

A. Présentation du Système

A.1 Expression du besoin

La société X, sous-traitante de l'industrie automobile, fabrique des pièces en ABS (matière synthétique) métallisé (insignes, éléments de décoration, etc.).

Le procédé de fabrication (métallisation de pièces en ABS, apparenté à la galvanoplastie) fait appel à une unité de production développée par la société **TUBALEX**. Le sujet se place du point de vue d'un technicien IRIS en charge de la gestion et de la maintenance de la ligne de production.

A.2 Principe général

Le procédé se décompose en quatre grandes phases :

1. Prétraitement de la surface dans un bain d'acide sulfochromique (afin de rendre cette surface rugueuse).
2. Désoxydation de la surface,
3. Post-activation de la surface par dépôt électrolytique d'un film riche en cuivre, très conducteur,
4. Dépôt électrolytique de la couche de finition, en l'occurrence du chrome.

Chacune de ces phases demande l'immersion des pièces à traiter dans un bain, pour une durée déterminée, différente pour chaque phase. Ces différentes phases sont en général séparées par des cuves de rinçage.

Le nombre de bains identiques pour une phase donnée dépend de la durée relative de cette phase dans la chaîne. Chaque outillage n'est immergé que dans un bain par phase.

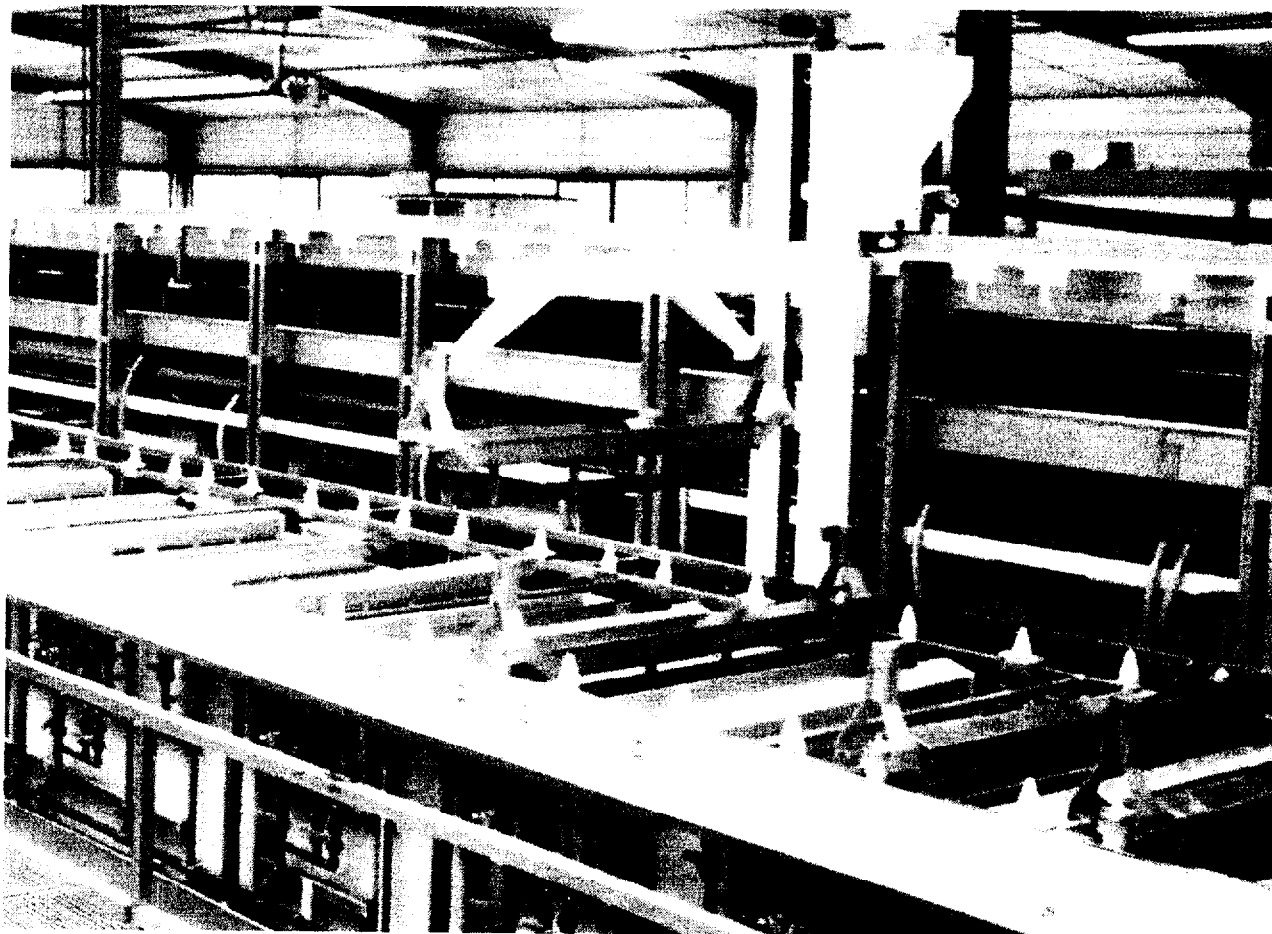


Fig. 1 : Photo de l'installation.

A.3 Détail de l'installation

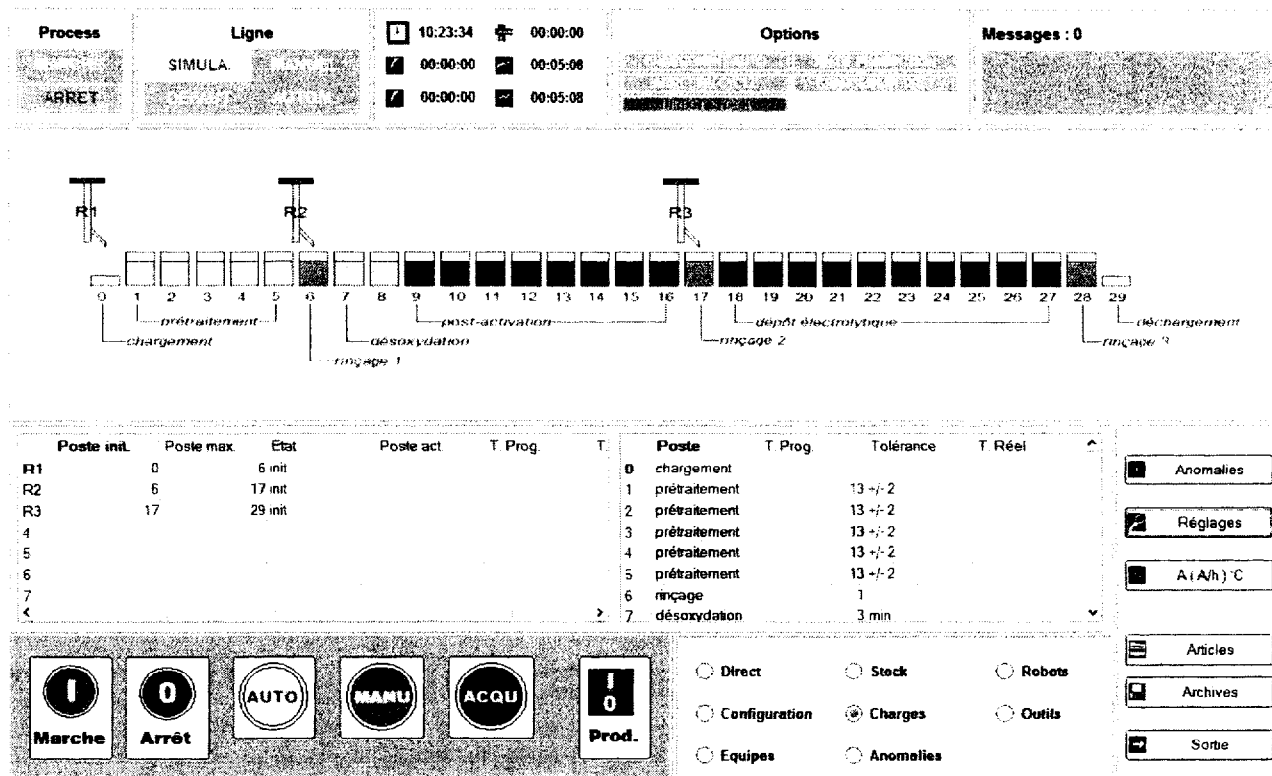


Fig. 2 : Capture d'écran du poste de contrôle/commande (pour information).

