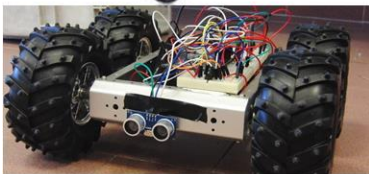
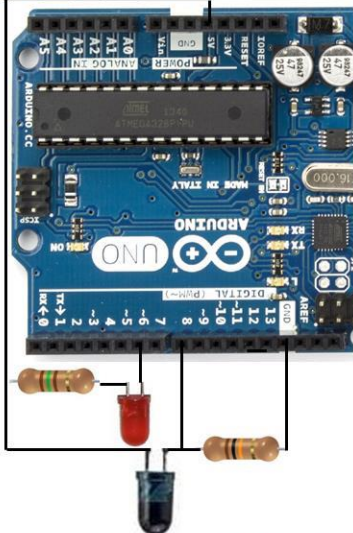
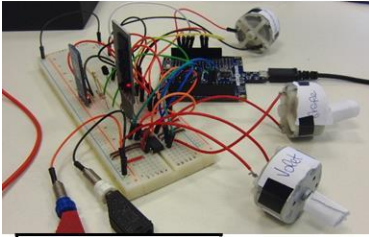
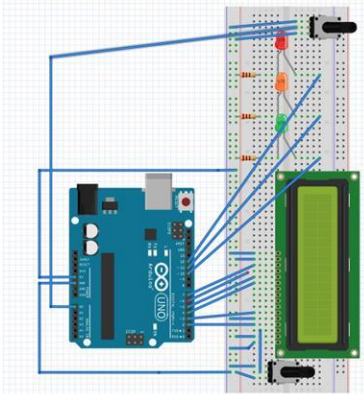


ECOLE POLYTECHNIQUE UNIVERSITAIRE DE NICE SOPHIA-ANTIPOLIS

*Cycle Initial Polytechnique
Deuxième année*



Travaux Dirigés d'électronique avec Arduino



Pascal MASSON



Année scolaire 2015/2016

SOMMAIRE

TD No. 1 : Entrées analogique et LED	6
TD No. 2 : Ecran LCD	7
TD No. 3 : PWM	8
TD No. 4 : Transmission IR	10

TD No. 1 : Entrées analogique et LED

Exercice 1.1 : Réalisation d'un voltmètre

A l'aide du moniteur série du logiciel Arduino, afficher la tension obtenue avec un potentiomètre de 10 k Ω .

Exercice 1.2 : Réalisation d'un voltmètre à LED

I.2.1. Réaliser un voltmètre qui donne une indication sur la valeur de la tension avec des LED comme illustré à la figure (1.2). La tension variable à mesurer sera obtenue avec un potentiomètre de 10 k Ω .

I.2.3. A l'aide du moniteur série, relever la tension maximale mesurée par l'entrée analogique lorsque les 3 LED sont allumées. Déterminer pourquoi vous n'obtenez pas 5 V mais un peu moins.

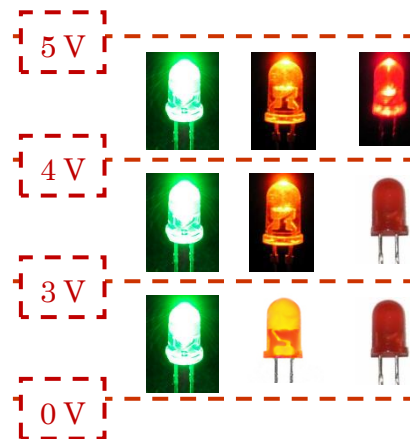


Figure 1.2. Seuils d'allumage des LED en fonction de la tension mesurée

Exercice 1.3 : Indicateur de tension à fenêtre

Dans cet exercice on cherche à régler une tension entre 2.4 V et 2.6 V comme illustré à la figure (1.2).

