**Rapport Final : Système de Contrôle d'Accès pour un Bâtiment Sécurisé**

**Introduction**

Ce projet a pour objectif de concevoir un système de contrôle d'accès sécurisé pour un bâtiment en utilisant des concepts de logique formelle, d'automates finis, de circuits logiques et de vérifications formelles. Le processus, depuis la spécification initiale jusqu'à la validation, est détaillé dans ce rapport.

**1. Présentation des spécifications initiales**

Le système doit gérer les accès à un bâtiment sécurisé selon les règles suivantes :

1. La carte d'accès doit être valide pour activer la vérification du code.
2. Si le code est correct, l'accès est accordé.
3. En cas de carte ou de code invalide, une alarme doit être déclenchée.
4. Le système peut être réinitialisé à tout moment pour revenir à l'état initial.

**États définis :**

* **État initial** : Le système est en attente d'une carte.
* **Vérification de carte** : Validation de la carte d'accès.
* **Accès accordé** : L'utilisateur obtient l'accès.
* **Alarme déclenchée** : Le système signale une anomalie.

**2. Description des modèles d'automates et des circuits logiques réalisés**

**2.1 Modèle d'automate**

Un automate fini déterministe (AFD) a été conçu pour représenter les transitions du système :

* Transitions définies par des règles logiques simples :
  + **Carte valide** → Transition vers VERIFICATION\_CARTE.
  + **Carte invalide** → Transition vers ALARME.
  + **Code correct** (après une carte valide) → Transition vers ACCES\_ACCORDE.
  + **Code incorrect** → Transition vers ALARME.

**2.2 Circuits logiques**

Des circuits logiques simples, basés sur les expressions booléennes suivantes, ont été implémentés pour optimiser les règles :

* **Validation de carte** : Carte == "valide"
* **Validation de code** : Code == "1234"

Ces règles ont été simplifiées à l'aide de l'algèbre de Boole pour minimiser les redondances.

**3. Résultats des vérifications formelles et des optimisations**

**3.1 Vérifications formelles**

Les méthodes de *model checking* ont été appliquées pour vérifier la conformité :

* L'automate respecte les règles définies, sans transitions indésirables.
* Les tests ont confirmé que :
  + Une carte invalide déclenche systématiquement une alarme.
  + Une carte valide suivie d'un code correct accorde l'accès.
  + Les scénarios non spécifiés (entrées incohérentes) ne provoquent pas de comportement imprévu.

**3.2 Optimisations**

Les règles logiques ont été simplifiées à l'aide de :

* **Carte de Karnaugh** pour réduire les expressions booléennes.
* Réduction des états inutiles dans l'automate.

**4. Discussion des défis rencontrés et des solutions apportées**

**Défis :**

1. **Gestion des transitions multiples** :
   * Initialement, l'état VERIFICATION\_CARTE ne gérait pas la vérification du code de manière fluide.
   * Solution : Séparer la gestion des transitions pour la carte et le code.
2. **Interaction utilisateur** :
   * Le système devait être interactif tout en respectant les règles.
   * Solution : Ajout d'un menu dynamique pour permettre une interaction intuitive.
3. **Validation formelle** :
   * Vérifier l'absence de transitions imprévues dans l'automate.
   * Solution : Test exhaustif des scénarios possibles avec des outils de simulation.

**5. Conclusion sur l'efficacité et la fiabilité du système conçu**

Le système de contrôle d'accès conçu respecte les spécifications initiales. Il offre les garanties suivantes :

* **Fiabilité** : Les règles logiques et les transitions d'états ont été validées formellement.
* **Efficacité** : Le système utilise des circuits logiques simplifiés pour réduire les coûts et améliorer les performances.
* **Simplicité d'utilisation** : Le menu interactif facilite l'utilisation par l'utilisateur final.

**Bilan final :** Le projet atteint son objectif principal de fournir un système sûr, fiable et facile à intégrer dans un bâtiment sécurisé. Il peut être étendu pour inclure des fonctionnalités supplémentaires, comme l'enregistrement des utilisateurs ou une gestion centralisée.