Formation Automatique et Informatique Industrielle

Master 1 S2

Matière : Systèmes Embarqués et Systèmes Temps Réel SE-STR

Par : ATOUI Hamza

Plan de TP

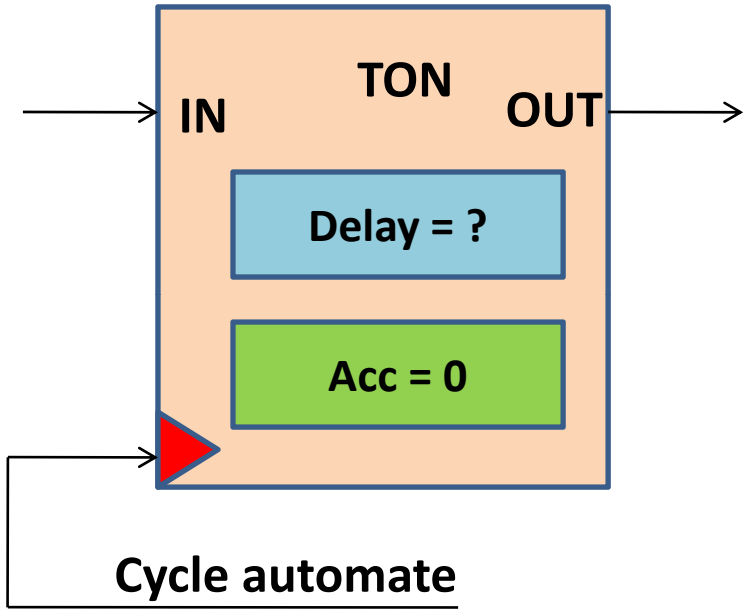
- LAB préparatoire (réalisation d'un TIMER de type TON).
- Astuce très utile.
  - Comment augmenter le nombre E/S d'un µC ?
  - L'histoire de l'horloge cycle automate.
- LAB1 (GRAFCET séquence linéaire).
- LAB2 (GRAFCET séquences synchronisées).
- LAB3 (GRAFCET sélection de séquence).

LAB préparatoire

- Dans le monde de l'automatisme, on trouve la notion d'activer une action durant un certain temps de retarder…
- Un automaticien me dit « c'est simple, ce sont les TIMERs !».
- Les deux célèbres TIMERs sont :
  - Timer On (TON) : permet de gérer des retards à l'enclenchement.
  - Timer Off (TOF) : permet de gérer des retards au déclenchement.
- Dans ce LAB on va focaliser sur le type TON et on laisse le TOF pour les meilleurs.

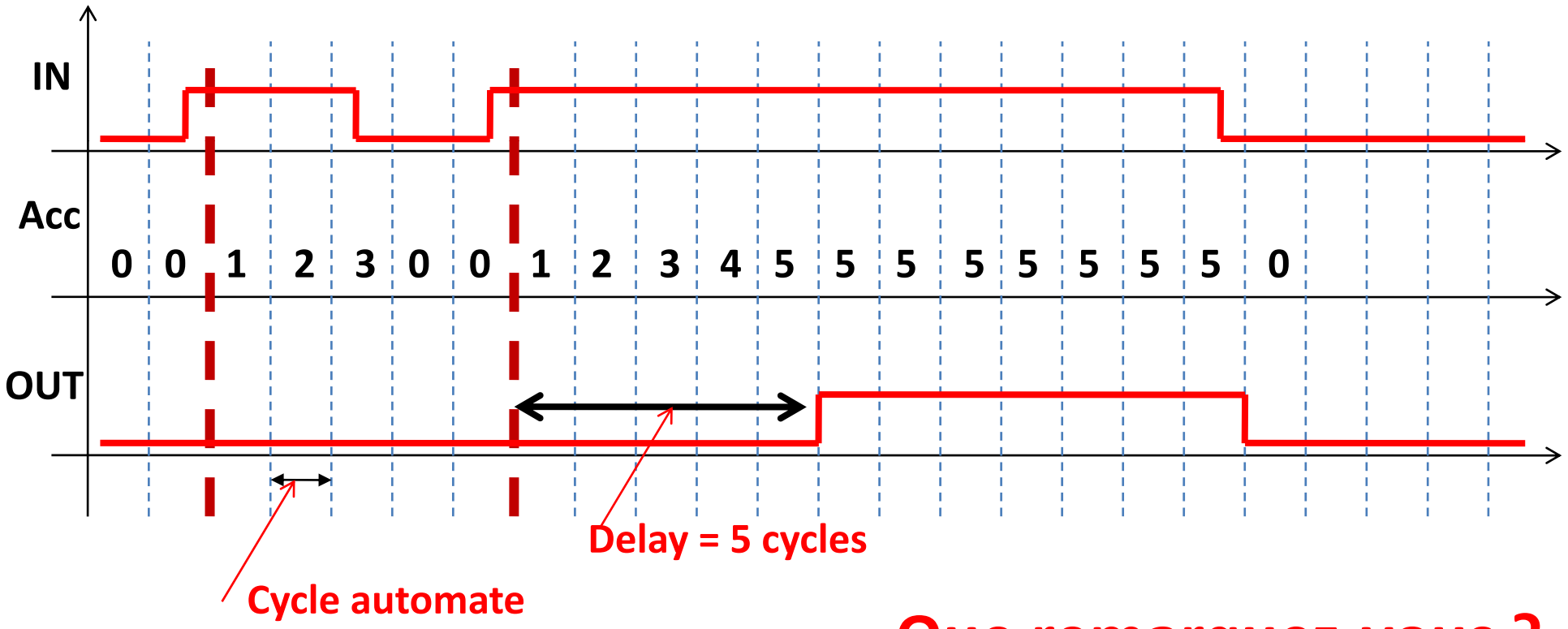
LAB préparatoire

- Le symbole de TIMER-TON:
- IN: entrée d'activation (démarrage)
- OUT: sortie du TIMER qui indique que le « Delay » est épuisé.
- Delay: le délai (retard) à réaliser.
- Acc: Accumulateur indique le temps épuisé en cours.
- Entrée d'horloge: la plus part des temps est alimenté par le cycle automate.
- Au moment de l'exploitation, on configure seulement le paramètre Delay (***dans notre cas***).



