

### 1. Décodage des informations issues de la carte GPS en vue de leur exploitation

*Objectif de cette partie :* **proposer** une solution au décodage de la trame au format texte du GPS pour en extraire la position géographique de la prise de vue.

En montagne, il est plus difficile d'identifier précisément les lieux de prises de vues que dans une ville, dans laquelle chaque rue porte un nom et chaque bâtiment un numéro. Un point GPS est donc nécessairement associé à chaque prise de vue.

Cela peut être réalisé par un GPS de poche. Cependant, dans un souci d'ergonomie, il est souhaitable d'afficher directement la position en incrustation vidéo sur l'écran de contrôle.

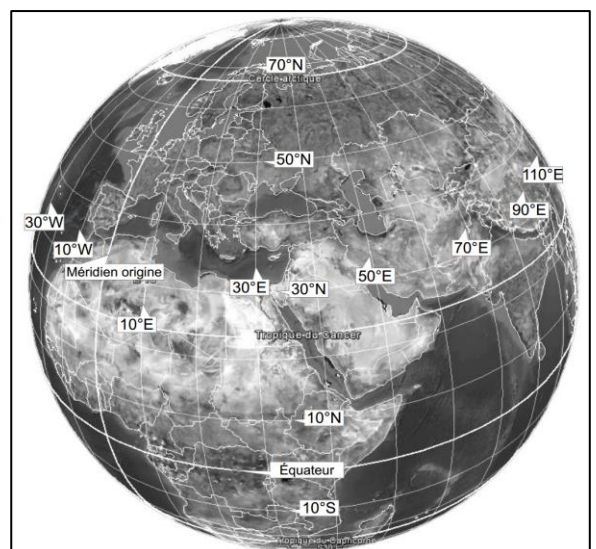
Un module GPS est donc interfacé à la carte électronique. Les informations de sortie du GPS sont traitées par un microcontrôleur et transmises à l'écran de contrôle.

Ce module doit permettre de déterminer la longitude et la latitude du lieu.

La **latitude** est une coordonnée géographique représentée par une valeur angulaire. Elle caractérise la position d'un point sur Terre par rapport à l'équateur qui est le plan de référence. Elle est exprimée en degré vers le nord ou vers le sud.

La **longitude** est également une coordonnée géographique représentée par une valeur angulaire. Elle caractérise la position d'un point sur Terre par rapport au méridien de Greenwich. Elle est exprimée en degré vers l'est ou l'ouest.

Par exemple, la tour Eiffel se situe  $48^{\circ} 51,497' N$  et  $2^{\circ} 17,655' E$  (en degré et minute décimale d'angle).

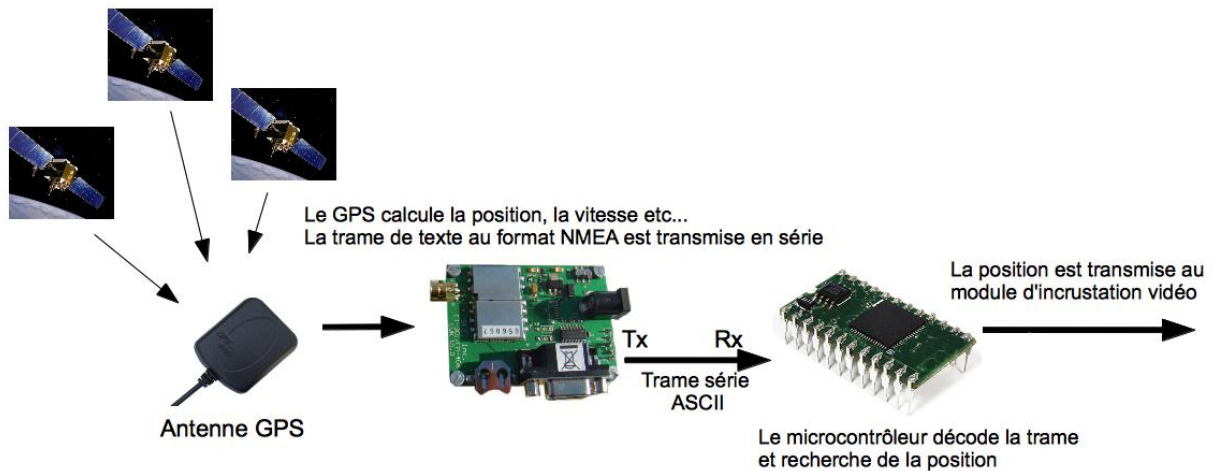


**Figure 12 : latitudes et longitudes sur le globe terrestre**

Le GPS, pour Global Positioning System, que l'on peut traduire en français par « système de localisation mondial » est un système de géolocalisation.

Un récepteur GPS capte les signaux d'au moins quatre satellites équipés d'horloges atomiques. En calculant les temps de propagation de ces signaux depuis les satellites, il peut déterminer sa distance par rapport à ceux-ci. Il peut ainsi situer précisément en trois dimensions n'importe quel point sur Terre.

Les signaux émis par les satellites sont captés par l'antenne et traités par le module GPS.



**Figure 13 : chaîne d'acquisition du GPS**

Pour recevoir la trame série codée en ASCII avec le microcontrôleur, il faut, dans un premier temps, configurer cette liaison. L'instruction *OPENCOM* permet d'ouvrir et de configurer le port série.

