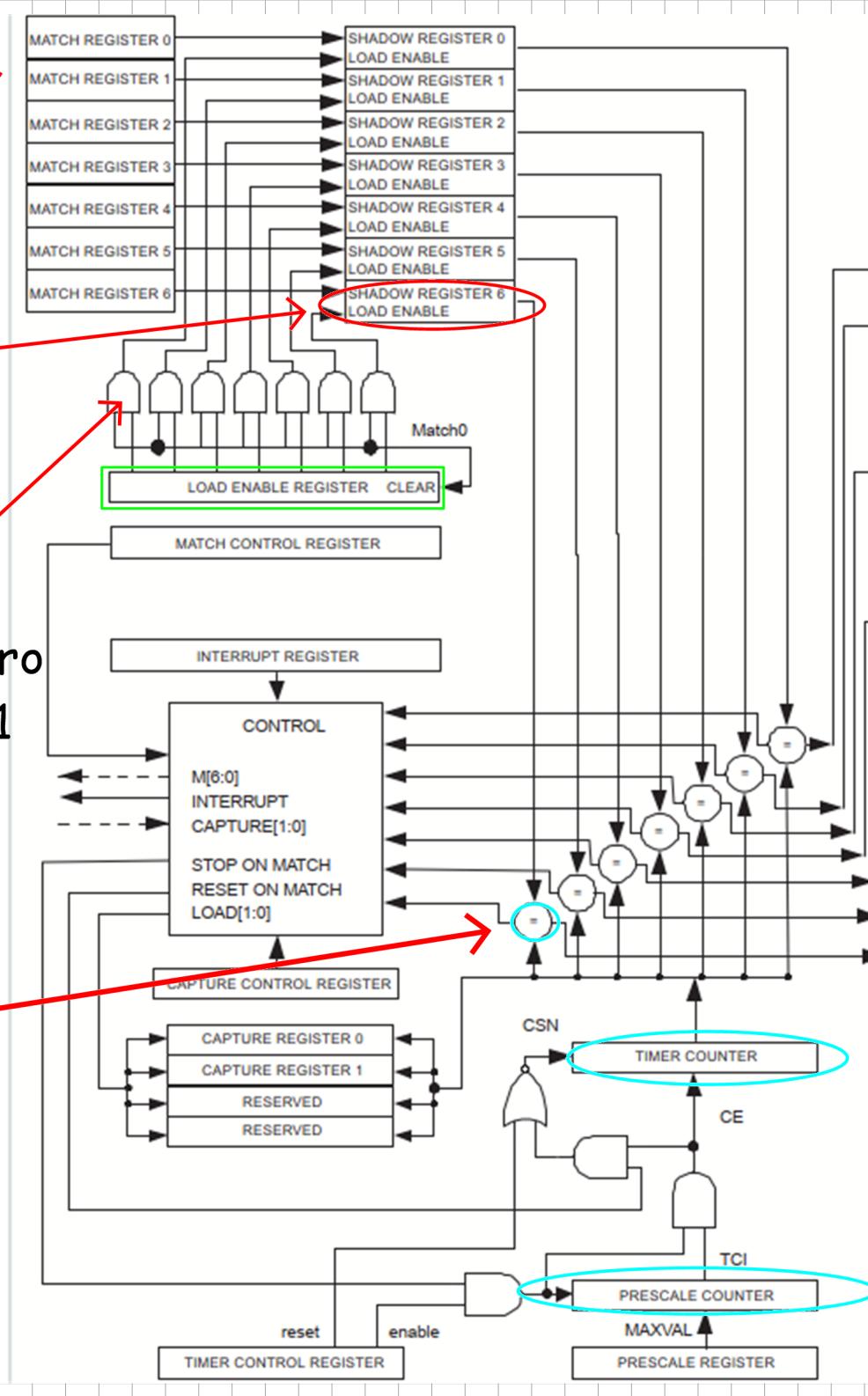
