

CARTE RADIOSHIELD PILOTÉE PAR ARDUINO

Envoyer une trame

Dans ce document, est présenté un projet qui permet uniquement d'envoyer des trames APRS depuis une position fixe. C'est basée autour d'une **carte Arduino** (figure 1 à gauche) sur laquelle vient s'ajouter la **carte Radioshield** (figure 1 à droite).

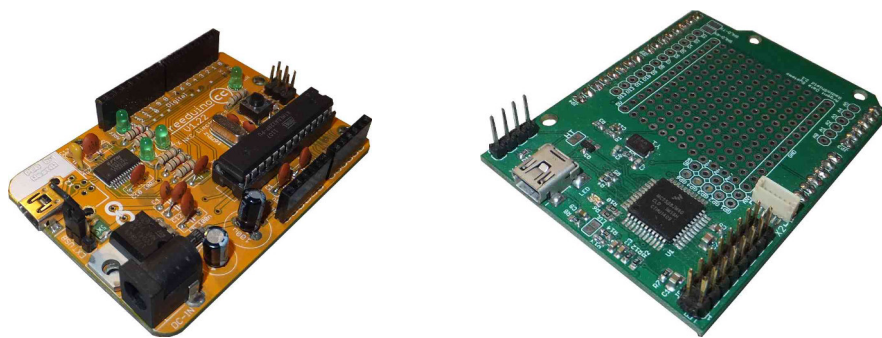


Figure 1

Materiel nécessaire :

- Carte Arduino
- Carte Radioshield (modèle composants traversants ou CMS)
- Logiciel de programmation Arduino
- Emetteur (Baofeng UV5R par exemple)

La **carte Arduino** possède le programme qui permet de piloter la carte Radioshield par des commandes et lui envoie des informations à convertir en AX25 puis en AFSK.

La **carte Radioshield** est un « shield Arduino » qui est conçu pour faire des projets autour de la radio comme générer des trames AX25 pour pratiquer l'APRS. Elle fonctionne en se connectant à la carte de développement Arduino, même dimension de carte, même espacement entre les broches (c'est fait pour).

Dans le **programme**, il y a les commandes qui permettent de donner des ordres à la carte RS. C'est aussi dans le programme que contient les informations relatives à la station (indicatif, position etc).

Schéma de principe

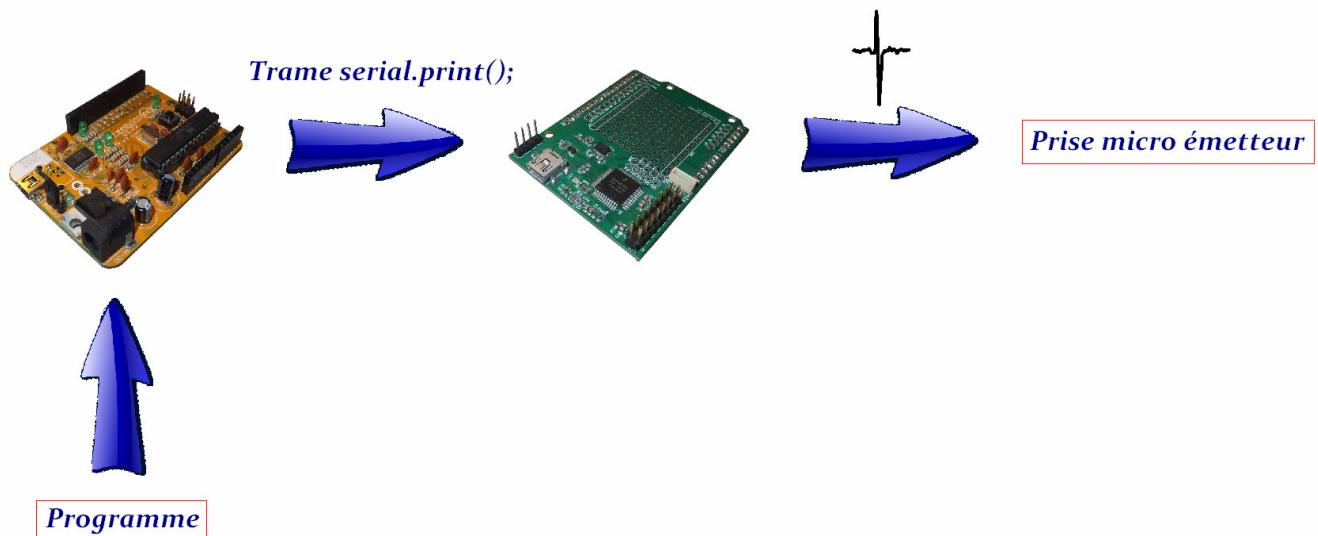


Figure 2

Le programme est compilé dans l'ATMEGA de l'Arduino. Le microcontrôleur envoie dans une trame USB les données dans la carte RS qui restitue en AFSK (audio) la trame APRS. Le signal va ensuite sur l'entrée micro de l'émetteur. La trame est donc envoyée sur l'air.

Conseil : Débrancher la carte RS lors de la programmation de la carte Arduino.



