

Document de formation
pour une solution complète d'automatisation
Totally Integrated Automation (T I A)

MODULE B3
Techniques de régulation avec STEP 7

Ce document a été édité par Siemens A&D SCE (Automatisierungs- und Antriebstechnik, Siemens A&D Cooperates with Education) à des fins de formation.
Siemens ne se porte pas garant de son contenu.

La communication, la distribution et l'utilisation de ce document sont autorisées dans le cadre de formation publique. En dehors de ces conditions, une autorisation écrite par Siemens A&D SCE est exigée (M. Knust : E-mail : michael.knust@hvr.siemens.de).

Tout non-respect de cette règle entraînera des dommages et intérêts. Tous les droits, ceux de la traduction y compris, sont réservés, en particulier dans le cas de brevets ou de modèles déposés.

Nous remercions l'entreprise Michael Dziallas Engineering et les enseignants d'écoles professionnelles ainsi que tous ceux qui ont participé à l'élaboration de ce document.

Sommaire :

1. Avant-propos	5
2. Connaissances de bases en régulation	7
2.1 La régulation selon la norme DIN 19226 :	7
2.2 Composition d'une boucle de régulation	8
2.3 Linéarisation	11
2.4 Echelon de consigne et boucle de régulation	12
2.5. Systèmes stables	13
2.5.1. Système proportionnel sans retard de temps	13
2.5.2. Système proportionnel avec un retard de temps	14
2.5.3 Système proportionnel avec deux retards de temps	15
2.6 Systèmes instables	17
2.7 Type de régulateurs	18
2.7.1 Régulateur tout ou rien	18
2.7.2 Régulateur trois états	20
2.7.3 Les régulateurs continus	21
2.8 Choix et paramétrage du régulateur	27
2.9 Régulateurs numériques	29
3. Réalisation d'un PID sous STEP7 avec le bloc de régulation continue (S)FB41 "CONT_C"	31
3.1 Régulateur PID, données du problème	31
3.2 Le (S)FB 41 „CONT_C“	33
3.2 Exemple de programmation	34
4. Réglage de systèmes	48
4.1 Généralités	48
4.2 Mise au point du régulateur PI selon Ziegler- Nichols	49
4.3 Mise au point du régulateur PI selon Chien, Hrones et Reswick	49
4.4 Exemple d'application	51
4. Régulateur binaire tout ou rien	54
4.1 Enoncé du problème	54
4.2 Programmation possible	56
5. Annexes	59

Légende des symboles utilisés dans ce module :



Information



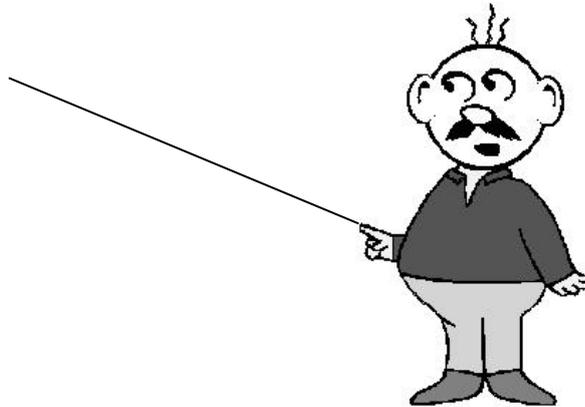
Programmation



Exemple



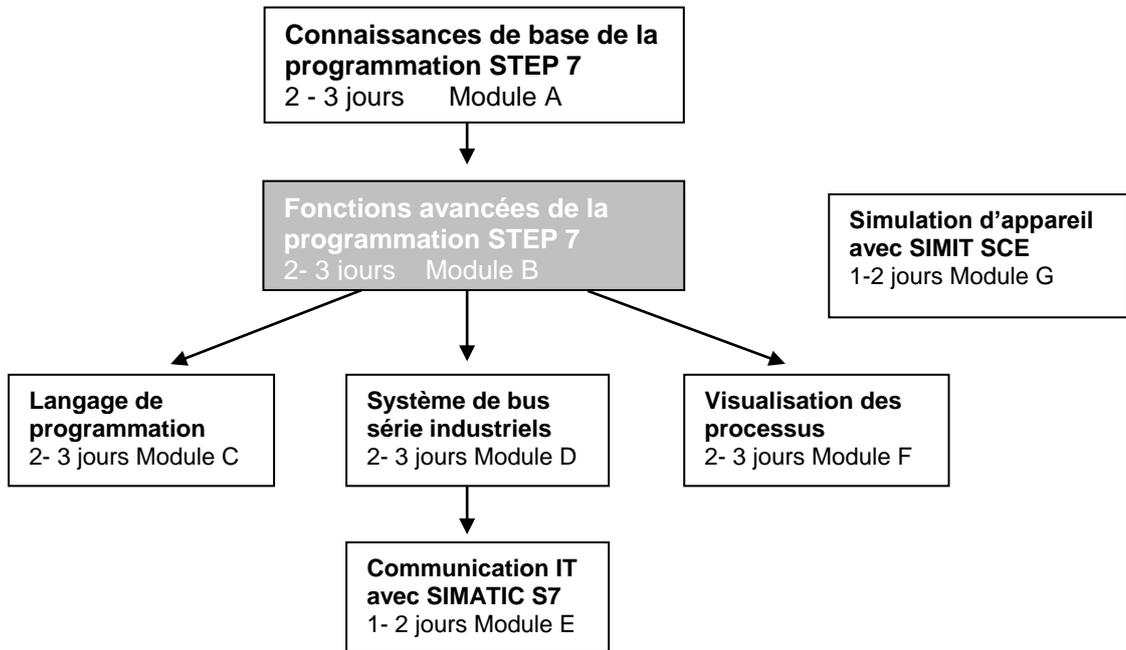
Indication



1. Avant-propos



Le contenu du module B3 est assigné à l'unité ,Fonctions avancées de la programmation STEP7'.



Objectifs :

Le lecteur apprendra à mettre en œuvre et à programmer un régulateur PID dans le cadre d'une application analogique avec STEP7.

- Appel du programme ,Régulateur PID' de STEP 7
- Interaction du régulateur PID avec un procédé analogique
- Configuration des paramètres du régulateur PID

Pré-requis :

Les connaissances suivantes sont requises pour l'étude de ce module :

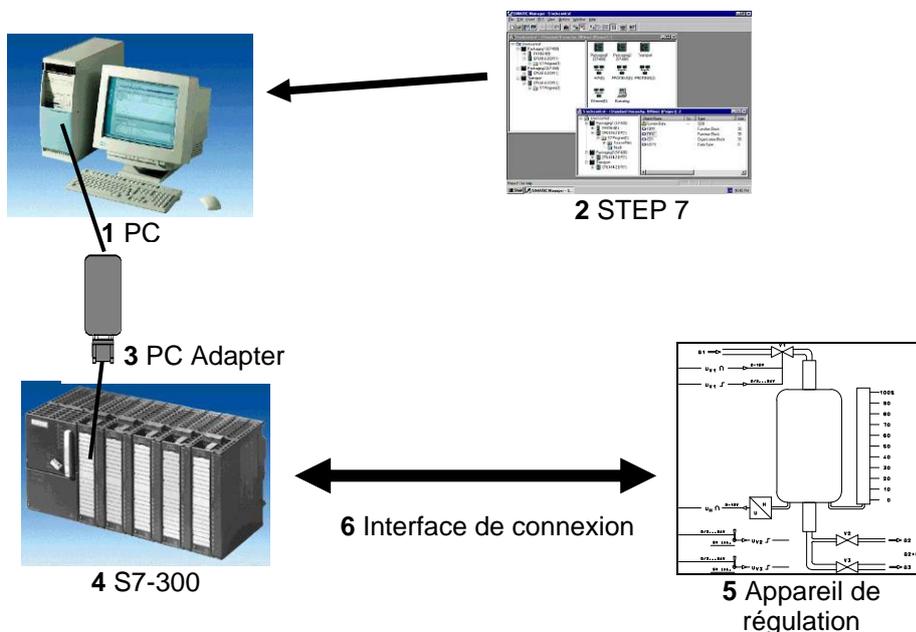
- Systèmes d'exploitation : Windows 95/98/2000/ME/NT4.0/XP
- Base en programmation d'automate avec STEP7 (Ex : Module A3 ,Startup', programmation automate avec STEP 7)
- Traitement de données analogiques avec STEP 7 (Ex : Module B2 – Traitement de données analogiques)

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
-------------------	---------------------	----------------------------	---------------



Configurations matérielles et logicielles requises

- 1 PC, système d'exploitation : Windows 95/98/2000/ME/NT4.0/XP avec
 - Minimum : 133MHz et 64Mo RAM, 65 Mo d'espace disponible
 - Optimal : 500MHz et 128Mo RAM, 65 Mo d'espace disponible
- 2 Logiciel STEP 7 V 5.x
- 3 Interface ordinateur MPI (Ex : PC- Adapter)
- 4 Automate SIMATIC S7-300
 - Exemple de configuration :
 - Bloc d'alimentation : PS 307 2A
 - CPU : CPU 314
 - Entrées numériques : DI 16x DC24V
 - Sorties numériques : DO 16x DC24V / 0,5 A
- 5 Appareil de régulation (Ex.: W-3545-5C de chez WUEKRO)
- 6 Interface de connexion entre l'appareil de régulation et la carte d'entrées/sorties de l'automate programmable.



Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

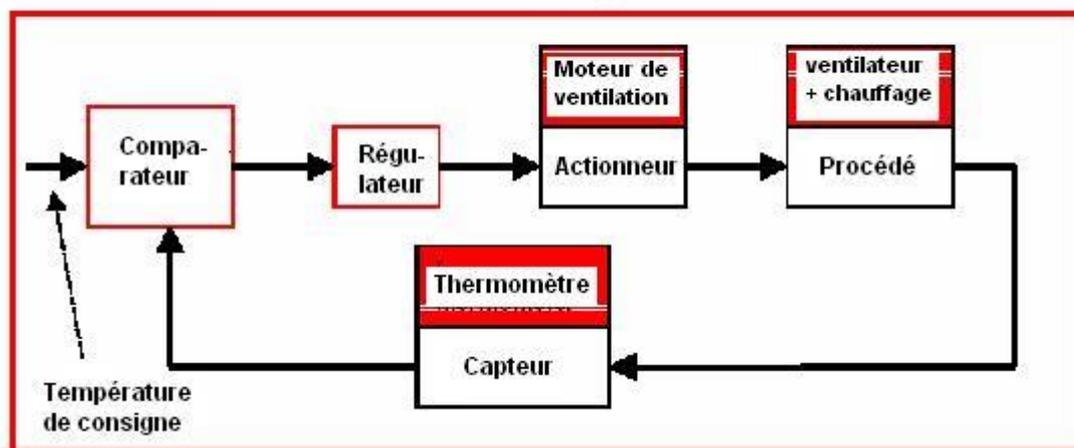
2. Connaissances de bases en régulation

2.1 La régulation selon la norme DIN 19226 :

„Selon la norme DIN 19226 sur les techniques de régulation, la commande est le processus dans un système par lequel une ou plusieurs grandeurs, appelées grandeurs d'entrées, influent sur d'autres grandeurs, appelées grandeurs de sorties, selon les lois propres au dit système.“

La grandeur à réguler est mesurée de façon continue et est ensuite comparée avec une valeur donnée en entrée du système. La régulation devra, en fonction du résultat de la comparaison, rapprocher la valeur à réguler de celle donnée en entrée.

Schéma d'une régulation

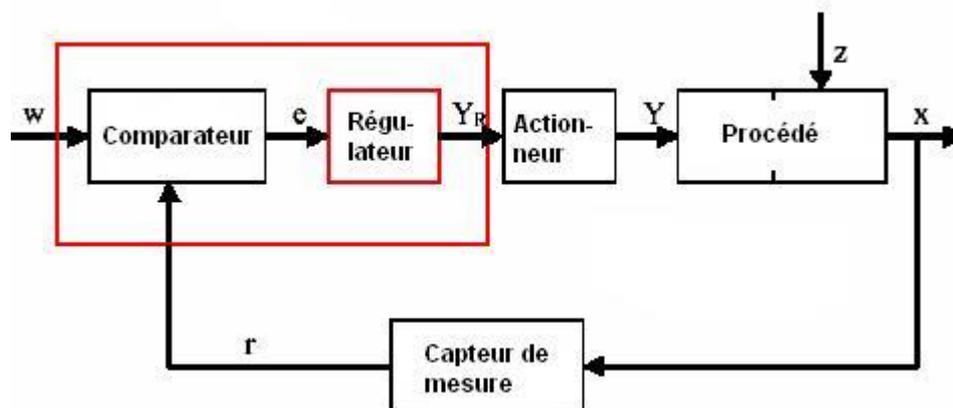


Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

2.2 Composition d'une boucle de régulation

Dans le paragraphe suivant vont être expliqués un par un les termes spécifiques aux techniques de régulations tels que « valeur de consigne », « valeur de sortie » ... etc.

Ceux-ci sont illustrés dans le schéma suivant :



1. La sortie x

Elle est le résultat de la régulation, la variable que le système va influencer et/ou essayer de garder constante. Dans l'exemple d'un chauffage, ce serait la température ambiante. Cette variable est la valeur réelle de la grandeur mesurée.

2. Le retour r

Dans une boucle de régulation, la sortie est constamment contrôlée, il est ainsi possible de réagir à toute variation indésirable de celle-ci. La valeur mesurée (proportionnelle à la sortie) est appelée retour (ou mesure). Dans l'exemple d'un chauffage, cette valeur correspondrait à la tension de mesure d'un thermomètre.

3. La consigne w

La consigne est la grandeur qui doit commander la sortie, c'est à dire la valeur vers laquelle celle-ci doit tendre, pour finalement l'égaliser. Si la consigne est constante, la sortie doit le rester (comportement statique). Au contraire si celle-ci change, le but de la régulation dynamique est alors de reproduire les changements de consigne le plus fidèlement possible au niveau de la sortie.

Attention : la consigne n'a pas la même dimension que la sortie, elle doit être en accord avec la dimension de la mesure. Dans notre exemple, la consigne doit correspondre à la tension que délivrerait le thermomètre à la température souhaitée et non la température elle-même. La sortie est une température (degré), la consigne une tension (Volt).

4. La perturbation z

La perturbation est la grandeur qui influe de manière indésirable sur la sortie et qui l'éloigne de la valeur souhaitée (consigne). La seule existence d'une perturbation rend nécessaire la mise en œuvre d'une régulation statique. Dans notre cas de chauffage, la perturbation serait dépendante de la température extérieure, de l'isolation de la pièce et de tout autre élément poussant la température à s'écarter de la température souhaitée.

5. Le comparateur

C'est l'endroit où la mesure actuelle de la sortie et la valeur de la consigne sont comparées. Dans la plupart des cas les deux grandeurs sont des tensions. La différence des deux grandeurs ainsi obtenue est appelée erreur e . La valeur de l'erreur est passée en entrée du régulateur pour y être traitée.

6. Le régulateur

Le régulateur est l'élément central d'une régulation. Il évalue l'erreur calculée par le comparateur, c'est à dire l'écart entre la sortie et la consigne, et en déduit à partir de celle-ci une valeur régulée ou valeur de correction à transmettre au procédé, afin de corriger la sortie. La sortie du régulateur serait dans le cas de notre chauffage la tension à appliquer au moteur de ventilation.

La façon (l'algorithme) dont le régulateur calcule la valeur régulée à partir de l'erreur est la principale activité de la régulation. Ce point est traité plus en détail dans la partie II.

7. L'actionneur

L'actionneur est en quelque sorte „l'organe exécutif“ de la régulation. Il reçoit la variable régulée du régulateur. Cette variable d'entrée lui permet de savoir comment il doit influencer sur la sortie de la régulation. Dans notre exemple, l'actionneur serait le moteur de la ventilation. En fonction de la valeur régulée obtenue (tension d'entrée), la position du ventilateur est modifiée (et donc indirectement la température).

8. Le procédé

Le procédé est le cœur du système régulé ou encore la partie originelle. C'est également la partie qui agit directement sur la valeur de sortie. Dans notre cas, ce serait l'ensemble ventilateur plus chauffage.

9. Le système régulé

Le système régulé est le système où se trouve la grandeur à réguler. Dans le cas de notre chauffage, ce serait la pièce pour laquelle la température est régulée.

10. Retard ou temps mort

Le temps mort est le temps nécessaire pour qu'une modification de la sortie du régulateur provoque un changement mesurable de la sortie régulée.

Dans notre exemple, ce serait le temps écoulé entre un changement de la tension d'entrée du mélangeur et une variation mesurable de la température ambiante.

