

# Le drone et la tablette

PHILIPPE LE GAL, DENIS LE ROUX, PATRICK LE YONDRE, GUY RIPOCHE <sup>[1]</sup>

*Les produits mécatroniques intègrent par définition de multiples technologies et permettent ainsi d'aborder de nombreux champs. Le drone en est une parfaite illustration : il offre des possibilités de séquences pédagogiques aussi conséquentes que variées, autour des matériaux, du comportement mécanique des structures ou, comme celle proposée ici, de la communication de l'information.*

Conçus pendant longtemps au domaine militaire, les drones se sont vu confier de nombreuses missions civiles : surveillance d'ouvrages d'art difficilement accessibles (ponts, tunnels, pylônes, etc.), surveillance de l'environnement (pollutions, incendies, etc.), prises de vues aériennes (reportages, cinéma, etc.).

L'AR.Drone <sup>1</sup> de la société Parrot (voir « En ligne » en encadré), conçu pour une utilisation en extérieur comme en intérieur grâce à une carène prévue pour le protéger des chocs et pour éviter le contact avec les hélices en rotation, est le premier quadricoptère piloté par *smartphone* ou tablette numérique <sup>2</sup>. À une distance de quelques dizaines de mètres, en plus du pilotage intuitif qu'il permet, le mobile (*smartphone* ou tablette) reçoit par Wi-Fi et affiche en direct sur son écran les images de l'une ou l'autre des deux caméras embarquées. Plus qu'un engin télécommandé, il s'agit d'une véritable plate-forme de jeu multijoueur.

L'utilisation d'un *smartphone* ou d'une tablette numérique pour piloter un modèle réduit par de simples mouvements est une évolution naturelle liée aux nouvelles possibilités qu'offrent ces concentrés de technologies. De nouvelles applications mobiles et de nouveaux usages apparaissent au fur et à mesure que ces équipements gagnent en puissance de calcul et en autonomie. Associés à l'étude de systèmes techniques en STI2D, ces nouveaux outils vont permettre d'élargir la palette des moyens d'apprentissage mis à la disposition des équipes enseignantes pour les activités d'expérimentation, d'observation et d'analyse.

## Le « système à enseigner » pour les STI2D

Les activités pédagogiques sont organisées autour de plusieurs équipements et logiciels complémentaires à l'AR.Drone lui-même. Le tout constitue le « système à enseigner » AR.Drone conçu, réalisé et commercialisé par la société DMS (voir « En ligne » en encadré). Il est composé des éléments suivants <sup>3</sup> :

### mots-clés

communication, information, informatique, mécatronique, programmation, réseau

● Un drone réel et un drone didactisé monté sur un banc d'essai pour réaliser les activités pratiques en toute sécurité

● Une tablette tactile Apple (iPad 2), appelée ici « station-sol »

● Sept applications développées spécialement pour les activités d'enseignement par Ripple Motion (voir « En ligne » en encadré), une société nantaise spécialisée dans le développement d'applications mobiles, dans le cadre d'un partenariat avec DMS <sup>4</sup>

● Un environnement multimédia d'apprentissage (modèle 3D, analyse SysML, vidéos, etc.)

L'AR.Drone est un système à la croisée de plusieurs domaines des STI2D <sup>5</sup>. Tout d'abord, l'information et la communication : c'est un système informatique embarqué communicant, riche en capteurs et actionneurs (moteurs *brushless*, altimètre à ultrasons, centrale inertielle, caméras, Wi-Fi, etc.). Puis les matériaux : les contraintes de poids imposées pour ce type d'aéronef sont importantes et nécessitent des matériaux adaptés. Enfin, le comportement mécanique des structures :



1 L'AR.Drone de Parrot



2 L'AR.Drone piloté par *smartphone*



3 Le « système à enseigner » de DMS

### En ligne

Parrot, fabricant de l'AR.Drone :

<http://ardrone.parrot.com/>

DMS, qui conçoit, réalise et commercialise le « système à enseigner » :

[www.dmseducation.net](http://www.dmseducation.net)

Ripple Motion, qui développe les applications spécifiques :

[www.ripplemotion.fr/](http://www.ripplemotion.fr/)

[1] Enseignants au lycée Nicolas-Appert d'Orvault (44).

les principes de sustentation et de déplacement mis en œuvre dans l'AR.Drone répondent à la fonction première du système.

La part de l'information est de plus en plus importante dans tous les systèmes pluritechniques actuels étudiés en STI2D. Le programme en aborde les différents aspects, de sa conversion sous une forme numérique pour faciliter les traitements logiques à son traitement informatique en passant par son transport via les réseaux et leurs protocoles. Ici, nous étudierons la chaîne d'information depuis le gyroscope chargé de mesurer les variations d'angles de roulis et de tangage de l'AR.Drone jusqu'à la station-sol via le réseau Wi-Fi.

Nous aborderons dans un premier temps les mécanismes de connexion au réseau Wi-Fi, puis nous étudierons les principales structures matérielles et logicielles du drone en détaillant un capteur particulier. Enfin, nous donnerons un exemple d'activité pratique dont le thème est l'analyse des communications Wi-Fi entre l'AR.Drone et la station-sol.

Au cours de cette étude, différents diagrammes SysML nous serviront de fil conducteur. Rappelons que l'analyse SysML permet :

- d'établir le contexte du système avec ses frontières ;
- de décoder l'organisation fonctionnelle d'un système à travers la découverte des cas d'utilisation et des relations entre les cas d'utilisation ;
- d'étudier les structures d'un système grâce aux diagrammes de définition de blocs et aux diagrammes de blocs internes ;
- d'appréhender l'organisation logicielle d'un système en étudiant son comportement avec les diagrammes de séquence et les diagrammes d'état.

### La connexion au réseau Wi-Fi

Première étape pour pouvoir piloter l'AR.Drone, se connecter au réseau Wi-Fi qu'il met à disposition de la station-sol dès qu'il est sous tension : c'est le cas d'utilisation « configurer le réseau Wi-Fi » du diagramme 6.