Figure 3, carte Radioshield connectée à la carte Arduino

Le programme

Le programme est relativement simpliste pour ce projet : Une dizaine de lignes ! On retrouve un élément de base dans la programmation Arduino : **Serial.print()**. Toutes les informations envoyées vers la carte RS est dans cette instruction. **Serial.begin(4800)** permet de fixer la vitesse de transmission à 4800 bauds.



```
void setup()
{
  Serial.begin(4800);
}

void loop()
{
  Serial.print("L255\n");
  Serial.print("MF4FGV\r\n");
  Serial.print("!=4848.12N/00501.53E-");
  Serial.print("Test projet FGV-Tracker");
  Serial.print("\r\n");
  delay(5000);
}
```

Figure 4

Le programme attends 5 secondes avant de reprendre à *void loop()* avec l'instruction **Delay(5000)**. C'est à dire qu'une trame sera envoyée toute les 5 secondes.

Les commandes et les symboles

```

Programme_RadioShield_Fixe
void setup()
{
  Serial.begin(4800);
}

void loop()
{
  Serial.print("L255\n");
  Serial.print("MF4FGV\r\n");
  Serial.print("!=4848.12M/00501.53E-");
  Serial.print("Test projet FGV-Tracker");
  Serial.print("\r\n");
  delay(5000);
}
    
```

M : Désigne que ce qui suit est l'indicatif

! : Ordonne à la carte RS que ce qui suit est à envoyer

Symbole désignant l'icône sur la carte (ici, une maison jaune sur APRS.FI)

```

Programme_RadioShield_Fixe
void setup()
{
  Serial.begin(4800);
}

void loop()
{
  Serial.print("L255\n");
  Serial.print("MF4FGV\r\n");
  Serial.print("!=4848.12M/00501.53E-");
  Serial.print("Test projet FGV-Tracker");
  Serial.print("\r\n");
  delay(5000);
}
    
```

L : Désigne le volume de sortie (échelle de 0 à 255)

Commande de fin de ligne (CRLF)

Figure 5

Les informations

```

Programme_RadioShield_Fixe
void setup()
{
  Serial.begin(4800);
}

void loop()
{
  Serial.print("L255\n");
  Serial.print("MF4FGV\r\n");
  Serial.print("!=4848.12M/00501.53E-");
  Serial.print("Test projet FGV-Tracker");
  Serial.print("\r\n");
  delay(5000);
}
    
```

Indicatif

Position (latitude et longitude)

Commentaire

Figure 6

Connecter au poste

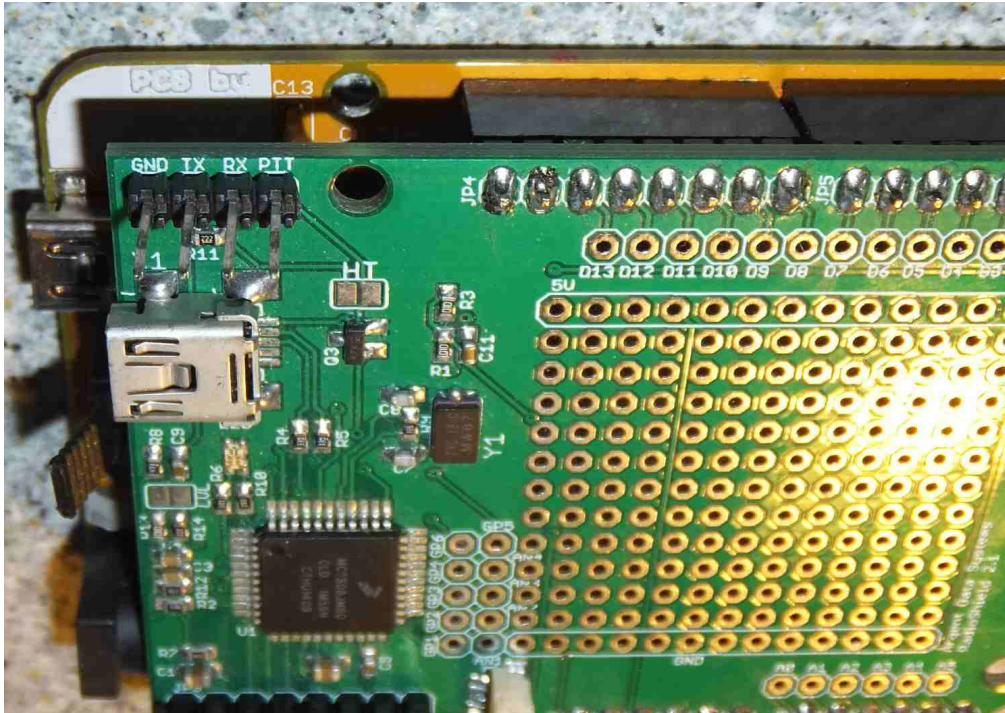


Figure 7

Connecter le GND au GND du poste, la sortie TX à l'entrée micro et le PTT au PTT. Pas difficile HI.

Lors de l'envoi d'une trame, le GND broche 1 et le PTT broche 4 sont électriquement reliés.

CONCLUSION

En faisant les étapes suivantes, il est possible d'envoyer une trame APRS sur l'air :

- Recopier les lignes du programme *figure 4*
- Compiler ces lignes sur le μC de la carte de développement Arduino
- Connecter la carte RS sur la carte de développement
- Connecter un pocket en mode VOX sur la carte RS (bornier 4 pins)

Peut-être que le programme peut être mieux écrit, de manière plus logique. Ce projet permet en outre d'éviter d'allumer un ordinateur pour juste envoyer des trames APRS toutes les 30 minutes. Il suffit juste d'alimenter la carte Arduino par le connecteur d'alimentation en +5VDC et d'allumer un poste. La LED de la carte Radioshield s'allume rouge lors de l'envoi d'une trame.