

## 5. Etude d'un Convertisseur Analogique Numérique - CAN

### OBJECTIFS :

- Comprendre le rôle primordial d'un convertisseur analogique numérique en électronique,
- Mesurer les caractéristiques du CAN,
- Apprendre à utiliser le CAN à l'aide d'un microcontrôleur.

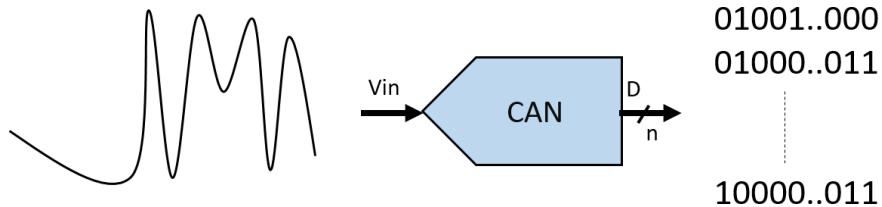


Figure 16 : Convertisseur analogique numérique  $n$  bits

Le convertisseur analogique numérique permet de transformer une tension analogique  $V_{in}$  en une tension numérique représentée par une donnée  $D$  sur  $n$  bits. (Figures 16 et 17).

Une tension analogique présente une infinité de valeur en amplitude et en temps.

Une tension dite numérique est constituée d'un nombre limité de données en amplitude et en temps. Chaque donnée est appelée échantillon.

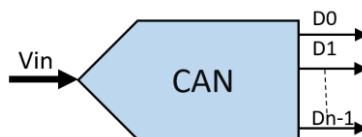


Figure 17 : Convertisseur analogique numérique  $n$  bits

Le CAN présent sur la carte est un convertisseur 8 bits. Il est contrôlé par le microcontrôleur.

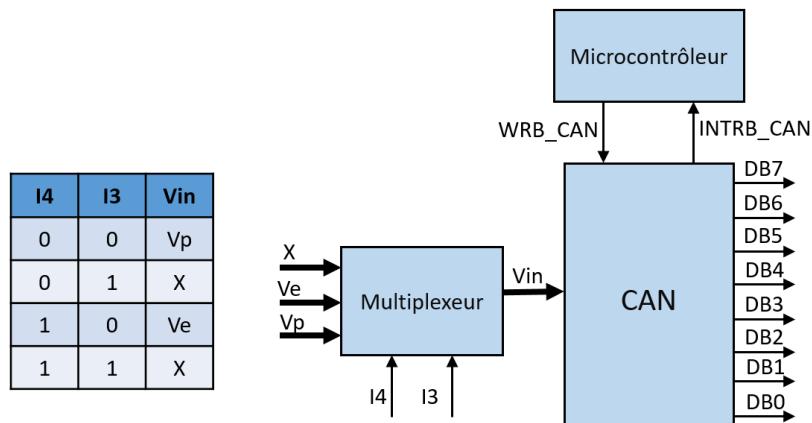


Figure 18 : Convertisseur analogique numérique 8 bits

Ce CAN peut convertir 3 tensions différentes : X, Ve et VP, selon la position des 2 interrupteurs I3 et I4.

Le signal WRB\_CAN est l'ordre de conversion générée par le microcontrôleur. C'est une entrée du CAN.

Le signal INTRB\_CAN est l'information de fin de conversion générée par le CAN. C'est une sortie du CAN.

Les sorties DBi peuvent prendre uniquement deux valeurs dites logiques : 0 ou 1.

Comme indiqué sur la figure 18 chaque sortie DBi du CAN est connectée à une LED.

Lorsque DBi = 0, la LEDi est allumée

Lorsque DBi = 1, la LEDi est éteinte

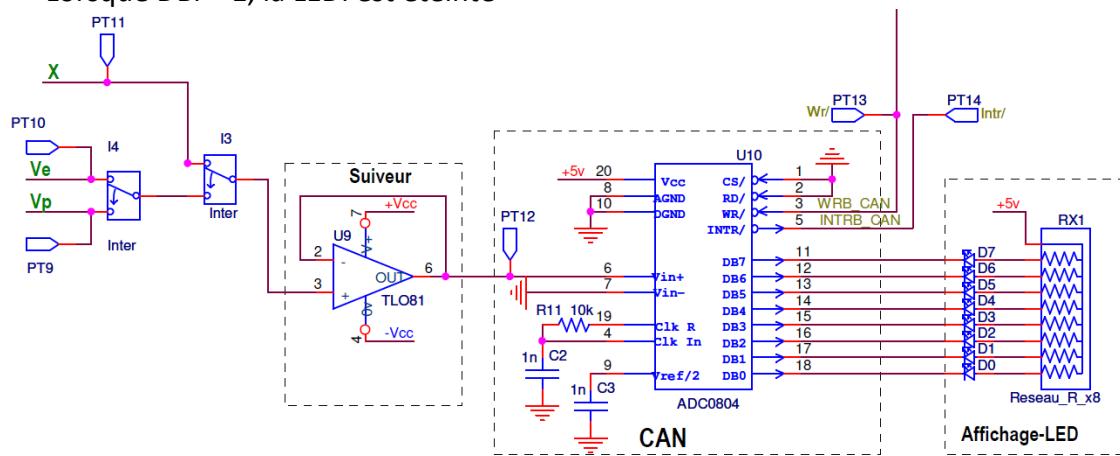


Figure 19 : Schéma comprenant le CAN (carte)

La tension d'entrée analogique du CAN Vin est convertie en une donnée numérique sur 8 bits DB[7:0] suivant l'équation suivante. Vin\_num représente la tension analogique correspondante à la tension Vin numérisée. Chaque donnée numérique DBi sur 1 bit est égale soit à 0 soit à 1.

$$Vin\_num = \sum_{i=0}^{7} \frac{5}{2^{8-i}} \times DBi$$

On peut donc considérer que chaque bit DBi de la donnée sur 8 bits représente une partie de la tension analogique Vin\_num selon le tableau ci-dessous.

Sortie	DB7 (MSB)	DB6	DB5	DB4	DB3	DB2	DB1	DB0 (LSB)
Tension	2,5V	1,25 V	625 mV	312,5 mV	156,25 mV	78,125 mV	39,1 mV	19,5 mV

Tableau 5 : Tableau de conversion du CAN

Vin\_num s'exprime donc de la manière suivante :

$$Vin\_num = DB7 \times 2,5 V + DB6 \times 1,25 V + DB5 \times 625 mV + DB4 \times 312,5 mV + DB3 \times 156,25 mV + DB2 \times 78,125 mV + DB1 \times 39,1 mV + DB0 \times 19,5 mV$$

Avec **DBi** = 1 ou 0

*Exemple :*

Pour une tension d'entrée Vin égale à 3,125V, la sortie numérique constituée de 8 bits est égale à :

DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
1	0	1	0	0	0	0	0

*Tableau 6 : Exemple de conversion*

$$\text{En effet, } \text{Vin} = \text{DB7} \times 2,5 + \text{DB5} \times 0,625 = 3,125\text{V}$$

## PREPARATION

La valeur numérique maximale en sortie du CAN est donc DB<sub>i</sub> = 1 pour i de 0 à 7 (DB = 11111111).

- Calculer la valeur maximale de la tension analogique pouvant être convertie.

Vin max = 4,980675 V
----------------------

La valeur numérique minimale en sortie du CAN correspondant à une tension analogique non nulle est donc DB<sub>0</sub> = 1 et DB<sub>i</sub> = 0 pour i de 1 à 7 (DB = 00000001).

- Calculer la valeur minimale de la tension analogique pouvant être convertie. Cette tension est appelée pas de quantification.

Vin min = 19,5 mV
-------------------

Pour des tensions analogiques d'entrées égales à 0V, 1V, 3V et 4 V en vous aidant des informations précédentes.

- Donner la valeur des 8 bits DB<sub>i</sub>.

Vin (V)	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
0	0	0	0	0	0	0	0	0
1	0	0	1	1	0	0	1	1
3	1	0	0	1	1	0	1	0
4	1	1	0	0	1	1	0	0

- Donner l'état de chacune des diodes : allumée (A)/éteinte (E).

