

Polytech Nice Sophia

Rapport projet Arduino 2017/2018



---

La sarbacane connectée  
Par Pujol Elise et Maugan Sciarra

---

# Sommaire

---

1. Présentation du projet
    - 1.1 Français
    - 1.2 Anglais
  2. Objectif
    - 2.2 Principal
    - 2.3 Contrôle par Bluetooth
  3. Matériel utilisé
    - 3.1 Orientation
    - 3.2 Lancement
    - 3.3 Electronique
    - 3.4 Support
  4. Fonctionnement du projet
    - 4.1 Descriptif
    - 4.2 Code
  5. Problèmes rencontrés
  6. Conclusion
- 
-

# 1. Présentation du projet

---

## 1.1 Français

Arrow-Matics est une sarbacane électronique créé sous le cadre du projet arduino de CIP2. Le nom est un mélange du mot flèche en anglais (arrow) et automatique car la sarbacane est totalement automatique. Le concept du projet est de contrôler, avec le téléphone, le lancement de la flèche, l'orientation de la sarbacane Bluetooth.

Il suffit donc de connecter le téléphone à la puce bluetooth de l'arduino pour avoir le contrôle sur la sarbacane. Des flèches permettent de contrôler les rotations et un bouton vert d'enclancher la pression et le lancement.

C'est un objet créé pour l'amusement qui cependant à une grande puissance de tir.

## 1.2 Anglais

Arrow-Matics is an electronic blow pipe created as part of the CIP2 arduino project. Its name is a mix between "arrow" and automatic because it is an entirely automatic blow pipe. The project's concept is to be able to control, with the phone, the arrow's launching, the blow pipe's Bluetooth orientation.

You just have to connect your phone to the arduino's Bluetooth chip to have control over the blow pipe. Arrows will allow you to control the rotations and a green button will let you snap the pression off and begin the launching . This object was created for fun however it has a non negligible shooting power.

---

---

## 2. Objectif

### 2.1 But principal :

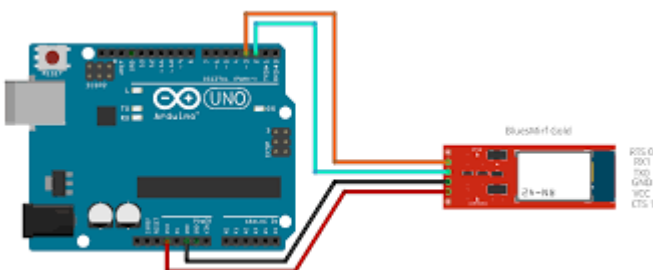
L'objectif de la conception de ce projet est de transformer un jouet pour enfant basique, sans électronique et sans complexité, en un objet totalement électronique tout en gardant sa fonction principale. Ici, la fonction principale est de tirer des fléchettes grâce au souffle, ainsi le projet avait pour but aussi de lier à la fois électronique et amusement. Ce projet avait aussi comme but d'apprendre à coopérer et travailler en duo.

### 2.1 Contrôlable par Bluetooth :

Orienter/viser : pouvoir orienter la sarbacane horizontalement et verticalement pour viser

Lancement : lorsque l'orientation semble correcte, envoyer la fléchette.

Toutes ces fonctionnalités sont réunies sur une application téléphonique où se trouve des flèches directionnelles pour l'orientation, un bouton de lancement et un dernier pour recharger la sarbacane.



### 3. Matériel utilisé

Sarbacane est la base de notre projet. C'est l'objet à automatiser.

#### 3.1 Pour l'orientation :

Utilisation deux moteurs arduino pas à pas (15281367ND).  
L'un est placé verticalement et l'autre horizontalement.

