

Project: Conception du filtre de sortie d'un amplificateur numerique

Nov-Dec 2024

Introduction

Les audiophiles savent bien qu'un amplificateur n'en est pas un autre. Les bons amplis audio sont caractérisés par une classe (le plus souvent A ou B), qui correspond à divers critères de qualité. Un de ces critères de qualité est le *taux de distorsion harmonique* (TDH), qui caractérise la linéarité de l'amplificateur : un ampli imparfaitement linéaire auquel on présente un signal parfaitement sinusoïdal génère des harmoniques.

La classe A : elle traite l'entièreté du signal audio, alternances positives comme négatives. Elle est considérée comme étant la meilleure du point de vue musical, mais son très faible rendement provoque un échauffement considérable et limite la puissance disponible en sortie.

La classe B : elle traite séparément deux alternances via deux transistors, l'un étant activé lorsque l'autre est bloquant, ce qui cause des problèmes de raccordement pour les passages par zéro, c'est-à-dire lors de la commutation des deux transistors. En effet, la coordination de ces commutations est extrêmement difficile à réaliser. Ce problème de raccordement crée une distorsion de la forme d'onde, ce qui se traduit par l'apparition d'harmoniques non désirées. Par contre, son rendement est de loin meilleur que celui de la classe A.

La classe AB : elle est fort proche de la classe B, mais fait partiellement travailler chacun des deux transistors sur l'alternance traitée par l'autre (grâce à une polarisation de faible valeur), ce qui permet de raccorder les deux parties plus correctement que dans la classe B.

A l'époque, en 2004, on a vu apparaître des amplis dits "numériques", auxquels on a associé la classe D, et qui présentent de très bonnes caractéristiques de linéarité. Ce type d'amplification offre de nombreux avantages tels qu'une température de fonctionnement extrêmement basse pour des puissances disponibles importantes, un encombrement réduit, et donc un large

domaine d'utilisation (appareil auditif, lecteur mp3, ...). Une représentation des classes B et D est donnée à la Figure 1.