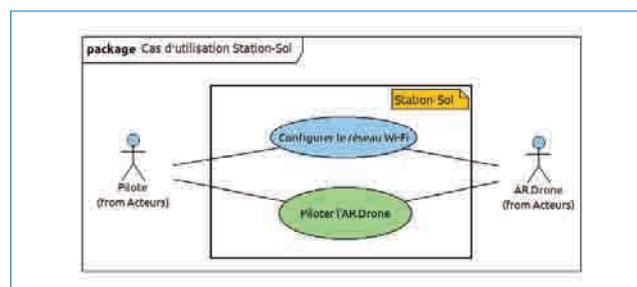
Dès que l'AR.Drone est sous tension et prêt à décoller, il émet dans son environnement radio des balises (*beacons*) permettant à la station-sol de repérer le point d'accès Wi-Fi et de débiter le processus de connexion. Les balises sont des trames Ethernet définies par le protocole IEEE 802.11(a, b, g ou n) diffusées habituellement 10 fois par seconde par les points d'accès Wi-Fi. Pour le pilote, cela se traduit simplement par l'apparition du réseau de l'AR.Drone dans l'écran de configuration des réseaux Wi-Fi de la station-sol 7.



4 Un exemple d'application développée par Ripple Motion



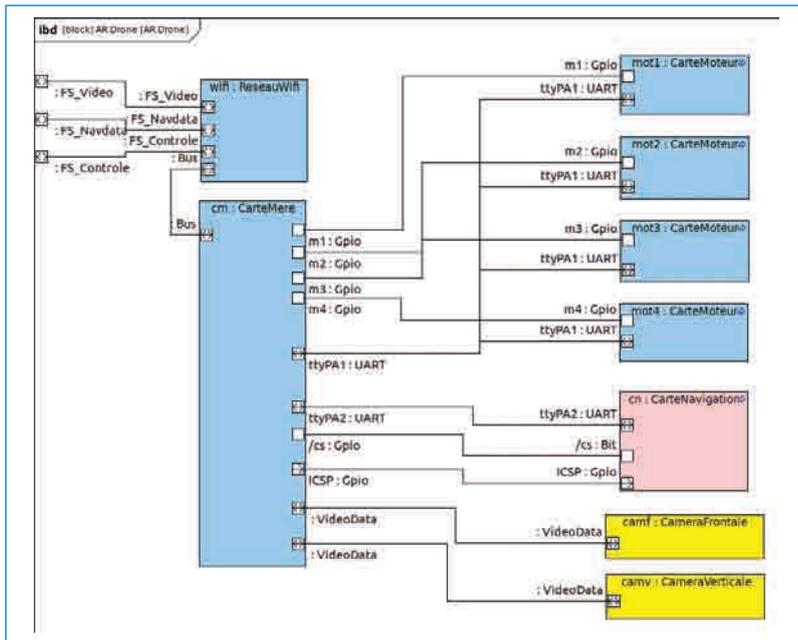
5 Les caractéristiques de l'AR.Drone



6 Le diagramme des cas d'utilisation de la station-sol



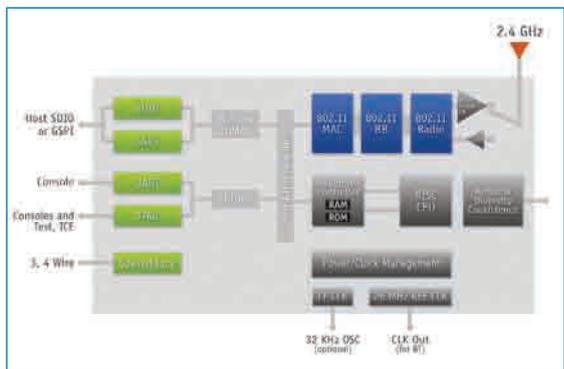
**7** Le réseau Wi-Fi de l'AR.Drone



**8** Le diagramme de blocs internes de l'AR.Drone



**9** Le chipset Atheros AR6102G et l'antenne Wi-Fi de l'AR.Drone



**10** Le chipset Atheros AR6102G

- 7** Application
- 4** Transport
- 3** Réseau
- 2** Liaison
- 1** Physique

**11** Le modèle OSI de l'AR.Drone

```
$ iwlist wlan0 scan
Cell 01 - Address: 00:26:7E:4F:CB:57
Channel:6
Frequency:2.437 GHz (Channel 6)
Quality=70/70 Signal level=-35
dBm
Encryption key:off
ESSID:"ardrone_046565"
Bit Rates:1 Mb/s; 2 Mb/s; 5.5
Mb/s; 11 Mb/s; 24 Mb/s 54 Mb/s
Mode:Master
...
```

**12** Le détail des informations présentes dans les balises émises par l'AR.Drone

Ce processus de connexion apparemment simple cache des mécanismes assez complexes que nous allons à présent détailler.

Nous pouvons dans un premier temps examiner le diagramme de blocs internes de l'AR.Drone afin d'y situer la fonction de communication Wi-Fi **8**. Elle est assurée par le composant Atheros AR6102G, comme on peut le voir sur la carte mère de l'AR.Drone **9**. Ce composant sera chargé de l'émission vers la station-sol des données de navigation et des vidéos prises par les caméras embarquées, et de recevoir les commandes de pilotage. La documentation technique nous donne des indications sur ses caractéristiques :

**« Wi-Fi Modules AR6102G**

*The Atheros AR6102 is a member of the WLAN ROCm (Radio On Chip) family of chips. The compact size and low power consumption of this design make it an ideal vehicle for adding WLAN to hand-held and other battery-powered consumer electronic devices...*

**IEEE 802.11b/g compliant**

*Data rates of 1~54 Mbps for 802.11g »*

Dans le cas de l'AR.Drone, le composant assure donc les communications suivant la norme IEEE 802.11g (la norme IEEE 802.11b est désormais obsolète) pour un débit allant de 1 à 54 Mbps en fonction de l'environnement radio et de la distance entre la station-sol et l'AR.Drone.