La ligne de fabrication à laquelle nous nous intéressons est composée de 28 bains alignés, numérotés de 1 à 28, précédés d'un poste de chargement de numéro 0 et suivis d'un poste de déchargement de numéro 29.

L'espace séparant un bain du bain suivant est de 50 centimètres. Chaque bain est équipé de capteurs et d'actionneurs permettant de

- Remplir/Vidanger/Vérifier les niveaux du bain,
- Chauffer/Refroidir les bains,
- Contrôler l'électrolyse,
- Contrôler la présence de l'outillage.

L'ensemble de ces capteurs/actionneurs est piloté par un boîtier d'Entrées/Sorties situé à proximité du bain.

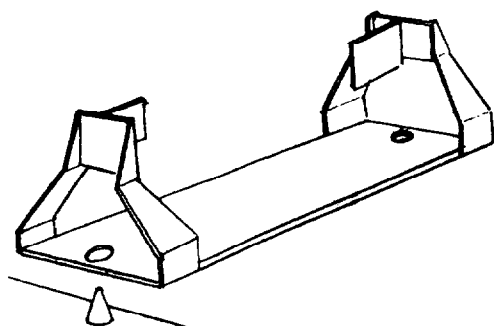


Fig. 3 : Détail d'un outillage.

Les pièces à métalliser sont fixées par grappes à un **outillage** (encore appelé **portant**) qui peut être déposé précisément sur un bain par le biais de plots de centrage.

Ces portants sont manipulés par des robots deux axes équipés d'une potence disposant d'un mécanisme de portage simple ne demandant aucune manipulation extérieure.

Note : Par la suite on ne parlera plus que d'outillage.

Ce mécanisme permet au robot d'immerger un outillage dans un bain, de le sortir, de l'égoutter et de le transporter d'un bain vers un autre.

La ligne de fabrication est équipée de trois robots (R1, R2 et R3). Chaque robot possède une zone d'influence limitée lui permettant de desservir plusieurs bains, et présentant une intersection d'un bain avec les robots voisins.

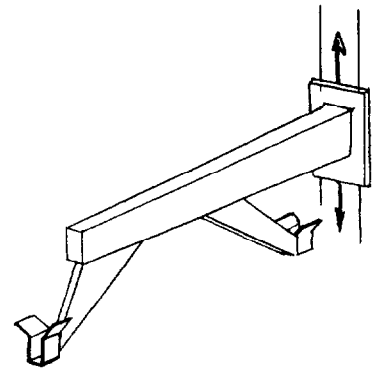


Fig. 4 : Détail d'une potence.

Chaque robot est équipé de capteurs et d'actionneurs permettant le mouvement de la potence sur la ligne (horizontalement et verticalement) . Ces capteurs et actionneurs sont pilotés par le biais d'un module CAN fixé sur le robot...