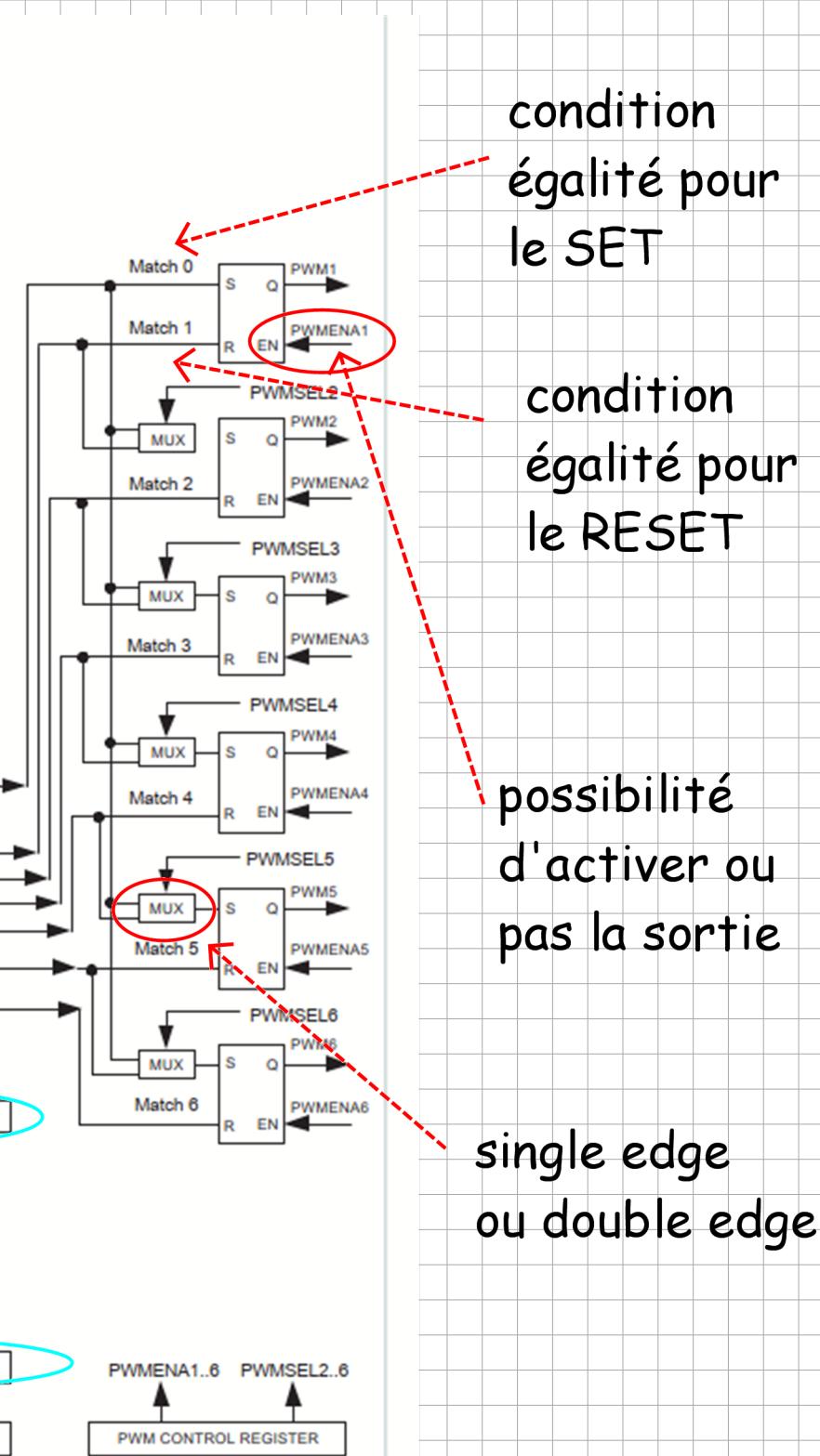


