

### Présentation

Cet article fait suite à un premier qui proposait une introduction progressive et pragmatique aux systèmes asservis. Le TP se déroulait sur l'axe linéaire [Control'X](#) de la société DMS éducation.

Ce nouvel article présente deux TP qui s'appuient eux aussi sur le système Control'X. A eux deux, ces TP permettent de parcourir la totalité d'un cycle de conception de type cycle en V. Le premier TP permet d'obtenir un modèle validé de boucle ouverte et le second exploite ce modèle pour effectuer la synthèse d'un correcteur.

Ces TP s'adressent à des élèves CPGE plutôt de première année pour le 1<sup>er</sup> TP et plutôt de deuxième année pour le second TP. Le public peut être élargi, sans restriction aucune, aux classes de BUT ou BTS.

Les buts pédagogiques sont multiples : la première finalité consiste à effectuer la synthèse d'un correcteur.

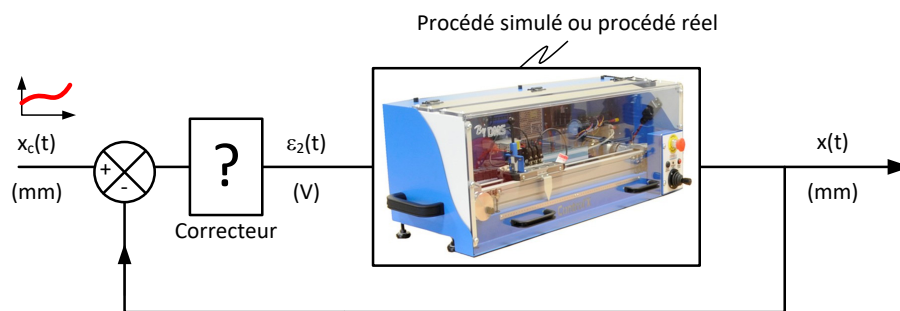


Figure 1 : but : caler le correcteur

Mais ce premier point n'est en fait qu'un alibi. Ce que l'on cherche, c'est aussi et surtout à replacer cette activité dans un cycle de conception tel qu'il pourrait être suivi par un ingénieur en activité. Le but est de mettre en œuvre de façon concrète deux outils : le cycle en V et le triptyque "souhaité-simulé-réel".

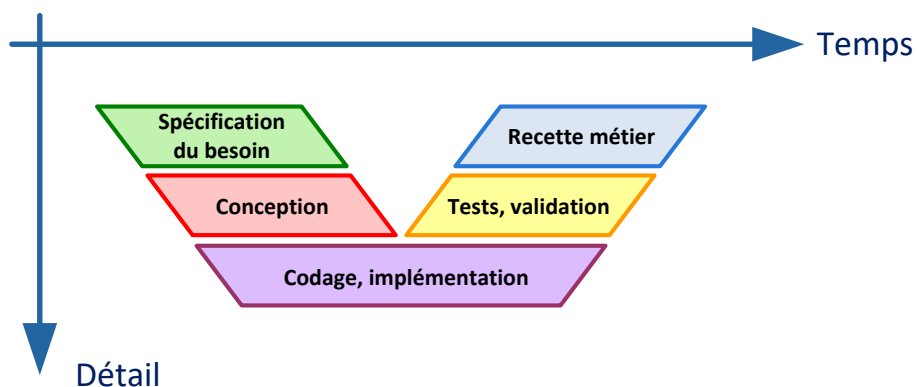


Figure 2 : cycle en V simplifié à l'extrême : comment lui donner un sens pratique ?

