Intégrateurs analogique et numérique

1. Introduction

L'intégration d'un signal est une opération courante en traitement du signal. Par exemple, le signal délivré par un capteur de vitesse doit être intégré pour fournir la position. De même, le signal délivré par un accéléromètre doit être intégré pour fournir la vitesse.

Pour un signal analogique, l'intégration est définie par :

$$V_s(t) = \frac{1}{\tau} \int_0^t V_e(t')\, dt' \tag{1}$$

où τ est une constante homogène à un temps (éventuellement négative). On vérifie aisément que la fonction de transfert en régime sinusoïdal est :

$$H(\omega) = \frac{1}{j\tau\omega} \tag{2}$$

En régime sinusoïdal, un intégrateur est donc caractérisé par un déphasage de $\pm\frac{\pi}{2}$ et par un gain en décibel décroissant à -20 dB par décade.

Considérons un signal d'entrée périodique dont la série de Fourier s'écrit :

$$V_e(t) = \frac{A_0}{2} + \sum_{n=1}^{\infty} A_n \cos\left(2\pi n f\, t + \phi_n\right) \tag{3}$$

La composante constante $A_0/2$ (appelée *composante continue*) donne en sortie :

$$V_s(t) = \frac{1}{\tau}\frac{A_0}{2} t \tag{4}$$

Tout signal périodique comporte une composante continue, aussi petite soit elle. On voit donc que l'intégrateur idéal produit une dérive de la tension en sortie. Autrement dit, il est instable.

L'objectif de ces TP est la mise en œuvre d'un filtre intégrateur analogique, puis d'un filtre intégrateur numérique. En verra en particulier comment résoudre le problème de la dérive due à la composante continue.

Toutes les observations et les choix effectués devront être reportés dans le compte-rendu écrit. À la fin de chaque partie, un compte rendu oral devra être fait à l'examinateur.

2. Matériel et outils numériques

Matériel utilisé :

- ▷ Circuit intégrateur avec alimentation double +15/0/-15.
- ▷ Générateur de signaux (GBF).
- ▷ Oscilloscope.
- ▷ Petits fils
- ▷ Carte d'acquisition SysamSP5.

▷ Ordinateur et environnement Python.

Outils numériques :

▷ bode.py : script python pour tracer le diagramme de Bode d'une fonction de transfert.

▷ analogiqueNumerique.py : numérisation d'un signal avec la carte Sysam SP5.

▷ traitementSignal.py : fonctions de traitement du signal (analyse spectrale, calcul de filtres, filtrage, réduction).

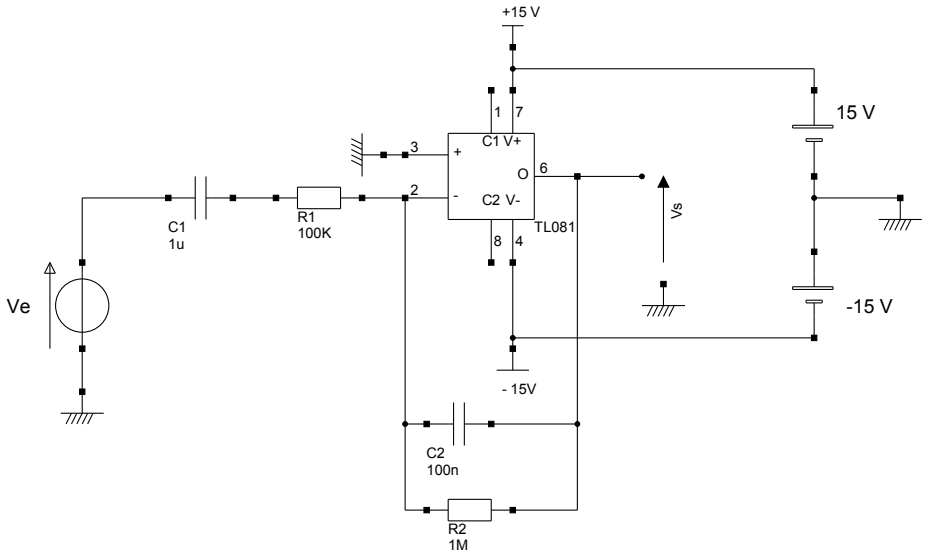
Les scripts `analogiqueNumerique.py` et `traitementSignal.py` contiennent des fonctions à utiliser. Il faut les enregistrer dans le dossier personnel de travail. Pour utiliser les fonctions dans un script python, faire les importations suivantes :

```
from analogiqueNumerique import *
from traitementSignal import *
```