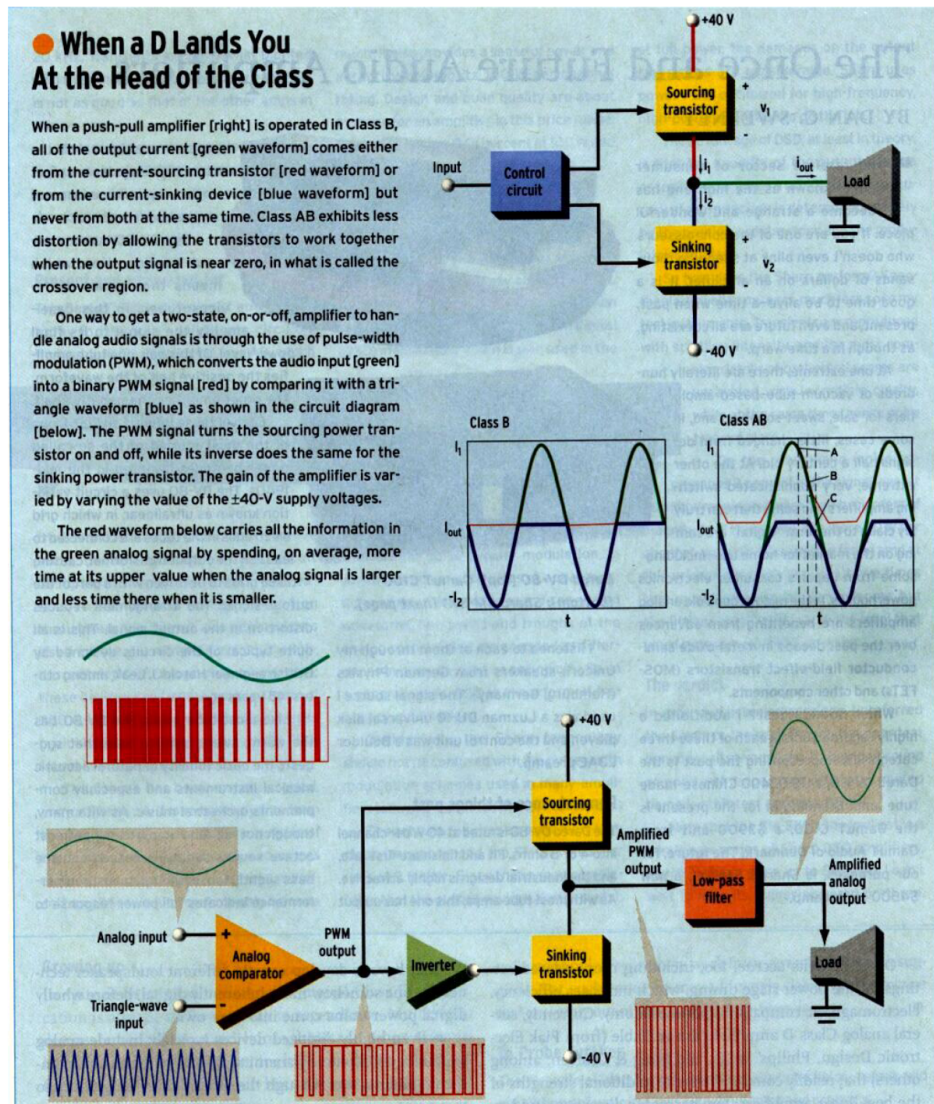


FIGURE 1 – Schéma de principe d'un ampli audio (d'après [?])

Le but de ce projet est de comprendre le fonctionnement de principe d'un amplificateur audio numérique de classe D basé sur la modulation PWM¹ et d'en simuler le fonctionnement de base sur Python, et de concevoir le filtre analogique final.

1. Pulse Width Modulation

1 Principe de fonctionnement

Dans un ampli classique, le signal analogique (ici supposé être une sinusoïde pour simplifier) est amplifié par deux transistors (ou, en pratique, par des circuits plus complexes assimilables à des transistors), en charge respectivement des flancs positifs en négatifs de l'entrée. Chaque transistor fournit un courant de sortie proportionnel à son courant de base. Le problème majeur est celui de l'action plus ou moins coordonnée de ces transistors : au passages par zéro, le premier doit "s'éteindre" et laisser "s'allumer" l'autre. En pratique, la coordination parfaite est très difficile à assurer, ce qui provoque une distorsion de la forme d'onde et fait donc apparaître des harmoniques.

Dans un ampli numérique de classe D, le signal d'entrée est tout d'abord utilisé pour moduler la largeur d'un train d'impulsions rectangulaires correspondant à la PWM. Cette technique consiste à comparer le signal d'entrée à un signal triangulaire. La sortie du comparateur est un signal rectangulaire dont la fréquence est liée à la fréquence du signal d'entrée. L'information est donc bien conservée. Cette opération transforme ainsi immédiatement le signal continu d'entrée en un signal discret à 2 valeurs : $+V$ et $-V$. L'amplification est alors réalisée à l'aide de deux transistors en charge des tensions $V+$ et $V-$ respectivement. La zone de transition n'existe plus : lorsqu'un transistor fonctionne, on est certain que l'autre est bloqué, et réciproquement. Il reste évidemment à reconstituer, après amplification, la sinusoïde cachée dans le signal PWM. Cette opération se fait par filtrage du signal PWM.

1.1 Modulation PWM

La modulation PWM peut être réalisée par simple comparaison entre le signal d'entrée et un signal triangulaire.

1. Vérifiez-le conceptuellement sur un graphique réalisé sous Python (utilisez pour cela une sinusoïde de 0,1 Hz d'amplitude unitaire et une onde triangulaire de 1 Hz d'amplitude égale à 1,2 V).
2. Quelle est (qualitativement) l'influence de la fréquence du signal triangulaire sur le résultat ?
3. Quelle est (qualitativement) l'influence de l'amplitude du signal triangulaire sur le résultat ?
4. Créez à l'aide de Python un signal PWM obtenu par comparaison de 5 ms d'une onde triangulaire de fréquence 20 000 Hz avec une sinusoïde à 1 000 Hz. On choisira un fréquence d'échantillonnage F_e de 1 MHz, afin d'avoir une représentation assez fidèle du signal modulé.
5. Que se passe-t-il si on diminue la fréquence d'échantillonnage (essayez $F_e/2$, $F_e/3$ et $F_e/10$) ?
6. En dessous de quelle fréquence d'échantillonnage est-on certain que la suite d'échantillons ne donnera plus une image fidèle du signal PWM ?