11. Boucle de régulation

Réaction d'une boucle de régulation à la modification des grandeurs d'environnement :

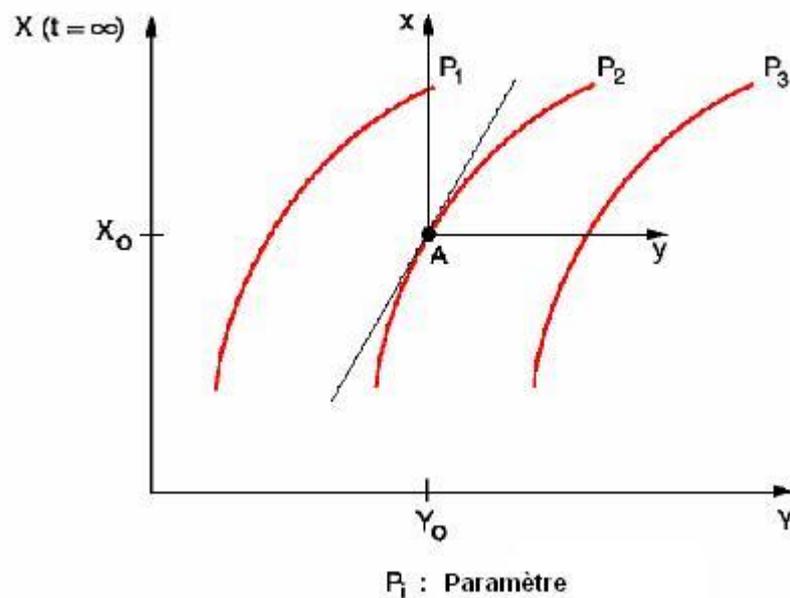
A quelle nouvelle valeur doit être ajustée la sortie si l'on veut atteindre un état d'équilibre ?

A quoi peut s'apparenter le passage à un nouvel état d'équilibre en fonction du temps ?

2.3. Linéarisation

Dans le cas d'un système stable, la sortie va, au bout d'un certain temps, reprendre une valeur constante après l'application d'un échelon de consigne en entrée.

En représentant la sortie en fonction de l'entrée, on obtient ce type de courbe caractéristique :



Pour de tels systèmes non linéaires, seule une étude locale sera possible. Celle-ci se concentrera autour d'un point de fonctionnement (A) où seule la tangente en ce point sera considérée. Le système est alors considéré comme linéaire autour du point de fonctionnement.

Le point de fonctionnement devient le point d'origine des différentes grandeurs ($x(t)$, $y(t)$ et $z(t)$), un changement de repère est nécessaire :

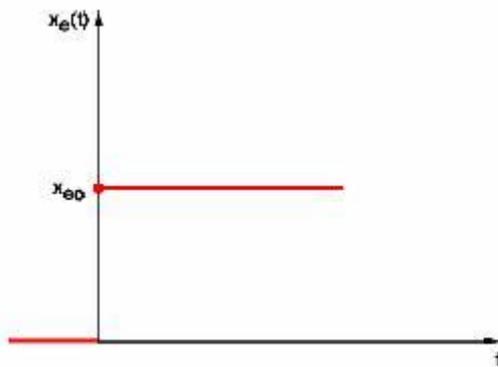
$$x = X - X_0 \quad y = Y - Y_0 \quad z = Z - Z_0$$

2.4 Echelon de consigne et boucle de régulation

Pour étudier le comportement d'un système, d'un régulateur ou d'une boucle de régulation, une seule et même fonction sera utilisée pour le signal d'entrée : l'échelon de consigne.

Selon l'objet de l'étude (une boucle de régulation complète ou juste un régulateur), il est possible de simuler la consigne $w(t)$, la sortie $x(t)$, la valeur réglée $y(t)$ ou la perturbation $z(t)$ avec ce même échelon de consigne.

On utilisera la notation $x_e(t)$ pour désigner ce signal d'entrée en échelon.

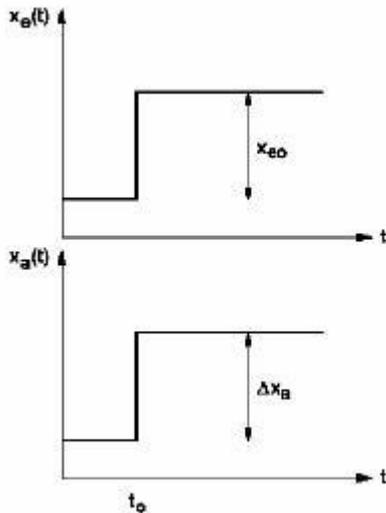


$$x_e(t) = \begin{cases} 0 & t < 0 \\ x_{BO} & t \geq 0 \end{cases}$$

2.5. Systèmes stables

2.5.1. Système proportionnel sans retard de temps

Le système est dit purement proportionnel, on le notera système P.



Application d'un échelon de consigne au temps t_0

Sortie en fonction de la valeur réglée :

$$x = K_{ss} \cdot y$$

K_{ss} : facteur proportionnel de réglage

$$K_{ss} = \frac{\Delta x}{\Delta y} = \tan \alpha$$

Sortie en fonction de la perturbation :

$$x = K_{BZ} \cdot z$$

K_{BZ} : facteur proportionnel de perturbation

Plage de réglage :

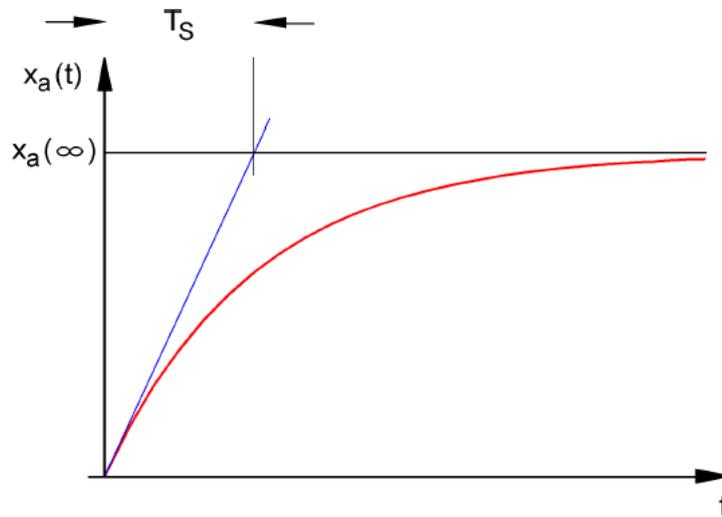
$$y_h = y_{\max} - y_{\min}$$

Plage de sortie :

$$x_h = x_{\max} - x_{\min}$$

2.5.2. Système proportionnel avec un retard de temps

On utilisera l'abréviation P-T1 pour ce type de système (retard simple).



Equation différentielle pour une entrée quelconque $x_e(t)$:

$$T_S \cdot \dot{x}_a(t) + x_a(t) = K_{PS} \cdot x_e(t)$$

Solution de l'équation pour une entrée en échelon :

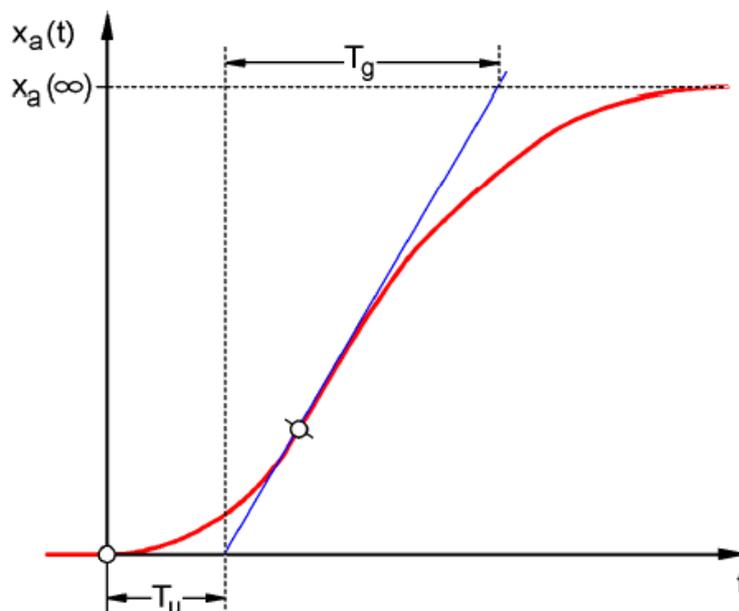
$$x_a(t) = K_{PS} (1 - e^{-t/T_S}) \cdot x_{e0}$$

$$x_a(t = \infty) = K_{PS} \cdot x_{e0}$$

T_S : Constante de temps

2.5.3 Système proportionnel avec deux retards de temps

On utilisera l'abréviation P-T2 pour ce type de système.



Réponse indicielle d'un système P-T2

T_u : Temps de retard pur T_g : Temps de « compensation »

Le système va être considéré comme la mise en cascade de deux systèmes de type PT1 indépendants avec leurs propres constantes de temps TS1 et TS2.

Critère de régulation d'un système P-T2 :

$$\frac{T_u}{T_g} < \frac{1}{10} \rightarrow \text{régulation facile} \quad \frac{T_u}{T_g} \approx \frac{1}{6} \rightarrow \text{régulation possible} \quad \frac{T_u}{T_g} > \frac{1}{3} \rightarrow \text{régulation difficile}$$

Plus le facteur T_u / T_g est grand, plus le système est difficilement régulier.

2.5.4 Système proportionnel avec n retards de temps

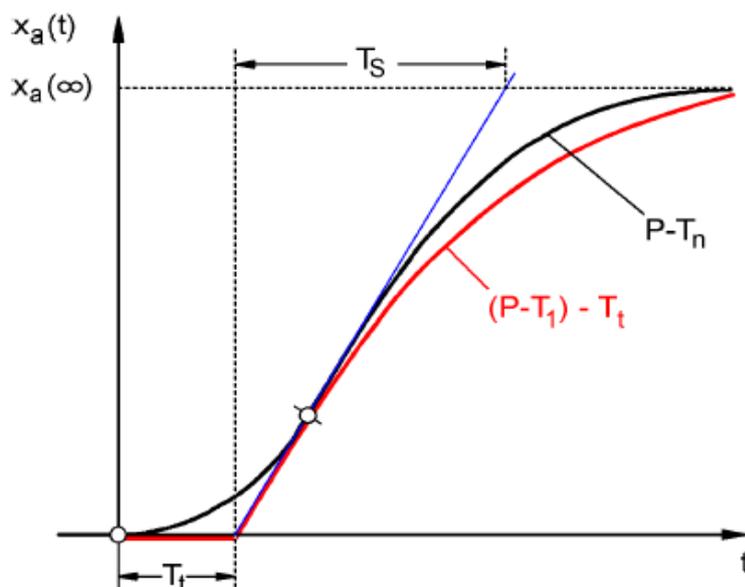
On utilisera l'abréviation P-Tn pour ce type de système.

La description de leur comportement par rapport au temps se fait grâce à une équation différentielle de degré n.

La réponse indicielle du système est équivalente à celle d'un système de type P-T2. Les retards peuvent également être assimilés aux constantes de temps T_u et T_g .

Remarque : les systèmes avec beaucoup de retard peuvent être assimilés à une cascade de systèmes P-T1 avec retard pur.

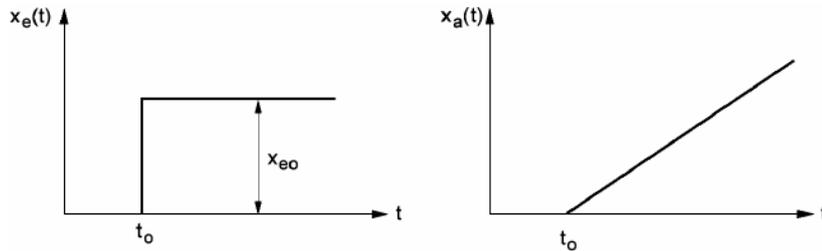
Ceci vaut pour : $T_t \gg T_u$ et $T_S \gg T_g$.



Réponse indicielle assimilée pour les systèmes de type P-Tn

2.6 Systèmes instables

Après avoir été perturbée, la valeur de sortie du système augmente de manière infinie. La réponse d'un système instable à un échelon de consigne diverge.



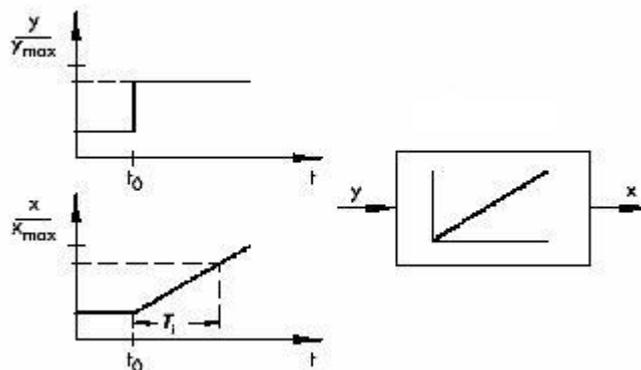
Exemple : Régulation de niveau

Considérons un conteneur dont le débit de remplissage est égal au débit d'écoulement, il présente donc un niveau constant de contenu.

Si le débit d'entrée ou de sortie se trouve être modifié, le niveau de liquide va alors augmenter ou diminuer, et ce, d'autant plus rapidement que la différence entre le débit d'entrée et de sortie est importante.

L'exemple montre qu'un comportement instable a dans la pratique le plus souvent une limite physique. Dans notre cas, le niveau de liquide va augmenter ou diminuer jusqu'aux limites du conteneur (jusqu'à ce qu'il soit complètement vide ou qu'il déborde).

On dit également que ce type de système a un comportement intégral, ou système de type I. La réponse à un échelon de consigne de ce genre de système est décrite par la figure ci-dessous :



Si l'on considère le signal d'entrée $x_e(t)$, on a :

$$X_a(t) = K_i \int x_e(t) dt \rightarrow \text{système intégral}$$

K_i : Facteur intégral

2.7 Type de régulateurs

2.7.1 Régulateur tout ou rien

La propriété essentielle d'un régulateur tout ou rien réside dans le fait qu'il ne peut prendre que deux états. Il peut de ce fait être considéré comme le régulateur le plus simple. Ce type de régulateur sera utilisé quand la simplicité de réalisation de la régulation prévaut sur l'observation stricte de la consigne ou encore lorsque la valeur régulée ne peut pas être de type continu.

Dans notre cas de chauffage avec un thermostat et ventilation, il s'agira d'une régulation continue. Par contre pour maintenir constante la température de l'eau dans une chaudière, il sera typiquement utilisé un régulateur tout ou rien, d'une part parce qu'une erreur de quelques degrés est tolérable et d'autre part parce qu'il est plus simple d'allumer ou d'éteindre le brûleur plutôt que de doser la force de la combustion.

Pour maintenir la sortie à la valeur exacte de la consigne, le régulateur doit passer très rapidement de l'état activé à l'état éteint. Pour éviter une commutation trop rapide, une hystérésis de réglage va être définie. Cette zone est limitée par un seuil supérieur et un seuil inférieur définissant une zone autour de la consigne et entre lesquels la sortie pourra varier sans déclencher de commutation du régulateur. La régulation ne sera déclenchée que lorsque la sortie franchit un des deux seuils.

Dans le cas de la chaudière, le brûleur est allumé lorsque la température de l'eau est inférieure de x degrés par rapport à la valeur de consigne. Il restera allumé jusqu'à ce que la température atteigne une valeur supérieure de x degrés à la consigne. Une fois ce seuil supérieur atteint, le brûleur est éteint.

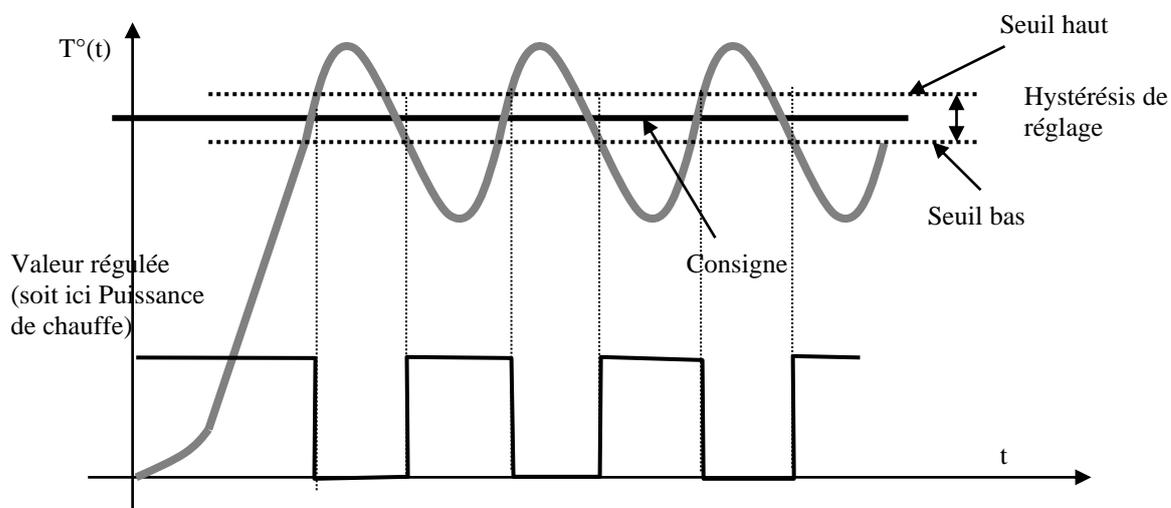
Un autre exemple typique est celui du réfrigérateur.

Le groupe frigorifique (compresseur) ne connaît que l'état allumé ou l'état éteint. Il est donc mis en route lorsque la température du réfrigérateur est déjà supérieure de quelques degrés à la température souhaitée. Il s'éteint quand la température devient inférieure de quelques degrés à la température programmée.

La sortie d'un système avec régulateur tout ou rien va typiquement osciller de façon périodique autour de la consigne en restant dans la zone d'hystérésis.