Vin (V)	LED7	LED6	LED5	LED4	LED3	LED2	LED1	LEDO
0	A	A	A	A	A	A	A	A
1	A	A	E	E	A	A	E	E
3	A	A	A	E	E	A	E	A
4	E	E	A	A	E	E	A	A

## MANIPULATION

Comme indiqué sur la figure 18, le circuit CAN est piloté par le microcontrôleur présent sur la carte Arduino (voir figure 6). Ce microcontrôleur est un circuit qui permet d'exécuter une suite d'instructions (un code informatique) pour effectuer un traitement ou générer des signaux.

Dans le cadre de cette manipulation, le code `CAN_V1.ino` va permettre de générer le signal périodique WRB\_CAN. Ce signal WRB\_CAN est connecté en entrée du circuit CAN. Lorsque ce signal passe de l'état haut à l'état bas, le CAN convertit en numérique la tension présente sur son entrée (`Vin` sur la figure 18 et `PT12` sur la figure 9).

Le signal INTRB\_CAN, généré par le circuit CAN indique que la conversion est terminée et que le résultat est disponible sur les sorties DBO à DB7 (figure 18) connectées à des LEDs (figure 19).

2 scénarios différents vont être étudiés avec deux entrées différentes choisies grâce aux 2 interrupteurs, I3 et I4. → Code : `CAN_V1.ino`.

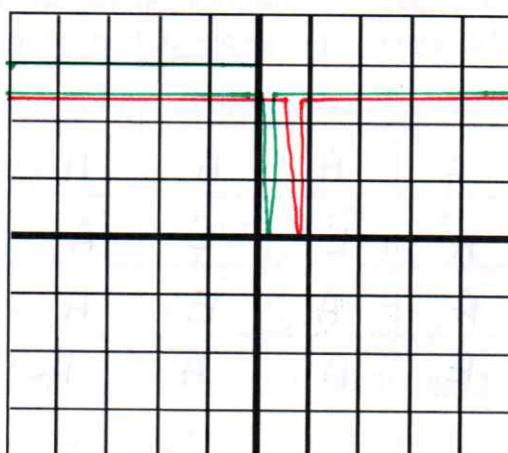
- Une tension ( $V_p$ ) contrôlée par le potentiomètre
- Une tension ( $V_e$ ) externe

### 5.1. Conversion de la tension $V_p$

5.1.1. Programmer le microcontrôleur avec le code `CAN_V1.ino`. Ne pas oublier de « téléverser ». Utiliser l'annexe p. 69 sur la programmation de la carte Arduino en respectant dans l'ordre les étapes A à D.

5.1.2. Observer à l'aide de l'oscilloscope les 2 signaux de contrôle WRB\_CAN (PT13) sur CH1 et INTRB\_CAN (PT14) sur CH2.

5.1.3. Tracer les oscillogrammes correspondants.



Calibre Axe horizontal :  $2V$  /div

Calibre Axe vertical :  $250\mu s$  /div

**Tension WRB\_CAN :**

Tension état bas :  $80mV$  Durée état bas :  $14\mu s$

Tension état haut :  $4,88V$  Durée état haut :  $2\text{ ms}$

Période :  $2\text{ ms}$  Fréquence :  $500\text{ Hz}$

**Tension INTRB\_CAN :**

Tension état bas :  $160mV$  Durée état bas :  $22\mu s$

Tension état haut :  $4,88V$  Durée état haut :  $2\text{ ms}$

Période :  $2\text{ ms}$  Fréquence :  $500\text{ Hz}$

- 5.1.4. A partir des observations précédentes, déduire la période d'échantillonnage et la fréquence d'échantillonnage du CAN.

$$\text{Tech} = 2\text{ ms}$$

$$\text{Fech} = 500\text{ Hz}$$

- 5.1.5. Positionner les interrupteurs I3 et I4 pour sélectionner l'entrée Vp correspondant à la sortie du montage potentiomètre et donner le Point Test correspondant.

$$V_p = PT \quad 9$$

- 5.1.6. Régler le potentiomètre pour obtenir les valeurs de tension d'entrée du CAN à 0V, 1V, 3V et 4V, donner les états des LEDs et comparer à la préparation ?

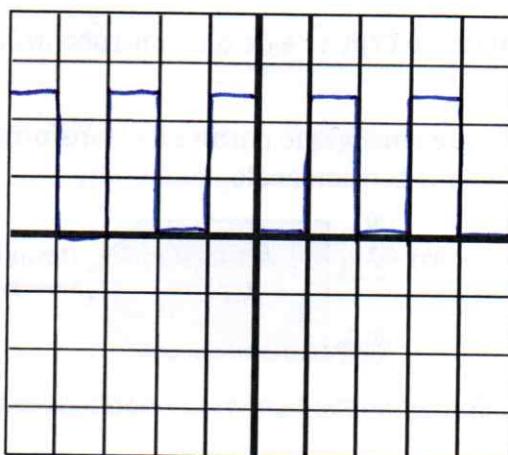
Vin (V)	LED7	LED6	LED5	LED4	LED3	LED2	LED1	LEDO
0	A	A	A	A	A	A	A	A
1	A	A	E	E	A	A	A	A
3	E	A	A	E	A	E	A	A
4	E	E	A	A	A	A	A	A

■ : + par rapport à la préparation

## 5.2. Conversion de la tension Ve

5.2.1. Régler la fréquence du GBF à 1Hz, sélectionner « square » avec le bouton « wave » et relier la sortie « TTL/CMOS output » du GBF sur l'entrée Ve de la carte (ne pas oublier la masse).

5.2.2. Observer avec l'oscilloscope (CH1) la sortie « TTL/CMOS output » du GBF et relever cette tension.



Calibre Axe horizontal : 500 ms /div

Calibre Axe vertical : 2 V /div

5.2.3. Positionner les interrupteurs I3 et I4 pour appliquer la tension Ve à l'entrée du circuit CAN via le circuit suiveur (figure 19). Cette tension Ve correspond à la tension appliquée sur la borne Ve située en haut à gauche de la carte (Figure 7).

- Quels sont les états des LEDs pour l'état haut : 5 V et l'état bas : 0 V ?

état bas 0V : toutes allumées  $\Rightarrow 0000000$

état haut 5V : 3 du bas allumées  $\Rightarrow 1111000$

- Est-ce normal ? Pourquoi ?

$\rightarrow 0V \rightarrow$  aucune tension donc aucune éteinte

$\rightarrow 5V \rightarrow$  toutes les LEDs devraient être éteintes.

5.2.4. Refaire les mêmes observations à l'oscilloscope et sur l'état des LEDs pour des fréquences réglées avec le GBF de 10Hz, 100Hz puis 1kHz.

- Que se passe-t-il quand la fréquence augmente (rappel  $T=1/f$  avec T la période en secondes (s) et f la fréquence en Hertz (Hz)).

10 Hz  $\rightarrow$  période de 100 ms

100 Hz  $\rightarrow$  période de 10 ms

1 kHz  $\rightarrow$  période de 1 ms

} les LEDs clignotent de 0 en 1

## 6. Etude d'un Convertisseur Numérique Analogique - CNA

### OBJECTIFS :

- Comprendre le rôle primordial d'un convertisseur numérique analogique en électronique,
- Mesurer les caractéristiques du CNA,
- Apprendre à utiliser le CNA à l'aide d'un microcontrôleur

Le convertisseur numérique analogique permet de transformer une donnée numérique D représentée sur n bits en une tension analogique Vout.

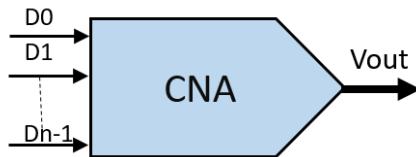


Figure 20 : Convertisseur Numérique Analogique sur n bits

Le CNA présent sur la carte est un convertisseur 8 bits. Il est contrôlé par le microcontrôleur.

