

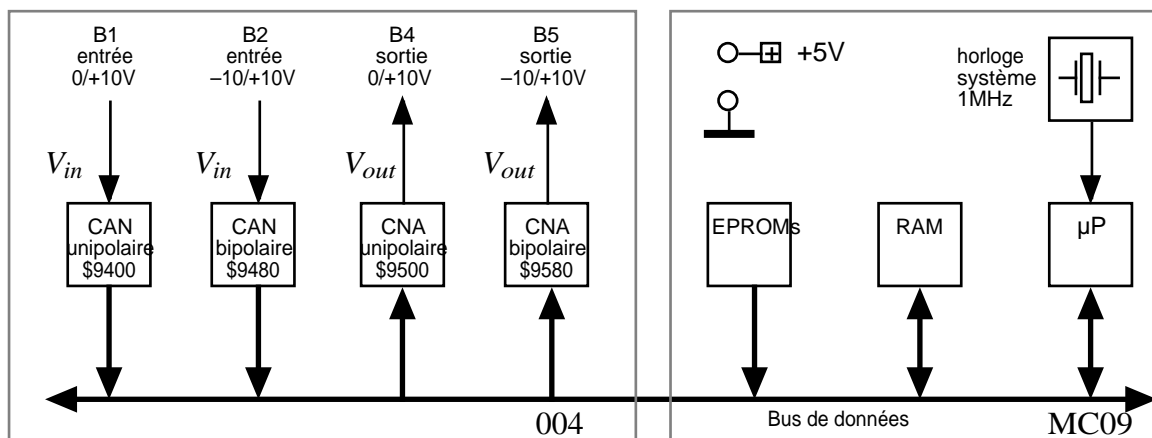
B15 - Initiation au traitement numérique du signal

But : réaliser par voie numérique quelques fonctions analogiques étudiées précédemment (voir partie A du programme). Matériel :

- μP 6809 sur carte MC09 (SERIElectronique), utilisé dans le TP B11/codage et traitement numérique.

- carte d'E/S analogique ref 004, incluant deux CAN 8 bits AD7574 (circuit déjà utilisé dans le TP B13/CAN) et deux CNA 8 bits AD7274.

Dans tout le TP, on pourra si nécessaire se reporter au listing de l'EPROM "TPTS2" pour consulter le détail des programmes machine.

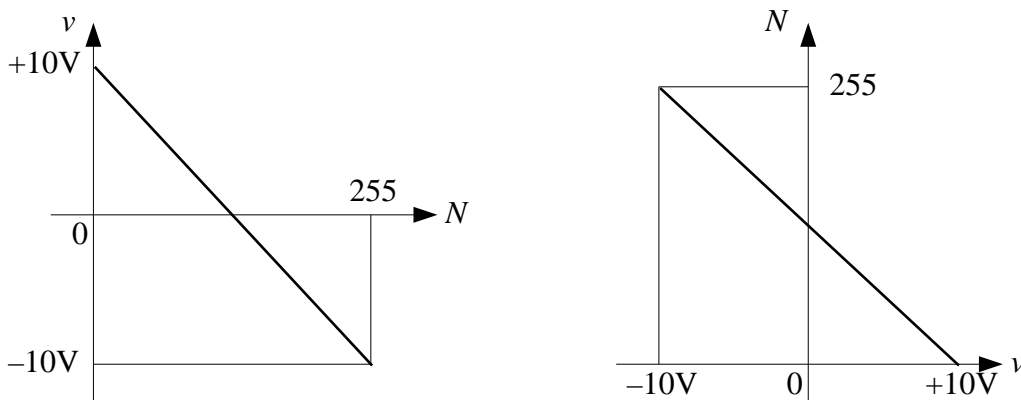


0- Codage numérique de l'information analogique

En mode bipolaire $[-10 / +10V]$, on utilise un code binaire décalé, format 8 bits, décalage $d = 128$, commun au CAN et au CNA.

L'octet le plus faible ($N = 0 = \$00$) est affecté à la tension d'entrée la plus grande ($V_{in} = +10V$), l'octet le plus grand ($N = 255 = \$FF$) à la tension la plus faible ($V_{in} = -10V$)

Les relations entre N et v sont : $v = -q.N + b$ et, réciproquement : $N = \lfloor c.v + d \rfloor$:



Calculer les coefficients q , b et c des droites $v(N)$ et $N(v)$. Préciser les unités.

Remarque : le coefficient q n'est autre que l'incrément ou résolution du CAN (cf TP B13).

I- Fonction : Générateur analogique de signaux (TP A25) → Générateur numérique**0) Principe : génération d'un signal continu**

\$AC00 * Programme : génère un signal continu
 \$0F02 * Entrée : N, octet hexadécimal à convertir
 * Sortie : B5, sortie analogique [-10V,+10V]

Algorithme :

Charger l'accumulateur A avec N, contenu dans la mémoire d'adresse \$0F02
 Envoyer (A) vers le CNA
 Fin

Écrire un nombre binaire (un octet codé en hexadécimal) à l'adresse \$0F02. Exécuter le programme à l'adresse \$AC00.

Quel est le nombre permettant d'obtenir en sortie +10V ? Puis -10V ? 0V ? +5V ? -5V ?

1) Valeurs $v_s(k)$ incluses dans le programme

\$AC0B * Programme : génère un signal rectangulaire
 \$0F00 * Entrée M1 : durée T1 du niveau 1
 \$0F01 * Entrée M2 : durée T2 du niveau 2
 \$0F02 * Entrée N1 : niveau 1
 \$0F03 * Entrée N2 : niveau 2
 * Sortie analogique : borne B5, [-10V,+10V]

Algorithme: Répéter

Charger A avec niveau 1
 Charger B avec durée T1
 Envoyer (A) vers CNA
 Attendre durée T1
 Charger A avec niveau 2
 Charger B avec durée T2
 Envoyer (A) vers CNA
 Attendre durée T2

Jusqu'à ordre de fin (* clavier RST)

a) Entrer par exemple : $N_1 = \$00$; $N_2 = \$FF$; $M_1 = \$FF$; $M_2 = \$7F$. Mesurer à l'oscilloscope les durées T_1 et T_2 , la fréquence et le rapport cyclique du signal.

b) Quelle est la fréquence minimale du signal généré ? La fréquence maximale ?

c) Les temporisations sont obtenues par le sous-programme *délai* ci-dessous (cf listing EPROM TPTS2). L'accumulateur B préalablement chargé par un nombre M compris entre 1 et 255 est décrémenté jusqu'à 0 :

```
delai  DECB          (B)-1 -> B
      BNE  delai    B = 0 ?
      RTS
      si oui, retour de sous-programme
```

La temporisation découle des relations : $T_1 = \theta.M_1 + \Delta t$, et $T_2 = \theta.M_2 + \Delta t$, où :

- θ est la durée d'exécution de la boucle programmée dans *délai*, qui comprend 2 instructions (DECB et BNE). D'après la table de programmation du μP 6809 (cf TP B11), calculer θ .