A.4 Synoptique général

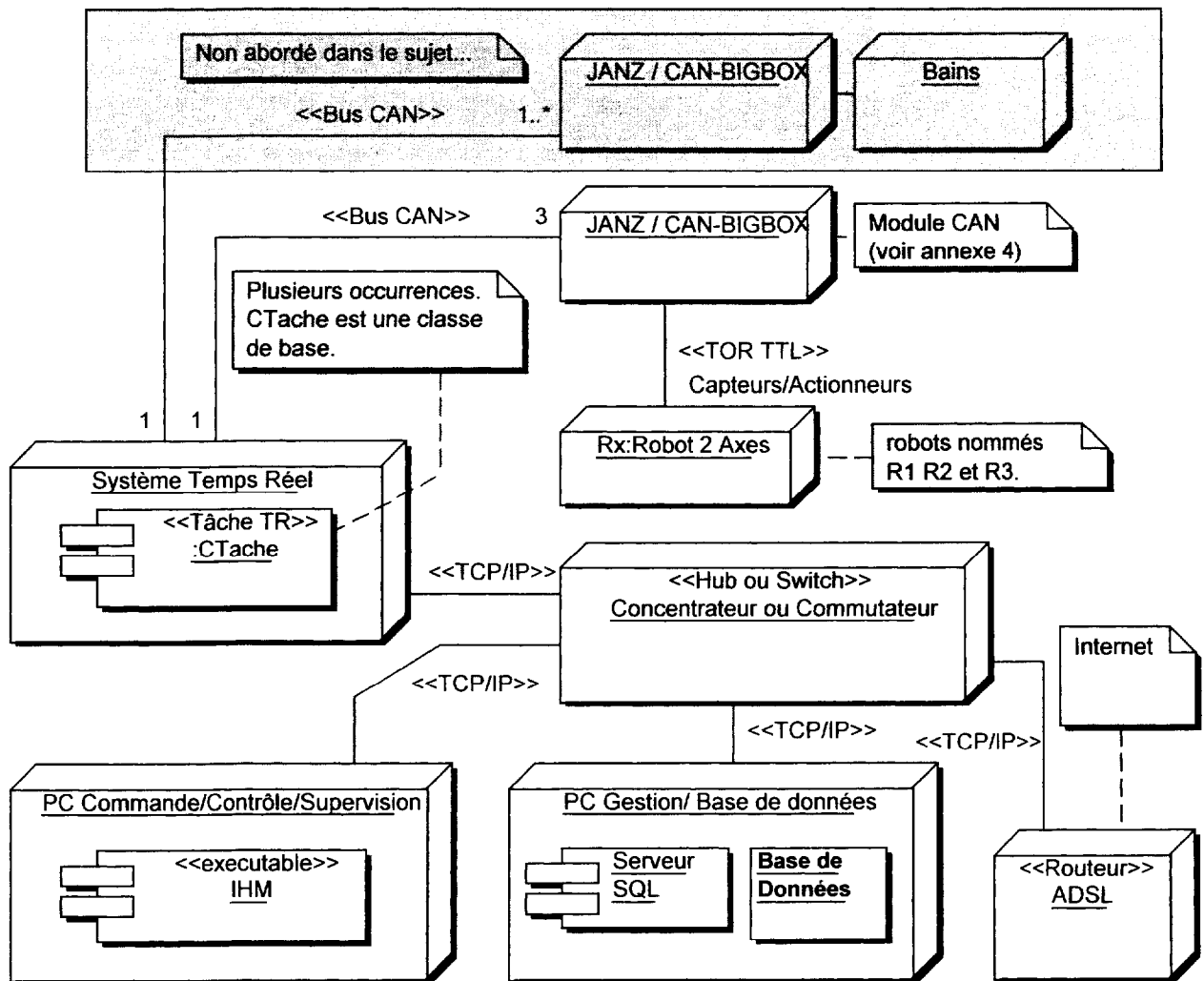


Fig. 5 : Diagramme de déploiement.

B. Analyse du Système

B.1 Répartition des bains

Les 30 postes (28 bains + chargement et déchargement) sont répartis de la manière suivante :

N° du poste	Phase	Durée en minutes	Remarques
0	Chargement		
1-5	Prétraitement	13'±2'	5 postes identiques (traitement //)
6	Rinçage 1	1'	Zone de partage entre R1 et R2
7-8	Désoxydation	3'	2 postes identiques (traitement //)
9-16	Post-activation	21'±2'	8 postes identiques (traitement //)
17	Rinçage 2	1'	Zone de partage entre R2 et R3
18-27	Dépôt électrolytique	28'±3'	10 postes identiques (traitement //)
28	Rinçage 3	1'	
29	Déchargement		

Rappel : Chaque outillage n'est immergé que dans un bain par phase.

On appelle **transfert** l'action consistant à déplacer un outillage d'un poste vers un autre poste.

Chaque transfert est composé de la séquence suivante (voir annexe 2):

- Déplacement horizontal du robot vers le poste source.
- Déplacement vertical du robot vers l'outillage (descente).
- Accrochage de l'outillage.
- Déplacement vertical pour sortie de bain de l'outillage (montée et égouttage).
- Déplacement vertical de dégagement du robot (montée).
- Déplacement horizontal du robot vers le poste destination.
- Déplacement vertical du robot vers le bain (descente).
- Décrochage de l'outillage.
- Déplacement vertical de dégagement du robot (montée).

L'ensemble de ces opérations requiert une durée de 1 minute. Seuls les transferts au niveau des postes de chargement et de déchargement ne respectent pas exactement ces opérations.

