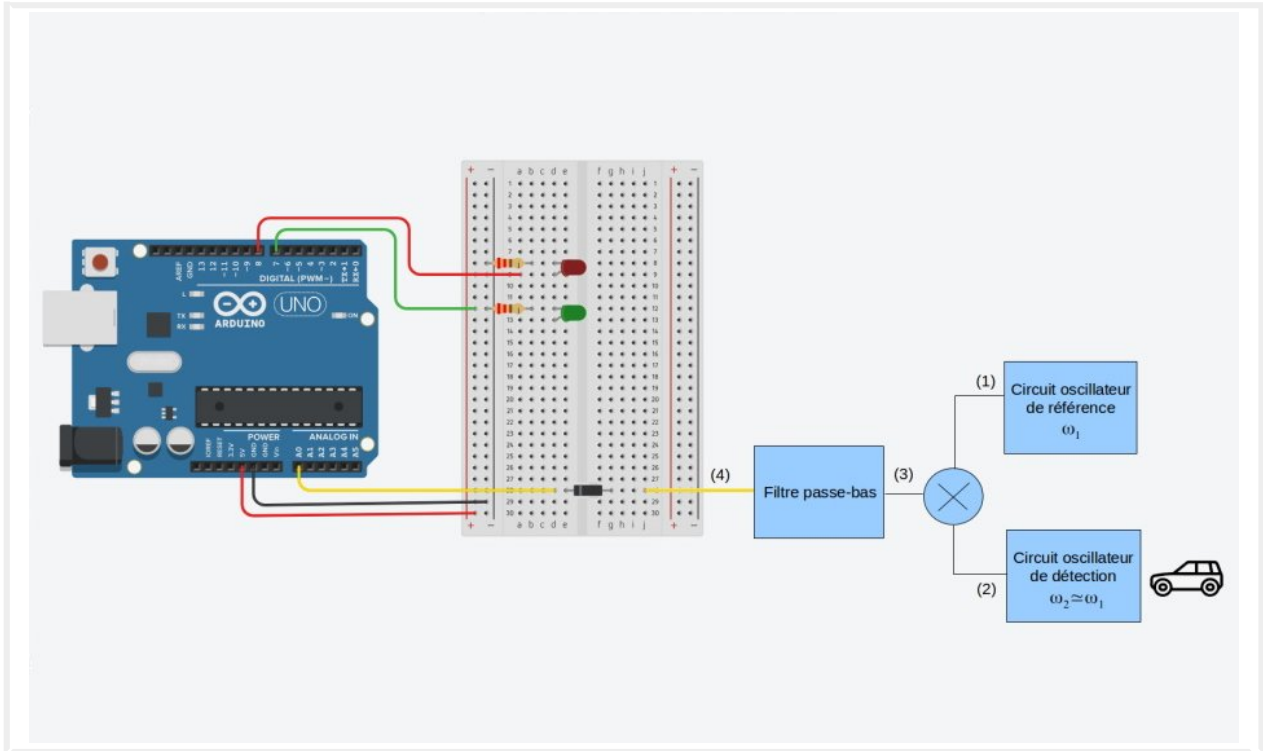


Feu de circulation

(Simulation du fonctionnement d'un feu de circulation à détection de présence)



Liste des composants

- . 1 Montage détecteur de métaux à battement de fréquence
- . 1 DEL rouge
- . 1 DEL verte
- . 1 Diode de redressement 1N4004
- . 2 résistances de 220 Ω
- . 1 plaques d'essais
- . Fils de connexion

Objectif

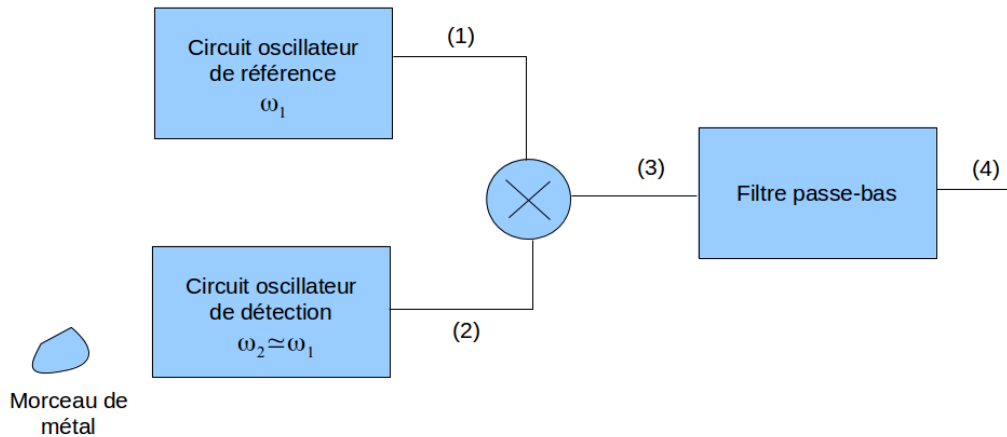
L'objectif du montage est de simuler le fonctionnement d'un feu de circulation à détection de présence.

