

|                                      |   |
|--------------------------------------|---|
| <b>Objectifs</b>                     | - Comprendre et mettre en œuvre une transmission LoRa   |
| <b>Prérequis</b>                     | - KRL19 Cours Introduction à la modulation LoRa   |
| <b>Ressources à disposition</b>      | <ul style="list-style-type: none"> <li>- PC station de travail sous Windows</li> <li>- Analyseur Logique</li> <li>- Cours Introduction à la modulation LoRa</li> <li>- Carte de développement ATMEGA328</li> <li>- <i>Modem LoRa empilable "Dragino"</i></li> </ul> |
| <b>Capacité / compétences visées</b> | <ul style="list-style-type: none"> <li>- S7 Réseaux, télécommunications et modes de transmission</li> <li>- S7.3 Protocoles de bas niveau : Liaisons RS232C, RS485, SPI, etc.</li> <li>- S7.4 Transmission sans fil</li> </ul>                                      |

## 1 Mise en situation

### 1.1 Présentation du contexte

Les objets connectés utilisent principalement, pour envoyer leurs messages, quatre techniques économes en énergie qui sont : **ZigBee** (ZigBee Alliance), **Sigfox** (modem et collecte par la société Sigfox) , **Lora** (Semtech), **nRF24L01** (Nordic Semiconductor).

La portée du modem **LoRa** peut atteindre une dizaine de km.

La bande de fréquence utilisée étant publique, il faut non seulement s'astreindre à une utilisation raisonnable du canal, mais aussi prévoir un moyen de contourner les collisions, et de reconnaître l'émetteur du message.

### 1.2 Problématique

Dans ce TP vous allez :

- Mettre en œuvre un modem Lora de Semtech : le modem SX1276 monté sur le shield Dragino.
- Déterminer un protocole permettant d'identifier l'objet émetteur.
- Déterminer un moyen de contourner les collisions.

### 1.3 Application

La station météo connectée doit se réveiller toutes les demi-heures pour mesurer les conditions météo et les envoyer au serveur. Ce TP vous permettra, en étendant votre protocole, de transmettre les relevés des capteurs sur une grande distance.

## 2 Travail à faire

### 2.1 Organisation

Vous travaillez en équipe de 3 à 4 personnes.

Certaines tâches peuvent être réalisées en parallèle, chaque étudiant réalise alors une tâche différente.

Certaines tâches nécessitent d'être réalisées en séquence. Plutôt qu'attendre, vous pouvez commencer ou terminer une autre tâche ou consulter le cours associé au TP.

### 2.2 Lancement du projet sur GitHub et prise en main du matériel

|  |   |   |
|--|---|---|
| <p align="center"><b>Création du projet</b></p> <ul style="list-style-type: none"> <li>▪ Dans Github Desktop créez un nouveau dépôt <i>VotreEquipe-LoRa</i>.</li> <li>▪ Publiez en gardant ce dépôt privé, avec un readme.me par défaut.</li> <li>▪ Invitez les autres membres de l'équipe.<br/>Invitez votre professeur.</li> </ul> | <p align="center"><b>Montage du shield LoRa</b></p> <ul style="list-style-type: none"> <li>▪ Téléversez le programme blink sur la carte Arduino.</li> <li>▪ Débranchez l'Arduino</li> <li>▪ Embrochez le shield LoRa</li> </ul> | <p align="center"><b>Récupération des sources</b></p> <p>Clonez ou téléchargez le projet<br/> <a href="https://github.com/DaKprofSNir/LoRaDemo">https://github.com/DaKprofSNir/LoRaDemo</a></p>   |
| <p align="center"><b>Sources du programme</b></p> <ul style="list-style-type: none"> <li>▪ Copiez les répertoires de <i>LoRaDemo</i> dans le projet <i>VotreEquipe-LoRa</i>.</li> <li>▪ Publiez</li> </ul>   | <p align="center"><b>Documentation</b></p> <ul style="list-style-type: none"> <li>▪ Créez un répertoire .\doc</li> <li>▪ Copiez-y les fichiers pdf de KRL19</li> <li>▪ Publiez</li> </ul>                                       | <p align="center"><b>Personnalisation</b></p> <ul style="list-style-type: none"> <li>▪ Indiquez les noms des participants dans le fichier README.md</li> <li>▪ Publiez</li> </ul>   |
| <p align="center"><b>Validation du montage</b></p> <ul style="list-style-type: none"> <li>▪ Dans LoRaDemo.ino Remplacez "Equipe Prof" par votre équipe.</li> <li>▪ Publiez</li> <li>▪ Téléversez</li> <li>▪ Affichez la console</li> <li>▪ En cas d'erreur, appelez votre professeur.</li> </ul>                                     | <p align="center"><b>Observation de la transmission</b></p> <ul style="list-style-type: none"> <li>▪ Les transmissions reçues sont affichées au téléprojecteur. Retrouvez-vous la vôtre ?<br/><br/>.....</li> </ul>             | <p align="center"><b>LoRaDemo.ino</b></p> <ul style="list-style-type: none"> <li>▪ Identifiez la ligne initialisant le modem à 868 MHz.</li> <li>▪ Ouvrez le document KRL19 Cours La modulation LoRa.pdf</li> <li>▪ Quelles sont les autres bandes de fréquence possibles ?<br/><br/>.....</li> </ul> |
| <p align="center"><b>Id unique pour l'équipe</b></p> <ul style="list-style-type: none"> <li>▪ Pour distinguer vos futures communications, convenez d'un octet qui sera propre à l'équipe.</li> <li>▪ Astuce : Préférez un chiffre ou une lettre du code ASCII</li> <li>▪ ID de l'équipe :<br/><br/>.....</li> </ul>                  | <p align="center"><b>Donnez l'algorithme général de transmission d'une trame</b></p> <p>.....</p> <p>.....</p> <p>.....</p>   |   |

