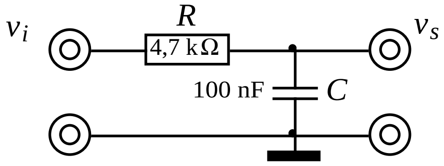
B21 - Discrétisation

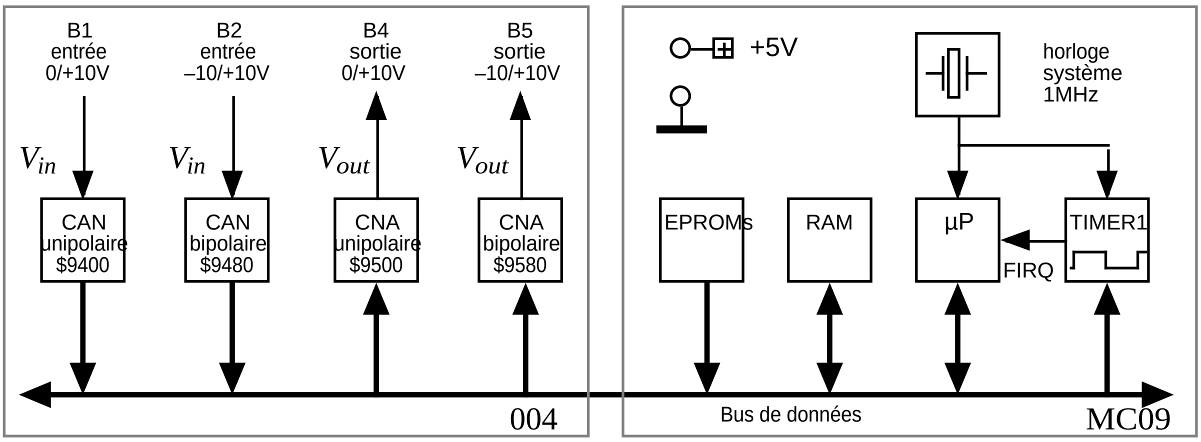
But : discrétiser ***en temps réel*** *un signal analogique...*

Matériel : système analogique identique au TP A14 :

$$H(p) = \frac{1}{1+\tau p} \ ; \ F_0 = \frac{1}{2\pi RC}$$



Système numérique identique aux TP B15, B22, B23 :



I- Contrôle de la période d'échantillonnage

Pour réaliser un filtre numérique (et, d'une manière générale, tout type de traitement numérique du signal), il est capital de contrôler avec précision la période avec laquelle les échantillons successifs du signal d'entrée $v_e(t)$ sont acquis, car T_e est au système numérique ce que τ est au système analogique : le paramètre fondamental ! Or la méthode par temporisation logicielle vue précédemment *(TP B15-B17)* présente de nombreux inconvénients. Il suffit par exemple de modifier une ligne de programme pour que T_e soit modifiée.

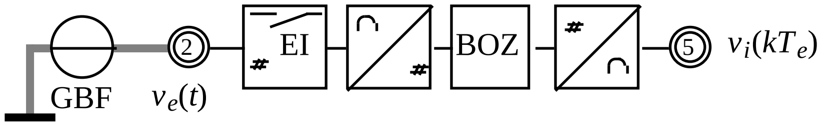
Si l'on veut que la période d'échantillonnage soit indépendante de la vitesse avec laquelle le programme est exécuté, il faut disposer d'un circuit supplémentaire jouant le rôle de "base de temps" (*timer*). Ici, ce rôle est rempli par un circuit spécialisé, le temporisateur programmable 6840 (TIMER1 sur le schéma ci-dessus), qui génère des impulsions à intervalle fixé vers l'entrée FIRQ du µP (FIRQ : *Fast Interrupt Request* - demande d'interruption rapide). Cet intervalle de temps est la période d'échantillonnage.

Dans ce type de fonctionnement, appelé "programmation par interruptions" ou "par événements" ou encore "programmation temps réel", le µP exécute une boucle d'attente. Lorsqu'il reçoit une impulsion sur son entrée FIRQ, l'exécution est déroutée vers un "sous-programme d'interruption", qui exécute l'algorithme désiré. Dès que ce calcul est terminé, le µP retourne dans sa boucle d'attente.