- Δt est un temps mort, dû à l'exécution du reste du programme (dont l'appel au sous-programme *délai*).

d) Soit $M_1 = M_2 = \$01$. Mesurer Δt à l'oscilloscope.

e) Que se passe-t-il si M_1 ou $M_2 = 0$?

f) Conclusion : quels sont les inconvénients de cette procédure de temporisation ?

g) Application : Modulation de Largeur d'Impulsions (MLI)

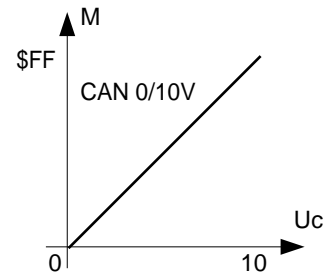
\$AC24 * Programme : génère un signal rectangulaire de rapport
 * cyclique proportionnel à une tension continue
 \$0F02 * Entrée N1 : niveau 1
 \$0F03 * Entrée N2 : niveau 2

- * Entrée analogique : borne B1, [0,+10V]
- * Sortie analogique : borne B5, [-10V,+10V]

Algorithme :

Répéter

```
Charger A avec niveau 1
Envoyer (A) vers CNA
Acquisition de T1 par le CAN
Attendre durée T1
Charger A avec niveau 2
Envoyer (A) vers CNA
Calcul de T2
Attendre durée T2
```



Jusqu'à ordre de fin


On génère un signal rectangulaire de rapport cyclique proportionnel à une tension d'entrée continue u_C comprise entre 0 et 10V. La tension u_C est acquise par le CAN 0/10V (borne entrée B1).

La durée T_1 est fonction de l'octet fourni par ce CAN : $T_1 = \theta.M + \Delta t$

La durée T_2 est fonction de son complément logique : $T_2 = \theta.\overline{M} + \Delta t$

Vérifier le fonctionnement du programme. Mesurer la fréquence du signal.

h) Etablir l'expression littérale du rapport cyclique α en fonction de M , puis de u_C .

i) Application numérique.  Rappel (voir cours § B11 page 3) : $M + \overline{M} = \$FF = 255$

Calculer α_{\min} et α_{\max} .

2) Valeurs $v_s(k)$ calculées

```
$AC3E      * Programme : génère une dent de scie
$0F00      * Entrée M : durée Te d'un échantillon
           * Sortie analogique : borne B5, [-10V,+10V]
```

Algorithme :

Initialiser A

Répéter

```
Charger A
Envoyer (A) vers CNA
Attendre durée Te
Incrémenter A
```

Jusqu'à ordre de fin

Le programme calcule la suite des valeurs des échantillons $v_s(k) = q.k + b$ par incréments successives à partir de $k = 0$. Lorsque $k = 255$, l'incrément suivante remet v_s à sa valeur initiale : $v_s(256) \equiv v_s(0)$.

a) Mesurer la fréquence maximale de fonctionnement (délai fixé à $M = \$01$).

b) Observer, en agrandissant l'image, les «marches d'escalier» qui composent la dent de scie.

La hauteur d'une "marche" n'est autre que l'incrément q du CAN. Le vérifier.

c) Restitution du signal analogique par filtre de lissage passe-bas du 2ème ordre (sortie sur la borne B3) : observer le signal obtenu.

3) Valeurs $v_s(k)$ lues dans une table

```
$AC4B      * Programme : génère un signal quelconque à partir
           * d'une table de valeurs
$0F00      * Entrée M : durée Te d'un échantillon
$0F01      * Entrée K : nombre d'échantillons/période
$0000      * Début de la table des échantillons
           * Sortie analogique : borne B5, [-10V,+10V]
```

Algorithme (page suivante) :

```

Répéter
  Pointer le début de la table
  Répéter
    Lire une valeur et charger A
    Envoyer (A) vers CNA
    Attendre durée Te
    Incrémenter le pointeur
  Jusqu'à fin de la table
Jusqu'à ordre de fin
    
```

Exemple : génération d'une sinusoïde d'amplitude A , de période T , formée de K échantillons de durée T_e par période, soit : $T = K.T_e$. Par exemple : $A = 6,25 \text{ V}$; $K = 20$; $f = 100 \text{ Hz}$.

a) Calculer successivement :

- Période $T = \frac{1}{f} \Rightarrow$ durée d'un échantillon $T_e = \frac{T}{K} \Rightarrow$ calcul du délai M d'après $T_e = \theta.M + \Delta t$
- Angles $\omega t = k \frac{2\pi}{K} \Rightarrow$ tensions $v_s = A \sin(\omega t) \Rightarrow$ niveaux $N = \lfloor c.v + d \rfloor$

Remplir le tableau :

k	$\omega t \text{ (rad)}$	$v_s \text{ (V)}$	$N \text{ (base10)}$	$\$N \text{ (hexa)}$	adresse
0					\$0000
1					\$0001
2					\$0002
etc...					etc...

b) Entrer au clavier à partir de l'adresse \$0000, la table des K entiers N calculés selon la méthode décrite. Exécuter et vérifier le fonctionnement du programme.

c) Modifier, au choix, la forme du signal obtenu, en agissant sur une ou plusieurs valeurs de la table.

II- Fonction : Calcul analogique (TP A21) → Calcul numérique

1) Recopie du signal d'entrée

```

$AC67          * Programme : recopie le signal d'entrée
                * Entrée analogique : borne B2, [-10V,+10V]
                * Sortie analogique : borne B5, [-10V,+10V]
    
```

Algorithme :

```

Répéter
  Acquisition de  $V_e(k)$  sur le CAN
   $V_s(k) = V_e(k)$ 
  Sortie de  $V_s(k)$  vers le CNA
Jusqu'à ordre de fin
    
```

Le signal $v_e(t)$, appliqué sur l'entrée 2 est obtenu à partir d'un GBF (signal sinusoïdal). Exécuter le programme. Observer le signal résultant $v_s(t)$ en sortie B5.

a) Période d'échantillonnage : mesurer T_e

b) En appliquant le théorème de Shannon, en déduire la fréquence maximale admissible F_{\max} du signal $v_e(t)$. Expérimenter. Que se passe-t-il au-delà de F_{\max} ?

c) Vérification de la période d'échantillonnage : le programme est le suivant :

Temps d'exécution des instructions (d'après table de programmation du $\mu P 6809$) ↓

prog3a LDA	can_bi	résultat de la conversion du CAN -> (A)	5 μs
STA	cna_bi	(A) -> CNA bipolaire	5 μs
NOP		attendre 2 μs (NOP = No Operation)	2 μs
NOP		attendre 2 μs	2 μs
NOP		attendre 2 μs	2 μs
BRA	prog3a	saut vers début du programme	3 μs

Calculer T_e .

d) Le CAN AD7574 étant identique à celui utilisé dans le TP B13, observer le signal \overline{RD} (broche 15). Quelle est la durée des impulsions formant ce signal ? Quelle est leur période ?

e) *Durée de conversion* : observer le signal \overline{BUSY} (broche 14) du CAN. Comparer la durée de ce signal $T_{\overline{BUSY}}$ à T_e . En déduire le rôle des trois instructions NOP ?