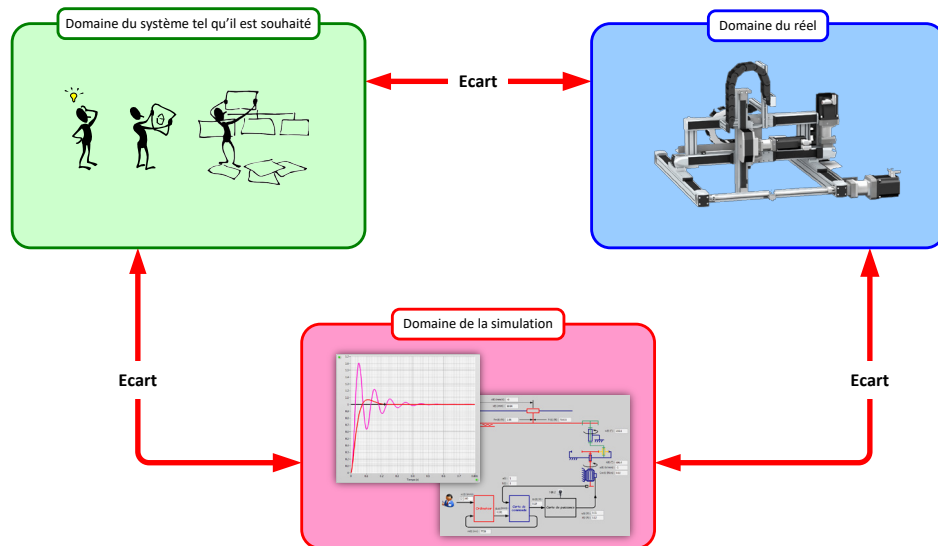


Figure 3 : triptyque "souhaité-simulé-réel": comment le parcourir ?

Les élèves ont bien compris que sur nos systèmes du laboratoire, on s'intéresse fréquemment à au moins l'un des trois écarts simulé-réel, simulé-souhaité ou souhaité-réel. Mais dès qu'on pousse un peu plus loin les élèves dans leurs réflexions en leur posant des questions du type :

- Pourquoi s'intéresser à tel ou tel écart ?
- Qui est amené à analyser tel écart dans un cycle de conception ?
- A quel instant est-on amené à analyser tel écart ?
- Quel est l'ordre chronologique d'analyse de ces écarts ?

... et bien les réponses sont beaucoup plus évasives et c'est bien normal !

Dans les deux TP proposés, on cherche à suivre une démarche typée "Model-Based Design". Derrière ces termes qui peuvent paraître abstraits, on va découvrir qu'il ne s'agit ni plus ni moins finalement que d'une démarche de conception qui suit le parcours du cycle en V en misant le plus possible et le plus longtemps possible sur le modèle du système à concevoir.

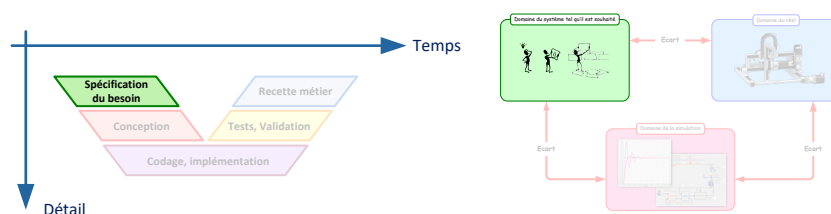
Suivre le cycle en V permet de donner du sens à nos apprentissages. A chaque étape, on fait les choses parce qu'on a besoin de les faire, parce qu'on s'est fixé un but : concevoir un produit industriel par exemple. Analyser un écart n'est pas une fin en soi.

Ici le support de TP, Control'X n'est qu'un alibi. Le but que l'on s'est fixé n'est d'ailleurs pas la conception d'un axe linéaire dans son ensemble mais juste la conception d'un correcteur permettant au final de satisfaire un cahier des charges.

## Résumé des activités clés des TP

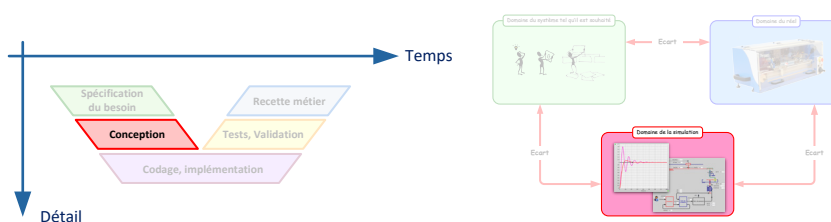
On détaille ci-après les grandes étapes de cheminement suivi dans ces deux travaux pratiques.

### 1° Etape : prise de connaissance du cahier des charges



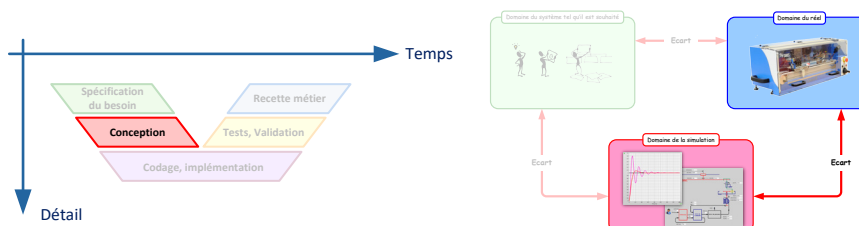
Cette étape est le point de départ sur le cycle en V mais comme on va le voir, cette donnée est mise en attente dans un premier temps.