LAB préparatoire

- Pour apprendre le fonctionnement de TIMER-TON, on propose les chronogrammes suivants (Delay=5):



Que remarquez-vous ?

LAB préparatoire

- Faire un algorithme ou un organigramme présente l'émulation du comportement de TIMER-TON.

- Utiliser le µC PIC ou la plateforme ARDUINO pour programmer deux TIMER-TON (pour le µC PIC on préfère d'utiliser le C-Langage HI-TECH C PRO).

Astuce très utile

- Le PIC16F84A communique avec le monde externe par 13 E/S. Imaginez un processus industriel manipule 16 E/S ou plus; **comment faire pour contrôler et commander (automatiser) ?**

- La plus part des étudiants me disent qu'il faut changer le µC par un autre possède plus d'E/S. Absolument, c'est juste! ***Mais, on insiste pour résoudre le problème par le PIC16F84A!!!***

Astuce très utile

- Donc, la question qu'il faut poser : **comment augmenter le nombre E/S d'un µC ?**

- L'idée de la solution vienne de circuit mémoire, un bus de données de N bits attaque M éléments mémoire au même temps, mais un seul élément communique à la fois à travers le bus par le biais d'une adresse unique pour chaque élément, dont cette adresse attaque un décodeur interne dans le circuit mémoire et génère un signal d'activation spécifique et unique pour cet élément.

Astuce très utile

- Donc, pour chaque élément mémoire, on a besoin d'un signal d'activation unique avec le partage de deux signaux de contrôle RD (demande de lecture) et WR (demande d'écriture).

- N.B: un élément mémoire est un registre de N bits avec un registre est un ensemble de bascules.

Astuce très utile

- Apparemment, la solution se base sur l'utilisation des registres. Le composant électronique registre présente le nom général d'une famille de types:
  - Registre entrée parallèle sortie parallèle (**PIPO**).
  - Registre entrée parallèle sortie série (**PISO**).
  - Registre entrée série sortie parallèle (**SIPO**).
  - Registre entrée série sortie série (**SISO**).
- **Quels types parmi qu'on peut utiliser pour résoudre notre problème ?**

Astuce très utile

- Attention: l'augmentation des E/S se fait sur les entrées toutes seules et les sorties toutes seules, ce n'est pas comme les E/S d'un µC qu'on peut configurer la même patte soit en entrée ou bien en sortie.

- **Au moment d'augmenter le nombre des entrées, quels sont les types à utiliser? Et la même question pour les sorties**.

Astuce très utile

- Pour les entrées : on peut utiliser le PIPO et le PISO.
- Pour les sortie : on peut utiliser le PIPO et le SIPO.

- Donc, le type SISO est inutile dans notre cas !

- **Comment ça???**

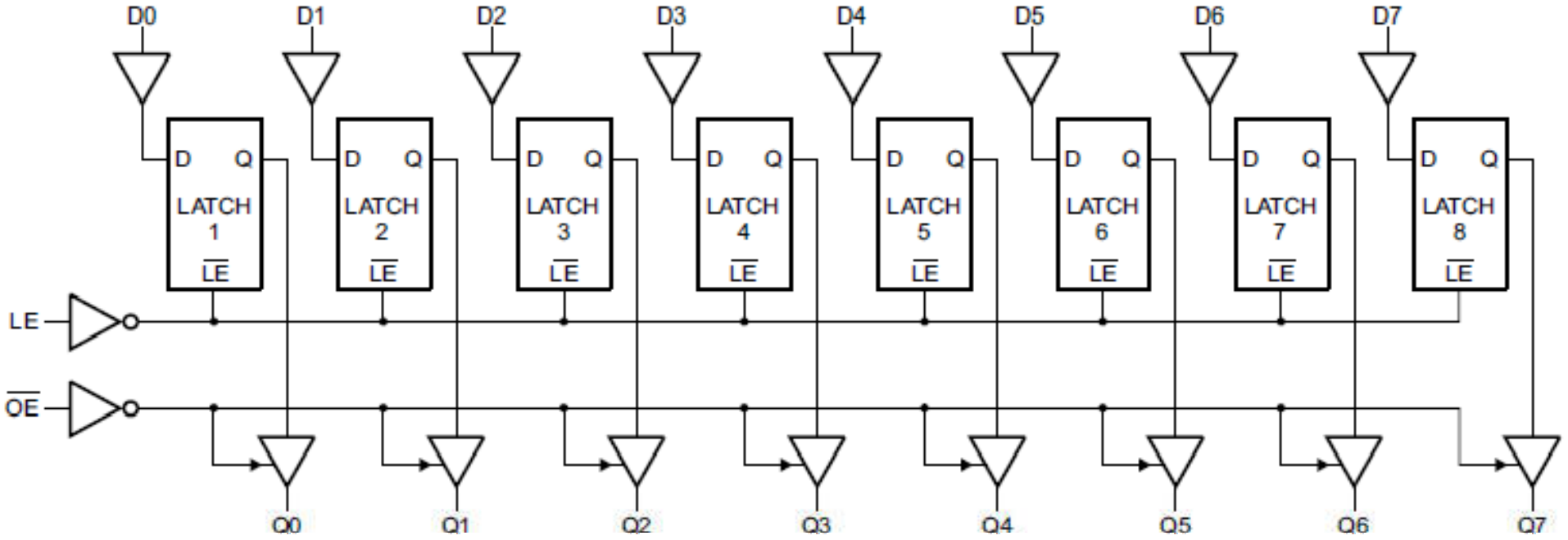
Astuce très utile

- Avant de repondre à cette question, on va faire une recherche sur les circuits intégrés qui réalisent les types PIPO, SIPO et PISO.

- Le 74HC573 est un PIPO.
- Le 74HC165 est un PISO/SISO.
- Le 74HC595 est un SIPO.

