



**Rapport EI ST5**  
**Conception de la commande-contrôle d'un sous-système d'une**  
**centrale nucléaire**

Groupe 2 : Erwann Henon, Alexian Hélaine, Yann Boula de Mareuil,  
Adrien Ternois, Léo Marie, Mateo Weill

Novembre 2024

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Ingénierie des exigences</b>	<b>3</b>
2.1	Environnement du système . . . . .	3
2.2	Exigences à respecter . . . . .	3
2.3	Justification pour la commande des vannes associées . . . . .	3
2.4	Justification pour la Logique de Commande des Pompes . . . . .	4
2.5	Ouverture sur les exigences dépendantes du temps et les exigences probabilistes. . . . .	4
<b>3</b>	<b>Conception de la commande des vannes</b>	<b>6</b>
3.1	Régulation de la température . . . . .	6
3.2	Limitation du débit dans les échangeurs et les pompes . . . . .	6
3.3	Régulation du débit . . . . .	7
3.3.1	Configuration à 2 échangeurs . . . . .	8
3.3.2	Configuration à 1 échangeur . . . . .	9
3.4	Commande finale des vannes . . . . .	10
<b>4</b>	<b>Vérification et analyse de la conception</b>	<b>11</b>
4.1	Test de la régulation . . . . .	11
4.2	Guide d'exploitation . . . . .	12
4.2.1	Limites de charge . . . . .	12
<b>5</b>	<b>Commande des pompes</b>	<b>12</b>
5.1	Présentation des éléments de contrôle . . . . .	13
5.2	Conception du Programme d'Automatisation . . . . .	13
5.3	Bloc Fonctionnel PO_SWITCH . . . . .	13
5.4	Bloc Fonctionnel ST_1SM_BFE . . . . .	13
5.5	Schéma Fonctionnel . . . . .	14
5.6	Analyse des résultats . . . . .	14
5.7	Bilan de la commande des pompes . . . . .	15
<b>6</b>	<b>Conclusion</b>	<b>17</b>

# 1 Introduction

Ce rapport examine la conception et mise en œuvre d'un système contrôle-commande pour le Service de Refroidissement Intermédiaire (SRI) en centrale nucléaire. L'objectif principal est de concevoir un système de réfrigération fiable et adaptable aux exigences strictes de sûreté et de flexibilité énergétique. Notre équipe "Ingénierie systèmes et contrôle-commande" a été missionnée pour garantir que le modèle respecte les contraintes de disponibilité, de régulation de température et de débit, et qu'elle prenne en compte les exigences de sécurité imposées par les autorités.

Les pages qui suivent se proposent de rapporter les étapes significatives de notre démarche, comprenant la simulation de la maquette SRI, la régulation des deux niveaux de température et débit, l'implémentation du programme automate pour la gestion des pompes de secours. L'objet de ce rapport est de proposer une solution contrôle efficace et robuste et qui satisfasse le besoin d'exploitation à travers des conditions de fonctionnement différentes, tout en assurant l'opérationnel en environnement dangereux comme celui du nucléaire.

## 2 Ingénierie des exigences

### 2.1 Environnement du système

Le SRI (Service de Refroidissement Intermédiaire) est un système de réfrigération de l'eau pour les équipements auxiliaires situés principalement dans la salle des machines d'une centrale nucléaire. Il assure la circulation d'eau en boucle fermée à travers des échangeurs thermiques, en utilisant de l'eau brute provenant d'une source froide comme moyen de refroidissement.

Les principales parties prenantes sont :

- **le propriétaire de la centrale** : il est responsable de la disponibilité et du fonctionnement optimal de la centrale, et impose des exigences de disponibilité et de maintenance pour garantir une opération continue.
- **les autorités locales et régulateurs environnementaux** : ces derniers veillent à la conformité environnementale, notamment en ce qui concerne les rejets thermiques dans l'environnement. Ils imposent des restrictions sur les températures de l'eau rejetée pour limiter l'impact écologique.
- **les opérateurs de la centrale** : ils utilisent et surveillent le SRI et doivent gérer les opérations quotidiennes ainsi que les procédures d'arrêt et de maintenance. Ils interviennent également en cas de panne ou de mal fonctionnement du système.

### 2.2 Exigences à respecter

Les exigences du système SRI sont divisées en deux catégories. Il y a tout d'abord les **exigences liées à la fonctionnalité et la disponibilité** :

- **Température de consigne** : La température de l'eau du SRI en sortie doit être maintenue à 17 °C, avec un écartement à 16 °C et une limite supérieure à 30 °C pour minimiser les variations thermiques et éviter les dommages aux équipements.
- **Débit et pression des pompes** : Le système doit maintenir un débit constant avec une pression de 2,44 bars et éviter la cavitation en ne descendant pas sous 700 m<sup>3</sup>/h.
- **Temps de disponibilité** : Le système SRI doit avoir un MTBF supérieur à 10 000 heures et un MTTR inférieur à 100 heures, correspondant à une disponibilité de plus de 99%.

Et à ces exigences s'ajoute une **exigence environnementale** :

- **Contrôle de la température de rejet** : La différence de température entre l'eau captée et l'eau rejetée doit être limitée à 7 °C en conditions normales et à 5 °C en période de canicule, lorsque la température de la source froide dépasse 22 °C, pour éviter les impacts thermiques excessifs.

### 2.3 Justification pour la commande des vannes associées

Les exigences suivantes sont essentielles pour garantir un contrôle efficace et fiable.

- **Régulation de la température (17 °C en sortie avec écrêttements à 16 °C et 30 °C) :** La température de l'eau en sortie des échangeurs doit être maintenue à une valeur spécifique pour éviter des variations qui pourraient endommager les équipements de refroidissement et les systèmes auxiliaires. En réglant précisément l'ouverture des vannes de série, la commande assure que la température est stabilisée à 17 °C. L'écrêttement aux bornes de 16 °C et 30 °C protège le circuit contre les surchauffes et le sous-refroidissement, ce qui est essentiel pour la durabilité des équipements et la sécurité de la centrale.
- **Limitation de la vitesse du fluide dans les échangeurs (6 m/s maximum) :** Les échangeurs, en raison de leur contact avec l'eau brute, risquent de s'encrasser, ce qui affecterait l'efficacité du transfert thermique. En contrôlant l'ouverture des vannes, il est possible de limiter la vitesse d'écoulement dans les échangeurs, réduisant ainsi le risque d'encrassement et prolongeant la durée de vie des équipements. Cette exigence de vitesse garantit un fonctionnement fiable à long terme.
- **Maintien d'un débit constant dans le circuit :** Le débit doit rester stable pour éviter les variations brusques, qui peuvent entraîner des fluctuations de température indésirables et nuire au bon fonctionnement des pompes. La vanne bypass compense les variations de débit en agissant de manière complémentaire avec les vannes de série, assurant ainsi un débit constant, indépendamment de la configuration de refroidissement (1 ou 2 échangeurs). Cela permet de protéger les composants en assurant un flux de refroidissement continu.
- **Positions de sécurité des vannes (ouvertes pour les vannes série, fermée pour la vanne bypass en cas de défaillance) :** La sûreté passive est un principe de sécurité qui consiste à configurer les vannes pour qu'elles se positionnent automatiquement en cas de panne (sans intervention humaine). En cas de manque d'air, les vannes de série s'ouvrent pour sur-refroidir, tandis que la vanne bypass se ferme. Cette configuration réduit le risque de surchauffe, protégeant ainsi les équipements critiques et répondant aux exigences de sécurité essentielles dans un environnement nucléaire.