### 2° Etape : modélisation



Les élèves sont ici amenés à mettre en place un modèle de connaissance.

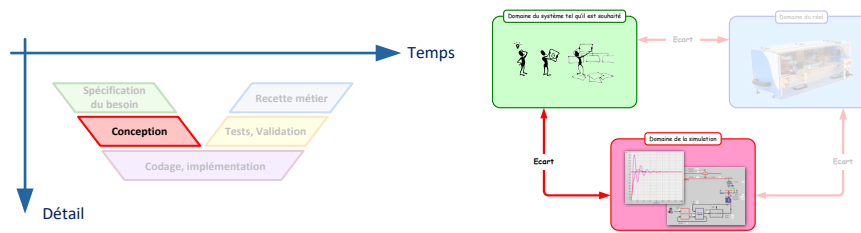
### 3° Etape : validation du modèle



Dans cette étape, le modèle de connaissance doit être validé : évidemment, il faut ici pouvoir disposer à ce stade du système réel ou au moins d'un prototype de ce système.

C'est dans cette 3° étape qu'un écart est analysé pour la première fois (écart simulé-réel). Cet écart est analysé par le scientifique qui souhaite simuler le comportement du système réel (simuler dans le sens "faire semblant"). Mais comme on va le voir, ce modèle n'est pas une finalité, c'est un moyen. On va pouvoir se rendre compte ici de la nécessité d'introduire des non-linéarités pour pouvoir garantir une certaine robustesse à notre modèle.

#### 4° Etape : conception de l'algorithme de correction



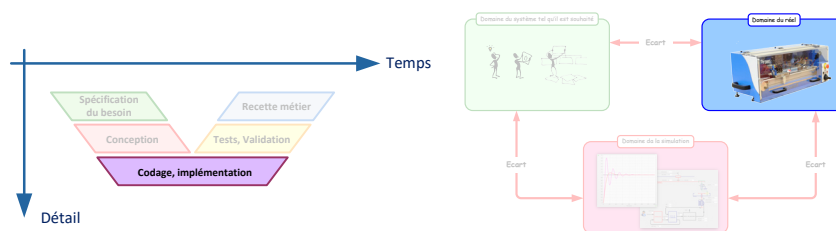
C'est dans cette 4° étape qu'un nouvel écart est analysé, l'écart souhaité-simulé : le correcteur à synthétiser doit permettre au système simulé de valider le cahier des charges.

Le défi pour le professeur consiste ici à expliquer que la synthèse de correcteur s'effectue sur le modèle linéaire alors même qu'on le sait parfois (souvent) incapable de refléter le comportement réel (en cas de forte saturation notamment) ... mais que cela reste le meilleur moyen d'obtenir un correcteur performant.

Le logiciel de pilotage de Control'X, dispose de certaines fonctions spécifiquement développées pour justifier ou expliquer ce point.

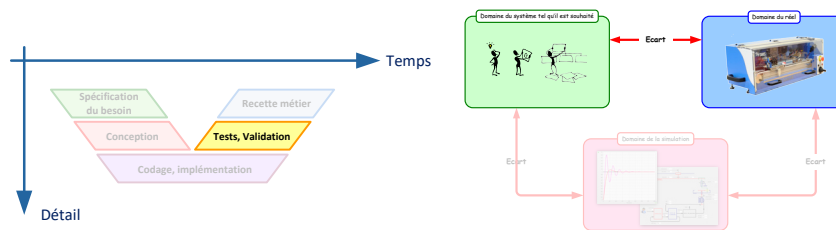
Dans cette étape le système virtuel (le modèle validé) permet de se livrer à toute une batterie de tests pour garantir qu'en toutes circonstances le système simulé se comportera convenablement.

#### 5° Etape : implémentation du correcteur sur le système réel



Sur Control'X, l'implémentation du correcteur sur le système réel est immédiate et transparente.

