

Attention, mes TP sont longs, faites en le maximum, sans figoler les cas particuliers improbables.

Déroulement du TP : faire un rapport de TP pour le groupe (à rendre à la fin du TP ou dans une semaine maximum). Une fois résolue une question, me présenter votre programme (qui fonctionne), mettre par écrit votre programme, ainsi qu'une rapide présentation des essais que vous avez effectués, et vos éventuelles remarques.

Documents disponibles : documentation succincte STEP 7 (à rendre à la fin du TP), documentation en ligne de Step7.

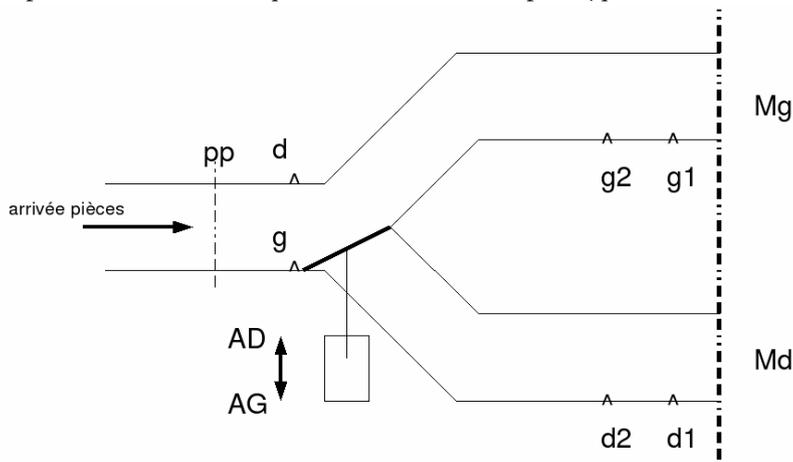
0 - Lecture (et compréhension) du petit document sur STEP7. Familiarisez vous avec le matériel et logiciel, préparez le projet en décrivant le matériel (exactement comme décrit, jusqu'à la valise en haut de la page 2, mais pas l'AS400 ni les mnémoniques). Les connexions IP et Profibus doivent être bien définies (tout ce qui est écrit dans le document est utile), les adresses doivent correspondre à l'exemple donné. Puis enregistrez votre projet sous un nouveau nom (l'ancien nom servira de base aux prochaines questions et aux prochains TP).

1 – Combinatoire simple : une sortie

Ecrire un programme n'utilisant que des fonctions **combinatoires**, utilisant trois entrées tout ou rien, que nous numérotions 1, 2 et 3, et une sortie qui s'allumera quand la somme des numéros d'entrée est paire (vous utiliserez un tableau de Karnaugh). Par exemple, si on appuie sur le 1 et le 3, 3+1 étant pair il faut allumer la sortie. Vous utiliserez obligatoirement le langage à contacts (CONT). Je propose de considérer que 0 est un nombre pair (quand aucun n'est appuyé).

2 - Combinatoire

Un tapis (que l'on ne commande pas ici) amène des pièces vers deux machines Md et Mg. Un capteur pp détecte l'arrivée d'une nouvelle pièce, un aiguillage les dirigera vers la voie sélectionnée. Le temps de passage de la pièce devant pp est suffisant pour ne déplacer l'aiguillage que pendant que pp=1, et les pièces arrivent assez espacées pour chaque pièce soit positionnée devant sa machine avant que la suivante ne soit détectée par pp. Chaque machine prend une pièce quand elle en a besoin (une à la fois). On veut réguler le flux et donc mettre autant de pièces en attente de chaque côté. En cas d'égalité,



on dirige la pièce vers le côté opposé à celui choisi pour la pièce précédente (il suffit de regarder la position de l'aiguillage d ou g pour le savoir). Dans un premier temps, considérons ne compter au maximum que deux pièces dans chaque voie (ce qui n'empêche pas qu'il y en ait plus, alors on envoie de chaque côté alternativement). Le but final est de trouver les équations des sorties (AD et AG) en fonction des entrées (en combinatoire). Pour cela, vu le nombre de capteurs en jeu, trouvez **d'abord** s'il y en a plus en attente en G (PG), en D (PD) ou égalité (E), en fonction de g1, g2, d1, d2. Je vous conseille de prévoir tous les cas possibles, pour rechercher les trois équations à l'aide d'un tableau de Karnaugh (ou au moins deux des trois). Vous utiliserez trois mémoires de l'automate (M0.0 à M0.2). Les équations de AD et AG seront **ensuite** bien plus faciles à trouver (même sans tableau de Karnaugh), en fonction de PG, PD, E, pp, d et g. Programmez le en langage à contacts (un réseau par sortie ou mémo). Vous utiliserez obligatoirement des mnémoniques. Montrez moi votre simulation.

3 – Langage LOG

Affichez désormais le programme précédent en langage LOG (portes logiques). Dites moi dans votre rapport quel langage vous préférez.

4- langage LIST

Affichez le programme en LIST et surtout, essayez de le comprendre ! Nous réutiliserons ce langage par la suite.

5- séquentiel (bascules)

Testez les **bascules** (RS), obligatoirement en langage LOG : On désire automatiser la porte d'un garage collectif. On utilise deux sorties : Montée (M) et Descente (D) de la porte, et les capteurs suivants : 3 boutons poussoirs Ouvrir (o), Fermer (f), Stop (s), et deux capteurs de position pour la porte : haut (h ouverte) et bas (b fermée).

Quand on demande la montée (et que la porte n'est pas ouverte), on ouvre la porte, jusqu'à ce qu'elle soit en haut. Quand on demande la fermeture, on ferme la porte. La fermeture en cours de la porte est transformée en montée par un ordre de montée. L'ordre Stop arrête la porte dans sa position actuelle, et ce jusqu'à un ordre de montée ou de descente.