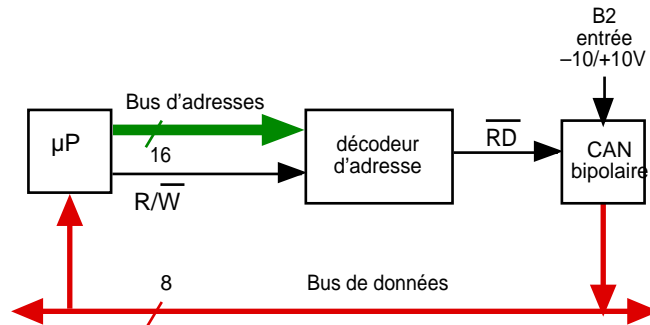
f) Pour savoir ce qui se passe si on diminue la temporisation insérée dans la boucle par les instructions NOP, écrire le programme suivant (en mémoire vive) et le tester ($f = 1\text{kHz}$). Conclure.

```

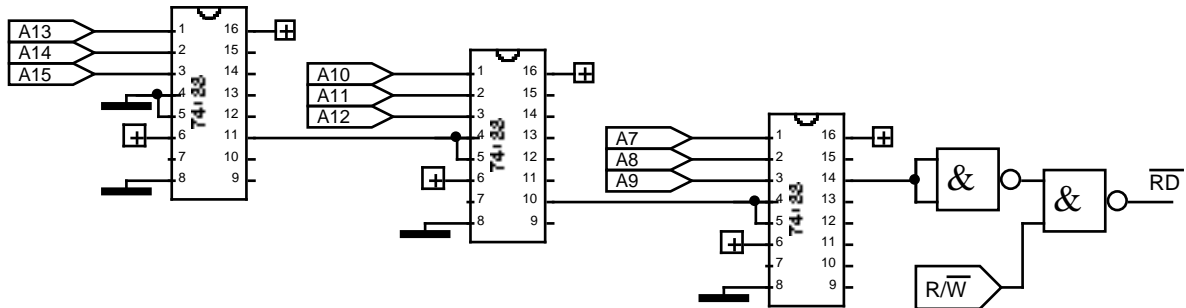
0000 B6 94 80      prog3b  LDA    can_bi
0003 B7 95 80                STA    cna_bi
0006 20 F8              BRA    prog3b
0008 3F                  SWI
    
```

g) *Dynamique d'entrée* : que se passe-t-il si la tension d'entrée excède 10V ?

h) *Adressage* : le résultat de la conversion est lu pendant l'instruction LDA can_bi en positionnant le signal \overline{RD} à "0". Pour cela, le μP écrit un "1" sur la ligne de lecture-écriture R / \overline{W} (Read actif à l'état haut) et écrit l'adresse du CAN sur le bus d'adresse :



Le schéma électrique du décodeur d'adresse du CAN est le suivant :

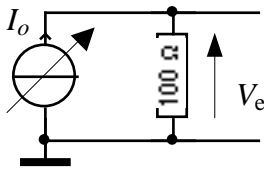


D'après ce schéma, calculer l'adresse du CAN (les bits A_0 à A_6 n'étant pas décodés, on les suppose à "0") et établir la relation booléenne $\overline{RD} = f(R / \overline{W}, A_7, \dots, A_{15})$, sachant que :

- les différents circuits sont sélectionnés par un 0 sur leur entrée de sélection (*CS* : *Chip Select*).
- la table de vérité du 74138 (décodeur/démultiplexeur "1 parmi 8") est la suivante :

broche	6	4	5	3	2	1	15	14	13	12	11	10	9	7
fonction	CS1	CS2	CS3	A2	A1	A0	Out 0	Out 1	Out 2	Out 3	Out 4	Out 5	Out 6	Out 7
	X	X	1	X	X	X	1	1	1	1	1	1	1	1
	X	1	X	X	X	X	1	1	1	1	1	1	1	1
	0	X	X	X	X	X	1	1	1	1	1	1	1	1
A0	1	0	0	1	1	1	1	1	1	1	1	1	1	0
A1	1	0	0	1	1	0	1	1	1	1	1	1	0	1
A2	1	0	0	1	0	1	1	1	1	1	1	0	1	1
CS2	1	0	0	1	0	0	1	1	1	1	0	1	1	1
CS3	1	0	0	0	1	1	1	1	1	0	1	1	1	1
CS1	1	0	0	0	1	1	1	1	1	0	1	1	1	1
Out 7	1	0	0	0	1	0	1	1	0	1	1	1	1	1
GND	1	0	0	0	1	0	1	1	0	1	1	1	1	1
	1	0	0	0	0	1	1	0	1	1	1	1	1	1
	1	0	0	0	0	0	0	1	1	1	1	1	1	1

2) Application : opération : $y = ax + b$



On désire convertir un courant analogique I_o délivré par un transmetteur 4/20 mA en une tension V_s comprise entre 0 et +8V. Pour cela, on transforme d'abord I_o en une tension V_e à l'aide d'une résistance de 100 Ω . On pose : $V_s = r.V_e + s$.

Pour simplifier, on se limite à un mode de fonctionnement unipolaire, donc non signé.

La tension V_e est appliquée au CAN unipolaire (entrée B1). La tension V_s est issue du CNA unipolaire (sortie B4). CAN et CNA unipolaires [0/10V] sont tels que :

- à une tension de 0 V correspond une valeur hexadécimale \$00
- à une tension de +10V correspond une valeur hexadécimale \$FF

a) Calculer r et s sous forme décimale puis hexadécimale.

b) Compléter puis tester le programme suivant, qui exécute l'opération $V_s = r.V_e + s$:

```

0000 B6 94 00      prog3c   LDA    can_un      Ve -> (A) (entrée B1)
0003 C6 **         LDB    #r          r -> (B)
0005 3D           MUL    (A) X (B) -> (A:B)
0006 C0 **         SUBB   #s          (B) - s -> (B)
0008 F7 95 00     STB    cna_un     (B) -> Vs (sortie B4)
000B 20 F3        BRA    prog3c   retour debut
000D 3F           SWI                   FIN
    
```



Ligne 0005 : une multiplication entre deux entiers de 8 bits (*instruction MUL*) est susceptible de donner un résultat dont la taille peut atteindre 16 bits (car $\$FF \times \$FF = 255^2 = \$FE01$). Ce résultat est sauvegardé dans les deux accumulateurs A (octet de poids fort) et B (octet de poids faible). Mais pour rester en format 8 bits, ici seul le poids faible est retenu (ligne 0006). Une tension d'entrée excessive se traduira donc par un dépassement de calcul.

- c) Vérifier le bon fonctionnement du programme ; utiliser un GBF pour générer $v_e(t)$.
- d) Période d'échantillonnage : mesurer T_e . Pourquoi est-elle modifiée par rapport au programme prog3a ?
- e) Dynamique d'entrée : mettre en évidence le dépassement de calcul lorsque $V_e > 2$ V ou $V_e < 0,4$ V.
- f) Résolution : calculer les incréments de tension V_s , de tension V_e , de courant I_o ?

III- Fonction : Comparaison TOR analogique (TP A24) → Comparaison numérique

Soit un système de refroidissement à ventilateur commandé par un comparateur à hystérésis :

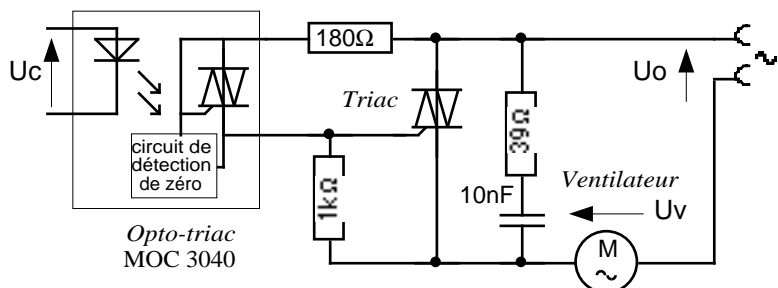
a) Donner un exemple de chronogramme des tensions U_c (tension de commande TOR), U_0 et U_v .