1.2 Réponse d'un circuit non-linéaire en régime périodique - Taux de distorsion harmonique (TDH)

Un système non-linéaire est (entre autres choses) caractérisé par le fait que, lorsqu'on lui soumet un signal $x_{T0}(t)$ sinusoïdal pur en entrée, le signal de sortie $y_{T0}(t)$ contient des composantes à des fréquences harmoniques de la fréquence du signal d'entrée ($f_k = k.f_0$), créées par le système. Ces harmoniques additionnelles résultent de la *distorsion* du signal d'entrée.

Le *taux global de distorsion harmonique* (TDH) est une grandeur permettant de caractériser la linéarité ou la non-linéarité d'un système. Il est défini par le rapport entre la valeur efficace du signal de sortie dont on a éliminé le fondamental et la valeur efficace du signal de sortie complet :

$$\begin{aligned} TDH &= \frac{\sqrt{(\sum_{k=-\infty}^{\infty} |Y_{k.f_0}|^2) - (|Y_{f_0}|^2 + |Y_{-f_0}|^2)}}{\sqrt{\sum_{k=-\infty}^{\infty} |Y_{k.f_0}|^2}} \\ &= \sqrt{1 - \frac{(|Y_{f_0}|^2 + |Y_{-f_0}|^2)}{\sum_{k=-\infty}^{\infty} |Y_{k.f_0}|^2}} < 1 \end{aligned} \quad (1)$$

où $|Y_{k.f_0}|$ est la valeur absolue de la transformée de Fourier du signal $y_{T0}(t)$ en la fréquence $f_k = k.f_0$.

Exemple

Si on introduit un signal $x(t) = \cos(\omega_0 t)$ à l'entrée d'un système non-linéaire très simple caractérisé par la relation $y(t) = x(t) + x^2(t)$, il vient immédiatement (cf. Figure 2) :

$$y(t) = \cos(\omega_0 t) + \cos^2(\omega_0 t) = \frac{1}{2} + \cos(\omega_0 t) + \frac{\cos(2\omega_0 t)}{2} \quad (2)$$

On en déduit que le *taux de distorsion harmonique* du système est de :

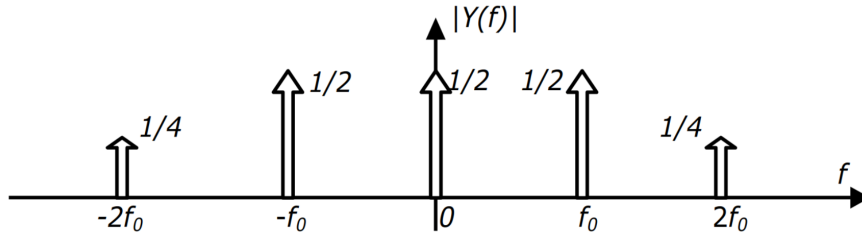


FIGURE 2 – Transformée de Fourier du signal de sortie $y(t)$

$$TDH = \sqrt{1 - \frac{1/4 + 1/4}{1/16 + 1/4 + 1/4 + 1/4 + 1/16}} = \sqrt{\frac{6}{14}} \quad (3)$$

1.3 Amplification

1. En supposant que les transistors amplificateurs commutent très rapidement, simuler sous Python l'amplification du signal utile précédent (cf. section 1.1, question 4) par un facteur 100.
2. Utilisez la fonction `freqz(x, 1, 213)`² (qui calcule la Transformée de Fourier du signal `x`) pour visualiser la transformée de Fourier du signal PWM après amplification.
3. Retrouve-t-on le signal de départ (amplifié) dans la transformée du signal PWM (amplifié) ?

