

BTS INFORMATIQUE ET RESEAUX

POUR L'INDUSTRIE ET LES SERVICES TECHNIQUES

Session 2007

Épreuve E.4

Étude d'un Système Informatisé

Prélèvements sur sites volcaniques

Sujet

Durée : 6h00

Coefficient 5

"Calculatrice autorisée (conformément à la circulaire n° 99-186 du 16 novembre 1999)."

Aucun document autorisé.

Ce document comprend 43 pages composées de :

Sujet : pages 1 à 18 sur papier rose
Annexes : pages 1 à 14 sur papier vert
Document réponse : pages 1 à 13 sur papier blanc
à rendre obligatoirement (même vierge) .

Toutes les réponses sont à fournir sur le livret « Document Réponse » à l'exclusion de tout autre support.

Les réponses doivent être exclusivement fournies dans les emplacements prévus à cet effet. Si nécessaire le candidat a la possibilité de rectifier ses réponses sur la page non imprimée en regard.

On ne justifiera une réponse que si le document le demande.

Temps conseillés et barème indicatif :

Lecture du sujet	30 minutes		
B. Le robot mobile	90 minutes	23 questions	40 points
C. Contrôle-Commande	90 minutes	8 questions	25 points
D. Communication et réseau	90 minutes	14 questions	35 points
Relecture	60 minutes		

A. Présentation du Système

A.1. Expression du besoin

Depuis l'antiquité, l'activité volcanique a été examinée et répertoriée du fait même de l'impact des éruptions sur l'activité humaine. Les habitants des zones à risques vivaient dans la peur d'éruptions aussi soudaines qu'imprévisibles qui détruisaient leurs champs et leurs villages et causaient d'importantes pertes humaines.

Quelques personnes s'intéressèrent alors de plus près aux volcans et s'aventurèrent auprès des cratères dans le but de comprendre les phénomènes volcaniques. Dès lors, de nombreux scientifiques payèrent de leur vie des observations faites à proximité des zones d'éruptions.

La fin du XX^{ème} siècle notamment a coûté la vie à de nombreux vulcanologues du fait d'un regain en fréquence comme en intensité de l'activité volcanique.

1500 volcans potentiellement en activité sont répertoriés sur la planète, dont 500 sont effectivement rentrés en activité au cours du XX^{ème} siècle et 70 actifs et en phase éruptive à l'heure actuelle. 10% de la population terrestre vit sous la menace des volcans qui ont coûté la vie à 30.000 personnes au cours des 50 dernières années.

Des avancées en matière de prévision et de prédiction des éruptions ont été faites récemment grâce à l'étude des grandes éruptions. La surveillance des zones à risques est cependant insuffisante et une catastrophe volcanique peut intervenir à tout moment.

C'est ce constat, allié avec des avancées technologiques récentes dans le domaine de la robotique qui a conduit la Communauté Européenne à mener un nouveau projet nommé ROBOVOLC dont le but est l'étude et la réalisation d'un robot mobile pour l'exploration volcanique. Ce projet débuté en mars 2000 rassemble plusieurs partenaires dont des universités, des laboratoires de recherche et des entreprises privées.

L'objectif majeur du robot étudié est de minimiser les risques pris par les vulcanologues et les techniciens impliqués dans des activités à proximité des cratères en phase éruptive. Il est à noter que les observations les plus intéressantes sont faites au cours des phases paroxysmiques des éruptions, au cours desquelles le risque est bien entendu maximum.