#### 3.2 Pour le lancement :

Compresseur à air RING, 12V, 10bar  
Electrovanne arduino  
Tuyau

#### 3.3 Côté électronique :

Carte Arduino Mega 2560  
Module Bluetooth HC-06  
Deux modules relais 5V

#### 3.4 Support :

Plastique  
Mecano  
Moteur pas à pas pour faire contre poid

---

Adapteur 12V/5V pour connecter les moteurs

Batterie externe pour le fonctionnement du compresseur

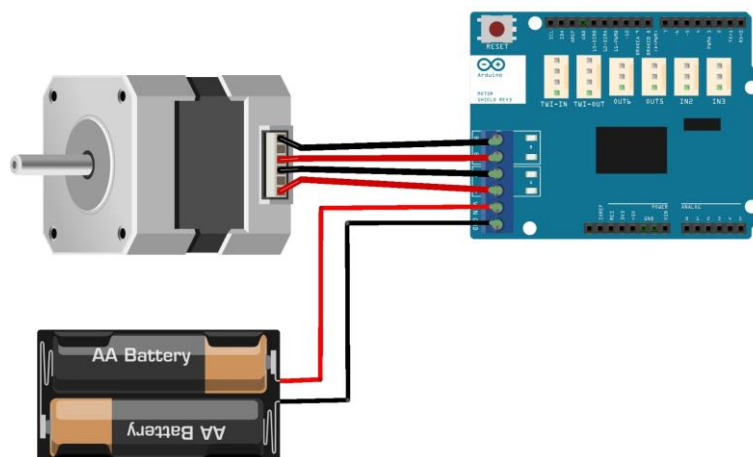
## 4. Fonctionnement du projet

---

### 4.1 Descriptif

Le compresseur à air délivre de l'air à pression normal. Il est relié à une électrovanne qui va emmagasiner l'air sorti du compresseur. L'électrovanne est connectée avec un tuyau à la sarbacane. Le programme pour l'arduino est réalisé en sorte qu'au bout d'un certain temps (moins d'une seconde) l'électrovanne s'ouvre et l'air propulse correctement la fléchette.

Pour l'orientation de la sarbacane, un moteur s'occupe du mouvement horizontal et un second du déplacement vertical. Celui de l'orientation horizontale est placé en dessous et est fixé au support où la sarbacane y est fixée. Celui pour l'axe vertical, il est fixé au-dessus sur la sarbacane directement, et la fait pivoter suivant un axe horizontal qui fait basculer la sarbacane vers l'avant ou l'arrière.



La pile ici représente l'alimentation par USB

## 4.2 Code

Pour chacune des rotations le programme est celui-ci :

```
char recvChar;
//On lit caractere par caractere sur le BTSerie et on affiche sur le Terminal Serie
if (BTSerie.available()) {
    recvChar = BTSerie.read(); //lecture
    if(recvChar == 'A') {
        rotG=true;
    }
    if(recvChar == 'a') {
        rotG=false;
    }
    if(rotG==true) {
        monMoteurPap.step(1);
        dep-=1;
    }
    if (rotG==false) {
        monMoteurPap.step(0); }
```

Le caractère envoyé par l'utilisateur est interprété et en fonction du caractère reçu cela lance la rotation.

