

# Mise en œuvre d'une carte Arduino MKR FOX 1200

## Connexion de la carte à une machine de dev



Connecter le port miniUSB de la carte à un port USB de l'ordinateur de dev.  
La LED verte près du port USB de l'Arduino s'allume.

Suivant la machine de dev (Linux, Mac, Windows) le pilote du port USB apparait dans le répertoire /dev de la machine ou sur un port COM (nativement ou après installation d'un utilitaire).

```
>ls /dev
```

cu.jpdmsiPhone-WirelessiAP	rdisk1s3
cu.usbmodem1461	rdisk1s4
disk0	sdt
disk0s1	stderr
disk0s2	stdin
disk0s3	stdout
disk1	sysrtrace
disk1s1	tty
disk1s2	tty.Bluetooth-Incoming-Port
disk1s3	tty.MALS
disk1s4	tty.SOC
dtrace	tty.jpdmsiPhone-WirelessiAP
dtracehelper	tty.usbmodem1461
fbt	ttyp0

Le port USB de la carte se trouve sur **cu.usbmodem1461** ( **tty.usbmodem1461** ).

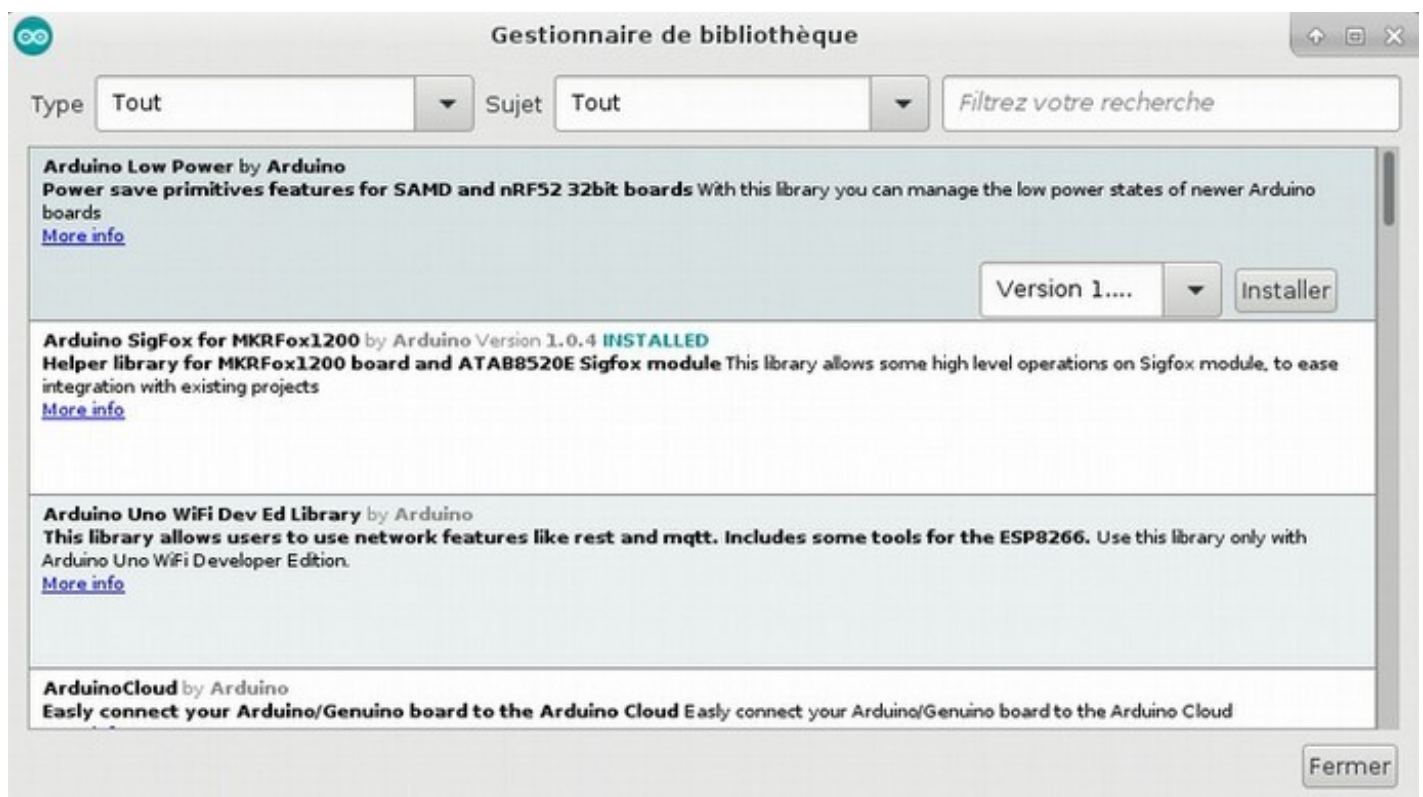
# Installation et configuration de l'outil Arduino de développement