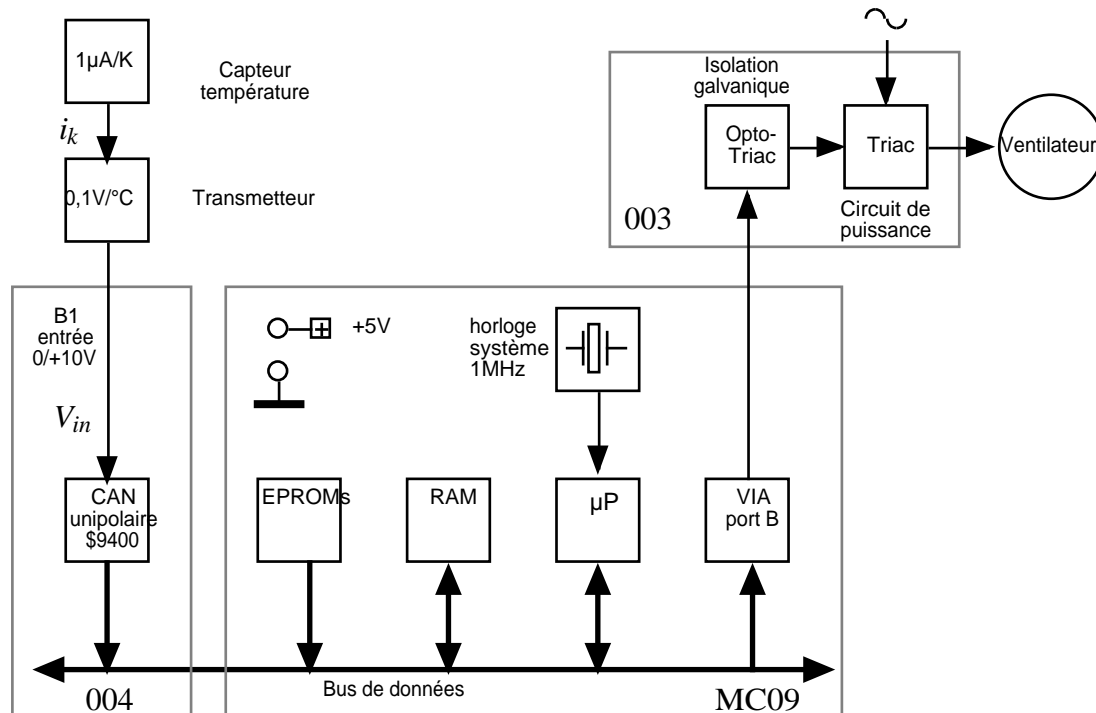
(NB : tenir compte du circuit de détection du zéro secteur)

Programme :

```

$AD21      * Programme : comparateur à hystérésis
$0F00      * Entrée Nb : température seuil bas
$0F01      * Entrée Nh : température seuil haut
           * Entrée analogique : borne B1, [0,+10V]
           * Sortie binaire : commande de triac, sur carte 003
    
```





Le ventilateur est en marche lorsque la température est trop élevée. Il est à l'arrêt lorsque la température est redescendue en dessous d'un certain seuil.

b) Sachant que le transmetteur délivre une tension V_{in} valant $0,1 \text{ V}/^\circ\text{C}$ au CAN unipolaire (dont la tension d'entrée est comprise entre 0 et 10V), quelle est la gamme de température mesurable ?

c) Soit N le nombre hexadécimal codant la température. Calculer N_b pour $\theta = 22^\circ\text{C}$ et N_h pour $\theta = 27^\circ\text{C}$.

d) Tracer la courbe de l'état du ventilateur (Marche/Arrêt) en fonction de θ .

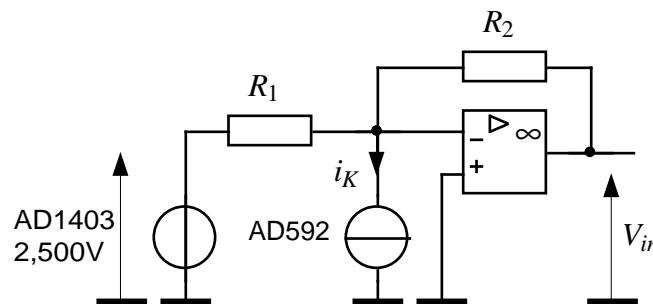
e) Tracer un organigramme simplifié du programme.

f) Exécuter le programme et vérifier son fonctionnement (connecter la résistance chauffante située devant le ventilateur et la sonde à une source de tension $\approx 12 \text{ V}$).

g) Le capteur de température est une sonde semi-conductrice AD592 (Analog Devices). Cette sonde se comporte en récepteur de courant : elle absorbe un courant constant i_K , étalonné à $1\mu\text{A}/\text{K}$.

Elle fonctionne dans une gamme de température comprise entre -25°C et $+105^\circ\text{C}$.

Le schéma du transmetteur est le suivant :



Le circuit AD1403 est une référence de tension de précision égale à $2,500\text{V}$.

Établir la relation $V_{in} = f(i_K, R_1, R_2)$; A.N. : calculer R_1 et R_2 .

Commentaires

0- Codage numérique de l'information analogique

L'équation de la droite $n(v)$ est :

$$n = c.v + d \text{ avec } c = -\frac{255}{20} = -12,75 \text{ V}^{-1} \text{ et } d = 127,5$$

En admettant que N est arrondi à l'entier le plus proche :

$$N = \lfloor -12,75.v + 128 \rfloor$$

NB1 : $\lfloor x \rfloor$ signifie "partie entière de x "

NB2 : l'arrondi à l'entier le plus proche se calcule à partir de $\lfloor x + 0,5 \rfloor$

Réciproquement, l'équation de la droite $v(N)$ est :

$$v = -q.N + b \text{ avec } q = \frac{\text{pleine échelle}}{2^8 - 1} = \frac{20}{255} \approx 78,4 \text{ mV/unité et } b = 10V$$

I- Fonction : Générateur analogique de signaux (TP A25) → Générateur numérique

0) Principe : génération d'un signal continu

```
* 2-0 : - génère un signal continu
*       - échelle : [$00,$FF] <-> [+10V,-10V]
*       - sortie CNA : borne 5
*
* $0F02 : octet hexadécimal à convertir
*
```

```
AC00 B6 0F 02      prog0  LDA      $0F02          donnée -> A
AC03 B7 95 80      STA      cna_bi          (A) -> CNA
AC06 3F             SWI
```

+10V → \$00; +5V → \$40 ; 0V → \$80; -5V → \$BF ; -10V → \$FF

1) Valeurs $v_s(k)$ incluses dans le programme

* Sous-programme de temporisation (pour 2-1 & sq)

```
*       - durée tempo = 5μs x (B)
*
```

```
AC07 5A           delai  DECB          (B)-1 -> B
AC08 26 FD       BNE      delai      B = 0 ?
AC0A 39           RTS          si oui, retour
```

