

Réalisation d'un synthétiseur

But de la manipulation

L'objectif principal de cette manipulation est de réaliser un système numérique relativement complexe : un synthétiseur.

Pour ce faire, il sera nécessaire de comprendre le fonctionnement d'un convertisseur analogique-numérique ainsi que d'un convertisseur numérique-analogique

Prérequis

Avant d'entrer au laboratoire, il est conseillé de lire les sections 1 à 4 du document *Programmation d'une carte à microcontrôleur*

Il vous est demandé de répondre aux prédéterminations.

Objectifs

A la fin de ce laboratoire, vous devez être capable:

- De réaliser un programme sur un microcontrôleur.
- D'expliquer le fonctionnement d'un convertisseur analogique-numérique.
- De donner l'utilité d'une interruption.
- De lier un cahier des charges aux périphériques d'un microcontrôleur.

Interfaces analogiques d'un système à μ C

Introduction

Les microprocesseurs sont présents dans la plupart des processus industriels (contrôle de la température, affichage d'une vitesse, systèmes d'alerte) ou grand public (systèmes multimédias, électroménagers, brosse à dents électrique, domotique, ...).

Ces processus interagissent souvent avec des grandeurs analogiques. Un processeur étant un être binaire, il est nécessaire de transformer les grandeurs physiques analogiques en nombres binaires et inversement.

La transformation du physique vers le numérique se fait en deux étapes :

- Transformation de la grandeur physique à mesurer (température, vitesse, ...) en tension. C'est le rôle du capteur (thermocouple, accéléromètre,...).
- Conversion de cette tension en grandeur binaire, C'est le rôle du convertisseur analogique-numérique (CAN ou ADC).

Le chemin inverse permet de transformer une grandeur numérique en une grandeur physique :

- Le convertisseur numérique-analogique (CNA ou DAC) permet de transformer un nombre binaire en une tension.
- L'actionneur transforme la tension en une autre grandeur permettant d'actionner un système (ex : four à induction, moteur électrique, vannes, ...)

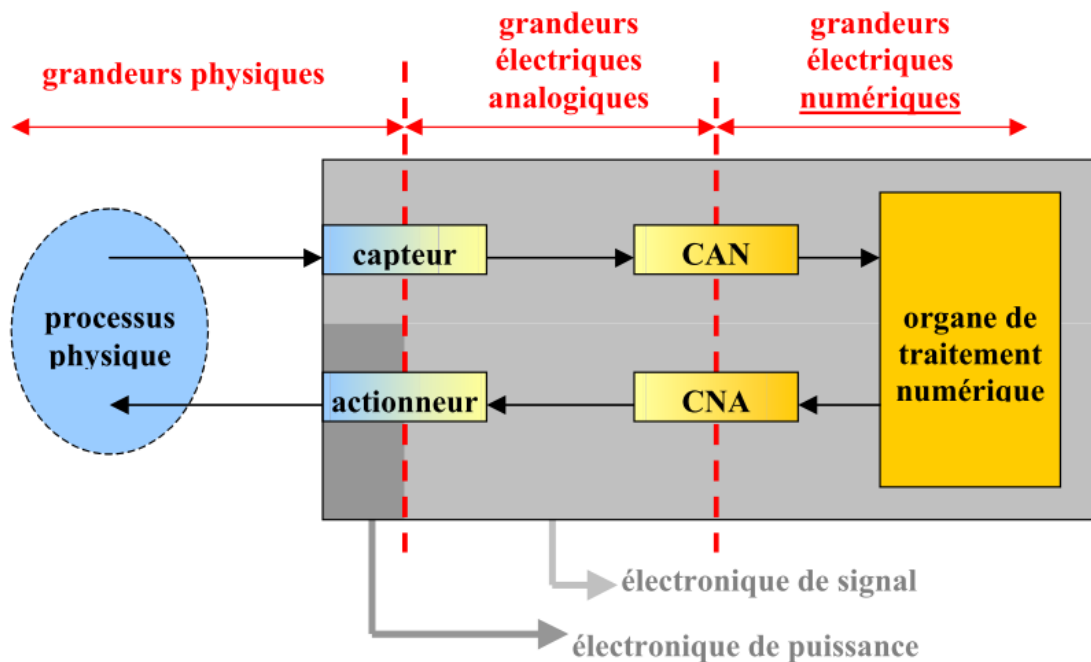


Figure 1 : Architecture d'un processus

Conversion analogique-numérique

Le convertisseur analogique-numérique a pour rôle de transformer une tension analogique (évoluant de manière continue) en un nombre binaire codé sur un nombre de bits définis (10 ou 12 pour le dsPIC). Pour ce faire, le convertisseur réalise une double quantification :

- Quantification dans le temps : c'est l'échantillonnage
- Quantification en niveau.

