



# Câblage et configuration d'un nœud ConneCSenS pour y interfacer un capteur de température type PT

Révision 1



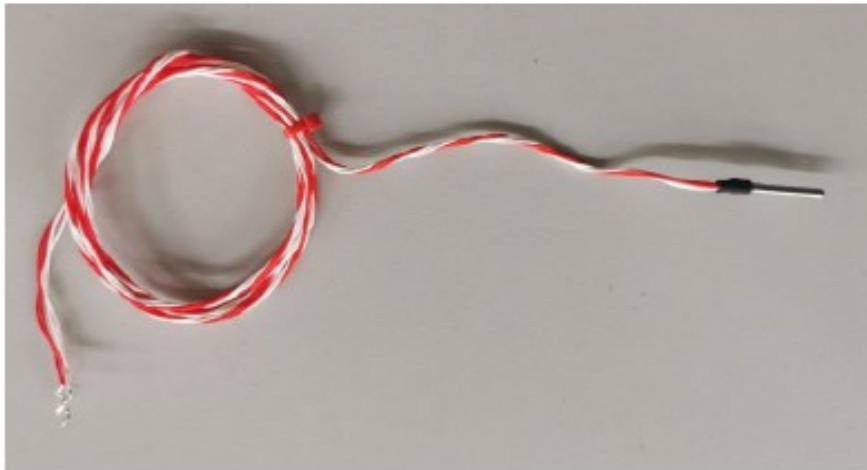
## 1 Objectif de ce document

Ce document explique comment interfacer une sonde platine résistive avec le nœud ConneCSenS. Le câblage et la configuration du capteur et du nœud sont étudiés.

## 2 La sonde PT100/PT1000

### 2.1 Présentation

Une sonde PT est une sonde platine résistive. Cette sonde possède une résistance à 0°C, c'est la valeur après PT. Par exemple une sonde PT100 aura une résistance de 100 Ohm à 0°C.



### 2.2 Câblage

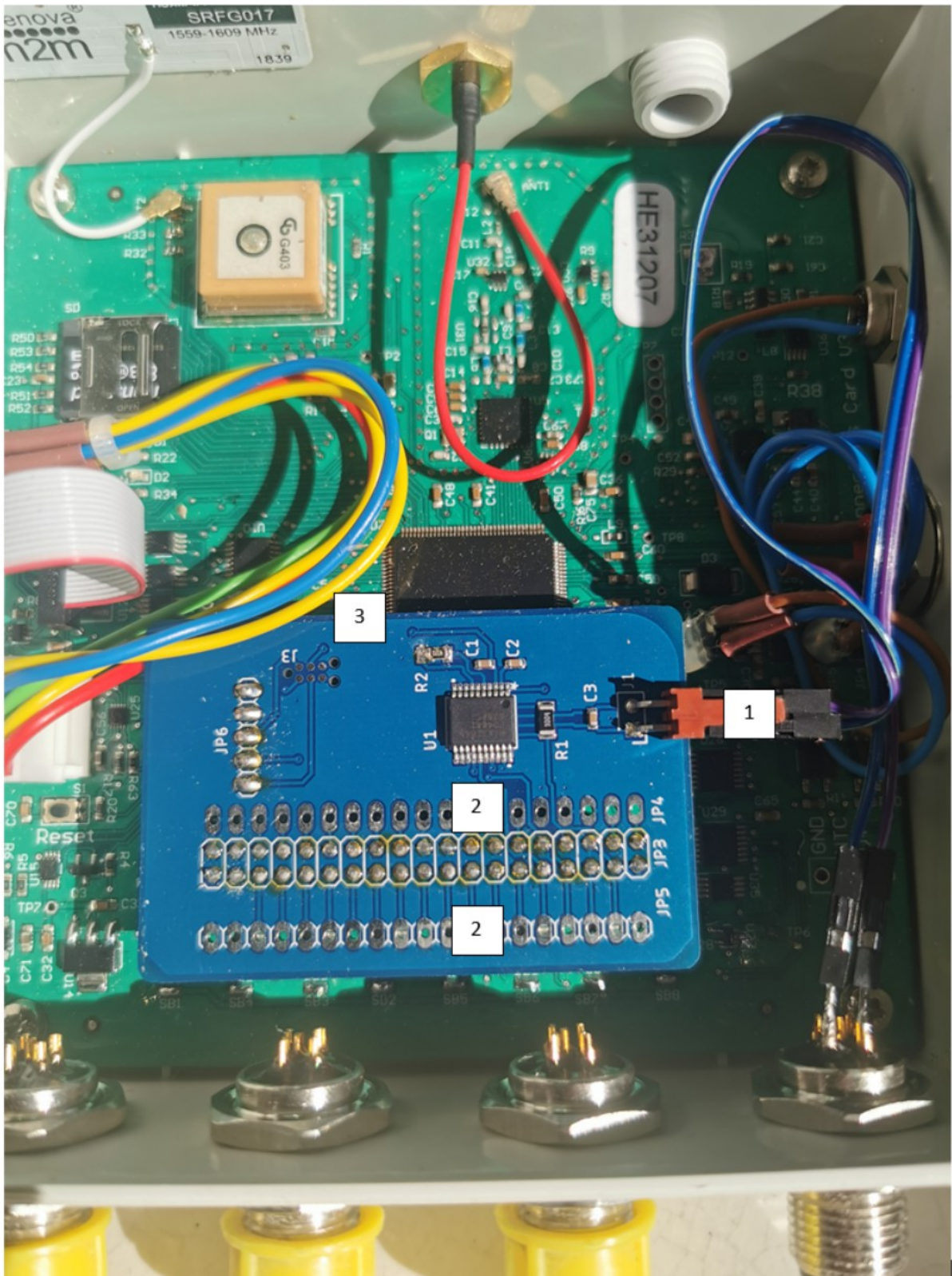
Le capteur est livré avec un câble qui se termine par des fils nus. Ce chapitre explique comment y monter un connecteur M12 femelle pour le connecter au nœud ConneCSenS. Vous pouvez sauter ce chapitre si votre capteur est déjà équipé de ce connecteur.

