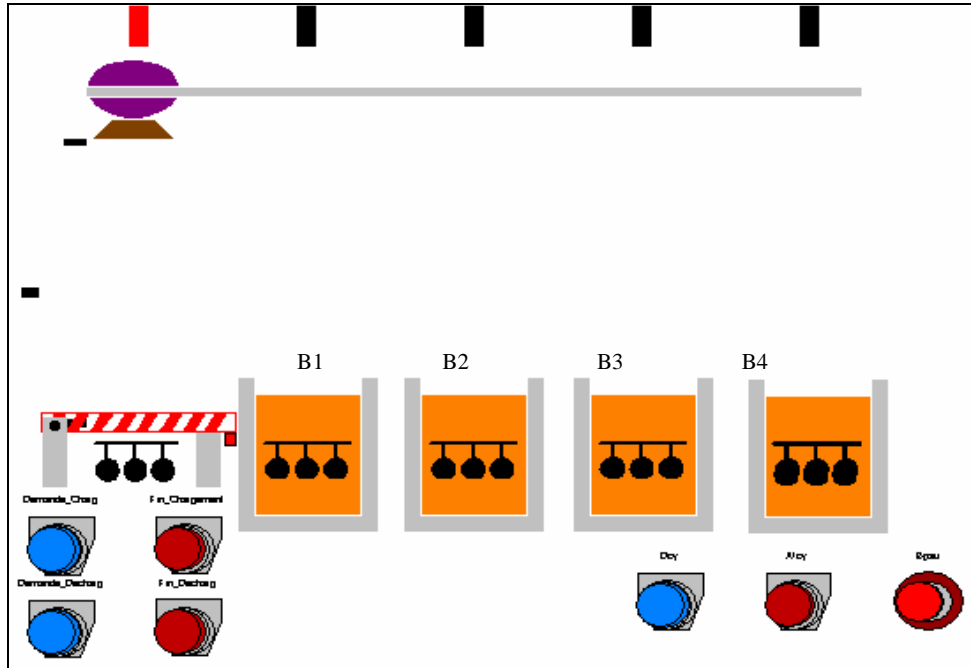


CONTROLBUILD : 2^{ème} partie

PRÉSENTATION DU PROJET :

Une chaîne de traitement de surface est schématisée ci-dessous :



Cette chaîne est constituée de quatre bains de traitement et d'un poste de chargement/déchargement. Les lots de pièces à traiter sont accrochés sur une barre via un bouton poussoir. Une demande de chargement est faite, si elle est accordée la barrière s'ouvre. La barre est située dans la zone de chargement, un BP de fin de chargement doit être enfoncé pour indiquer la fin de l'opération. Les pièces sont ensuite acheminées via un robot muni d'un électro-aimant, dans les bains dans l'ordre B1, B2, B3, B4 afin d'y subir un traitement par électrolyse, puis elles reviennent au poste initial pour le déchargement si le poste est disponible...

Les durées de séjour dans chaque bain sont déterminées par des temporisations, qui sont du même ordre de grandeur. Les temps de déplacement du robot sont négligeables par rapport au temps de traitement. Le maximum de lots doit être traité en même temps.

Le chargement et le déchargement se font manuellement, la production va donc subir des variations de débits qui ne doivent en aucun cas pénaliser le traitement des pièces déjà présentes dans la chaîne...

TRAVAIL À RÉALISER : PARTIE COMMANDE

90 % du code associé à la partie commande a été réalisé. Il ne reste plus qu'à coder un élément : le gestionnaire de priorité qui est réalisé avec une pile FIFO.

Découverte de la partie commande :

- Rechercher le projet : copier et décompresser le projet Bain_Etudiant_Save.zip dans votre dossier personnel Mes documents\Votre Nom\ERII31\TP2_bain\ puis ouvrir le projet Bain_Etudiant.cbp
- Analyse :
 - Sélectionner Applications→Chaine_de_traitement dans la colonne 1
 - Ouvrir le MAC « chaine_de_traitement » dans la colonne 2
 - On peut lancer l'exécution en cliquant sur « l'homme qui court » de chaîne de traitement. Ensuite on ouvre le MAC Partie Opérative et on ouvre : Vue→Synoptique.

Q1 : La représentation SADT de la partie commande associée à la partie opérative est-elle correcte ?

- Ouvrir le MAC « partie_commande »

Q2 : Expliquer les différents éléments constituant ce MAC : le plus simple est sans doute de partir du GEMMA pour déterminer le rôle des autres éléments...