---

---

Voici tout le code de notre projet :

```
#include <Stepper.h>
int nombre_pas=100;
int broche1=44;
int broche2=40;
int broche3=38;
int broche4=42;
////////
int broche21=52;
int broche22=48;
int broche23=46;
int broche24=50;
//////////
int broche31=53;
int broche32=49;
int broche33=47;
int broche34=51;
bool rotD=false;
bool rotG=false;
bool rotH=false;
bool rotB=false;
bool recharg=false;
int pression=36;
int lancement=34;
int dep=0;
bool depV=false;
Stepper monMoteurPap = Stepper(nombre_pas, broche1, broche2,broche3,broche4); // put your setup code here, to run once:
Stepper monMoteurPap2 = Stepper(nombre_pas, broche21, broche22,broche23,broche24); // put your setup code here, to run once:
Stepper monMoteurPap3 = Stepper(nombre_pas, broche31, broche32,broche33,broche34); // put your setup code here, to run once:

#include <SoftwareSerial.h> //Software Serial Port
#define RxD 11 //Pin 10 pour RX, PB2 sur votre board, a brancher sur le TX du HC-06
#define TxD 10 //Pin 11 pour TX, PB3 sur votre board, a brancher sur le RX du HC-06
SoftwareSerial BTSerie(RxD,TxD);

void setup(){
  Serial.begin(9600);
  // Configuration du bluetooth
  pinMode(RxD, INPUT);
  pinMode(TxD, OUTPUT);
  pinMode(pression,OUTPUT);
  digitalWrite(pression,HIGH);
  pinMode(lancement,OUTPUT);
  digitalWrite(lancement,HIGH);
  BTSerie.begin(9600);
  Serial.println("En attente de la commandes AT");
  delay(100);
  // Test des commandes AT
  BTSerie.print("AT+VERSION"); //Demande le NÂ° de version
  delay(1000);
  monMoteurPap.setSpeed(200);
  monMoteurPap2.setSpeed(200);
  monMoteurPap3.setSpeed(200);
}

void loop() {
  char recvChar;
  //On lit caractere par caractere sur le BTSerie et on affiche sur le Terminal Serie
  if (BTSerie.available()) {
    recvChar = BTSerie.read(); //lecture
    if(recvChar == 'A') {
      rotG=true;
    }
    if(recvChar == 'a') {
      rotG=false;
    }
    if(recvChar == 'B') {
      rotD=true;
    }
  }
}
```



```

    }
    if(recvChar == 'a') {
    rotG=false;
    } if(recvChar == 'B') {
    rotD=true;
    }
    if(recvChar == 'b') {
    rotD=false;
    }
    if(recvChar == 'C') {
    rotH=true;
    }
    if(recvChar == 'c') {
    rotH=false;
    }
    if(recvChar == 'D') {
    rotB=true;
    }
    if(recvChar == 'd') {
    rotB=false;
    }
    if(recvChar == 'R') {
    recharg=true;
    }
    if(recvChar == 'r') {
    recharg=false;
    }
    Serial.print(recvChar); //écriture
}

if(rotD==true) {
monMoteurPap.step(-1);
dep+=1;
}
if (rotD==false) {
monMoteurPap.step(0);
}
if(rotG==true) {
monMoteurPap.step(1);
dep-=1;
}
if (rotG==false) {
monMoteurPap.step(0); }
//////////

if(rotH==true) {
monMoteurPap2.step(-1);
dep+=1;
}
if (rotH==false) {
monMoteurPap2.step(0);
}
if(rotB==true) {
monMoteurPap2.step(1);
dep-=1;
}
if (rotB==false) {
monMoteurPap2.step(0); }
//////////
if(recharg==true) {
int i=0;
while(i<340){
monMoteurPap3.step(1);
i++;}
recharg=false;
}
if (recharg==false) {
monMoteurPap3.step(0); }
if (recvChar=='P'){
digitalWrite(pression,LOW);
}
if (recvChar=='p'){
digitalWrite(pression,HIGH);
}
if (recvChar=='G'){
digitalWrite(pression,LOW);
delay(1000);
digitalWrite(lancement,LOW);
}
}

```

```
    }
    if (recvChar=='g'){
        delay(200);
        digitalWrite(lancement,HIGH);
        digitalWrite(pression,HIGH);
    }
    if (recvChar=='L'){
        depV=true;
        if(dep>0){
            rotG=true;
        }
        if(dep<0){
            rotD=true;
        }
    }
    if (dep == -10){
        rotG=false;
        rotD=false;
        dep=0;
    }
    if (depV=true && dep==10){ //a corrigé !!!
        depV=false;
        rotG=false;
        rotD=false;
        dep=0;
        int i=0;
        while(i<560){
            monMoteurPap.step(-1);
            i++;
        }
        dep=560;
    }
    // Serial.println(dep);

    //On lit caractere par caractere sur le terminal et on affiche sur le BT Serie
    if (Serial.available()) {
        recvChar = Serial.read(); //lecture
        BTSerie.write(recvChar); //écriture
    }
}
```