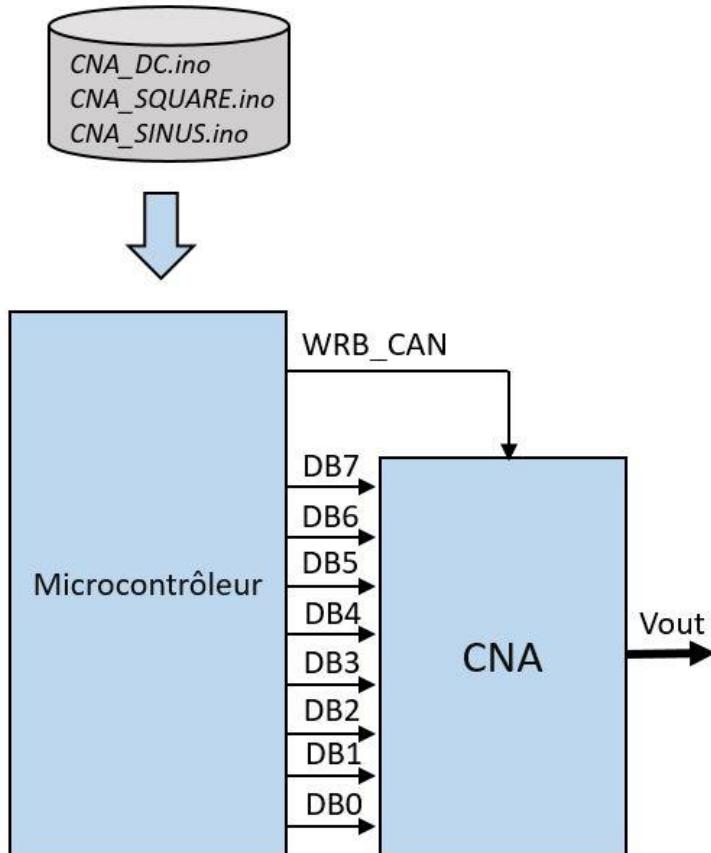


Figure 21 : Convertisseur Numérique Analogique sur 8 bits

Le signal WRB\_CNA est l'ordre de conversion généré par le microcontrôleur. C'est une entrée du CNA.

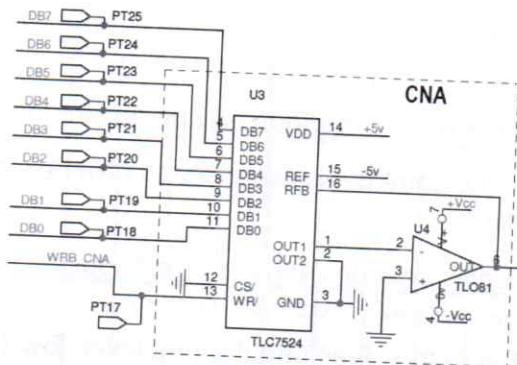


Figure 22 : Schéma du CNA (carte)

La tension de sortie,  $V_{out}$  est proportionnelle aux entrées numériques converties en décimal :

$$V_{out} = -V_{ref} \times D/256$$

$V_{ref}$  est la tension du signal appliquée sur l'entrée REF (dans le cas présent  $V_{ref} = -5V$ ). L'équation suivante permet de convertir une donnée numérique DB sur 8 bits en une donnée D en base 10.

Chaque donnée numérique DB<sub>i</sub> sur 1 bit est égale soit à 0 soit à 1.

$$D = \sum_{i=0}^7 2^i \times DB_i$$

Cette équation peut donc s'exprimer de la manière suivante :

$$D = 128 \times DB_7 + 64 \times DB_6 + 32 \times DB_5 + 16 \times DB_4 + 8 \times DB_3 + 4 \times DB_2 + 2 \times DB_1 + DB_0$$

avec  $DB_i = 1$  ou 0

## PREPARATION

- Quelles sont les valeurs minimale et maximale que le CNA peut convertir ?

Vout min = 0

Vout max = 6,98 V

- Repérer sur le CNA les broches sur lesquelles arrivent les données numériques d'entrée à convertir ainsi que les Points Test correspondants.

Broches : 4  
5  
6  
7

PT: 25  
24  
23  
22

Broches : 8  
9  
10  
11

PT: 21  
20  
19  
18

- Donner les Points Test de la tension de sortie et du signal de contrôle WRB\_CNA (Figure 9).

Vout : PT 26

WRB\_CNA : PT 17

## MANIPULATION

Un CNA convertit un signal d'entrée numérique en une tension de sortie analogique.

Afin de tester le CNA, il est nécessaire de transmettre au CNA, le signal à convertir et un signal pour le contrôler (WRB\_CAN).

Trois scénarios vont être étudiés avec des entrées numériques :

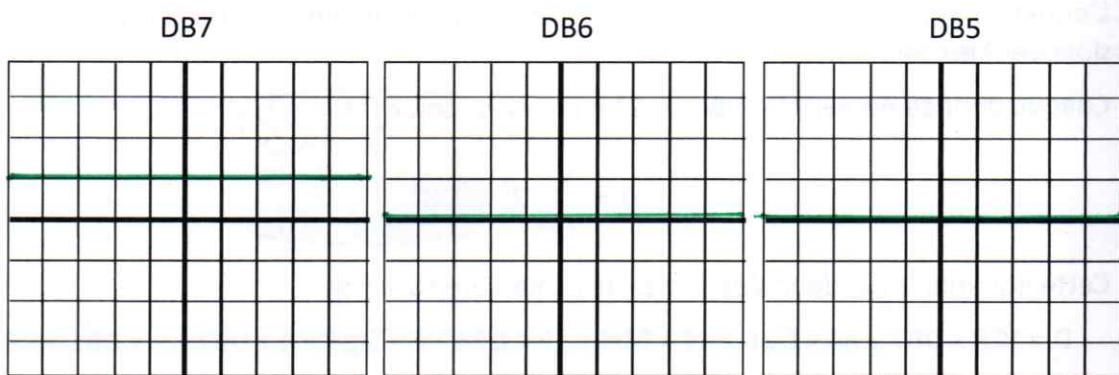
- Une tension continue → Code : CNA\_DC\_v1.ino
- Une tension de forme carrée avec une donnée level\_low (niveau bas) et une donnée level\_high (niveau haut) → Code : CNA\_SQUARE\_v1.ino
- Une tension de forme sinusoïdale de période T → Code : CNA\_SINUS\_v1.ino

### 6.1. Conversion de données numériques continues

6.1.1. Programmer le microcontrôleur avec le code CNA\_DC\_v1.ino.

6.1.2. Observer les données DB7 à DB0. → tension nulle sauf DB7

6.1.3. Tracer les données DB7 à DB5.



Calibre Axe horizontal : 100ms /div

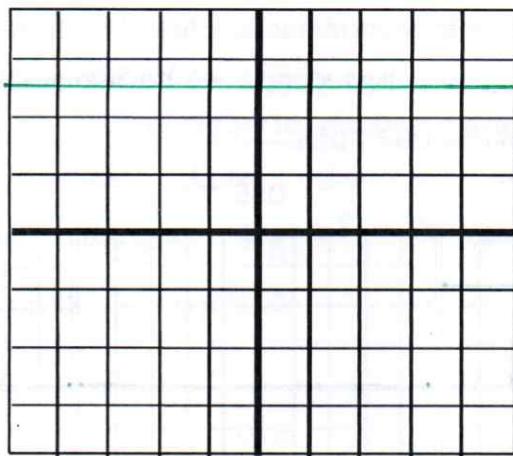
Calibre Axe vertical : 5V /div

6.1.4. En étudiant le code CNA\_DC\_v1.ino, expliquer les données relevées expérimentalement.

La ligne : byte data(nb\_data) (B100 0000)  
donne les données numériques . ↑ ↑  
DB7 DB0

Si 1 => tension de 5V  
0 => tension de 0V.

6.1.5. Observer et mesurer la tension convertie en sortie du CNA.



Calibre Axe horizontal : 100ms/div

Calibre Axe vertical : 1V /div

6.1.6. A partir des données DB7 à DB0 mesurées et sachant que la tension de référence  $V_{ref} = -5V$ , justifier la valeur de la tension en sortie mesurée expérimentalement.

$$V_{out} = -V_{ref} \times \frac{D}{256} \quad \text{Or, } D = 128 \text{ car seul DB7 a un signal}$$

$$\text{Donc } V_{out} = -(-5) \times \frac{128}{256} = 2,5V$$

6.1.7. Modifier les valeurs du signal numérique à convertir en modifiant le code du microcontrôleur. Ne pas oublier de reprogrammer le microcontrôleur à chaque fois que le code est modifié.