La tension variable à mesurer sera obtenue avec un potentiomètre de 10 k Ω

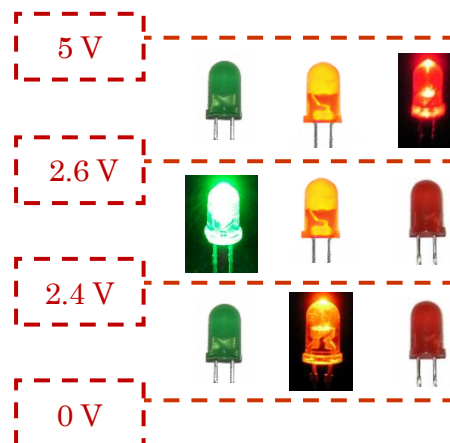


Figure 1.3. Seuils d'allumage des LED en fonction de la tension mesurée

Exercice 1.4 : Seuil de basculement d'une I/O

Une I/O de l'arduino configurée en entrée détecte les tensions 0 et 5 V, or les circuits que nous utilisons ne délivrent pas exactement ces tensions. A l'aide du moniteur série et d'un potentiomètre de 10 k Ω , déterminer la tension de basculement entre l'état LOW et l'état HIGH.

TD No. 2 : Ecran LCD

Exercice 2.1 : Réalisation d'un voltmètre

Faire les branchements de l'écran LCD et afficher la valeur de la tension obtenue sur une des entrées. On utilisera un potentiomètre de 10 k Ω

Exercice 2.2 : Smiley

Réaliser un symbole « smiley » et le faire circuler sur l'écran

TD No. 3 : PWM

Exercice 1 : Modulation de la luminosité d'une LED

1.1. Réaliser un montage et un programme qui allume progressivement une LED puis l'éteint progressivement

1.2. A l'aide de l'oscilloscope, mesurer les fréquences des signaux PWM des I/O n°3 et 5

Exercice 2 : Modulation de la luminosité d'une LED avec bouton poussoir

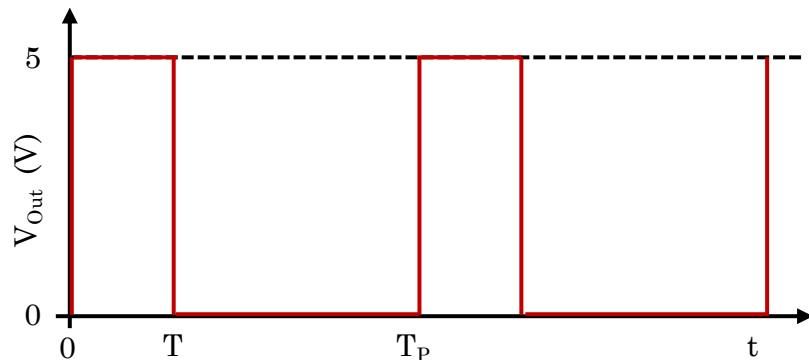
Réaliser un montage et programme qui allume progressivement une LED en fonction du temps durant lequel on appuie sur un bouton. Quand la LED est au maximum de son éclairement, elle s'éteint avant de recommencer le cycle.

Exercice 3 : Obtention d'une tension analogique

Un signal rectangulaire se compose d'une valeur moyenne et d'une somme de sinusoïdes d'amplitudes et de fréquences différentes. On obtient ces paramètres en effectuant une décomposition en série de fourrier du signal rectangulaire. La fréquence minimale (appelée le fondamental) des sinusoïdes est égale à celle de la fréquence du signal rectangulaire.