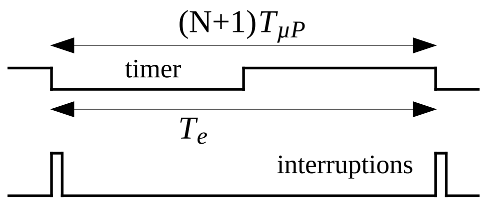
```
$AC73                * Programme : recopie le signal d'entrée avec contrôle
                     * de la période d'échantillonnage Te
$0F00                * Entrée NH (MSB)
$0F01                * Entrée NB (LSB)
                     * NH:NB = Te - 1 µs
                     * Entrée analogique : borne B2, [–10V,+10V]
                     * Sortie analogique : borne B5, [–10V,+10V]
Programme principal :
          Initialiser TIMER1
          Boucle :
```

```
                    Aller à Boucle
          Jusqu'à ordre de fin
Sous-programme d'interruption :
          Préchargement TIMER1
          Acquisition de Ve(k) sur le CAN
          Vs(k) = Ve(k)
          Sortie de Vs(k) vers le CNA
          Retour de sous-programme
```

Ce programme fait une simple recopie du signal $v(t)$. Il est équivalent à un Echantillonneur Instantané (EI) suivi d'un Bloqueur d'Ordre Zéro (BOZ) :



Programmation du *timer* :
$T_{\mu P}$ = durée d'un cycle machine.
Ici, $f_{\text{système}}$ = 1 MHz $\Rightarrow T_{\mu P}$ = 1 µs.



Exemple : on veut F_e = 10kHz $\Rightarrow T_e$ = 100µs $\Rightarrow N$ = 99 = $0063 $\Rightarrow$ adresses [0F00] = $00 : [0F01] = $63. Remplir ces deux adresses (`MEM 0F00 00 INC 63 FIN`). Exécuter le programme (`EXC AC73`).

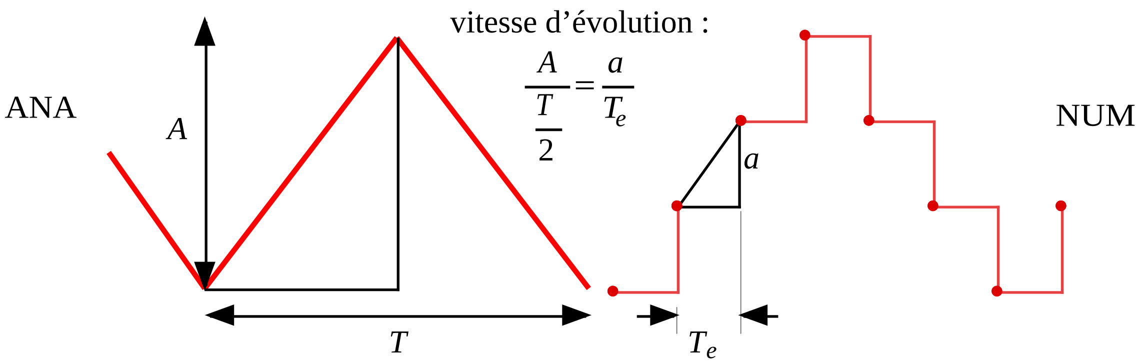
$v_e(t)$ est une tension triangulaire d'amplitude crête à crête A = 20V (= PE, pleine échelle). Vérifier que la période d'échantillonnage est bien de 100µs.

II- Influence de la fréquence d'échantillonnage sur la résolution pratique

Calculer la résolution *théorique* q du système d'après : $q = \dfrac{A}{2^n - 1}$ (n nb de bits du système, ici 8).

Relever $v_i(t)$ pour différentes fréquences de $v_e(t)$: $F_e/2$ (fréquence de Shannon), $F_e/4$, $F_e/8$.

On sait que les vitesses d'évolution des signaux analogique et numérique sont identiques :



En déduire, dans chaque cas, la valeur de la résolution *a pratique.*

Quelle doit être la valeur maximale de la fréquence du signal analogique (en fonction de F_e) pour avoir $a = q$ sachant que n = 8 ?