match modifié
 quand on veut
 recopie dans
 shadow :
 - instant du
 Match 0
 - bit de LER
 portant le numéro
 de match est à 1



testeur d'égalité



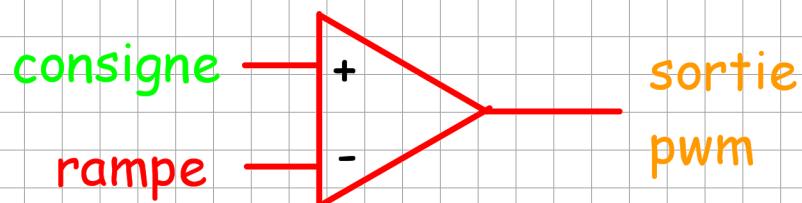
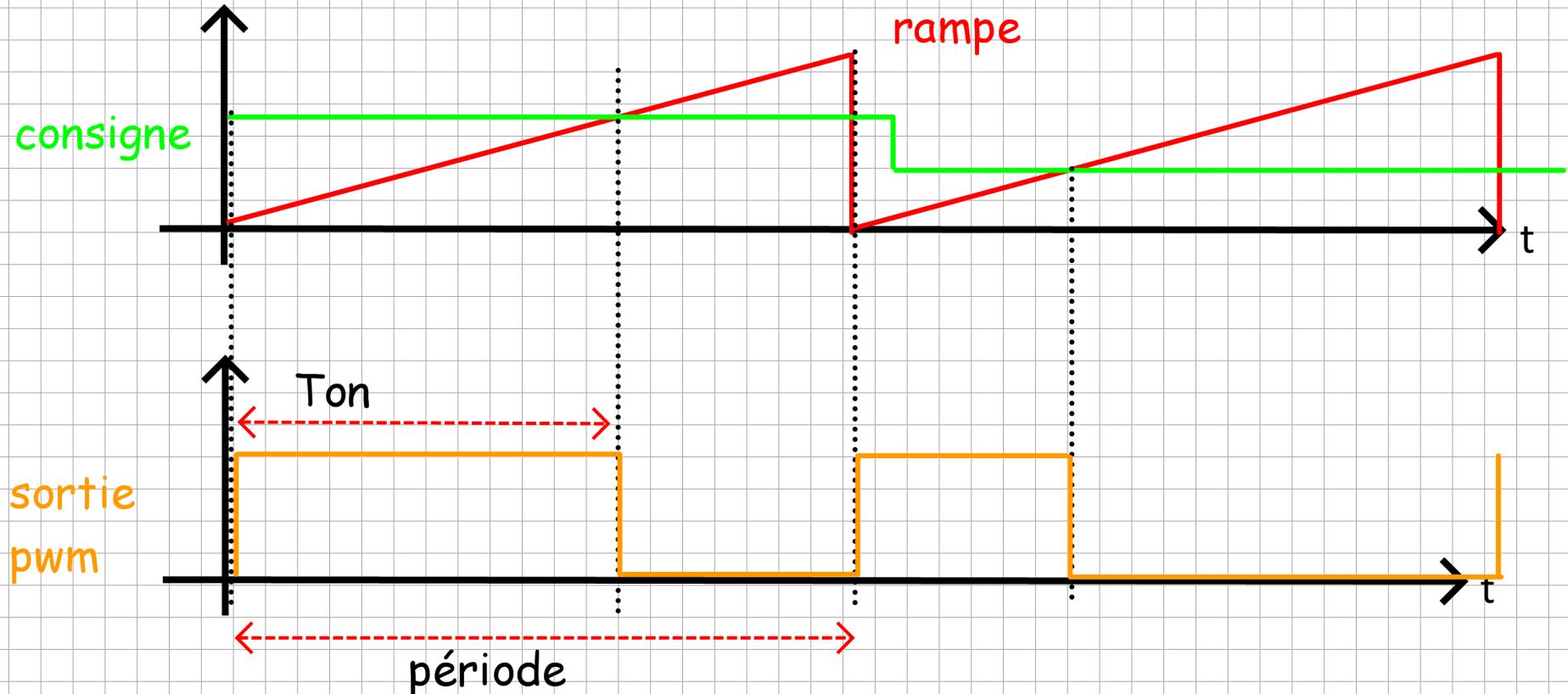
condition égalité pour le SET