A.2. Système technique

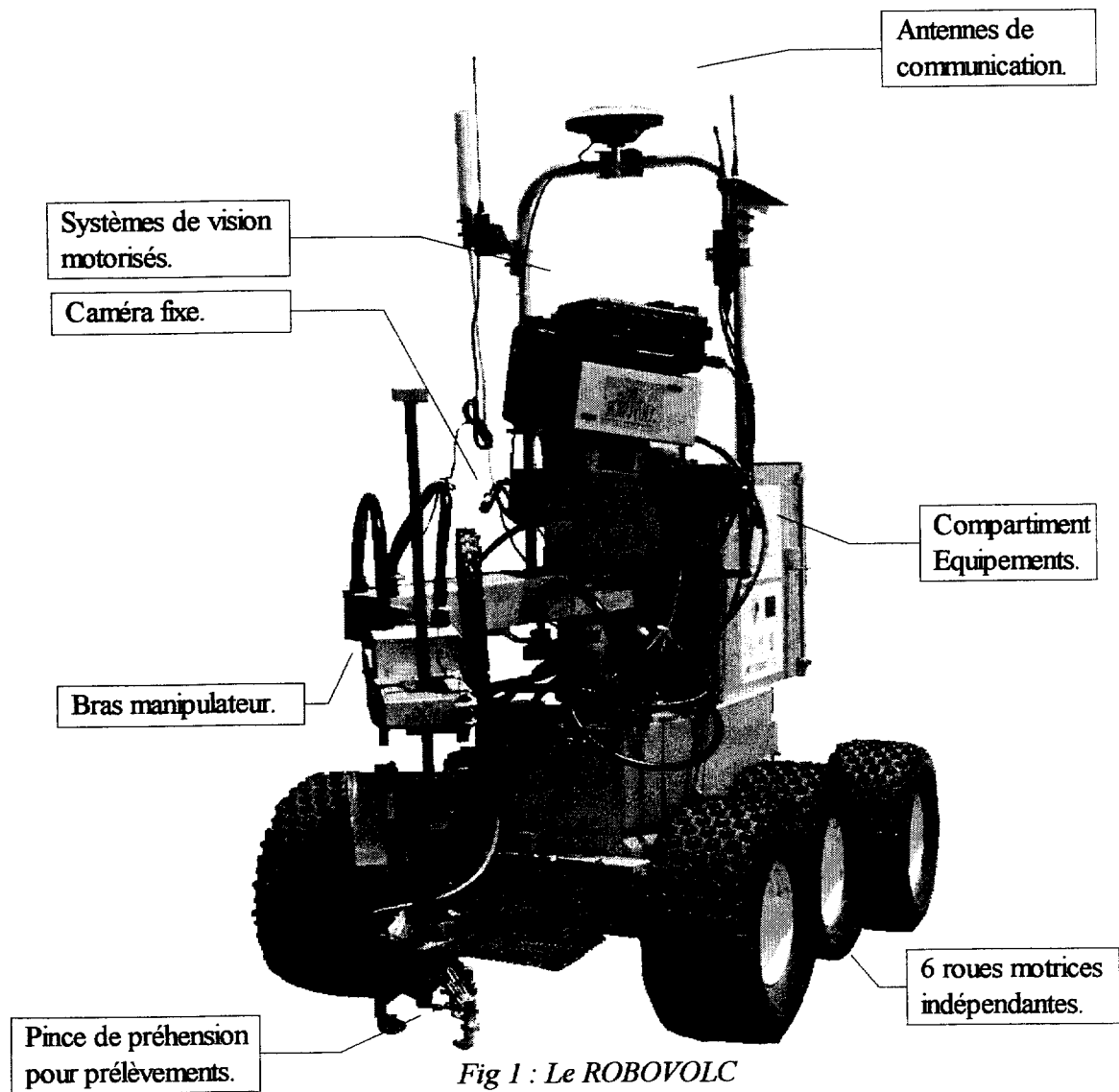
Le cahier des charges établi par l'ensemble des partenaires spécifie que le robot doit être capable de :

- S'approcher d'un cratère actif,
- Collecter des échantillons de rejets éruptifs,
- Collecter des échantillons gazeux,
- Collecter des données physiques et chimiques,
- Surveiller une bouche de cratère.

Lors de l'étude préliminaire, et pour atteindre ces objectifs, l'équipe de ROBOVOLC a décomposé le système en plusieurs sous-systèmes :

- Sous-système de déplacement,
- Sous-système de navigation,
- Sous-système de vision,
- Sous-système de prélèvement,
- Sous-système de communication,
- Sous-système de supervision.

Le sujet s'intéressera à la problématique du déplacement d'un point A à un point B, sur un site volcanique, ce qui l'amènera à aborder les sous-systèmes de déplacement, de navigation, de communication et de supervision du robot.



L'ensemble des éléments visibles sur cette photographie est repris dans le diagramme de déploiement ci-après.