Voici un schéma de câblage du capteur vers le connecteur M12 femelle, où le connecteur est vu de derrière (les points noirs sont les plots où les fils sont vissés ou soudés) et le rectangle noir à gauche représente l'extrémité du câble du capteur :



Il existe des méthodes de mesures différentes suivant le nombre de fils. Une sonde PT1000 auras 2 fils et une sonde PT100 aura 4 fils. En réalité la sonde PT100 est moins précise que la sonde PT1000. La méthode 4 fils permet de s'affranchir de la résistance des fils (cette méthode peut ne pas être pertinente selon la longueur des fils).

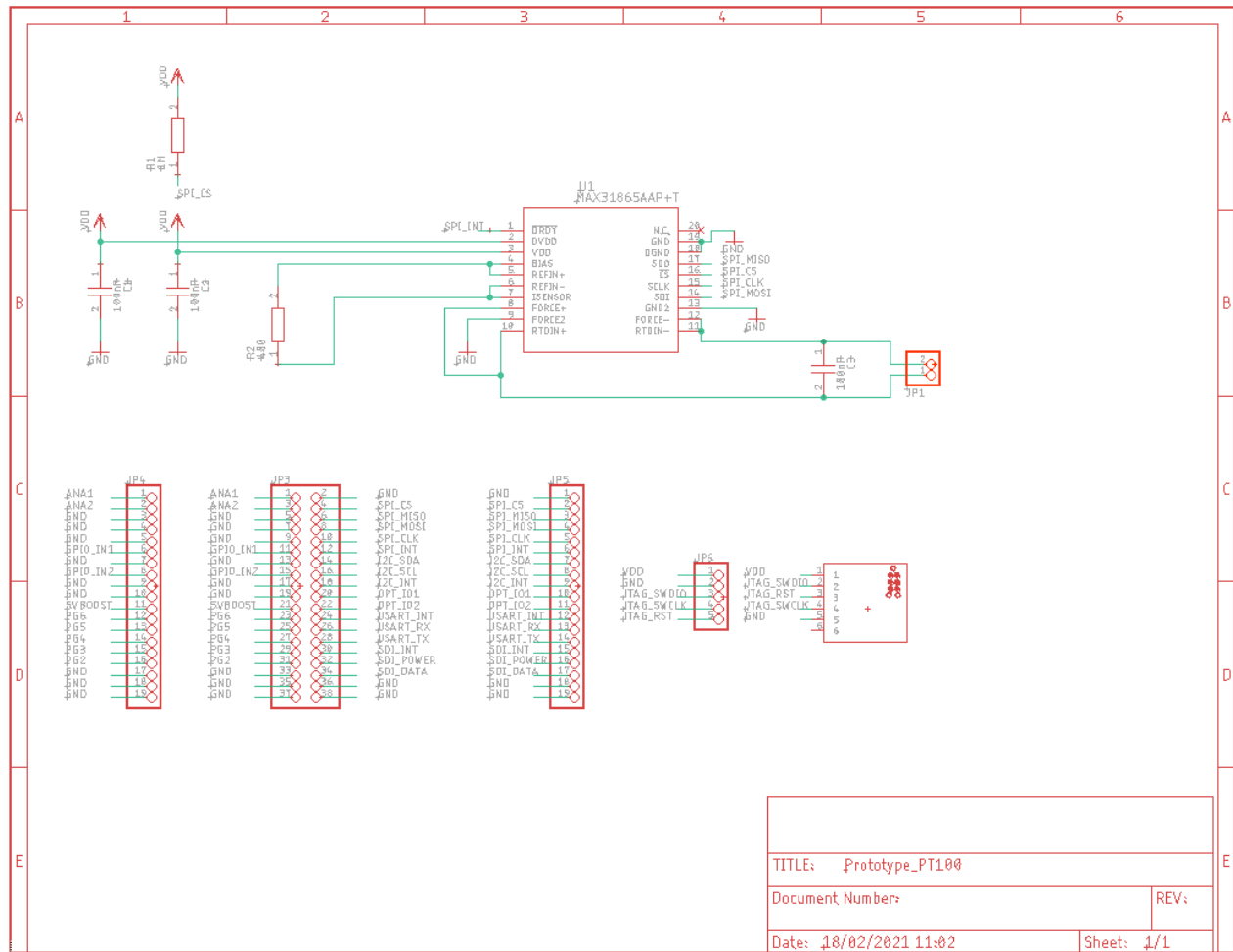
Sur la carte d'adaptation, on retrouve 2 broches pour venir connecter la sonde. La carte vient se placer complètement sur le GPIO. Sur la face supérieure de la carte on retrouve une extension du GPIO sur laquelle nous pouvons venir souder des fils sur les pastilles. La carte communique avec le nœud via une **interface SPI**



On y retrouve en :

- 1 : La sonde connectée au nœud
- 2 : L'extension du GPIO pour venir souder des fils sur les pastilles
- 3 : Le connecteur JTAG déporté

On retrouvera le schéma de routage sur l'image suivante :