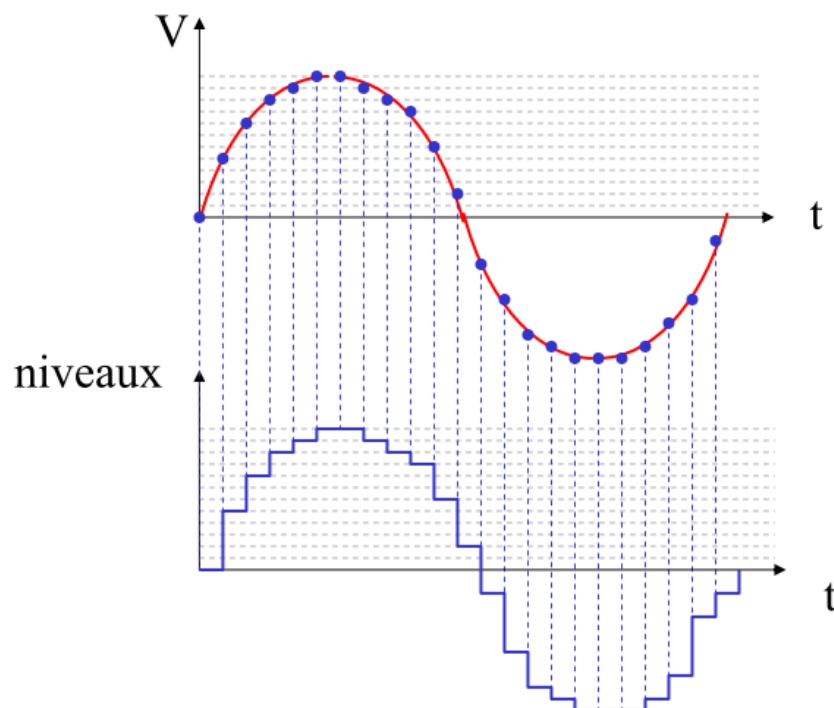


Figure 2 : Illustration de la double quantification

Le rendu d'un convertisseur numérique-analogique dépend donc essentiellement de deux paramètres :

- La fréquence d'échantillonnage.

- La résolution, c'est-à-dire le nombre de bits N codant la grandeur convertie, qui peut également être définie comme étant la plus petite variation de tension détectable : si le convertisseur a une plage de fonctionnement de 0V à 5V et convertit la grandeur sur 10 bits, on a :