condition égalité pour le RESET

possibilité d'activer ou pas la sortie

single edge ou double edge

rapport cyclique = $T_{on} / \text{Période}$



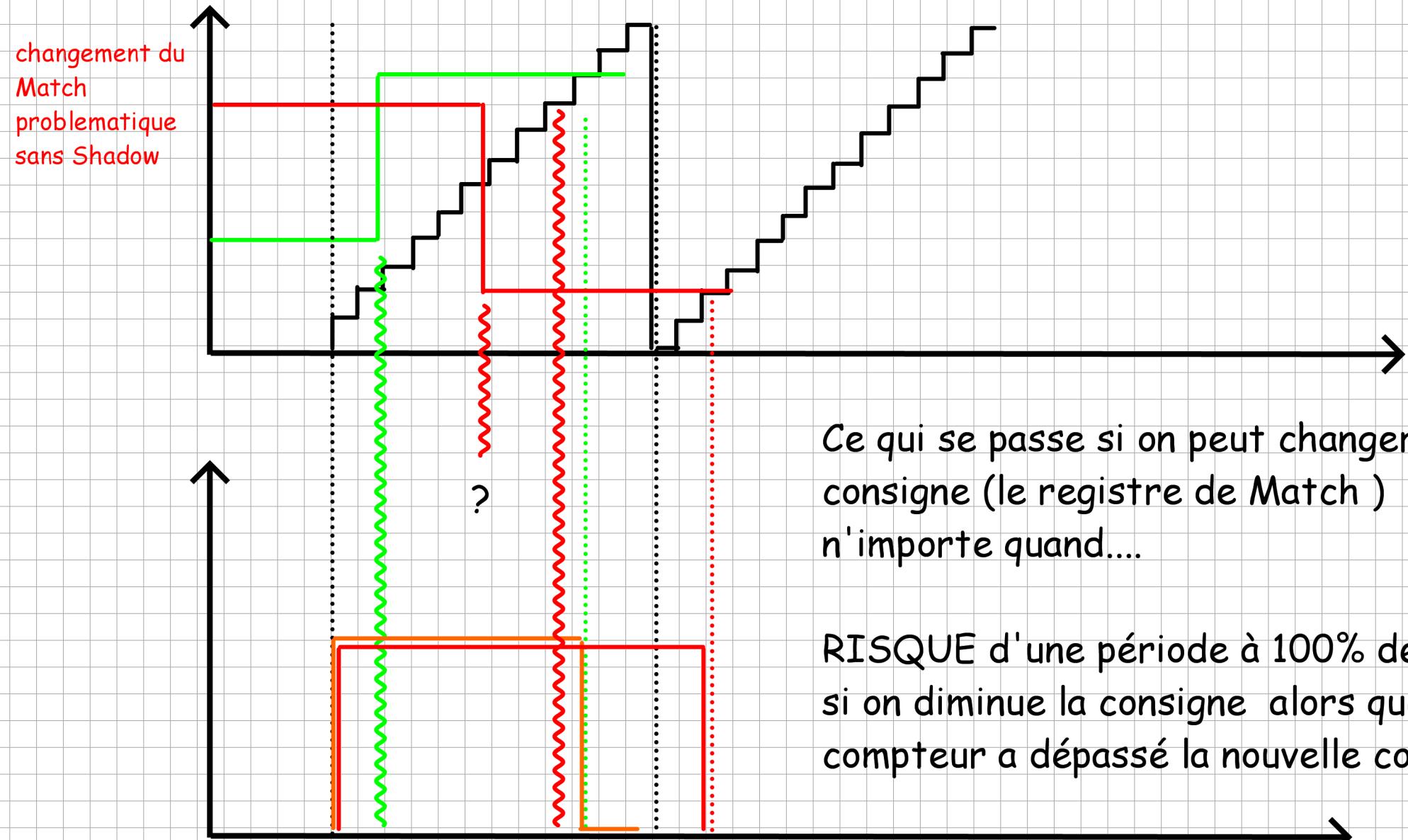
En numérique

rampe linéaire : escalier = compteur

consigne : registre

un comparateur numérique étant trop gourmand en portes logiques, on le remplace par :
deux testeurs d'égalité : un pour faire le front montant du PWM
un pour faire le front descendant du PWM

Il faut mémoriser l'état entre ces deux fronts : BASCULE set /Reset



POUR LUTTER contre ce problème : On autorise seulement un changement en début de période de PWM...

