

Exercices du cours VSE

Exercices de vérification SystemVerilog

Vérification formelle d'un processeur compteur de 1s

semestre automne 2024 - 2025

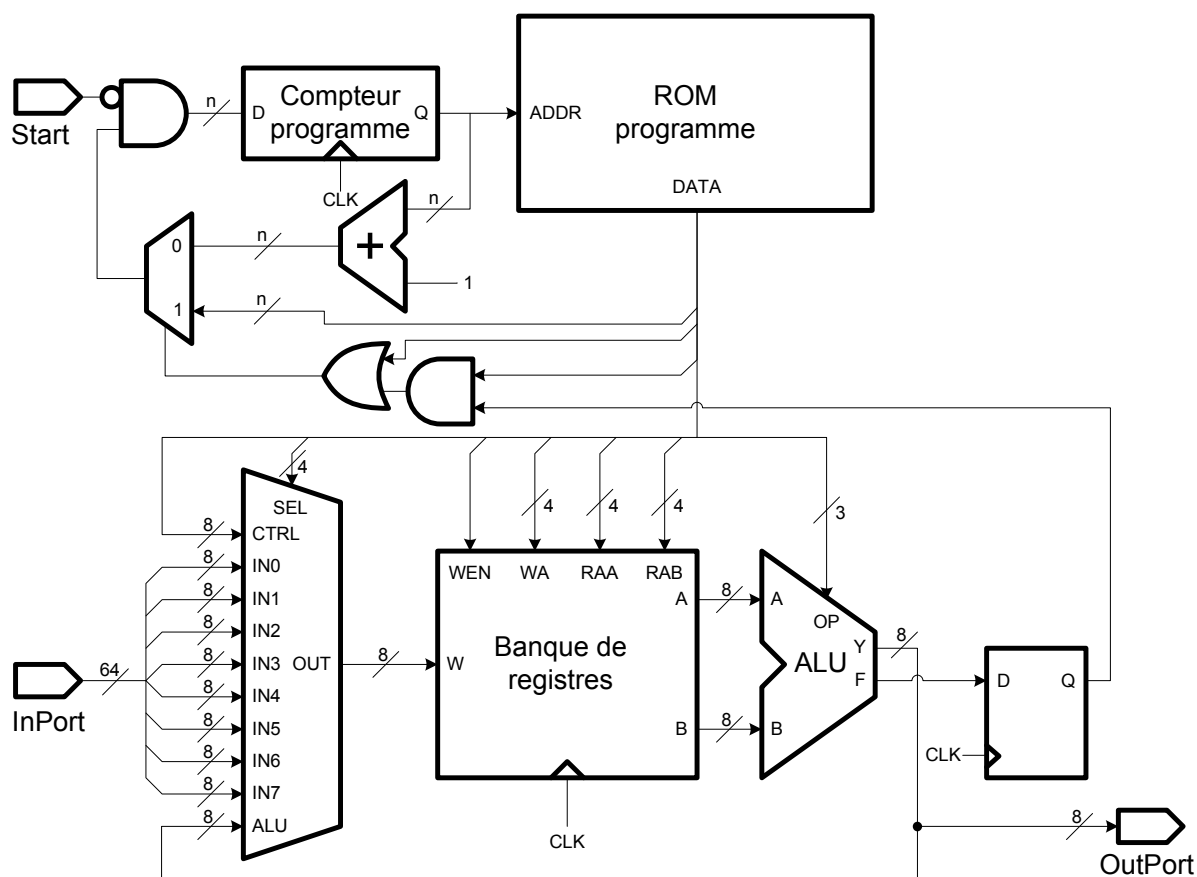
Contexte

Une équipe a développé un processeur micro-programmé capable de compter le nombre de 1s présents sur un bus d'entrée. Celui-ci est décomposé en plusieurs modules et nous sommes intéressés à vérifier formellement le design proposé.

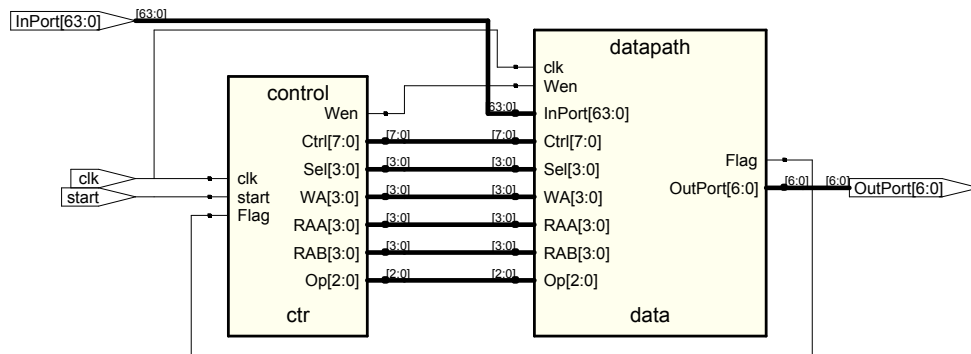
Nous allons répartir la charge sur nos différentes équipes de développeurs.