$$1 \text{ LSB} = \frac{5V}{2^{10}} = 4.9mV$$

1 LSB (Least Significant Bit) est la plus petite variation de tension détectable par le CAN.

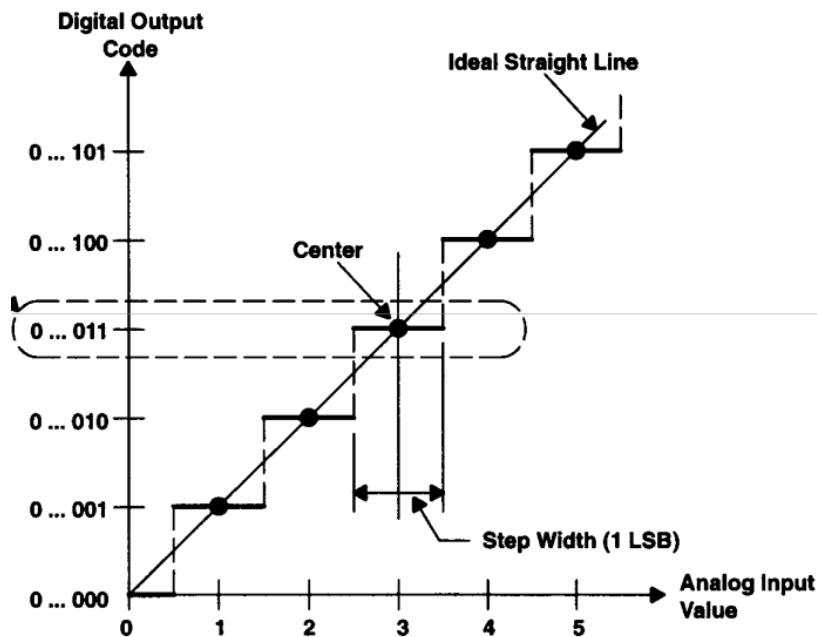


Figure 3 : Fonction de transfert d'un ADC

Conversion numérique-analogique

Le principe d'un convertisseur numérique-analogique est l'inverse de celui d'un CAN : il transforme un nombre codé sur N bits en une tension.

La tension obtenue est continue dans le temps (le convertisseur garde sa sortie constante tant qu'une nouvelle conversion n'est pas effectuée), mais elle est toujours quantifiée en niveau car seules 2^N tensions différentes sont réalisables.

Prédéterminations

- Calculez la résolution du CAN se trouvant sur la carte Explorer16 sachant que la plage de tension d'entrée est de 0→3,3V et qu'il convertit sur 12 bits.
- Déterminez l'espace mémoire nécessaire au stockage d'un fichier sonore de 3s, échantillonné à 11kHz, en sachant que chaque échantillon est stocké sur un octet.

Manipulation

Réalisation d'un voltmètre

Dans un premier, vous allez vous familiariser avec l'utilisation du convertisseur analogique-numérique à l'aide d'une application simple : un voltmètre mesurant la tension à la sortie d'un potentiomètre connecté à l'entrée AN0. Un retour visuel se fera en utilisant les LED comme *bargraph* (plus la tension est élevée, plus le nombre de LED allumées doit être élevé).

Le *bargraph* devra être rafraîchi à une cadence de 1kHz.

- Tracez le schéma-bloc complet de votre système.
 - Quels périphériques allez-vous utiliser ?
 - Différenciez clairement les périphériques internes et les interfaces du μC .
- Dessinez le diagramme de votre algorithme.

- Ecrivez le code de votre voltmètre en vous aidant des fichiers *ADC.h* et *ADC.c* qui contiennent les fonctions nécessaires à l'utilisation de l'ADC1 pour convertir l'entrée AN0, en utilisant le polling.

Votre code utilise un double polling (sur le Timer et sur l'ADC). Pour éviter cela, il est possible de configurer l'ADC pour que le débordement du Timer lance automatiquement la conversion, en modifiant le champ *AD1CON1bits.SSRC* (cf. Guide de programmation).

- En vous basant sur la fonction *adcPollingInit()*, écrivez une fonction *adcTimerInit()* qui configure l'ADC1 dans le mode décrit ci-dessus et adaptez votre code en conséquence.
- Quelle technique respecte le mieux la période d'échantillonnage ?

Réalisation d'un synthétiseur

Un synthétiseur est un appareil capable de générer une onde sonore arbitraire et de faire jouer ce son par des baffles. Pour cela, le μ C doit envoyer un à un les échantillons correspondant à la forme d'onde voulue vers le CNA dont la sortie est connectée à l'amplificateur audio de la carte d'extension.

Dans notre cas, nous allons nous contenter d'un simple sinus. Votre programme devra :

- Créer un vecteur de *int* contenant une période complète de sinus d'amplitude 2000. Pour info, vous devez inclure *math.h* dans votre fichier *main* pour utiliser la fonction *sin()*.
- Envoyer un échantillon du sinus toutes les 50 μ s au CNA, en utilisant les fonctions de *audio.h*.

Amélioration

On veut pouvoir démarrer et arrêter le signal du synthétiseur en utilisant le bouton A du clavier de la carte d'extension.

- Ajoutez cette fonctionnalité à votre synthétiseur.
- Pour quelle raison ne fonctionne-t-il plus correctement ?
- Modifiez votre programme pour que l'envoi des échantillons au CNA soit réalisé dans la routine d'interruption du Timer.
- En quoi cela corrige-t-il le problème ?

Améliorations supplémentaires

On peut ajouter d'autres fonctionnalités à notre synthétiseur :

- Réglage de l'amplitude du signal au moyen du potentiomètre.
- Modification de la fréquence du signal via le clavier.