## 2.3 Liste de matériel

Composant	Fonction	Référence fabricant	Fournisseur	Quantité
MAX31865	Capteur température	MAX31865AAP+T	Farnell	1
Connecteur 5 contacts femelle pour CI	Connecteur 5 contacts femelle pour CI	W3481105TRC	RS	1
Connecteur 38 contacts femelle pour CI	Connecteur 38 contacts femelle pour CI	SSW-119-01-T-D	RS	1
Résistance 402 Ohms	Résistance conditionnement capteur	CRCW1206402R FKEA	RS	1
Résistance 1M Ohms	Pull-up sélection esclave	CRCW12061M00 FKEAHP	RS	1
Condensateur 100nF	Condensateurs de découplage	06035C104K4T4 A	Digikey	3

**Les valeurs du condensateur de 100 nF et de la résistance de 402 Ohms ne sont valable que dans le cas où l'on vient brancher une PT100. Pour une PT1000 on utilisera une résistance de précision de valeur 4 KOhms et un condensateur de 10 nF.**

## 3 Configuration logicielle du nœud ConnecSenS

Pour que ce capteur fonctionne il faut utiliser la version firmware 1.4.0 minimum. Il est conseillé d'utiliser la dernière version stable de la branche 1.X.X.

Pour que le nœud puisse utiliser ce capteur il faut lui déclarer qu'un capteur PT est connecté. Ceci se fait au moyen du fichier de configuration du nœud.