## 2.4 Justification pour la Logique de Commande des Pompes

Les exigences suivantes répondent aux impératifs de rapidité et de fiabilité :

- **Débit minimal pour éviter la cavitation ( $700 \text{ m}^3/\text{h} \rightarrow 194.4 \text{ kg/s}$ ) :** La cavitation, causée par un débit insuffisant, peut gravement endommager les pompes et nuire à l'efficacité du circuit de refroidissement. La logique de commande doit surveiller en continu le débit et s'assurer qu'il ne descend pas en dessous du seuil de  $700 \text{ m}^3/\text{h}$ .
- **Exigences de disponibilité (MTBF > 10 000 h et MTTR < 100 h) :** Les exigences de disponibilité reflètent le besoin de maintenir le SRI en fonctionnement continu pour assurer la stabilité de la centrale. En incluant une pompe de secours et en automatisant la reprise en normal-secours, le système peut répondre rapidement aux pannes, maintenant un taux de disponibilité élevé (> 99 %). Cette redondance est cruciale pour minimiser les interruptions et éviter toute perte de refroidissement.
- **Délai de grâce et temporisation pour la reconfiguration :** Ces temporisations sont nécessaires pour éviter les reconfigurations instantanées, qui peuvent causer des perturbations brusques dans le système. En attendant 5 secondes pour confirmer un défaut et 7 secondes supplémentaires avant de démarrer une pompe de secours, le système filtre les défauts transitoires et effectue une transition en douceur. Cette approche permet d'éviter des démarrages et arrêts de pompes fréquents, ce qui allonge leur durée de vie et garantit un fonctionnement plus stable du circuit.

Nouvelles exigences dans le modèle Modelica : req\_temp et req\_deb\_pompe

## 2.5 Ouverture sur les exigences dépendantes du temps et les exigences probabilistes.

Pour l'exigence de délai de grâce et de temporisation pour la reconfiguration, on peut créer une batterie de scénarii et les tester avec le code SysPro. Quand à l'exigence de disponibilité, voici une esquisse de démarche à suivre : On fait d'abord un inventaire des différentes défaillances potentielles et d'évaluer leur impact sur l'ensemble du système. Ensuite, on estime la fréquence de ces défaillances en s'appuyant sur des données historiques, permettant d'obtenir des valeurs réalistes et précises pour chaque type d'incident. On mesure/estime également les temps de réparation associés afin de calculer l'indisponibilité du système. Enfin, on réalise des simulations de Monte Carlo pour modéliser de manière probabiliste le comportement du système sur une période donnée. Ces simulations intègrent les fréquences de défaillance et les temps de réparation, offrant une vue d'ensemble sur la fiabilité du système et la probabilité de respecter les exigences d'indisponibilité ou d'arrêt maximal autorisé.

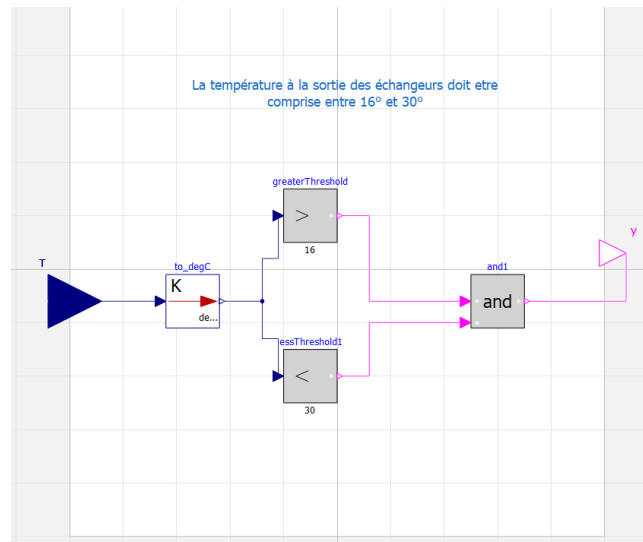


Figure 1: Exigence liée à la température

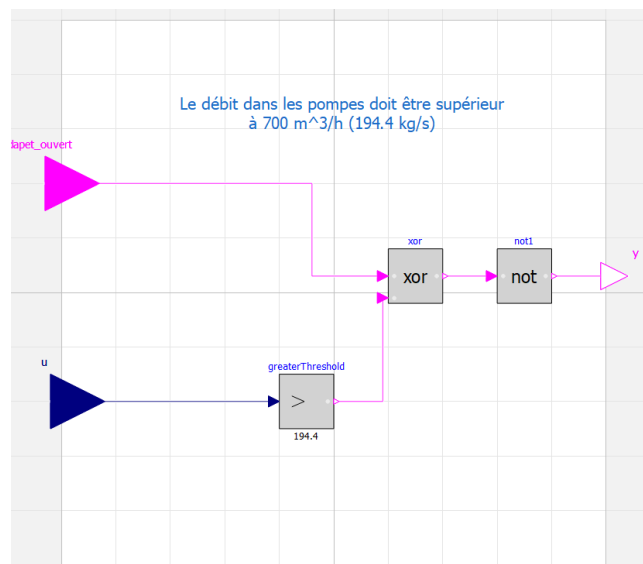
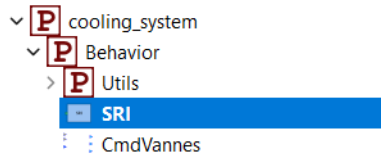


Figure 2: Exigence liée au débit dans les pompes

### 3 Conception de la commande des vannes

#### 3.1 Régulation de la température

On crée le modèle `CmdVannes` dans le package `Behavior`, puis on l'instance dans le modèle `SRI`



Les entrées de ce modèle sont la température, et par la suite l'ouverture ou non de chaque branche d'échangeur. Les sorties sont les ouvertures des vannes séries (et bypass par la suite)

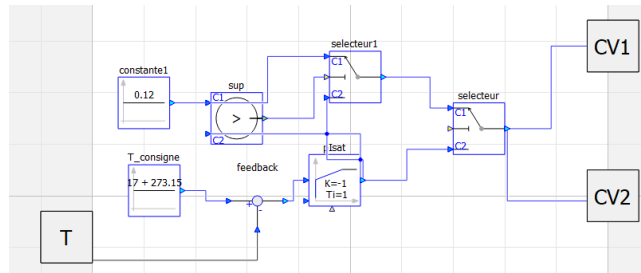


Figure 3: Première version du schéma bloc réalisant la régulation de la température pour les vannes séries

#### 3.2 Limitation du débit dans les échangeurs et les pompes

On maximise le débit (et donc la vitesse) dans les deux échangeurs en fermant la vanne bypass et en ouvrant les vannes séries suivant une rampe (ouverture progressive de 0 : complètement fermé à 1 : complètement ouvert). Dans cette situation, on remarque (figure 4) que la vitesse dans les échangeurs ne dépassent pas 6 m/s, ce qui veut dire qu'il n'y aura jamais de problème avec cette exigence si les deux échangeurs sont activés.

