

Université Paris 7 : Diderot

Conducteur automatique Scade

Tiziri GUELLAL
Belaïd LAGHA

Table des matières

Introduction.....	
Programme du conducteur.....	
Suivi d'une trajectoire et rectification chemin.....	
Lecture couleurs « ColorDetect ».....	
Rectification trajectoire «PID» «ErrorCalculate».....	
Lecture itinéraire et instructions.....	
Interprétation de l'action lue « isAction ».....	
Tourner « Turn» « TurnAction ».....	
Calcul angle rotation « CalculTimeTurn » « TimeT ».....	
Détection des obstacles « Stop Action ».....	
Problèmes rencontrés.....	
Conclusion.....	

Introduction

Dans ce projet il s'agit de réaliser un conducteur automatique d'une voiture qui se déplace dans un environnement virtuel partant d'un point A initial à un point B final.

- Un environnement est représenté par un tableau d'itinéraires
- Les valeurs possibles d'un itinéraire sont : go, turn et stop
- La voiture possède trois détecteurs : RoadColor pour la route, Feux de signalisation (Vert rouge et Orange) et Sensor pour les obstacles
- La route a une ligne bleu centrale, à gauche se trouve du Cyan et à droite se trouve du Magenta.

Un environnement virtuel est défini afin de procurer une base à l'exécution de notre conducteur, il lit les itinéraires, les obstacles ..etc

Programme du conducteur automatique

Afin de rendre notre conducteur utilisable nous avons rajouté l'opérateur IA dans le diagramme, cet opérateur prend SensorsData, Itinerary en entrée et Arriving et RotationalSpeed en sortie.

Le véhicule est en état de marche quand il passe à l'état « Running » dans l'automate de la (figure-1)

Le conducteur automatique gère les tâches suivantes dans l'opérateur IA

- Suivre une trajectoire et rectifier son chemin
 - Opérateur **ColorDetect**
 - Opérateur **GoAction**
 - Opérateur **PID**
 - Opérateur **IntegralOp**
 - Opérateur **DerivatOp**
 - Opérateur **ErrorCalculate**
 - Opérateur **VitesseInterval**
 - Opérateur **GoAhead**
- Lire un itinéraire et suivre ses instructions à savoir avancer à la vitesse indiquée, tourner avec un angle donné et s'arrêter
 - Opérateur **isAction**
 - Opérateur **turn**
 - Opérateur **AfterTurn**
- Détecter les signaux de signalisation
 - Opérateur **ColorDetect**
- Détecter des obstacles
 - Opérateur **StopAction**
 - Opérateur **FrontDetect**