On utilise pour cela un montage détecteur de métaux à battement de fréquence, relié à une entrée analogique de l'Arduino Uno, qui en présence de métal (en l'occurrence, une voiture) permettra à l'Arduino d'allumer la DEL verte et d'éteindre la DEL rouge, puis l'inverse en absence de métal.

Le principe général du détecteur de métaux à battement de fréquence est d'utiliser deux oscillateurs, l'un fixe, l'autre dont la fréquence d'oscillation dépend de la présence d'un métal.

La présence d'un métal près d'une bobine modifie, par mutuelle induction, son inductance propre apparente et donc, si un oscillateur est construit avec cette dernière, celui-ci aura sa fréquence d'oscillation qui varie légèrement lors la présence de métal.

Il suffit ensuite de comparer les fréquences des deux signaux, pour cela on les fait « battre » en les multipliant l'un par l'autre puis en filtrant le signal résultant.



Un oscillateur de référence (un GBF) fournit un signal sinusoïdal (1) de pulsation ω_1 et un oscillateur de détection un signal sinusoïdal (2) de pulsation ω_2 .

$$u_1 = U_{m1} \cos(\omega_1 t + \varphi_1) \text{ et } u_2 = U_{m2} \cos(\omega_2 t + \varphi_2)$$

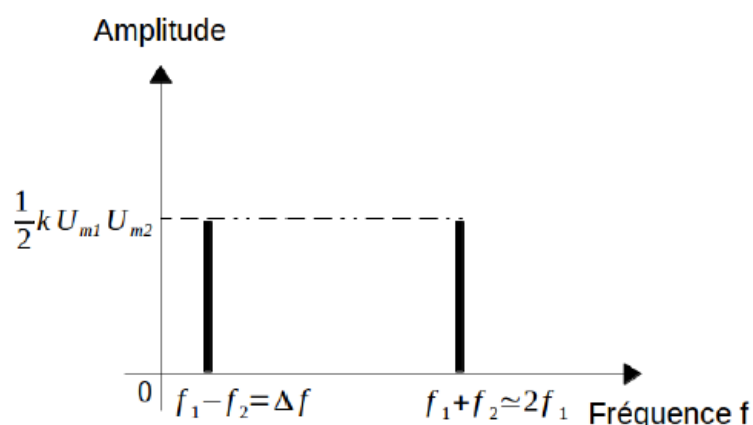
Lorsqu'aucun métal n'est présent, on a $\omega_1 = \omega_2$. Lors de la présence d'un métal, la pulsation du signal (2) est légèrement modifiée.

Après multiplication des deux signaux, on obtient en (3) un signal :

$$u_3 = kU_{m1}U_{m2} \cos(\omega_1 t + \varphi_1) \cos(\omega_2 t + \varphi_2)$$

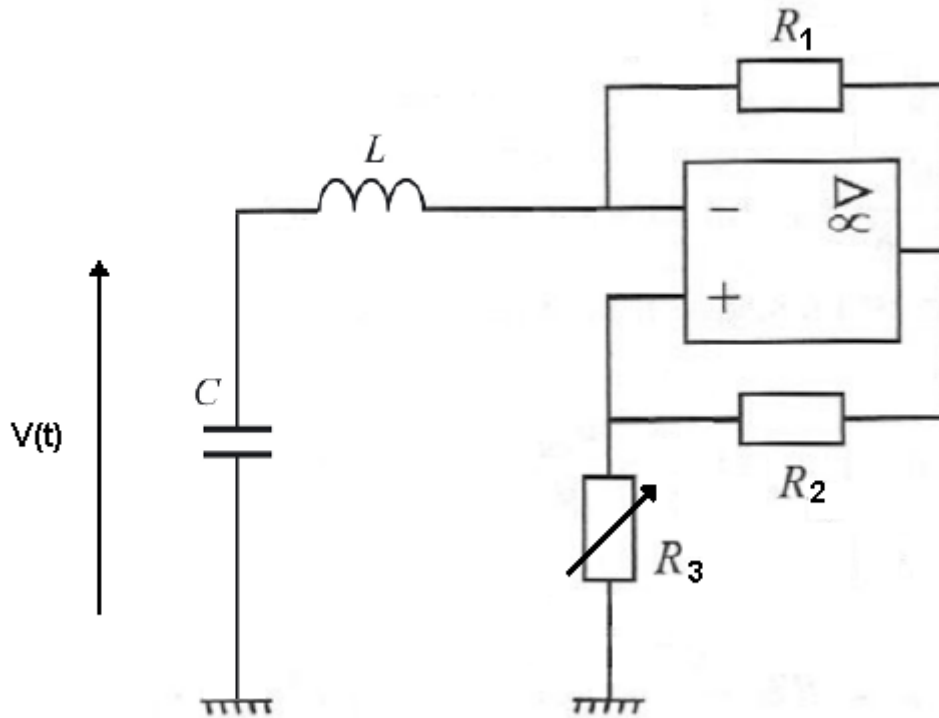
$$u_3 = \frac{kU_{m1}U_{m2}}{2} (\cos((\omega_1 - \omega_2)t + (\varphi_1 - \varphi_2))) + \cos((\omega_1 + \omega_2)t + (\varphi_1 + \varphi_2)))$$