III- Mise au point de l'algorithme d'acquisition

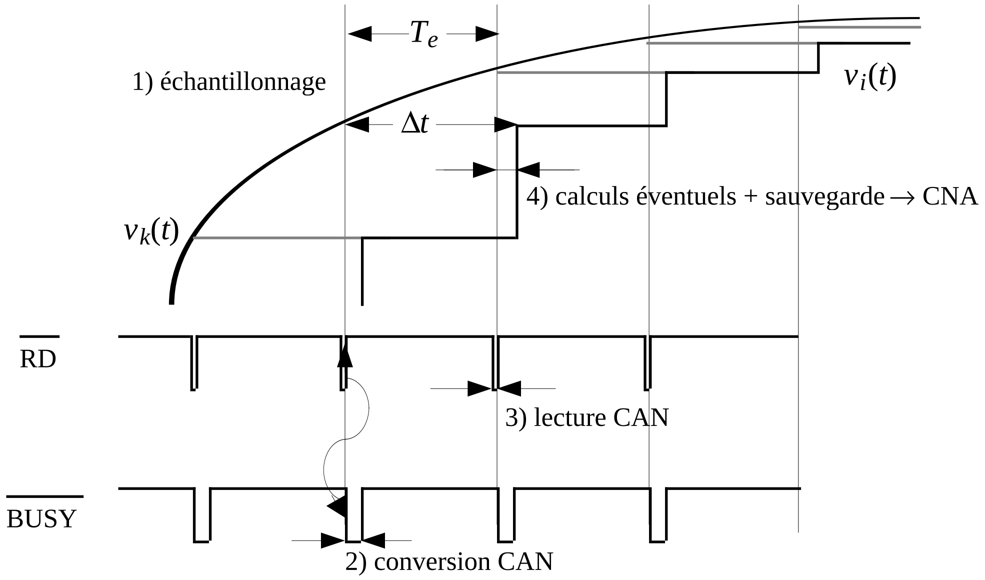
a) Soit T_e = 100µs. Sur le montage et avec le programme précédents, mesurer le retard Δt *(ou "temps d'acquisition")* de v_i par rapport à v_e (méthode : mesure directe de Δt à l'oscilloscope).

Evaluer Δt en fonction de T_e .

Problème : le système numérique introduit un retard Δt non négligeable, supérieur à une période d'échantillonnage. Si le système était inséré dans une boucle, il en découlerait un décalage temporel entre la consigne et la mesure : on ne pourrait parler d'échantillonnage "instantané". Tout se passe comme si on comparait la consigne du jour avec la mesure de la veille !

Dans le programme \$AC73, le sous-programme d'interruption qui effectue l'acquisition du signal par le CAN et le restitue par le CNA est détaillé ci-dessous. Le retard apporté par le système numérique découle du chronogramme de l'acquisition de données .

```
BF 88 02  acqu    STX   drtim1   démarrage décompteur
B6 94 80          LDA   can_bi   transfert CAN -> CNA
B7 95 80          STA   cna_bi
3B                RTI            retour d'interruption
```



Le CAN reçoit sur son entrée $\overline{\mathrm{RD}}$ un zéro logique pendant un cycle machine (1µs) au cours de l'exécution de l'instruction de lecture du CAN (`LDA can_bi`). Or le signal de commande $\overline{\mathrm{RD}}$ (READ) du CAN est tel que *(rappel : voir TP B12-B13 CAN/CNA pour les signaux de contrôles* $\overline{\mathrm{RD}}$ *et* $\overline{\mathrm{BUSY}}$ *)* :

- un front montant de $\overline{\mathrm{RD}}$ déclenche une conversion. Le résultat de cette conversion est mis en mémoire dans le registre de sortie du CAN.

- lorsque $\overline{\mathrm{RD}}$ = 0, le contenu de ce registre (résultat de la conversion précédente) est présent sur le bus de sortie du CAN ($\Rightarrow$ lecture du CAN)