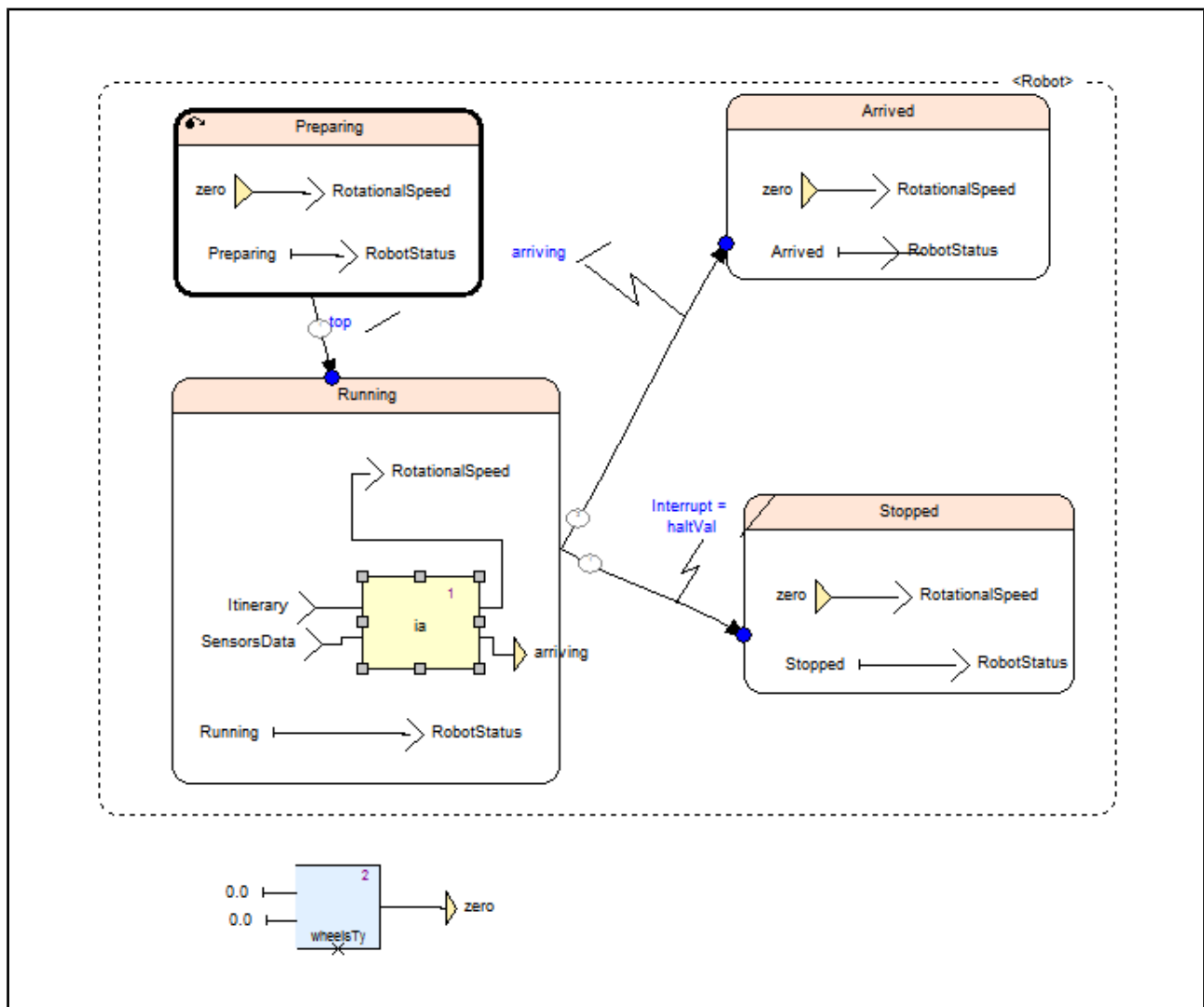


Figure 1 : Automate Run

Opérateur IA

L'opérateur IA est décrit par un automate de 8 états qui initialement est dans l'état « départ » qui selon l'itinéraire lu va dans les états *Go*, *Turn* ou *stop*

Automate IA

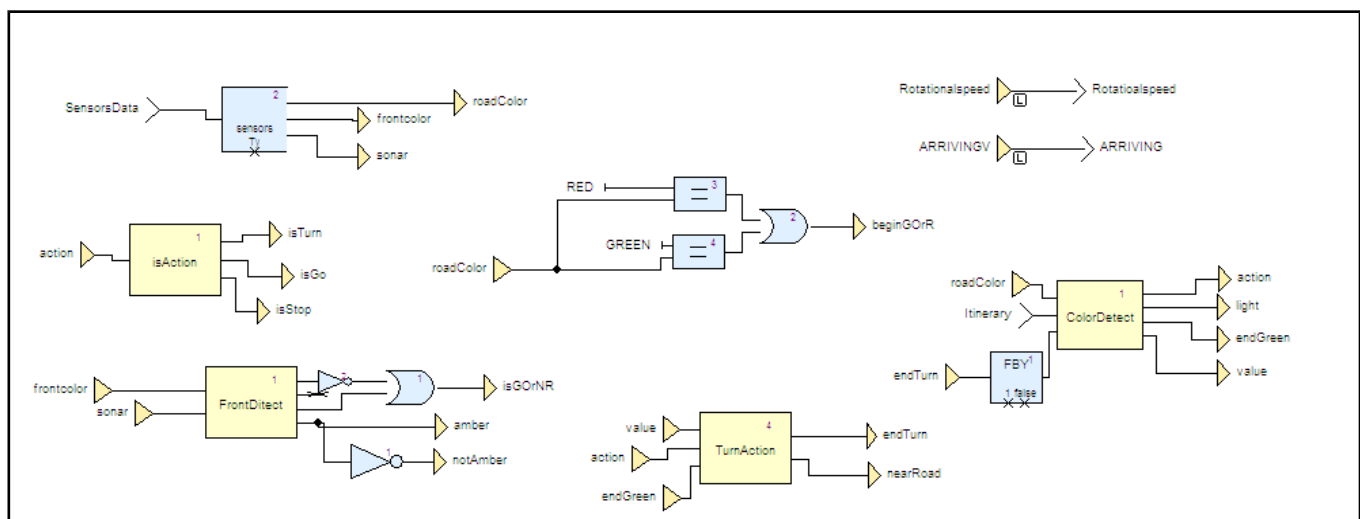


Figure 2 : Opérateur « IA »