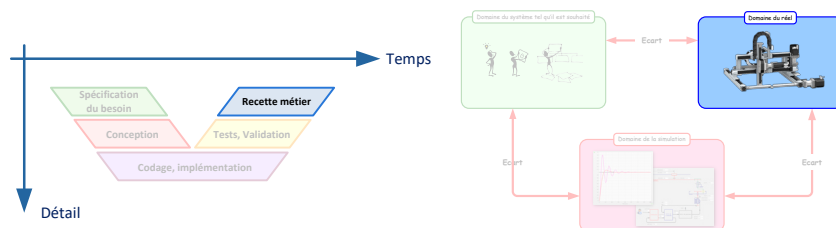
## 6° Etape : validation du comportement réel



La finalité de cette étape consiste à valider que le correcteur qui fonctionnait sur le système simulé permet effectivement de respecter les performances du cahier de charges mais sur le système réel cette fois-ci.

C'est dans cette 6° étape que l'écart souhaité-réel est enfin analysé. Il s'agit du seul écart qui finalement intéresse l'utilisateur final du système. Le modèle ne nous est plus d'aucune utilité, ça n'a été qu'un moyen de conception.

## 7° Etape : la "recette métier" est obtenue



Ca y est, le cycle de conception est terminé, la "recette métier" est obtenue ! Il nous resterait peut-être à justifier le terme "cycle" lorsque l'on parle de cycle de conception ou de cycle en V : cela se justifie en précisant que la démarche suivie n'est pas forcément linéaire. A tout instant, des retours en arrière, des itérations peuvent être effectués en cas de difficulté.

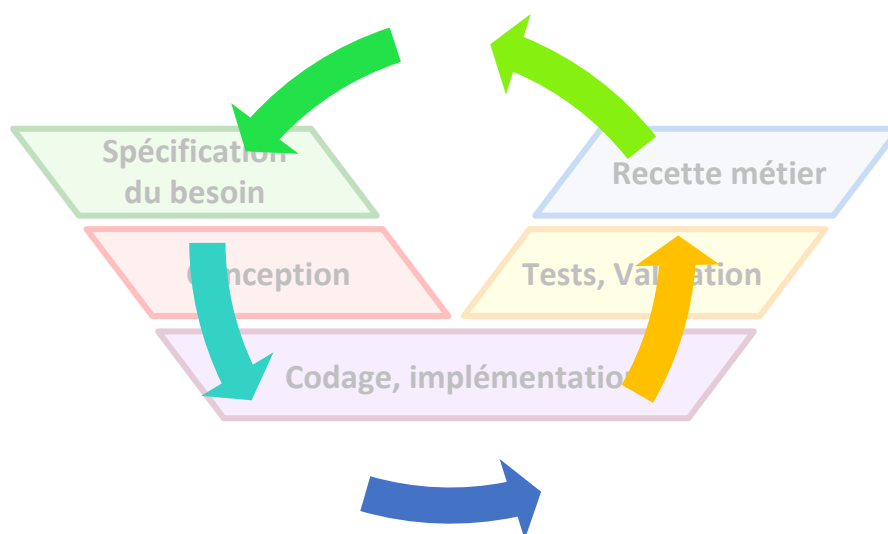


Figure 4 : justification de la notion de "cycle" de conception

## Conclusion

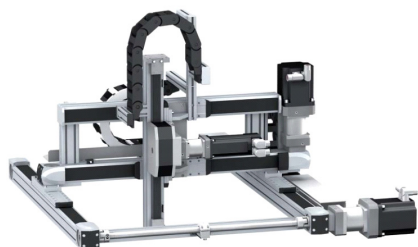
---

On espère qu'à l'issue de ces deux TP, en plus des activités purement techniques, les élèves ont pu éclaircir un peu la vision qu'ils peuvent avoir de notre matière. Qui fait quoi, quand, comment et pourquoi ? Quelle est la force d'une démarche de conception reposant le plus longtemps possible sur un modèle. Est-elle forcément plus performante qu'une démarche de type "essai-erreur".

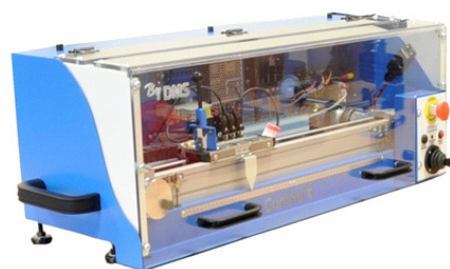
Un diaporama permettant de résumer tout cela permet d'appuyer un peu le discours du professeur. Il est l'évolution, abondamment remaniée, d'une conférence donnée aux journées de l'UPSTI à Marrakech en 2015.