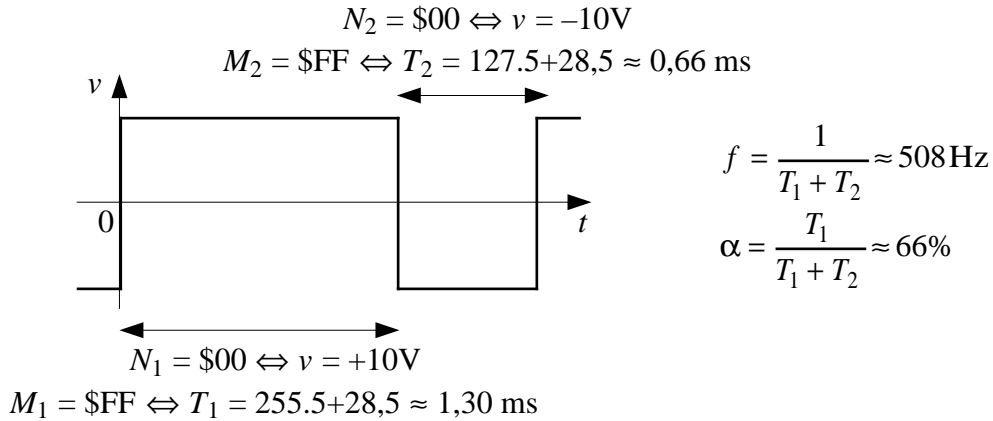
```
* 1-1 : - génère un signal rectangulaire
*       - échelle : [$00,$FF] <-> [+10V,-10V]
*       - sortie CNA : borne 5
*
```

```
* $0F00 : durée niveau 1
* $0F01 : durée niveau 2
* $0F02 : niveau 1
* $0F03 : niveau 2
*
```

```
AC0B B6 0F 02      prog1  LDA      $0F02          niveau 1 -> CNA
AC0E B7 95 80      STA      cna_bi
AC11 F6 0F 00      LDB      $0F00          tempo niveau 1
AC14 8D F1         BSR      delai
AC16 B6 0F 03      LDA      $0F03          niveau 2 -> CNA
AC19 B7 95 80      STA      cna_bi
```


AC1C F6 0F 01	LDB	\$0F01	tempo niveau 2
AC1F 8D E6	BSR	delai	
AC21 20 E8	BRA	progl	boucle
AC23 3F	SWI		

a) Voir calcul ci-dessous (c)



b) Voir calcul ci-dessous (c)

c) Calcul théorique : le temps mort Δt correspond au temps d'exécution des instructions du programme principal, qui inclue notamment l'appel au sous-programme de délai et le retour de ce sous-programme. Ce temps mort se répète deux fois, puisque le programme s'exécute en deux parties, une pour chaque niveau de tension. D'après les données constructeurs (cf table de programmation μP 6809 MOTOROLA), on trouve, sachant qu'un cycle machine dure $1 \mu s$ dans ce système :

AC07 5A	delai	DECB		3 μs	
AC08 26 FD		BNE	delai	2 μs	=>delai = 3+2 = 5 μs
AC0A 39		RTS		5 μs	retour sous-prog
	*				
AC0B B6 0F 02	progl	LDA	\$0F02	5 μs	1ère partie
AC0E B7 95 80		STA	cna_bi	5 μs	
AC11 F6 0F 00		LDB	\$0F00	5 μs	
AC14 8D F1		BSR	delai	7 μs	appel sous-prog
...					
AC21 20 E8		BRA	progl	3 μs	3/2= 1,5 μs /partie

$$\Delta t_1 = 5 + 5 + 5 + 7 + 5 = 27 \mu s$$

$$\Delta t_2 = 5 + 5 + 5 + 7 + 5 + 3 = 30 \mu s$$


NB₁ : l'appel au sous-programme nécessite 7 (BSR) + 5 (RTS) = 12 μs

NB₂ : la dernière instruction (BRA prog1) permet au programme de se dérouler en boucle indéfiniment. Elle n'est exécutée que dans la deuxième partie du programme.

Il est donc possible de calculer les fréquences minimale et maximales :

$$f_{\min} = \frac{10^6}{2M_{\max} \cdot \theta + \Delta t_1 + \Delta t_2} = \frac{10^6}{2 \cdot 255.5 + 27 + 30} \approx 384 \text{ Hz}$$

$$f_{\max} = \frac{10^6}{2M_{\min} \cdot \theta + \Delta t_1 + \Delta t_2} = \frac{10^6}{2 \cdot 1.5 + 27 + 30} \approx 14925 \text{ Hz}$$

e)  Si l'on choisit $M = 0$, l'instruction DECB (adresse AC07) décrémente B d'une unité. Or, si B est préchargé à zéro, il contient -1 après décrémentation, soit \$FF en code complément à deux. Donc tout se passe comme si on avait préchargé B avec cette valeur : au lieu d'atteindre un maximum

de fréquence, on observe un minimum de fréquence !

f) les inconvénients d'une temporisation logicielle sont nombreux :

- imprécision due à l'existence de temps morts provoqués par l'exécution d'instructions qui ne font pas spécifiquement partie du sous-programme de temporisation.

- sensibilité de la temporisation à la fréquence de l'horloge système : *si celle-ci est modifiée, la temporisation est modifiée !*

- la modification du programme, par exemple par l'insertion de lignes de code supplémentaires, modifie le temps d'exécution de celui-ci !

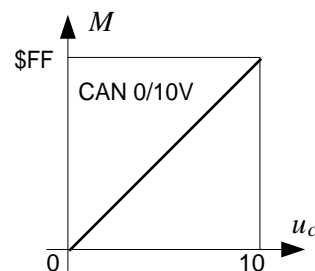
Remède : pour obtenir une temporisation stable, il est préférable de faire appel à un circuit spécialisé ("TIMER") indépendant du processeur et de prévoir une programmation par interruptions (voir TP B21).

g)

```
* 1-1bis:- génère un signal rectangulaire de rapport
* cyclique proportionnel à une tension continue
*   - entrée : éch : [$00,$FF] <-> [0,+10V]
*           CAN : borne 1
*   - sortie : éch : [$00,$FF] <-> [+10V,-10V]
*           CNA : borne 5
*
* $0F02 : niveau 1
* $0F03 : niveau 2
```

AC24 B6 0F 02	proglb	LDA	\$0F02	5 µs	niveau 1 -> CNA
AC27 B7 95 80		STA	cna_bi	5 µs	
AC2A F6 94 00		LDB	can_un	5 µs	durée niveau 1 =
AC2D 8D D8		BSR	delai	5 µs	valeur lue dans le CAN
AC2F B6 0F 03		LDA	\$0F03	5 µs	niveau 2 -> CNA
AC32 B7 95 80		STA	cna_bi	5 µs	
AC35 F6 94 00		LDB	can_un	5 µs	durée niveau 2 =
AC38 53		COMB		2 µs	256 - durée niveau 1
AC39 8D CC		BSR	delai	7+5 µs	
AC3B 20 E7		BRA	proglb	3 µs	boucle
AC3D 3F		SWI			

h) Equation du CAN 0-10 V : $M = \left\lfloor \frac{255}{10} u_c \right\rfloor$



$$\Rightarrow \alpha = \frac{T_1}{T_1 + T_2} = \frac{M \cdot \theta + \Delta t_1}{M \cdot \theta + \Delta t_1 + M \cdot \theta + \Delta t_2} = \frac{M \cdot \theta + \Delta t_1}{255 \cdot \theta + \Delta t_1 + \Delta t_2} \approx \frac{M}{255} = \frac{u_c}{10} \text{ pour } M \text{ assez grand.}$$

NB₁ : M étant un nombre entier, on remarque que le rapport cyclique varie par incréments d'1/255ème.