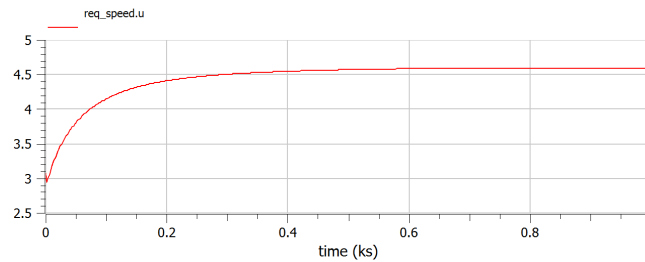


Figure 4: Vitesse du fluide pour une rampe de commande d'ouverture des vannes séries - vanne bypass fermée dans la configuration avec 2 échangeurs

Quand il n'y a plus qu'un seul échangeur, on remarque que la contrainte de vitesse n'est plus respectée (figure 5).

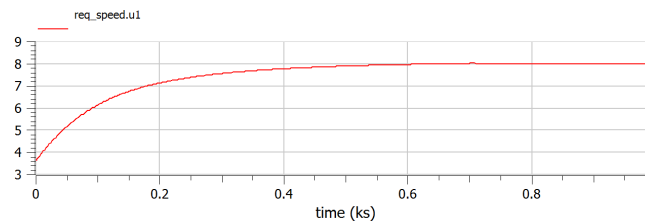


Figure 5: Vitesse du fluide pour une rampe de commande d'ouverture des vannes séries - vanne bypass fermée dans la configuration avec 1 échangeur

En appliquant une rampe au signal d'ouverture de la vanne bypass, on obtient une ouverture minimale de 0.161 dans le cas où un seul échangeur est actif (figure 6).

- $ouv\_bypass\_min = 0.16$

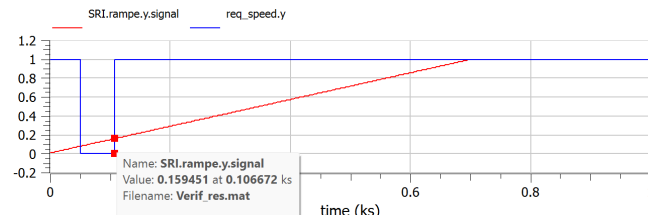


Figure 6: Determination de  $V_{bypass}^{min}$  dans la configuration à 1 échangeur

Avec 2 échangeur il n'y a pas de valeur minimale d'ouverture.

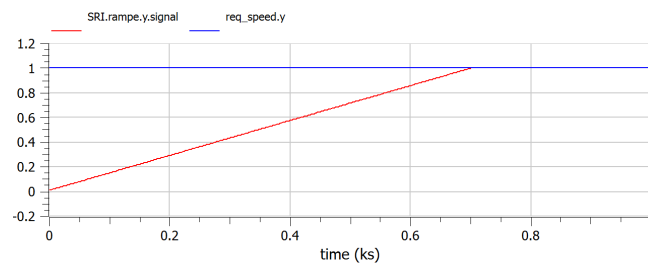


Figure 7: Determination de  $V_{bypass}^{min}$  dans la configuration à 2 échangeurs

De la même manière, dans le cas où la vanne bypass est complètement fermée, pour être sûr de respecter le minimum de débit dans les pompes, on obtient une ouverture minimal de 0.06 dans le cas 2 échangeurs et 0.12 dans le cas 1 échangeur.

- Pour 2 échangeurs :  $ouv\_serie\_min = 0.06$

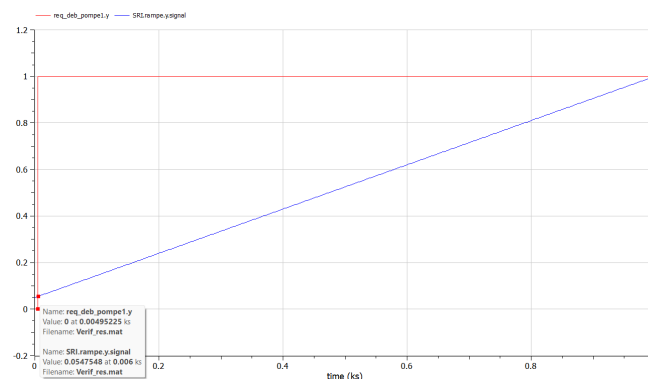


Figure 8: Determination de  $V_{serie}^{min}$  dans la configuration à 2 échangeurs

- Pour 1 échangeur :  $ouv\_serie\_min = 0.12$

### 3.3 Régulation du débit

Les vannes réglantes ont également pour rôle de réguler le débit à une valeur constante. Mais le circuit n'est pas instrumenté avec un capteur de débit, donc la régulation doit être en boucle ouverte. Cette fois-ci, la simulation ne nous sera pas d'un grand secours, nous allons devoir procéder analytiquement. L'objectif est donc

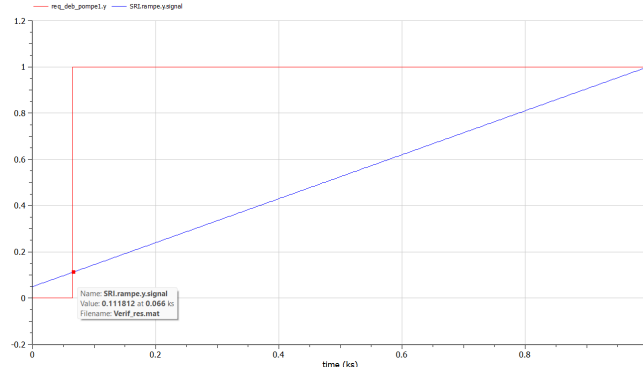


Figure 9: Determination de  $V_{serie}^{min}$  dans la configuration à 1 échangeur

de déterminer un calcul du signal d'ouverture de la vanne de bypass en fonction de celui des vannes en série, de sorte que le débit résultant de ces ouvertures soit une constante. L'ouverture d'une vanne est proportionnelle à son coefficient de débit (CV), qui, à pression fixée, est proportionnel au débit. La formule suivante permet d'exprimer le CV équivalent de l'ensemble des trois branches ( $CV_{tot}$ , qui doit rester constant) par rapport au CV des vannes ( $CV_s$  et  $CV_{by}$ ) et à celui des échangeurs ( $CV_e$ ) :

$$CV_{tot} = CV_{by} + conf \cdot \sqrt{\frac{CV_s^2 \cdot CV_e^2}{CV_s^2 + CV_e^2}} \quad (1)$$

où  $conf = 1$  pour la configuration à 1 échangeur et  $conf = 2$  pour la configuration à 2 échangeurs. D'après la fiche constructeur, on peut calculer :  $CV_e = 3040$ .

