

TP FPGA

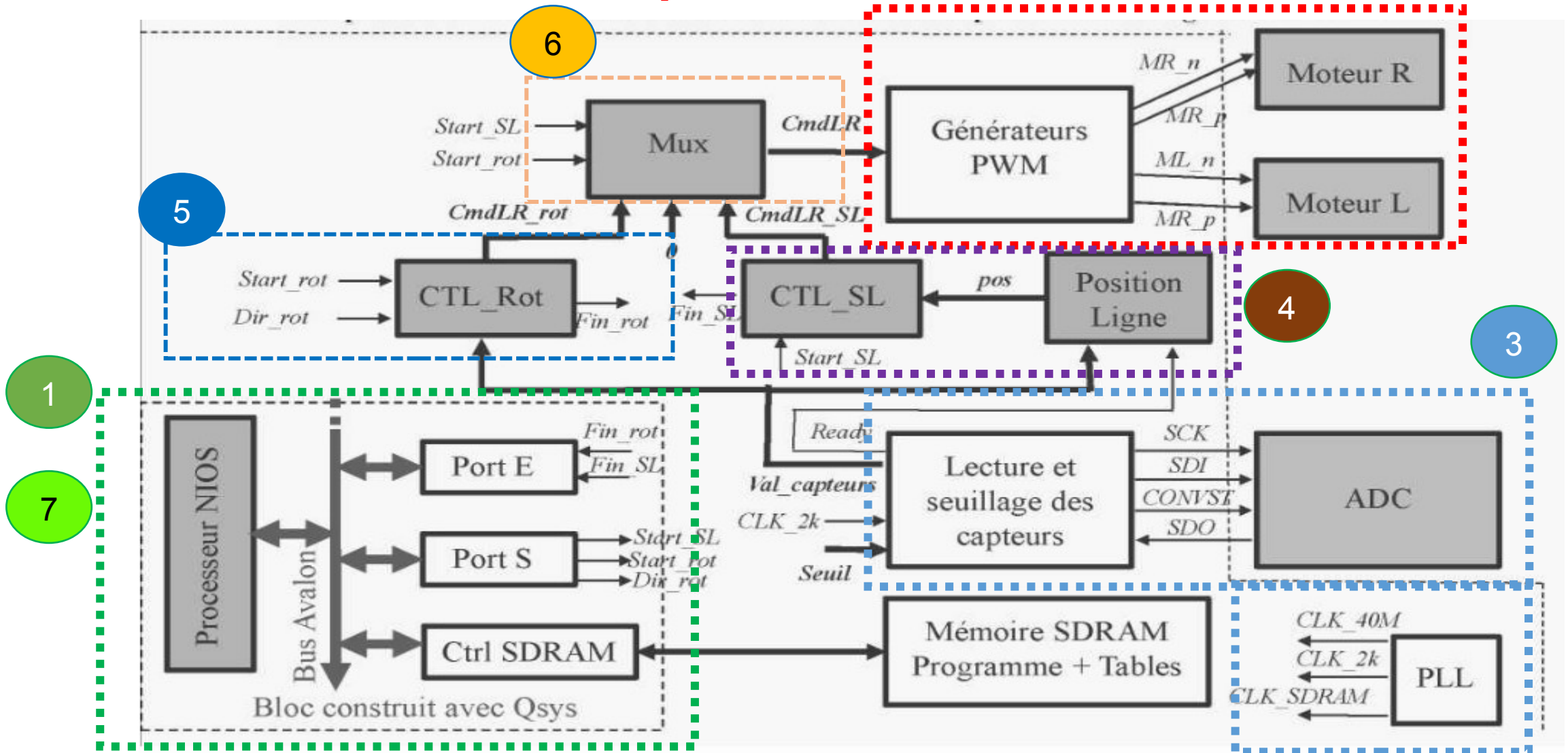
09/02/2026

Différentes étapes

1. Implémenter une architecture NIOS-SDRAM-PIO 20 Pts
2. Caractérisation des moteurs 20 Pts
3. Calcul de la position du robot 20 Pts
4. Suivi de ligne 20 Pts
5. Gestion de la rotation 20 Pts
6. Gestion des aller-retours 20 Pts