- Ouvrir le MAC « gestion_priorites » : son rôle est de définir à n'importe quel moment le poste (zone de chargement/déchargement ou bain de traitement) le plus prioritaire, c'est à dire le poste qui doit être déchargé par le robot. Étant donné que plusieurs postes peuvent demander le transfert pratiquement en même temps, on utilise une pile FIFO qui va ranger dans une table les demandes faites par les postes.

Q3 : Le MAC « gestion_priorites » contient un appel à un bloc fonctionnel pile FIFO (pour l'instant avec un contenu minimaliste). A quoi correspondent les entrées et sorties de ce BF ?

Q4 : Qu'est ce qu'un bloc fonctionnel ?

- Si l'on clique sur FIFO on se retrouve dans la description du bloc fonctionnel pile FIFO. Il est dangereux d'en modifier le contenu à partir de cet endroit car ici on se trouve dans une instance du Bloc FIFO...

Q5 : Qu'est ce qu'une instance ? Si on observe le contenu du MAC « partie commande », on trouve des MAC « prod », « GEMMA » ... Ces MACS apparaissent aussi dans le groupe PC de la fenêtre principale, quelle est la différence entre les deux ?

- Fermer toutes les fenêtres sauf la fenêtre principale contenant l'architecture du projet. Dans la première colonne, sélectionner Bibliothèques Spécifiques→fg dans la colonne 1. Ouvrir le composant fifo de la colonne 2.

Q6 : Quelle action, pour l'instant, la pile FIFO réalise t-elle ? Vérifier votre réponse en lançant la simulation du procédé complet : ouvrir le MAC « chaine_de_traitement » puis cliquer sur l'icône démarrer, ouvrir le synoptique associé au MAC « partie_operative » Vues→synoptique.

Q7 : Pour compléter le bloc fonctionnel « fifo », on va avoir besoin d'une détection de front montant. On vous demande donc de créer un bloc fonctionnel « detection_front_montant » : dans la bibliothèque spécifique on crée un nouveau composant dans un langage à choisir ? Tester ce composant.

- On va modifier le bloc fonctionnel « fifo » afin qu'il réalise les actions suivantes :
 - Si apparition d'un front montant sur « Ecrire_Valeur », on range dans la pile la valeur « Valeur_A_Ecrire ».
 - Si apparition d'un front montant sur « Lire_Valeur », on positionne sur la sortie « Valeur_Lue », la valeur à sortir qui est égale à la première rentrée. Ceci est possible si la pile n'est pas vide.
 - La sortie « Occupee » doit être vraie si « Ecrire_Valeur » est vraie ou bien si « Lire_Valeur » est vraie. Attention, on ne peut pas sauvegarder une valeur si la pile est pleine sous peine de perdre des données...
 - Si la pile est vide, on positionne la sortie « Vide » à vraie.

Q8 : Sachant qu'il vous faut stocker 5 demandes au maximum (tableau taille = 6), que l'on utilise pour se déplacer dans le tableau un indice « IndIn » lors de l'entrée des valeurs et un indice « IndOut » lors de la sortie des valeurs, que les fronts montants seront détectés avec le bloc fonctionnel « detection_front_montant » précédemment réalisé, modifier le BF « fifo ». Tester le résultat seul, puis dans l'application complète.

Q9 : Maintenant que le projet fonctionne correctement, il vous est demandé d'expliquer les grafjets PROD et ROBOT, cela doit compléter votre réponse Q2... Quelle méthode utilise-t-on dans PROD ? Pourquoi est-elle aussi compliquée ? Quels sont les liens de communication entre les deux grafjets ?

TRAVAIL À RÉALISER : PARTIE OPÉRATIVE

La partie opérative a été simulée de façon simple en une seule règle globale : moteurs+réducteurs+capteurs+aimant.

Notre prochaine étude va donc consister à créer, dans un nouveau projet, un robot conçu à partir de modules élémentaires.

Création d'un nouveau projet :

- Création du projet : Fichier→Nouveau→ dans le répertoire Mes documents\Votre Nom\ERII31\Robot\test_robot.cbp.
- Création de la bibliothèque : afin qu'il soit facilement réutilisable, nous allons créer un robot faisant partie d'une bibliothèque.
 - Colonne 1, sélectionner Bibliothèque spécifique→Nouveau groupe application/biblio(clique droit souris)
 - Créer un groupe nommé B1
 - Sélectionner B1→nouvelle application/biblio→nom : robot, groupe : B1
 - Afin de ranger correctement nos futurs composants, nous allons créer des groupes de composants (Actionneurs, Capteurs, Divers). Robot→Nouveau groupe de composants
- Nous allons maintenant créer le MAC représentant le robot. On sélectionne Robot puis dans la colonne 2 (composant) → Nouveau composant→ nom : Robot, type : MAC.