### 3.3.1 Configuration à 2 échangeurs

On trouve tout d'abord  $CV_{tot} = 5683,5$  par simple application numérique de ce qui précède. De même en utilisant (1) on obtient :

$$CV_{by} = CV_{tot} - 2 * \sqrt{\frac{CV_s^2 CV_e^2}{CV_s^2 + CV_e^2}} \quad (2)$$

On a alors

$$CV_{by_{max}} = 5209.37 \text{ et donc } ouv_{by_{max}} = 0.65$$

Nous avons rencontré de nombreuses difficultés à cette étape de notre travail, malgré des discussions avec les encadrants. Le lien entre l'ouverture et le coefficient de débit demeure une zone d'ombre de ce projet, malgré de nombreuses tentatives d'explication claire et précise. Malgré tout, nous avons réussi à obtenir la courbe en figure 10 :

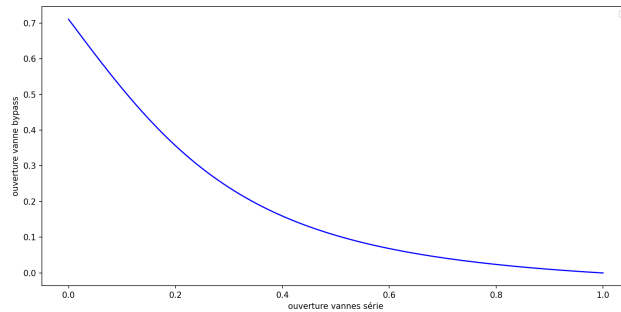


Figure 10: Graphique liant l'ouverture de la vanne bypass à l'ouverture des vannes séries



En posant les équations aux limites (c'est à dire les valeurs de  $ouv_{by}$  en 0 et en 1), on obtient :

$$\begin{cases} a = 0.71 \\ b = 0 \end{cases} \quad (3)$$

Ces valeurs sont identiques à celles obtenus en utilisant polyfit. En traçant la courbe  $x \rightarrow a(1-x)^3 + b$  (et donc pas le courbe  $x \rightarrow ax^3 + b$ ) on obtient la figure 11

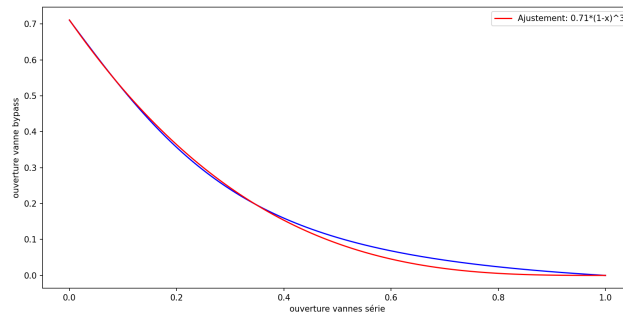


Figure 11: Superposition des 2 courbes liant l'ouverture bypass à l'ouverture série dans le cas 2 échangeurs

### 3.3.2 Configuration à 1 échangeur

Tout d'abord, il ne faut pas faire l'erreur de penser que  $CV_{tot}$  change entre ces 2 configurations ! Il faut garder en tête une des exigences, qui stipule que le débit doit rester constant lors du fonctionnement, même si on ferme une branche d'échangeur. Dès lors, on a toujours  $CV_{tot} = 5683,5$ . On obtient alors l'équation (eq. 4) et la courbe suivantes (figure 12).

$$CV_{by} = CV_{tot} - \sqrt{\frac{CV_s^2 CV_e^2}{CV_s^2 + CV_e^2}} \quad (4)$$

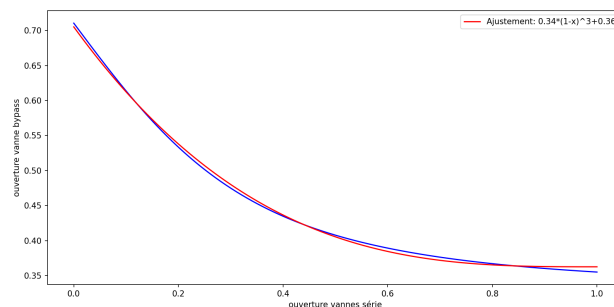


Figure 12: Superposition des 2 courbes liant l'ouverture bypass à l'ouverture série dans le cas 1 échangeur

C'est à cette étape que l'on rencontre des problèmes dans la modélisation. On constate en effet que la condition de vanne bypass fermée et de vanne série ouverte n'est pas respectée puisque la courbe ne prend pas la valeur 0 en 1. Par ailleurs lors de l'implémentation des coefficients trouvés sur OpenModelica, la régulation de température ne fonctionne plus, de même que le débit qui n'est pas constant. En changeant les coefficients à la main on parvient à obtenir une régulation de température efficace et un débit presque constant (moins 100 kg/s de variation même en ajoutant raisonnablement des pits de puissance mais pas en cas de mauvais fonctionnement d'une pompe ou d'un échangeur) comme le montre les figures 13 et ?? en suivant le scénario 1 sans arrêt de la pompe ni de l'échangeur. Dans ce second cas on constate que la régulation de température et que le débit constant n'est plus vérifié bien que la régulation de température continue de fonctionner lors de l'arrêt de la pompe 2 (figures 14 et 15). Cela provient de la mauvaise compréhension de la formule entre  $V_{bypass}$  et  $V_s$  et de sa modélisation en polynôme de degré 3.