OPENCOM canal, debit, protocole, recvsiz, sendsizcanal :

- canal RS232, de 0 à 3 suivant type de circuit ;
- debit, en bits/s (ou bauds dans le cas d'une liaison série RS232) ;
- protocole, détaillé dans le tableau ci-dessous ;
- recvsiz, taille du buffer de réception ;
- sendsizcanal, taille du buffer d'émission.

Voici un exemple de configuration pour le canal 1 avec un débit de 9600 bits/s, des mots de 8 bits, une parité paire, 2 bits de stop et des buffers de 20 et 50 octets.

**OPENCOM 1,9600,23,20,50**

Chacun des bits de l'octet protocole a la signification suivante :

- le bit de poids faible bit0 et le bit1 déterminent la taille des mots ;
- le bit2 détermine le nombre de bits de stop ;
- le bit3 et le bit4 déterminent la parité.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
			Parité		Stop Bit	Bit	# of Bits
			0	0 = Aucune	0=1 bit de Stop	0	0 = 5 bits
			0	1 = Reservé	1=2 bit de Stop	0	1 = 6 bits
			1	0 = Pair		1	0 = 7 bits
			1	1 = Impair		1	1 = 8 bits

**Tableau 7 : définition du protocole du port série**

Les données présentes en sortie du module GPS sont transmises sur le canal 1 du microcontrôleur au format de 8 bits, avec un débit de 4 800 bits/s, un bit de stop et sans contrôle de parité. Le buffer de réception (mémoire tampon) est fixé à 50 octets et celui d'émission à 0.

**Q1. Déterminer** l'ensemble des paramètres à passer à l'instruction OPENCOM.

Pour tester le dispositif, la trame issue du module GPS a été transmise vers le port série d'un ordinateur et affichée au format texte dans un logiciel de type terminal.

La trame est transmise en boucle par le GPS. Cette trame est au standard NMEA, elle comporte de nombreuses informations telles que l'heure, la vitesse, la position, etc.

La virgule est utilisée pour séparer les différents champs.

```
$GPGGA,122755.095,4849.7970,N,00134.7688,W,1,03,2.3,-48.0,M,48.0,M,,0000*54  
$GPGSA,A,2,27,26,05,,,,,,,,,2.5,2.3,10*30
```

**Figure 14 : trame NMEA transmise par le GPS**

La trame ci-dessus a été reçue à la position **48° 49,7970' N** (Nord) et **1° 34,7688' W** (Ouest).

Le traitement des chaînes de caractères nécessite de bien les identifier et d'en connaître précisément le format.

Le format numérique de la longitude est différent de celui de la latitude avec un chiffre significatif supplémentaire. Ceci s'explique par la valeur de l'angle qui ne dépasse pas 90° pour une latitude et qui peut atteindre 180° pour une longitude.

L'objectif est d'extraire les suites de caractères définissant la latitude et la longitude, afin d'afficher la position géographique de la prise de vue sur l'écran de contrôle.

Chaque ligne débute par un caractère \$. Il faut ensuite repérer la suite de caractères GPGGA, puis passer deux virgules pour commencer à lire les informations relatives à la latitude.

L'algorithme permettant d'extraire la latitude de la trame NMEA est fourni ci-dessous. Les phrases notées en italique et précédées par // sont des commentaires.



# Ballon de therographie

## Analyse d'une trame GPS

Spé SI  
2021

TD

// La fonction **serie(x,y)**, appelée plusieurs fois, permet de lire le nombre **y** d'octets reçus successivement sur le port **x**. Pour lire un seul octet, il suffit de positionner **y** à 1.

// Déclaration des variables

**reception**, **latitude**, **longitude** de type chaîne de caractères

**compteur** de type entier

// Recherche de l'entête de la trame NMEA

Début

**reception** ← "" // Initialisation de la variable **reception** avec une chaîne vide

Répéter // Recherche de la suite de caractères \$GPGGA

Répéter

**reception** ← serie(1,1)

Jusqu'à **reception** = "\$"

**reception** ← "" // Réinitialisation de la variable **reception** avec une chaîne vide

**reception** ← serie(1,5)

Jusqu'à **reception** = "GPGGA"

// La variable **reception** ayant maintenant pour contenu la suite de caractères **GPGGA**, on recherche le début de la chaîne de caractères relative à la latitude

**reception** ← "" // Réinitialisation de la variable **reception** avec une chaîne vide

**compteur** ← 0 // Initialisation de la variable **compteur**

Pour **compteur** allant de 1 à 2

Répéter

**reception** ← serie(1,1)

Jusqu'à **reception** = ","

**reception** ← "" // Initialisation de la variable **reception** avec une chaîne vide

Fin pour

// Lecture de la suite de caractères relative à la latitude

**latitude** ← "" // Initialisation de la variable **latitude** avec une chaîne vide

**latitude** ← serie(1,11) // La variable **latitude** contient maintenant la bonne valeur

// Appel de la fonction **afficher (variable,1)** qui permet de convertir les derniers caractères stockés dans la variable **reception** en une donnée lisible par l'opérateur et d'afficher ces informations sur la première ligne de l'écran de contrôle.

afficher (**latitude**,1)

**Q2. Compléter** l'écriture de cet algorithme afin d'extraire et d'afficher la longitude.