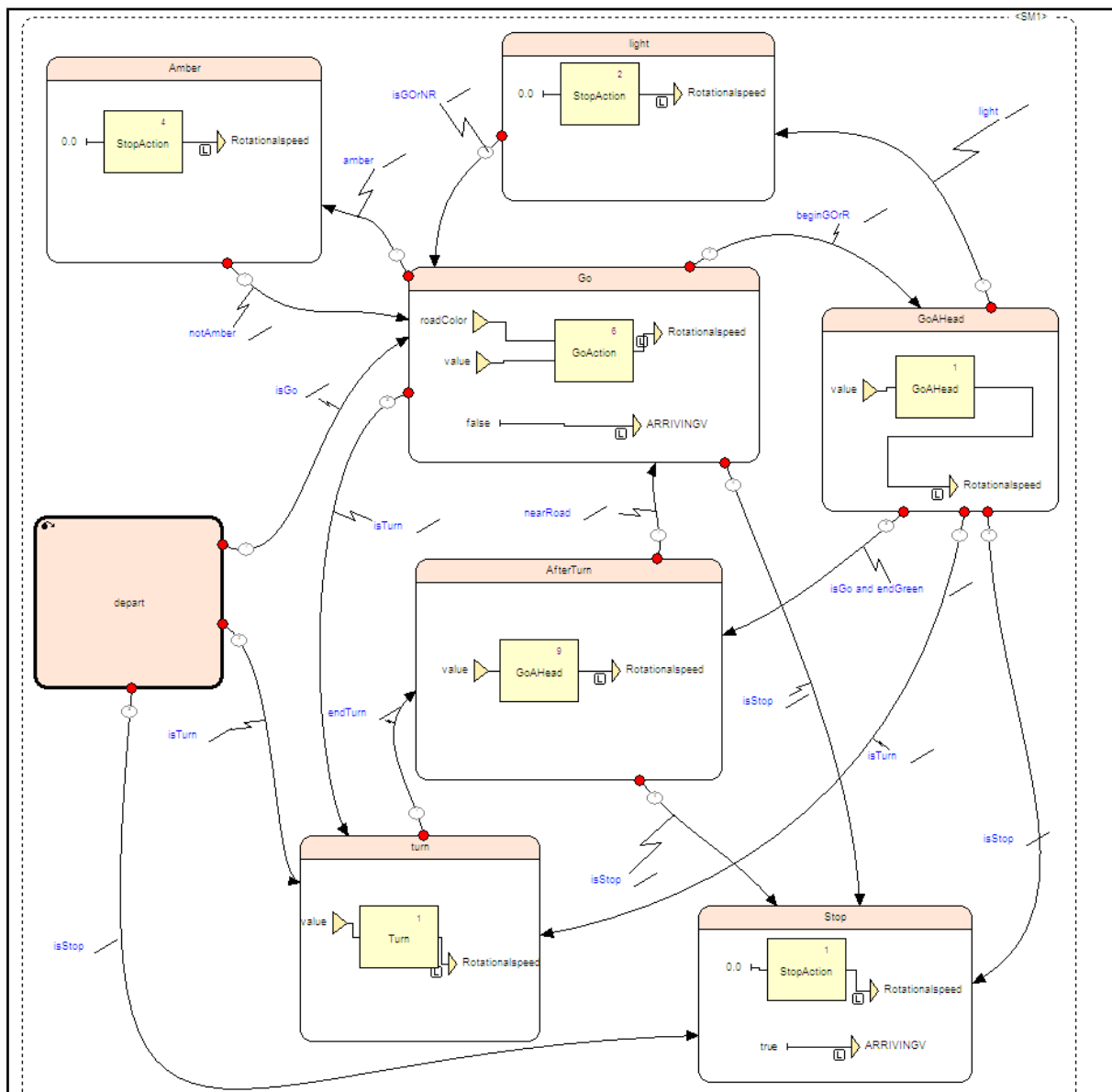


Figure 3: Automate IA

L'opérateur IA est composé d'un automate qui décrit le comportement du véhicule chaque comportement est décrit dans un état,

Au lancement du Robot l'opérateur IA démarre dans l'état départ et avec ses différents capteurs (ventral et frontal) il lit et interprète les données liées à la couleur de la route, les instructions lues et les feux et obstacles détectés

- il va dans l'état Go si l'instruction lue est go
 - si la couleur verte ou rouge est détectée (capteur ventral) l'automate va dans l'état GoAhead
 - si le véhicule vient de sortir du rouge sur la route l'automate sera sur l'état light
 - si le véhicule sort du vert et l'action = Go alors il retourne dans l'état Go si il l'action = Turn il va dans l'état Turn
 - si un obstacle est détecté l'automate va dans l'état amber

Suivi de trajectoire et rectification d'erreur

Le suivi de la trajectoire est géré par l'opérateur « ColorDetect » qui récupère les valeurs de couleur lues sur la route et suit la ligne bleue et respecte le code de couleur qui indique l'action à réaliser

Opérateur ColorDetect

L'opérateur ColorDetect décide de l'action à exécuter en prenant compte de la couleur détectée « **RoadColor** » sur le capteur ventral et la prochaine instruction lue de l'itinéraire « **Itinerary**. »

- Si la couleur détectée est verte alors on lit la prochaine action à appliquer (le signal n'est transmis qu'au moment du front descendant c-à-d au moment ou la couleur verte détectée passe du vert à la couleur de la route)
- Si la couleur est rouge alors l'indiquer au contrôleur (idem le signal est transmis au moment du front descendant).

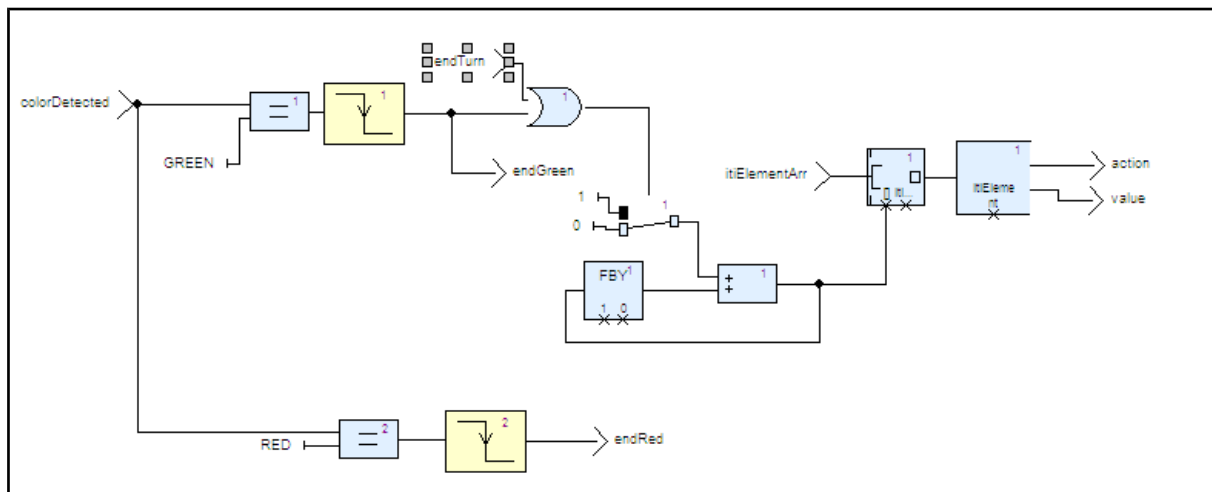


Figure 4: Opérateur ColorDetect

Rectification de la trajectoire et Vitesse max autorisée

Opérateur GoAction

L'opérateur GoAction permet d'avancer sur la trajectoire tout en respectant la vitesse maximale autorisée, il calcule le degré d'éloignement de la ligne bleue et rectifie la trajectoire