Astuce très utile

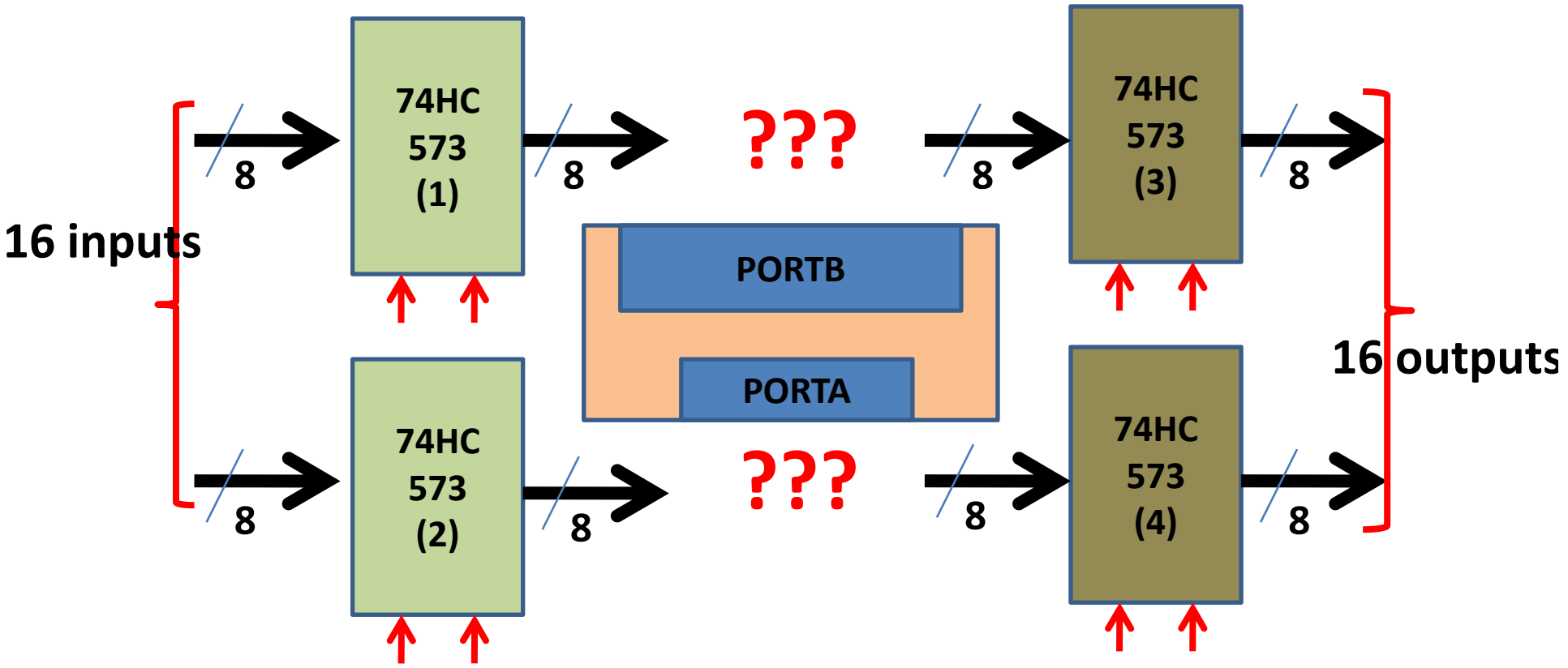
- La logique interne de 74HC573 :



Que remarquez-vous ?

Astuce très utile

- On demande de faire le schéma d'un système à base de PIC16F84A pour gérer 16 entrées et 16 sorties par le biais de 4 circuits 74HC573 comme indique la synoptique suivante:
- Utiliser le PORTB pour créer le bus et le PORTA pour commander les signaux de contrôle des circuits 74HC573



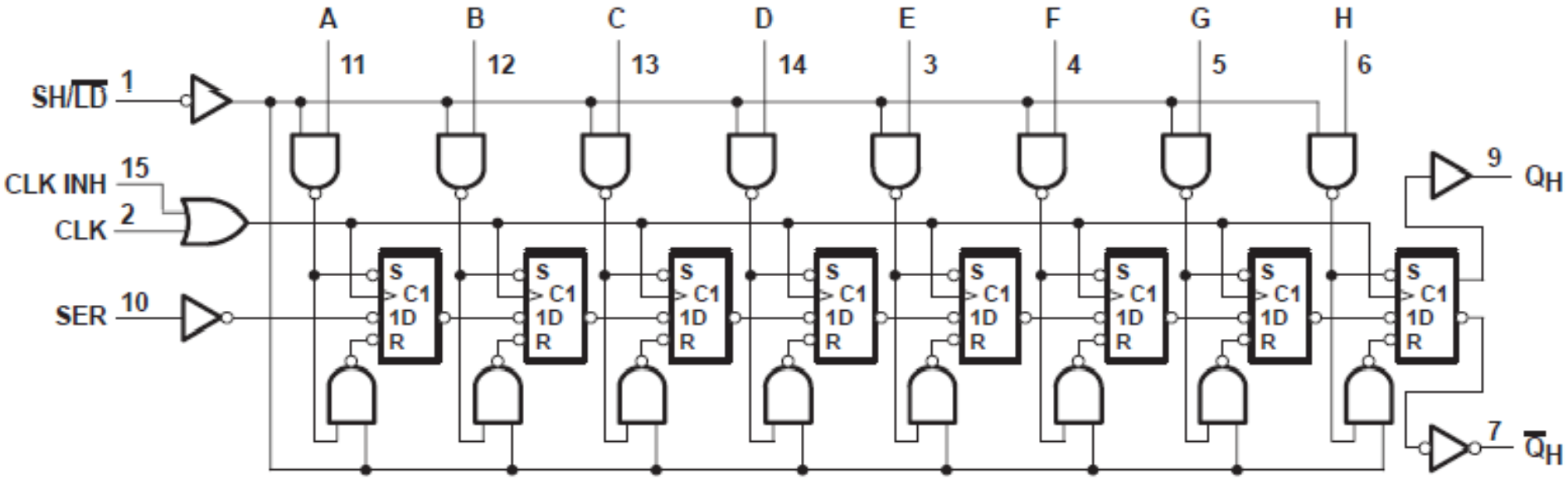
Astuce très utile

- Faire un algorithme ou un organigramme qui réalise le travail suivant:
  - Lire les 8 entrées de circuit (1) est recopier dans les 8 sorties de circuit (3).
  - Lire les 8 entrées de circuit (2) est recopier leur état inverse dans les 8 sorties de circuit (4).