## Quelques mots sur l'axe linéaire utilisé

L'axe linéaire Control'X commercialisé par la société DMS est une version didactique d'un robot multiaxes commercialisé par Schneider. Il est en tout point identique à celui utilisé dans l'industrie en ce qui concerne sa partie mécanique : réducteur, poulies-courroie, chariot, guidages.



Robot multiaxe industriel



Control'X



Figure 5 : du système industriel au système pédagogique

Control'X est un axe linéaire asservi en position. Il repose principalement sur les éléments matériels suivants :

- Un PC qui fait tourner un logiciel de contrôle-commande
- Une carte de commande
- Une carte de puissance (variateur de vitesse)
- L'axe proprement dit : moteur, réducteur, poulies-courroie, chariot et codeur incrémental

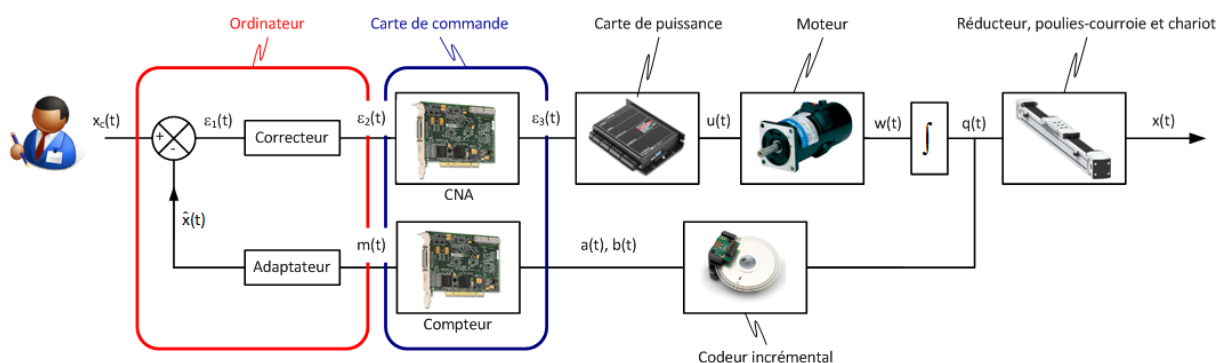


Figure 6 : structure de l'asservissement

Control'X est équipé de très nombreux capteurs, le seul indispensable à l'asservissement de l'axe dans son contexte d'origine est le codeur incrémental monté en bout d'arbre moteur.