Connectez votre nœud en USB à un PC. Il doit être reconnu comme un périphérique de stockage de masse (un disque ou une clef USB) par votre ordinateur. Lorsque viendra le moment de débrancher le nœud de votre ordinateur, n'oubliez pas au préalable de démonter (d'éjecter) le disque correspondant à votre nœud avec votre système d'exploitation (en général faites un clic droit sur l'icône du disque et choisissez l'option « éjecter » ou « démonter »).

Le fichier de configuration à changer ou à créer s'appelle (chemin Windows) : <lettre\_du\_noeud>:\env\config.json. Il s'agit d'un fichier texte, qui contient des données encodées au format JSON. Il est possible de l'ouvrir avec un éditeur de texte simple,

comme NotePad, ou avec des éditeurs plus évolués qui proposent au minimum de la coloration syntaxique pour faciliter la lecture. Le choix de l'éditeur est libre, faites selon vos préférences et vos possibilités.

La trame du fichier de configuration à écrire (ou à copier) est la suivante :

```
1  {
2      "name": "Y1207",
3      "experimentName": "YesitisEngie",
4      "addGeoPosToAllReadings": false,
5
6      "logs": {
7          "defaultLevel": "DEBUG"
8      },
9
10     "network": {
11         "type": "LoRaWAN",
12         "devEUI": "434E535308E31207",
13         "appEUI": "A87F1729918331B4",
14         "appKey": "436f6e6e656353656e5330315a415455",
15         "periodMn": 15,
16         "useAdaptativeDataRate": false,
17         "dataRate": 5
18     },
19
20     "sensors": [{
21         "name": "RTD1",
22         "type": "MAX31865",
23         "periodMn": 15,
24         "RTDType": "PT100",
25         "nbWires": 2,
26         "RrefOhms": 402.0,
27         "use5VwhenActive": false
28     }],
29
30     "time": {
31         "syncMethod": "manualUTC",
32         "GPS": {
33             "periodDay": 100
34         },
35         "manualUTC": {
36             "year": 2021,
37             "month": 05,
38             "day": 19,
39             "hours": 09,
40             "minutes": 38,
41             "seconds": 0
42         }
43     }
44 }
```

### 3.1 Configuration générale

La valeur du paramètre `name`, ici `"Y1207"`, est le nom assigné au nœud. Choisissez ce nom selon vos besoins et vos contraintes. Sa longueur ne doit pas dépasser 31 caractères. Il est préférable de choisir un nom aussi court que possible, car il sera envoyé par ondes radios.

Il est conseillé de laisser la section `debug` telle qu'elle est donnée dans l'exemple de configuration. Ce paramètre peut être passé en INFO pour un système stable et éprouvé (il y aura moins de détails d'écrit dans le fichier LOG, ce qui augmentera l'autonomie du nœud).

### 3.2 Configuration réseau

La section `network` contient les informations du réseau LoRaWAN.

Le paramètre `type` indique le type de réseau de communication utilisé par le nœud. Sa valeur est obligatoirement `"LoRaWAN"`.

La valeur du paramètre `devEUI` doit être changée pour chaque nœud. Elle est l'identifiant unique du nœud dans le réseau LoRaWAN. Cet identifiant vous est donné par le gestionnaire du réseau et il devrait figurer sur une étiquette collée sur le nœud.

Les paramètres `appEUI` et `appKey` permettent de se connecter au réseau LoRaWAN et vous sont communiqués par le gestionnaire du réseau. Ils devraient être communs à tous les nœuds en votre possession, ou à tout le moins à tous les nœuds d'un même site expérimental. Contactez le gestionnaire de réseau LoRaWAN en cas de doute.