- Faire le programme soit en assembleur ou en C-Langage HI-TECH C PRO.

Astuce très utile

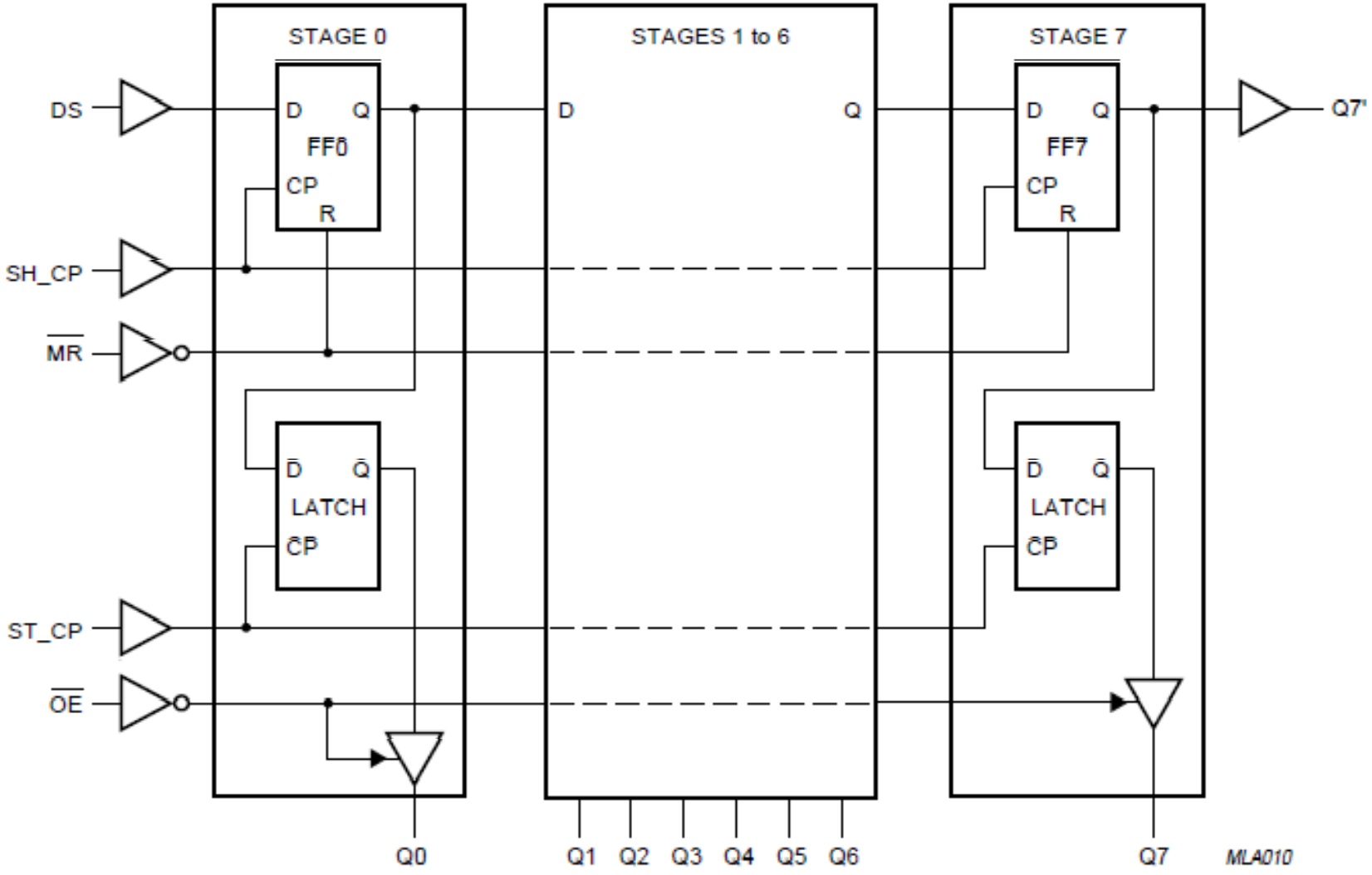
- La logique interne de 74HC165 :



Que remarquez-vous ?

Astuce très utile

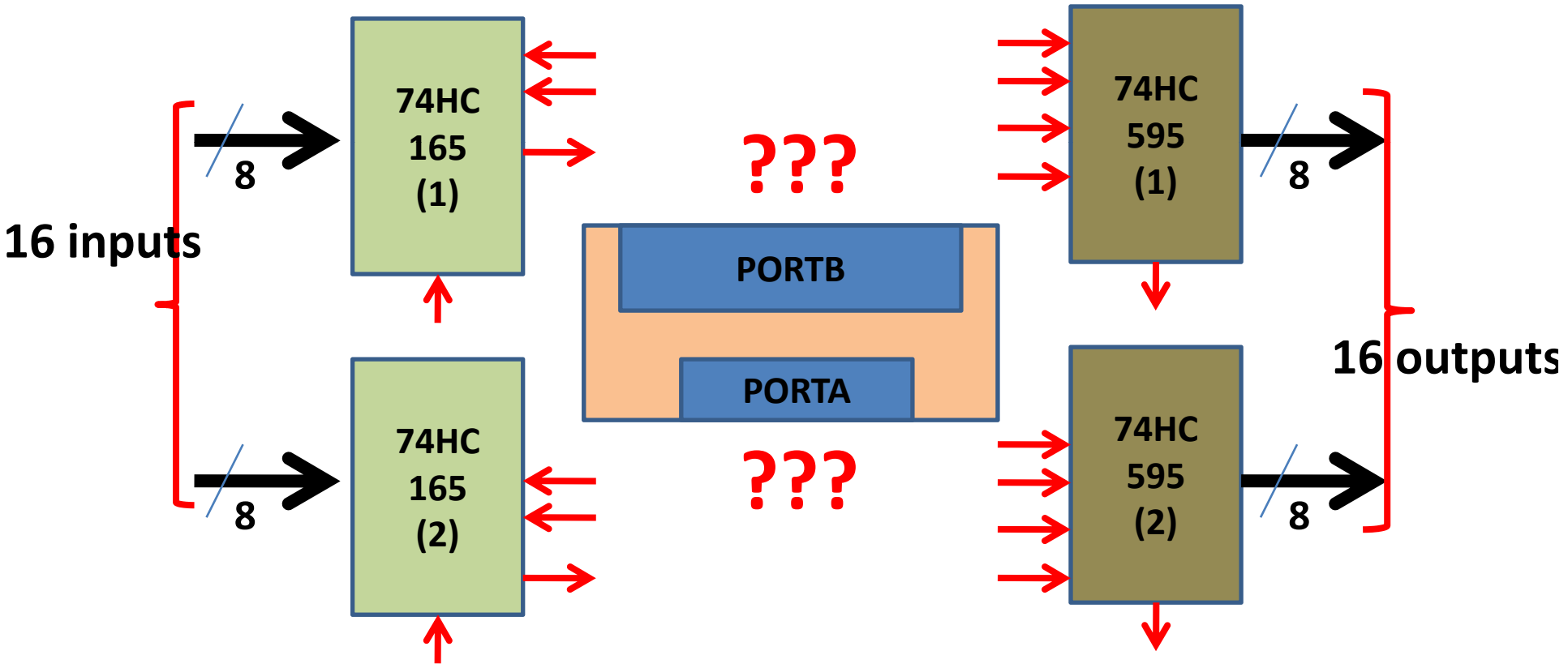
- La logique interne de 74HC595 :