NB₂ : Pour être tout-à-fait précis, et comme précédemment, on peut calculer les temps morts pour chaque niveau :

$$\Delta t_1 = 5 + 5 + 5 + 7 + 5 = 27 \mu\text{s}$$

$$\Delta t_2 = 5 + 5 + 5 + 2 + 7 + 5 + 3 = 32 \mu\text{s}$$

$$i) \alpha_{\min} = \frac{\theta + \Delta t_1}{255 \cdot \theta + \Delta t_1 + \Delta t_2} = \frac{5 + 27}{255 \cdot 5 + 27 + 32} \approx 2,4\%$$

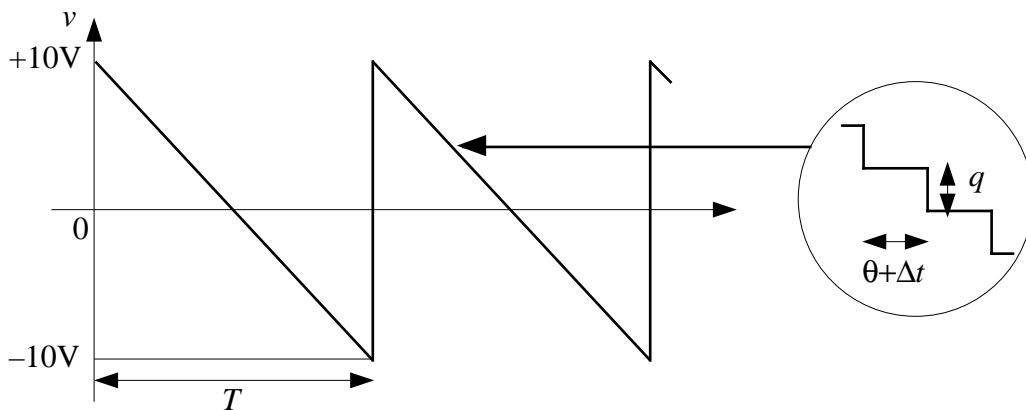
$$\alpha_{\max} = \frac{255.\theta + \Delta t_1}{255.\theta + \Delta t_1 + \Delta t_2} = \frac{255.5 + 27}{255.5 + 27 + 32} \approx 98\%$$

2) Valeurs $v_s(k)$ calculées

* 1-2 : - génère une dent de scie
 * - sortie CNA : borne 5
 *
 * \$0F00 : durée d'un échantillon (B)
 *

AC3E 4F	prog2	CLRA	RAZ de A
AC3F B7 95 80	debut2	STA cna_bi	5 μ s (A) -> CNA
AC42 F6 0F 00		LDB \$0F00	5 μ s durée d'un échantillon
AC45 8D C0		BSR delai	7+5 μ s
AC47 4C		INCA	2 μ s (A)+1 -> A
AC48 20 F5		BRA debut2	3 μ s
AC4A 3F		SWI	

a)



b) Le délai (largeur d'une "marche d'escalier") vaut : $\theta + \Delta t = 5 + 5 + 5 + 7 + 5 + 2 + 3 = 32 \mu$ s

La fréquence vaut : $f = \frac{1}{255(\theta + \Delta t)} \approx 122$ Hz

La hauteur d'une "marche" est l'incrément : $q \approx 78$ mV

c) Après filtrage, on constate que les "marches" ont disparu et que le signal est lissé.

NB : la pente de la dent de scie est inversée, car le filtre passe-bas introduit un signe négatif (montage AOP inverseur)

3) Valeurs $v_s(k)$ lues dans une table

* 1-3 : - génère un signal quelconque à partir
 * d'une table de valeurs
 * - sortie CNA : borne 5
 *
 * \$0F00 : durée d'un échantillon (B)
 * \$0F01 : nombre d'échantillons/période (1 à 256)
 * \$0000 : début de la table des échantillons
 * (variable interne : \$0F02 : compteur)
 *

AC4B 10 8E 00 00	prog3	LDY #0000	adr base table -> Y
AC4F B6 0F 01		LDA \$0F01	nb d'échant./période
AC52 B7 0F 02		STA \$0F02	décompteur en \$0F02
AC55 A6 A0	lect3	LDA 0,Y+	4+2 μ s lecture table
AC57 B7 95 80		STA cna_bi	5 μ s

AC5A F6 0F 00	LDB	\$0F00	5 μ s	durée d'un échantillon
AC5D 8D A8	BSR	delai	7+5 μ s	
AC5F 7A 0F 02	DEC	\$0F02	7 μ s	décrémenter compteur
AC62 26 F1	BNE	lect3	3 μ s	si pas = 0, continuer
AC64 20 E5	BRA	prog3		
AC66 3F	SWI			

$$a) \Delta t = 6 + 5 + 5 + 7 + 5 + 7 + 3 = 38 \mu\text{s}$$

$$\text{Période demandée : } T = \frac{1}{f} = \frac{1}{100} = 0,01 \text{ s} = 10000 \mu\text{s}$$

$$\Rightarrow \text{durée d'un échantillon } T_e = \frac{T}{K} = \frac{10000}{20} = 500 \mu\text{s}$$

$$\Rightarrow \text{calcul du délai } M \text{ d'après } T_e = \theta \cdot M + \Delta t : M = \frac{T_e - \Delta t}{\theta} = \frac{500 - 38}{5} \approx 92 = \$5C$$

tableau :

k	$wt \text{ (rad)}$	$v_s \text{ (V)}$	$N \text{ (base10)}$	$\$N \text{ (hexa)}$	adresse
0	0,00	0,00	128	80	\$0000
1	0,31	1,93	103	67	\$0001
2	0,63	3,67	80	50	\$0002
3	0,94	5,06	63	3F	\$0003
4	1,26	5,94	51	33	\$0004
5	1,57	6,25	48	30	\$0005
6	1,88	5,94	51	33	\$0006
7	2,20	5,06	63	3F	\$0007
8	2,51	3,67	80	50	\$0008
9	2,83	1,93	103	67	\$0009
10	3,14	0,00	127	7F	\$000A
11	3,46	-1,93	152	98	\$000B
12	3,77	-3,67	175	AF	\$000C
13	4,08	-5,06	192	C0	\$000D
14	4,40	-5,94	204	CC	\$000E
15	4,71	-6,25	208	DO	\$000F
16	5,03	-5,94	204	CC	\$0010
17	5,34	-5,06	192	C0	\$0011
18	5,65	-3,67	175	AF	\$0012
19	5,97	-1,93	152	98	\$0013

II- Fonction : Calcul analogique (TP A21) → Calcul numérique

1) Recopie du signal d'entrée

```
* 3-1 : - recopie le signal d'entrée
*       - entrée : éch : [$00,$FF] <-> [+10V,-10V]
*       CAN : borne 2
*       - sortie : éch : [$00,$FF] <-> [+10V,-10V]
*       CNA : borne 5
*
```

AC67 B6 94 80	prog3a	LDA	can_bi
AC6A B7 95 80		STA	cna_bi
AC6D 12		NOP	
AC6E 12		NOP	
AC6F 12		NOP	
AC70 20 F5		BRA	prog3a
AC72 3F		SWI	

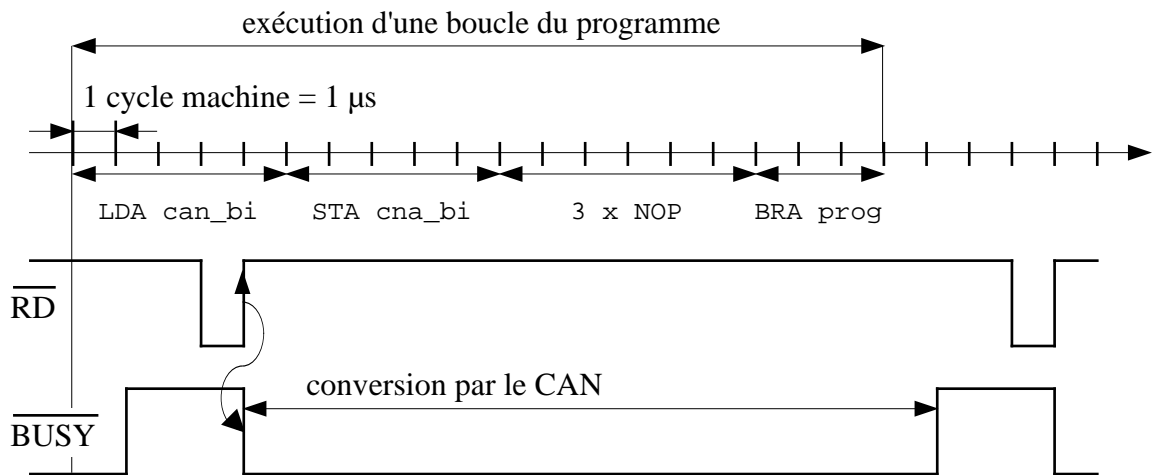
a) et c) $T_e = 5 + 5 + 2 + 2 + 2 + 3 = 19 \mu\text{s} \Leftrightarrow F_e = 52,6 \text{ kHz} \Rightarrow$ b) $F_{\text{max}} = F_e/2 \approx 26,3 \text{ kHz}$

d) et e) Période $19 \mu\text{s}$.

