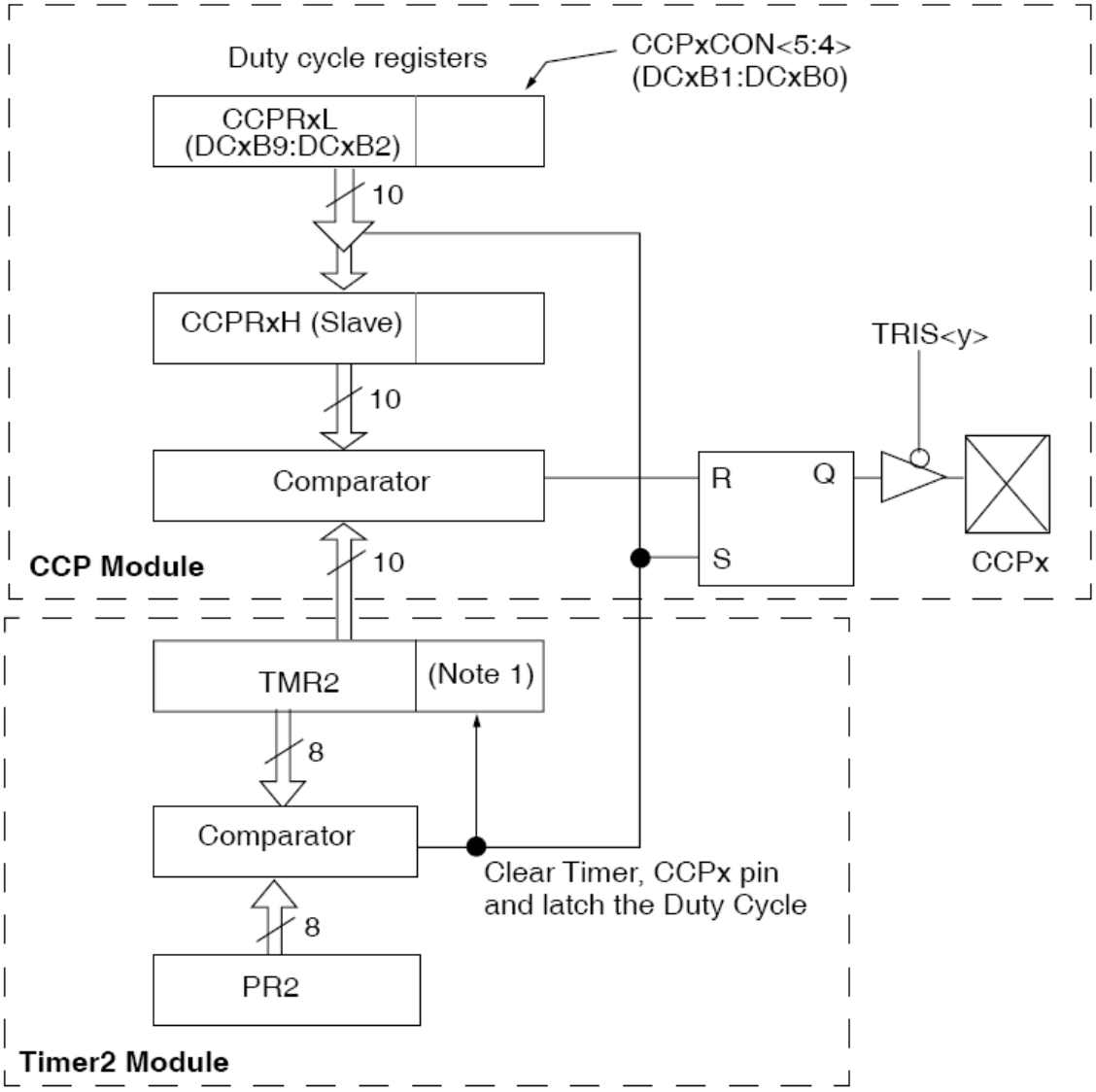
Périphériques
Compare Capture PWM – Mode capture