Que remarquez-vous ?

Astuce très utile

- On demande de faire le schéma d'un système à base de PIC16F84A pour gérer 16 entrées et 16 sorties par le biais de 2 circuits 74HC165 et 2 circuits 74HC595 comme indique la synoptique suivante:



Astuce très utile

- Faire un algorithme ou un organigramme qui réalise le travail suivant:
  - Lire les 8 entrées de circuit 74HC165 (1) est recopier dans les 8 sorties de circuit 74HC595 (1).
  - Lire les 8 entrées de circuit 74HC165 (2) est recopier leur état inverse dans les 8 sorties de 74HC595 (2).

- Faire le programme soit en assembleur ou en C-Langage HI-TECH C PRO.

- Comparer les deux solutions.

Astuce très utile

- Au moment de la réalisation d'un GRAFCET/LADDER par un schéma électronique, les étapes du GRAFCET/LADDER sont des cellules mémoires de type RS.

- ***La bascule RS est un circuit séquentiel asynchrone sensible à la moindre impulsion sur ses entrées S et R. un parasite (événement fugitif) sur une des entrées S/R fait une mise à jour d'une étape, donc une fausse évolution !!!***
- ***Comment luter contre les parasites générés par l'environnement/capteur sur une entrée ?***

Astuce très utile

- La plus simple solution, c'est d'utiliser des bascules synchrones cadencées par une horloge (CLK) au lieu d'utiliser des bascule asynchrones par ce que la bascule synchrone évolue chaque niveau bas ou haut de signal d'horloge ou bien sur ses francs.

- La période de l'horloge donne une idée sur la largueur de l'impulsion parasite qu'on va filtrer. Si la période est de 20ms, donc les parasites qu'on va filtrer sont de l'ordre de ¼ de la période = 5ms au maximum.

Astuce très utile

- On appelle la période de signal d'horloge dans l'automatisme par la durée de cycle automate.

- Dont, ce cycle fait 3 phases : lecture des entrées, traitement et mise à jour des sorties (sur une plateforme programmable API, µC, µP…).

Astuce très utile

- ***Question pour champion***: si notre PIC utilise un Quartz de 4MHz et le cycle automate est de l'ordre de 10ms (100Hz).

- **Dans quel cas le PIC travaille en temps réel ?**

Astuce très utile

- Faire un algorithme ou un organigramme assure le cycle automate chaque 10ms par un PIC cadencé par un Quartz de 4MHz avec la temporisation de cycle se fait par le biais de TIMER0.
- Détail:
  - Configurer le TIMER0 pour générer un interruption chaque 1ms.
  - Le traitement de l'interruption de TIMER0 décrémente un compteur commence par 10, si ce dernier tombe à zéro une case mémoire du nom « flag_cycle » positionne à 1 avec le chargement du compteur par 10.
  - Le programme principal boucle infiniment sur l'attente de positionnement de « flag_cycle », si le cas, effacer la case mémoire « flag_cycle » et lancer le cycle automate.

Astuce très utile

- La ligne LADDER qu'on va réaliser par le cycle automate :