1 Processeur complet

L'architecture générale du système est la suivante :



Elle est toutefois décomposée en une partie contrôle et une unité de traitement :

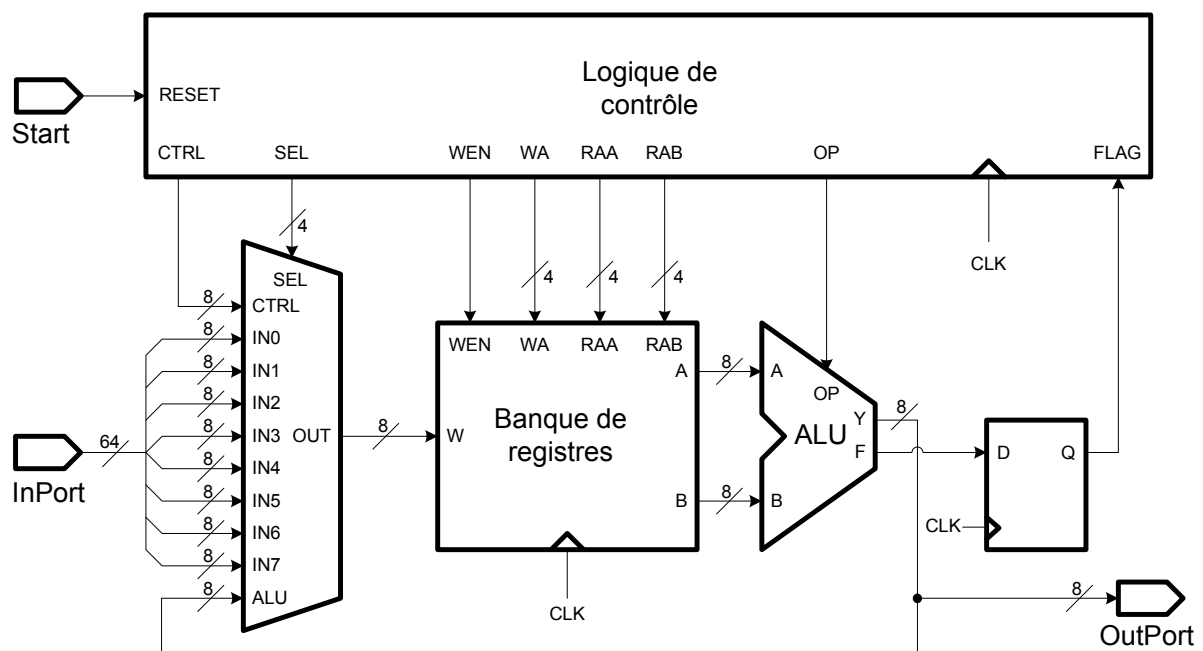


La partie contrôle est elle-même décomposée en un compteur programme et une mémoire ROM.

2 Unité de traitement

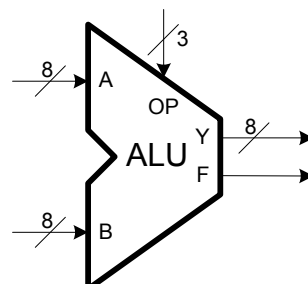
L'unité de traitement est décomposée en une banque de registres, une unité arithmétique et logique, et le reste des éléments (MUX + bascule).