dont le spectre est représenté ci-dessous :



Il ne reste plus qu'à effectuer un filtrage des hautes fréquences pour récupérer un signal (4) dont la fréquence est Δf .

Circuit oscilateur de détection (oscillateur LC à résistance négative) :



- 1 AO
- 1 Condensateur : $C = 0,1 \mu F$
- 1 Bobine 1000 spires, $L = 44mH$
- 2 Résistance $R_1 = R_2 = 1k\Omega$
- 1 Potentiomètre $R_3 = 10k\Omega$

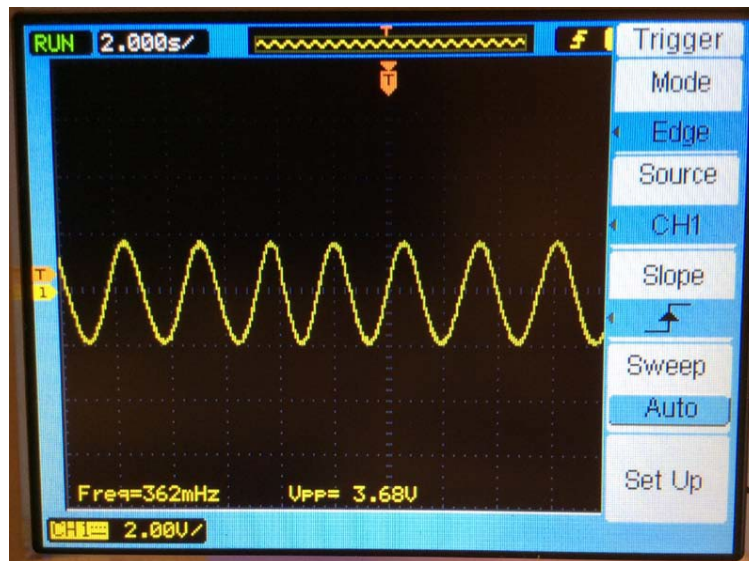
Oscillogramme du circuit oscilateur de détection en l'absence de métal :



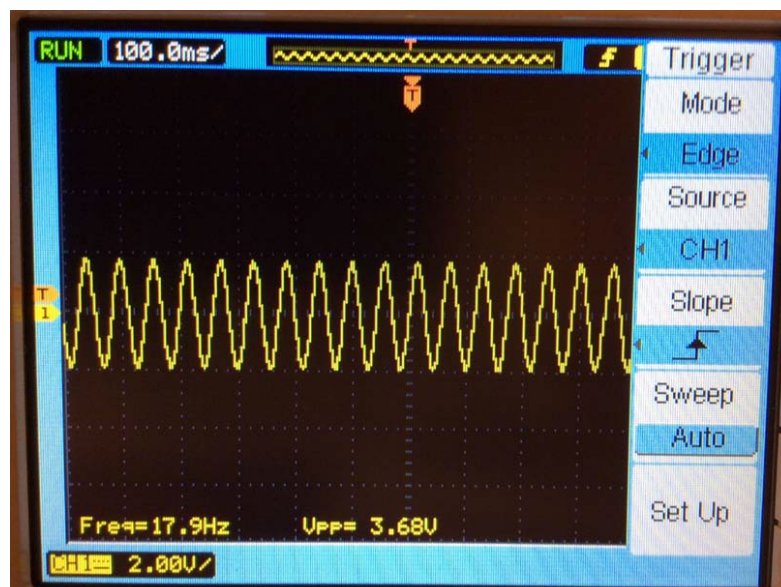
Pour le circuit oscillateur de référence, on utilise un GBF réglé à la même fréquence que le circuit de détection en l'absence de métal :



En sortie du filtre passe-bas et en l'absence de métal, la fréquence du signal est très basse (Δf très petit) :



En approchant du métal de la bobine du circuit détecteur, la fréquence du signal en sortie du filtre passe-bas augmente (Δf augmente) :



Attention :

Appliquer une tension supérieure à 5 volts ou inférieure à 0 volt sur une broche analogique endommagera immédiatement et définitivement la carte Arduino.

