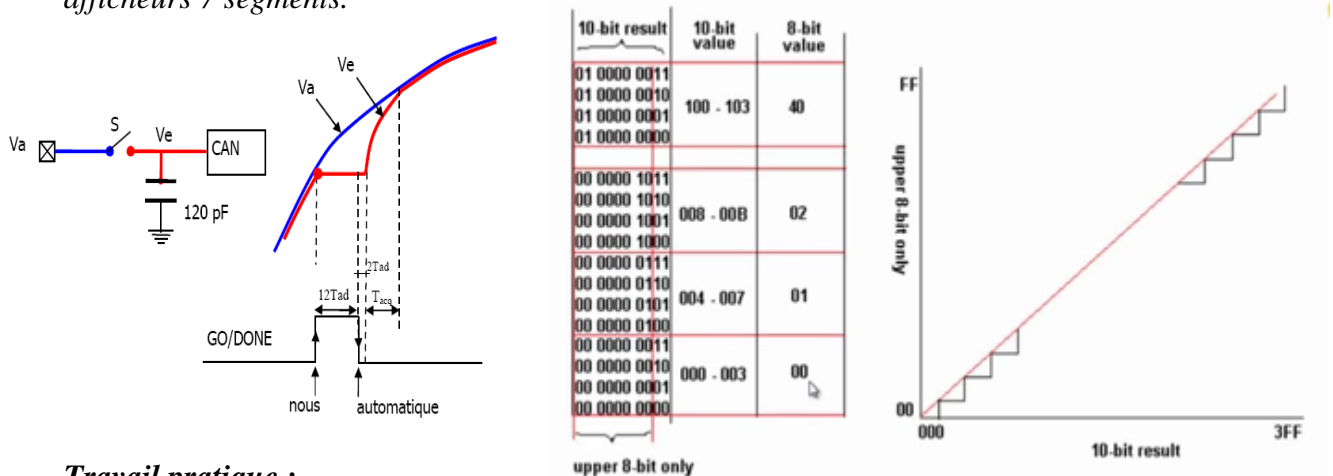


**Application: Convertisseur A/N à l'aide de l'assembleur PIC16F877**

Réaliser une conversion analogique / numérique d'une valeur comprise entre 0V et 5V réglable à l'aide du potentiomètre de votre platine carte de développement EasyPIC.

1. Dans un premier temps, afficher les 8 bits de poids fort sur les leds du PORTD.
2. Dans un deuxième temps, afficher les valeurs en décimal (de 0 à 255) sur les afficheurs 7 segments.

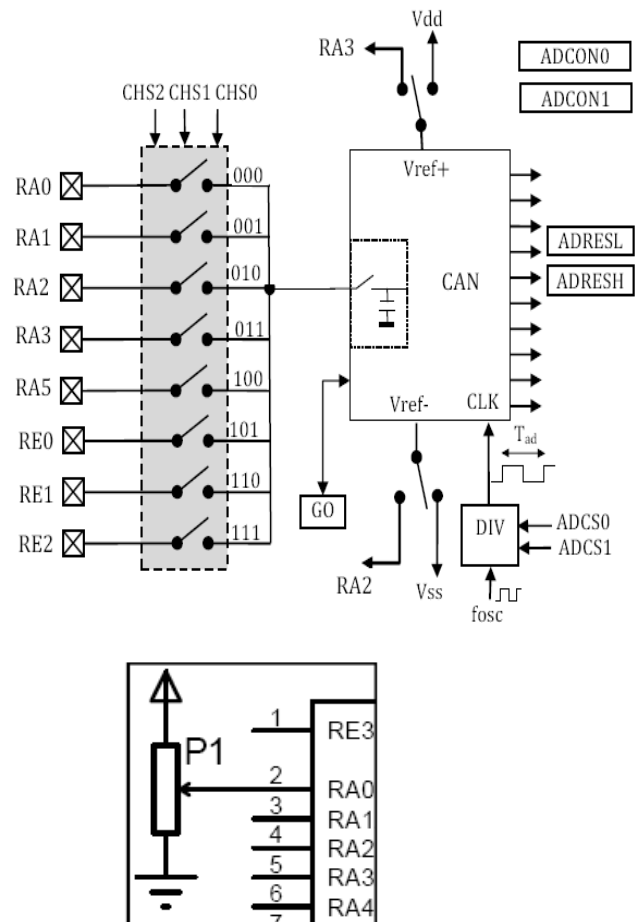


**Travail pratique :**

- 1) Observer la carte EasyPIC5 et localiser le potentiomètre P1
- 2) A l'aide du jumper J15, Brancher RA0 sur le potentiomètre P1
- 3) S'assurer sur SW1 que RA0 n'est pas connecté à une résistance de pullup/down
- 4) S'assurer sur SW6 que PORTA n'est pas connecté aux LEDs