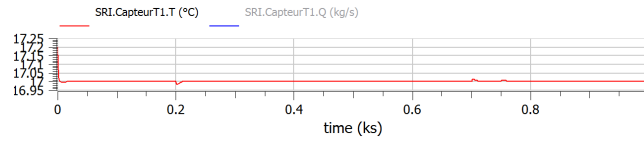


Figure 13: Evolution de la température

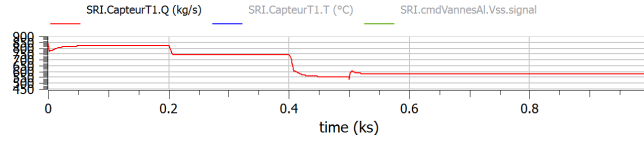


Figure 14: Evolution du débit et de la température dans le premier scénario avec arrêt de la pompe 2 et maintenance d'un échangeur

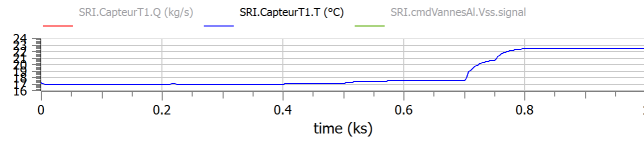


Figure 15: Evolution de la température dans le premier scénario avec arrêt de la pompe 2 et maintenance d'un échangeur

Enfin, on cherche la puissance maximale pour laquelle la régulation de température ne fonctionne plus. Il s'agit de 40MW dans notre cas (figures 16 et 17)

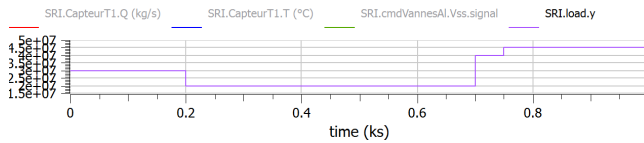


Figure 16: Evolution du débit et de la température dans le premier scénario avec arrêt de la pompe 2 et maintenance d'un échangeur

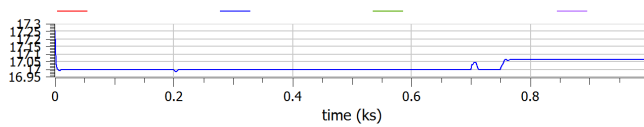


Figure 17: Evolution de la température dans le premier scénario avec arrêt de la pompe 2 et maintenance d'un échangeur

### 3.4 Commande finale des vannes

Au final, on obtient la commande de vanne suivante. On utilise des limiteurs pour implémenter les ouvertures minimales et maximales, et des sélecteurs pour changer ces extremums selon le nombre d'échangeur actif. Le sélecteur contrôlé par un créneau juste avant la sortie commandant la vanne bypass sert uniquement à régler une erreur d'initialisation du modèle OpenModelica et n'a pas d'influence sur le fonctionnement du système.

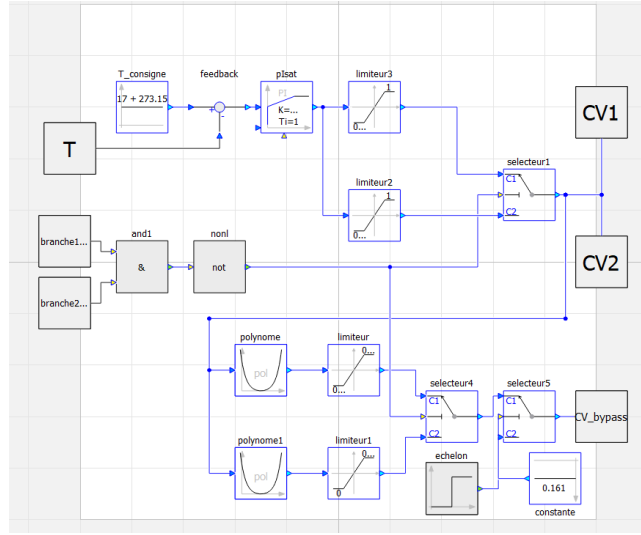


Figure 18: Commande finale des vannes sur OpenModelica

## 4 Vérification et analyse de la conception

### 4.1 Test de la régulation

On cherche à simuler le scénario suivant :

1. A 200 secondes, un client de 10MW est “déconnecté” du circuit.
2. A 400 secondes, la pompe 2 est arrêtée.
3. A 413 secondes, la pompe 3 est démarrée.
4. A 500 secondes, une opération de nettoyage de l'échangeur 1 commence.
5. A 700 secondes, un client de 10 MW est “connecté” au circuit.
6. A 750 secondes, un client de 5 MW est “connecté” au circuit.

On a alors le scénario Figure 19 :

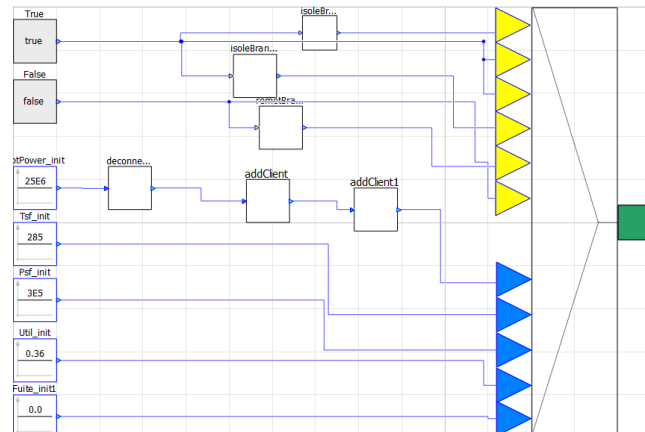


Figure 19: Schéma bloc du scénario

où l'évènement RemetBranche est une adaptation de IsoleBranche destiné à effectuer l'opération inverse.

On obtient l'ensemble de résultats Figure 20

On remarque donc qu'une bonne partie des exigences sont satisfaites, même si, par exemple, on ne parvient pas à conserver la consigne à 17°C

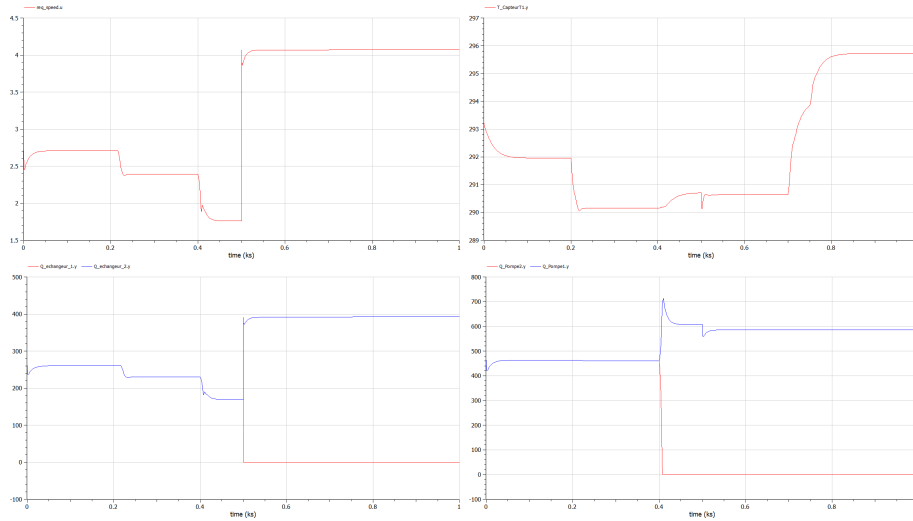


Figure 20: Résultats de simulation du scénario

## 4.2 Guide d'exploitation

### 4.2.1 Limites de charge

On souhaite déterminer à partir de quelle charge

- La consigne de régulation de la température à 17°C n'est plus tenue
- La limite haute de température de 30°C dans la branche froide du circuit de refroidissement est dépassée
- Le delta de température imposé à la source froide excède la limite autorisée