Les détails internes du composant AR6102G **10** nous indiquent qu'il gère uniquement les deux premières couches du modèle OSI **11** : la couche « physique » (802.11 radio : modulation BPSK à 64-QAM et codage DSSS à OFDM) et la couche « liaison de données » (802.11 BB et MAC). Le reste de la pile de protocoles (couches de « réseau » à « application ») sera pris en charge par le noyau Linux qui équipe l'AR.Drone.

Examinons à présent les informations présentes dans les balises émises par l'AR.Drone. Pour cela, nous utiliserons un utilitaire en ligne de commande (iwlist) présent sur un PC équipé d'un système d'exploitation Linux à proximité de l'AR.Drone sous tension. Une fois lancée en mode « scanner » en précisant le nom de l'interface Wi-Fi du PC (ici wlan0), la commande nous retourne les détails de l'environnement radio du PC **12**. L'utilitaire va examiner cet environnement pendant quelques secondes, et, pour chaque canal scruté, va nous donner des informations sur les points d'accès Wi-Fi répertoriés (un seul dans notre cas).

La première indication donnée est l'adresse MAC du point d'accès Wi-Fi associé à la première cellule (Cell 01 - Address: 00:26:7E:4F:CB:57). Les trois premiers octets de l'adresse MAC nous donnent des indications sur le constructeur de l'interface Ethernet 802.11. En entrant ces octets dans l'interface de recherche de l'application iPad Se connecter fournie avec le système à enseigner, nous avons confirmation qu'il s'agit bien d'un équipement fabriqué par la société Parrot, en l'occurrence d'un AR.Drone **13**.

SOURCE: ATHEROS

Les trois lignes suivantes nous renseignent sur le canal de communication. Précisons que la norme IEEE 802.11b/g prévoit jusqu'à 13 canaux Wi-Fi en France **14** dans la bande ISM 2,4 GHz (*Industrial, Scientific and Medical*). Dans notre cas, l'AR.Drone a choisi de présenter son point d'accès sur le canal n° 6 (2,437 GHz) après avoir lui-même analysé l'environnement radio pour trouver le canal le moins encombré. Il est aussi intéressant de relever l'information concernant la puissance du signal reçue ( $\text{Signal level} = -35 \text{ dBm}$ ), indiquant une bonne présence du signal, comme nous le confirme l'indicateur de qualité de lien ( $\text{Quality} = 70/70$ ).

L'information suivante (**Encryption key:off**) nous indique que l'AR.Drone permet une connexion à son réseau sans clé d'authentification. Aucun mot de passe ne sera demandé lorsque la station-sol tentera une connexion au réseau Wi-Fi de l'AR.Drone.

Le nom du réseau Wi-Fi est donné par son SSID (*Service Set Identifier*) ou ESSID (*Extended Service Set Identifier*) : ESSID: "ardrone\_046565". C'est le nom qui apparaît dans l'interface de sélection du réseau Wi-Fi.

Les trames de balise contiennent aussi les informations sur les débits offerts par le point d'accès de l'AR.Drone: Bit Rates: 1 Mb/s; 2 Mb/s; 5.5 Mb/s; 11 Mb/s; 24 Mb/s 54 Mb/s. Le débit pourra en effet varier en fonction des perturbations de l'environnement radio et de la distance entre la station-sol et l'aéronef. C'est la modulation choisie par le point d'accès qui permet de faire varier le débit (de BPSK à 64-QAM, soit de 1 à 54 Mbps).

Enfin, la dernière indication (**Mode:Master**) nous donne le mode de fonctionnement du point d'accès de l'AR.Drone. Il s'agit ici du mode « infrastructure » que l'on rencontre le plus fréquemment dans un réseau Wi-Fi. C'est une architecture centralisée basée sur des points d'accès auxquels les clients Wi-Fi s'attachent pour accéder au réseau. Il est à noter que le mode choisi par les ingénieurs de Parrot dans les premières versions du micrologiciel de l'AR.Drone était le mode « *ad hoc* » où les clients Wi-Fi dialoguent directement entre eux.

#### ● Exploitations pédagogiques possibles

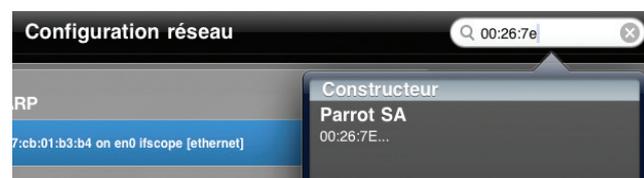
Transmission de l'information : modulations d'amplitude, modulations de fréquence, modulations de phase

Caractéristiques d'un canal de transmission : notion de fréquence de canal, introduction aux notions de puissance des signaux

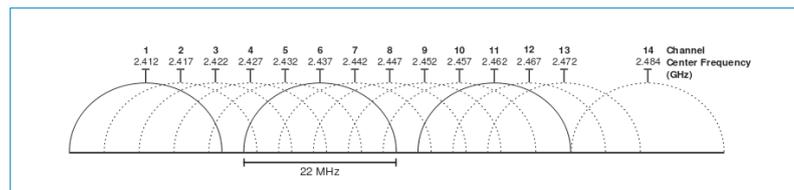
Adresse physique (MAC) du protocole Ethernet

#### L'obtention d'une adresse IP

Une fois que la station-sol est connectée au réseau Wi-Fi de l'AR.Drone, il faut qu'elle obtienne une adresse IP et un masque de sous-réseau. Pour cela, elle va diffuser une requête DHCP (*Dynamic Host Configuration Protocol*)



**13** La recherche du constructeur par adresse MAC avec l'application Se connecter



**14** Les canaux Wi-Fi en Europe

```
$ telnet 192.168.1.1
Trying 192.168.1.1...
Connected to 192.168.1.1.
Escape character is '^]'.
BusyBox v1.14.0 (2011-12-13 10:35:27 CET) built-in shell (ash)
Enter 'help' for a list of built-in commands.
#
```

**15** L'ouverture d'une session Telnet sur l'adresse IP de l'AR.Drone

```
# ps
      PID USER      VSZ   STAT COMMAND
      1  root        2732 S   init

...

      951  root        2736 S   telnetd -l /bin/sh
      953  root        2732 S   udhcpd /tmp/udhcpd.conf
      960  root        2732 S   inetd
      961  root        2736 S   /bin/sh /bin/check_update.sh
      962  root        12076 S  /bin/program.elf

...
```

**16** La liste des processus exécutés par l'AR.Drone

sur le réseau Wi-Fi, et l'AR.Drone, qui est serveur DHCP, va lui fournir ces éléments de configuration.