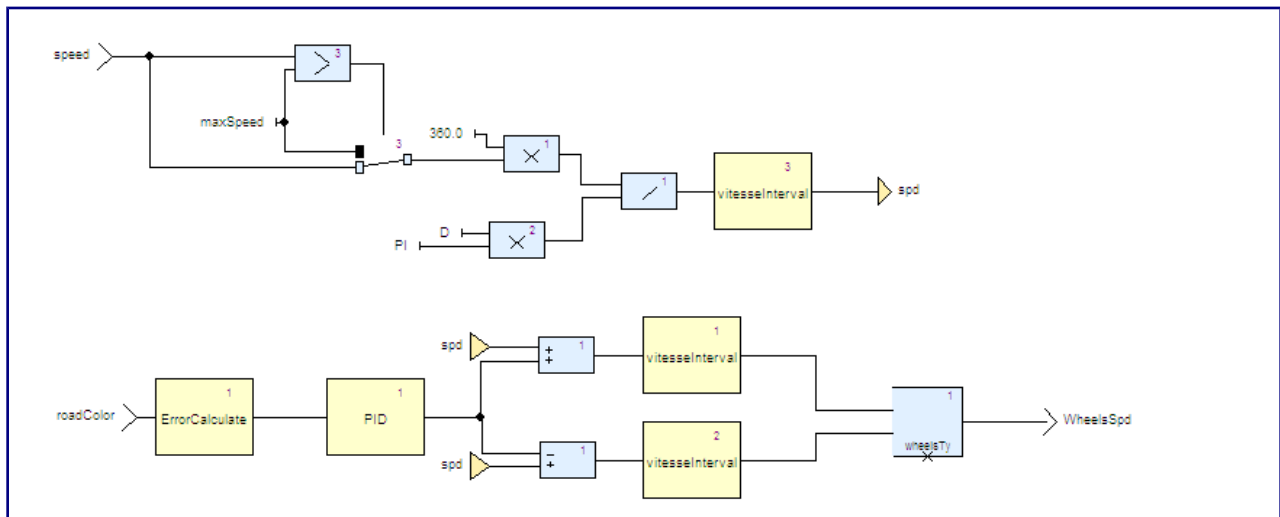


Figure 5: Opérateur GoAction

L'opération de rectification de trajectoire va remettre le véhicule sur la ligne bleue si jamais il s'en éloigne, cette rectification est faite par un PID

- Calcul de l'erreur avec l'opérateur « ErrorCalculator »
- Rectification avec les trois pilotes du PID

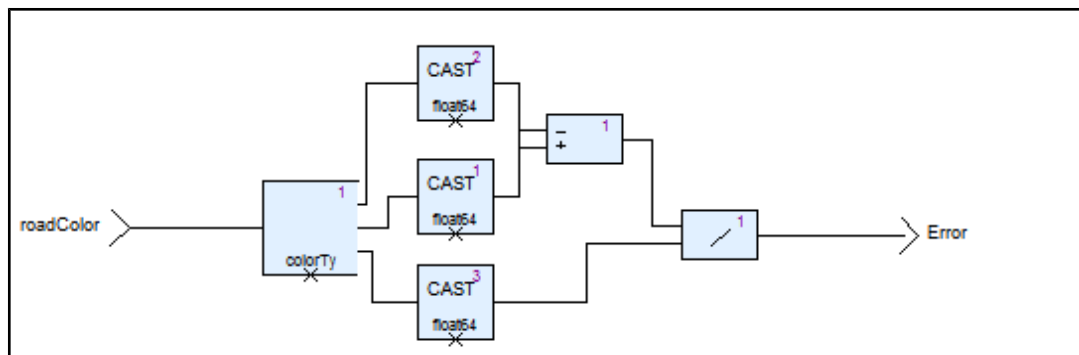


Figure 6: Opérateur « ErrorCalculate »

ErrorCalculate retourne le degré d'éloignement du véhicule de la ligne bleue en utilisant la formule suivantes :

$$\frac{colorRed - colorGreen}{Colorbleu}$$

Les trois pilotes font en sorte de remettre le véhicule sur le chemin selon le degré d'éloignement de la ligne bleue

La phase proportionnelle :

Dans cette phase la correction est appliquée pour tout écart entre la mesure et la consigne(l'erreur), plus cette erreur est grande plus la correction apportée l'est aussi.

La phase intégral :

Cette phase en complément à l'action proportionnelle apporte une notion de temps d'intégration à la correction et permet de stabiliser dans le temps l'action proportionnelle, plus l'erreur mesurée est constante plus la correction l'est aussi.

Il s'agira ainsi de surveiller l'évolution en conservant les différentes erreurs (écart mesurée) pour affiner le temps d'intégrale à appliquer à la correction.

La phase dérivée :

Cette phase permet d'anticiper un dépassement de la consigne en introduisant une correction en cas de risque.