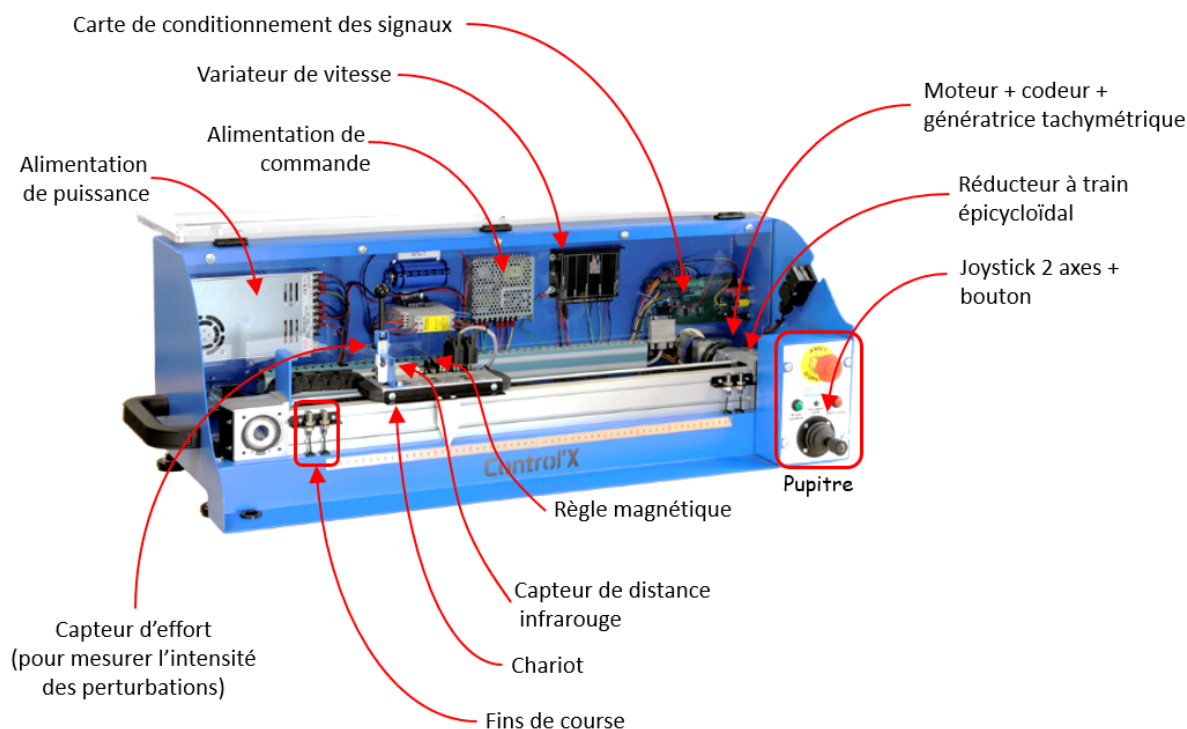


Figure 7 : instrumentation

Parmi les capteurs " pédagogiques", on trouve :

- Capteur de tension en sortie de carte de commande (entrée de variateur)
- Capteur de tension moteur
- Capteur d'intensité moteur
- Génératrice tachymétrique montée en bout d'arbre moteur
- Capteur d'efforts exercés sur le chariot
- Règle magnétique montée sur le chariot
- Capteur de tension pour signaux optionnels (codeur supplémentaire, accéléromètre, capteur de distance infrarouge)



Les schémas ci-après présentent plus finement l'architecture de Control'X :