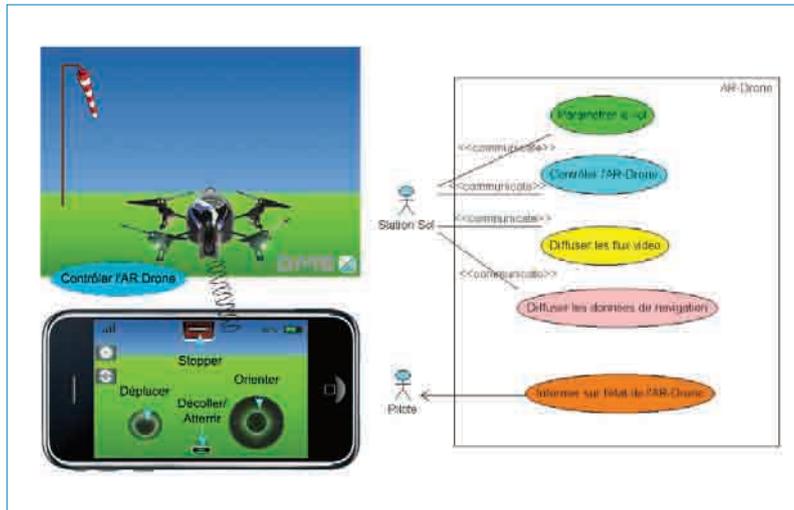
Examinons le fonctionnement interne de l'aéronef en demandant l'ouverture d'une session Telnet sur l'adresse IP de ce dernier (192.168.1.1) à partir d'un PC connecté sur son réseau Wi-Fi. Le résultat est présenté en **15**. Nous sommes en présence d'un terminal Linux classique, et nous allons entrer quelques commandes afin de comprendre l'architecture logicielle du drone. Tout d'abord, nous allons lister les processus exécutés (commande *ps*) **16**. Parmi eux, plusieurs attirent notre attention :

- *udhcpd* : c'est le serveur DHCP. Cela nous confirme que c'est bien l'AR.Drone qui fournira une adresse IP et un masque de sous-réseau à la station-sol qui se connectera à son réseau.

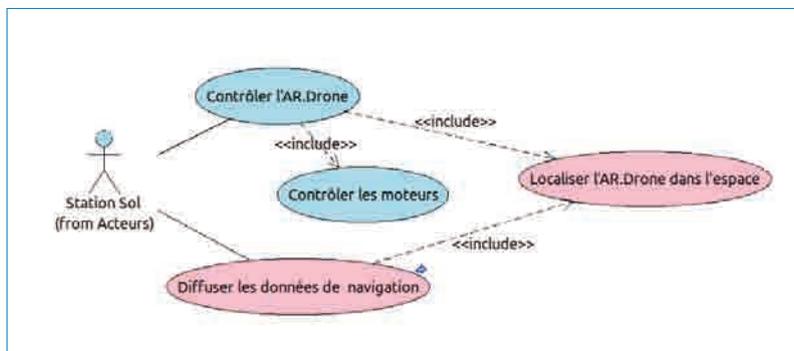
- *program.elf* : c'est dans ce processus que se trouve toute l'intelligence de l'AR.Drone. La partie logicielle de l'ensemble des cas d'utilisation qui seront évoqués plus loin est codée dans ce programme.

```
# cat /tmp/udhcpd.conf
start 192.168.1.2
end 192.168.1.5
...
opt subnet 255.255.255.0
opt lease 1200
```

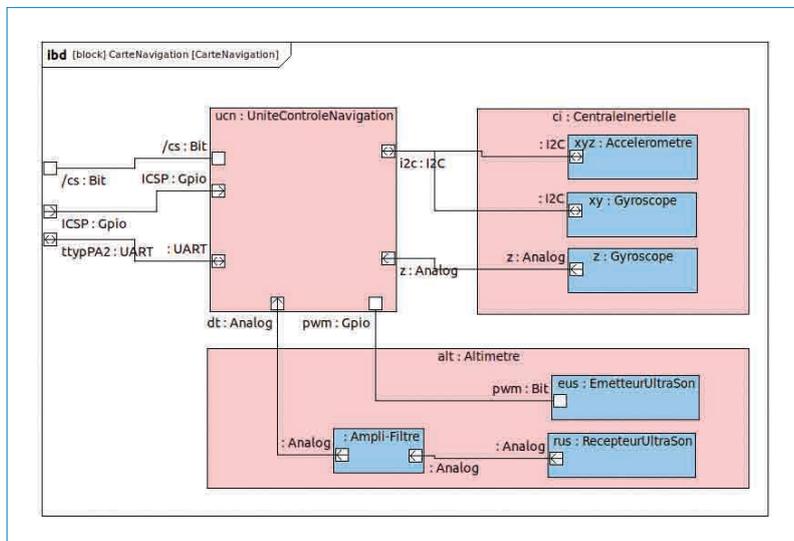
**17 La liste des processus exécutés par l'AR.Drone**