Question B.1.1

En analysant le tableau ci-dessus, déterminer le nombre de transferts que devra subir un outillage entre le poste de chargement et le poste de déchargement.

Question B.1.2

Calculer les durées théoriques minimale et nominale nécessaires à la production d'une pièce sur la chaîne, hors phase de chargement et de déchargement.

En entrée de la chaîne, un outillage avec ses grappes de pièces à métalliser est disponible toutes les 3 minutes.

Question B.1.3

Justifier numériquement le choix du nombre de bains pour chacune des différentes phases, hors phase de chargement et de déchargement.

B.2 Mouvement du robot R1

Le robot R1 a la particularité d'être responsable du poste de chargement. La mise en place de l'outillage se fait manuellement, et bien que l'opération soit simple et ne nécessite que quelques secondes, on souhaite que le robot passe le plus de temps possible au poste de chargement afin de permettre aux opérateurs de faire leur travail dans les meilleures conditions possibles.

Question B. 2.1

Etablir le nombre des transferts que doit effectuer le robot R1 pour un outillage donné. Préciser succinctement ces transferts.

On s'intéresse maintenant au déplacement horizontal du robot R1 entre le poste 0 et l'un des postes 1 à 5. L'accélération du robot sur cet axe est supposée constante et égale à 2 m/s^2 . La décélération est de 8 m/s^2 . La vitesse maximale est de 1 m/s , une vitesse plus faible ($0,2 \text{ m/s}$) étant engagée à l'approche du poste à desservir. (voir Annexe 7 : Formulaire de mécanique)

Question B.2.2

Calculer le temps nécessaire au robot, partant de l'arrêt complet, pour atteindre sa vitesse maximale.

Question B.2.3

Calculer la distance parcourue par le robot pendant cette phase d'accélération.

Question B.2.4

Calculer la distance parcourue pour passer de la vitesse maximale à la vitesse faible.

Question B.2.5

Calculer la distance parcourue par le robot pour passer de la vitesse faible à l'arrêt.

Des capteurs de type inductif sont placés sur les robots afin de détecter des drapeaux métalliques situés dans l'axe du poste pour l'arrêt, ainsi qu'à dix centimètres de part et d'autre de l'axe du poste. Les marqueurs sont nommés G_n , C_n et D_n avec n numéro du bain (voir annexe 1).

Question B.2.6

Quel est le principe de fonctionnement d'un capteur inductif ?

Question B.2.7

Quel est l'intérêt de ce type de capteur dans l'environnement industriel étudié ?

C. Communication : Bus CAN

Les divers capteurs et actionneurs de chaque robot sont reliés aux entrées/sorties d'un module CAN-BIGBOX fabriqué par la société JANZ (voir Annexe 1 et Annexe 6).

C.1 Commande des moteurs et Lecture des capteurs

Pour lire les entrées, il faut demander au module CAN-BIGBOX via le bus CAN d'effectuer une lecture à l'adresse \$1000 (1000 en hexadécimal) ; pour piloter un actionneur, de réaliser une écriture à l'adresse \$1001.

Question C.1.1

Le capteur Z d'un robot est représenté par le bit de poids faible de l'octet situé à l'adresse \$1000 (une valeur logique 1 indique que le robot est en position d'initialisation). Proposer une opération logique (opérateur et opérande) afin d'isoler ce bit dans l'octet lu.

L'octet permettant de piloter les actionneurs d'un robot est organisé comme suit :

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Marche(1) Arrêt (0)	X	Rotation : Horaire (1) Trigo (0)	X	Vitesse : Grande (1) Petite (0)	X	X	X

Note 1 : un arrêt n'est pas obligatoire lors d'un changement de vitesse

Note 2 : X = Bit non utilisé. Valeur indifférente.

Question C.1.2

Quelles valeurs hexadécimales doit prendre successivement cet octet pour ordonner un déplacement en petite vitesse puis en grande vitesse dans le sens horaire ?

Question C.1.3

Vérifier l'adéquation de ce module aux besoins de l'application. Indiquer pour quelles raisons les autres modules proposés n'ont pas été retenus. (voir annexe 4 - Equipements : documents constructeurs)

C.2 Dépose et Reprise d'outillage

UML permet de représenter graphiquement le comportement d'une méthode ou le déroulement d'un cas d'utilisation, à l'aide de diagrammes d'activités (une variante des diagrammes d'état-transitions).