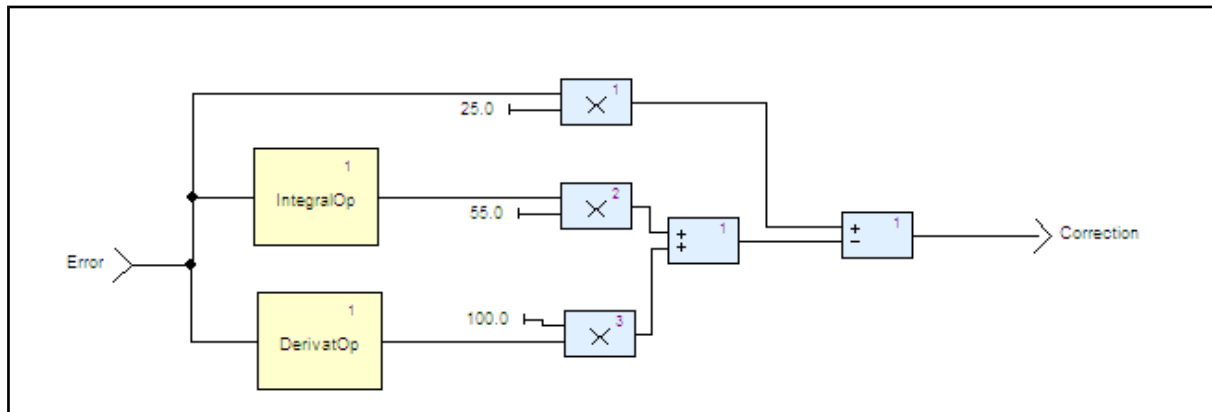


Figure 7 : Opérateur « PID »

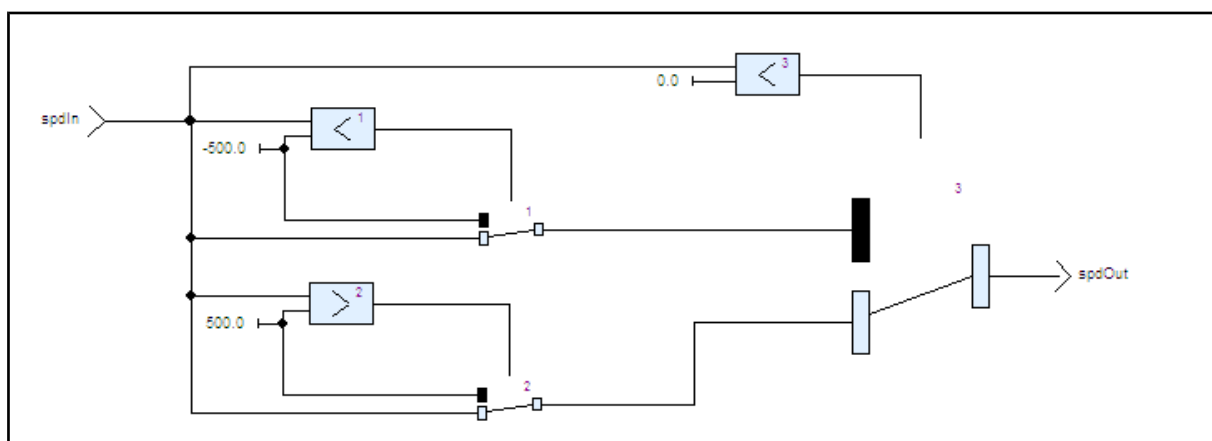


Figure 8: Opérateur vitesseIntervalle

L'opérateur VitesseIntervalle vise à garder la vitesse des roues comprises dans l'intervalle $MaxRotationWheel$ et $-MaxRotationWheel$ vitesse angulaire maximale autorisée des roues en marche avant ou en marche arrière

Opérateur GoAhead

L'opérateur GoAhead est appelé au moment où le véhicule détecte des indicateurs verts ou rouges sur la route (indicateurs de lecture d'action, Feu rouge), l'utilité de cet opérateur c'est d'éviter que le PID ne se mêle de la direction du véhicule lors d'une rotation, car nous sommes tombés sur des cas où le PID rectifie la trajectoire du véhicule après la lecture des couleurs verts et rouges sur la route (indicateurs de lecture d'action, Feu rouge),

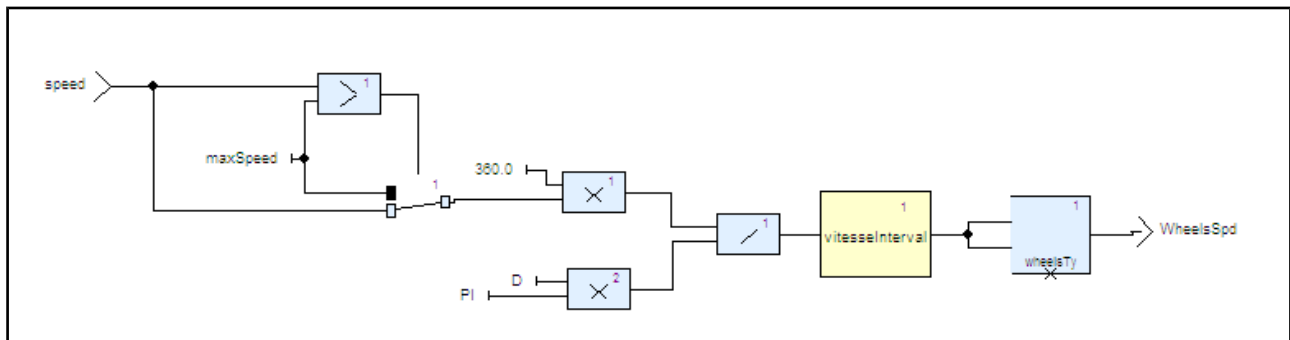


Figure 9 : Opérateur GoAhead

Lecture itinéraire et instructions

Interprétation de l'action lue opérateur « isAction »

L'action retournée par l'opérateur « ColorDetect » est ensuite prise en entrée par l'opérateur « isAction » celle-ci vérifie de quelle action il s'agit.