Le choix des seuils de l'hystérésis se fait en fonction de la tolérance que l'on s'octroie par rapport à la consigne. Une hystérésis large va autoriser de grandes variations autour de la consigne. Une hystérésis fine étroite va assurer une plus grande fidélité à la consigne, mais une fréquence de commutation d'état du régulateur plus élevée.

La régulation tout ou rien est illustrée sur le graphe ci-dessous :



Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	----------------------------	----------------------------	---------------

2.7.2 Régulateur trois états

Les régulateurs trois états sont la deuxième grande catégorie de régulateurs discrets. La différence avec le régulateur tout ou rien réside dans le fait qu'il peut prendre trois états différents : influence positive, inactive (pas d'influence) et influence négative sur le système réglé.

Un exemple d'utilisation est une régulation à l'aide d'une vanne qui ne peut être qu'ouverte complètement ou fermée. Cela est le cas dans la régulation d'un barrage. Dès lors que le niveau maximum d'eau du barrage est atteint, le moteur de commande de la vanne est activé positivement (tension positive) pour procéder à l'ouverture. Le moteur reste ensuite inactif, c'est-à-dire sans stimulation, tant que l'eau n'a pas atteint un niveau minimum. Une fois ce niveau minimum dépassé, le moteur reçoit une commande négative (tension négative) pour procéder à la fermeture de la vanne. Ainsi le procédé connaît trois états : moteur en rotation positive, moteur inactif et moteur en rotation négative.

2.7.3 Les régulateurs continus

Les régulateurs discrets précédemment abordés ont pour principal avantage leur simplicité de mise en œuvre et de réalisation qui les rend par conséquent beaucoup moins coûteux que les régulateurs continus. Ils ont cependant un grand nombre de désavantages.

Par exemple pour des systèmes à charge élevée comme de gros électromoteurs ou gros groupes frigorifiques, cela peut conduire à la surcharge de l'alimentation lors d'une commutation. Dans ce genre de cas, on ne commute généralement pas de l'état éteint à allumé, mais d'un état de pleine puissance à un état de faible puissance correspondant à la charge résiduelle. Malgré cette amélioration, l'utilisation de régulateurs continus est beaucoup plus appropriée à ce genre d'applications.

Imaginez-vous également un moteur de voiture pour lequel le nombre de tours par minute serait réglé discrètement. Il n'y aurait alors le choix qu'entre « ralenti » et « plein gaz ». Lors du passage soudain à l'état de puissance maximum, il serait totalement impossible de transférer l'effort à la route par l'intermédiaire des pneus. Il semble évident que ce genre de véhicule serait totalement inapproprié à la circulation routière. On utilisera là encore un régulateur de type continu pour ce genre d'applications.

Les régulateurs continus se répartissent en trois grands types différents qui seront abordés plus en détail dans les pages suivantes.

2.7.3.1 Le régulateur proportionnel (Type P)

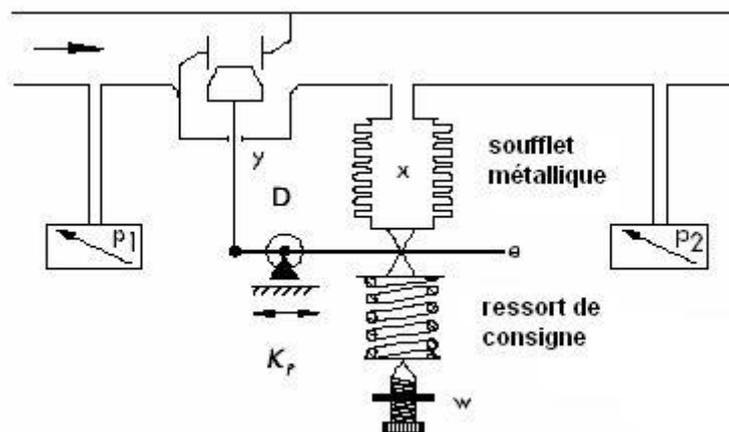
Dans le cas d'un régulateur P, la variable réglée est toujours proportionnelle à l'erreur calculée ($y \sim e$). Il en résulte qu'un régulateur P réagit sans temps de retard à tout écart entre la sortie et la consigne, et seul dans ce cas sera générée une valeur réglée y .

Le régulateur proportionnel de pression représenté sur l'image ci-dessous compare la force FS du ressort de consigne avec la force FB qui résulte de la pression p2 dans le soufflet métallique.

Si les forces ne sont pas égales, le levier pivote autour du point D. Le déplacement du levier va agir sur la position de la vanne \tilde{n} et par conséquent sur la pression p2, et ce jusqu'à avoir atteint un nouvel équilibre des forces.

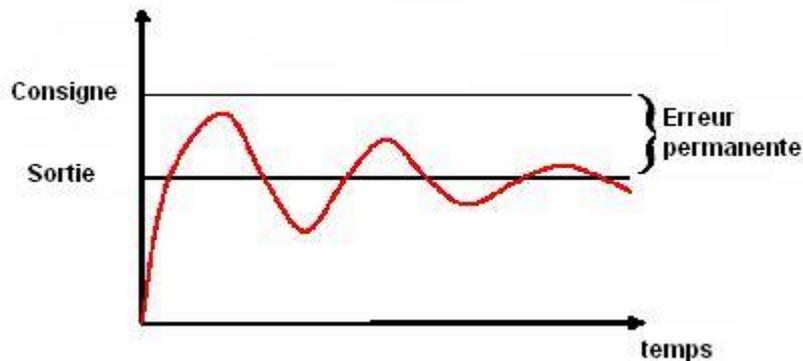
L'amplitude de la variable réglée y dépend de la valeur de l'erreur e et de la valeur du gain proportionnel K_p .

Pour corriger au mieux l'erreur, il faut choisir un facteur de gain proportionnel aussi grand que possible. Une augmentation de ce facteur entraîne une réaction plus rapide du régulateur. Une trop grande valeur de ce facteur va apporter un risque d'oscillation et d'instabilité du système.



$$y = K_p \cdot e$$

Le graphique suivant décrit le comportement d'un régulateur proportionnel :



L'avantage de ce type de régulateur réside d'une part dans sa simplicité (la réalisation électronique se résume dans certains cas à une simple résistance) et d'autre part dans sa rapidité de réaction (par rapport à d'autres types de régulateur).

Le désavantage principal d'un régulateur proportionnel réside en la présence d'une erreur de régulation permanente. En effet, la valeur de consigne ne sera, même à long terme, jamais complètement atteinte. Ce désavantage tout comme celui d'un temps de réaction non idéal ne se laissent qu'imparfaitement compenser par une augmentation du facteur de gain proportionnel, qui tend à rendre le système oscillant. Un trop grand gain proportionnel engendre dans le pire des cas une oscillation permanente, la sortie s'éloignant périodiquement de la valeur de consigne.

Ce problème d'une erreur de régulation permanente peut être résolu avec un régulateur de type intégral.

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	----------------------------	----------------------------	---------------

2.7.3.2 Le régulateur intégral (type I)

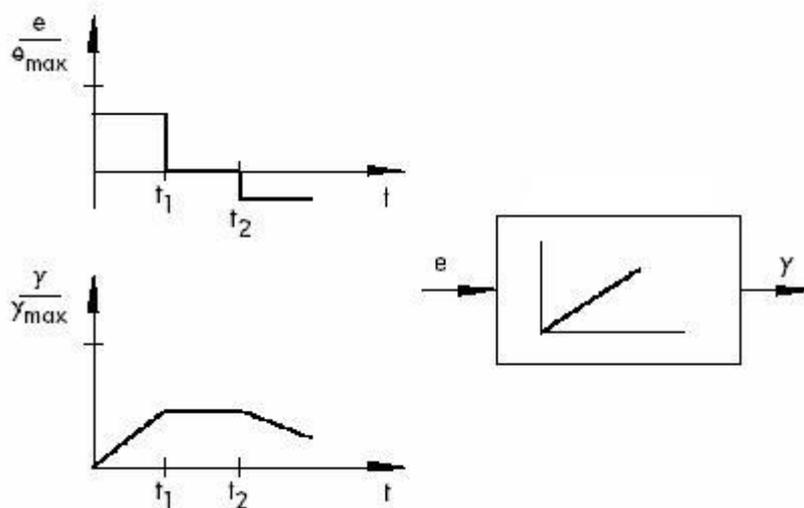
Un régulateur de type intégral permet de complètement compenser une erreur de régulation constante. Tant que l'erreur n'est pas nulle, la valeur de la variable régulée est ajustée.

La régulation se termine lorsque la sortie a atteint la valeur de la consigne ou que la variable régulée a atteint un seuil maximal fixé par les propriétés du système (U_{max} , P_{max} ...etc.).

La formulation mathématique de ce comportement intégral est : la variable régulée est proportionnelle à l'intégrale par rapport au temps de l'erreur.

$$y = K_i \int e dt \qquad K_i = \frac{1}{T_n}$$

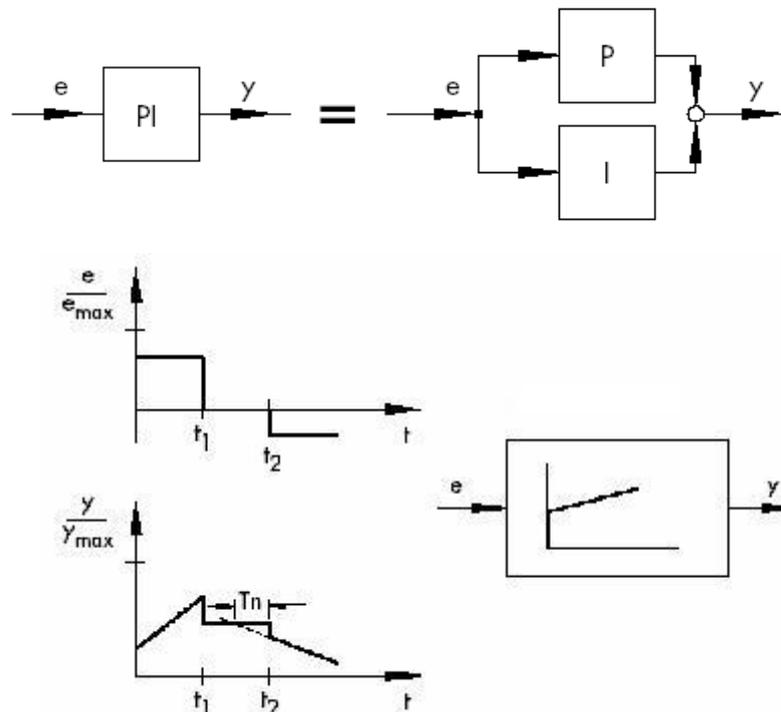
La vitesse avec laquelle la variable régulée augmente (ou diminue) dépend de l'erreur de régulation et de la constante de temps d'intégration choisie.



2.7.3.3 Régulateur de type PI

Le régulateur de type PI est l'un des plus utilisés dans la pratique. Il s'agit d'un couplage parallèle entre un régulateur proportionnel et un régulateur intégral.

S'il est correctement paramétré, le régulateur PI cumule les avantages des deux types de régulateurs (stable, rapide, faible erreur résiduelle), tout en compensant leurs désavantages respectifs.



Le comportement dans le temps est caractérisé par le facteur proportionnel K_p et la constante d'intégration T_n . Grâce à la partie proportionnelle, le régulateur réagit immédiatement à toute erreur de régulation e , alors que la partie intégrale ne fera effet qu'après un certain temps. La constante d'intégration T_n est le temps qu'il faut à la partie I pour avoir la même amplitude que prend la partie P de façon instantanée. Si l'on veut renforcer la partie intégrale, il faut comme pour un régulateur intégral réduire la constante T_n .

Interprétation:

En fonction du dimensionnement des constantes K_p et T_n , il va être possible de diminuer les risques d'oscillation de la sortie en agissant sur la dynamique même de la régulation. Domaine d'application du régulateur PI : boucle de régulation rapide qui ne tolère aucune erreur résiduelle.

Exemple : Régulation de pression, de température...

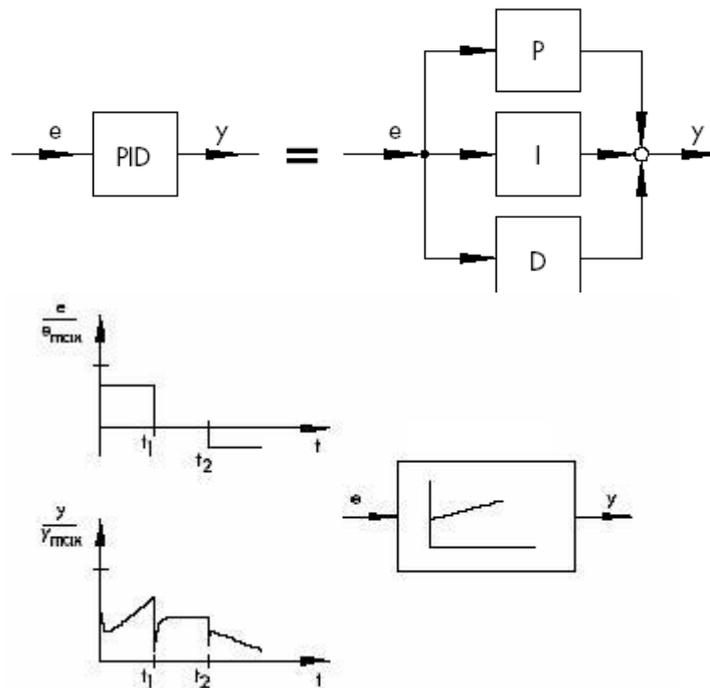
Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

2.7.3.4 Le régulateur dérivé (type D)

Le régulateur dérivé établit une valeur régulée en fonction de la vitesse de variation de l'erreur et pas en fonction de l'amplitude comme pour le régulateur P. C'est pour cette raison qu'il réagit beaucoup plus rapidement qu'un régulateur P. Même face à une petite erreur il va générer une grosse valeur régulée dès lors qu'il y a une variation d'amplitude de l'erreur. Le régulateur D sera inefficace face à une erreur résiduelle permanente, quelque soit sa valeur puisque celle-ci reste constante (pas de variation d'amplitude donc pas de réaction du régulateur). C'est pourquoi ce type de régulateur sera rarement utilisé seul dans la pratique, il est couramment associé à un régulateur de type P.

2.7.3.5 Le régulateur PID

Le régulateur universel PID s'obtient donc en ajoutant une composante de Type D à un régulateur PI. Comme pour un régulateur PD, la partie dérivée assure une plus grande rapidité pour atteindre la valeur de consigne en ajoutant une action anticipatrice au régulateur.



$$y = K_p \cdot e + K_i \int e dt + K_D \frac{de}{dt} \quad K_i = \frac{K_p}{T_n}; K_D = K_p \cdot T_v$$

2.8 Choix et paramétrage du régulateur

Pour l'obtention d'un résultat de régulation satisfaisant, le choix du type de régulateur est déterminant. Le paramétrage de celui-ci n'en est pas moins important, il faut donc judicieusement choisir les constantes K_p , T_n et T_v . Un compromis va généralement devoir être fait entre un système stable mais lent, et une régulation rapide avec des risques d'oscillation et donc d'instabilité.

Dans le cas d'un système non linéaire où il peut être défini un point de fonctionnement, les paramètres du régulateur devront être adaptés au comportement du système autour de ce point. S'il n'est pas possible de définir un point de fonctionnement, il faut trouver une régulation qui soit suffisamment rapide et stable pour convenir à toute la plage de travail.

En pratique, les régulateurs sont généralement paramétrés avec des valeurs obtenues de façon expérimentale. Si ces paramètres ne sont pas satisfaisants, il faudra alors pratiquer une analyse précise du système en utilisant différentes techniques ou théories pour déterminer les réglages adéquats.

Une méthode d'analyse est par exemple celle de Ziegler-Nichols par recherche de la juste oscillation. C'est une méthode simple qui s'applique à la plupart des systèmes. Cette méthode ne s'applique qu'aux systèmes dont la sortie de régulation peut être amenée à osciller. Il faut suivre le protocole suivant :

Choisir les constantes K_p et T_v du régulateur aussi petite que possible, et une grande valeur pour T_n (influence minime du régulateur)

Amener manuellement le système autour de son point de fonctionnement.

Positionner manuellement la valeur régulée au niveau prédéfini avant de passer en fonctionnement automatique.

Augmenter progressivement K_p (réduire X_p) jusqu'à l'obtention d'une oscillation harmonique de la valeur régulée. Par application de petit saut au niveau de la consigne, le système devrait alors rentrer en oscillation.

Le facteur K_p ainsi déterminé sera appelé facteur proportionnel critique et noté $K_{p,crit}$.

La durée d'une oscillation complète sera notée T_{crit} et aura été mesurée avec un chronomètre sur plusieurs périodes d'oscillation.

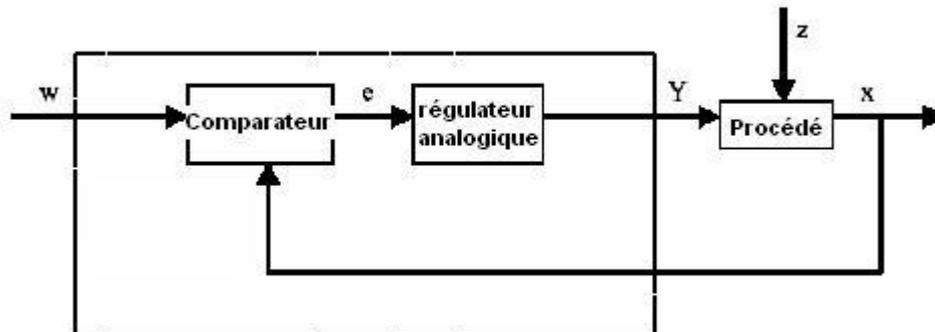
L'utilisation des valeurs obtenues pour $K_{p,crit}$ et T_{crit} dans le tableau ci-dessous donnera les paramètres K_p , T_n et T_v du régulateur à utiliser.