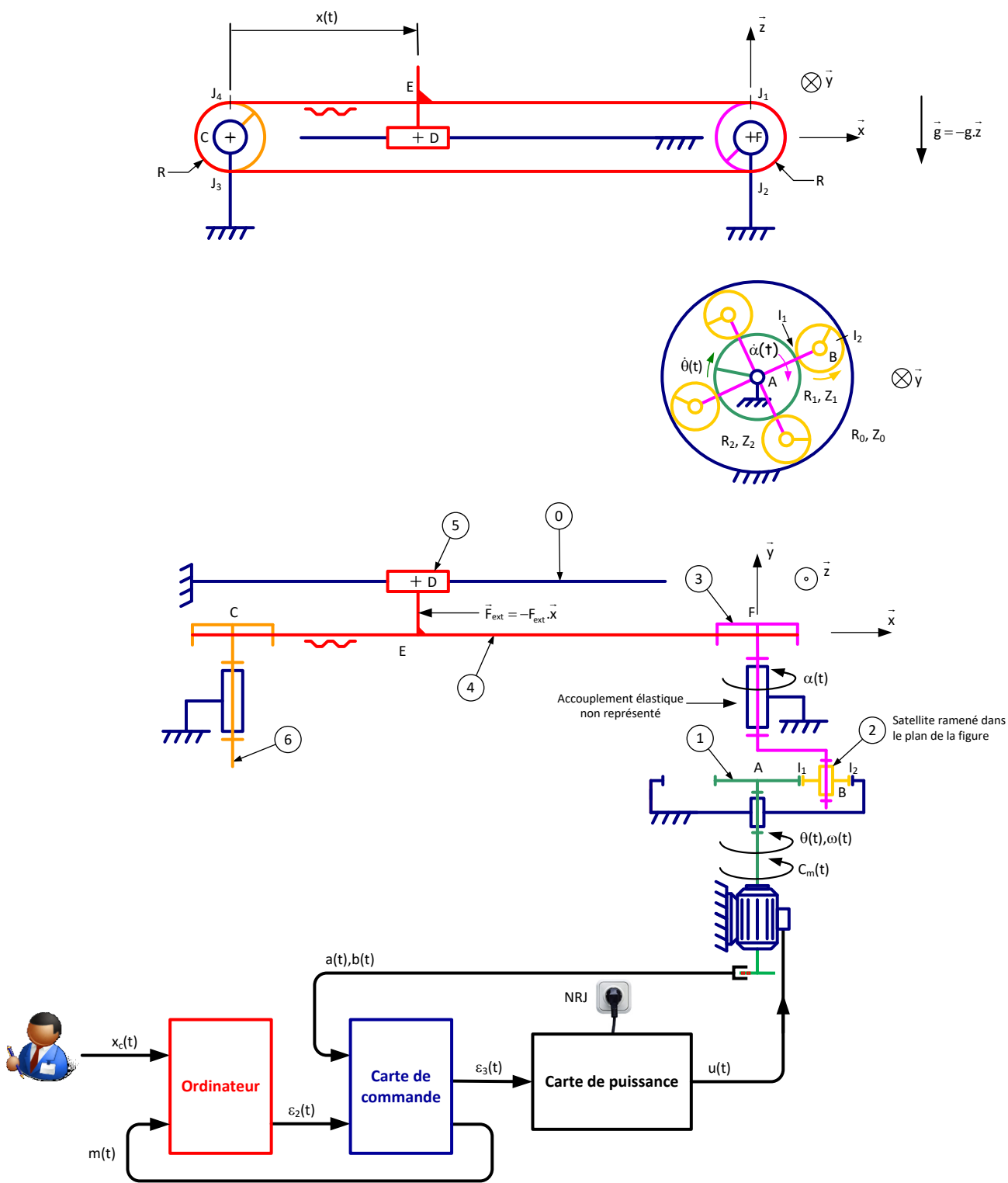


Figure 8 : structure générale

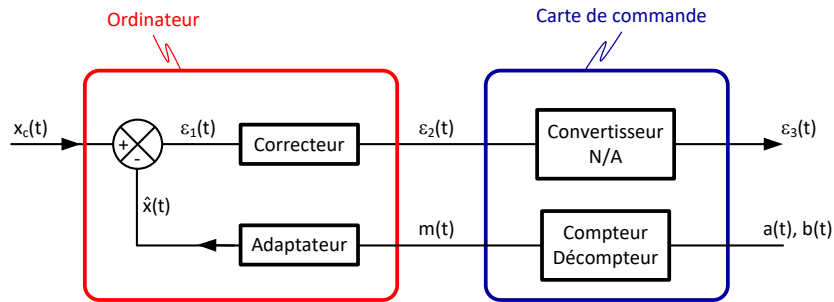


Figure 9 : détail des fonctions remplies par l'ordinateur et la carte de commande

Le schéma ci-dessous met en évidence l'organisation structurale et fonctionnelle des composants nécessaires à la mise en œuvre de l'asservissement de position.

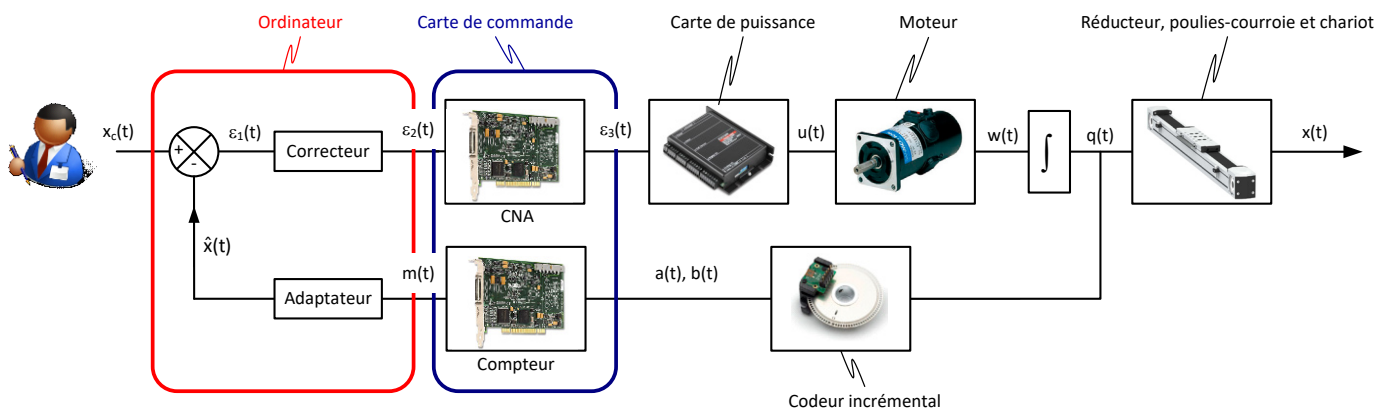


Figure 10 : schéma bloc de l'asservissement

### Fonction de la carte de commande :

#### Coté acquisition :

- Acquérir les deux signaux  $a(t)$  et  $b(t)$  en quadrature de phase venant du codeur incrémental ;
- Compter ou décompter les impulsions  $a(t)$  et  $b(t)$  issues du codeur pour élaborer un signal numérique  $m(t)$  en incréments image de la position linéaire  $x(t)$  du chariot ;
- Transmettre le résultat de ce comptage  $m(t)$  à l'ordinateur.

#### Coté génération

- Recevoir le signal "écart corrigé"  $\varepsilon_2(t)$  provenant de l'ordinateur ;
- Elaborer la tension analogique  $\varepsilon_3(t)$  par conversion numérique analogique de  $\varepsilon_2(t)$ .