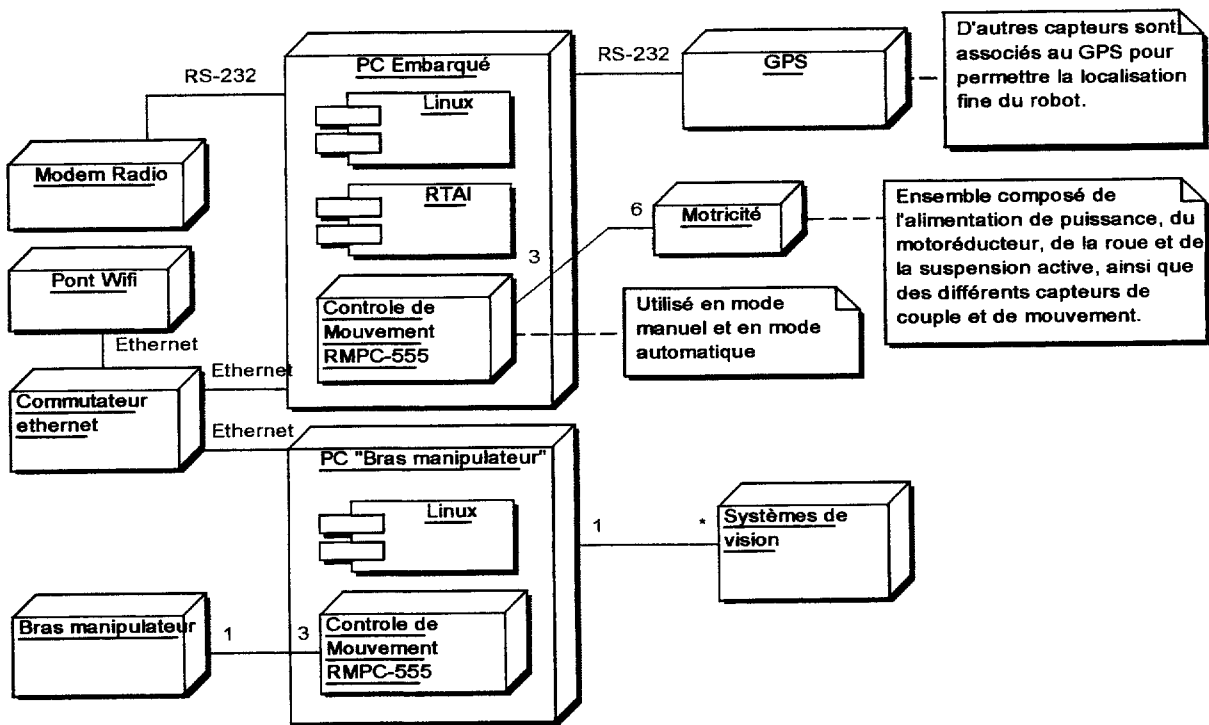


Fig 2 : Diagramme de déploiement du robot mobile.

La partie mobile est contrôlée à distance par un poste de contrôle localisé dans une zone sécurisée. Ci-dessous, le diagramme de déploiement de cette partie :

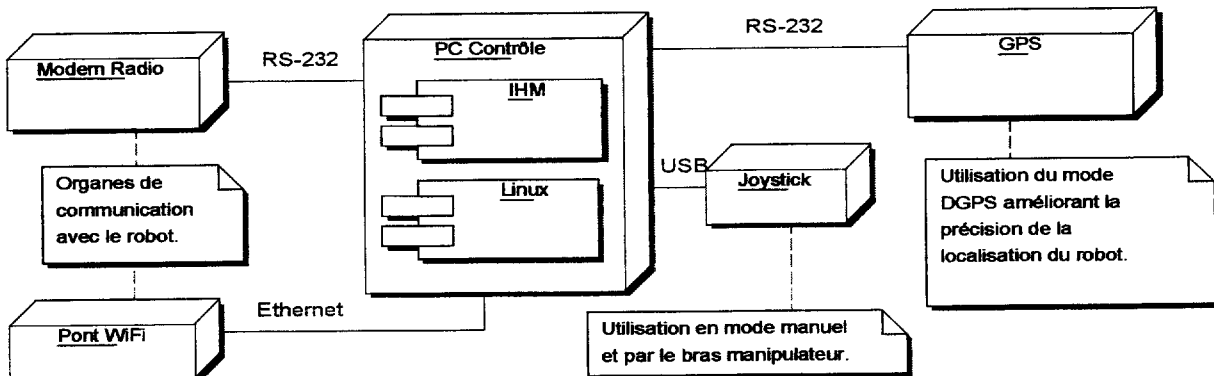


Fig 3: Diagramme de déploiement de la partie contrôle.