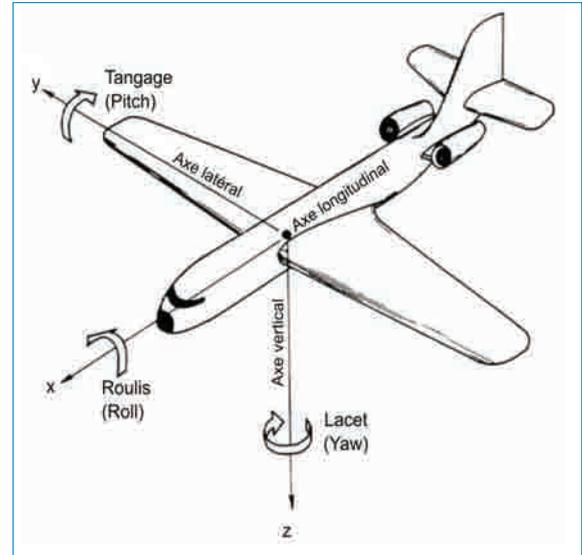
**18 Le diagramme des cas d'utilisation de l'AR.Drone**



**19 Un exemple de relations entre cas d'utilisation**



**20 Un exemple de diagramme de blocs internes de la carte de navigation**



**21 Le rôle stabilisateur de la centrale inertielle**

- `check_update.sh` : c'est ce processus qui est chargé de la mise à jour du micrologiciel de l'AR.Drone en cas de besoin.

Le fichier de configuration du serveur DHCP (`/tmp/udhcpd.conf`) **17** permet d'obtenir la liste des adresses IP qui seront distribuées ainsi que le masque de sous-réseau associé. La station-sol recevra la première adresse IP disponible, 192.168.1.2, associée au masque de sous-réseau 255.255.255.0.

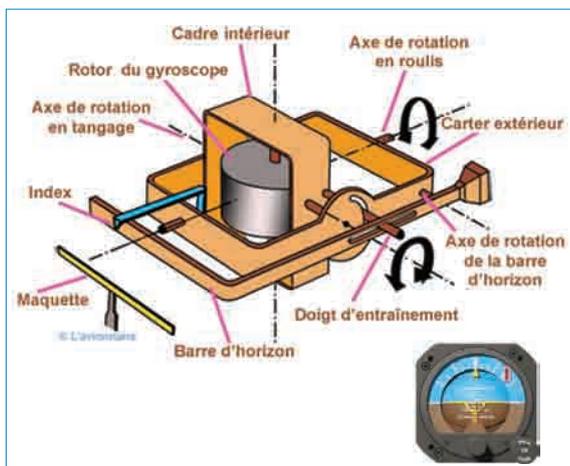
**• Exploitations pédagogiques possibles**

- Adresse physique (MAC) du protocole Ethernet et adresse logique (IP) du protocole IP
- Lien adresse MAC-IP : protocole ARP
- Test de connectivité entre la station-sol et l'AR.Drone (ping)
- En spécialité SIN, capture de trames et découverte du protocole DHCP

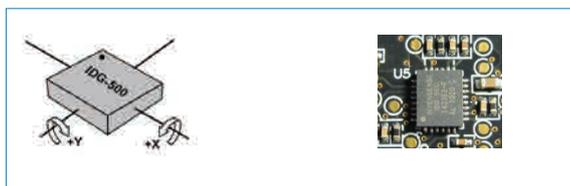
**Le fonctionnement interne de l'AR.Drone**

Nous allons à présent aborder le fonctionnement interne de l'AR.Drone en en découvrant les principaux cas d'utilisation. Ces cas d'utilisation peuvent se retrouver en pilotant l'aéronef et en étudiant ses interactions avec la station-sol (iPad) ; ils sont synthétisés dans le diagramme **18**. De même, il est assez simple de mettre en évidence les relations (inclusion et extension) entre les cas d'utilisation en analysant le comportement du drone en vol afin de compléter le diagramme **6** précédent **19**.

Il serait ensuite intéressant de relier les cas d'utilisation avec les structures matérielles et logicielles qui participent à leur déroulement. Par exemple, le cas d'utilisation « localiser l'AR.Drone dans l'espace » sera en grande partie réalisé par la carte de navigation, dont nous proposons un diagramme de blocs internes en **20**. Comme nous pouvons le voir sur le diagramme de blocs internes de l'AR.Drone **8**, la carte mère est



22 L'horizon artificiel d'un avion et son gyroscope



23 L'IDG-500 sur la carte de navigation, et les axes x et y de son gyroscope

reliée à la carte de navigation par une liaison série RS232 associée à quelques signaux de contrôle.

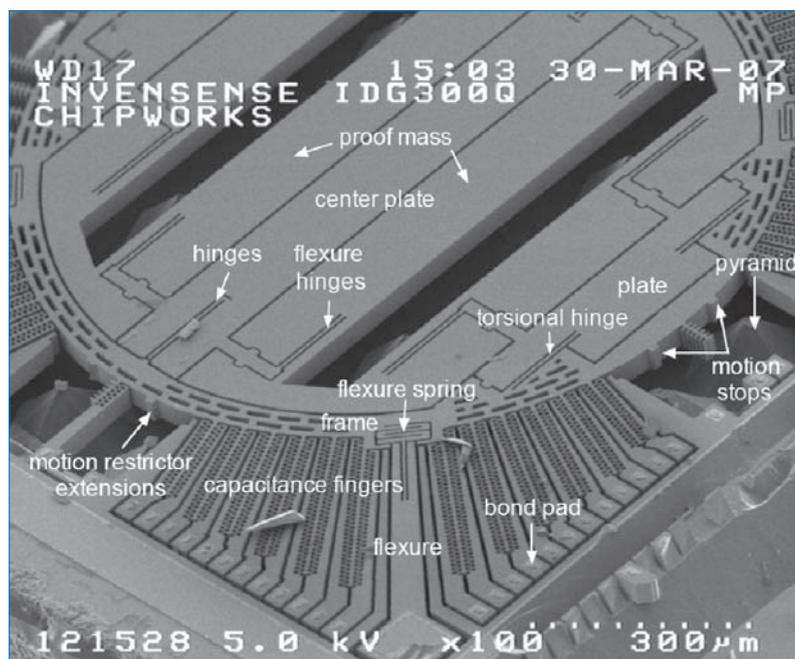
La carte de navigation est composée de trois blocs :

- **L'unité de contrôle de navigation**, un microcontrôleur chargé de prélever les mesures brutes sur les capteurs de la centrale inertielle et sur l'altimètre, et de les envoyer en permanence vers la carte mère
- **L'altimètre** chargé de mesurer l'altitude par rapport au sol
- **La centrale inertielle** chargée de mesurer les accélérations linéaires et angulaires de l'aéronef grâce aux accéléromètres et gyroscopes intégrés 21

Nous allons maintenant nous intéresser au gyroscope XY.