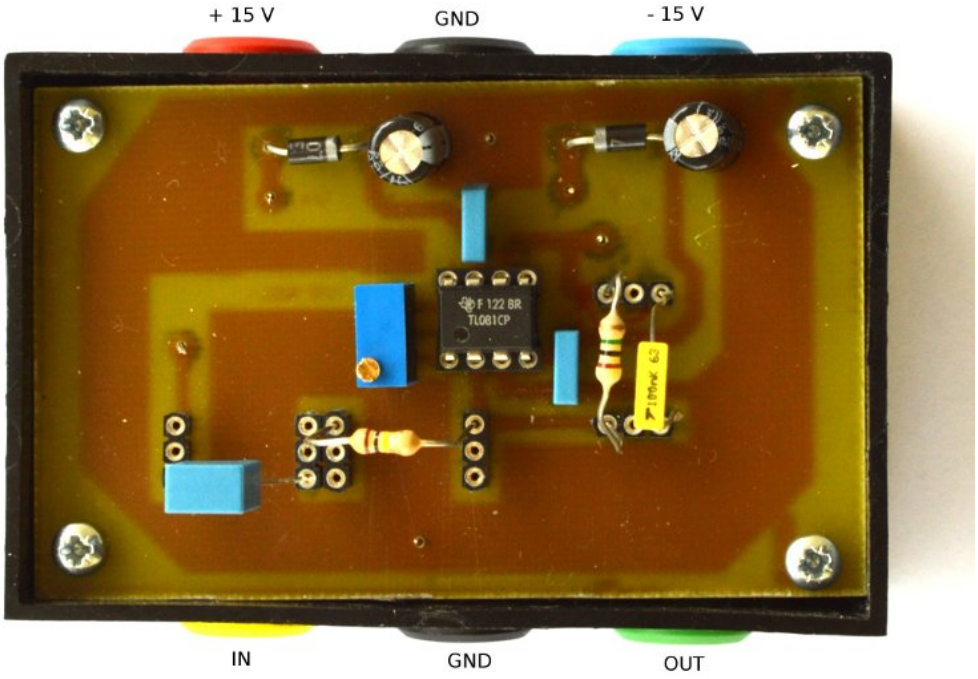
3. Intégrateur analogique

3.a. Circuit

Voici le schéma du filtre intégrateur :



L'amplificateur linéaire intégré (ALI) TL081 est un circuit actif. Le circuit est alimenté par une alimentation double de deux fois 15 volts. La photographie suivante montre la platine utilisée :



Les différentes bornes sont :

- ▷ Les bornes rouge, noire et bleue à brancher sur les bornes $+15\ V$, 0 et $-15\ V$ de l'alimentation double.
- ▷ L'entrée est appliquée entre les bornes jaune et noire. La sortie est obtenue entre les bornes verte et noire.

Les deux condensateurs et les deux résistances peuvent être changées. Les valeurs par défaut sont :

$$\begin{aligned} R_1 &= 100\ k\Omega \\ C_1 &= 1{,}0\ \mu F \\ R_2 &= 1{,}0\ M\Omega \\ C_2 &= 100\ nF \end{aligned}$$

La fonction de transfert du filtre s'écrit comme le produit de deux fonctions de transfert du premier ordre :

$$H(\omega) = H_1(\omega)H_2(\omega) \tag{5}$$

$$H_1(\omega) = \frac{jR_1C_1\omega}{1 + jR_1C_1\omega} \tag{6}$$

$$H_2(\omega) = \frac{-R_2/R_1}{1 + jR_2C_2\omega} \tag{7}$$

3.b. Étude théorique

Quelle est l'opération de filtrage réalisée par H_1 ? Même question pour H_2.
Tracer le diagramme de Bode avec les valeurs données. De quel type de filtre s'agit-il ?
Quel est le domaine de fréquence où le filtre effectue une intégration ?
Quelle est l'utilité de la partie passe-haut du filtrage ?