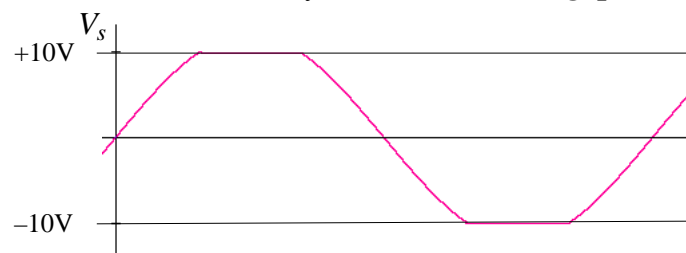
L'instruction `LDA can_bi` charge l'accumulateur A par le résultat de la conversion effectuée précédemment par le CAN bipolaire $+10/-10 \text{ V}$. Cette instruction dure 5 cycles machines, soit $5 \mu\text{s}$. La lecture proprement dite ($\overline{\text{RD}}=0$) dure un seul cycle machine, les autres cycles consistant pour le microprocesseur à aller chercher le code opératoire (`B6`) en mémoire, le décoder, positionner le bus d'adresse à la valeur indiquée (`9480`), et charger l'instruction suivante.

⚠ Bien entendu, il faut que : $T_{\overline{\text{BUSY}}} \leq T_e$: le CAN doit pouvoir convertir le signal en un temps plus court que le temps d'exécution d'une boucle de programme ! Comme ce temps de conversion est de l'ordre de 15 à $17 \mu\text{s}$, il faut que le temps d'exécution d'une boucle soit un peu supérieur : c'est le rôle des 3 instructions `NOP` ("No Operation") de ralentir cette exécution en ajoutant un temps mort de $6 \mu\text{s}$ (une instruction `NOP` s'étend sur 2 cycles machines).

f) Sans ces 3 instructions `NOP`, une boucle dure $5 + 5 + 3 = 13 \mu\text{s}$, ce qui est insuffisant : on constate que le signal d'entrée n'est plus recopié convenablement.



g) Le signal doit être compris entre -10 et $+10 \text{ V}$ (\Leftrightarrow la dynamique d'entrée est égale à la Pleine Echelle du CAN, soit 20 V). Au-delà de 10 V , il y a *saturation analogique* :



h) L'adresse du CAN correspond aux conditions :

1er décodeur, sortie 11 : $A_2 A_1 A_0 = 100 \Rightarrow A_{15} \cdot \overline{A_{14}} \cdot \overline{A_{13}}$

2ème décodeur, sortie 10 : $A_2 A_1 A_0 = 101 \Rightarrow \overline{A_{12}} \cdot \overline{A_{11}} \cdot A_{10}$

3ème décodeur, sortie 14 : $A_2 A_1 A_0 = 001 \Rightarrow \overline{A_9} \cdot \overline{A_8} \cdot A_7$

Soit, au total : $\overline{A_{15}} \cdot \overline{A_{14}} \cdot \overline{A_{13}} \cdot \overline{A_{12}} \cdot \overline{A_{11}} \cdot A_{10} \cdot \overline{A_9} \cdot \overline{A_8} \cdot A_7$

Don, en commençant par le bit de poids fort (A_{15}), l'adresse vaut : `%100101001XXXXXXX`, où la lettre "X" désigne les bits d'adresse qui ne sont pas décodés. En prenant arbitrairement $X = 0$, il vient : adresse CAN = `% 1001 0100 1000 0000 = $9480`

Enfin, une lecture directe du schéma donne : $\overline{\text{RD}} = (\overline{\text{R}} / \overline{\text{W}}) \cdot (\overline{A_{15}} \cdot \overline{A_{14}} \cdot \overline{A_{13}} \cdot \overline{A_{12}} \cdot \overline{A_{11}} \cdot A_{10} \cdot \overline{A_9} \cdot \overline{A_8} \cdot A_7)$

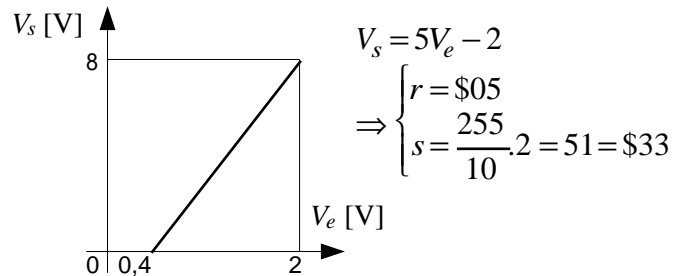
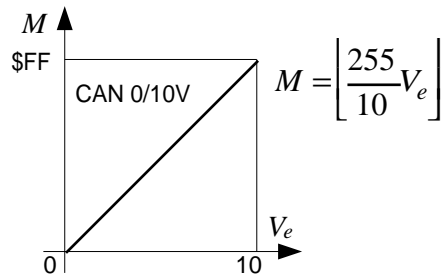
2) Application : opération : $y = ax + b$

Conversion courant/tension par une résistance de 100Ω : $V_e = 100 I_o$

On pose : $V_s = r.V_e + s$

Equation du CAN 0-10 V :

Mise à l'échelle :



d) Temps d'exécution d'une boucle de programme :

0000 B6 94 00	prog3c	LDA	can_un	5 μ s
0003 C6 05		LDB	#r	2 μ s
0005 3D		MUL		11 μ s
0006 C0 33		SUBB	#s	2 μ s
0008 F7 95 00		STB	cna_un	5 μ s
000B 20 F3		BRA	prog3c	3 μ s
000D 3F		SWI	FIN	

$T_e = 5 + 2 + 11 + 2 + 5 + 3 = 28 \mu$ s

e) Le signal V_s doit être compris entre 0 et 8V (et non entre 0 et 10 V !). Au-delà, il y a **saturation numérique par dépassement de capacité de calcul** :

- Le résultat (intermédiaire) de l'instruction de multiplication (MUL, ligne 0005) doit être inférieur ou égal à \$FF = 255 :

$$5M \leq 255 \Rightarrow M \leq \frac{255}{5} = 51 \Rightarrow V_e \leq \frac{10}{255} \cdot 51 = 2 \text{ V}$$

Exemple : si $V_e = 4 \text{ V}$, alors $M = 102$ et $5M = 510$, qui est un nombre écrit sur 9 bits !