## 2.3 Conception d'un protocole de communication

Vous travaillerez sur un programme plus complet : le projet LoRaSender.ino

### Premier essai de LoRaSender.ino

- Ouvrez LoRaSender.ino dans l'IDE ARduino
  - Téléversez
  - Ouvrez le moniteur série
  - Le montage affiche-t-il un compte-rendu d'envoi de messages ?
- .....

### Votre identificateur d'équipe

- Recherchez dans la fonction loop() le ligne `Trame.id = 0x30;`
- Remplacez-y la valeur 0x30 par votre id d'équipe.
- Téléversez
- Publiez

### Validation de la transmission

- Observez l'apparition de messages signés de votre id au vidéo-projecteur.
- .....

### 2.3.1 Observation du programme

#### Le format de la trame

- La trame émise est une structure C++ que vous pouvez modifier en ajoutant des champs.
  - Quel est le type de cette structure ?
- .....

- Quelle est la variable qui contient la trame ?
- .....

#### La communication avec le modem

- Donnez les trois fonctions utilisées pour communiquer :
- .....
- .....
- .....

#### Le remplissage de la trame

- Comment accède-t-on à un champ de la trame ?
- .....

#### Exercice

- Modifiez le type byte de N pour `uint16_t`
- en Fin Emission LoRa décrémente N.

#### Exercice

- N étant devenu un `uint16_t`,
- Adaptez le remplissage de la trame pour un nombre dont le maximum est 65535.

### 2.3.2 Problématique "Contourner les collisions"

Vous avez certainement observé que certains de vos messages n'étaient pas reçus.

#### Proposez différentes stratégies pour gérer les collisions

.....

.....

.....

.....

.....

.....

#### Le champs ts

- Stratégie la plus simple :
- .....

Modifiez le programme de façon à ce que ts représente ...

.....

## 2.4 Étendre le protocole

### Problématique :

On désire ajouter à ce protocole les mesures effectuées par un capteur.

#### Créez un nouveau projet pour cette version étendue

- Copiez le dossier `.\LoRaSender` vers `.\LoRaSenderEtendu`
- Renommez dans `.\LoRaSenderEtendu` `LoRaSender.ino` en `LoRaSenderEtendu.ino`
- Publiez

#### Proposez un champ pour un bouton poussoir

```
struct __attribute__((packed))
trame_t {

    uint8_t id;

    uint8_t ts;

    char data[8];

    .....
};
```

#### Proposez un champ pour une lecture analogique sur A0

```
struct __attribute__((packed))
trame_t {

    uint8_t id;

    uint8_t ts;

    char data[8];

    .....
};
```

#### Proposez des champs pour les deux valeurs renvoyées par un capteur i2c DHT12

```
struct __attribute__((packed))
trame_t {

    uint8_t id;

    uint8_t ts;

    char data[8];

    .....
    .....
};
```

#### Codez la lecture d'une LDR sur la broche A0

Dans `setup()` :

.....

.....

Dans `loop()` :

.....

#### Codez la lecture du capteur DHT12 (optionnel)

En vous servant de la librairie  
[https://github.com/DaKprofSNir/DaK\\_DHT12](https://github.com/DaKprofSNir/DaK_DHT12)

## 2.5 Décoder le nouveau protocole dans le récepteur

Vous disposez du code source du récepteur dans le répertoire `.\LoRaReceiver`

C'est ce programme dont vous observez la sortie au vidéoprojecteur.

Vous pouvez le perfectionner en y utilisant la même définition de trame que celle de votre programme.

### 3 Questions de cours

**LoRa concerne les réseaux**

- ☐ B2C
- ☐ M2M
- ☐ B2B

**Donnez les trois caractéristiques communes aux modulations LoRa, Sigfox, nRF24L01**

.....

.....

.....

**Quelles sont les bandes de fréquences libres (dites fréquences régionales) utilisables en Europe ?**

.....

**Dans la pratique, quelles sont nos limites en utilisation de la bande libre ?**

.....

**Dans quelle couche du modèle OSI se situe LoRa ?**

.....

**Dans quelle couche du modèle OSI se situe LoRaWan ?**

.....

**Quelles sont les sociétés qui développent LoRaWan ?**

.....

**Les communications LoRa et LoRaWan sont-elles chiffrées ?**

.....

**Comment appelle-t-on le dispositif qui reçoit et concentre les émissions provenant des IOT pour les relayer par exemple vers un serveur sur Internet ?**

.....

**La consommation d'un modem LoRa, la consommation d'une LED ?**

.....

### 4 Bilan et conclusion

.....

.....

.....

.....

.....