B. Le robot mobile

Le robot mobile est piloté à distance depuis le poste de contrôle. L'opérateur visualise en permanence les images transmises par la caméra embarquée, et reçoit cycliquement des informations sur la position géographique du robot.

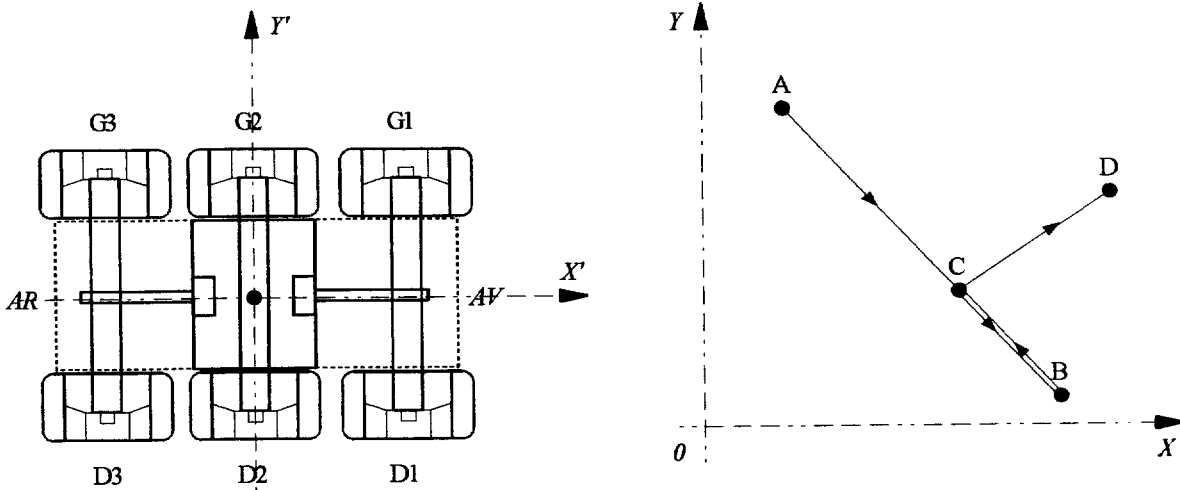
Ces informations sont obtenues localement sur le robot par un système GPS (*Global Positioning System*), et récupérées sur le poste de pilotage par l'intermédiaire de la liaison radio.

Pour ses déplacements, le robot est soit en mode automatique (il se dirige automatiquement vers un point géographique qui lui a été spécifié), soit en mode manuel (il est piloté manuellement, à distance, par l'opérateur).

B.1. Motorisation

Dans ce qui suit, nous nous intéressons exclusivement à la gestion des trajectoires du robot. Celui-ci est équipé de 3 essieux articulés. Chacune des six roues dispose d'une motorisation indépendante avec asservissement de vitesse.

Les figures suivantes montrent le robot en vue de dessus avec la dénomination des roues par rapport au sens normal d'avancement, et un exemple de trajectoire manuelle passant respectivement par les points A, B, C et D.



D'après les schémas ci-dessus, et sachant qu'une consigne positive fait tourner la roue dans le sens d'avancement du robot, on souhaite faire suivre au robot la trajectoire ABCD.

Question B.1.1.

Sur le document réponse, compléter le tableau exprimant les consignes à appliquer aux 6 roues à l'aide des symboles - (recule), 0 (arrêt), + (avance)

Dans la réalité, pour piloter le robot, il est nécessaire de contrôler finement la vitesse de rotation de chaque roue afin de minimiser les glissements, notamment en mode automatique, lorsque le robot doit suivre un cap de manière autonome...