- si l'action retournée est **Turn** alors l'automate passe à l'état **Turn**
- si l'action retournée est **Go** alors l'automate passe à l'état **Go**
- Si l'action retournée est **Stop** alors l'automate passe à l'état **Stop**

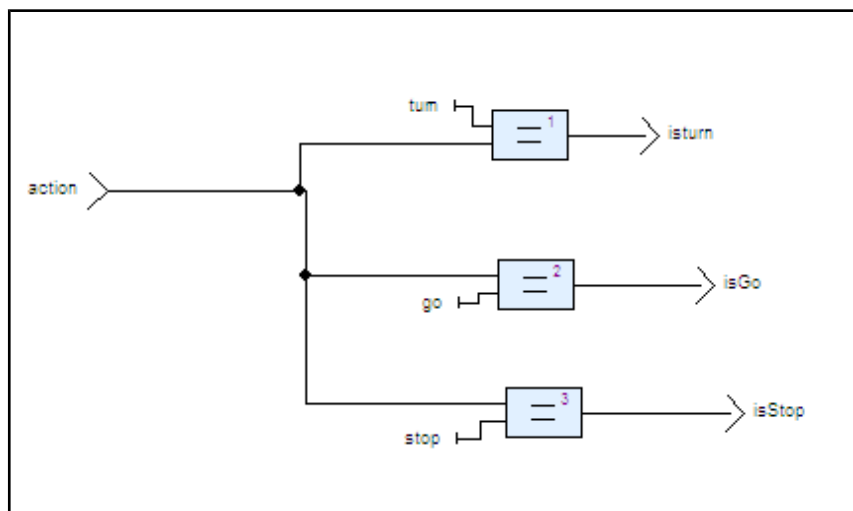


Figure 10 : Opérateur isAction

Tourner

L'automate passe à l'état Turn si l'instruction «turn» a été lue, à ce moment là le véhicule tourne d'un degré alpha donné (également lu sur l'itinéraire).

Afin de faire tourner le véhicule dans un sens ou l'autre d'un degré alpha il faut donner des vitesses inverses à chacune des deux roues durant un certain nombre de tics. Nous avons fixé cette vitesse à 200 et -200 (constante TurnSpeed), le calcul du degré de rotation revient à calculer le nombre de tics nécessaires.

Le choix de cette vitesse est arbitraire nous l'avons fixée de sorte à ce qu'elle ne

dépasse pas l'intervalle de vitesse autorisée

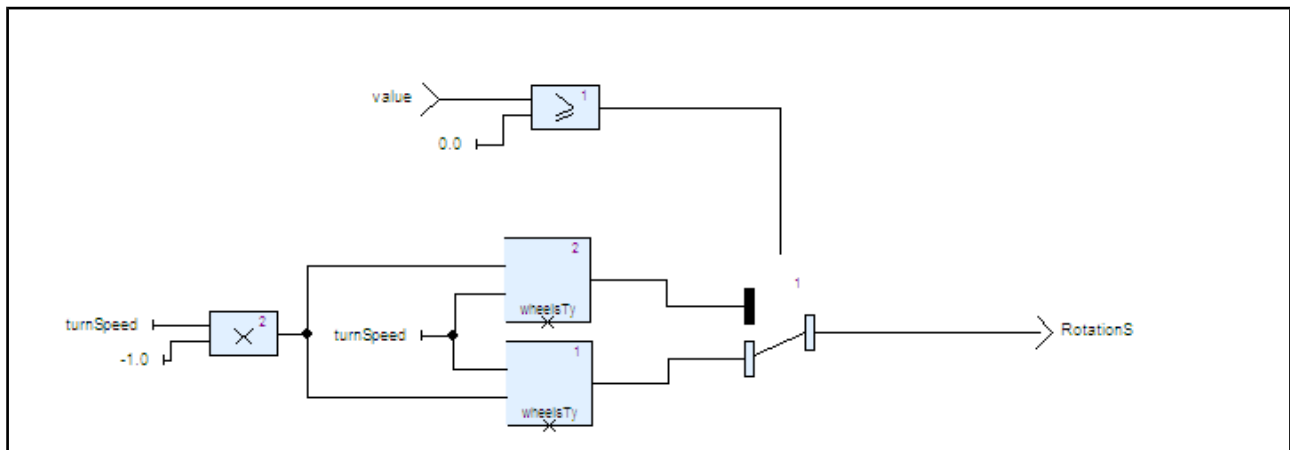


Figure 11 : Opérateur Turn

Calcul du nombre de tics nécessaire à une rotation d'un degré alpha

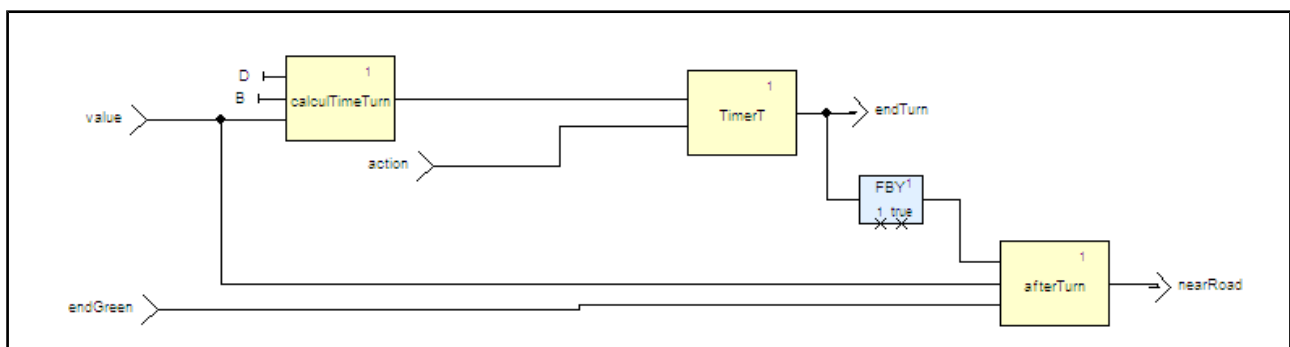


Figure 12 : Opérateur TurnAction

calculTimeTurn : calcule le nombre de tics nécessaire à un angle alpha selon la formule suivante :