Le paramètre `periodHr` ou `periodMn` règle la fréquence de transmission des données collectées par le nœud. Son unité est ici en heures, et sa valeur est un nombre réel positif. Changez cette valeur selon les conditions de votre site expérimental, selon vos besoins et vos contraintes. Il est fortement déconseillé d'utiliser une valeur inférieure à 10 minutes, pour des raisons réglementaires et pour éviter de réduire trop fortement l'autonomie énergétique du nœud. Il est conseillé de faire au moins une transmission par jour. Pour que le nœud soit en mesure d'envoyer toutes les données collectées, il faut éviter que la période d'envoi des données soit supérieure à environ 8 fois la période du capteur mesuré le plus souvent. Ce ratio dépend de la qualité de la transmission radio. Si elle est bonne, alors il peut être augmenté. Si au contraire elle est mauvaise, alors il est conseillé de réduire ce ratio. A titre indicatif, dans de bonnes conditions, le nœud est en mesure d'envoyer 250 octets par transmission, contre 40 octets dans de mauvaises conditions. Les données collectées d'un capteur produisent en général entre 10 et 15 octets de données à envoyer. Il vous faut donc trouver une période qui permette au moins d'envoyer toutes les données collectées. Par ailleurs, le nœud est capable de rattraper son retard si des envois précédents ont échoué. Mais encore faut-il qu'il puisse envoyer plus de données que toutes celles collectées depuis le dernier envoi ou la dernière tentative d'envoi. Aussi, il est conseillé de fixer un ratio de 1 à 4 entre la période d'envoi des données collectées et la période du capteur mesuré le plus souvent.



### 3.3 Configuration des capteurs

Vient ensuite la configuration des capteurs au moyen du tableau `sensors`. Chaque élément de cette liste est un objet JSON qui décrit un capteur. Un capteur PT100 est défini dans l'exemple de configuration donné plus haut.

#### 3.3.1 Configuration d'un capteur : paramètres obligatoires

Configurez d'abord le nom du capteur au moyen du paramètre `name`. Dans le fichier d'exemple le nom du premier capteur défini est `"RTD1"`. Mais libre à vous de choisir la valeur qui vous convient le mieux. Cette valeur doit cependant faire moins de 31 caractères de long. Il est ici aussi conseillé de choisir un nom court, car il est envoyé par ondes radios.

Le `type` de ce capteur est obligatoirement `"MAX31865"`

La période de mesure du capteur est fixée, en heures, par le paramètre `periodHr`. Il est possible d'utiliser à la place `periodMn`, `periodDay` ou `periodSec` pour fixer cette période en minutes, jours ou secondes respectivement.

Le `RTDType` de ce capteur est obligatoirement de type PT, mais il n'y a pas de restriction sur la valeur de la sonde à 0°C : `"PT100"` ou `"PT1000"` ou `"PT99.758"`

Le paramètre `RrefOmhs` indique la valeur de la résistance de référence utilisée par le MAX31865 pour mesurer la résistance de la sonde de température. Cette valeur est un nombre réel strictement positif. Le circuit intégré calcule le ratio entre les résistances, de référence et de sonde. Il est donc nécessaire de connaître le plus précisément possible cette valeur. Elle est susceptible de varier d'une carte d'adaptation à l'autre, mais également en fonction de la sonde. Le constructeur conseille par exemple d'utiliser une valeur de 400  $\Omega$  pour des sondes PT100 et de 4 k $\Omega$  pour les sondes PT1000.

**Attention !**

**Avec une version antérieure à la 1.4.2, la résistance à renseigner pour une PT1000 doit être 100 fois inférieure à sa valeur réelle (40.07 au lieu de 4007).**

Le paramètre `Use5VWhenActive` indique si c'est le nœud qui alimente le capteur. Ce paramètre doit être à `"false"` car l'alimentation utilisée est celle du connecteur JTAG.

#### 3.3.2 Configuration d'un capteur : paramètres facultatifs

`ComputeDeltaT` : (optionnel, valeur par défaut : false) indique si la différence de température entre la mesure courante et la précédente doit être calculée et envoyée. À défaut, seule la température mesurée sera enregistrée et transmise.

`FilterFreqHz` : (optionnel, valeur par défaut : 50) indique la fréquence de filtre à utiliser pour éliminer le bruit ambiant. La valeur est un nombre entier positif, dont les seules valeurs possibles sont 50 et 60. L'objectif est d'éliminer le bruit produit par les alimentations secteur. Nous ne devrions pas être confrontés à ce problème dans nos applications. Mais si l'équipement est installé dans un pays non européen, il est alors tout de même préférable d'adapter la valeur à la fréquence secteur du pays. Utilisez par exemple la valeur 60 si le capteur est aux États-Unis d'Amérique.