3.c. Étude expérimentale

L'étude se fait avec un générateur de signaux et un oscilloscope. On utilisera un signal sinusoïdal et un signal en créneau. On ne dépassera pas la fréquence de 1 kHz.

Déterminer expérimentalement le domaine de fréquence où l'intégration est réalisée.
Quelle est l'influence d'un décalage de tension (composante continue) appliqué en entrée ? Expliquer l'observation à l'aide du diagramme de Bode.
Que se passe-t-il si le condensateur C_1 est remplacé par un court-circuit ? Expliquer.

3.d. Modification du filtre

Comment faut-il modifier les valeurs des résistances pour que la fréquence minimale d'intégration soit 10 fois plus faible ?

4. Intégration numérique

4.a. Intégrateur élémentaire

L'intégrateur numérique est défini par la relation de récurrence suivante :

$$y_n = y_{n-1} + \beta(x_n + x_{n-1}) \tag{8}$$

où x_n est l'entrée du filtre, y_n la sortie, et β un coefficient ajustable en fonction du gain souhaité.

Il s'agit d'un cas particulier de filtre récursif, à réponse impulsionnelle infinie. Le principe du filtrage récursif est présenté en annexe.

Tracer le diagramme de Bode de ce filtre, avec la fonction `reponse_freq_filtre_bode` du script traitementSignal.py. On pourra dans un premier temps choisir $\beta = 0,01$.
Pour une fréquence d'échantillonnage de 2 kHz, quel est le domaine de fréquence où l'intégration est réalisée ?
Quelle est l'influence du paramètre β ?

Pour mettre en œuvre ce filtre, on commence par numériser un signal créneau avec la carte Sysam SP5. On utilisera pour cela la fonction `numerisation_EA0` du script analogiqueNumerique.py. La fréquence d'échantillonnage est fixée à $f_e = 2\ kHz$. Les

fonctions de traitement seront ajoutées à la suite de la numérisation. On pourra ainsi faire varier la fréquence du signal et observer l'effet du filtre en exécutant le script. Pour le calcul du signal filtré, on pourra programmer une boucle afin d'appliquer soit la réalisation directe de type 1, soit la réalisation directe de type 2 (voir annexe). Pour la réalisation de type 2, on peut utiliser la fonction `filtrage` fournie.

Faire la numérisation d'un signal en créneau d'environ 10 Hz et calculer le signal filtré. Tracer les deux signaux sur la même figure.
Quel est l'inconvénient de ce filtre ?
Pourquoi le filtre analogique étudié précedemment n'avait pas ce défaut ?

4.b. Filtre récursif passe-haut

Pour éliminer le phénomène de dérive, on considère un filtre passe-bande récursif (filtre RII). On utilisera la fonction `filtre_RII_passe_bande` fournie pour calculer les coefficients du filtre. Afin d'obtenir un comportement intégrateur dans la partie passe-bas, il faut un filtre d'ordre 1. Deux fréquences de coupure doivent être données pour définir un filtre passe-bande. On pourra par exemple choisir les fréquences réduites $f_{c1} = 0,0005$ et $f_{c2} = 0,001$.

Calculer les coefficients du filtre passe-bande et tracer son diagramme de Bode théorique. Combien a-t-il de coefficients a_k et de coefficients b_k ?
Numériser un signal créneau et lui appliquer le filtre passe-bande. Si nécessaire, multiplier les coefficients b_k par un facteur supérieur à 1 pour augmenter le gain.