Dans le domaine de l'aviation, les gyroscopes se retrouvent dans plusieurs instruments de vol, par exemple l'horizon artificiel 22, le directionnel (indicateur de cap) et l'indicateur de virage. Semblable à celle d'une toupie qui se stabilise autour de son axe de rotation, c'est la propriété de l'immobilité de l'axe du rotor du gyroscope dans l'espace qui est utilisée ici : le gyroscope de l'horizon artificiel pivote dans un plan horizontal et préserve son orientation par rapport à l'horizon réel lorsque l'avion se cabre ou pique.

Le bloc xy : *Gyroscope* de la figure 20 est la structure qui nous intéresse dans cette étude. Cette instance de bloc est en partie réalisée grâce au composant Invensense IDG-500. Ce composant mesure les variations de vitesse angulaire autour des axes x pour le roulis et y pour le tangage 23. Ces mesures sont transmises à l'unité de traitement de la carte de navigation par bus i2c puis à la carte mère via la liaison série RS232.



24 Détails microscopiques d'un capteur gyroscopique MEMS de la famille IDG

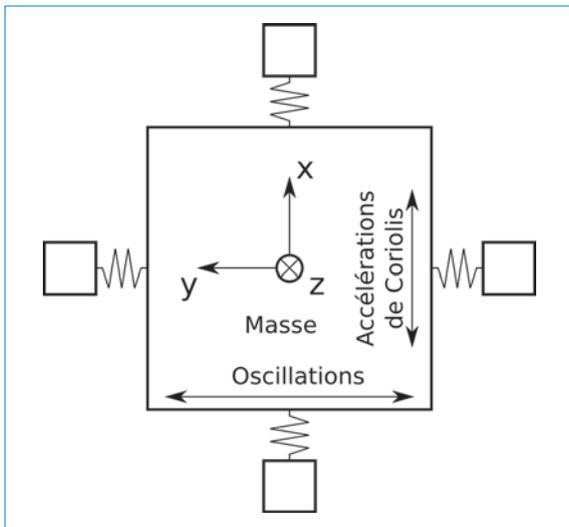
C'est la commande de ces angles qui permet à l'AR.Drone de se déplacer sur le plan xy. De même, sa stabilisation en cas de perturbation entraînant une variation de vitesse angulaire sur les axes x ou y (due au vent, par exemple) se fait par l'asservissement en vitesse des moteurs jusqu'à ce que le gyroscope ne détecte plus de variation angulaire.

Les dérives dans le plan xy sont aussi détectées grâce à la caméra verticale par l'analyse de l'image du sol. Le couplage du gyroscope et de la caméra permet d'avoir une bonne stabilisation, à condition d'avoir une image du sol présentant des contrastes.

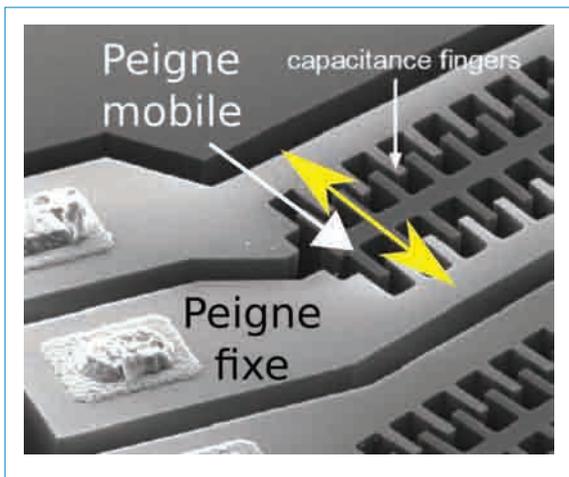
L'extrait de la documentation constructeur suivant nous indique que le gyroscope est intégré dans le silicium du composant IDG-500 : c'est un MEMS (*MicroElectroMechanical System*) 24 :

« The IDG-500 MEMS rate gyroscope is an integrated dual-axis angular rate sensor designed specifically for complex motion processing of A/V and PC remotes, as well as gaming controllers. Offering significant performance and packaging improvements over the IDG-300, the IDG-500 delivers 500°/sec full scale range and 2mV/°/sec sensitivity in a 44% smaller 4x5x1.2mm package size making it ideal for general motion processing applications. . . »

Un gyroscope MEMS utilise une structure vibrante plutôt que le traditionnel disque en rotation pour déterminer son orientation. Il mesure la vitesse angulaire grâce à l'accélération de Coriolis (l'accélération tangentielle que subit un objet placé sur un disque en rotation, proportionnelle à la distance de l'objet par rapport au centre du disque). En mesurant cette accélération, on peut mesurer le couple externe qui l'a engendrée. Les deux masses liées (*proof masses*) vibrent à leur fréquence de résonance le long d'un axe



25 La rotation autour de l'axe z



26 La variation de la capacitance par l'accélération de Coriolis

25. Si on applique une rotation à la structure, l'accélération de Coriolis qu'elle subit va déplacer le système de masses sur l'axe orthogonal à l'axe de vibration d'une valeur proportionnelle à l'accélération subie selon l'équation :

$$F_c = 2 m v \Omega$$

avec  $F_c$  la force produite par la masse  $m$  en vibration sur un axe, se déplaçant à la vitesse  $v$  dans un mouvement de rotation  $\Omega$ .

Le déplacement du système de masses est déduit de la variation de capacitance mesurée aux bornes des peignes interdigités (*capacitance fingers*) reliant le système de masses au support 26. Après filtrage et amplification, le composant fournit une tension proportionnelle à la vitesse angulaire de l'axe en rotation.

Le composant IDG-500 intègre deux structures similaires placées orthogonalement afin de mesurer les vitesses angulaires sur les deux axes x et y.

