**TimeNtp**

**TimeNtp()**

Constructeur vide.

**TimeNtp(String server, int zone)**

Constructeur initialisant les attributs NtpServerName et timeZone avec ses paramètres.

**void init(String server, int zone)**

Initialise l’objet pour le rendre fonctionnel. (Initialisation du serveur, de la timeZone et du tempo auquel récupérer l’heure).

**time\_t getTime(void)**

Retourne le temps en secondes depuis le 1er Janvier 1970.

**time\_t globalGetNTPTime()**

Retourne le résultat de getNtpTime().

**time\_t globalGetNTPTime();**

Récupère l’heure sous forme d’un paquet depuis un serveur ntp (avec un timeout de 7sec), puis l’ajuste avant de la retourner. L’ajustement est fait en plusieurs étapes. Le temps NTP est le nombre de secondes ecoulée depuis 1900. Or le temps UNIX commence le 1er Janvier 1970, on enleve donc 70ans en seconde a la date NTP (2208988800 secondes). La timeZone est ensuite appliquée, suivie de l’heure d’été si présente.

Retourne 0 si echec.

**void sendNTPpacket(IPAddress &address)**

Efface le buffer du paquet, puis le rempli selon le [contenu d’un paquet NTP](https://www.cisco.com/c/en/us/about/press/internet-protocol-journal/back-issues/table-contents-58/154-ntp.html). Ce paquet est ensuite envoyé à l’adresse précisée en paramètre.

**bool summertime(int year, byte month, byte day, byte hour, byte tzHours)**

Retourne true si l’heure actuelle est en heure d’été et false sinon.

|  |
| --- |
| **TimeNtp** |
| * getTimeObject : TimeNtp \* * Udp : WifiUdp * ntpServerName : String * timeZone : uint8\_t * localPort : unsigned int * NTP\_PACKET\_SIZE : int * packetBuffer[] : byte * tempo : unsigned int * gotTime : bool |
| * getNtpTime() : time\_t * sendNtpPacket(address : IPAddress&) * globalGetNtpTime() : time\_t * summertime(year : int, month : byte, day : byte, hour : byte, tzHours : byte) : bool * TimeNtp() : TimeNtp * TimeNtp(server : String, zone : int) : TimeNtp * getTime() : time\_t * init(server : String, zone : int) : void |