- Le télécharger
- L'installer
- Configuration (dans le menu « outils ») :



1. On configure le type de carte
2. On configure le port d'accès à la carte.

- On installe les librairies nécessaires :



Attention : Ouvrir dans « outils » ou en cliquant sur la loupe en haut à droite le moniteur série.



## Tester un premier programme qui blink la led par défaut

```
// the setup function runs once when you press reset or power the board
void setup() {
    Serial.begin(9600);
    while(!Serial) {};
    Serial.println("Application Arduino");
    // initialize digital pin LED_BUILTIN as an output.
    pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
    digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the
voltage level)
    delay(1000); // wait for a second
    digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the
voltage LOW
    delay(1000); // wait for a second
}
```

## Tester la librairie SIGFOX

```
#include <SigFox.h>

/*
    Test de la librairie SIGFOX
*/

void setup() {
    Serial.begin(9600);
    while(!Serial) {};
    Serial.println("Application Arduino");
    if (!SigFox.begin()) {
        Serial.println("Shield error or not present!");
    }
    sigfoxID();
}

// the loop function runs over and over again forever

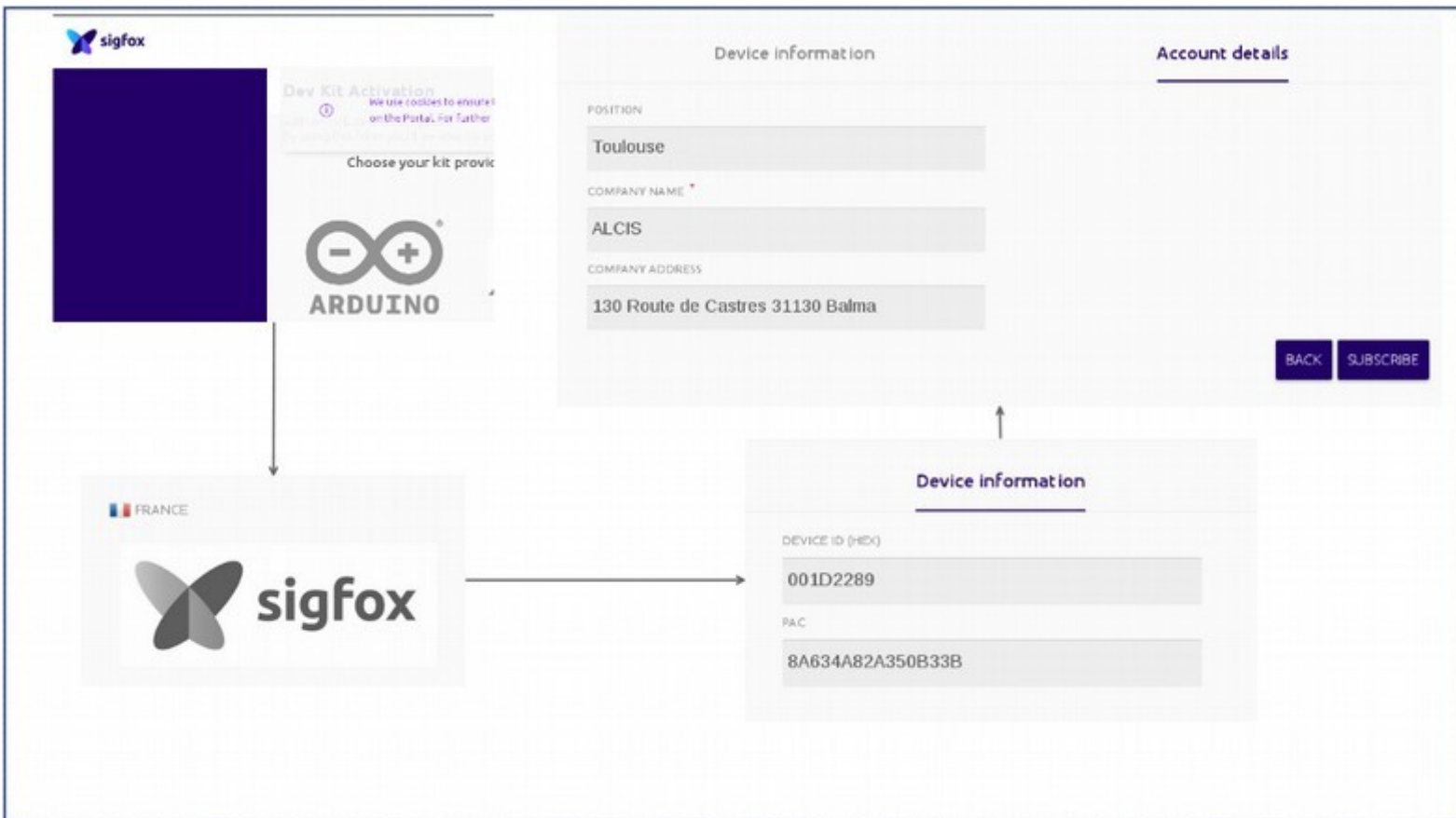
void loop() {
    while(1);
}

void sigfoxID() {
    // Affichage des informations du module SIGFOX
    Serial.println("MKRFox1200 Sigfox informations");
    Serial.println("Version = " + SigFox.SigVersion());
    Serial.println("ID = " + SigFox.ID());
    Serial.println("PAC = " + SigFox.PAC());
    Serial.println("\nTemperature interne du module : " +
(String)SigFox.internalTemperature());
    Serial.println("Pour enregistrer votre module :
https://backend.sigfox.com/activate");
    delay(100); // 100ms (le temps de l'affichage)
    // Send the module to the deepest sleep
    SigFox.end();
}
```

## Résultat sur le moniteur série :

```
Application Arduino
MKRFox1200 Sigfox informations
Version = 2.3
ID = 001D188E
PAC = 60A25D16E246E22A
|
Temperature interne du module : 20.00
Pour enregistrer votre module : https://backend.sigfox.com/activate
```