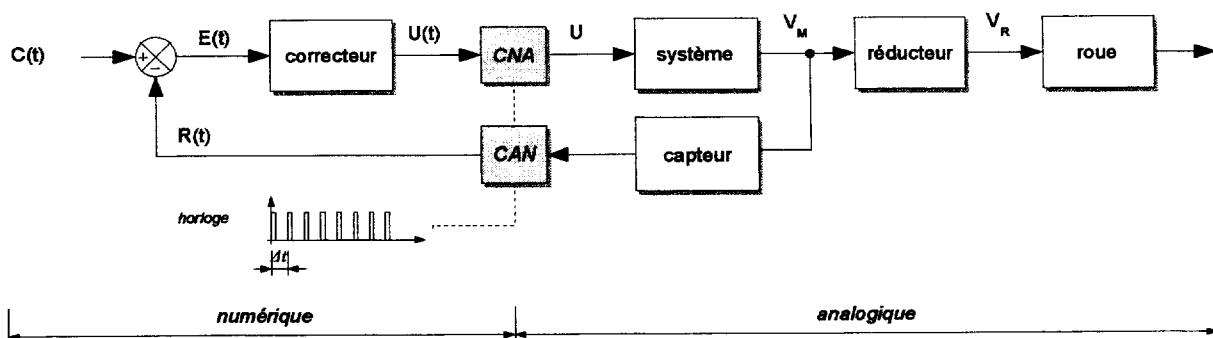
Les roues sont équipées de pneumatiques spéciaux dont le diamètre extérieur D est de 300 mm. On suppose un déplacement sans glissement ni patinage et on souhaite appliquer aux roues une consigne de vitesse Ω_R pour obtenir une vitesse de déplacement V_D en ligne droite égale à 0,2 mètres par secondes.

On rappelle que $V_D = \Omega_R * R$

Question B.1.2.

Exprimer Ω_R en fonction de D et de V_D . Puis faire l'application numérique en radian par secondes (rad/s) et convertir le résultat en tours par minute (trs/min).

Le système d'asservissement qui équipe chacune des roues est destiné à contrôler la vitesse de rotation de la roue, et doit permettre au système embarqué de détecter un glissement (manque d'adhérence) ou un patinage de celle-ci (comme par exemple quand la situation du robot fait que momentanément la roue ne touche plus le sol...).



Le bloc « système » représenté sur le schéma précédent est constitué d'un variateur et d'un moteur. Ce dernier est accouplé à un réducteur qui entraîne à son tour la roue. Le capteur permet d'obtenir une grandeur proportionnelle à la vitesse de rotation réelle de la roue, il est accouplé à l'axe de sortie du moteur.

La valeur $U(t)$ en entrée du CNA (Convertisseur Numérique-Analogique) est codée sous forme d'un entier non signé sur 16 bits, elle est convertie en grandeur analogique U telle que $-10V \leq U \leq +10V$ pour $U(t)$ évoluant de $0x0000$ à sa valeur maximale $0xFFFF$.

Question B.1.3.

Quelle valeur numérique de $U(t)$ faut-il appliquer pour obtenir une valeur nulle en sortie de CNA ?

Question B.1.4.

A quelle consigne U correspond la valeur hexadécimale d'entrée $U(t) = 0xA000$?

Le capteur utilisé pour mesurer la vitesse de rotation est de type dynamo tachymétrique, ce choix répond aux exigences de tenue en température et de robustesse. Ce capteur fournit une tension directement proportionnelle à la vitesse de rotation de la roue, cette tension varie au maximum entre $-610mV$ et $+650mV$.

Le CAN (Convertisseur Analogique-Numérique) employé possède plusieurs canaux de conversion A/N 12 bits d'une linéarité de ± 1 bit. Le temps de conversion par canal est de $25\mu s$.

- plage d'entrée réglable matériellement : $0-10V$, $\pm 5V$, $\pm 10V$
- gain appliqué aux entrées configurables logiquement : $\times 1$, $\times 10$, $\times 100$

Question B.1.5.

Indiquer sur le tableau du document réponse les combinaisons valides de choix de plage et de gain cohérentes avec la tension retournée par la dynamo tachymétrique.

Question B.1.6.

Parmi les configurations valides ci-dessus, quelle est celle qui semble la mieux adaptée au problème ?

Question B.1.7.

Calculer la résolution en mV du CAN pour la configuration retenue.

Question B.1.8.