Pour faciliter le test du filtre à différentes fréquences, on utilisera le script acquisitionSansFinFiltrage.py, qui fait un filtrage en flux continu en traçant les signaux. Le script devra être complété pour définir les coefficients du filtre.

Déterminer la plage de fréquence pour laquelle le filtre réalise l'intégration.
Que se passe-t-il si un décalage de tension (composante continue) est ajoutée à l'entrée ?
Modifier le filtre pour que la fréquence basse de cette plage soit abaissée. Quel est l'inconvénient de ce changement ?
Que faut-il faire pour intégrer un signal de fréquence 1 kHz ?

5. Annexe : filtrage numérique

Soit un signal numérique constitué par la suite de nombres x_n, échantillonnés à la fréquence f_e. D'une manière générale, un filtre linéaire est défini par la relation de récurrence suivante :

$$y_n = \sum_{k=0}^{N-1} b_k x_{n-k} - \sum_{k=1}^{M-1} a_k y_{n-k} \tag{9}$$

La suite y_n est le signal de sortie du filtre. Il est donc défini par M coefficients a_k et N coefficients b_k. Par convention, on a toujours $a_0 = 1$. Lorsque tous les autres a_k sont nuls,

le filtre a une réponse impulsionnelle finie (filtre RIF) ; les coefficients b_k définissent alors la réponse impulsionnelle. Lorsque certains coefficients a_k sont non nuls (pour $k > 0$), le filtre est dit *récursif* car la sortie à l'instant n dépend de manière récursive de la sortie aux instants antérieurs. Un tel filtre a en général une réponse impulsionnelle infinie (filtre RII).

Selon l'application visée et la sélectivité souhaitée, on utilisera un filtre RIF ou un filtre RII. Les filtres RIF peuvent être très sélectifs avec un grand nombre de coefficients. Les filtres RII sont plus efficaces lorsque le nombre de coefficients doit être petit (inférieur à 10), mais ils peuvent être instables.

La réponse fréquentielle d'un filtre (gain et déphasage) est obtenue avec sa fonction de transfert en Z :

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots}{1 + a_1 z^{-1} + a_2 z^{-2} + \cdots} \tag{10}$$

en posant :

$$z = e^{j2\pi \frac{f}{f_e}} \tag{11}$$

Le rapport f/f_e doit être inférieur à $1/2$ (condition de Nyquist-Shannon).

Le fichier traitementSignal.py contient deux fonctions permettant de calculer le gain et le déphasage :

- `(f,g,phi)=reponse_freq_filtre(a,b)`
- `(f,log_f,g_db,phi)=reponse_freq_filtre_bode(a,b,fmin)`

Ces deux fonctions prennent en argument les listes des coefficients sous forme de tableaux numpy. La fréquence calculée est en fait le rapport de la fréquence réelle sur la fréquence d'échantillonnage. La première fonction répartit les fréquences de manière linéaire sur l'intervalle $[0, 1/2]$. La seconde les répartit de manière logarithmique, c'est pourquoi il faut lui fournir la fréquence minimale `fmin`.

Le calcul de la sortie par la relation (9) nécessite de fixer les valeurs initiales des $M - 1$ premières valeurs de y_n. On choisit généralement des valeurs initiales nulles.

Le calcul directe par la relation (9) est appelé *réalisation directe de type 1*. Une autre manière de calculer la sortie est d'utiliser un signal intermédiaire w_n défini par :

$$w_n = x_n - a_1 w_{n-1} - a_2 w_{n-2} - \cdots - a_{M-1} w_{n-M+1} \tag{12}$$

La sortie est alors :

$$y_n = b_0 w_n + b_1 w_{n-1} + b_2 w_{n-2} + \cdots + b_{N-1} w_{n-N+1} \tag{13}$$

On montre que ces deux relations sont équivalentes à la relation (9) (en régime stationnaire). La condition initiale consiste à fixer les $M - 1$ premières valeurs de w_k, en général à zéro. Ces deux relations définissent la *réalisation directe de type 2*. La fonction `filtrage` effectue le calcul par cette méthode.