3.1. Déterminer la valeur moyenne, A_0 , du signal rectangulaire de la figure (3.3.1)

Figure 3.1. Signal PWM en sortie de l'arduino



3.2. On souhaite filtrer ce signal pour ne conserver que la valeur moyenne

3.2.1. Quel type de filtre faut-il utiliser ?

3.2.1. Représenter la courbe de ce filtre sur la figure (3.2)

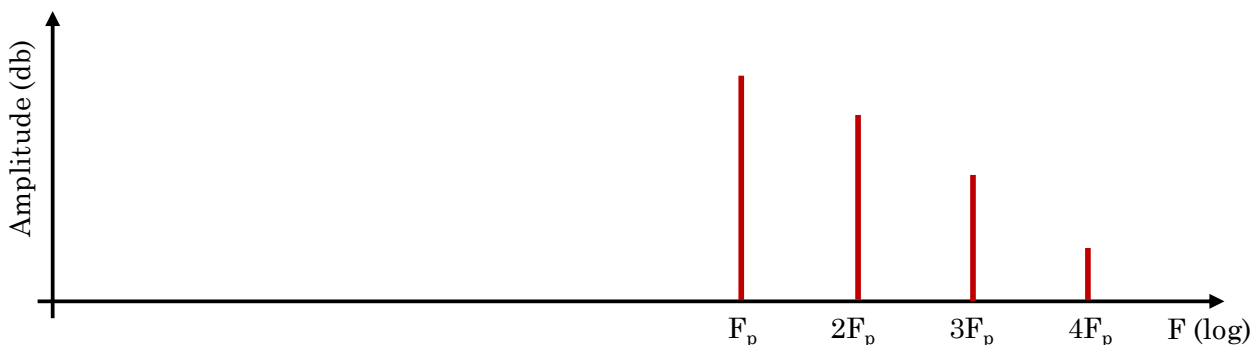


Figure 3.2. Représentation des quatre premières fréquences qui constituent un signal rectangulaire

3.2.3. Comment faut-il choisir la fréquence de coupure, F_c , du filtre ?

3.2.4. La fréquence du signal PWM de l'I/O n°11 est de 490 Hz. Vous avez à disposition une capacité de 470 nF, que faut-il choisir comme valeur de résistance ?

3.3. Modulation PWM

On souhaite transformer un signal sinusoïdal en PWM puis filtrer ce signal et obtenir à nouveau le signal sinusoïdal.

3.3.1. La fréquence de la sinusoïde est notée F_s , positionner cette fréquence sur la figure (3.2.).

3.2.2. La sinusoïde est obtenue avec un potentiomètre. Représenter sur la figure (3.3), le schéma du montage qui permet d'appliquer la « sinusoïde » sur l'entrée analogique n°0, de générer le PWM sur la sortie I/O n°11 et mesurer la valeur moyenne sur l'entrée analogique n°1 de l'arduino.



Figure 3.3. Schéma du montage

3.2.3. Une entrée analogique varie de 0 à 5 volts ce qui est traduit par l'arduino en un chiffre (entier) qui va de 0 à 1023. Une sortie PWM est ajustable avec une variable (entière) qui va de 0 à 255. Réaliser le montage de la figure (3.3) et le programme qui permet de réaliser cet exercice. Il faudra afficher sur votre moniteur série les tensions appliquées aux entrées analogiques n° 0 et 1.

3.2.4. Visualiser les signaux PWM et PWM filtrés avec l'oscilloscope.

TD No. 4 : Transmission IR

Exercice 1 : Code morse sans arduino

1.1. Afin de prendre en main la diode IR et le phototransistor IR, réaliser le circuit de la figure (1.1) qui permet de transmettre un code morse sans utiliser l'arduino (sauf le 5 V et la masse). Il faudra utiliser un bouton poussoir pour générer le code et utiliser une LED rouge pour visualiser le code reçu.