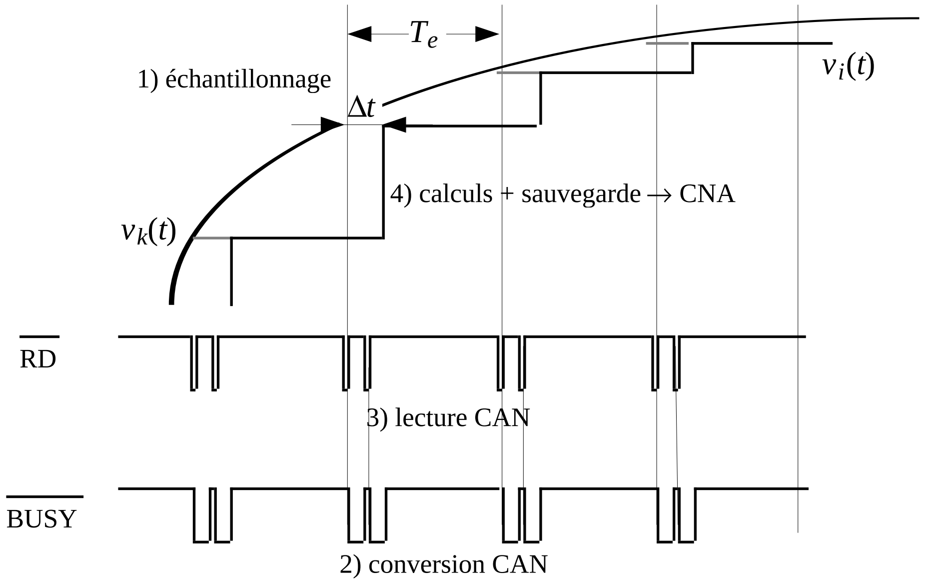
- lorsque $\overline{\mathrm{RD}}$ = 1, la sortie du CAN est déconnectée (état haute impédance)

b) On modifie donc le sous-programme d'acquisition comme suit : une première exécution de l'instruction LOAD (`LDA can_bi`) déclenche une conversion. Après un délai un peu supérieur au temps de conversion du CAN, une deuxième exécution de cette instruction permet de lire le résultat de cette conversion (et en déclenche une seconde, qui reste inutilisée).

Exécuter le programme d'adresse \$AF00, valeur hexadécimale de T_e = 100 µs à l'adresse \$0F00:0F01 comme d'habitude. Evaluer de nouveau le retard Δt en fonction de T_e.

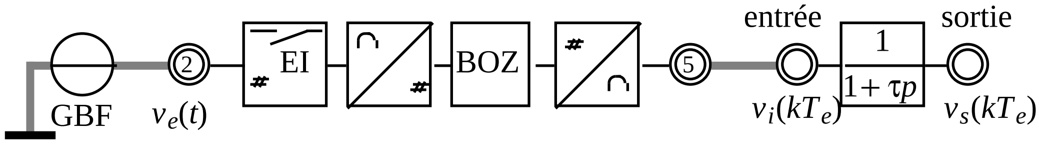
A.N. : si l'on veut que $\Delta t < T_e/100$, avec $a = q$ pour $n = 8$, quelle devraient être les valeurs maximales de F_e et de la fréquence $f_{\max}$ du signal ?

```
BF 88 02  acqbis  STX   drtim1   démarrage décompteur
B6 94 80          LDA   can_bi   lancement conversion       5µs
12                NOP            délai d'attente de
12                NOP            fin de conversion :
12                NOP
12                NOP
12                NOP
12                NOP
12                NOP
12                NOP                                      + 8 x 2
B6 94 80          LDA  can_bi    lecture CAN               + 5
B7 95 80          STA  cna_bi    transfert CAN -> CNA      + 5 = 31 µs
3B                RTI
```



Δt = *temps d'exécution des instructions à vérifier sur la table de programmation du µP 6809 (colonne ~)*

IV- Application : influence de l'échantillonnage sur le comportement en fréquence d'un système du 1er ordre



$v_e(t)$ est maintenant une tension sinusoïdale d'amplitude crête à crête ≤ 20V. On observe $v_s(t)$.

a) Tracer au tableur, la courbe $G(f) = 20\log|H(p)|$, en échelle semi-log, pour $0{,}1F_0 \le f \le 10F_0$.

b) Exécuter le programme $AF00. Soit $T_e = 0{,}7\tau$ puis $0{,}1\tau$. Relever dans chaque cas la courbe de gain du système sur le graphe précédent ($0{,}1F_0 \le f \le 10F_0$). Conclusion.

c) *Calculs théoriques :*

- connaissant $H(p)$, établir l'expression de $H_B(z^{-1})$ *(cf table transf. z §B22).* On pose : $\alpha = e^{-T_e/\tau}$.

- sachant que $z^{-1} = e^{-j\omega T_e} = \cos\omega T_e - j\sin\omega T_e$, montrer que : $\left|H_B(j\omega)\right| = \dfrac{1-\alpha}{\sqrt{1+\alpha^2-2\alpha\cos\omega T_e}}$

- en déduire la fréquence de coupure à –3dB pour $T_e = 0{,}1\tau$.

Rappel : $G = 20\log|H_B| = -3 \Leftrightarrow \left|H_B(j\omega)\right| = \dfrac{1}{\sqrt{2}}$