	K_p	T_n	T_v
P	$0,50 \cdot K_{p,crit}$	-	-
PI	$0,45 \cdot K_{p,crit}$	$0,85 \cdot T_{crit}$	-
PID	$0,59 \cdot K_{p,crit}$	$0,50 \cdot T_{crit}$	$0,12 \cdot T_{crit}$

2.9 Régulateurs numériques

Jusque là, seuls les régulateurs analogiques (continus) ont été considérés, c'est-à-dire, ceux qui à partir d'une erreur analogique calcule une valeur régulée, elle aussi de type analogique.

Ci-dessous, le schéma classique d'une telle régulation :



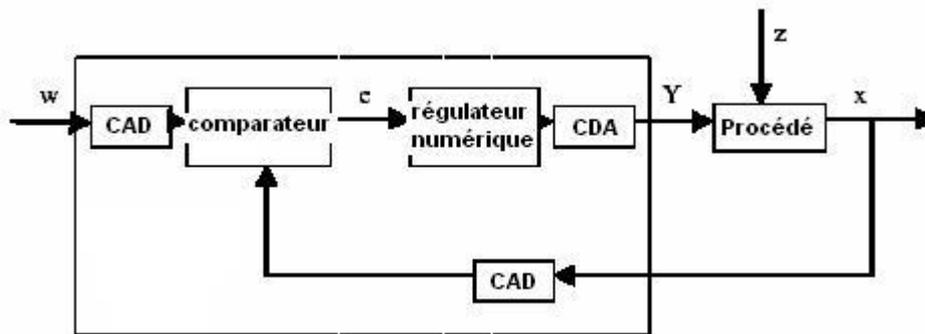
Pourtant, il est bien souvent avantageux de travailler avec une erreur de régulation de type numérique. D'une part parce que l'algorithme reliant l'erreur et la sortie du régulateur va pouvoir être réalisé en programmant un ordinateur ou calculateur, offrant ainsi une plus grande flexibilité que ne le fait une réalisation par circuit analogique. D'autre part, l'électronique numérique permet une plus grande possibilité d'intégration des circuits, de telle sorte que plusieurs régulateurs peuvent être contenus dans un très petit espace. Il est enfin possible d'utiliser le même ordinateur pour gérer plusieurs boucles de régulation si sa capacité le permet.

Pour rendre possible un traitement numérique des données de régulation, il est nécessaire de placer un convertisseur analogique – numérique (CAD) sur la branche de mesure.

L'erreur sera ensuite réalisée grâce à un comparateur numérique en soustrayant la mesure et la consigne pour donner une erreur numérique (entrée du régulateur). La sortie numérique du régulateur sera ensuite traitée avec un convertisseur numérique – analogique (CDA), afin d'obtenir une sortie du système de type analogique. L'ensemble convertisseur, comparateur et régulateur ont, vu de l'extérieur, les mêmes propriétés qu'une régulation analogique.

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	----------------------------	----------------------------	---------------

La structure d'une régulation numérique est représentée sur le schéma suivant :



En plus des avantages évidents que comporte le régulateur numérique, il pose en soi tout de même divers problèmes, et notamment celui de la précision de la numérisation. Les critères pour une bonne régulation numérique sont :

- La résolution du Convertisseur numérique analogique.

Elle permet de choisir la finesse avec laquelle sera reconstitué le signal analogique. La résolution doit être choisie de telle sorte qu'aucune information utile à la régulation ne soit perdue.

- La fréquence d'échantillonnage du convertisseur analogique – numérique.

C'est la fréquence avec laquelle le convertisseur analogique - numérique va mesurer le signal analogique et le numériser. Cette fréquence doit être suffisamment élevée pour que le régulateur puisse réagir même à de petits changements de la sortie.

- La fréquence d'horloge.

Un calculateur numérique fonctionne différemment d'un régulateur analogique, il est commandé par une fréquence d'horloge. La vitesse de calcul de l'ordinateur doit être assez élevée pour qu'aucune modification significative de la sortie n'ait lieu pendant un même cycle de calcul.

Un régulateur numérique doit avoir, vu de l'extérieur, une qualité comparable à un régulateur continu, tant en précision qu'en rapidité de réaction, pour pouvoir être utilisé.

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	----------------------------	----------------------------	---------------

3. Réalisation d'un PID sous STEP7 avec le bloc de régulation continue (S)FB41 "CONT_C"

3.1 Régulateur PID, données du problème

Ce module explique la réalisation pas à pas d'un régulateur PID avec un système SIMATIC S7-300. Dans notre exemple, la tension de sortie du système régulé devra être maintenue constante grâce à une régulation continue. En fonction des réglages appliqués au système nous simulerons une régulation de type P et PI. Le facteur de transfert K_s ainsi que les constantes de temps sont paramétrables.

L'automate programmable est relié au système régulé avec des entrées / sorties analogiques pour influencer sur celui-ci.

Dans cet exemple, le (S)FB 41 de l'AP SIMATIC S7-300 va être utilisé en tant que régulateur continu numérique. Son rôle sera de calculer une valeur d'ajustement y en fonction de l'erreur (différence mesure/consigne) $e=w-x$ selon l'algorithme d'un régulateur PID, et de livrer cette grandeur d'ajustement y sur sa sortie analogique.

Pour cela, il est nécessaire de définir les paramètres de régulation suivants :

KP : coefficient de gain proportionnel

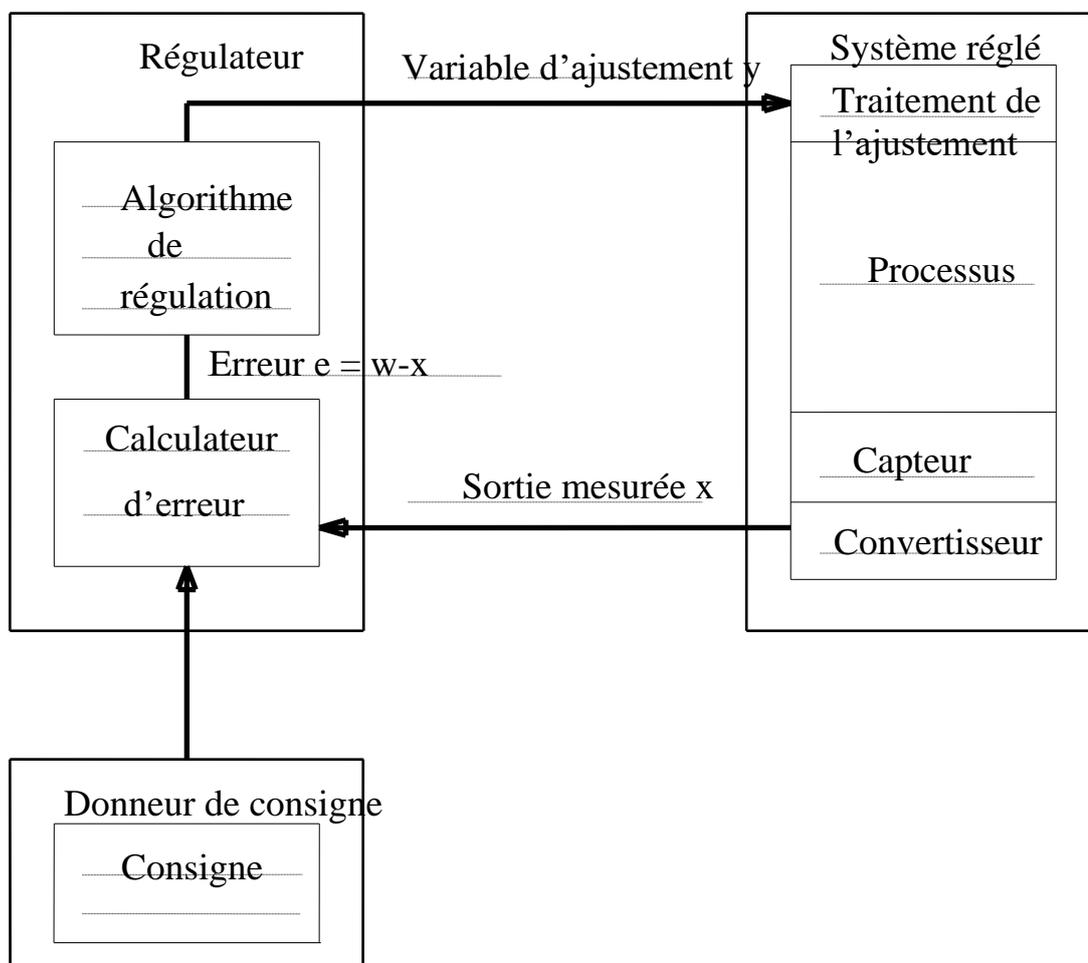
TN : temps d'intégration

TV : temps de dérivation

Tableau récapitulatif des entrées / sorties utilisées dans l'exemple

Variables d'entrée	Représentation symbolique	Type	Adresse
Valeur réelle mesurée	X	Entrée analogique 1:0..10V	PEW 288
Valeur de consigne	w	Appel du (S)FB41 dans l'OB1 Entrée analogique 2 : 0..10V	PEW 290
Variables de sortie			
Tension de sortie (variable d'ajustement)	Y	Sortie analogique 1: 0..10V	PAW 288

Schéma fonctionnel d'une régulation avec un PID



Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	----------------------------	----------------------------	---------------

3.2 Le (S)FB 41 „CONT_C“



Le bloc FB 41 « CONT_C » (continuous controller) sert à réguler des processus industriels à grandeurs d'entrée et de sortie continues sur les automates programmables SIMATIC S7. Le paramétrage vous permet d'activer ou de désactiver des fonctions partielles du régulateur PID et donc d'adapter ce dernier au système régulé.

Utilisation :

Vous pouvez utiliser le régulateur comme régulateur PID de maintien autonome mais aussi comme régulateur en cascade, de mélange ou de rapport dans des régulations à plusieurs boucles. Sa méthode de travail se base sur l'algorithme PID du régulateur à échantillonnage à sortie analogique, complété le cas échéant par un étage conformateur d'impulsions assurant la formation des signaux de sortie à modulation de largeur d'impulsions pour régulations à deux ou trois échelons avec organes de réglage proportionnels.

Description :

En plus des fonctions traitant la consigne et la mesure, le FB réalise un régulateur PID prêt à l'emploi avec sortie continue de la grandeur de réglage et possibilité d'influencer à la main la valeur de réglage.

Selon le type de CPU, il sera mis en œuvre grâce au FB41 (pour les CPU 3xx-2DP) ou au SFB41 (pour les CPU sans interface Profibus).

Il propose les fonctions partielles suivantes :

- *Branche de consigne*
- *Branche de mesure*
- *Formation du signal d'erreur*
- *Algorithme PID*
- *Traitement de la valeur de réglage manuelle*
- *Traitement de la valeur de réglage*
- *Action anticipatrice*

Etats de fonctionnement :

Démarrage et redémarrage.

Le bloc FB 41 « CONT_C » dispose d'un sous-programme de démarrage qui est exécuté quand le paramètre d'entrée COM_RST = TRUE. A la mise en route, l'intégrateur est positionné de façon interne sur la valeur d'initialisation I_ITVAL. En cas d'appel dans un niveau d'alarme d'horloge, il continue à travailler à partir de cette valeur.

Toutes les autres sorties sont positionnées sur leurs valeurs par défaut.

Informations d'erreur :

Le bloc ne procède pas à un contrôle interne d'erreur. Le mot indicateur d'erreur RET_VAL n'est pas employé.

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------



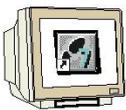
Mise en œuvre d'un régulateur PID avec Step 7 grâce au (S)FB41 "CONT_C"

La programmation d'un SIMATIC S7-300 en tant que régulateur PID se fait avec le logiciel STEP 7, outil unique de programmation de tous vos types d'application. Ici n'est mentionné que l'essentiel, pour plus d'informations vous pouvez vous référer à la documentation spécifique à STEP7.

Le paramétrage du régulateur PID se fait grâce à l'outil : « **Paramétrage de la régulation PID** », les paramètres choisis seront sauvegardés dans le DB d'instance local associé à l'appel du (S)FB 41.

Ceci est réalisé de la manière suivante :

3.2 Exemple de programmation



L'utilisateur doit suivre les étapes suivantes pour la réalisation de la configuration matérielle de l'automate, l'écriture du Programme S7, le réglage du régulateur PID ainsi que son chargement et son exécution dans la CPU :

1. Ouvrir SIMATIC Manager



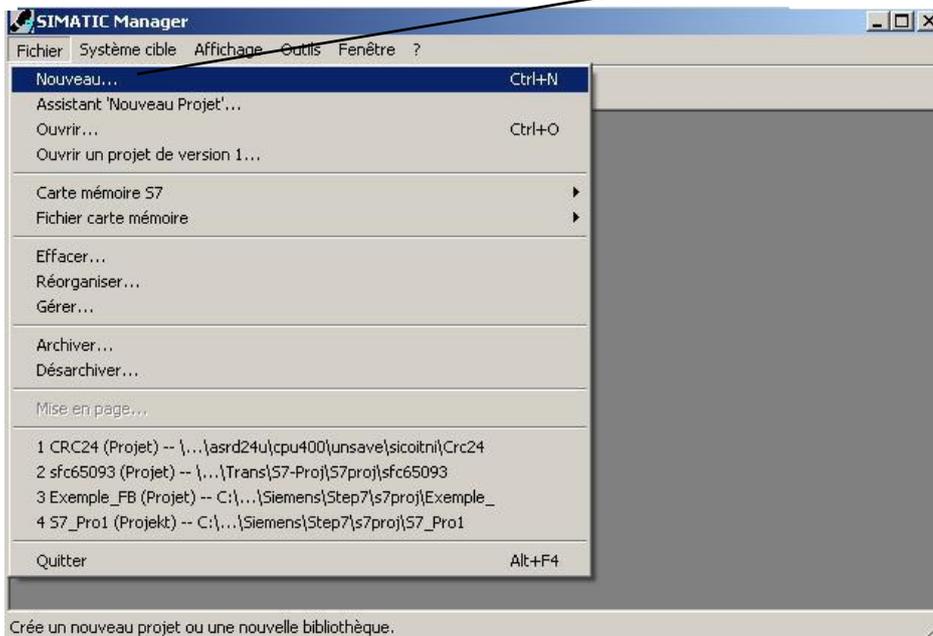
SIMATIC Manager

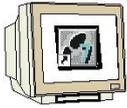
Cliquer sur l'icône 'SIMATIC Manager' !



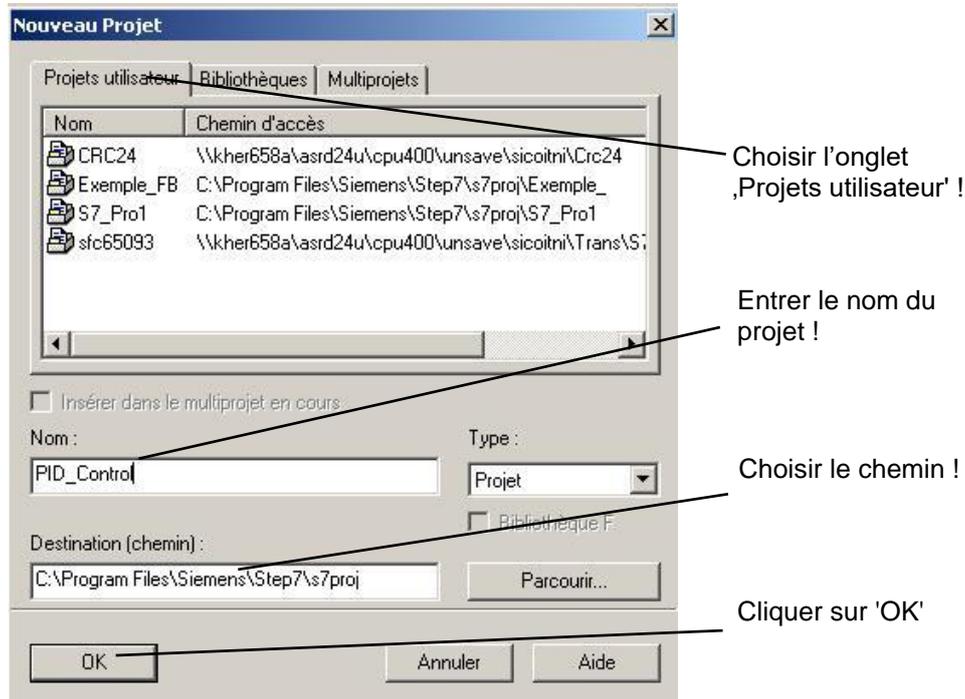
2. Créer un nouveau Projet (→ Fichier → Nouveau)

Cliquer sur 'Nouveau' !