Quel(s) autre(s) type(s) de capteur peut être employé pour mesurer une vitesse de rotation ?

B.2. Localisation

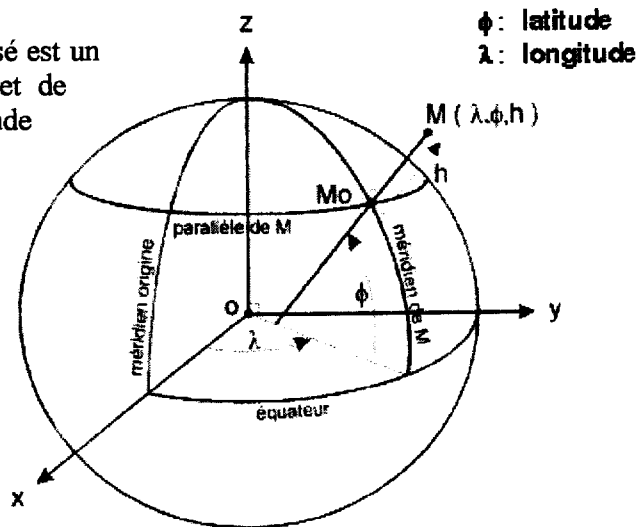
Les trajectoires du robot sont définies par des points cartographiques. Les positions à atteindre sont de la forme (x,y) , les coordonnées retournées par le robot de la forme (x,y,z) .

Le système de coordonnées géographiques utilisé est un quadrillage (non rectangulaire) de méridiens et de parallèles. Une position est définie par une latitude Nord-Sud et une longitude Est-Ouest exprimées en général en degrés par rapport à l'équateur et au méridien d'origine (Greenwich).

Les cartes intégrées au système distant de pilotage utilisent le système de coordonnées planes UTM (*Universal Transverse Mercator*) WGS84, en km. Le récepteur GPS implanté sur le robot utilise le standard NMEA (*National Marine & Electronics Association*) dans sa version NMEA-0183, pour l'émission de ses données.

Sous ce standard, toutes les données sont transmises en mode série asynchrone sous forme de trames de caractères ASCII. Les trames commencent par le caractère '\$' et se terminent par les caractères CR (0x0D) et LF (0x0A).

Le protocole utilisé est 4800 bauds série asynchrone, sans parité, 8 bits de données et 1 bit de stop.



Question B.2.1.

Tracer le chronogramme de transmission des caractères CR et LF. Identifier clairement sous le chronogramme les bits de données sous la forme $b7..b0$.

Calculer en ms la durée de transmission d'un bit, notée Mt sur le document réponse.

Le récepteur GPS est interfacé avec le système informatique du robot par un circuit UART (*Universal Asynchronous Receiver / Transmitter*) de type 16550A à l'adresse 0x03F8. Le circuit est cadencé par un quartz de 1,8432 Mhz.

Question B.2.2.

A partir de l'annexe A.1, compléter le tableau réponse de manière à configurer l'UART pour communiquer avec le GPS. Pour chaque ligne, indiquer le nom du registre concerné, son adresse relative, et la valeur hexadécimale à y écrire.

L'annexe A.2 présente le format des trames NMEA transmises par le GPS sur la liaison série. Le système se doit de contrôler la validité des trames reçues en recalculant la somme de contrôle lorsqu'elle existe.

Question B.2.3.

Proposer le prototype en langage C d'une fonction `sommeDeControle()` qui reçoit en argument un pointeur sur le début d'une trame, et qui calcule puis retourne la somme de contrôle de cette trame sous forme d'un mot de 8 bits non signé.

Question B.2.4.

Proposer une implémentation de cette fonction, en considérant que la présence dans la trame des marqueurs '\$' et '*' est vérifiée au préalable par le système utilisant la fonction.

Au cours d'une mission, le robot reçoit la trame GLL suivante en provenance de son récepteur GPS:

\$GPGLL,2114.35,S,05540.305,E,093518,A,*2F<CR><LF>

Question B.2.5.

Exprimer en degrés, minutes, secondes les coordonnées géographiques du robot à cet instant.

Question B.2.6.

En tenant compte des caractéristiques du protocole série asynchrone utilisé, calculer le temps de transmission d'une trame constituée de N caractères.