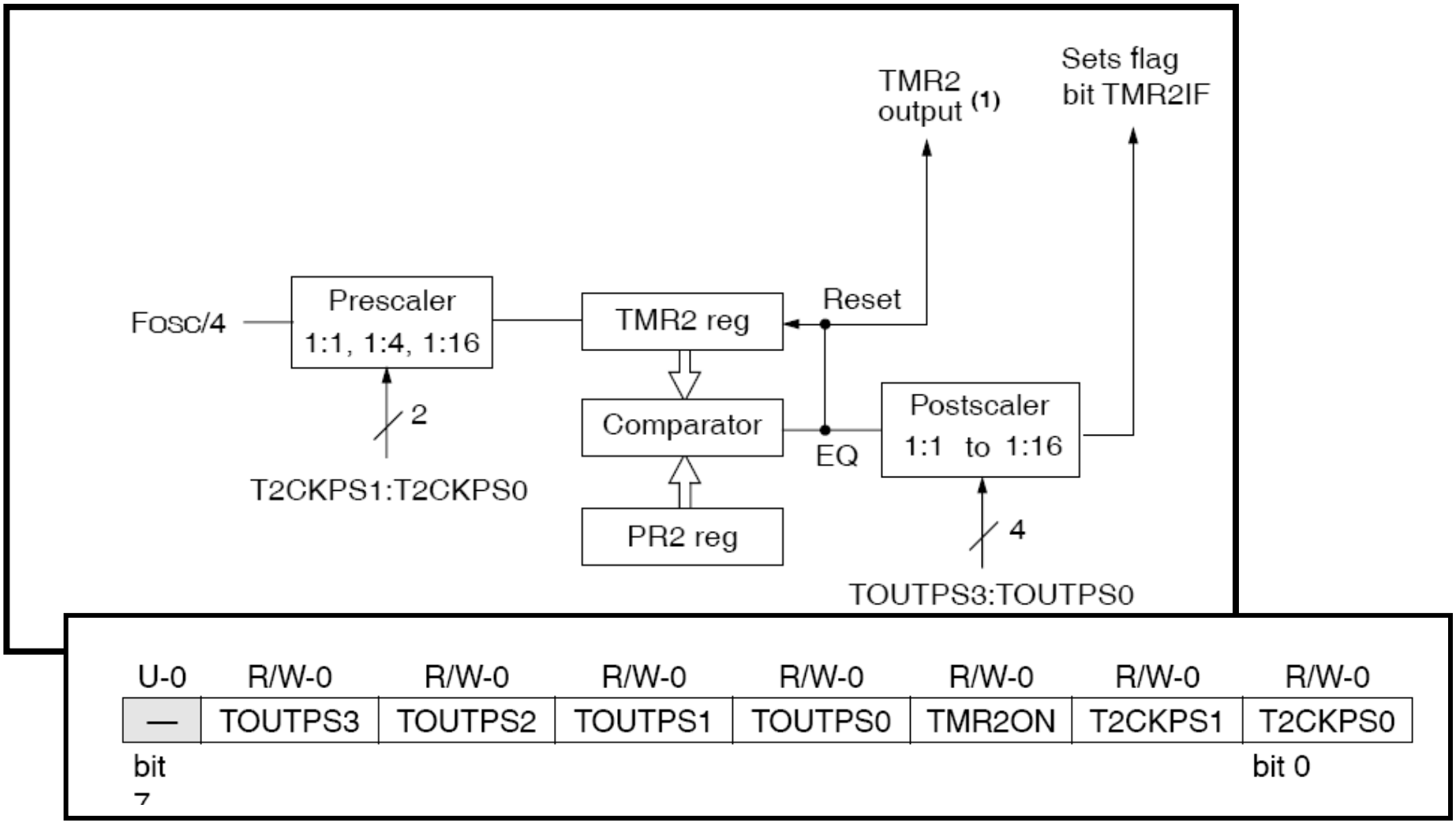
Périphériques
Compare Capture PWM – Mode compare



Périphériques
Compare Capture PWM – Mode PWM



Périphériques
Compare Capture PWM – Mode PWM
TIMER 2



U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0
bit 7							bit 0