6.1.8. Observer les signaux DB7 à DB0 et la tension en sortie convertie. Justifier le résultat obtenu.

En modifiant tq : B1000010

On a une tension à DB7 et DB1 et une tension  $V_{out} \approx 2,64V$

En effet :  $V_{out} = -V_{ref} \times \frac{D}{256}$  avec  $D = 130$

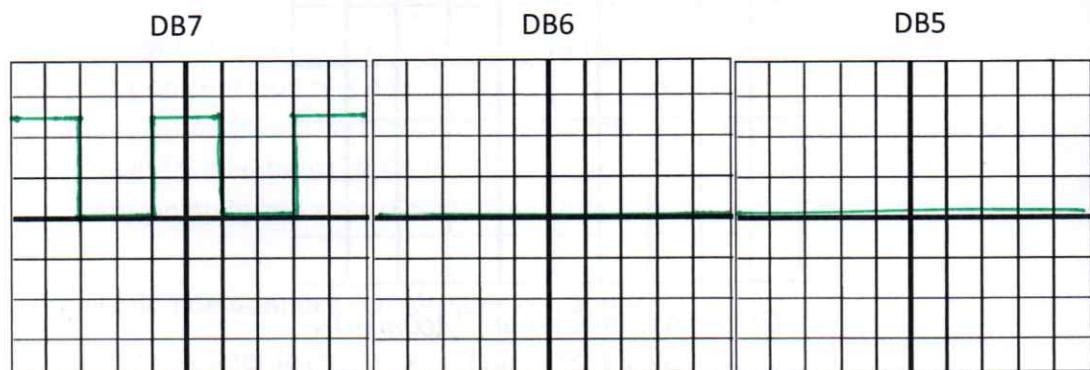
$$V_{out} = 5 \times \frac{130}{256} = 2,64V$$

## 6.2. Conversion d'un signal carré

6.2.1. Programmer le microcontrôleur avec le code CNA\_SQUARE\_v1.ino.

6.2.2. Observer les données DB7 à DB0. → tension nulle sauf DB7

6.2.3. Tracer les données DB7 à DB5.



Calibre Axe horizontal :  $50\mu s$  /div

Calibre Axe vertical :  $2V$  /div

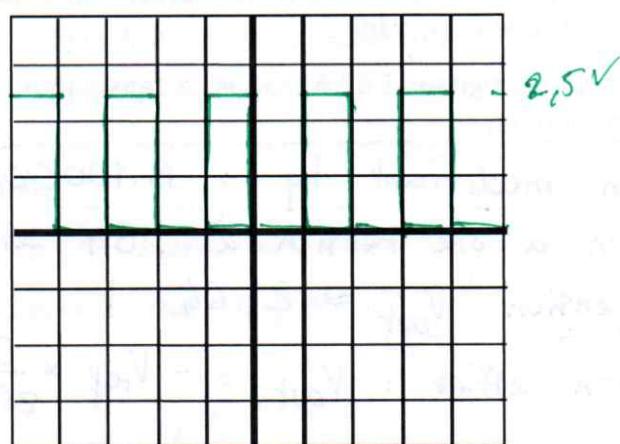
6.2.4. En étudiant le code CNA\_SQUARE\_v1.ino, expliquer la forme et la valeur des signaux DB7 à DB0 relevés expérimentalement.

paramètre data-h et data-l  
uniquement pour DB7 → donne un signal carré

6.2.5. Quelle est la période des signaux DB7 à DB0 ?

$$T = 124 \mu s$$

6.2.6. Tracer l'oscillogramme de la tension convertie en sortie en indiquant la période du signal et la durée à l'état bas et à l'état haut.



Calibre Axe horizontal :  $50\mu s$  /div

Calibre Axe vertical :  $1V$  /div

6.2.7. Modifier l'état bas level\_low et l'état haut level\_high du signal. Comment procédez-vous ?

En modifiant les valeurs de data-h et data-l  
(on remplace les 0 par des 1)

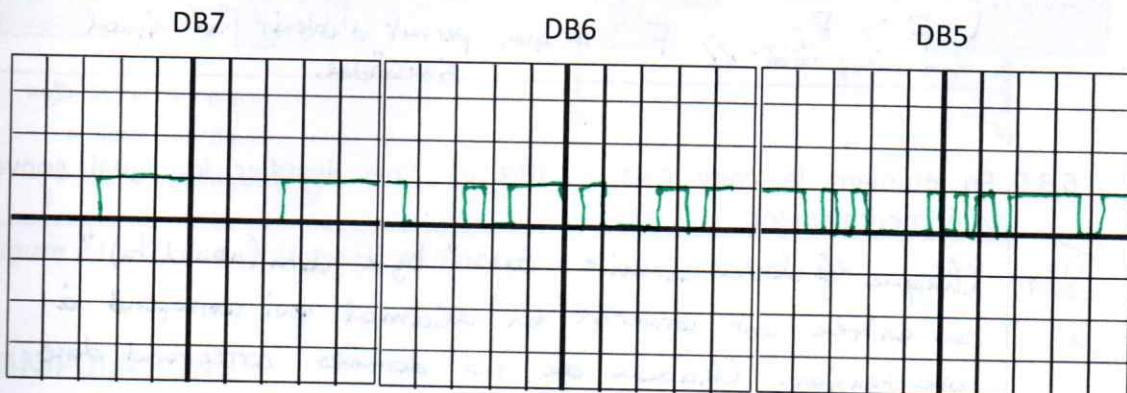
### 6.3. Conversion d'un signal sinusoïdal

6.3.1. Programmer le microcontrôleur avec le code CNA\_SINUS\_v1.ino.

6.3.2. Observer les données DB7 à DB0 à l'oscilloscope. En étudiant le code CNA\_SINUS\_v1.ino, expliquer les résultats mesurés.

Les données des data donnent de nombreuses valeurs pour chaque signal ce qui donne des signaux carrés complexes.

6.3.3. Tracer les données DB7 à DB5.



Calibre Axe horizontal : 10ms /div

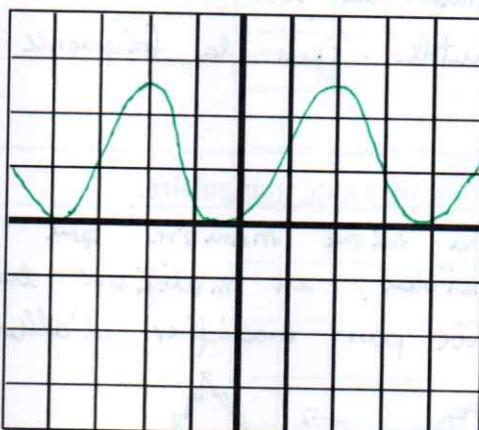
Calibre Axe vertical : 5V /div

6.3.4. Observer la tension convertie en sortie. Tracer approximativement son allure en indiquant la période du signal. Justifier l'allure.

$$T = 55 \text{ ms}$$

6.3.5. Observer simultanément la tension convertie et le signal WRB\_CNA. Ce signal permet de déclencher la conversion.

6.3.6. Tracer ces 2 tensions (sortie du CNA et WRB\_CNA) et utiliser des calibres différents pour mettre en évidence les informations utiles.



Calibre Axe horizontal : 12,5 ms /div

Calibre Axe vertical : 2V /div



Calibre Axe horizontal : 1ms /div

Calibre Axe vertical : 2V /div



### 6.3.7. Mesurer la durée des paliers de la tension convertie. *période d'échantillonnage*

- Cette mesure s'appelle la période d'échantillonnage.

$$T_{ech} = 1 \text{ ms}$$

*(1) la  $T_{ech}$  est grande (2) les paliers sont petits => (3) petits*

- En déduire la fréquence d'échantillonnage.

$$F_{ech} = 1.000 \text{ Hz}$$

### 6.3.8. Mesurer ou calculer avec l'oscilloscope la fréquence du signal converti ?

- Comparer cette fréquence avec la fréquence d'échantillonnage.

$$T = 55 \text{ ms} \quad F = 18,2 \text{ Hz}$$

*( $T = 50 \text{ ms}$   
 $F = 20 \text{ Hz}$ )  $F_{ech} \gg F$  ce qui permet d'obtenir un signal sinusoïdal.*

### 6.3.9. En étudiant le code CNA\_SINUS\_v1.ino, justifier le signal converti expérimentalement.

*en ajoutant  
le binaire (10000000)  
à son argument  
la période de  
5ms*

*(chaque donnée binaire dans "byte data(nb-data)" en entrée est convertie en décimal qui correspond à une tension. Chaque de ces données correspond donc à un palier représenté dans le signal de sortie.)*

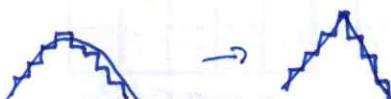
$$\Rightarrow \text{selon } V_{out} = -V_{ref} \times \frac{D}{256}$$

### 6.3.10. Proposer une solution pour augmenter la fréquence du signal généré par le microcontrôleur et converti par le CNA.