Une activité représente une exécution d'un mécanisme, un déroulement d'étapes séquentielles. Le passage d'une activité vers une autre est matérialisé par une transition qui peut être conditionnelle ou automatique.

Les transitions sont déclenchées par la fin d'une activité et provoquent le début d'une autre activité.

Question C.2

On fournit en annexe 3 le diagramme d'activité décrivant les opérations élémentaires nécessaires à la prise d'un outillage. Réaliser sur le même modèle et en vous aidant de l'annexe 2, le diagramme d'activités lié à la dépose d'un outillage.

C.3 Protocole CAN

Le dialogue entre le système temps réel et un module CAN (ordre ou information capteur) est basé sur des trames qui comportent un champ de données de 16 bits exactement (voir annexe 1).

Question C.3.1

En vous référant à la documentation du bus CAN fournie en annexe 5 et aux informations ci-dessus, déterminer la longueur L en bits d'une trame de données circulant sur le bus (on ne tient pas compte des problèmes de *bit-stuffing*).

Question C.3.2

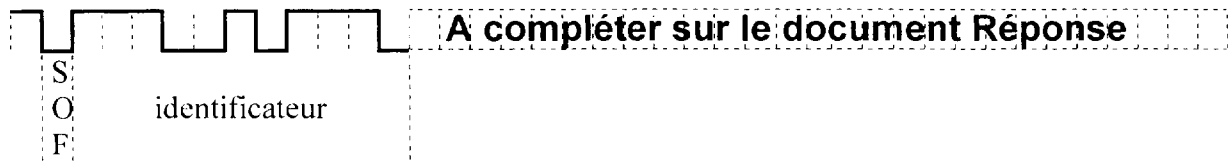
En déduire le nombre maximum de trames pouvant circuler sur le bus en une seconde.

Question C.3.3

Dans le cas le plus défavorable d'un déplacement horizontal entre des bords non voisins, chacun des trois modules CAN affectés aux robots émet et reçoit un maximum de 8 trames par seconde. Déterminer le taux de charge maximum du bus.

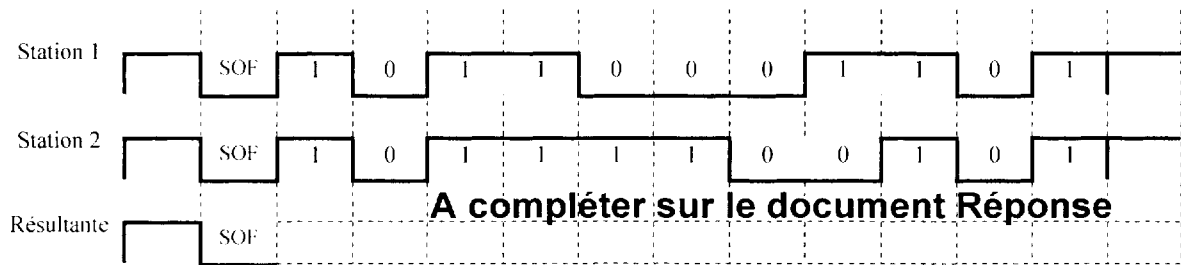
Question C.3.4

On donne le SOF et le champ d'arbitrage d'une trame contenant l'ordre « A\006 » (Caractère ASCII 'A' (41H) suivi de l'octet de valeur 6). Compléter la trame jusqu'au champ CRC exclu. (on ne tient pas compte de la règle du « stuffing »)



Question C.3.5

Deux stations désirent émettre une trame dont les chronogrammes sont donnés ci-après. Dans le cas où ces trames seraient émises en même temps, le mécanisme d'arbitrage du bus CAN va résoudre le conflit (voir annexe 5). Compléter le troisième chronogramme (résultante sur le bus) et indiquer quelle station a réussi à émettre sa trame.