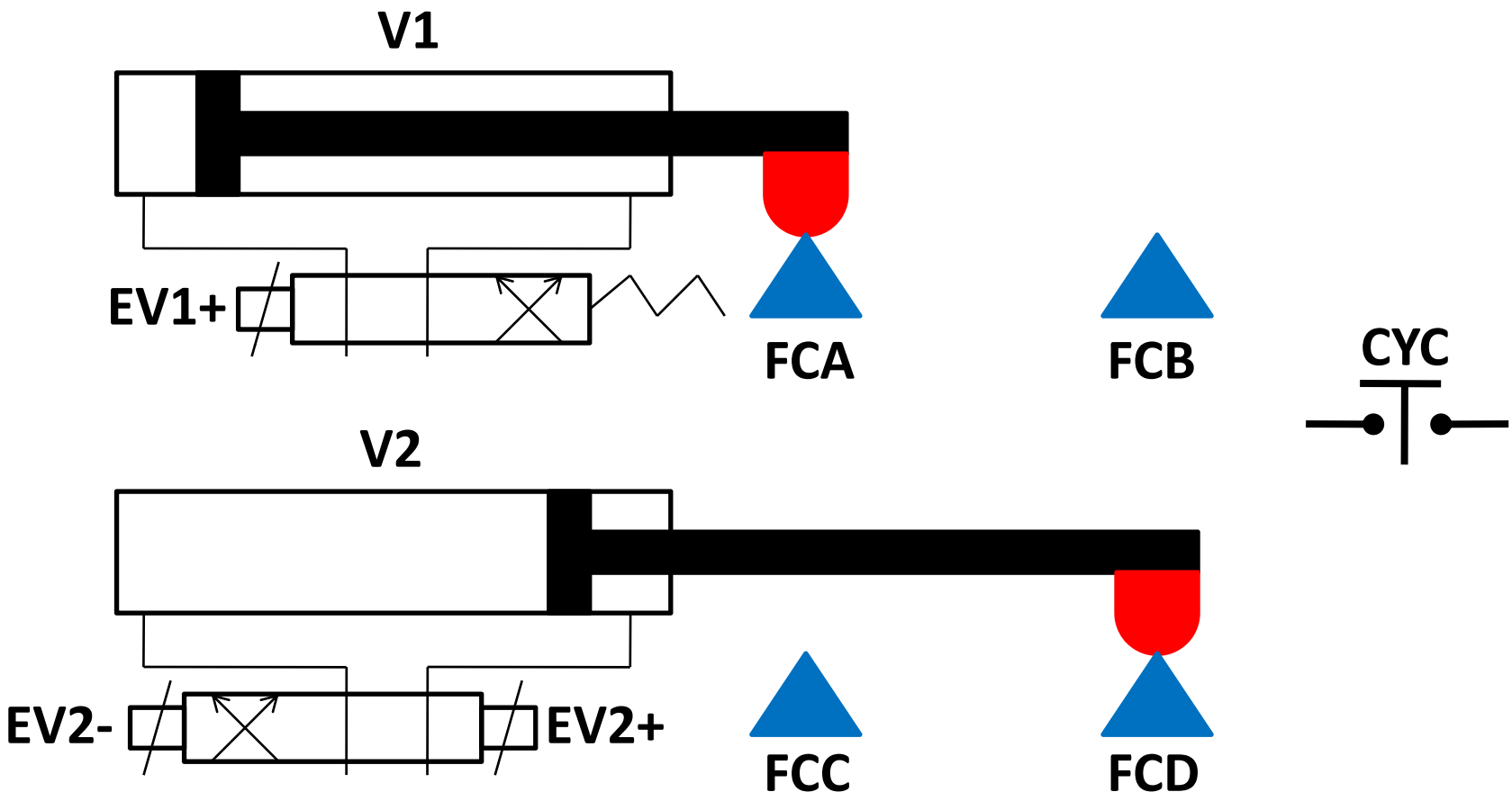
- Faire le programme en assembleur ou en C-Langage HI-TECH C PRO.

LAB1 (GRAFCET séquence linéaire)

Étude d'un cycle carré de deux vérins

- **Description de processus**: le système possède deux vérins V1 alimenté par un distributeur pneumatique 2/4 monostable et un autre vérin alimenté aussi par un distributeur pneumatique 2/4 bistable avec 4 fin de course et un bouton poussoir pour démarrer le cycle comme indique la figure suivante :

LAB1 (GRAFCET séquence linéaire)



LAB1 (GRAFCET séquence linéaire)

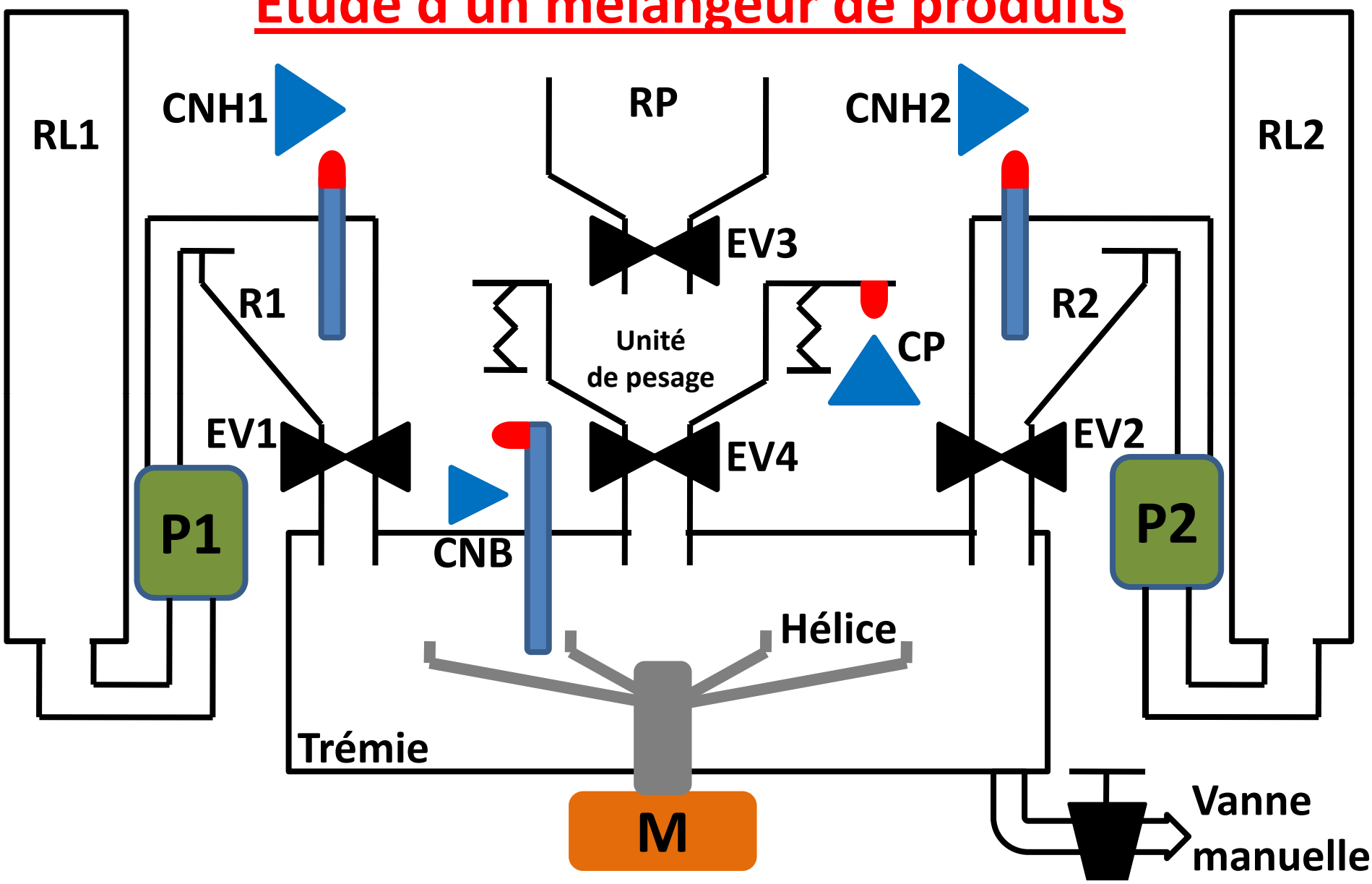
- **Mode de fonctionnement (cahier des charges)** :
- **État de repos** : le vérin V1 à la position FCA et le vérin V2 à la position FCD comme indique la figure de diapo précédente.
- **Cycle automate** : au moment d'appui sur le bouton CYC, le vérin V1 sort vers la position FCB, si ce dernier est actif le vérin V2 rentre vers la position FCC; le positionnement de FCC provoque la rentrer de vérin V1 vers la position FCA puis le cycle recommence par l'appui sur le bouton CYC.