*On peut baisser la période d'échantillonnage, afin d'avoir avec le même nombre de données en entrée une période en sortie plus faible. Donc la fréquence de sortie sera augmentée*

### 6.3.11. Proposer une solution pour générer un signal triangulaire.

*On peut procéder de la même manière que pour générer un signal sinusoïdal, en modifiant les données binaires en entrée pour modifier l'allure de la courbe :*



## 7. Test de la chaîne de traitements

**OBJECTIF :** Concevoir une chaîne de traitements constituée d'un microcontrôleur, d'un CNA, d'un filtre passe-bas, d'un amplificateur et d'un comparateur.

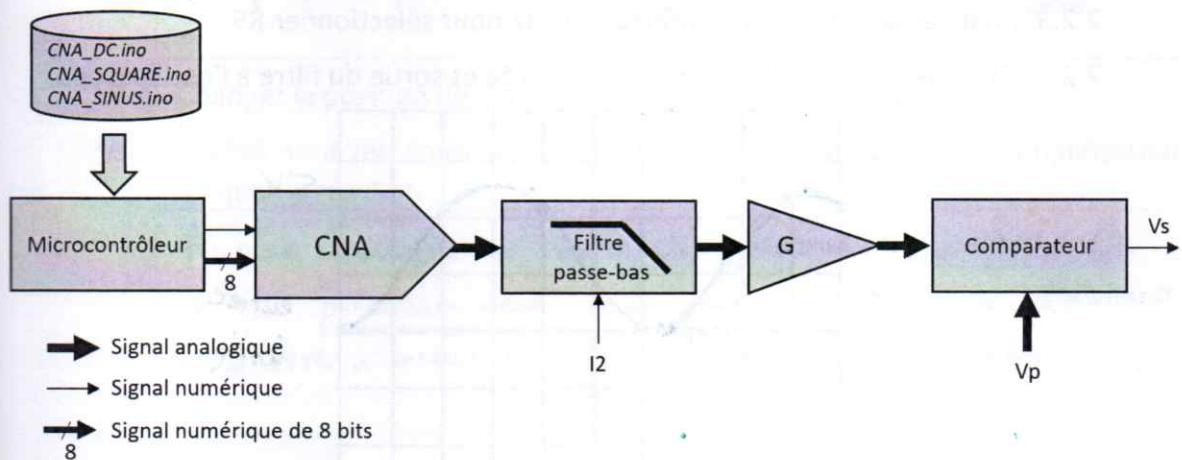


Figure 23 : Chaîne de traitements numérique et analogique d'un signal généré par le microcontrôleur

### MANIPULATIONS

#### 7.1. Microcontrôleur

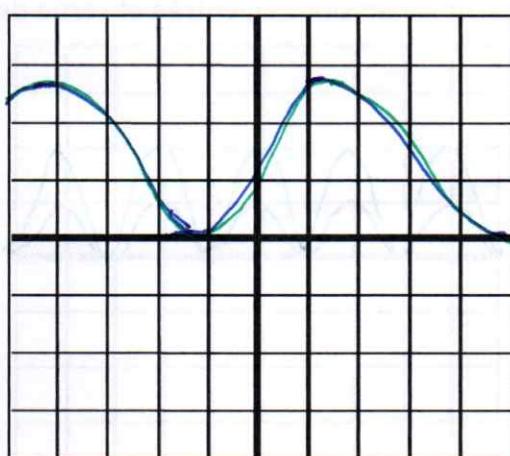
7.1.1. Programmer le microcontrôleur avec le code `CNA_SINUS_v1.ino`.

#### 7.2. Filtre passe-bas

7.2.1. Positionner l'interrupteur I1 pour connecter la sortie du CNA à l'entrée du filtre passe-bas.

7.2.2. Positionner I2 pour utiliser la résistance R4 (haut).

7.2.3. Observer les tensions en entrée et sortie du filtre à l'oscilloscope (indiquer les point-test correspondants).



PT entrée: 26 / 30  
PT sortie: 27

Calibre Axe horizontal : 10ms /div

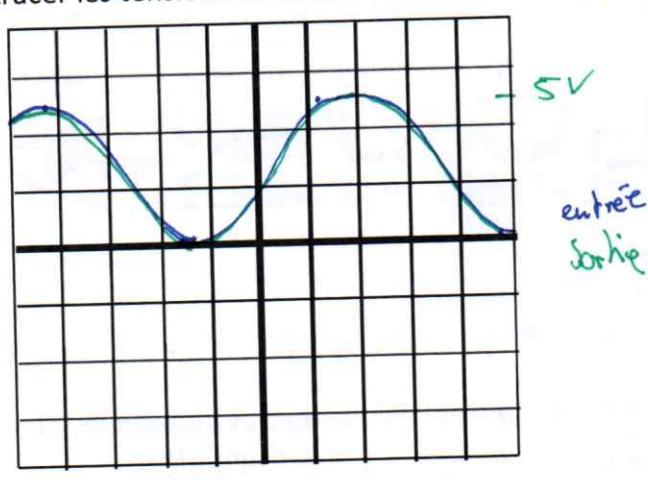
Calibre Axe vertical : 2V /div

7.2.4. Les tensions observées sont-elles identiques ? Expliquer le résultat.

Oui, le filtre commence à filtrer à partir de 2k Hz (avec R5).  
Or, la fréquence en sortie du OVA est d'environ 20 Hz et  $20 \text{ Hz} \ll 2\text{k Hz}$ , donc le filtre ne filtre pas la tension.

7.2.5. Changer la position de l'interrupteur I2 pour sélectionner R5.

7.2.6. Observer et tracer les tensions en entrée et sortie du filtre à l'oscilloscope.



Calibre Axe horizontal : 10ms /div

Calibre Axe vertical : 2V /div

7.2.7. Quelles sont les conséquences sur la tension de sortie du filtre ? Expliquer ce résultat.

La tension de sortie ne change pas - avec R5 le filtre commence à filtrer à 100 Hz, or la fréquence d'entrée est de 20 Hz donc le filtre ne filtre pas.

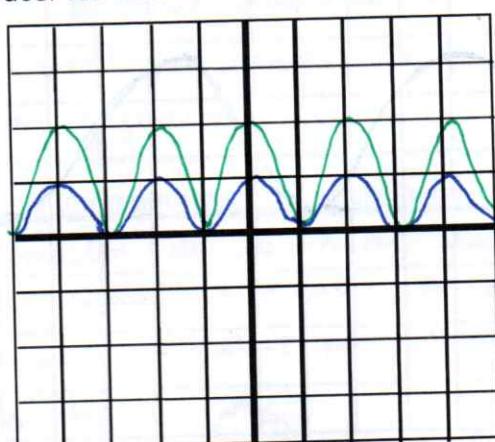
### 7.3. Amplificateur

7.3.1. Positionner I2 pour utiliser la résistance R4 (haut)

7.3.2. Observer et tracer les tensions en entrée et sortie de l'amplificateur.

PT entrée : PT27

PT sortie : PT28



Calibre Axe horizontal : 25ms /div

Calibre Axe vertical : 5V /div

7.3.3. Les tensions observées sont-elles identiques ? Expliquer le résultat.

Non, car l'amplificateur a un gain de 2 et il fonctionne donc la tension de sortie est amplifiée par 2 :  $V_{max\text{ entrée}} = 5V$   $V_{max\text{ sortie}} = 10V$

7.3.4. Changer la position de l'interrupteur I2 pour sélectionner R5.

7.3.5. Quelles sont les conséquences sur la tension de sortie de l'amplificateur ? Expliquer le résultat.

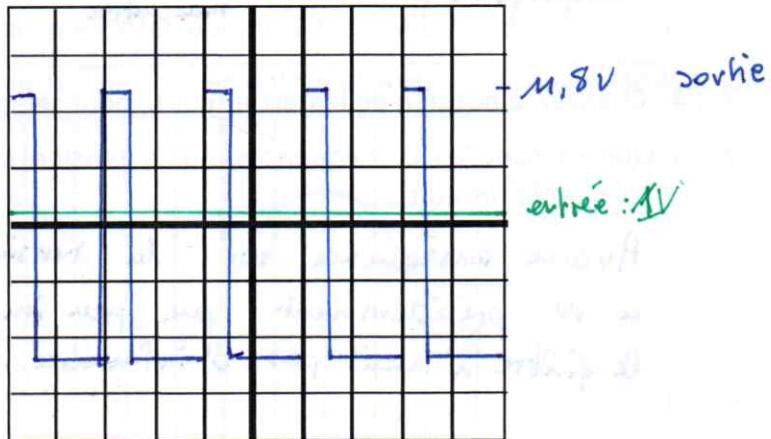
Aucune conséquence sur la tension de sortie. On a vu précédemment que peu importe la résistance, le filtre n'avait pas d'influence sur la tension.