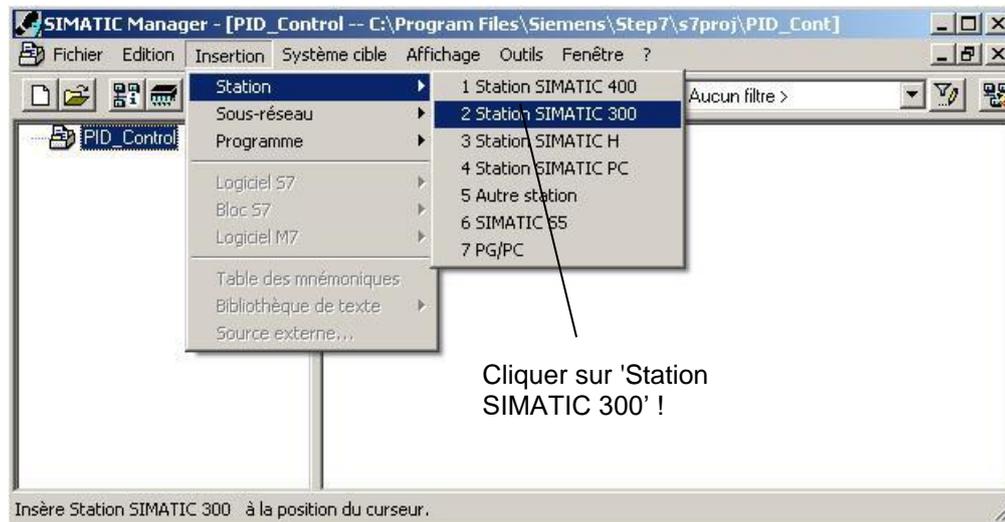




3. Créer un nouveau projet, donner le chemin de sauvegarde ainsi que le nom du projet (→ Projets utilisateur → PID_Control → OK)



4. Insérer un objet de type 'Station SIMATIC 300' (→ Insertion → Station → Station SIMATIC 300)

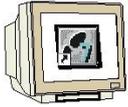


Pré-requis

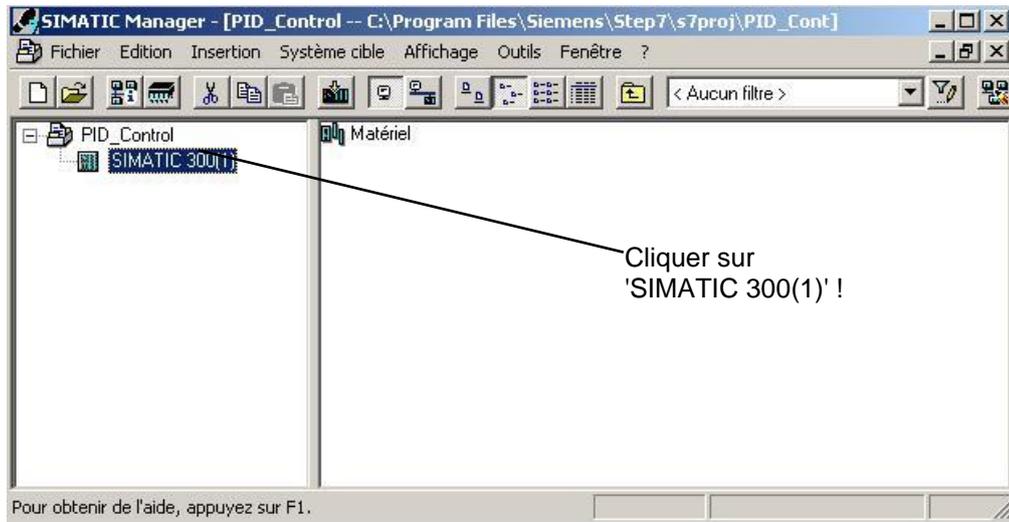
Analyse du problème

Bloc de régulation (S)FB41

Mise en œuvre



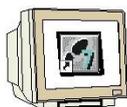
5. Sélectionner ,SIMATIC 300 (1)'



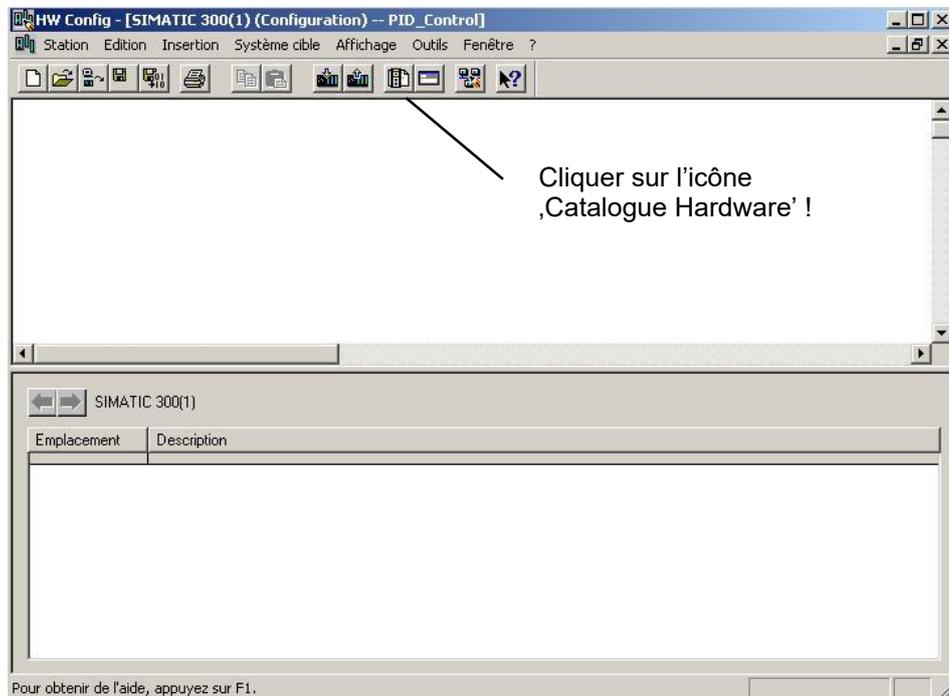
6. Ouvrir le programme de configuration matérielle (→ Edition → Ouvrir un objet)



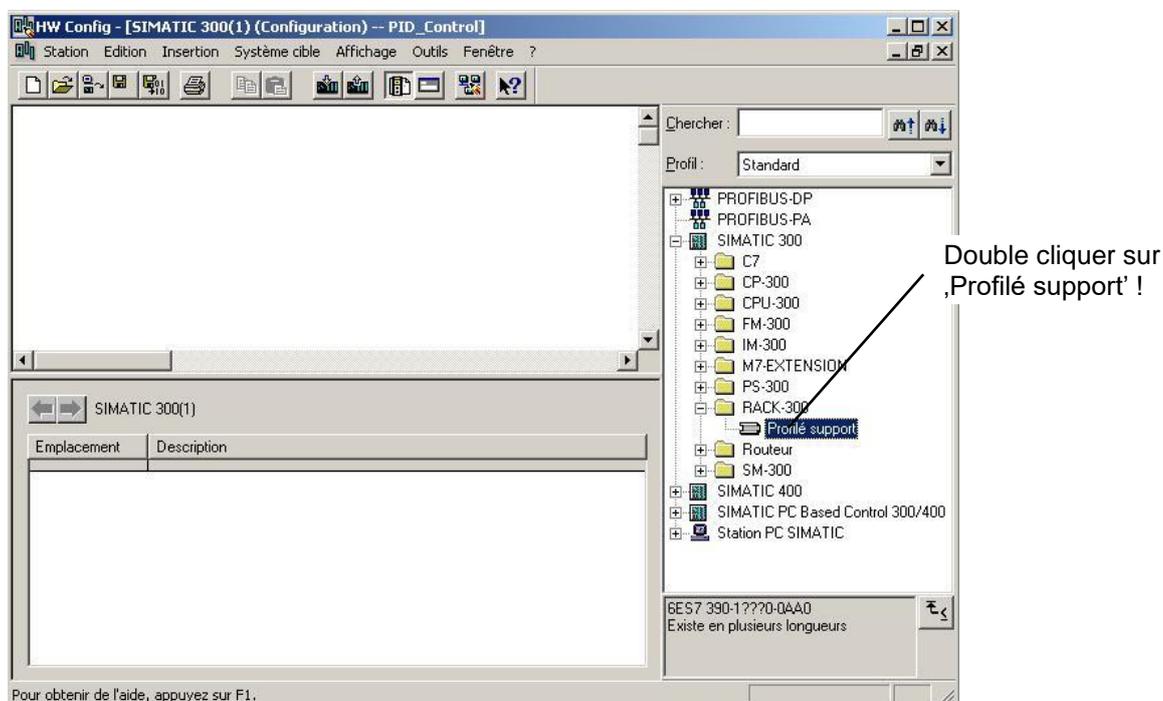
Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------



- Ouvrir le catalogue Hardware. Il se compose des répertoires suivants :
 - PROFIBUS-DP, SIMATIC 300, SIMATIC 400 et SIMATIC PC Based Control,
 - ainsi que tous les supports et composants modules d'interface nécessaires à la configuration de vos installations.

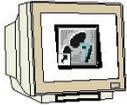


- Insérer un support profilé (→ SIMATIC 300 → RACK-300 → Profilé Support).



Il apparaît alors automatiquement un tableau pour la configuration du Rack 0.

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------



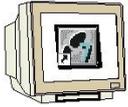
- Choisir les composants présents sur votre rack dans le catalogue Hardware et les insérer dans le tableau de configuration. Pour se faire, il suffit de cliquer sur l'élément voulu et de le faire glisser à la place souhaitée dans le tableau.

Emplacement	Module	Référence	Fi..	A...	A...	A...	C...
1	PS 307 2A	6ES7 307-1BA00					
2	CPU 314	6ES7 314-1AF10	V2.0	2			
3							
4	DI16xDC24V	6ES7 321-1BH82			0...1		
5	DO16xDC24V/0.5A	6ES7 322-1BH81				4...5	
6	AI4/AO2x8/8Bit	6ES7 334-0CE01			288...	288...	
7							
8							



Remarque ! La place n°3 du rack est réservée pour certains composants et reste donc vide pour notre exemple.

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

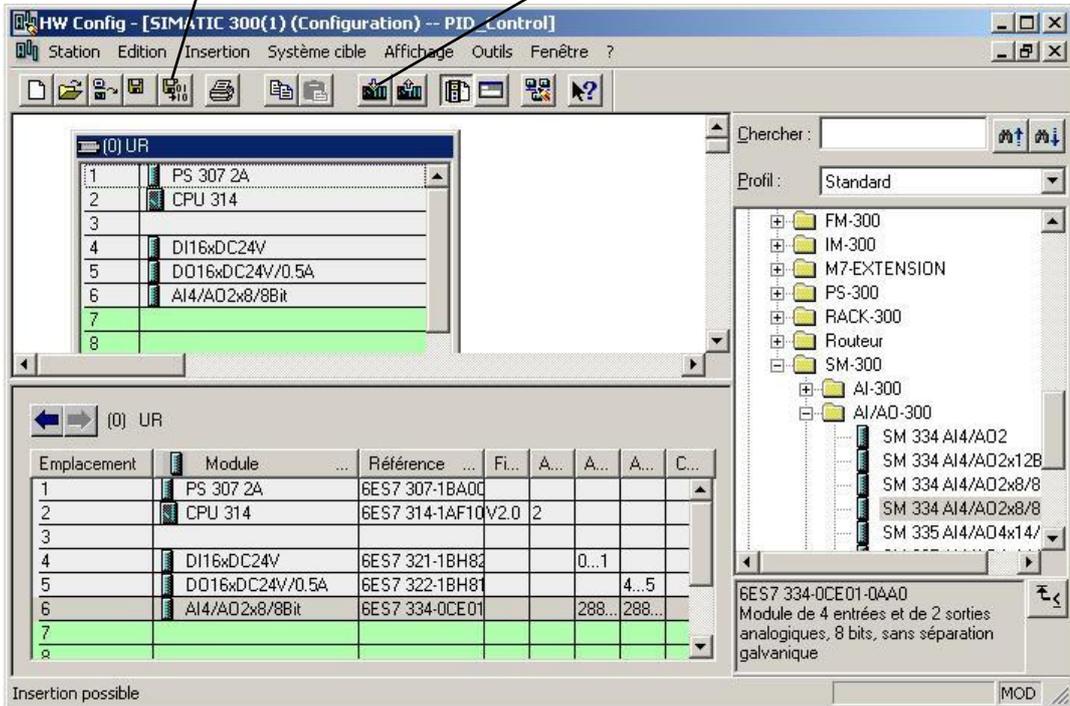


11. Noter les adresses des éléments d'entrées / sorties. Celles-ci sont attribuées automatiquement en fonction de la place sur le rack. Il faut auparavant choisir une vue détaillée dans le menu 'Affichage'. (→ Affichage → Détail).

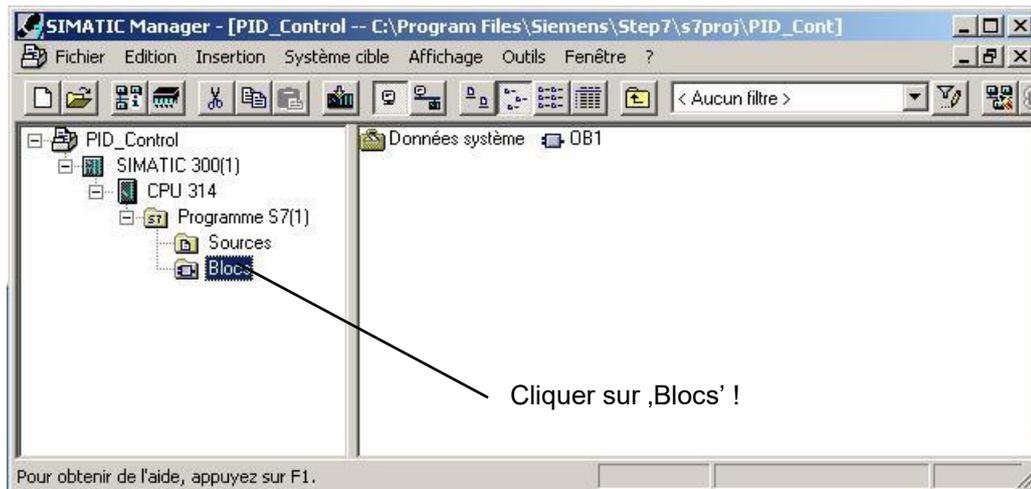
Sauvegarder et charger la configuration matérielle dans l'automate (le bouton de commande d'état doit être sur la position 'Stop' !)

Cliquer sur l'icône ,compiler et sauvegarder' !

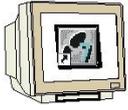
Cliquer sur l'icône ,charger' !



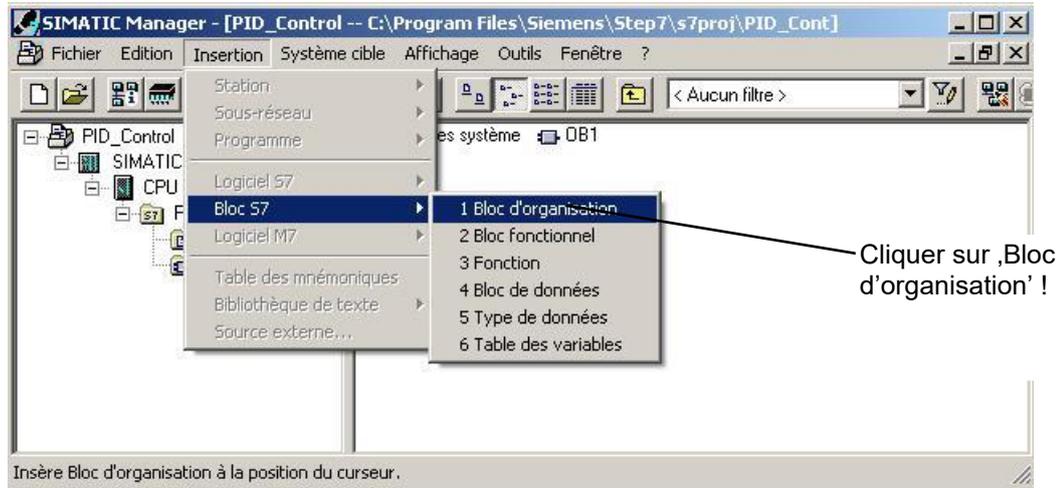
12. Sélectionner le dossier Blocs dans SIMATIC Manager