LAB1 (GRAFCET séquence linéaire)

- Faire une étude préliminaire présente le remplissage d'un tableau à deux colonnes (les entrées et les sorties de processus).
- Faire le GRAFCET de processus.
- Faire la conversion de GRAFCET en LADDER passant par le remplissage de la table de condition d'activation et désactivation de chaque étape avec les différentes équations des sorties.
- Utiliser le µC PIC16F84A pour réaliser l'automatisme de processus par:
  - La connexion entre les pattes de µC avec les E/S de processus.
  - La conversion de GRAFCET en C-Langage.
  - La conversion de LADDER en C-Langage.
- N.B : Le cycle automate est de 10ms assuré par le module TIMER0.

LAB2 (GRAFCET séquences synchronisées)

Étude d'un mélangeur de produits



LAB2 (GRAFCET séquences synchronisées)

- **Description de processus**: le système possède trois réservoirs principaux **RL1** de liquide 1, **RL2** de liquide 2 et **RP** de poudre. Une **unité de pesage** pour la poudre. Une **trémie** et une **hélice** pour mélanger les trois produits. Deux petits réservoirs **R1** et **R2** pour stocker le volume des liquides 1 et 2 à mélanger. Une **vanne manuelle** pour évacuer le produit final.
- Un moteur **M** pour tourner l'hélice. Deux pompes **P1** et **P2** pour remplir R1 et R2. Deux électrovannes **EV1** et **EV2** pour évacuer R1 et R2 dans la trémie. Une électrovanne **EV3** pour évacuer la poudre dans l'unité de pesage et une autre **EV4** pour évacuer la poudre pesée dans la trémie. Un capteur de pesage **CP** indique le poids désiré. Deux flotteurs avec fin de course niveau haut **CNH1** et **CNH2** indiquent le remplissage de R1 et R2. un flotteur avec fin de course niveau bas **CNB** indique le niveau bas de produit final dans la trémie.

LAB2 (GRAFCET séquences synchronisées)

- **Mode de fonctionnement (cahier des charges)** :
- **État de repos** : les deux réservoirs R1 et R2 sont vide, l'unité de pesage est vide avec M, P1 et P2 sont à l'arrêt. Toutes les électrovannes sont fermé (EV1, EV2, EV3 et EV4).
- **Cycle automate** : l'ouverture de la vanne manuelle provoque la diminution de produit final dans la trémie jusqu' au son niveau bas qui sera détecté par la fin de course CNB du flotteur. Si le capteur CNB est actif, l'automate lance plusieurs opérations simultanément:
  - Remplissage de R1 er R2 jusqu'au niveau haut CNH1 et CNH2 par le biais des pompes P1 et P2.
  - Pesage de poudre par l'ouverture de EV3 jusqu'à la détection du poids désiré par CP (unité de pesage).

LAB2 (GRAFCET séquences synchronisées)

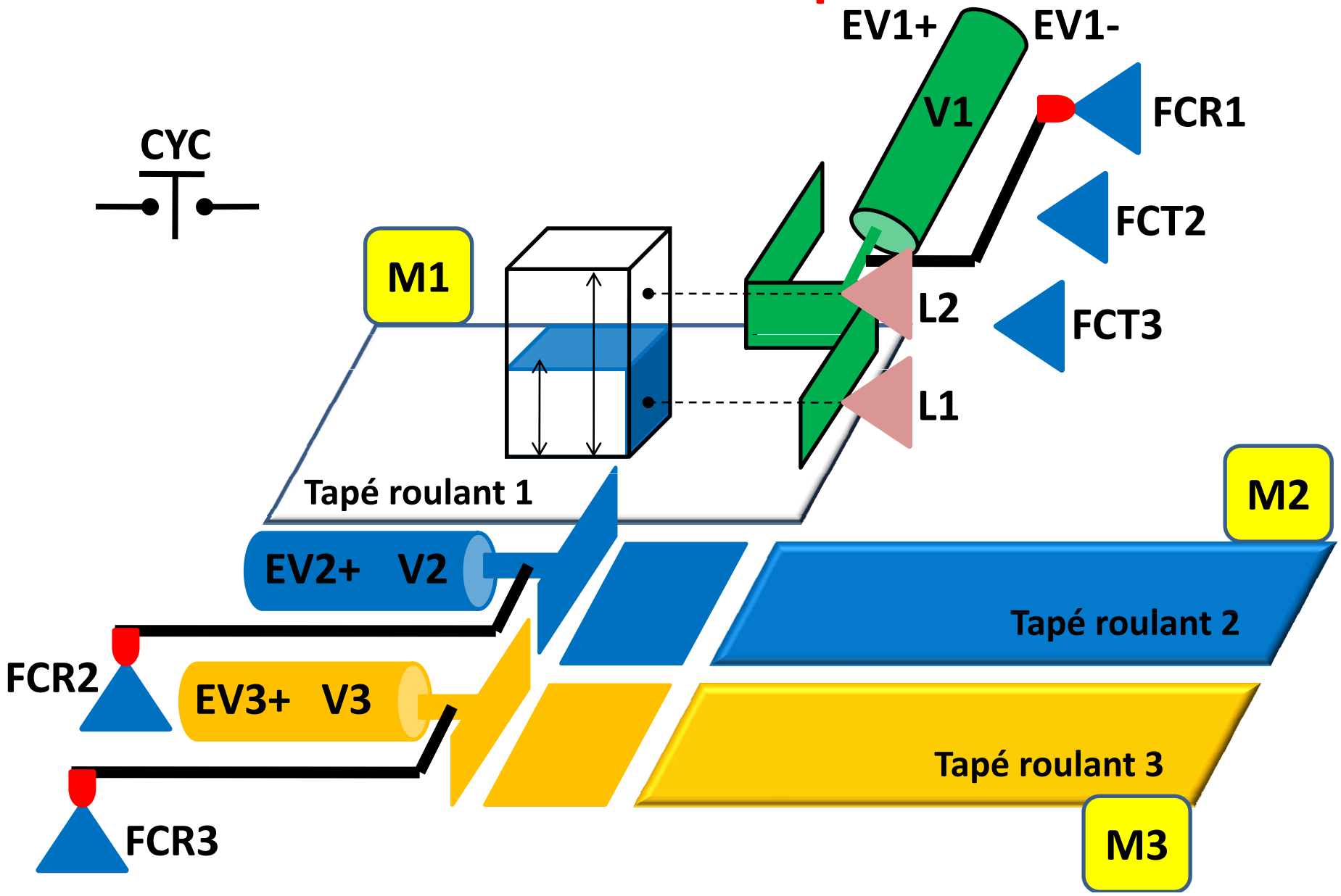
- Si le volume et le poids désirés sont obtenu, l'automate donne l'ordre au moteur M pour tourner avec EV1, EV2 et EV4 pour évacuer les produits dans la trémie, tous durant 30s.
- L'automate lance un autre cycle si le produit final atteint son niveau bas dans la trémie.