$$\frac{\alpha \times B}{vitesse \times D} \text{ si angle } > 0$$

$$-\frac{\alpha \times B}{vitesse \times D} \text{ si angle } < 0$$

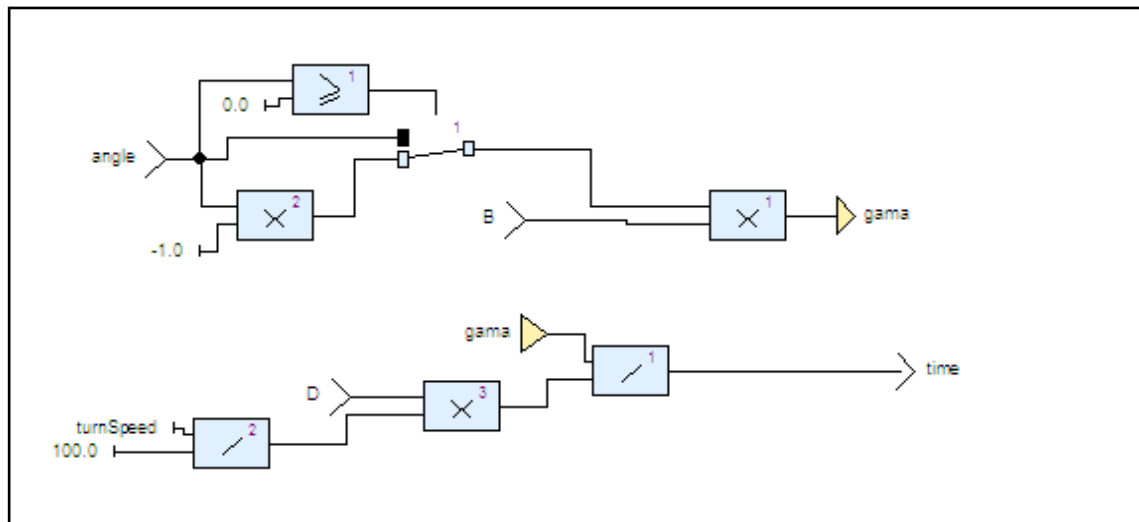


Figure 13: Opérateur « calculTimeTurn »

TimerT : renvoie la variable endTurn pour arrêter la rotation après le nombre de tics retourné par la fonction susmentionnée

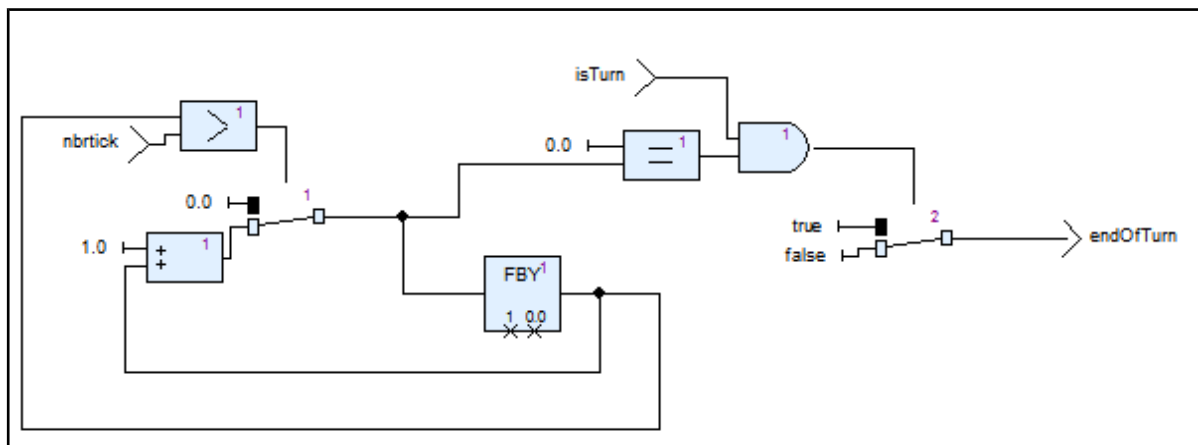


Figure 14 : Opérateur « TimerT »

Opérateur AfterTurn

Problème rencontré sur la map5 : Après avoir tourné d'un angle négatif $\alpha < 0$ le véhicule continue à détecter le magenta de l'ancien chemin c'est la que le PID vient redresser le véhicule pour le ramener de nouveau sur l'ancien chemin. La solution serait de laisser le véhicule avancer un certain moment ,le temps qu'il quitte le magenta ($\alpha < 0$) ou cyan ($\alpha > 0$) de l'ancien chemin.

AfterTurn vient justement régler ce problème, il va compter le temps nécessaire au véhicule pour sortir complètement de l'ancien chemin en fonction de sa vitesse.

Après plusieurs tests nous sommes arrivés à déduire que :

$$\text{largeur_bande_magenta} \sim \text{largeur_bande_cyan} \sim 100$$

Détection d'obstacles

Cet opérateur détecte un obstacle et s'arrête à une distance 12 de cet obstacle, il détecte également la couleur du feu rouge.