Elle correspond à la partie basse du schéma suivant :



3 Unité arithmétique et logique

L'unité arithmétique et logique offre l'interface suivante :

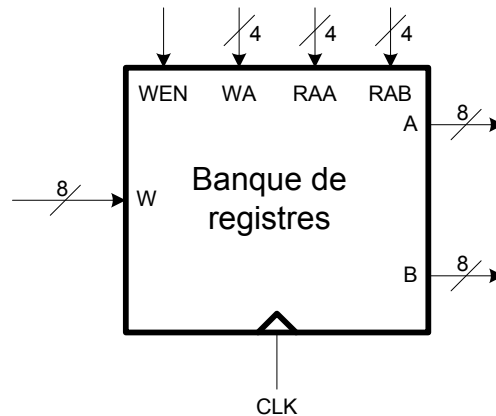


En fonction de la valeur de `OP` fournie, l'opération est la suivante :

ADD	000	$Y = A + B$
SHR	001	$Y = A \gg 1$
EQ	010	$F = (A == B)$
AND	011	$Y = A \text{ and } B$
MOV	100	$Y = A$

4 Banque de registres

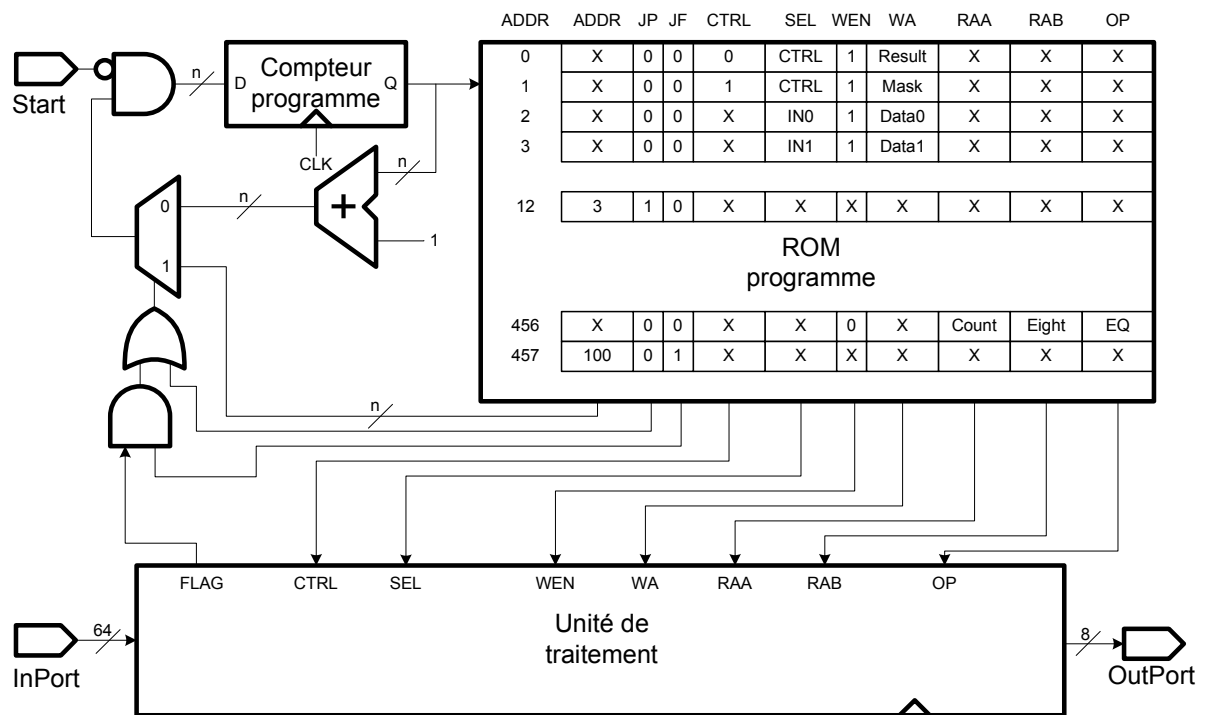
La banque de registre correspond au composant suivant :



Elle contient 16 éléments de 8 bits. Deux ports de lecture A et B permettent de lire simultanément les données présentes aux adresses RAA et RAB. Les lectures sont asynchrones et l'écriture est par contre synchrone, activée par le signal `WEN` permettant d'écrire la donnée `W` à l'adresse `WA`.

5 Compteur programme

Le compteur programme contient le compteur et la logique permettant d'effectuer des sauts conditionnels et inconditionnels, notamment en fonction du flag présent en sortie de l'ALU. Par défaut il s'incrémente, mais si `start` est à 1 il est remis à 0. Si ce n'est pas le cas, si `Flag` est actif et que `JF` l'est également alors il doit être chargé avec l'adresse fournie en entrée. Ceci est également appliqué si l'entrée `JP` est active.



6 Composants à valider

Les composants suivants peuvent être validés individuellement :

1. Compteur programme
2. ALU
3. Banque de registres
4. Unité de traitement
5. Tout sauf la mémoire
6. Bloc complet