## Inscription du module chez SIGFOX



**Type:** Basic

**Name:** Marcon Stephane (1)

**Description:** Marcon Stephane EVK

**Timezone:** UTC

**Business contact email:** jpdumas@btslivh.eu

**Billable:** true

**Client name:** ALCIS

**Client address:** 130 Route de Castres 31130 Balma

**Parent group:** SIGFOX\_France EVK

**Creation date:** 2018-02-14 16:07:44

**Created by:** Marcon Stephane

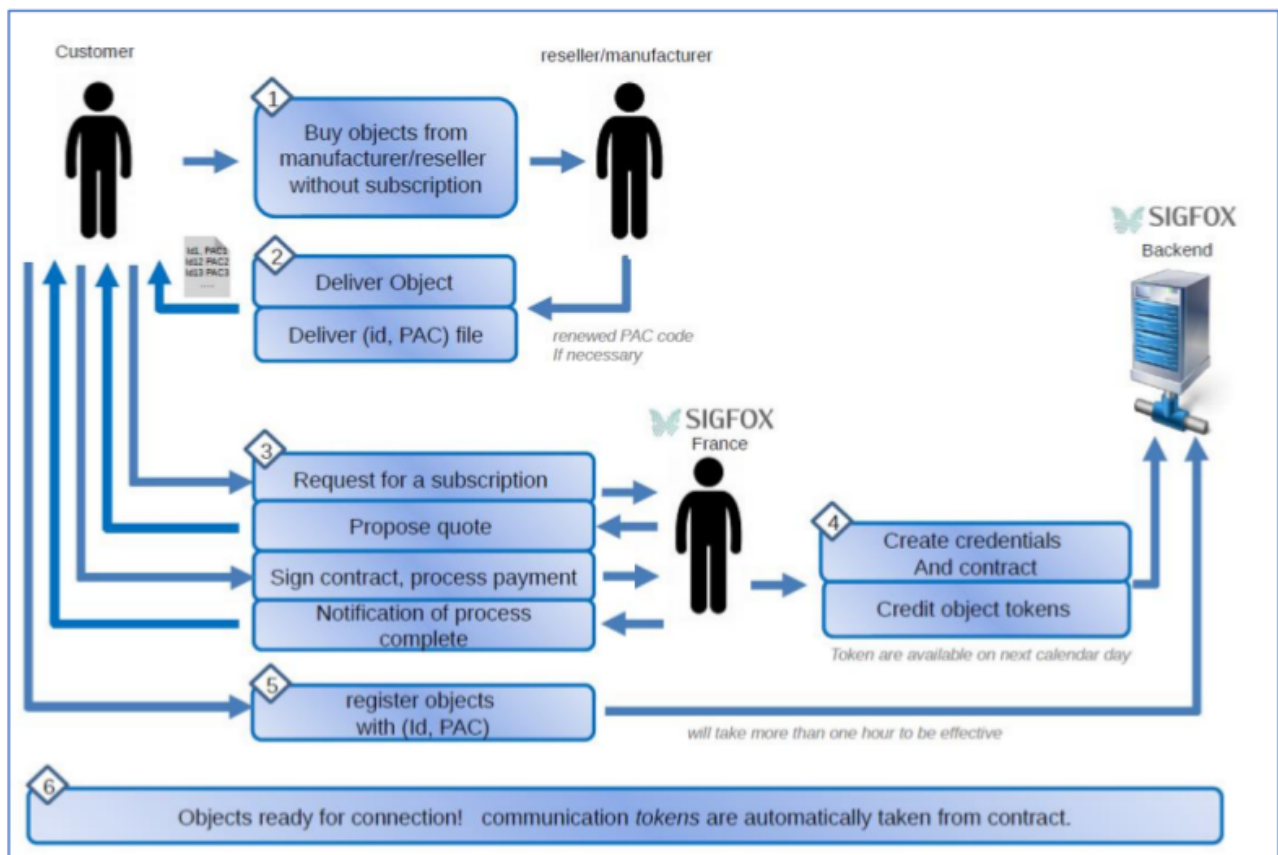
**Last edition date:** 2018-02-14 16:07:44