**Déroulement d'une Conversion**

Le PIC dispose d'un échantillonneur bloqueur intégré constitué d'un interrupteur  $S$ , d'une capacité de maintien  $C=120\text{ pF}$  et d'un convertisseur Analogique numérique 10 bits. Pendant la conversion, la tension  $V_e$  à l'entrée du convertisseur A/N doit être maintenue constante. Au départ il faut commencer par faire l'acquisition du signal en fermant l'interrupteur  $S$ , ceci se fait à l'aide du registre  $ADCON0$ , soit au moment de la validation du module par le bit  $ADON$  soit après un changement de canal si  $ADON$  est déjà à positionné. Après la fin de l'acquisition, on peut démarrer une conversion en positionnant le bit  $GO_DONE$ , l'interrupteur  $S$  s'ouvre pour assurer le blocage de la tension. La conversion commence, elle est réalisée en  $12T_{AD}$ , à la fin, le bit  $GO_DONE$  repasse à 0, le drapeau  $ADIF$  passe à 1 et le résultat est chargé dans les registres  $ADRESL$  et  $ADRESH$ . Le module met  $2T_{AD}$  supplémentaires pour fermer l'interrupteur  $S$  ce qui démarre une nouvelle phase d'acquisition pendant laquelle la tension  $V_e$  rejoint la tension analogique d'entrée  $V_a$ . Le temps d'acquisition dépend de la constante de temps  $RC$ ,  $R$  étant la somme des résistances entre le module de conversion et la source de la tension analogique. Après la fin de l'acquisition, on peut démarrer une nouvelle conversion et ainsi de suite.



**Programmation**

- 1) Configurer les E/S en Analogique/Numérique/Référence (ADCON1)
- 3) Configurer les entrées analogiques en entrées (TRISA, TRISE)
- 4) Définir l'horloge de conversion à l'aide du diviseur DIV dans ADCON0
- 5) Choisir le canal à convertir et valider le module (ADCON0)
- 6) Attendre le temps d'acquisition (12  $\mu$ s)
- 7) Lancer la conversion, GO = 1 (ADCON0)
- 8) Attendre fin de conversion, GO = 0 ou drapeau ADIF=1
- 9) Traiter le résultat
- 10) Si l'on désire prendre d'autres mesures, recommencer au point 7 en faisant attention aux Timings

	Div	20Mhz	5Mhz	4Mhz	2Mhz	1Mhz
00	2	0,1 $\mu$ s	0,4 $\mu$ s	0,5 $\mu$ s	1 $\mu$ s	2 $\mu$ s
01	8	0,4 $\mu$ s	1,6 $\mu$ s	2 $\mu$ s	4 $\mu$ s	8 $\mu$ s
10	32	1,6 $\mu$ s	6,4 $\mu$ s	8 $\mu$ s	16 $\mu$ s	32 $\mu$ s
11	RC	Non utilisable				$\approx$ 4 $\mu$ s

- 1- Exécuter le programme. Ensuite procédez par interruption
- 2- Ecrivez un programme qui affiche le résultat de la conversion sur 10 bits

Compte rendu : Interprétez le programme et ajouter au programme une partie du code qui affiche le résultat de la conversion sur 10 bits dans les afficheurs 7 segments. Tester le programme sur la carte easypic5

Déterminer le pas de la conversion dans chaque cas

```

;tp2_an.asm  conversion analogique numérique
processor    16F877A
include      <P16F877.I16>
__CONFIG    _XT_OSC & _WDT_OFF & _PWRTE_ON &
_CP_OFF & _BODEN_ON & _LVP_OFF & _CPD_OFF &
_WRT_ENABLE_ON & _DEBUG_OFF
ORG 0x000    ; commencer a 0
goto start
start nop
BANKSEL TRISD
clrf TRISD    ;PORTD en sortie
BANKSEL PORTD
clrf PORTD    ;RAZ du PORTD
movlw B'10000001' ;Fosc/32, A/D activé,
movwf ADCON0    ; échantillonner Channel 0
BANKSEL OPTION_REG
movlw B'10000111' ;TMR0 prescaler, 1:256
movwf OPTION_REG
movlw B'00001110' ;Left justify, 1 entree analog A0
movwf ADCON1 ;VDD et VSS references
;Main
banksel TMR0
movlw b'00000000'
movwf TMR0
bcf INTCON,TOIF
Loop btfs INTCON,TOIF ; attendre debordement
goto Loop
BANKSEL PORTD
bsf ADCON0,GO ;debut conversion A/D pour canal 0
Wait btfs PIR1,ADIF ;attendre fin de conversion
goto Wait
movf ADRESH,W ;écrire A/D résultat au PORTD
movwf PORTD    ; allumer LEDs
bcf PIR1,ADIF    ;mettre a zero drapeau de fin
goto Loop    ;recommencer Loop
END

```

Les registres pour configuration du CAN.

ADRESH	A2D Result Register - High Byte (registre resultat)							
ADRESL	A2D Result Register - Low Byte(registre resultat)							
ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	-	ADON
ADCON1	ADFM	-	-	-	PCFG3	PCFG2	PCFG1	PCFG0