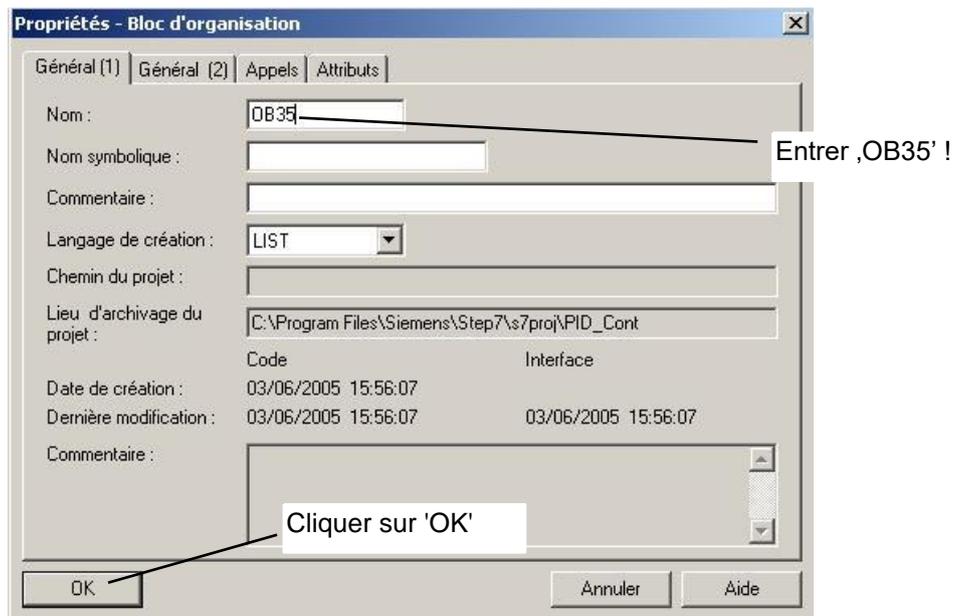
Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------



13. Insérer un bloc d'organisation (→ Insertion → Bloc S7 → Bloc d'organisation)

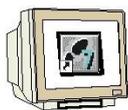


14. Le nommer OB35 (→ OB35 → OK)

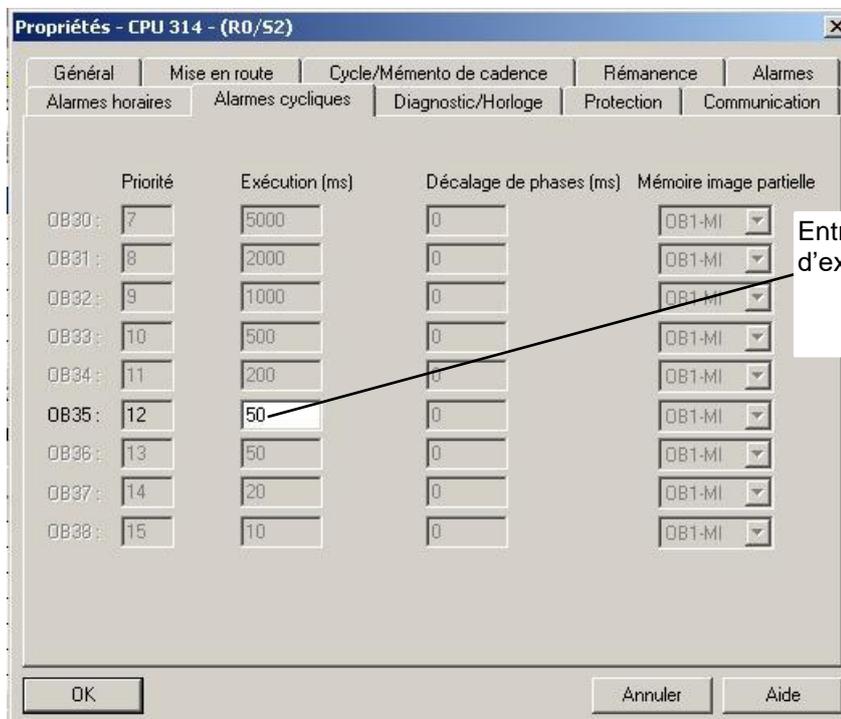


Remarque ! L'OB35 est un bloc dit d',alarme cyclique', son utilisation assure une fréquence d'appel constante du bloc SFB41 (régulateur PID). Ceci est primordial pour que le régulateur puisse être optimisé grâce au réglage de ses paramètres KP, TN et TV. Ceci serait impossible si l'on utilisait l'OB1 qui offre une fréquence d'appel incertaine...

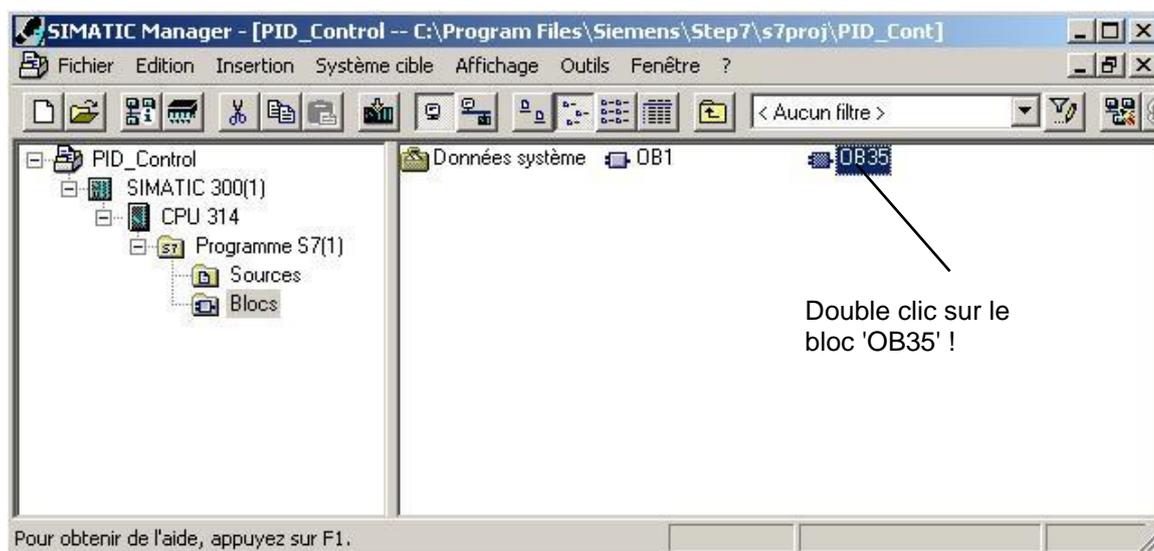
Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------



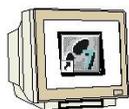
15. Dans le programme de configuration matérielle, il est possible de choisir la fréquence d'exécution de l'OB35 en allant dans les propriétés de la CPU sous l'onglet 'Alarmes cycliques'. Ce temps ne doit pas être trop court. On doit s'assurer que le reste du programme et notamment l'OB1 ait le temps de s'exécuter entre deux appels de l'OB35.



16. Ouvrir l'OB35 à partir de Simatic Manager (→ OB35)



Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

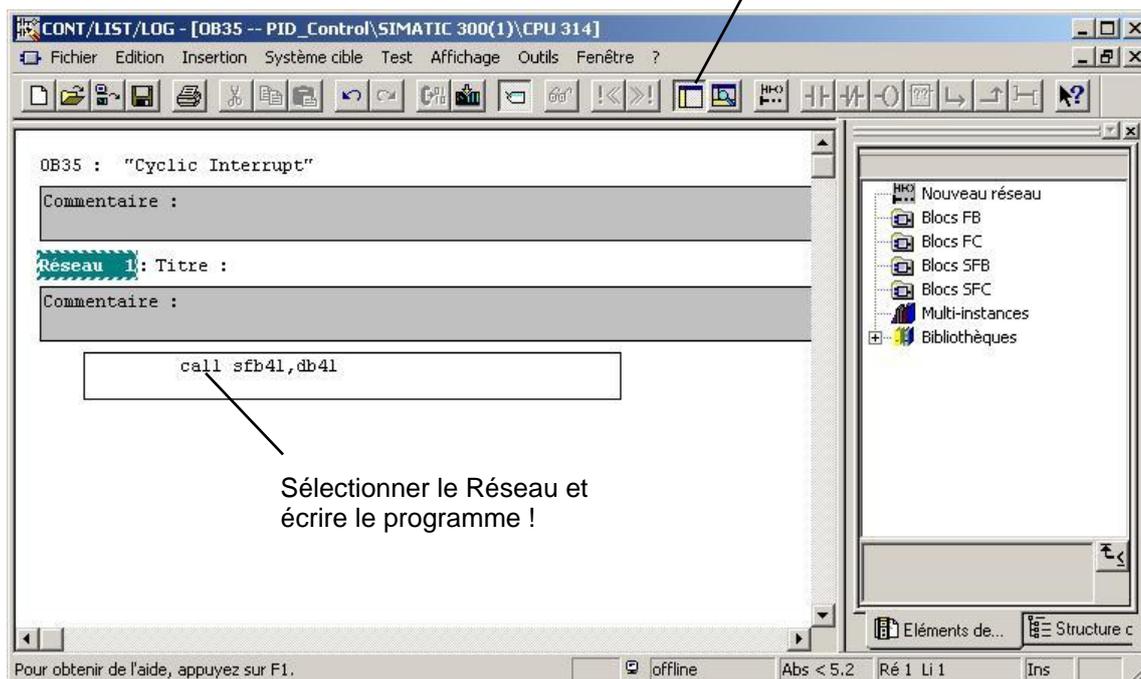


16. L'éditeur 'CONT/LIST/LOG' vous permet de modifier vos programmes STEP7.
L'OB35 avec un premier réseau a été ouvert.
Un programme STEP 7 peut être découpé en réseau. Un nouveau réseau est créé en cliquant sur l'icône correspondante. Une fois le réseau créé, il suffit de le sélectionner pour pouvoir taper le code du programme.



Remarque ! Il est possible d'insérer des commentaires en débutant la ligne par les caractères '//.

Cliquer pour créer un nouveau réseau !



Le réseau Call SFB41,DB41 appelle le bloc 'Régulateur PID' SFB41 en l'associant à un DB d'instance local.

18. Créer le DB d'instance.(→ Oui)



Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	----------------------



19. La valeur souhaitée, la valeur réelle et la variable d'ajustement doivent être affectées !

```

OB35 : "Cyclic Interrupt"
Commentaire :

Réseau 1 : Titre :
Commentaire :

CALL "CONT_C" , DB41
COM_RST :=
MAN_ON :=
PVPER_ON:=
P_SEL :=
I_SEL :=
INT_HOLD:=
I_ITL_ON:=
D_SEL :=
CYCLE :=T#50MS
SP_INT :=MD40
PV_IN :=
PV_PER :=PEW130
MAN :=
GAIN :=
TI :=
TD :=
TM_LAG :=
DEADB W :=
LMN_HLM :=
LMN_LLM :=
PV_FAC :=
PV_OFF :=
LMN_FAC :=
LMN_OFF :=
I_ITLVAL:=
DISV :=
LMN :=
LMN_PER :=PAM128
QLMN_HLM:=
QLMN_LLM:=
LMN_P :=
LMN_I :=
LMN_D :=
PV :=
ER :=
    
```

CYCLE :
Temps entre 2 appels du bloc. Dans notre cas, la fréquence d'appel de l'OB35.

SP INT:
Valeur souhaitée (consigne) donnée à partir d'une entrée analogique. Celle-ci doit être au format REAL (voir NW 2) !

PV PER
Valeur réelle mesurée , directement raccordée à une entrée analogique.

LMN PER:
Variable d'ajustement couplée à une sortie analogique.

```

PV_FAC :=
PV_OFF :=
LMN_FAC :=
LMN_OFF :=
I_ITLVAL:=
DISV :=
LMN :=
LMN_PER :=PAM128
QLMN_HLM:=
QLMN_LLM:=
LMN_P :=
LMN_I :=
LMN_D :=
PV :=
ER :=

Réseau 2 : Titre :
Commentaire :

L PEW 128
ITD
DTR
L 2.764800e+004
/R
L 1.000000e+002
*R
T MD 40
    
```

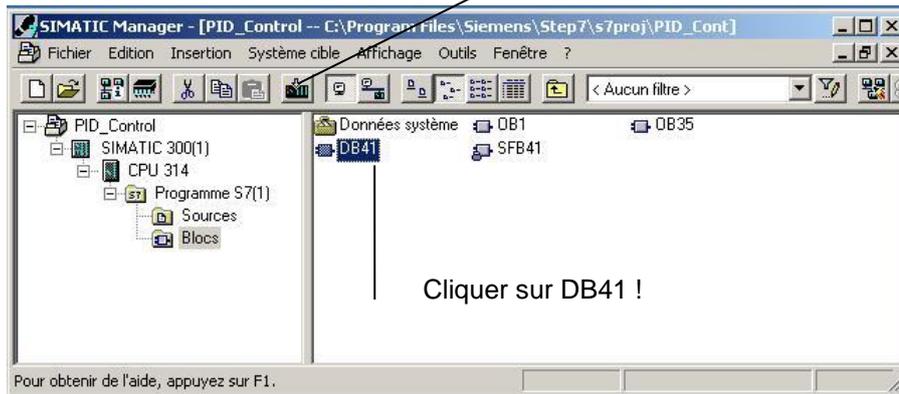
Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

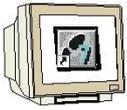


20. Sauvegarder le bloc d'organisation OB35 et le charger dans la CPU (L'interrupteur de commande d'état de la CPU doit être en position 'Stop')

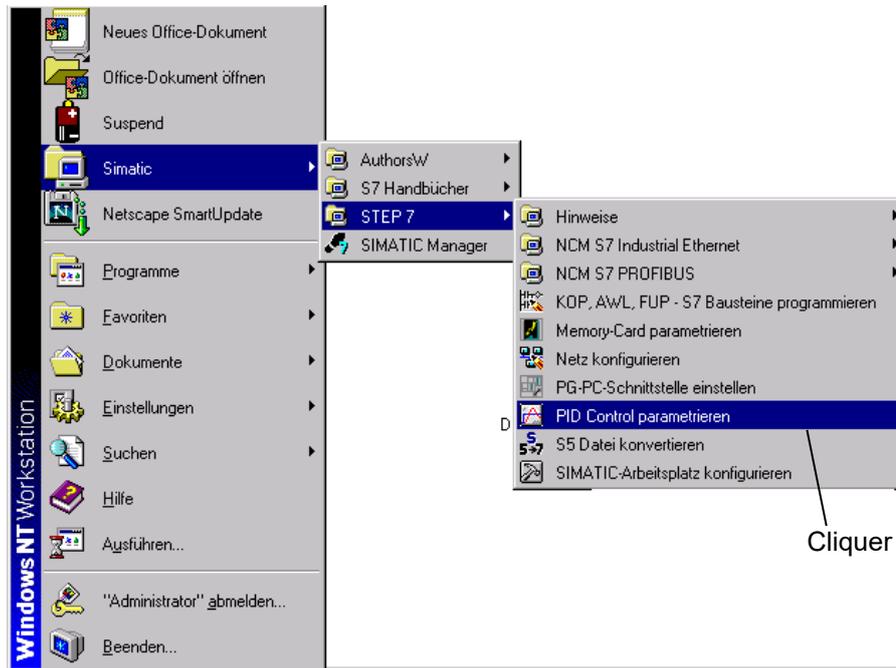
21. Sélectionner le bloc de données DB41 et le charger dans la CPU. (L'interrupteur de commande d'état de la CPU doit être en position 'Stop')

Cliquer sur l'icône
,charger' dans l'AP !





22. Ouvrir le programme 'Régulation PID' (→ Start → Simatic → STEP 7 → Régulation PID).

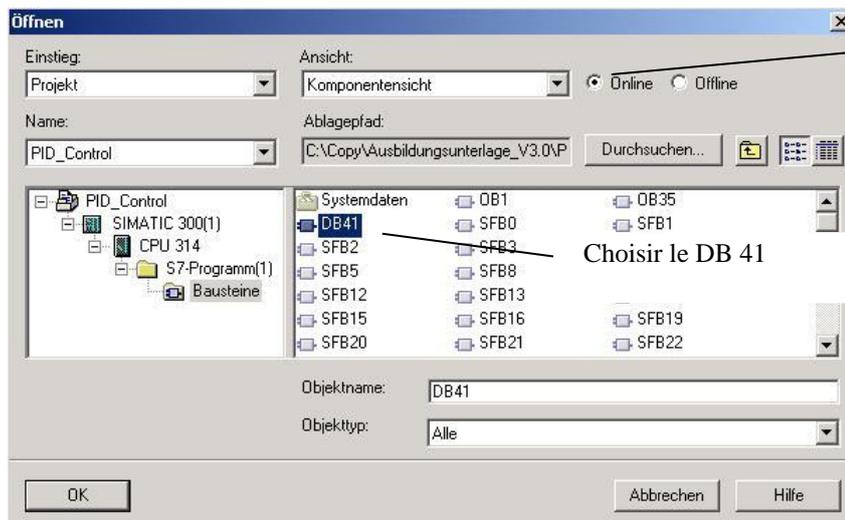
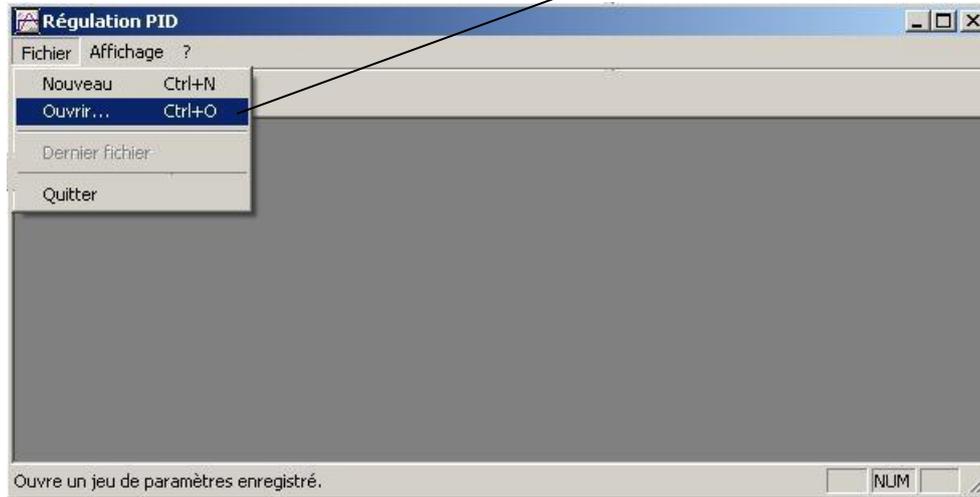


Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------



23. Ouvrir le bloc de données (→ Fichier → Ouvrir → **Online**
 → Choisir le bloc de données Ex.: DB41 → OK).

Cliquer sur 'Ouvrir' !