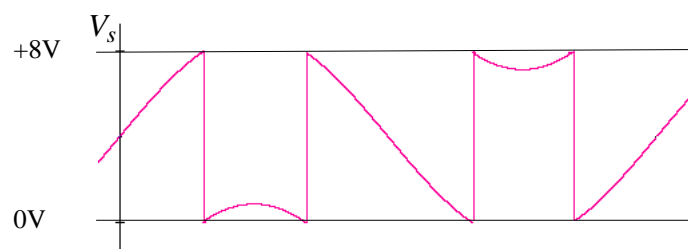
! Si V_e dépasse 2V, alors le résultat devient supérieur ou égal à 256 (\$100 en hexadécimal). Comme la retenue n'est pas prise en compte dans le programme, tout se passe comme si ce nombre repassait par zéro.

- Le résultat de la soustraction (SUB, ligne 0006) doit rester positif ou nul (puisque'on utilise en sortie un CNA unipolaire 0/10 V). Il faut donc que :

$$V_s = 5V_e - 2 \geq 0 \Rightarrow V_e \geq 0,4 \text{ V}$$

! Si V_e devient inférieur à 0,4V, alors le résultat devient inférieur à zéro. Or, en code complément à 2, qui est le code signé utilisé dans les calculs par le microprocesseur, -1 par exemple s'écrit \$1FF en hexadécimal. Non seulement la retenue n'est pas prise en compte, mais tout se passe comme si ce nombre repassait par \$FF (équivalent à la tension maximale de 10V).

- Conclusion : la dynamique d'entrée est donc égale à $2 - 0,4 = 1,6 \text{ V}$. Si l'on sort de cette échelle, on obtient le résultat suivant :



f) Résolution du CNA et du CAN : $\Delta V_s = \Delta V_e = \frac{10}{255} \approx 39 \text{ mV}$

Résolution du transmetteur : $\Delta I_o = \frac{\Delta V_e}{100} \approx 0,39 \text{ mA}$

! La Pleine Echelle du CAN (10V) est supérieure à la dynamique d'entrée \Rightarrow l'incrément, qui vaut $10/255$, est supérieur au rapport $1,6/255$. La résolution d'entrée est moins bonne que celle à laquelle on pourrait s'attendre avec un CAN de 8 bits car on n'utilise pas toute l'échelle possible (0,4-2V au lieu de 0-10V). Il en est de même de la sortie (0-8V au lieu de 0-10V).

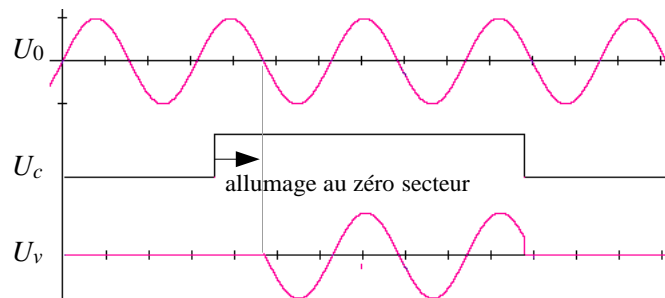
Conclusion : ne pas confondre dynamique d'entrée et pleine échelle !

III- Fonction : Comparaison TOR analogique (TP A24) \rightarrow Comparaison numérique

```
* exercice de synthèse n°1 : comparateur à hystérésis
*   - entrée : éch : [$00,$FF] <-> [0,+100°C]
*           CAN : borne 1
*   - sortie : 1 = arrêt moteur ($FF)
*           0 = marche moteur ($00)
*           VIA : connecteur 1
*
* $0F00 : température seuil bas
* $0F01 : température seuil haut
*
```

```
AD21 86 FF      tor      LDA      #$FF          port B VIA en sortie
AD23 B7 84 02          STA      cbvia
AD26 86 FF          LDA      #$FF          init: moteur à l'arrêt
AD28 B7 84 00          STA      rbvia
AD2B B6 94 00      trloop  LDA      can_un        boucle
AD2E B1 0F 00          CMPA     $0F00
AD31 25 07          BLO      off
AD33 B1 0F 01          CMPA     $0F01
AD36 22 09          BHI      on
AD38 20 F1          BRA      trloop
AD3A 86 FF      off      LDA      #$FF          arrêt moteur
AD3C B7 84 00          STA      rbvia
AD3F 20 EA          BRA      trloop
AD41 86 00      on      LDA      #$00
AD43 B7 84 00          STA      rbvia
AD46 20 E3          BRA      trloop
AD48 3F          SWI
```

a)

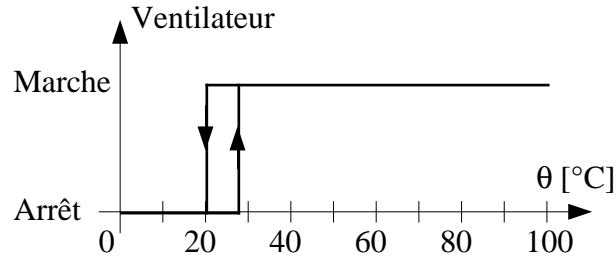


b) Gamme : 0/100°C

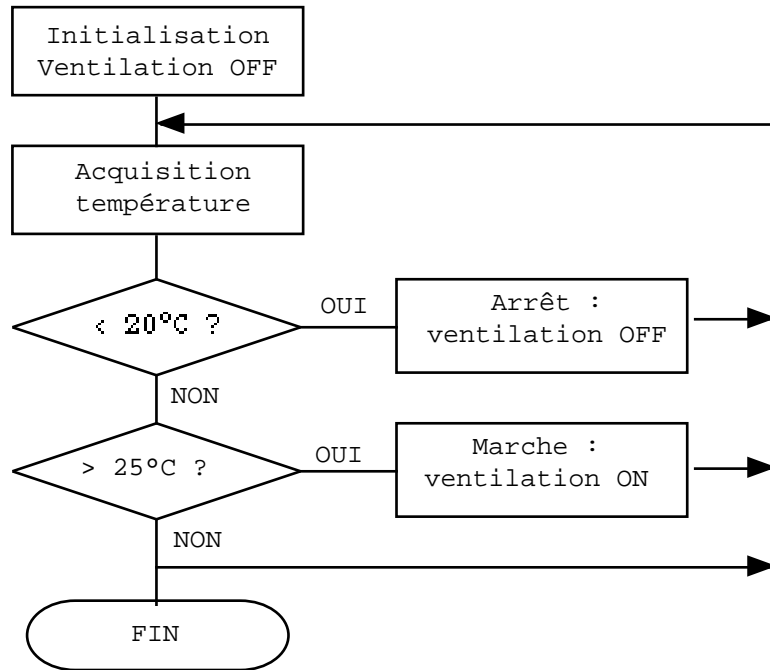
c) $N = \left\lfloor \frac{255}{100} \theta \right\rfloor \Rightarrow N_b = 56 = \38 pour $\theta \approx 22^\circ\text{C}$; $N_h = 68 = \$44$ pour $\theta \approx 27^\circ\text{C}$

(ces valeurs sont arrondies à l'entier inférieur)

d)



e)



h) En appliquant la loi des nœuds sur l'entrée - de l'AOP :

$$\sum i = 0 \Rightarrow \frac{V_{in}}{R_2} + \frac{2,500}{R_1} - i_K = 0 \Rightarrow V_{in} = R_2 \left(i_K - \frac{2,500}{R_1} \right)$$

$$V_{in} = 0 \text{ pour } i_K = 273 \mu\text{A} (0^\circ\text{C}) \Rightarrow R_1 = \frac{2,500}{273 \cdot 10^{-6}} = 9157,5 \Omega$$

$$V_{in} = 10 \text{ V pour } i_K = 373 \mu\text{A} (100^\circ\text{C}) \Rightarrow R_2 = \frac{10}{100 \cdot 10^{-6}} = 100 \text{ k}\Omega$$