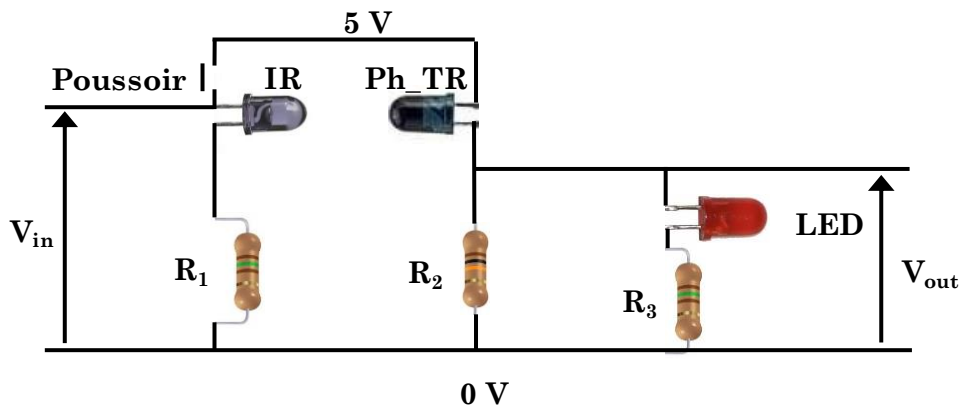


Figure 1.1. Montage transmission/réception d'un code morse

1.2. Avec les entrées analogiques de l'arduino et le moniteur série, mesurer et afficher les tensions V_{in} , V_{R1} et V_{out} . Que se passe-t-il pour la tension V_{out} si vous enlever la LED rouge ?

Exercice 2 : Manipulation d'un mot binaire

Compléter l'exercice 1 en utilisant votre arduino mais en connectant la LED sur une I/O différente de celle du phototransistor. L'entrée V_{in} (transmission) et la sortie V_{out} (réception) correspondent respectivement aux I/O n°8 et n°9.

2.1. Appliquer un mot binaire sur V_{in} , par exemple 1001010101101, avec un débit suffisamment lent pour voir la diode s'allumer et s'éteindre

2.2. Modifier le programme pour que le mot binaire soit fourni par le moniteur série

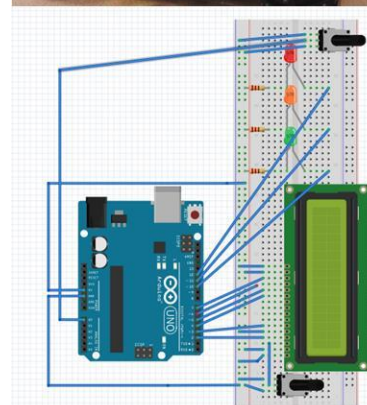
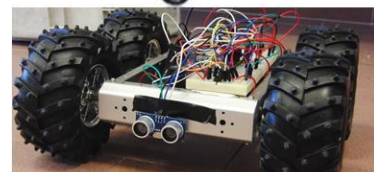
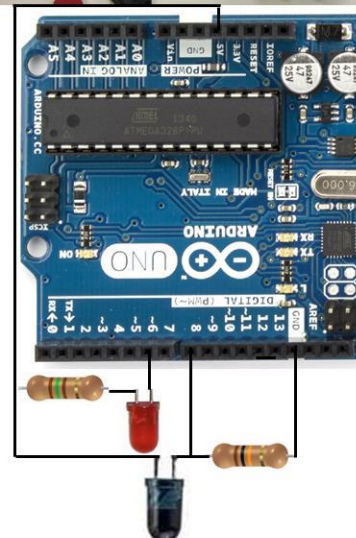
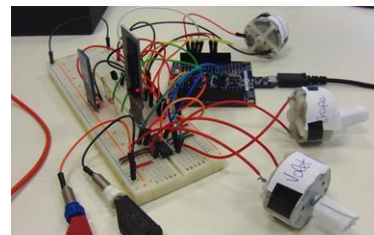
Exercice 3 : Communication entre 2 arduino

Ce travail se fait en binôme avec 2 diodes IR et 2 phototransistors IR.

A l'aide de la librairie « SoftwareSerial », chaque membre du binôme doit envoyer/recevoir un texte de/à l'autre membre

ECOLE POLYTECHNIQUE UNIVERSITAIRE DE NICE SOPHIA-ANTIPOLIS

*Cycle Initial Polytechnique
Deuxième année*



Pascal MASSON
Polytech'Nice Sophia-Antipolis, Département électronique
1645 route des Lucioles
06410 BIOT

Tél : 04 92 38 85 86
Email : pascal.masson@unice.fr