Ce capteur est en mesure de gérer deux alarmes ; l'une pour le dépassement d'un seuil haut de température et l'autre pour un seuil bas. Il est possible d'utiliser l'un ou l'autre ou les deux simultanément. Les deux seuils utilisent un système d'hystérésis, pour éviter

des basculements trop fréquents et injustifiés entre les états d'alarme et de non-alarme. La configuration se fait au moyen de la section optionnelle `alarm`, comme dans le cas de l'exemple de configuration suivant :

```
13=  "sensors": [{
14      "name": "PT100-1",
15      "type": "MAX31865",
16      "periodHr": 1,
17      "RTDType": "PT100",
18      "computeDeltaT": true,
19      "nbWires": 4,
20      "Rref0mhs": 412.0,
21      "filterFreqHz": 50,
22      "use5VWhenActive": true,
23=   "alarm": {
24       "temperatureHighSetDegC": 300.0,
25       "temperatureHighClearDegC": 250.0,
26       "temperatureLowClearDegC" : -10.0,
27       "temperatureLowSetDegC"  : -50.0
28     }
29   },
```

La configuration du seuil haut se fait au moyen des paramètres `temperatureHighSetDegC` et `temperatureHighClearDegC`, pour le seuil de déclenchement et le seuil de relâche respectivement. Les valeurs sont des nombres réels. Le seuil de déclenchement doit être strictement supérieur au seuil de relâche.

La configuration du seuil bas se fait au moyen des paramètres `temperatureLowSetDegC` et `temperatureLowClearDegC`, pour le seuil de déclenchement et le seuil de relâche respectivement. Les valeurs sont des nombres réels. Le seuil de déclenchement doit être strictement inférieur au seuil de relâche.

### 3.4 Configuration de l'heure

Elle se fait dans la dernière section du fichier d'exemple donné. Le cas typique est d'utiliser le GPS pour la mise à l'heure du nœud, aussi la section `manualUTC` n'est en fait pas utile. Elle est présente dans l'exemple de configuration au cas où il serait nécessaire de faire une mise à l'heure manuelle.

Le paramètre `syncMethod` indique la méthode utilisée par le nœud pour se mettre à l'heure. Écrivez la valeur `"GPS"` pour activer l'utilisation du GPS.

Dans la section `GPS` le seul paramètre vraiment obligatoire est `periodDay`. Il fixe la période de vérification et de remise à l'heure du nœud en nombre de jours. Il est conseillé d'utiliser une valeur comprise entre 1 et 7 jours. Plus la période est courte, moins il y a de risques de voir l'heure du

nœud dériver, mais plus l'autonomie énergétique du nœud est réduite. A titre indicatif, la dérive de l'heure du nœud est de moins de 5 secondes par semaine.

Le paramètre `timeoutSec` est optionnel et fixe le temps d'attente maximal, en secondes, pour obtenir l'heure et la position du GPS. Sa valeur par défaut est de 120 secondes, soit 2 minutes. Il est conseillé d'augmenter cette valeur si les conditions de réception GPS sont difficiles. Il est en revanche déconseillé de dépasser un temps d'attente de 5 minutes, faute de réduire l'autonomie énergétique du nœud.

### 3.5 Prise en compte de la nouvelle configuration

Sauvegardez votre fichier de configuration. Fermez votre éditeur de texte. Éjectez le disque correspondant à votre nœud dans votre système d'exploitation. Débranchez le câble USB qui relie le nœud à l'ordinateur. Le nœud redémarre alors (pour peu que vous ayez pensé à brancher sa batterie interne).

En cas de problème de syntaxe ou de problème majeur avec le fichier de configuration, les deux LEDs vertes en façade clignotent simultanément et rapidement. Dans ce cas, reconnectez le nœud en USB à votre ordinateur, rouvrez le fichier de configuration et cherchez l'erreur.

## 4 Assemblage final

La dernière étape, au cas où vous ne l'auriez pas encore faite, est de connecter le ou les capteurs au nœud. Vissez les connecteurs M12 femelles sur les capteurs M12 mâles câblés pour recevoir les capteurs.

En attente d'aller sur le terrain, débranchez le connecteur blanc de la batterie interne du nœud. Vous évitez ainsi qu'il la vide inutilement et de produire des données sans valeur.

Il est par ailleurs possible de recharger la batterie du nœud au moyen d'un câble USB connecté à un ordinateur ou à un chargeur USB.

## 5 Historique des révisions

Rév.	Date	Modifications
1	01/09/21	Révision initiale.