On va changer le match quand on veut, mais sa prise en compte sera en début de cycle uniquement --> utilise un registre SHADOW

Ce qui se passe si on peut changer la
consigne (le registre de Match)
n'importe quand....

Différence avec un Timer : configurer en plus

Configurer les PINSEL pour etre en sortie PWM : pour le PORT2 P2.0 à P2.15 --> PINSEL4

Autoriser les sorties PWM :

dans PCR bits 9 10 ...

PCR	PWM Control Register. Enables PWM outputs and selects PWM channel types as either single edge or double edge controlled.	R/W	0	PWM1PCR - 0xE001 804C
9	PWMENA1 1 The PWM1 output enabled. 0 The PWM1 output disabled.		0	
10	PWMENA2 1 The PWM2 output enabled. 0 The PWM2 output disabled.		0	

On pensera à se configurer en single edge

dans PCR bits 2 3 ...

2	PWMSEL2 1 Selects double edge controlled mode for the PWM2 output. 0 Selects single edge controlled mode for PWM2.	0
3	PWMSEL3 1 Selects double edge controlled mode for the PWM3 output. 0 Selects single edge controlled mode for PWM3.	0

Penser à activer les bits de LER pour la mise à jour des registres SHADOW

Exercice : On veut basculer P2.0 et P2.1 en PWM : fonctions PWM1.1 et PWM1.2

PINSEL4	Pin name	Function when 00	Function when 01	Function when 10	Function when 11	Reset value
1:0	P2.0	GPIO Port 2.0	PWM1.1	TXD1	TRACECLK	00
3:2	P2.1	GPIO Port 2.1	PWM1.2	RXD1	PIPESTAT0	00
5:4	P2.2	GPIO Port 2.2	PWM1.3	CTS1	PIPESTAT1	00
7:6	P2.3	GPIO Port 2.3	PWM1.4	DCD1	PIPESTAT2	00
9:8	P2.4	GPIO Port 2.4	PWM1.5	DSR1	TRACESYNC	00
11:10	P2.5	GPIO Port 2.5	PWM1.6	DTR1	TRACEPKT0	00
13:12	P2.6	GPIO Port 2.6	PCAP1.0	RI1	TRACEPKT1	00
15:14	P2.7	GPIO Port 2.7	RD2[2]	RTS1	TRACEPKT2	00
17:16	P2.8	GPIO Port 2.8	TD2[2]	TXD2	TRACEPKT3	00
19:18	P2.9	GPIO Port 2.9	USB_CONNECT[2]	RXD2	EXTIN0	00
21:20	P2.10	GPIO Port 2.10	EINT0	Reserved	Reserved	00
23:22	P2.11	GPIO Port 2.11	EINT1	MCIDAT1	I2STX_CLK	00
25:24	P2.12	GPIO Port 2.12	EINT2	MCIDAT2	I2STX_WS	00
27:26	P2.13	GPIO Port 2.13	EINT3	MCIDAT3	I2STX_SDA	00
29:28	P2.14	Reserved	Reserved	Reserved	Reserved	00
31:30	P2.15	Reserved	Reserved	Reserved	Reserved	00

Pinsel4 |= (0x01<<0) | (0x01 <<2);

config pour P2.0

config pour P2.1

on modifie deux pin : on utilise un "ou bit à bit" : |
pour effectuer les deux configurations pour les deux pattes

pour les deux pin P2.0 et P2.1 :
on veut la fonction 1 : 0x01

Pour P2.0 : bit 0 et 1 de pinsel 4
on utilisera un décalage de 0 : <<0

Pour P2.1 : bit 2 et 3 de pinsel 4
on utilisera un décalage de 2 : <<2