## 7.4. Comparateur

7.4.1. Régler la tension de seuil  $V_p$  à 1 V.

7.4.2. Observer et tracer les tensions à l'entrée et la sortie du comparateur à l'oscilloscope et expliquer le résultat.

*rapport cyclique : durée état haut  
= période*



Calibre Axe horizontal : 25ms /div

Calibre Axe vertical : 5V /div

7.4.3. Calculer le rapport cyclique de la sortie du comparateur et expliquer le résultat.

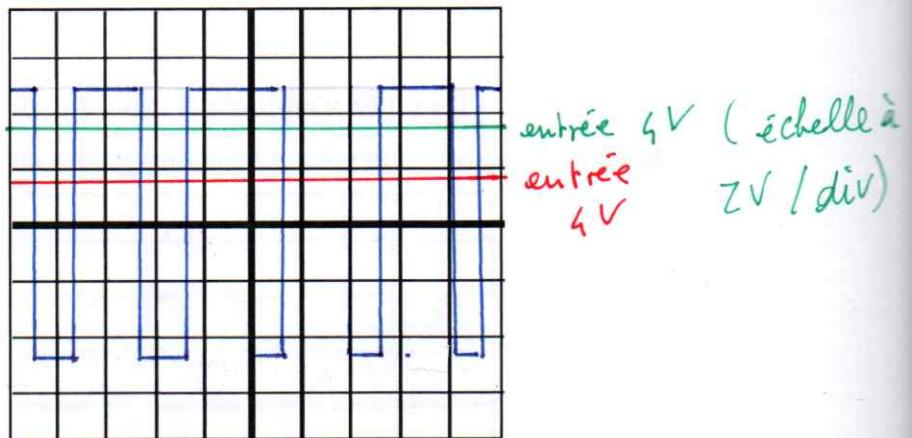
On compare à la tension en sortie de l'ampli: 10V

$$\text{durée état haut : } 16 \text{ ms} \\ \text{période : } 55 \text{ ms} \quad \text{siusäidet } \alpha = \frac{16}{55} \Rightarrow 0,29$$

Car on compare 1V à 10V donc faible intervalle

7.4.4. Régler la tension de seuil  $V_p$  à 4 V.

7.4.5. Observer et tracer les tensions à l'entrée et la sortie du comparateur à l'oscilloscope et expliquer le résultat.



Calibre Axe horizontal : 25ms /div

Calibre Axe vertical : 5V /div

7.4.6. Expliquer le résultat.

Comme  $V_p = 4 \text{ V}$ , par rapport à la situation précédente l'intervalle où  $V_{in} > V_p$  est plus court.

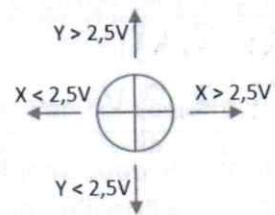
tension de sortie - 12V

## Partie 3 : Les capteurs

### 1. Joystick

**OBJECTIF :** Utiliser un capteur de position

Le joystick est un capteur analogique dont la tension en sortie varie avec la position du joystick sur 2 axes.



Il a 5 connexions :

- **Vcc** : connecté à l'alimentation 5V par la carte Arduino,
- **Gnd** : connecté à la masse,
- **X** : tension proportionnelle au déplacement du joystick sur l'axe horizontal,
- **Y** : tension proportionnelle au déplacement du joystick sur l'axe vertical,
- **BP** : tension égale à 0 V ou 5 V en fonction de l'état du bouton poussoir.

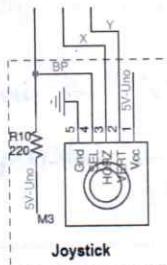


Figure 24 : Joystick

Le joystick permet de régler deux potentiomètres de  $10\text{ k}\Omega$  (un pour chaque axe) et de délivrer une tension proportionnelle au déplacement du joystick.

### PRÉPARATION

Repérer le numéro des Points Test (PT) correspondants aux sorties X, Y et BP du joystick en vous aidant du schéma de la carte capteur sur la figure 9.

X : PT 7  
Y : PT 8  
BP : PT 6

# MANIPULATION

Le joystick est alimenté par la carte Arduino.

### 1.1. Vérifier le fonctionnement du joystick à l'aide de l'oscilloscope

#### 1.1.1. Relever les tensions sur les sorties X, Y et BP à l'état repos.

$$\begin{array}{ll} X: 2,48 \text{ V} & \text{BP: } 5,04 \text{ V} \\ Y: 2,6 \text{ V} & \end{array}$$

1.1.2. Modifier doucement les positions X et Y du joystick tout en observant ces 2 tensions à l'oscilloscope. Comment varient ces valeurs ?

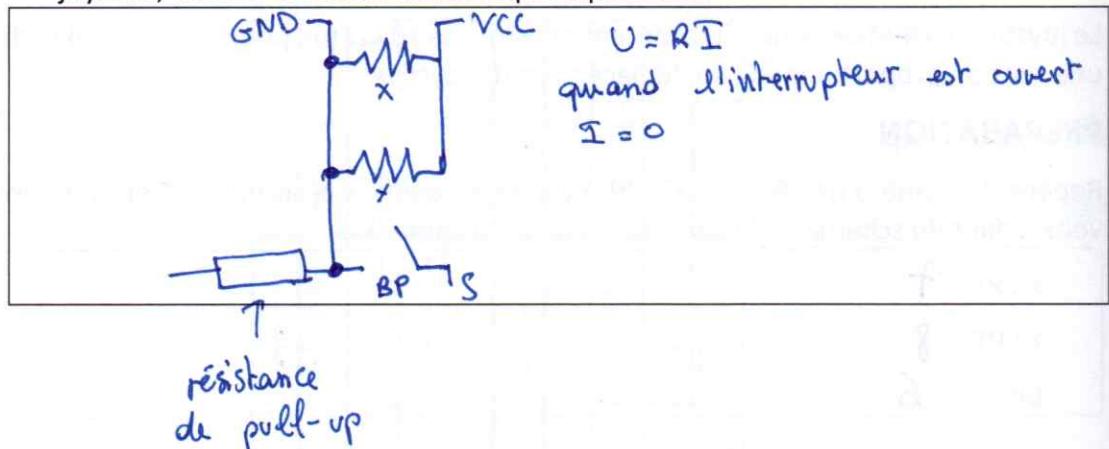
$X$  varie entre 0 et 5V ( $\pm 200$  mV)

$\chi$  varie entre 0 et 5V ( $\pm 200\text{mV}$ )

1.1.3. Appuyer sur le bouton poussoir et mesurer la tension sur la sortie BP à l'oscilloscope. Expliquer les différences entre l'état « non appuyé » et l'état « appuyé ».

État "non appuyé":  $V_{BP} = 5V \rightarrow$  interrupteur ouvert:  $V_S = V_{CC} = 5V$   
 État "appuyé":  $V_{BP} = 0V \rightarrow$  interrupteur fermé:  
 $V_S = V_{GND} = 0V$

1.1.4. A la suite de ces mesures et après avoir observé le composant physique du joystick, donner un schéma électrique équivalent.