Question B.2.7.

En déduire le temps de transmission de la trame GLL précédente.

B.3. Communication entre tâches

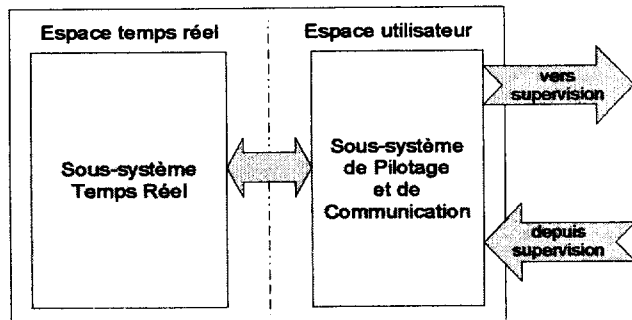
RTAI (*Real Time Application Interface*) est une extension libre du noyau Linux lui apportant des fonctionnalités temps réel dures.

RTAI est basé sur la notion de « double noyau » :

- Un noyau temps réel à priorités fixes chargé de gérer les tâches temps-réel dur.
- Le noyau Linux est considéré comme une tâche de plus faible priorité par le noyau temps-réel et il conserve la gestion des tâches situées dans l'espace utilisateur.

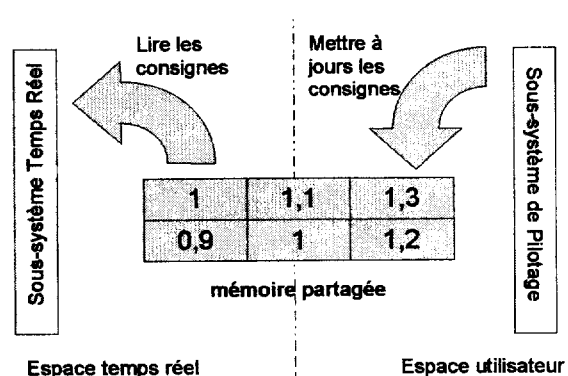
Le robot possède pour son fonctionnement deux sous-systèmes distincts :

- Le sous-système temps réel : pilotage temps réel des six roues et du bras de manipulation.
- Le sous-système non temps réel : communication avec le monde extérieur et transmission d'ordre au sous-système temps réel.



Robot

Le système de pilotage transmet des jeux de 6 valeurs de consigne des roues du robot. La communication entre le sous-système temps réel et le sous-système non temps réel de commande se fait grâce à l'utilisation d'une zone de mémoire partagée :



Deux tâches s'exécutent en parallèle :

- Le sous-système temps réel lit à période fixe les 6 consignes dans la mémoire partagée,
- Le sous-système de pilotage non temps réel écrit dans cette même mémoire partagée les 6 consignes à une fréquence voisine.

Le sous-système non temps réel écrit dans la mémoire partagée à partir de l'espace utilisateur, il est susceptible d'être interrompu au cours de cette écriture par le sous-système temps réel.

Question B.3.1.

En conséquence, quel est le risque encouru par l'application ?

L'écriture et la lecture de la mémoire partagée sont considérées comme des sections critiques, elles sont protégées par un sémaphore.

Question B.3.2.

Qu'appelle-t-on une section critique dans un système multitâche ?

L'annexe A3 présente un extrait de la bibliothèque RTAI relatif à la mise en oeuvre des sémaphores.

Question B.3.3.

Citer les fonctions permettant, depuis l'espace utilisateur, de réaliser les opérations présentées dans le tableau du document réponse.

Question B.3.4.

Implémenter une fonction de prototype `bool initSem(int numSem)` permettant d'initialiser le sémaphore `num` à la valeur 1, et retournant vrai en cas de succès ou faux en cas d'échec.

Soit la structure de la mémoire partagée suivante :

```
struct ShmMemWrite
{
    float v[6] ;
};
```

```
typedef struct
{
    float v[6] ;
} ShmMemWrite;
```

Forme C Ansi
équivalente.

Question B.3.5.

Insérer dans la fonction

`void envoyerConsignes(ShmMemWrite* shm, float* cons, int numSem)`
les lignes de code permettant de rendre la séquence d'écriture des six vitesses non interruptible.