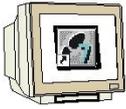


Accéder au bloc en mode 'Online'

Choisir le DB 41

Cliquer sur 'OK' !

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

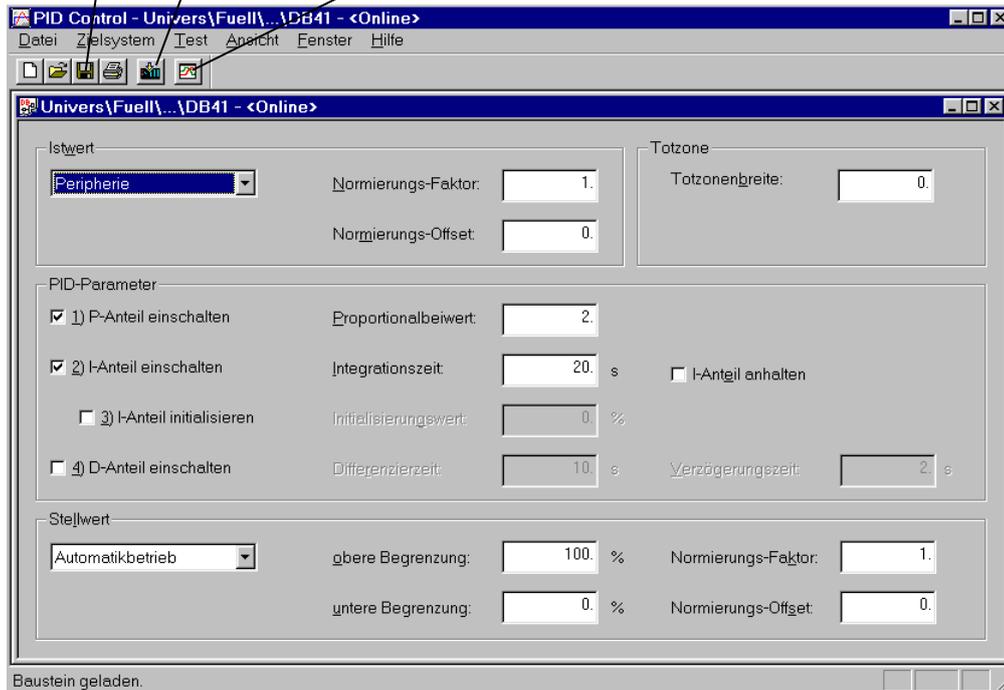


24. Le régulateur PID peut être paramétré avec le programme **,'Régulation PID'** .
 Le bloc de données DB sera ensuite sauvé (→ Sauvegarder) et chargé dans la CPU (→ Charger).
 C'est alors que peut intervenir la mise en route de la représentation graphique pour pouvoir observer le comportement de la régulation.

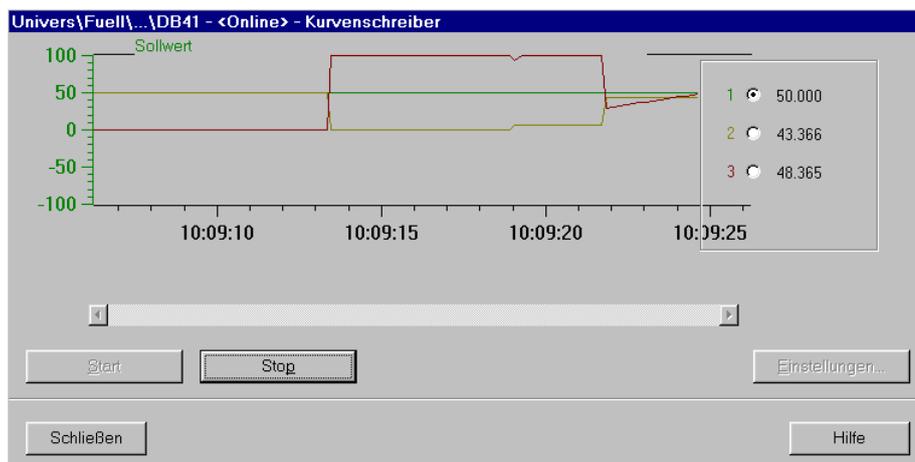
Cliquer sur ,Sauvegarder' !

Cliquer sur ,Charger' !

Mis en route de la représentation graphique !



25. Les courbes de la valeur souhaitée, la valeur réelle et la grandeur d'ajustement peuvent être visualisées grâce au graphique.



26. Mise en route du programme en positionnant l'interrupteur d'état de la CPU sur ,RUN'.

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	----------------------

4. Réglage de systèmes

4.1 Généralités

La mise au point du système sera réalisée selon le modèle d'un système de type PT2.

Approximation (T_u - T_g)

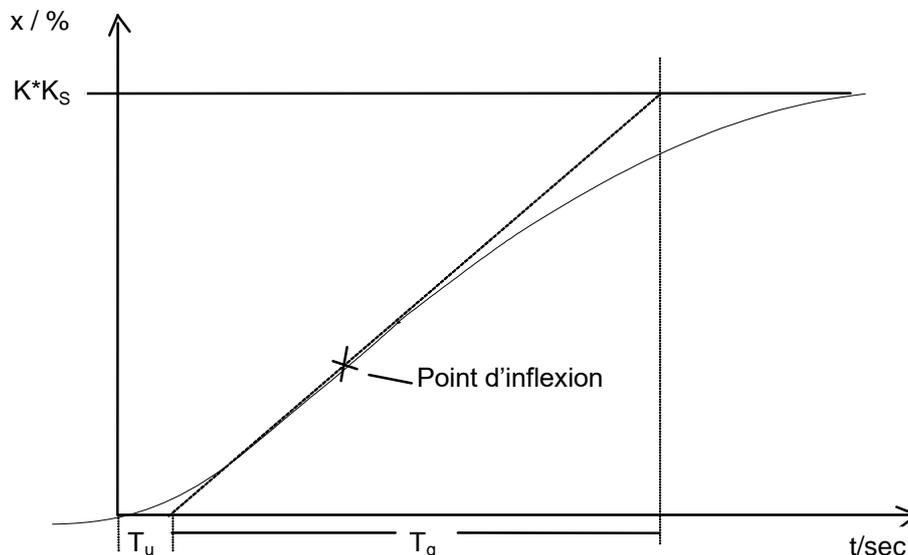
La base des méthodes de Ziegler- Nichols et de Chien, Hrones et Reswick est l'approximation (T_u - T_g). Les paramètres K_s (gain), T_u (retard pur), T_g (retard compensatoire) seront déterminés avec l'étude de la réponse indicielle du système en boucle ouverte. Les règles de mise au point décrites ici ont été établies de manière expérimentale avec un simulateur de signaux analogiques.

Les systèmes de type P-Tn peuvent, par approximation de n PT1 en cascade, de la même manière être réglés grâce à cette approximation (T_u - T_g).

Les paramètres utiles (K_s , T_u et T_g) sont déterminés, comme illustré sur le graphique de la réponse du système, à un échelon de consigne de hauteur K.

Il est nécessaire de mesurer la réponse du système jusqu'à ce que la valeur finale soit atteinte, afin de pouvoir déterminer avec précision le gain.

L'avantage principal de cette méthode est qu'elle s'applique également aux systèmes pour lesquels on ne dispose d'aucune description théorique de type analytique.



graphique : **Approximation (T_u - T_g)**

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

4.2 Mise au point du régulateur PI selon Ziegler- Nichols

Ziegler et Nichols posent les règles suivantes pour une régulation optimale d'un système P-Tn :

$$K_{PR} = 0,9 \frac{T_g}{K_S T_u}$$

$$T_N = 3,33 T_u$$

Les paramètres obtenus grâce à ces règles donnent en général de très bons résultats.

4.3 Mise au point du régulateur PI selon Chien, Hrones et Reswick

Cette méthode recommande différents paramètres selon le type de comportement et d'application recherché :

- Application de réponse à une perturbation :

Comportement apériodique
de courte durée

$$K_{PR} = 0,6 \frac{T_g}{K_S T_u}$$

$$T_N = 4 T_u$$

Comportement à 20% de
dépassement D

$$K_{PR} = 0,7 \frac{T_g}{K_S T_u}$$

$$T_N = 2,3 T_u$$

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

- Application de suivi de consigne :

Comportement apériodique
de courte durée

$$K_{PR} = 0,35 \frac{T_g}{K_S T_u}$$

$$T_N = 1,2 T_g$$

Comportement à 20% de
dépassement D

$$K_{PR} = 0,6 \frac{T_g}{K_S T_u}$$

$$T_N = T_g$$

4.4 Exemple d'application



Pour mesurer la réponse à un échelon de consigne, il faut effectuer quelques modifications dans les blocs OB35 et DB41. Suivre les étapes décrites :

Sauvegarder l'ancien projet sous un nouveau nom, et modifier comme suit l'interface d'appel du (S)FB41 :

1. Dans STEP7 donner directement une valeur de réglage.

Le code suivant générera une valeur de réglage manuelle en fonction de la position d'un bouton poussoir S1 relié à l'entrée E0.0.

```

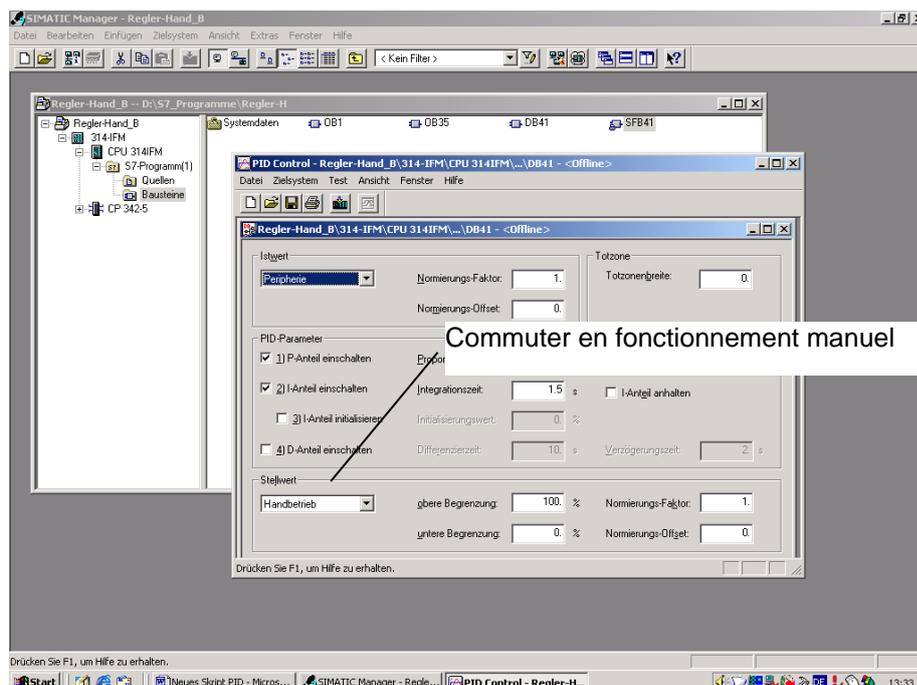
L      0.000000e+000      //Valeur 0% de type REAL 32-Bit
UN     E 0.0              //Négation de S1 (E 0.0)
SPB    M001              //Saute au repère M0001 si RLG = 1
L      9.000000e+001      // Valeur 90% de type REAL 32-Bit
M001:  T    MD 30         //Transfère la valeur dans le memento MD 30
    
```

La valeur de réglage prend la valeur $y = 90\%$ lorsque le bouton S1 est en position marche et la valeur $y = 0\%$ lorsque le bouton est à l'arrêt. Un saut de 0 à 90% est simplement réalisé en faisant passer le bouton de l'état arrêt à marche. (S' il n'y a pas de risque d'oscillation du signal, on peut utiliser la valeur 100% au lieu de 90%)

L'attribution des valeurs analogiques externes et des valeurs de position, dans OB1, se présente comme suit :

```

MAN      :=      MD 30      //valeur de réglage manuel
PV_PER   :=      PEW 130    //sortie x
LMN_PER  :=      PAW 128    //valeur régulée y
    
```



Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

Solution programmée :

La valeur de réglage manuelle prendra la valeur 0 ou 90% selon le résultat du réseau 2

```

Réseau 1: Titre :
Commentaire :

CALL "CONT_C" , DB41
COM_RST :=
MAN_ON :=
PVPER_ON:=
P_SEL :=
I_SEL :=
INT_HOLD:=
I_ITL_ON:=
D_SEL :=
CYCLE :=
SP_INT :=
PV_IM :=
PV_PER :=PEW130
MAN :=MD20
GAIN :=
TI :=
TD :=
TM_LAG :=
DEADB_W :=
LMN_HLM :=
LMN_LLM :=
LMN_ILM :=
PV_FAC :=
PV_OFF :=
LMN_FAC :=
LMN_OFF :=
I_ITLVAL:=
DISV :=
LMN :=
LMN_PER :=PAM128
QLMN_HLM:=
QLMN_LLM:=
LMN_P :=
LMN_I :=
LMN_D :=
PV :=
ER :=
    
```

Réseau 3: Titre :
 Commentaire :

```

E124.0 --(EN) MOVE
1.000000e+002 --(IN) ENO
002 --(OUT) MD20
    
```

Réseau 3: Titre :
 Commentaire :

```

E124.0 --(EN) MOVE
0.000000e+000 --(IN) ENO
000 --(OUT) MD20
    
```

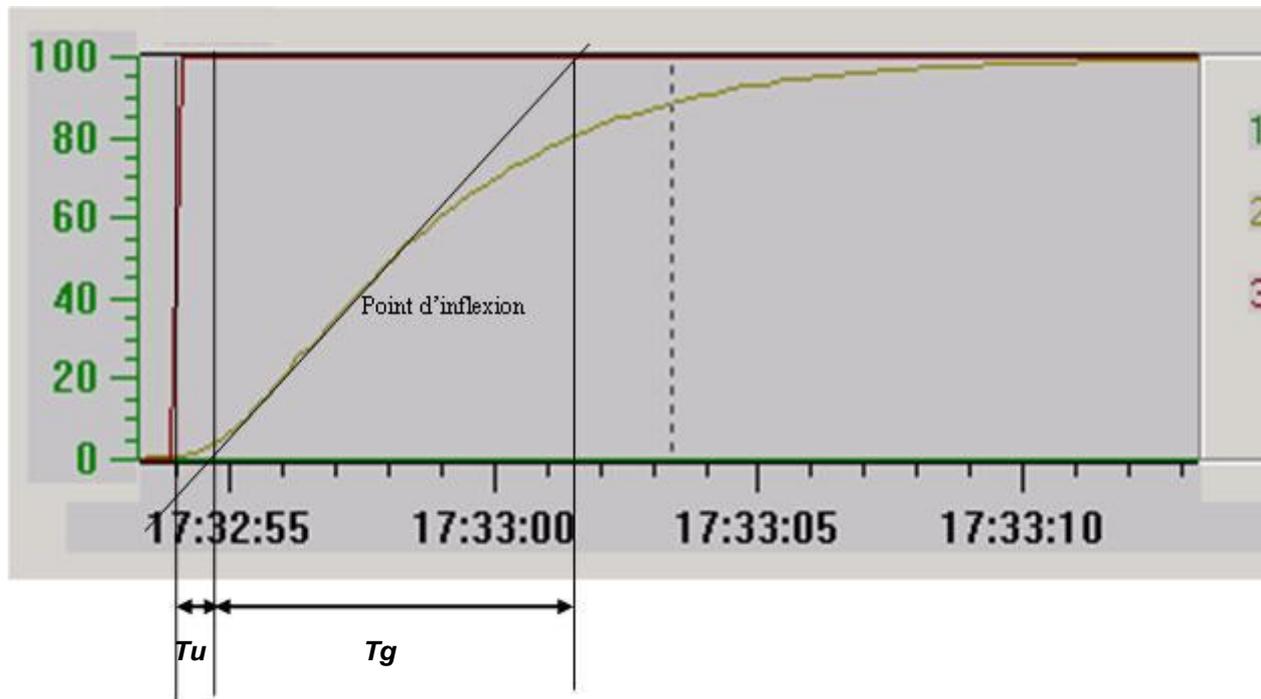
Pré-requis

Analyse du problème

Bloc de régulation (S)FB41

Mise en œuvre

La réponse indicielle du système, correspondant au saut de 0 à 100% de l'entrée, est ensuite tracée. Si le signal présente une instabilité dynamique (oscillations), il est conseillé d'utiliser une valeur maximale de 90%.



Réponse indicielle pour l'approximation (T_u - T_g)

Après avoir tracé la tangente au point d'inflexion, les différentes constantes peuvent être déterminées avec le graphique. Ainsi :

$$T_u = 0,7s$$

$$T_g = 7s$$

$$0,9 * K_s = 0,65$$

On obtient donc : $K_s = 0,72$ le rapport $T_g/K_s = 37,5s$.