Une diode de redressement en sortie du montage détecteur de métal permet de n'appliquer qu'une tension positive sur l'entrée analogique A0.

Il faudra cependant veiller à régler l'amplitude du signal du GBF afin de limiter au maximum à 5 V, le signal en sortie de la diode de redressement.

Le programme téléverser dans la mémoire de l'Arduino permet de mesurer la fréquence du signal appliqué sur l'entrée analogique A0 par mesure de la durée pour passer de la valeur maximale à la valeur minimale.

Si cette durée est supérieure à une valeur définie dans le programme, aucune voiture n'est détectée, la DEL rouge, initialement allumée, reste allumée.

Au contraire, si la durée mesurée est inférieure à la valeur définie, une voiture est détectée, la DEL rouge est alors éteinte et la DEL verte est allumée. Celle-ci sera ensuite éteinte (et la DEL rouge allumée) quand il n'y aura plus de détection de métal.

. Le programme

Voici le code de l'activité :

```
Feu_Circulation

// Déclaration des constantes et variables

int sensorVal = 0;
int OldsensorVal=0;
const int PinLEDR = 8;
const int PinLEDV = 7;
int FeuVert = 0;
unsigned long StartTime = 0;
unsigned long DeltaTime = 1000;

// Initialisation des entrées et sorties

void setup() {
  Serial.begin(9600);
  pinMode (PinLEDR, OUTPUT);
  pinMode (PinLEDV, OUTPUT);
  digitalWrite(PinLEDR, HIGH);
}
```

```

// Fonction principale en boucle

void loop() {
  sensorVal = analogRead(A0);
  if ((sensorVal > 100) && (abs(OldsensorVal - sensorVal)>80)) {
    StartTime = millis();
    OldsensorVal = sensorVal;
  }
  if ((sensorVal < 10) && (abs(OldsensorVal - sensorVal)>80)) {
    DeltaTime = millis() - StartTime;
    OldsensorVal = sensorVal;
  }

  if (FeuVert ==0) {
    if (DeltaTime < 300) {
      digitalWrite(PinLEDR, LOW);
      digitalWrite(PinLEDV, HIGH);
      FeuVert = 1;
    }
  }
  else {
    if (DeltaTime > 300) {
      digitalWrite(PinLEDR, HIGH);
      digitalWrite(PinLEDV, LOW);
      FeuVert = 0;
    }
  }
  delay(100);
}

```

Déroulement du programme :

– 1. Déclaration des constantes et variables :

. **int sensorVal = 0** (variable nombre entier pour stocker la valeur du potentiel de la broche du détecteur de métal)

. **int OldsensorVal = 0** (variable nombre entier pour stocker la précédente valeur du potentiel de la broche du détecteur de métal)

. **const int PinLEDR = 8** (constante nombre entier correspondant à la broche de la DEL rouge)

. **const int PinLEDV = 7** (constante nombre entier correspondant à la broche de la DEL verte)

. **int FeuVert = 0** (variable nombre entier indiquant l'état du feu vert, 0 : éteint, 1 : allumé)

. **unsigned long StartTime = 0** (variable nombre entier long positif pour stocker l'heure à laquelle l'amplitude du signal en sortie du détecteur de métal est maximal)

. **unsigned long DeltaTime = 1000** (variable nombre entier long positif pour le calcul de la durée en ms pour que l'amplitude du signal en sortie du détecteur de métal passe d'un maximum à un minimum)

– 2. Initialisation des entrées et sorties :

. **Initialisation de la liaison série à un débit de 9600 bauds,**

. **Initialisation des broches des DELs en sortie,**

. **La DEL rouge est allumée.**

– 3. Fonction principale en boucle :

–> **Lecture de la valeur de la broche du détecteur de métal,**

–> **Calcul de la durée en ms (DeltaTime) pour que l'amplitude du signal en sortie du détecteur de métal passe d'un maximum à un minimum,**

–> **Les DELS rouge et verte sont allumées ou éteintes en fonction de la valeur de DeltaTime :**

- **Si DeltaTime est inférieure à 300 ms, il y a détection d'une voiture. Le feu vert est allumé s'il ne l'était pas et dans ce cas, la DEL rouge est éteinte,**

- **Si DeltaTime est supérieur à 300 ms, il n'y a pas détection d'une voiture. Le feu vert est éteint s'il ne l'était pas et dans ce cas, la DEL rouge est allumée.**