Nota : la carte de commande ne constitue qu'une interface entre le PC et le procédé : ce n'est pas elle qui effectue les calculs de commande d'axe (adaptation, soustraction, correction).

## Fonction de l'ordinateur :

- Acquérir la consigne  $x_c(t)$  de l'utilisateur (entrée directement en mm par l'utilisateur).
- Adapter le signal de mesure  $m(t)$  en incréments en signal  $\hat{x}(t)$  en mm ;
- Soustraire le signal  $\hat{x}(t)$  au signal de consigne  $x_c(t)$  pour obtenir l'écart  $\varepsilon_1(t)$  ;
- Appliquer un algorithme de correction pour transformer l'écart  $\varepsilon_1(t)$  en écart corrigé  $\varepsilon_2(t)$ .

## Fonction de la carte de puissance :

Ici cette carte n'est utilisée qu'en mode "variateur de vitesse" : les boucles internes de courant et de vitesse sont désactivées.

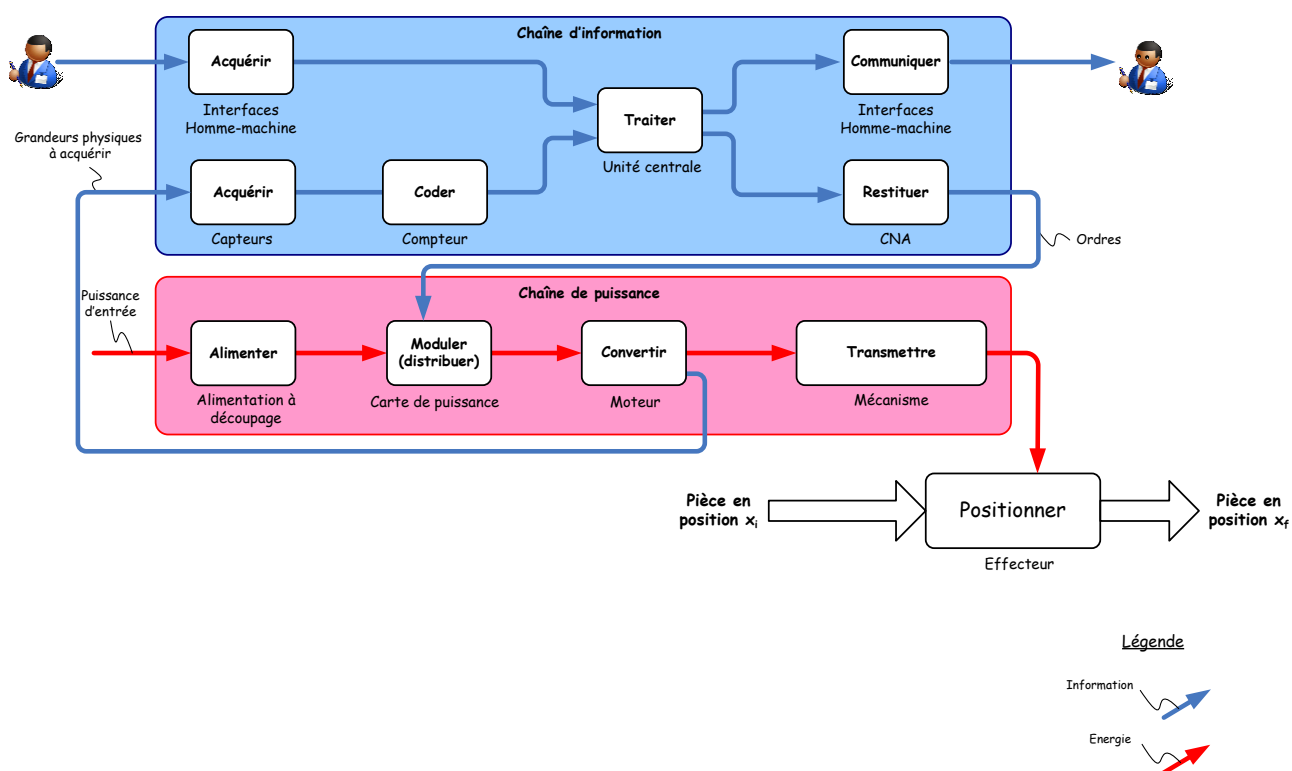


Figure 11 : chaîne fonctionnelle