Les autres éléments de la carte de navigation, qui ne seront pas détaillés, sont les suivants :



27 L'écran de l'application Piloter

- **z : Gyroscope** : Le guidage de l'AR.Drone nécessite que la mesure du lacet se fasse avec une plus grande précision. Cette instance est réalisée grâce au composant Epson XV3500CB. C'est l'unité de traitement de la carte de navigation qui est chargée de la conversion analogique-numérique du signal issu du gyroscope.

- **xyz : Accelerometre** : Cette instance de bloc est réalisée grâce au composant Bosch BMA 150. Ce dernier mesure les variations de vitesse linéaire sur les axes x, y et z et les transmet à l'unité de traitement de la carte de navigation par un bus i2c, puis à la carte mère via la liaison série RS232. Ces informations sont importantes pour l'asservissement de l'AR.Drone : en l'absence d'ordre de la station-sol, pour stabiliser l'AR.Drone, le logiciel embarqué sur la carte mère va chercher à compenser les perturbations détectées par les accéléromètres (dues au vent, par exemple) en faisant varier la vitesse des moteurs.

- **EmetteurUltraSon, RecepteurUltraSon et Ampli-Filtre** : Ces trois instances de blocs constituent l'altimètre. Les trains d'impulsions (*pwm*) envoyés par l'émetteur à ultrasons sont réfléchis par le sol (jusqu'à une altitude de 6 m environ), et l'écho est capté par le récepteur à ultrasons. Après amplification et filtrage, le retard temporel de l'enveloppe du signal d'écho sur le train d'impulsions correspondant donne des indications sur la distance de l'AR.Drone par rapport au sol. Les données brutes issues de l'altimètre sont transférées vers la carte mère par l'unité de traitement de la carte de navigation. C'est le logiciel embarqué sur la carte mère qui est chargé du calcul de l'altitude. L'altimètre seconde l'accéléromètre de l'axe z pour la mesure des variations d'altitude et permet aussi la stabilisation de l'aéronef.

Donnée de navigation	Type (entier ou flottant / booléen)	Unité	Plage de variation	Rôle
Altitude	Flottant	Mètre	0-6 m	Donne l'altitude de l'AR.Drone
Alarme angle	Booléen			Indique si les angles de roulis et tangage sont dans les limites acceptables
...	...	...	...	...

28 Le tableau de synthèse des éléments observés durant le TP

### ● Exploitations pédagogiques possibles :

Acquisition et codage de l'information : capteurs (approche qualitative des capteurs, grandeur mesurée et grandeurs d'influence)

Conditionnement et adaptation du capteur à la chaîne d'information, échantillonnage, blocage

Une grande partie des données issues de la carte de navigation sont transmises après interprétation par le logiciel de la carte mère vers la station-sol via le réseau Wi-Fi. Nous allons analyser dans l'activité pratique qui suit une partie de ces communications. Nous détaillerons en particulier la transmission de l'« alarme angle » qui se déclenche si le logiciel de l'AR.Drone détecte une variation des angles de roulis ou de tangage trop importante à partir des données brutes reçues de la carte de navigation.

### Une activité : l'analyse des communications

Cette activité de 3 heures, découpée en deux parties, va permettre aux élèves de comprendre comment un modèle de communication en couches permet la transmission des informations de navigation de l'AR.Drone vers la station-sol (altitude, niveau batterie, alarmes, etc.).

L'objectif principal de formation est de décoder l'organisation fonctionnelle, structurelle et logicielle du système, de manière à acquérir les compétences d'identification et de caractérisation des fonctions, des constituants, des entrées/sorties et de l'agencement matériel et/ou logiciel d'un système.

Les connaissances visées portent sur le codage (binaire, hexadécimal, ASCII) et le transcodage de l'information, ainsi que sur la transmission de l'information (constituants et interfaçages, canal de transmission, multiplexage, modèles en couche des réseaux, protocoles et encapsulation des données).

Les équipements nécessaires à cette activité sont le drone relié à sa base et l'iPad équipé des applications Piloter et Analyser 3.

Dans la première partie, nous allons tout d'abord amener l'élève à identifier le ou les acteurs et le ou les cas d'utilisation en jeu dans la problématique sur le diagramme des cas d'utilisation fourni avec le dossier technique (acteur station-sol et cas d'utilisation « diffuser les données de navigation »). Ensuite, après avoir utilisé l'application de pilotage, l'élève repère sur l'interface de celle-ci 27 les éléments graphiques



29 L'écran de l'application Analyser

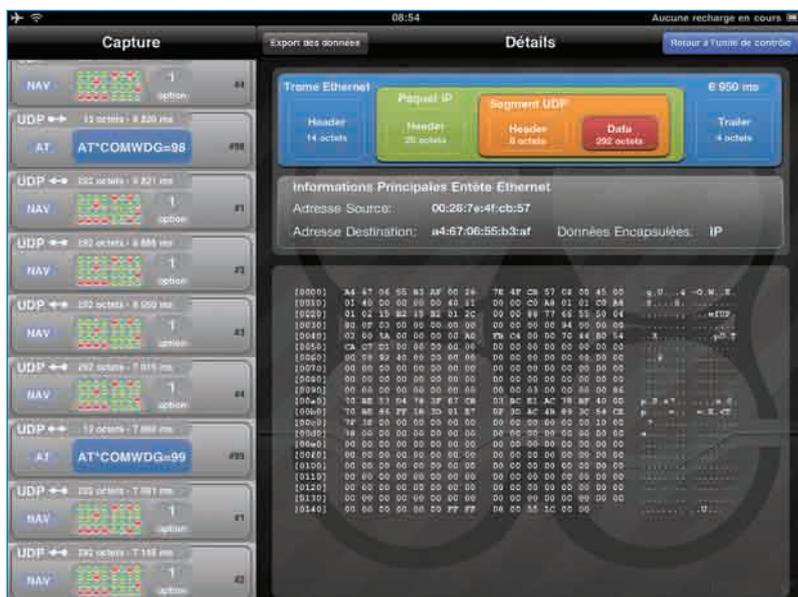
donnant l'état de l'AR.Drone en temps réel, c'est-à-dire les informations qu'il transmet en permanence à la station-sol.

