

### Production attendue :

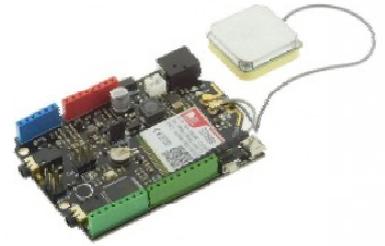
Un lien web permettant (au clic) d'identifier sur une carte géographique le lieu ayant pour coordonnées celles produites par la trame NMEA du module GPS SIM808.

Exemple de lien :

<https://www.google.fr/maps/place/47.85754,3.97350>

Latitude au  
format DD

longitude au  
format DD



SIM808 DFR3555

### Comment faire ?

- 1- Il faut convertir au format DD les coordonnées (latitude et longitude) issues du capteur GPS qui sont au format DMM.

On rappelle que les coordonnées au format DMM contiennent un nombre à virgule dont la partie entière correspond au nombre de degrés et dont la partie décimale correspond au nombre de minutes divisée par 100.

Exemple : pour la coordonnée DMM = 47.5145354 cela veut dire 47° 51,45354min. Converti en DD cela donne 47,857559°. On rappelle que 60min correspond à 1°.

- **Proposer** une solution mathématique pour récupérer les degrés et transformer les minutes en décimal de degré.
  - **Compléter** dans le programme [prog\\_02\\_creation\\_lien\\_Googlemaps\\_eleve.ino](#) la fonction **DMM\_vers\_DD** qui retournera une coordonnée au format DD si on lui passe une coordonnée au format DMM (nota : pour commencer simplement, les coordonnées GPS seront issues de variable globales et non pas du module GPS).
  - Tester votre fonction.
- 2- Il faut assembler des chaînes de caractères pour construire le lien Googlemaps. On distingue plusieurs chaînes de caractères à assembler :
    - ➔ La chaîne du lien vers GoogleMaps : "https://www.google.fr/maps/place/"
    - ➔ avec la chaîne de caractères issue de la latitude : "47.85754"
    - ➔ avec le séparateur entre la latitude et la longitude : ","
    - ➔ avec la chaîne de caractères issue de la longitude : "3.97350"
  - **Ouvrir et tester** le programme [prog\\_03\\_Conversion\\_float\\_string\\_Concat\\_eleve.ino](#) pour comprendre comment on assemble des chaînes de caractères et comment on convertit un float en string.
  - S'en inspirer pour **compléter** la fonction **transforme\_coordonnees\_en\_lien** du programme [prog\\_02\\_creation\\_lien\\_Googlemaps\\_eleve.ino](#) de façon à retourner le lien au format string (nota, pensez à faire des versions n+1 de votre programme).
  - **Tester** votre fonction.

- 3- Il faut maintenant transformer cette chaîne de caractères en un tableau de caractères car l'envoi d'un SMS se fait avec un tableau de caractères et non pas une chaîne de caractères. Ce travail est déjà fait en fin de fonction `setup()`. Vous n'avez rien à faire qu'à regarder les lignes de code et à **tester** le fonctionnement global du programme.
- 4- Pour terminer, il faut **remplacer** les coordonnées inscrites en dur dans le programme [prog\\_02\\_creation\\_lien\\_Googlemaps\\_eleve.ino](#) par le morceau de programme qui crée les coordonnées à partir des données du capteur GPS (vu dans l'étape précédente). Puis tester le bon fonctionnement.

L'idée c'est d'obtenir quelque chose comme cela :



```
COM10
23:50:28.513 -> latitude DMM : 47.514533
23:50:28.559 -> latitude DD : 47.857555
23:50:28.559 -> longitude DMM : 3.584243
23:50:28.604 -> longitude DD : 3.973738
23:50:28.604 -> lien Googlemaps au format string : https://www.google.fr/maps/place/47.857555,3.973738
23:50:28.698 -> lien Googlemaps adapté aux SMS : https://www.google.fr/maps/place/47.857555,3.973738
```

Options:  Défilement automatique  Afficher l'horodatage | Nouvelle ligne | 9600 baud | Effacer la sortie

**Attention 1** : il faut attendre une poignée de secondes, pour que des coordonnées soient affichées dans le moniteur série.

**Attention 2** : il faudra de préférence que le module GPS soit à l'extérieur pour recevoir un meilleur signal GPS.

**Comparer** la position pointée avec le lien sur une carte et la position réelle du module GPS (<https://www.google.fr/maps/place/>)

Notes personnelles :

---

---

---

---

---

---