On va utiliser le scénario Figure 21

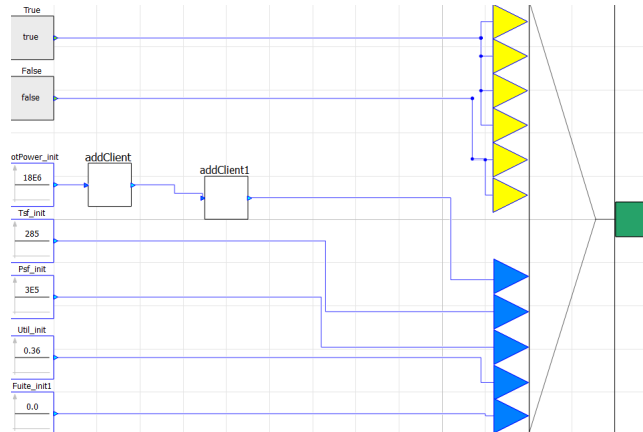


Figure 21: Scénario permettant de trouver les charges limites au comportement du système

En simulant ce scénario avec différentes valeurs de charge, on trouve que :

1. A partir d'une charge avoisinant 18,5 MW, le modèle ne parvient plus à maintenir une consigne à 17°C, il la dépasse légèrement, et de plus en plus lorsqu'on augmente la charge
2. Autour de 70MW, on franchit les 30°C à ne pas dépasser. Au-dessus de cette charge, il n'est donc plus possible que le SRI fonctionne en régime classique

## 5 Commande des pompes

Cette partie traite du développement du programme d'automatisation de gestion des pompes dans l'environnement de programmation Codesys. Le projet TP\_BackupPump a été utilisé comme base, fournissant les variables et objets d'environnement nécessaires pour la simulation et le contrôle des pompes.

## 5.1 Présentation des éléments de contrôle

Le système comporte trois pompes, chacune contrôlée par un opérateur via une Interface Homme-Machine (IHM) et surveillée par un automate. L'automatisme est configuré pour prendre en charge les scénarios de défaillance en assurant la continuité du débit. Les principaux éléments fonctionnels du système incluent :

- **Bloc fonctionnel PO\_SWITCH** : Ce bloc permet de relier l'IHM à l'automate. Il convertit l'ordre de démarrage ou d'arrêt d'une pompe, donné par l'opérateur, en un signal binaire compréhensible par l'automate.
- **Bloc fonctionnel ST\_1SM\_BFE** : Ce bloc simule la cellule électrique alimentant chaque pompe. Il transforme les signaux de commande de l'automate (SwitchOn et SwitchOff) en signaux d'état (*CB\_On*), indiquant si le moteur est en marche ou arrêté.
- **Interface de visualisation** : L'IHM permet de contrôler les pompes et de suivre leur état via des voyants lumineux (vert pour marche, rouge pour arrêt). Un interrupteur de simulation de défaillance a été ajouté pour tester les réponses de l'automate.
- **Variables globales (GVL)** : Elles stockent les états et commandes des pompes dans l'automate.

## 5.2 Conception du Programme d'Automatisation

Un programme nommé PUMP\_CTRL a été développé dans Codesys pour gérer automatiquement la reprise en cas de défaillance. Ce programme effectue les tâches suivantes :

1. **Surveillance de l'état des pompes** : L'automate surveille en permanence l'état de chaque pompe via le signal *CB\_On*. Si une pompe active présente une anomalie (absence de signal *CB\_On*), l'automate attend 5 secondes avant de confirmer la défaillance.
2. **Activation de la pompe de secours** : Après confirmation d'une défaillance, un délai de 7 secondes est observé avant le démarrage d'une pompe de secours disponible pour reprendre le relais.
3. **Gestion des défaillances multiples** : Si la pompe de secours ne se met pas en marche dans les 2 secondes suivant l'ordre de démarrage (par exemple, si elle est également en panne), l'automate activera une autre pompe disponible si possible. Cette exigence ne s'applique que si le circuit est dans une configuration avec une pompe en service.

## 5.3 Bloc Fonctionnel PO\_SWITCH

Le bloc PO\_SWITCH est conçu pour interpréter les commandes de l'interrupteur binaire de l'interface homme-machine (IHM) en les convertissant en signaux de démarrage ou d'arrêt pour chacune des pompes du système. Ce processus assure que chaque pompe répond de manière précise et immédiate aux actions de l'opérateur. Ainsi, lorsque l'opérateur active ou désactive un interrupteur, le bloc PO\_SWITCH traduit ces commandes en instructions d'enclenchement ou d'arrêt.

En plus de sa fonction principale de conversion des signaux, le bloc PO\_SWITCH permet également à l'automate programmable d'effectuer une lecture périodique de l'état des commandes d'arrêt et de démarrage. Cela garantit que le système est toujours en mesure de surveiller et de réagir aux changements de commandes, assurant ainsi une gestion fluide et sécurisée des opérations de chaque pompe. Cette capacité de mise à jour périodique des états des commandes contribue à la fiabilité et à la sécurité globales du processus de contrôle des pompes.

## 5.4 Bloc Fonctionnel ST\_1SM\_BFE

Le bloc ST\_1SM\_BFE a pour objectif de simuler de manière précise la cellule électrique associée à chaque pompe du système. Il reçoit deux types de signaux de commande, SwitchOn et SwitchOff, et utilise ces informations pour gérer le démarrage ou l'arrêt de la pompe correspondante. Ce bloc traite les signaux reçus en temps réel pour s'assurer que la pompe réagit correctement aux commandes de l'opérateur ou de l'automate.

En retour, le bloc ST\_1SM\_BFE fournit un retour d'état sous forme de signal binaire, nommé *CB\_On*, qui indique l'état de fonctionnement actuel de la pompe. Ce signal binaire, lorsqu'il est actif, confirme que le moteur est en marche et fonctionne correctement. Si le signal est inactif, cela signifie que le moteur est soit à l'arrêt, soit en situation de défaut. Cette fonctionnalité de retour d'état est essentielle pour permettre une détection rapide et précise des éventuels dysfonctionnements, garantissant ainsi un suivi en temps réel de l'état de la pompe et une intervention rapide en cas de problème. Le bloc permet donc de superviser efficacement la fiabilité et la sécurité des opérations de chaque pompe.