L'élève remplit le tableau de synthèse 28 avec les éléments observés, puis est amené à étudier la documentation du kit de développement de l'AR.Drone (en anglais) afin de lister les principaux canaux de communication entre la station-sol et le drone situés au niveau de la couche « transport » du modèle OSI :

« Information about the drone (like its status, its position, speed, engine rotation speed, etc.), called NavData, are sent by the drone to its client on UDP port 5554. These NavData also include tags detection information that can be used to create augmented reality games. They are sent approximately 30 times per second. »

L'information importante relevée ici est le numéro de canal, UDP 5554, réservé aux données de navigation.

Nous souhaiterions ensuite avoir des informations sur la taille et la fréquence de rafraîchissement des données de navigation ainsi que sur le débit lié à cet échange sur le réseau, afin d'estimer le taux d'occupation des données de navigation sur la bande passante Wi-Fi disponible. L'élève utilise à cet effet l'application iPad Analyser 29 qui permet de capturer en temps réel les communications entre l'AR.Drone et la station-sol et de mesurer les débits associés aux différents canaux.



**30** L'utilitaire de décodage de l'application Analyser

Il constate que le débit est de quelques kilooctets par seconde et n'utilise qu'une partie réduite de la bande passante.

Intéressons-nous à présent au contenu des données de navigation afin de comprendre comment l'AR.Drone informe la station-sol sur son état. Pour cela, l'élève utilise l'utilitaire de décodage intégré à l'application Analyser **30**. Il y recherche l'information « alarme angle » qui se déclenche si l'aéronef présente un angle de roulis ou de tangage trop important.

L'utilitaire de décodage affiche les trames capturées sur la partie gauche de l'écran (avec leur horodatage, leur type et leur taille), et le détail d'une trame sélectionnée sur la partie droite. La zone tactile en haut de l'écran permet de choisir le niveau d'encapsulation

voulu. Il est ainsi très simple d'isoler les données significatives de chaque couche du modèle OSI. L'utilitaire permet aussi d'exporter la capture sous forme de fichier texte pour une exploitation sur le poste de travail de l'élève.

En sélectionnant une trame de données de navigation sur la partie gauche puis le niveau d'encapsulation *data* en haut de l'utilitaire, on peut voir les informations échangées au niveau de la couche « application » du modèle OSI. L'élève est ensuite amené à décoder progressivement une trame à l'aide de la documentation Parrot, qui indique l'organisation des données dans la trame **31**, et lui permet de décoder les 32 bits de l'état du drone **32**. Par exemple, le mot hexadécimal *Drone State* extrait d'une trame capturée 8F 8A 04 50 en hexadécimal donne une fois décodé en binaire le tableau **33**. Le bit b19 à 1 indique que l'AR.Drone est en « alarme angle ». En situation de vol, cette alarme provoque l'arrêt des moteurs et donc la chute de l'AR.Drone.

La seconde partie de l'activité, que nous n'aborderons pas ici, cherchera à mettre en évidence le phénomène d'encapsulation des données applicatives par les processus associés aux différentes couches réseau. Dans les activités associées, l'élève sera amené à :

- capturer et décoder les en-têtes de la couche « transport » ;
  - capturer et décoder les en-têtes de la couche « réseau » ;
  - capturer et décoder les en-têtes de la couche « liaison de données » ;
  - caractériser les communications en termes de bande passante ;
  - identifier les informations caractéristiques à chaque couche du modèle OSI à partir des captures réalisées.
- Au terme de l'activité, l'élève sera capable de retrouver et décoder une information applicative (couche

Header 0x55667788	Drone state	Sequence number	Vision flag	Option 1			...	Checksum block		
				id	size	data	...	cks id	size	cks data
32 bit int.	32 bit int.	32 bit int.	32 bit int.	16 bit int	16 bit int	...	...	16 bit int	16 bit int	32 bit int

**31** Un extrait de la documentation constructeur sur l'organisation des données dans la trame

N° Bit	Nom	Description
b0	FLY_MASK	0: ardrone is landed 1: ardrone is flying
...	...	...
b19	ANGLES_OUT_OF_RANGE	0: Ok 1: out of range
...	...	...

**32** Un extrait de la table des bits d'état du drone

b31	32 bits de Drone State (0 ou 1)																															b0	
1	0	0	0	1	1	1	1	1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0

**33** Le décodage hexadécimal-binaire

« application ») dans une trame capturée sur un réseau Ethernet (avec le protocole applicatif fourni) ainsi que les caractéristiques essentielles des couches :

- **Transport** : numéros de port UDP et longueur du datagramme
- **Réseau** : adresses IPv4, version et protocole encapsulé
- **Liaison de données** : adresses MAC et protocole encapsulé

L'élève pourra aussi estimer le taux d'occupation d'un canal de communication à partir du relevé des débits sur ce canal et de la connaissance de la bande passante du même canal.

#### Un support attractif

Le système à enseigner AR.Drone est riche en activités pour l'enseignement technologique commun en STI2D,

ainsi que pour les spécialités SIN et ITEC. Des activités ont été également réalisées dans l'esprit « analyser, modéliser, expérimenter et communiquer » du nouveau bac SSI.

Comme nous l'avons vu, l'AR.Drone est aussi un excellent support pour étudier les principaux diagrammes de la norme SysML, qui servent de fil rouge à une grande partie des activités proposées.

D'autre part, les logiciels spécialement développés pour chaque activité pratique permettent une prise en main rapide et ludique du système. Et l'utilisation de la tablette tactile améliore l'interactivité de l'élève et du système, facilitant ainsi les apprentissages.

Enfin, ce système permettra au plus grand nombre d'appréhender de façon innovante les nouvelles technologies de la communication et de la mobilité. ■



# DES RESSOURCES POUR **ENSEIGNER**

Programmes • Disciplines • Formation • Outils pour la classe • Évaluations