D. Système temps réel

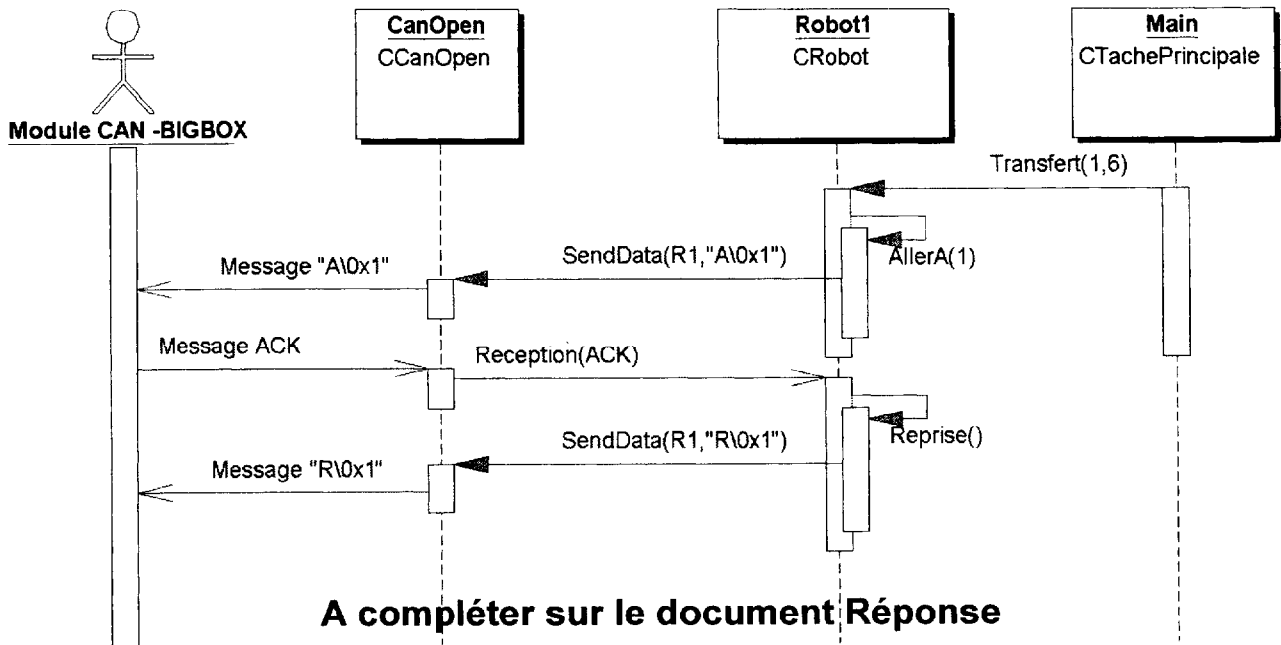
Le système temps réel prend en charge la gestion de tous les modules d'entrée-sortie reliés au bus CAN. Il est lui-même équipé d'une carte CAN qui sera gérée par un objet de type CCanOpen.

CCanOpen	
...	...
+	SendData(int Id, char* Msg) : void
+	OnReceive(int Id, char* Msg) : void
...	...

Nota: OnReceive est appelée de manière asynchrone sur réception d'une trame sur le bus CAN.

Question D.1

Complétez le diagramme de séquence ci-dessous, qui décrit une opération de transfert d'un outillage entre les bays 1 et 6...



Nota : Précision sur les messages (Notation UML 1.4)

- Message synchrone (appel de fonction ou de méthode).
- ⇒ Message asynchrone (lié à un évènement ou à une interruption).

D.2 Analyse d'une situation de panne

A la suite d'un incident, le câble qui relie le capteur horizontal du robot R1 au module CAN-BIGBOX est rompu. Le robot est en position haute.

Le système temps réel émet l'ordre A(1) (Aller au poste 1) vers le module CAN...

Question D.2.1

En l'absence de tout mécanisme de sécurité, que risque-t-il de se passer ?

Question D.2.2

De quelles informations et grandeurs physiques (citées en B.2) doit disposer le système temps réel pour déterminer le délai sous lequel il doit recevoir une information capteur en provenance du module CAN ?

Question D.2.3

Quel mécanisme logiciel des systèmes multi-tâches ou temps réel peut-on mettre en œuvre pour que la tâche en charge de la gestion du robot R1 soit réveillée à la suite de cet incident ?

Question D.2.4

Après ce réveil, quelle décision doit prendre la tâche en question ?