LAB2 (GRAFCET séquences synchronisées)

- Faire une étude préliminaire présente le remplissage d'un tableau à deux colonnes (les entrées et les sorties de processus avec Timers).
- Faire le GRAFCET de processus.
- Faire la conversion de GRAFCET en LADDER passant par le remplissage de la table de condition d'activation et désactivation de chaque étape avec les différentes équations des sorties.
- Utiliser le µC PIC16F84A pour réaliser l'automatisme de processus par:
  - La connexion entre les pattes de µC avec les E/S de processus.
  - La conversion de GRAFCET en C-Langage.
  - La conversion de LADDER en C-Langage.
- N.B : Le cycle automate est de 10ms assuré par le module TIMER0.

LAB3 (GRAFCET sélection de séquence)

Processus de classification des pièces selon leur taille



LAB3 (GRAFCET sélection de séquence)

- **Description de processus**: le système possède trois tapés roulants, un vérin poussoir double effet **V1** alimenté par un distributeur pneumatique 3/5 bistable (la possibilité d'immobiliser la tige à n'importe quelle position au moment de désactiver les bobines) et deux vérins poussoirs simple effet **V2/V3** chacun alimenté par un distributeur 2/3 monostable (au moment d'activer la bobine de distributeur, la tige sort vers l'extérieur et au moment de désactiver la tige rentre automatiquement vers l'intérieur sous l'effet du ressort de rappel).

LAB3 (GRAFCET sélection de séquence)

- Trois moteurs **M1**, **M2** et **M3** pour assurer le mouvement des tapés roulants, Trois fin de course **FCR1**, **FCT2** et **FCT3** indiquent la position de la tige de vérin V1, une fin de course **FCR2** indique la position de repos de vérin V2, une fin de course **FCR3** indique la position de repos de vérin V3, deux capteurs de présence **L1** et **L2** indiquent la présence et la taille de la pièce à classer devant le vérin V1 et un bouton poussoir **CYC** pour démarrer le cycle.

LAB3 (GRAFCET sélection de séquence)

- **Mode de fonctionnement (cahier des charges)** :
- **État de repos** : aucune pièce devant le vérin V1, le vérin V1 est à la position FCR1, le vérin V2 est à la position de repos FCR2, le vérin V3 est aussi à la position de repos FCR3 avec les moteurs (M1, M2 et M3) sont à l'arrêt.
- **Cycle automate** : au moment d'appui sur le bouton CYC, l'automate donne l'ordre au tapé roulant 1 à fonctionner par le biais de moteur M1. si une pièce arrive devant le vérin V1 le moteur M1 s'arrête et l'automate fait la mesure de la taille de la pièce :

LAB3 (GRAFCET sélection de séquence)

- Si la pièce est de petite taille (capteur L1 actif et capteur L2 inactif), l'automate donne l'ordre au vérin V1 pour sortir à la position FCT2 puis revenir à sa position de repos FCR1. si FCR1 est actif, l'automate donne l'ordre au vérin V2 pour pousser la pièce vers le tapé roulant 2 durant 2s puis le tapé roulant 2 déclenche par le biais de moteur M2 pour évacuer la pièce vers le lot des pièces de petite taille durant 10s.

LAB3 (GRAFCET sélection de séquence)

- Si la pièce est de grande taille (capteur L1 actif et capteur L2 actif), l'automate donne l'ordre au vérin V1 pour sortir à la position FCT3 puis revenir à sa position de repos FCR1. si FCR1 est actif, l'automate donne l'ordre au vérin V3 pour pousser la pièce vers le tapé roulant 3 avec le déclenchement de tapé roulant 3 durant 2s puis le tapé roulant 3 reste en mouvement pour évacuer la pièce vers le lot des pièces de grande taille durant 15s.
- Le cycle automate recommence si le bouton CYC est appui encore une autre fois.

LAB3 (GRAFCET sélection de séquence)

- Faire une étude préliminaire présente le remplissage d'un tableau à deux colonnes (les entrées et les sorties de processus avec Timers).
- Faire le GRAFCET de processus.
- Faire la conversion de GRAFCET en LADDER passant par le remplissage de la table de condition d'activation et désactivation de chaque étape avec les différentes équations des sorties.
- Est-ce que le PIC16F84A capable d'automatiser ce processus ? Si non, comment faire ?
- Utiliser le µC PIC16F84A pour réaliser l'automatisme de processus par:
  - La connexion entre le système à µC avec les E/S de processus.
  - La conversion de GRAFCET en C-Langage.
  - La conversion de LADDER en C-Langage.
- N.B : Le cycle automate est de 10ms assuré par le module TIMER0.