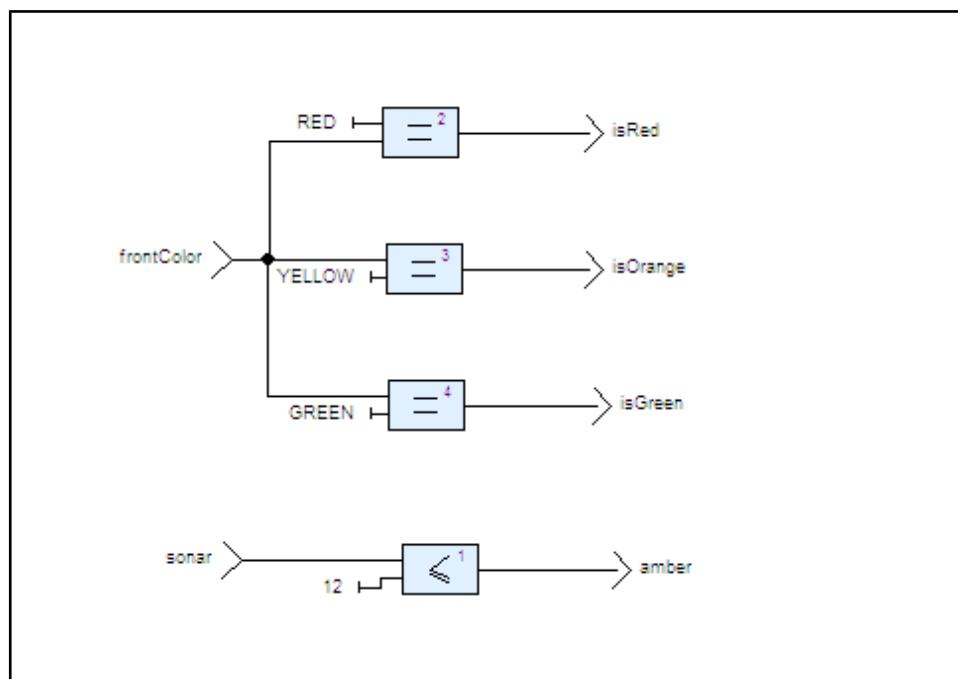


Figure 16 : Opérateur FrontDetect

Arrêt du véhicule

Opérateur stopAction

L'opérateur permet d'arrêter du véhicule en donnant la vitesse 0 aux roues du véhicule (lue sur l'itinéraire Stop 0)

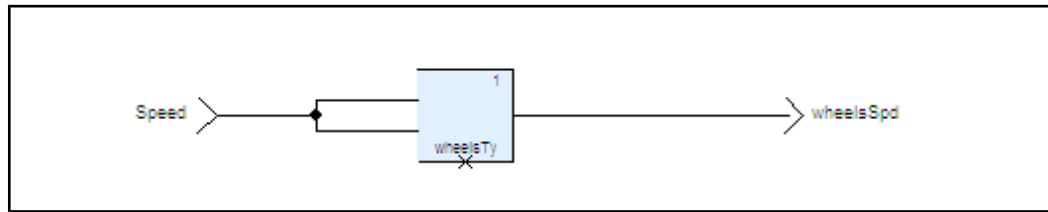


Figure 17 : Opérateur StopAcion

Résultat obtenus

- Le véhicule suit la route, et corrige sa trajectoire en cas de déviation
- Respecte vitesse maximale
- Lit les instructions et tourne avec le bon degré
- Couleurs de signalisation correctement interprétées
- Détection des obstacles et arrêt jusqu'à ce l'obstacle ait disparu

Les scores obtenus sont prélevés à la fin de l'itinéraire

MAP	Score	Qualité
Itinéraire 1	9 898	998
Itinéraire 2	10 000	997
Itinéraire 3	10 500	992
Itinéraire 4	10 796	992
Itinéraire 5	9 266	956
Itinéraire 6	5 289	989
Itinéraire 7	2296	989
Itinéraire 8	11 000	
Itinéraire 9	Ne détecte jamais le dernier point vert sur la route dernière intersection la seule couleur détectée « bleue » redescend à 5463 (c'est peut être problème de MAP → 2352	997
Itinéraire 10	11 000	986
Itinéraire >11	Vidéos indisponibles, point de départ inconnu	

Problèmes rencontrés

1. Lorsque le véhicule tourne sur une intersection selon un degré alpha si le véhicule n'était pas complètement sur la ligne bleue au moment de tourner le véhicule avec un angle de rotation qui ne le met pas exactement sur la ligne bleue, il continue donc de détecter les couleurs magenta ou cyan de l'ancien chemin
le problème soulevé est que le PID remet le véhicule dans la trajectoire initiale.

Solution :

- Optimisation dans les calculs du nombre de tics nécessaires à la rotation
- Quand le véhicule finit de tourner il avance d'une distance de la largeur de la bande magenta (ou cyan) afin d'être sûr que l'ancien chemin est complètement quitté

2. Le véhicule détecte toujours du bleu avec son capteur ventral, sauf dans le cas où il détecte les couleurs de signalisation sur la route, rouge(255,0,0) ou vert(0,225,0). Donc il y a possibilité de division par zéro dans le calcul d'erreur.

Solution : l'automate va dans l'état goAhead et ne fait donc aucun calcul d'erreur dans le GoAction et donc pas de division par zéro.

3. Map 9 dernier turn le roadDetector ne détecte pas le vert sur la route, il se pourrait qu'il y ait un problème sur la map