Le signal utilisé ici est échantillonné. Rappelons dès lors que l'estimation de la transformée de Fourier (calculée par `freqz`) entre $f = 0$ et $f = F_e/2$ ne donne une bonne image de la transformée du signal analogique sous-jacente que si la fréquence d'échantillonnage est supérieure à 2 fois la plus grande fréquence contenue dans le signal analogique de départ (théorème de Shannon). Or ici, le signal PWM amplifié est un signal rectangulaire dont le spectre s'étend théoriquement jusqu'à l'infini. Cette condition n'est donc pas strictement respectée. En pratique cependant, les composantes spectrales du signal PWM au-delà de $F_e/2$ sont inférieures à 1% de leur valeur en $F = 0$. On admet donc que la condition est à peu près vérifiée.

1.4 Filtrage

1. Quel type de filtre permettrait de retrouver la sinusoïde de départ, et, qualitativement, quelles en seraient les spécifications (A_p , A_s , Ω_p , Ω_s) ? Notez que nous sommes dans le cas d'un amplificateur de son.
2. Que peut-on dire (qualitativement) de l'évolution du TDH pour la modulation PWM du sinus considéré plus haut ?
3. Toujours qualitativement, quel ordre de grandeur attendons-nous du TDH pour l'ampli complet (avec filtrage final) ?

N.B : Effet de la fréquence de la sinusoïde : On constate en pratique que les raies adjacentes à la raie de l'onde triangulaire à 20 kHz sont un «*miroir doublé*» de la raie de la sinusoïde : si celle-ci passe à 2 000 Hz, la raie à gauche de 18 kHz à 16 kHz, et idem pour celle de droite. Si on donne la fréquence $f = 5\,000$ Hz, on aura une raie de gauche à 10 kHz. Stricto sensu, le filtre devrait avoir une bande passante de 10 kHz, ou il faudrait modifier le signal triangulaire pour le mettre à 40 kHz afin de garantir la conservation des fréquences utiles de 20 à 20000Hz.

2. Importez tout d'abord la librairie `SCIPY.SIGNAL` dans laquelle se trouve la fonction.

2 Conception du filtre de sortie

Il s'agit maintenant de réaliser l'approximation et la synthèse du filtre de sortie dont on a imposé les spécifications à la section précédente. On utilisera la technologie RCAO (filtre RC + amplificateur opérationnel)³.

1. A l'aide de Python⁴, réalisez l'approximation du filtre (pour les approximations de Butterworth, Chebychev I et Cauer). Pour chaque type, donnez le degré du filtre⁵ qui en résulte, la fonction de transfert opérationnelle, visualiser la position des pôles et des zéros ainsi que la réponse en fréquence (à l'aide de `SCIPY.SIGNAL.FREQS`) et vérifier que cette dernière respecte bien les spécifications.
2. En fonction des résultats, faites un choix sur l'approximation à retenir.
3. Réalisez la synthèse en cascade de cellules du second degré RCAO du filtre passe-bas normalisé retenu. Cela nécessite de réfléchir dans l'ordre aux groupements pôles-zéros, à la répartition de la constante K, au choix du type de cellules, à l'ordre dans lequel on les place et, pour la cellule d'ordre le plus bas, au calcul de ses paramètres.
4. Vérifiez sous Python la réponse en fréquence de chaque cellule isolément, ainsi que la réponse en fréquence du filtre global.

3. Ceci est un exercice d'école : vu les niveaux de tension en sortie de l'ampli, les amplis op classiques vont vite saturer. En pratique, on utiliserait des composants d'électronique de puissance.

4. cf. les fonctions `BUTTORD`, `CHEBLORD`, `ELLIPORD` de la librairie `SCIPY.SIGNAL`, en précisant le paramètre `analog=True`.

5. En pratique, pour éviter de rendre les points suivants trop fastidieux, on reverra éventuellement les spécifications pour ne pas dépasser un ordre 5 **dans le meilleur des cas**.