## 5.5 Schéma Fonctionnel

Un schéma fonctionnel détaillé a été conçu dans l'environnement de programmation Codesys, utilisant des blocs de contrôle, des temporisations et des conditions logiques pour assurer la détection rapide et fiable des défauts ainsi que la gestion automatisée des pompes de secours dans un système multi-pompes. Ce schéma est structuré au sein du programme **pump\_ctrl** en CFC (*Continuous Function Chart*), une approche graphique qui facilite la représentation de la logique de commande en continu pour des processus industriels nécessitant des opérations simultanées et des actions séquentielles. Dans ce contexte, l'implémentation se divise en trois sections principales : **Pump 1 Control**, **Pump 2 Control**, et **Pump 3 Control**, chacune de ces sections représentant le contrôle spécifique d'une pompe et intégrant une logique de gestion d'état et de sécurité.

Chaque schéma de contrôle de pompe est subdivisé en plusieurs fonctions essentielles à son bon fonctionnement, comme la surveillance de la disponibilité, la détection de défauts, et l'activation ou la désactivation des pompes en fonction des conditions du processus. Par exemple, la fonction **PO1\_FAULT\_CONF** a pour objectif de confirmer la présence d'un défaut dans la pompe 1 (PO1) selon une logique précise. Elle est déclenchée si, après une temporisation de 5 secondes, le signal **CR\_PO1\_E** reste à 0, ce qui signifie que la pompe ne s'est pas enclenchée malgré une consigne d'activation envoyée précédemment. Cette temporisation de 5 secondes est critique, car elle permet de tolérer un léger délai d'activation normal avant de signaler un défaut potentiel, réduisant ainsi le risque de fausses alertes dues à des latences opérationnelles. L'activation de **PO1\_FAULT\_CONF** entraîne alors des actions de sécurité, comme le déclenchement d'une alarme ou le basculement vers une pompe de secours, garantissant ainsi la continuité de service.

En parallèle, une autre fonction essentielle nommée **PO\_disp** est mise en œuvre pour surveiller la disponibilité de chaque pompe individuelle. La fonction **PO\_disp** vérifie si une pompe donnée est prête à être réactivée en analysant son état actuel et les commandes opérateurs. Pour qu'une pompe soit considérée comme disponible, il est nécessaire que la dernière commande opérateur ait été un arrêt et que la pompe ne soit actuellement pas en fonctionnement (indiqué par le signal **CR\_PO\_E = 0**). Ce mécanisme permet d'éviter de tenter d'activer une pompe déjà en cours d'utilisation ou qui vient tout juste d'être arrêtée, assurant ainsi une gestion optimisée des ressources et évitant les démarrages intempestifs.

Le programme **pump\_ctrl** est donc conçu pour gérer automatiquement les états de fonctionnement et de secours des pompes, en priorisant la fiabilité et la sécurité du système. En cas de défaut confirmé, les logiques de redondance intégrées dans chaque schéma (Pump 1, Pump 2, Pump 3) permettent de basculer rapidement vers une autre pompe disponible, minimisant les interruptions de service. De plus, le système de temporisation et de confirmation d'erreur implémenté dans **PO1\_FAULT\_CONF** contribue à différencier un vrai défaut d'une simple latence, tandis que la fonction **PO\_disp** assure une disponibilité optimale et contrôlée de chaque pompe.

En somme, cette approche modulaire et bien structurée du contrôle des pompes dans Codesys permet non seulement une maintenance facilitée, mais également une capacité d'évolution future, rendant possible l'ajout de nouvelles fonctions de contrôle ou de sécurité sans nécessiter de révisions majeures du code existant. Le système peut ainsi s'adapter à des exigences accrues ou à des configurations de pompes plus complexes.

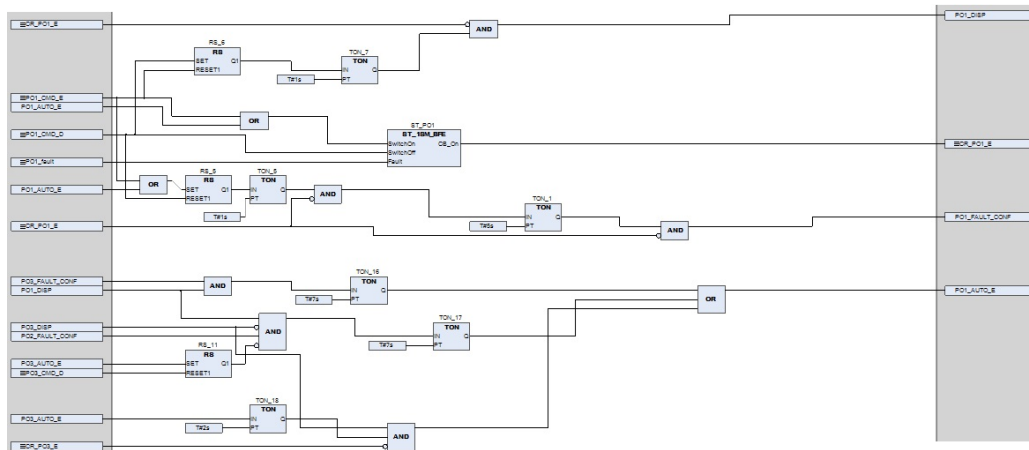


Figure 22: Schéma bloc commandant la pompe 1

## 5.6 Analyse des résultats

Dans le cas d'une activation de la pompe 1 et de la détection d'une défaillance sur celle-ci, le système fonctionne conformément aux exigences du cahier des charges en passant automatiquement à la pompe de secours. Ainsi, lorsque la pompe 1 présente une défaillance, la pompe 2 est sollicitée pour prendre le relais, à condition qu'elle soit disponible. Dans le cas où la pompe 2 est elle-même défaillante ou indisponible, le contrôle