$0 \rightarrow 0V$

$1023 \rightarrow 5V$

## 1.2. Vérifier le fonctionnement du joystick à l'aide du microcontrôleur et de l'ordinateur

Le rôle du microcontrôleur est d'acquérir les tensions analogiques disponibles sur le joystick (sorties X, Y et BP) de les convertir en numérique pour qu'elles puissent être exploitable par l'ordinateur.

L'ordinateur joue le rôle d'IHM (Interface Homme Machine) et assure donc la visualisation de ces données dans une « fenêtre » dite terminal ou moniteur.

Le microcontrôleur possède un CAN sur 10 bits. Par conséquent, chaque tension analogique convertie en numérique prendra une valeur en décimal comprise entre 0 et 1023 ( $1023 = 2^{10}-1$ ).

Il est rappelé que le CAN (externe au microcontrôleur) étudié dans la partie 5 est sur 8 bits. Par conséquent, la donnée numérique générée par ce CAN est comprise entre 0 et 255 ( $255 = 2^8-1$ ).

### 1.2.1. Programmer le microcontrôleur avec le code Joystick\_XYBP\_v1.ino.

### 1.2.2. Relever les tensions en X et Y ainsi que les valeurs numériques sur le « moniteur série » arduino pour les 9 positions suivantes du joystick :

Positions	à gauche en bas	à gauche au milieu	à gauche en haut	à droite en bas	à droite au milieu	à droite en haut	milieu	milieu en haut	milieu en bas
X (en V)	0	0	0	4,99	4,99	4,99	2,45	2,45	2,45
X (moniteur série)	1	1	1	1020	1020	1020	501	501	501
Y (en V)	4,99	2,54	0	4,99	2,54	0	2,54	0	4,99
Y (moniteur série)	1020	519	1	1020	519	1	519	1	1020

### 1.2.3. En fonction des caractéristiques du CAN du microcontrôleur et des positions du joystick, expliquer les données relevées.

Le CAN est codé sur 10 bits donc ~~peut~~ donne des valeurs comprises entre 0 et 1023. Le signal analogique en entrée est comprise entre 0 et 5V, donc 5V donne une sortie à  $1023 = 2^{10}-1$ , on a donc :

$$V_{out} = \frac{V_{in} \times 5}{(2^{10}-1)}$$

1.2.4. Expliquer le rapport entre les valeurs numériques relevées par le microcontrôleur (disponibles dans le « moniteur série ») et les tensions en X et en Y.

$$V_{out} = \frac{V_{in} \times 5}{(2^{10}-1)} \quad (\text{cf qn° précédente})$$

1.2.5. Appuyer sur le bouton poussoir, quelle est la conséquence ?

Le moniteur série affiche "BP appuye"

1.2.6. Donner une explication de l'observation.

D'après le code : si la tension est inférieure à 0,49 V (120 sur 1023) alors "appui" est vrai donc le moniteur affiche "BP appuye". Or, on a vu que lorsque BP est appuyé, la tension est de 0V. Donc on a BP appuyé qui affiche le bon message.

### 1.3. Utiliser le microcontrôleur pour interpréter les positions du joystick

1.3.1. Modifier le code pour afficher sur le « moniteur série » non pas la donnée convertie brute mais les 9 positions du joystick.

à gauche en bas	à gauche au milieu	à gauche en haut	à droite en bas	à droite au milieu	à droite en haut	milieu	milieu en haut	milieu en bas
-----------------	--------------------	------------------	-----------------	--------------------	------------------	--------	----------------	---------------

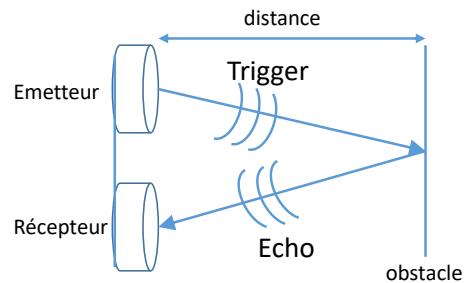
$$0 < X < 100 \\ 900 < Y < 1023$$

1.3.2. Modifier le code pour afficher sur le « moniteur série » l'état du bouton poussoir BP : « APPUYE » ou « NON APPUYE ».

## 2. Capteur Ultrason

**OBJECTIF :** Utiliser un capteur pour mesurer une distance.

Le capteur d'ondes ultrasons est un émetteur/récepteur. Le principe est d'émettre un signal ultrason (principe du SONAR) et de mesurer le temps entre l'émission et la réception après une réflexion sur un obstacle. La mesure de ce temps permet de calculer la distance entre le capteur et l'obstacle.



Ce capteur est composé de 2 cellules ultrason (1 émetteur et 1 récepteur) qui permet des mesures entre 3 cm et 4 m toutes les 50 ms.

Il a 5 connexions :

- **VCC** : Alimentation (habituellement 5V)
- **Gnd** : La masse
- **Trigger** : Emission d'un signal à ultrason
- **Echo** : Réception de l'écho
- **NC** : Non connecté.

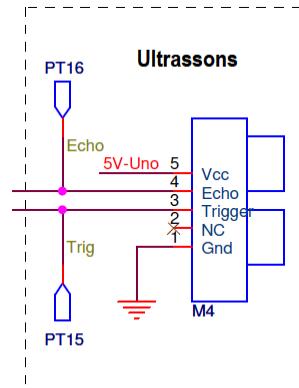


Figure 25 : Capteur ultrason

## PREPARATION

Repérer les broches (pin) du capteur correspondant à Trigger et Echo ainsi que les Points Test de la carte correspondants.

- Trigger : PT *15*
- Echo : PT *16*

- Le décalage temporel (retard) entre le signal émis (Trigger) et le signal reçu (Echo) correspond au temps aller-retour de l'onde entre le capteur et l'obstacle. Ce temps dépend de la distance et de la vitesse du signal ultrason :

$$t = 2d/v$$

Sachant que la vitesse du son dans l'air est d'environ 340m.s<sup>-1</sup>, calculer les retards pour les distances suivantes : d= 5 cm, 10 cm, 20 cm et 50 cm.

$$\text{Retard } (d = 5 \text{ cm}) = 2,9 \cdot 10^{-4} \text{ s} = 0,29 \text{ ms}$$

$$\text{Retard } (d = 10 \text{ cm}) = 5,9 \cdot 10^{-4} \text{ s} = 0,59 \text{ ms}$$

$$\text{Retard } (d = 20 \text{ cm}) = 1,2 \cdot 10^{-3} \text{ s} = 1,2 \text{ ms}$$

$$\text{Retard } (d = 50 \text{ cm}) = 2,9 \cdot 10^{-3} \text{ s} = 2,9 \text{ ms}$$

Pour utiliser ce capteur, il est nécessaire de lui fournir le signal d'émission Trigger et d'observer l'information de réception Echo.

Le microcontrôleur pilote le capteur (Trigger), récupère l'information de réception (Echo), l'interprète pour extraire la distance mesurée entre le capteur et l'obstacle.

Le signal Trigger est donc une sortie du microcontrôleur.

Le signal Echo est donc une entrée du microcontrôleur.

La distance est disponible dans la fenêtre terminal « série » de l'IDE d'Arduino (p.69).

# ANIPULATIONS

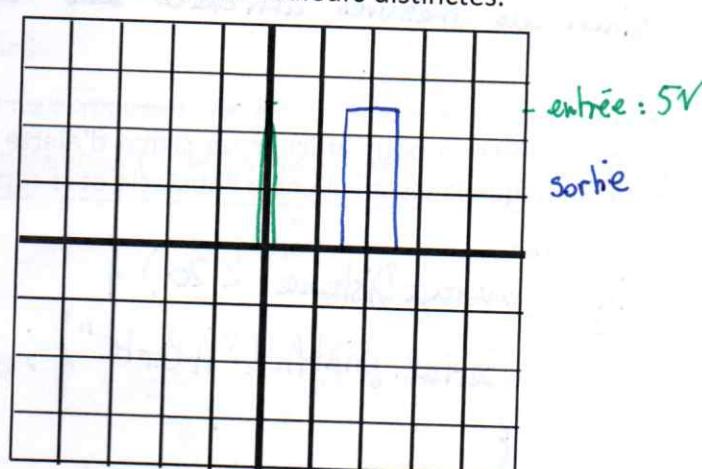
## 2.1. Mesures sur le capteur ultrason

023

2.1.1. Programmer le microcontrôleur avec le code Capteur\_ultrason\_v1.ino.

2.1.2. Observer à l'aide de l'oscilloscope les signaux Trigger (CH1) et Echo (CH2) en mettant un obstacle à quelques cm du capteur.