**Last edited by:** Marcon Stephane

**Mail de connexion:** [jpdumas@btslivh.eu](mailto:jpdumas@btslivh.eu)

**Mot de passe:** Alfox31#

## Enregistrement d'un module SIGFOX en général





## Envoi d'un message avec acquittement du serveur SIGFOX

```
void sendStringAndGetResponse(String str) {
  /*
    if (!SigFox.begin()) { //ensures SigFox is connected
      Serial.println("Shield error or not present!");
      return;
    }
  */
  SigFox.begin(); // Start the module
  delay(100); // Wait at least 30mS
  SigFox.status(); // Clears all pending interrupts
  delay(1);
  SigFox.beginPacket();
  SigFox.print(str);
  int ret = SigFox.endPacket(true); // send buffer to SIGFOX network
                                     // and wait for a response

  if (ret > 0) {
    Serial.println("No transmission");
  } else {
    Serial.println("Transmission ok");
  }
  Serial.println(SigFox.status(SIGFOX));
  Serial.println(SigFox.status(ATMEL));
  if (SigFox.parsePacket()) {
    Serial.println("Response from server:");
    while (SigFox.available()) {
      Serial.print("0x");
      Serial.println(SigFox.read(), HEX);
    }
  } else {
    Serial.println("Could not get any response from the server");
    Serial.println("Check the SigFox coverage in your area");
    Serial.println("If you are indoor, move near a window");
  }
  Serial.println();
  SigFox.end();
}
```

## Console :

```
Application Arduino
MKRFox1200 Sigfox informations
Version = 2.3
ID = 001D188E
PAC = 60A25D16E246E22A

Température interne du module : 17.00
Pour enregistrer votre module : https://backend.sigfox.com/activate
Type the message to be sent
Sending 123456789012
Getting the response will take up to 50 seconds
The LED will blink while the operation is ongoing
Transmission ok
OK
PA OFF
Response from server:
0x0
0x0
0x12
0x37
0x0
0x0
0xFF0x89
```

## Envoi d'un message du serveur SIGFOX sans acquittement

```
void sendString(String str) {
    SigFox.begin(); // Start the module
    delay(100); // Wait at least 30mS after first
configuration (100mS before)
    SigFox.status(); // Clears all pending interrupts
    delay(1);
    SigFox.beginPacket();
    SigFox.print(str);
    int ret = SigFox.endPacket(); // send buffer to SIGFOX network
    if (ret > 0) {
        Serial.println("No transmission");
    } else {
        Serial.println("Transmission ok");
    }
    Serial.println(SigFox.status(SIGFOX));
    Serial.println(SigFox.status(ATEL));
    SigFox.end();
}
```

```
Application Arduino
MKRFox1200 Sigfox informations
Version = 2.3
ID = 001D188E
PAC = 60A25D16E246E22A

Température interne du module : 20.00
Pour enregistrer votre module : https://backend.sigfox.com/activate
Type the message to be sent
Sending aaaazzzzzee
Transmission ok
OK
PA OFF
```



# Lecture des messages sur le serveur SIGFOX

## Accès au backend sigfox

<https://backend.sigfox.com/device/list>



DEVICE

DEVICE TYPE

USER

GROUP

DEVICE : information et configuration de chaque module par son ID

DEVICE TYPE : information et configuration par type de module

USER : information et configuration des utilisateurs identifiés

GROUP : information et configuration de groupe de DEVICE TYPE

Voir les messages : Sous DEVICE sélectionner le module en cliquant sur son ID.

Average Rssi	Average SNR	Communication status	Device type	Id	Last seen	Name	Token state
			Arduino ALCIS kit	1D2289	N/A	Device 001D2289	

Liste des messages recus (2 messages avec acquittement, 3 messages sans ACK)

page 1				
Time	Data / Decoding	Location	Link quality	Callbacks
2018-02-14 16:26:23	313233343536373839303132			
2018-02-14 16:25:27	313233343536373839303132			
page 1				

La localisation n'est pas précise tant que l'on installe pas de callback.

## Contacter la hotline SIGFOX

### Sigfox France Service Desk : numéro de téléphone

Créée par Christophe ALLASIA, dernière modification par Jon Regueiro le juin 01, 2017

Tout client français désireux d'avoir un support personnalisé aura maintenant la possibilité de nous contacter via le numéro de téléphone suivant :

Any French customer wishing to have a personalized assistance will now have the possibility to contact us via the following number:

**0 826 904 905**

Service 0,20 € / min  
+ prix appel