Création des composants élémentaires :

Le robot peut être décomposé en éléments de bases. Par exemple au niveau des actionneurs, il sera constitué entre autre :

- D'un moteur pour le déplacement horizontal, et d'un moteur pour le déplacement vertical de l'aimant (les caractéristiques de ces deux moteurs sont sans doute proches...).
- De réducteurs
- D'un aimant...

Au niveau des capteurs, la description d'un seul capteur TOR devrait suffire puisque les éléments de détections sont tous identiques

- On sélectionne le groupe capteur → Nouveau composant→nom : CapteurTOR, lang : ST (vous pouvez si vous le souhaitez utiliser un autre langage).
- Le capteur sera modélisé de la même façon que lors du TP1 sur la mise en oeuvre du tapis de convoyage.

Insertion d'un composant dans le MAC Robot :

La finalité est de créer un MAC (modèle d'assemblage de comportement), représentant le fonctionnement du robot. Il nous faut donc compléter ce MAC par des composants élémentaires.

- Le robot est constitué d'une partie Action et d'une partie Détection.
 - On crée donc un composant MAC élémentaire « Actions » (dans le groupe Actionneurs). Il devra par la suite regrouper les moteurs, réducteurs, aimant représentant la partie opérative du robot. Ses entrées seront : CmdMAV, CmdMAR, CmdDes, CmdMont, CmdAccroch, et ses sorties : PositionH (position horizontale du robot), PositionV (position verticale de l'aimant).
 - On pourrait créer un composant « Détection » mais il serait simplement constitué des 7 capteurs (5 capteurs position du robot et 2 capteurs position de l'aimant).
- On ouvre le MAC « Robot » et on y insère le MAC « Actions ». On place aussi 7 « CapteurTOR ». On sauve.
- Dans la fenêtre initiale de controlbuild, on sélectionne le MAC « Robot »→Arbre

Q10 : On souhaite continuer à définir les composants élémentaires. Quelle sont vos propositions pour les composants : moteur, réducteur_linéaire (transformation des tr/mn en m/s, avec un coefficient de réduction), distance_accomplie (en décimètre), aimant ?

Q11 : Créer un MAC « Déplacement_Horizontal ». Intégrer à l'intérieur tous les composants permettant de simuler le déplacement du robot ? (la zone fait 80 mètres de long et les postes sont distants de 20 mètres).

Q12 : Idem pour le MAC « Déplacement_Vertical ». (la hauteur est de 20 mètres).

Q13 : Insérer ces deux MAC dans le MAC « Actions », créer le modèle d'animation « Model_Action » puis simuler.

- Il ne reste plus qu'à finaliser le MAC « Robot ». Il manque pour l'instant les différentes jonctions entre blocs, et les paramètres « Position_Capteur » des 7 capteurs de l'application. Pour ce faire, dans le groupe « divers », nous allons créer un MAC « Parametres » qui aura pour seul objectif de posséder des sorties initialisées avec les paramètres fixes de l'application (exemple : variable de sortie « Position_P2 » de type entier valant 200 décimètres ...). Insérer le MAC « Parametres » dans le MAC « Robot ».

Q14 : Simuler l'application complète, cela fonctionne t-il ?

Q15 : Sélectionner le MAC « Robot » et dans la colonne 3, insérer une vue synoptique « Syn_Robot ». Ouvrir « Syn_Robot ». Visualiser l'arbre, sélectionner « Actions » et dans la fenêtre synoptique (zone du bas)→Ajouter un modèle d'animation. Répéter la chose avec tous les capteurs. Aménager les positions, rajouter des éléments visuels fixes pour agrémenter le tout (un rail, ...). Tester la simulation, conclusion ?

- La création du robot doit être terminée, il ne vous reste plus qu'à tester son fonctionnement à l'aide d'une partie commande. Il vous suffit de récupérer la partie commande de l'application précédente et de l'associer avec votre partie opérative nouvellement créée.

Q16 : Créer une vue synoptique pour le MAC « PO ». Insérer dans ce synoptique « Syn_Robot ». Simuler le tout et conclure...