2.1.3. Tracer ces 2 tensions avec deux couleurs distinctes.



Calibre Axe horizontal : 500μs/div

Calibre Axe vertical : 2V /div

La durée de l'état haut de la tension Echo est le temps de propagation aller/retour de l'onde dépend directement de la distance. Il correspond au temps donné dans la formule  $t = 2d/v$ .

2.1.4. Relever la durée de l'état haut de la tension Echo pour les positions de l'obstacle à 5, 10, 20 cm et 50 cm du capteur.

$$\text{Retard } (d = 5 \text{ cm}) = 320 \mu\text{s} = 0,32 \text{ ms}$$

$$\text{Retard } (d = 10 \text{ cm}) = 600 \mu\text{s} = 0,6 \text{ ms}$$

$$\text{Retard } (d = 20 \text{ cm}) = 1,12 \text{ ms}$$

$$\text{Retard } (d = 50 \text{ cm}) = 2,48 \text{ ms}$$

2.1.5. Comparer aux résultats de la préparation.

On a des résultats qui sont proches de ceux de la préparation. Les différences observées peuvent être dues à l'incertitude de placement de l'objet à x cm.

2.1.6. Ouvrir le terminal « série » du logiciel Arduino. La distance entre l'obstacle et le capteur y est affichée.

2.1.7. Vérifier que le capteur fonctionne correctement et que les distances mesurées sont cohérentes par rapport à la position de l'obstacle.

## 2.2. Utilisation du microcontrôleur pour interpréter les informations du capteur

- 2.2.1. Pour différentes positions de l'obstacle, observer les informations affichées dans le terminal « série » de l'IDE d'Arduino.

Commentaires : Quand il n'y a pas d'obstacles, la distance maximale est de 527 cm.  
Sinon les mesures affichées sont cohérentes.

- 2.2.2. Modifier le code pour afficher un signal d'alerte « ALERTE » sur le terminal « série » lorsque la distance entre l'obstacle et le capteur est inférieure à 20 cm.

Commentaires :

```
if (averageDistance < 20) {  
    Serial.println ("Alerte");  
}
```

- 2.2.3. Modifier le code pour allumer la LED disponible sur la carte arduino sur la broche 13 du microcontrôleur lorsque la distance entre l'obstacle et le capteur est inférieure à 20 cm.

Commentaires :

```
int led = 13  
dans setup: pinMode (led, OUTPUT);  
dans loop: if — {  
    digitalWrite (led, HIGH);  
    else digitalWrite (led, LOW);
```

- 2.2.4. Modifier le code pour faire clignoter à une fréquence d'environ 1 Hz la led disponible sur la carte arduino sur la broche 13 du microcontrôleur lorsque la distance entre l'obstacle et le capteur est inférieure à 10 cm.

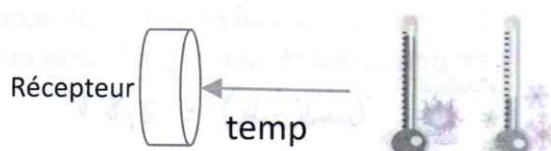
Commentaires :

```
while (averageDistance < 10) {  
    digitalWrite (led, HIGH);  
    delay (1000);  
    digitalWrite (led, LOW);  
}
```

## Capteur de température - LM35

2023

Le capteur de température génère une tension analogique dont la valeur varie en fonction de la température.



3 connexions :

- **Vcc** : Alimentation (habituellement 5V)
- **Gnd** : La masse
- **Vout** : Tension de sortie

LM35 : La tension de sortie du composant varie d'après ses caractéristiques de 10 mV par °C.

Sur la carte capteur, un amplificateur (TL081) a été ajouté afin d'amplifier la tension de sortie.

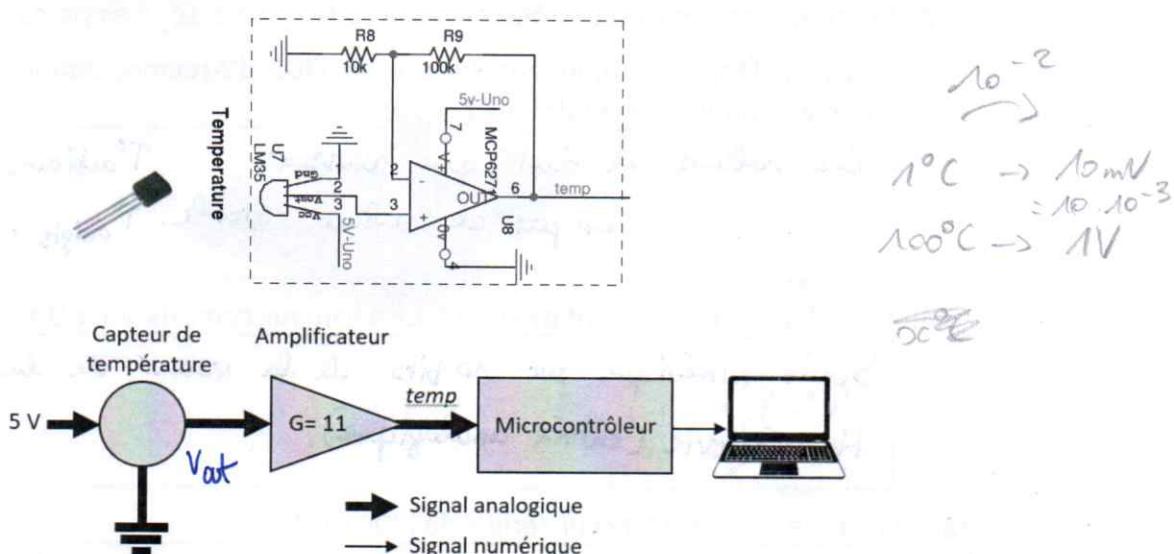


Figure 26 : Capteur de température

## PRÉPARATION

Repérer la sortie de l'amplificateur associé au capteur correspondant à Temp ainsi que le Point Test.

Temp : PT 5

En vous aidant du montage amplificateur de la figure 14, par identification, exprimer la tension VTTemp en fonction de la tension de sortie V<sub>OUT</sub>.

$$VTTemp = f(Vout) = 11 \times V_{out} \quad V_{out} = \frac{V_{temp}}{11}$$

A partir des caractéristiques du capteur LM35, exprimer la température TEMP en fonction de la tension V<sub>OUT</sub>.

$$Temp = f(Vout) = 10^2 \times V_{out}$$

$$Temp = \frac{100 \times V_{temp}}{11}$$

## MANIPULATIONS

- 3.1. Observer et mesurer à l'oscilloscope la tension VTemp pour 2 conditions de température différentes (ambiante et entre les doigts par exemple).

$$V_{temp} (\text{ambiante}) = 2,8 \text{ V} \approx 25^\circ\text{C}$$

$$V_{temp} (\text{doigts}) = 2,88 \text{ V} \approx 26,2^\circ\text{C}$$

Commentaires : On observe bien une augmentation de tension qui reflète une augmentation de la température quand on place le capteur de température entre nos doigts.

- 3.2. Programmer le microcontrôleur avec le code Capteur\_temperature\_v1.ino.

- 3.3. Ouvrir la fenêtre « moniteur série » de l'IDE d'Arduino, donner les valeurs qui s'affichent pour les 2 conditions ?

Les valeurs ne font que baisser  $T^\circ\text{ambiante} : 573$   
→ pas de valeur stable.  $T^\circ\text{doigts} : 640$

- 3.4. Les valeurs affichées sont des nombres décimaux compris entre 0 à 1023 : pourquoi ?

Sortie numérique sur 10 bits de la valeur de la température (entrée analogique)

- 3.5. Ouvrir la fenêtre « traceur série » de l'IDE (p. 71).

Commentaires : Courbe de la  $T^\circ$  (en 10 bits) en fonction du temps

- 3.6. Modifier le code du microcontrôleur pour afficher la température en °C.

Commentaires :

- 3.7. Modifier le code du microcontrôleur pour interroger le capteur toutes les secondes et afficher la moyenne de 5 mesures toutes les 5 secondes.

Commentaires :