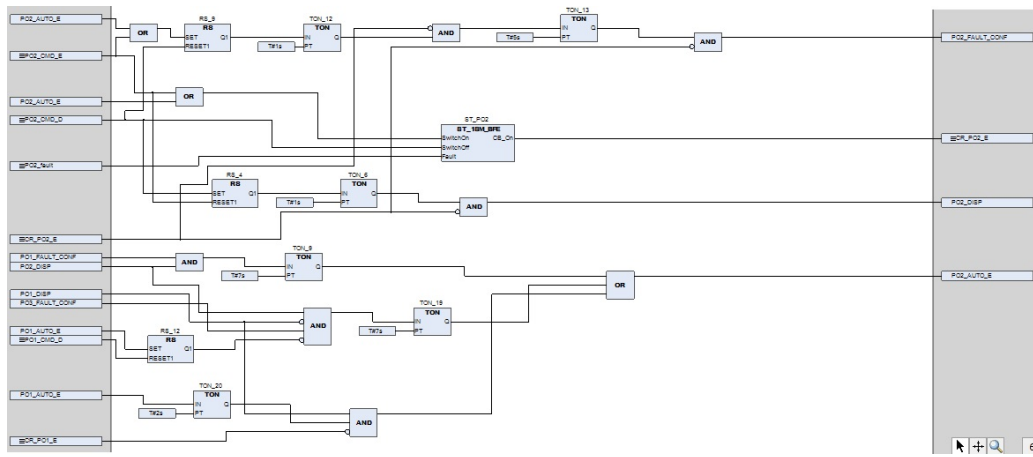


Figure 23: Schéma bloc commandant la pompe 2

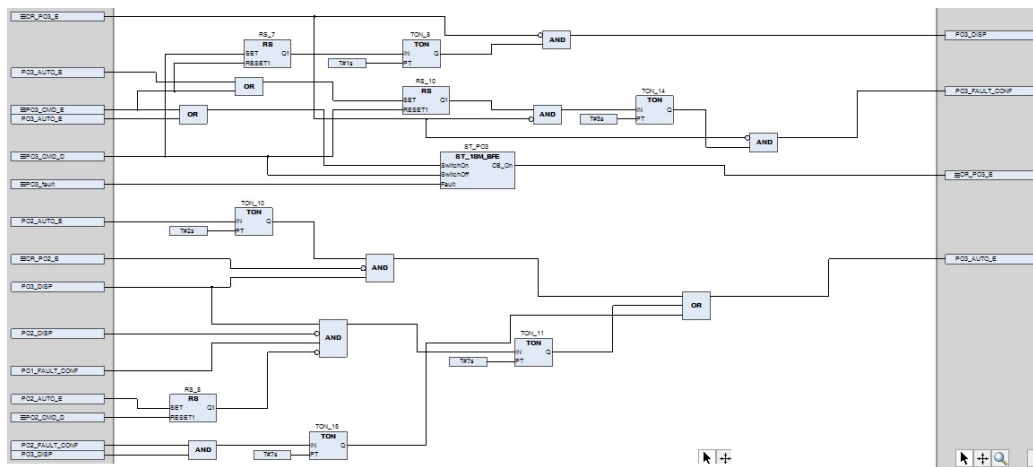


Figure 24: Schéma bloc commandant la pompe 3

basculer immédiatement sur la pompe 3, qui s'active dans les délais spécifiés, garantissant ainsi la continuité de fonctionnement du système.

Cette séquence de redondance est permise par la structure de contrôle en cascade implémentée dans notre schéma fonctionnel. Celui-ci assure une vérification en temps réel de l'état de chaque pompe, et déclenche automatiquement les pompes de secours dès qu'une défaillance est confirmée sur une pompe en fonctionnement. Le système de temporisation intégré dans chaque fonction de contrôle permet de respecter précisément les délais de basculement, comme spécifié dans le cahier des charges. En d'autres termes, cette architecture de contrôle garantit que, même en cas de multiples défaillances, une pompe disponible sera mise en route dans les temps impartis, minimisant ainsi les interruptions et assurant une continuité d'opération.

En résumé, nous observons que le schéma fonctionnel mis en place remplit son rôle de gestion des pompes en cas de défaillance de manière fluide et efficace. Il permet une activation automatique des pompes de secours dans un ordre hiérarchisé, assurant ainsi le bon fonctionnement du système en toutes circonstances, et ce, en respectant les délais d'activation fixés dans le cahier des charges.

## 5.7 Bilan de la commande des pompes

Cette phase du projet a permis de poser les bases du système de contrôle et de concevoir un programme d'automatisation complet pour assurer une gestion fiable des pompes, notamment en cas de défaillance. Une attention particulière a été portée à la définition des différents éléments de contrôle, tels que les capteurs de détection d'état, les commandes d'activation et de désactivation, ainsi que les signaux de retour d'information pour confirmer l'état des pompes en temps réel. Ces éléments sont intégrés de manière à garantir que chaque pompe soit surveillée et commandée de manière optimale, selon les spécifications établies dans le cahier des charges.

Le schéma fonctionnel de l'automate a été conçu pour répondre aux exigences de performance et de sécurité du système. Celui-ci intègre des étapes de détection de défaillances, de temporisation et de redémarrage automatique, ce qui assure une continuité de service même si une pompe devient inopérante. Chaque étape du

processus a été modélisée et programmée pour anticiper les cas de panne et déclencher le redémarrage des pompes de secours de manière automatique et rapide. En cas de défaillance d'une pompe, les délais de basculement ont été soigneusement configurés pour respecter les tolérances définies, évitant ainsi des temps d'arrêt prolongés.

En parallèle, la modélisation du schéma fonctionnel inclut des vérifications redondantes pour garantir que chaque pompe de secours soit prête à fonctionner lorsqu'elle est sollicitée. L'automate effectue ainsi une vérification régulière de la disponibilité de chaque pompe de secours et déclenche les actions nécessaires en fonction de l'état détecté. Ce système de redondance et de validation dynamique des états permet non seulement de répondre aux exigences de continuité opérationnelle, mais également de minimiser le risque de basculements inutiles ou inappropriés.

En somme, cette étape de conception a permis de structurer un programme d'automatisation robuste, capable de gérer les pompes en cas de défaillance, tout en assurant un niveau de réactivité et de sécurité adapté aux attentes du projet. Le système, grâce à ce schéma fonctionnel, peut gérer des défaillances de manière autonome, en activant les pompes de secours dans un ordre logique et optimisé. Les fonctions de contrôle sont ainsi conformes aux spécifications techniques et garantissent un fonctionnement en continu, sécurisant la gestion des pompes et répondant pleinement aux exigences définies dans le cahier des charges.



## 6 Conclusion

En conclusion, nous avons réussi à concevoir un Service de Refroidissement Intermédiaire (SRI) pour la centrale nucléaire et à en vérifier la compatibilité avec les exigences spécifiques. La prise en main des outils OpenModelica et CodeSys s'est cependant révélée complexe, nécessitant un investissement important en temps et en apprentissage technique. Nous ne comprenons toujours pas la raison de certaines erreurs liées à l'initialisation. Par ailleurs, bien que nous ayons atteint la majorité des objectifs, nous n'avons pas pu implémenter un scénario de test "grand chaud", ce qui aurait permis une évaluation plus complète de la robustesse du système dans des conditions extrêmes. De plus, nous avons rencontré des difficultés à concilier un débit constant avec une régulation fine de la température autour de 17°C. Enfin, il aurait été souhaitable de tester plus en profondeur la commande des pompes. Malgré tout, nous avons bien aimé travailler sur cet EI et remercions nos encadrants pour leur temps.