Architecture à Implémenter

2



1. Implémenter une architecture NIOS-SDRAM-PIO

1. Étudier les blocs VHDL Fournis ([répertoire « Modules »](#))
2. Dédire l'architecture pour chaque bloc
3. Lier les blocs précédents à l'architecture à Implémenter
4. Implémenter une architecture NIOS-SDRAM-PIO

Utiliser :

- Document 1 - ([3-Introduction_to_the_Altera_Qsys_Tool.pdf](#))
 - Document 2 - ([4-Using_the_SDRAM.pdf](#))
5. Dédire un programme C pouvant piloter un PIO

2- Caractérisation des moteurs

Implémenter le bloc  . Le coupler au bloc 

Modifier le programme C produit précédemment afin de déterminer :

1. **Vitesse minimale** à donner au robot pour que les roues commencent à tourner, le robot hors sol.
2. **Vitesse minimal** à donner au robot pour qu'il commence à se déplacer lorsqu'il est à l'arrêt posé au sol :
 - sans piles
 - avec piles.

Ajuster éventuellement les consignes pour chaque roue pour que le robot se déplace en ligne droite
3. Vitesse minimal que l'on peut donner au robot en mouvement sans qu'une des roues ne s'arrête. Déduire l'hystérésis éventuel.

3- Calcul de la position du robot

Implémenter le bloc 3. Le valider à l'aide du bloc 1.

- Créer une description VHDL permettant d'analyser le vecteur seuillé afin de déterminer si la ligne est présente sous les capteurs

Ce Bloc opérateur peut être construit de manière totalement combinatoire.

- Sachant que nous n'avons que 7 capteurs, pourquoi avoir utilisé la solution se basant sur l'erreur de position de la ligne entre -6 et +6 ?

4- Suivi de ligne

Créer en VHDL un automate permettant au robot de suivre une ligne noire (implémentation du bloc 4) sachant :

- L'automate devra attendre qu'un signal *start_SL* passe à 1 pour effectuer le suivi de ligne
- Les commandes PWM sur les 2 moteurs se fera de la manière suivante :
$$\text{PWMdroit} = \text{Constante} + \text{BIAS}(\text{POSL})$$
$$\text{PWMgauche} = \text{Constante} - \text{BIAS}(\text{POSL})$$

Constante est celle déterminée en

2

- Lorsqu'il n'y a plus de ligne (absence d'une séquence de 1 successifs sur le capteur), le robot devra s'arrêter avec un signal *fin_SL* à 1
- Effectuer un suivi de ligne avec *start_SL*=1. On affichera sur des LED pendant l'expérience les signaux *start_SL* et *fin_SL*

5- Gestion de la rotation

5

- Créer une description VHDL d'un automate (implémentation du bloc 5) permettant au robot de tourner sur place lorsque qu'il perdra la ligne noire. Il devra alors tourner jusqu'à retrouver la ligne et se positionner à peu près au milieu.
- L'automate devra attendre qu'un signal *start_rot* passe à 1 pour démarrer la rotation.
- La direction dans laquelle il tournera sera définie par le signal *dir_rot* (à gauche ou à droite).
- Lorsque le robot est positionné au milieu de la ligne, la rotation s'arrête et un signal *fin_rot* passe à 1.
- Valider en lançant cet automate de rotation lorsqu'on ne verra plus la ligne noire (*fin_SL*).
- Afficher ces signaux sur les LEDs pendant l'expérience. Valider la consistance de cet automate sur les deux sens de rotation.

6- Gestion des aller-retours

Le processeur NIOS est utilisé comme superviseur, créer un automate gérant le robot pour :

- A l'initialisation, le robot doit rester à l'arrêt tant qu'il ne détecte pas de ligne
- Dès qu'il détecte la ligne, il doit passer en mode suivi de ligne et y rester tant que `fin_SL=0`
- Quand `fin_SL=1`, il doit s'arrêter pendant une seconde puis passer en mode rotation. Il y reste tant que `fin_rot=0` puis s'arrête à nouveau 1 seconde avant de repasser en mode suivi de ligne
- Valider en faisant faire des aller-retours au robot en gérant le sens de rotation