## 5. Problèmes rencontrés

---

Tout au long de la réalisation du projet, il y a eu de nombreuses complications. Premièrement, le système de propulsion était indécis. En effet, il fallait assez de pression pour avoir une trajectoire de la fléchette correcte et linéaire. Plusieurs dispositifs étaient possibles.

L'électrovanne qui était un choix. L'air sortant du compresseur sera « stockée » dans une vanne et lorsqu'un certain niveau d'air est atteint donc un certain niveau de pression, l'air va être lâché d'un coup, donc il y aura une pression acceptable pour la propulsion. Seulement l'électrovanne trouvée emmagasine à peine moins la pression soutenable qu'envoie notre compresseur. Il en existe d'autres qui ont un niveau de pression plus haut seulement ils sont trop imposants pour notre projet. Il y a l'idée du solénoïde. Ce dernier est fait avec une bobine à l'intérieur, et à l'aide de l'arduino, un courant sera délivré et un champ magnétique se crée. Sur le solénoïde, il y a une tige en acier avec un piston, tendu et fixe grâce au champ, qui appuierait sur le tuyau du compresseur où l'air y sort. Lorsque que l'arduino lui demande de relâcher la pression, la tige se relève brutalement, et l'air emmagasinée dans le tuyau, aura assez de puissance pour propulser la fléchette. Seulement, notre tuyau est trop compact, pour que le solénoïde arrive à le compresser.

Il y a eu l'idée aussi de plier le tuyau avec un servomoteur. Seulement cette idée n'a pas été fluctuante, et l'idée a de suite été abandonnée à cause de la rigidité du tuyau.

Finalement, l'idée de l'électrovanne a été la plus convaincante.

Au niveau des branchements des moteurs, pas assez d'entrées Arduino étaient offertes par la plaque arduino. Après plusieurs recherches pour comprendre comment connecter deux Arduinos pour y avoir plus d'entrées, une arduino Mega était plus favorable.

De plus, lorsque tous les moteurs étaient branchés, et que le module bluetooth était connecté, cela faisait une chute de tension. Il fallait utiliser une autre source de courant que l'arduino.

Après tous ces problèmes réglés, le plus complexe est à venir. En effet, l'assemblage des pièces a été très complexe. Il fallait un support solide soutenu par un moteur qui permet de faire pivoter le support et donc la sarbacane. Les fixations ont été compliquées car assembler deux cylindres perpendiculairement est pas si simple car l'axe n'est pas fixe et donc les pièces bougent. De plus, il fallait faire en sorte que la sarbacane puisse arriver à la verticale pour pouvoir la recharger. Cette idée était beaucoup complexe, le système de rechargement a été laissé sur le côté.

---

## 6.Conclusion

En conclusion le projet fonctionne. Même si tout ce qui était prévu a pris du temps car il y avait beaucoup de choses à gérer, qui occupaient beaucoup de temps et il fallait être réactif pour rebondir aux problèmes survenus le long du projet. Ce projet a permis d'apprendre à travailler en groupe afin de le développer et de rendre ce travail à une date finale donc à gérer le temps.

## Perspective

En perspective on voudrait améliorer l'esthétique de notre projet. En effet cela semble plus être un prototype. Il faudrait aussi remplacer les liaisons des moteurs car celles-ci sont faibles et se déconnecter à certain moment. De plus, tous les branchements sont apparents et il faudrait faire une boîte pour les masquer. Nous remercions M.Masson et M.Ferrero de nous avoir encadrés dans ce projet. C'est eux qui nous ont donné la possibilité d'arriver à ce rendu.