Réglage du régulateur PI selon Ziegler-Nichols

A partir de l'approximation de (T_u - T_g) et des règles de Ziegler-Nichols, on en déduit les paramètres de réglage de notre régulateur :

$$K_{PR} = 12,5$$

$$T_N = 9,0s$$

Ces paramètres seront ensuite donnés au DB41.

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

4. Régulateur binaire tout ou rien

4.1 Enoncé du problème

La tension de sortie du système (par ex. un système de type PT2) doit être maintenue constante à l'aide d'un régulateur discontinu de type tout ou rien.

La tension de sortie de la carte analogique de L'AP est la valeur régulée (d'ajustement) y , celle-ci peut prendre la valeur „Activée“ (Tension = 10V) ou „Désactivée“ (Tension = 0V). La valeur régulée est reliée à l'entrée du procédé.

La tension de sortie du système sera notée x et est lue par l'intermédiaire d'un capteur relié à la carte d'entrée analogique de l'AP (PEW XY).

Le but de cette régulation est de maintenir la sortie X à un niveau de consigne w constant, en compensant l'entrée de perturbation z de type quelconque.

L'automate programmable S7-300 va être utilisé comme régulateur discontinu dans cet exemple. Il doit générer une valeur régulée y binaire en fonction de l'erreur (différence sortie/consigne, $e=w-x$),

Pour ne pas procéder à une activation / désactivation constante de la vanne V1 lorsque la sortie x a atteint la consigne w , il est nécessaire de définir une hystérésis de commutation (ou de réglage), zone dans laquelle l'état de la vanne ne devra pas changer. Cette hystérésis est comprise entre la limite supérieure X_o et une limite inférieure X_u . Nous appellerons $X_s = X_o - X_u$ „différence de commutation“.

X_s dépend de la valeur de consigne w , et est fixé à 25% de celle-ci :

$$X_s = w/4.$$

Un interrupteur S permet d'allumer ou d'éteindre le système de régulation.

La consigne w est donnée par un potentiomètre relié à la carte d'entrée analogique de l'AP.

Le programme est ensuite testé sur le procédé en question, pour vérifier le comportement de la sortie x et de la valeur régulée y dans le temps.

Tableau récapitulatif des entrées / sorties :

Variables d'entrée	Représentation symbolique	Type	Adresse
Interrupteur Marche/Arrêt	S	Marche : S = 1	E 0.0
Donneur de consigne	w	Potentiomètre	PEW 290
Sortie régulée	x	Entrée analogique:0..10 V	PEW 288
Variables de sortie			
Tension de sortie X	U1	Sortie numérique. 0 / 10 V	PAW 128 (10V)
Voyant de régulation		Allumé H0: = 1	A 4.1

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

Algorithme :

Régulateur éteint ? E0.0 = 0	
OUI	NON
Fermer la vanne V1; PAW 128 = 0V Initialisation des registres (effacer); MW 10 = 0 MW 12 = 0 MW 16 = 0 MW 18 = 0 MW 20 = 0 MW 22 = 0	Allumer le voyant de régulation A 4.1 = 1 Lire la valeur de sortie mesurée x PEW 130 Valeur dans MW 10 Lire la consigne w PEW 128 Valeur dans MW 12 Calcul de l'hystérésis de commutation $X_s = w/4$; MW 16 = SRW 2 (MW 12) Calcul de la demi hystérésis $X_1 = X_s/2$; MW 18 = SRW 1 (MW16) Calcul du seuil de commutation inférieur $X_u = w - X_1$; MW 20 = MW 12 - MW 18 Sortie x plus petite que le seuil de commutation inférieur ? $MW 10 < MW 20$
	NON
Tension de sortie activée; PAW 128 = 10 V	Calcul du seuil de commutation supérieur $X_o = w + X_1$; MW 22 = MW 12 + MW 18 Consigne X supérieure à X_o ? $MW 10 > MW 22$
	NON
	NON
Tension de sortie désactivée ; PAW 128=0V A 4.1 = 0	

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	----------------------

4.2 Programmation possible :

FCL : Titre :

Commentaire :

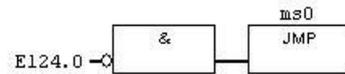
Réseau 1: Titre :

Commentaire :



Réseau 2: Titre :

Si E124.0 est à 0, saut à ms0 -> init des registres et remise à zéro de la sortie



Réseau 3: Lire la consigne et la sortie

Commentaire :

```

L   PEW  128
T   MW   12           // consigne
L   PEW  130
T   MW   10           // sortie mesurée
  
```

Réseau 4: Calcul de l'hystérésis de commutation

Xs = Xw : 4

```

L   MW   12           // consigne
SRD  2
T   MW   16           // Hysteresis Xs
  
```

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

Réseau 5 : Calcul de la demi hystérésis

$X_i = X_s : 2$

```

L   MW   16
L   2
/I
T   MW   18           // demi hystérésis
    
```

Réseau 6 : Calcul du seuil de commutation inférieur

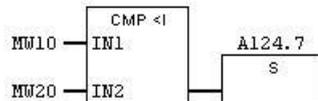
$X_u = X_w - X_i$

```

L   MW   12           // consigne
L   MW   18           // demi hystérésis
-I
T   MW   20           // seuil inférieur
    
```

Réseau 7 : Comparaison de la sortie avec le seuil de commutation inférieur

Si la sortie est inférieure au seuil de commutation, la sortie est mise à 1 -> ouverture de la vanne



Réseau 8 : Calcul du seuil de commutation supérieur

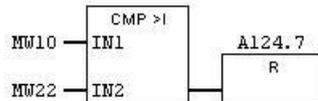
$X_o = X_w + X_i$

```

L   MW   18           // demi hystérésis
L   MW   12           // consigne
+I
T   MW   22           //seuil supérieur
    
```

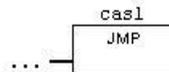
Réseau 9 : Titre :

Comparaison de la sortie mesurée avec le seuil de commutation supérieur. Si la comparaison est vraie, la sortie est mise à 0.



Réseau 10 : Titre :

Commentaire :



Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

Réseau 11 : Titre :

Commentaire :

```

ms0: L    0
      T    MW    10           // sortie mesurée
      T    MW    12
      T    MW    16
      T    MW    18
      T    MW    20           // seuil inférieur
      T    MW    22
      R    A    124.7
    
```

Réseau 12 : Titre :

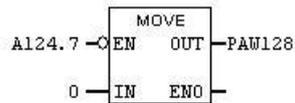
Commentaire :

```

cas1: NOP  0
    
```

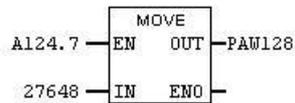
Réseau 13 : Titre :

Commentaire :



Réseau 14 : Titre :

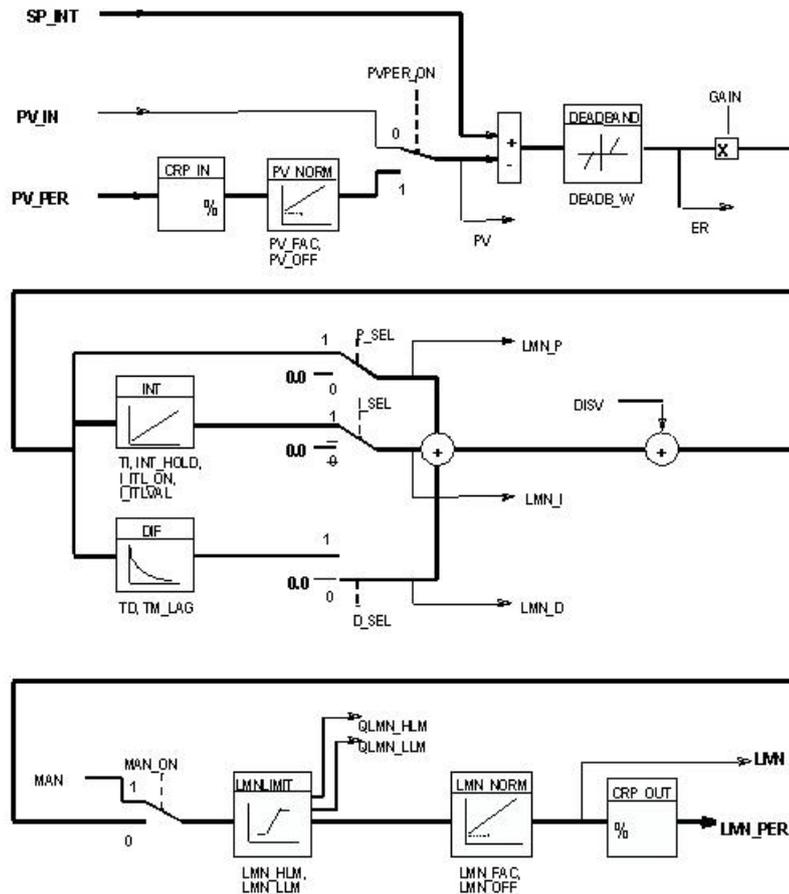
Commentaire :



Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

5. Annexes :

Schéma fonctionnel du bloc CONT_C (FB41) :



Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

Paramètres d'entrée du FB41 :



Paramètre	Type de données	Valeurs admises	Par défaut	Description
COM_RST	BOOL		FALSE	COMPLETE RESTART / Démarrage Le bloc renferme un sous-programme de démarrage qui est exécuté quand cette entrée est à 1.
MAN_ON	BOOL		TRUE	MANUAL VALUE ON / Activation du mode manuel Quand cette entrée est à 1, la boucle de régulation est interrompue. La valeur de réglage manuelle est sortie comme grandeur de réglage.
PVPER_ON	BOOL		FALSE	PROCESS VARIABLE PERIPHERY ON / Activation de la mesure de périphérie Pour que la mesure soit lue en périphérie, il faut relier l'entrée PV_PER à la périphérie et mettre à 1 l'entrée PVPER_ON.
P_SEL	BOOL		TRUE	PROPORTIONAL ACTION ON / Activation de l'action proportionnelle Dans l'algorithme PID, il est possible d'activer et de désactiver séparément chacune des actions. L'action P est active quand cette entrée est à 1.
I_SEL	BOOL		TRUE	INTEGRAL ACTION ON / Activation de l'action par intégration Dans l'algorithme PID, il est possible d'activer et de désactiver séparément chacune des actions. L'action I est active quand cette entrée est à 1.
INT_HOLD	BOOL		FALSE	INTEGRAL ACTION HOLD / Gel de l'action par intégration La sortie de l'intégrateur peut être gelée. Pour cela, il faut mettre à 1 cette entrée.
I_ITL_ON	BOOL		FALSE	INITIALIZATION OF THE INTEGRAL ACTION / Initialisation de l'action par intégration La sortie de l'intégrateur peut être positionnée sur la valeur initiale I_ITL_VAL. Pour cela, il faut mettre à 1 cette entrée.
D_SEL	BOOL		FALSE	DERIVATIVE ACTION ON / Activation de l'action par dérivation Dans l'algorithme PID, il est possible d'activer et de désactiver séparément chacune des actions. L'action D est active quand cette entrée est à 1.
CYCLE	TIME	≥ 1 ms	T#1s	SAMPLE TIME / Période d'échantillonnage Le temps s'écoulant entre les appels de bloc doit être constant. Il est indiqué au niveau de cette entrée.
SP_INT	REAL	-100,0 à 100,0 (%) ou grandeur physique ¹	0,0	INTERNAL SETPOINT / Consigne interne Cette entrée sert à introduire une valeur de consigne.
PV_IN	REAL	-100,0 à 100,0 (%) ou grandeur physique ¹	0,0	PROCESS VARIABLE IN / Mesure d'entrée Cette entrée permet de paramétrer une valeur de mise en service ou d'appliquer une mesure externe en virgule flottante.

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------

i

Paramètre	Type de données	Valeurs admises	Par défaut	Description
PV_PER	WORD		W#16#0000	PROCESS VARIABLE PERIPHERIE / Mesure de périphérie La mesure en format de périphérie est appliquée au régulateur par cette entrée.
MAN	REAL	-100,0 à 100,0 (%) ou grandeur physique ²	0,0	MANUAL VALUE / Valeur de réglage manuelle Cette entrée sert à introduire une valeur de réglage manuelle moyennant des fonctions de contrôle-commande.
GAIN	REAL		2,0	PROPORTIONAL GAIN / Coefficient d'action proportionnelle Cette entrée indique le gain du régulateur.
TI	TIME	>= CYCLE	T#20 s	RESET TIME / Temps d'intégration Cette entrée détermine la réponse temporelle de l'intégrateur.
TD	TIME	>= CYCLE	T#10 s	DERIVATIVE TIME / Temps de dérivation Cette entrée détermine la réponse temporelle de l'unité de dérivation.
TM_LAG	TIME	>= CYCLE/2	T#2 s	TIME LAG OF THE DERIVATE ACTION / Retard de l'action par dérivation L'algorithme de l'action D contient un retard qui peut être paramétré à cette entrée.
DEADB_W	REAL	>= 0,0 (%) ou grandeur physique ¹	0,0	DEAD BAND WIDTH / Largeur de zone morte Le signal d'erreur traverse une zone morte. Cette entrée détermine la taille de la zone morte.
LMN_HLM	REAL	LMN_ILLM à 100,0 (%) ou grandeur physique ²	100,0	MANIPULATED VALUE HIGH LIMIT / Limite supérieure de la valeur de réglage La valeur de réglage est toujours bornée à une limite supérieure et une limite inférieure. Cette entrée indique la limite supérieure.
LMN_LLM	REAL	-100,0 à LMN_HLM (%) ou grandeur physique ²	0,0	MANIPULATED VALUE LOW LIMIT / Limite inférieure de la valeur de réglage La valeur de réglage est toujours bornée à une limite supérieure et une limite inférieure. Cette entrée indique la limite inférieure.
PV_FAC	REAL		1,0	PROCESS VARIABLE FACTOR / Facteur de mesure Cette entrée est multipliée par la mesure. Elle sert à adapter l'étendue de valeur de mesure.
PV_OFF	REAL		0,0	PROCESS VARIABLE OFFSET / Décalage de mesure Cette entrée est ajoutée à la mesure. Elle sert à adapter l'étendue de valeur de mesure.
LMN_FAC	REAL		1,0	MANIPULATED VALUE FACTOR / Facteur de valeur de réglage Cette entrée est multipliée par la valeur de réglage. Elle sert à adapter l'étendue de réglage.

Pré-requis

Analyse du problème

Bloc de régulation (S)FB41

Mise en œuvre



Paramètre	Type de données	Valeurs admises	Par défaut	Description
LMN_OFF	REAL		0,0	MANIPULATED VALUE OFFSET / Décalage de valeur de réglage Cette entrée est ajoutée à la valeur de réglage. Elle sert à adapter l'étendue de réglage.
I_ITLVAL	REAL	-100,0 à 100,0 (%) ou grandeur physique ²	0,0	INITIALIZATION VALUE OF THE INTEGRAL ACTION / Valeur d'initialisation pour l'action par intégration La sortie de l'intégrateur peut être forcée par l'entrée I_ITL_ON. La valeur d'initialisation est appliquée à l'entrée I_ITLVAL.
DISV	REAL	-100,0 à 100,0 (%) ou grandeur physique ²	0,0	DISTURBANCE VARIABLE / Grandeur perturbatrice La grandeur perturbatrice est appliquée à cette entrée pour l'action anticipatrice.

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------



Paramètres de sortie du FB41 :

Paramètre	Type de données	Valeurs admises	Par défaut	Description
LMN	REAL		0,0	MANIPULATED VALUE / Valeur de réglage Cette sortie donne en virgule flottante la valeur de réglage agissant réellement.
LMN_PER	WORD		W#16#0000	MANIPULATED VALUE PERIPHERY / Valeur de réglage de périphérie Cette sortie fournit la valeur de réglage en format de périphérie.
QLMN_HLM	BOOL		FALSE	HIGH LIMIT OF MANIPULATED VALUE REACHED / Grandeur de réglage à la limite supérieure La valeur de réglage est toujours bornée à une limite supérieure et à une limite inférieure. Cette sortie signale le dépassement de la limite supérieure.
QLMN_LLM	BOOL		FALSE	LOW LIMIT OF MANIPULATED VALUE REACHED / Grandeur de réglage à la limite inférieure La valeur de réglage est toujours bornée à une limite supérieure et à une limite inférieure. Cette sortie signale le dépassement de la limite inférieure.
LMN_P	REAL		0,0	PROPORTIONALITY COMPONENT / Composante P Cette sortie contient la composante proportionnelle de la grandeur de réglage.
LMN_I	REAL		0,0	INTEGRAL COMPONENT / Composante I Cette sortie contient la composante intégrale de la grandeur de réglage.
LMN_D	REAL		0,0	DERIVATIVE COMPONENT / Composante D Cette sortie contient la composante différentielle de la grandeur de réglage.
PV	REAL		0,0	PROCESS VARIABLE / Mesure Cette sortie donne la mesure effective.
ER	REAL		0,0	ERROR SIGNAL / Signal d'erreur Cette sortie donne le signal d'erreur effectif.

Pré-requis	Analyse du problème	Bloc de régulation (S)FB41	Mise en œuvre
------------	---------------------	----------------------------	---------------