

# Console SU-EE100 sur FPGA.

**Réalisé par : Hiba KHECHAI & Zahra BENSLIMANE .**

**2020/2021**

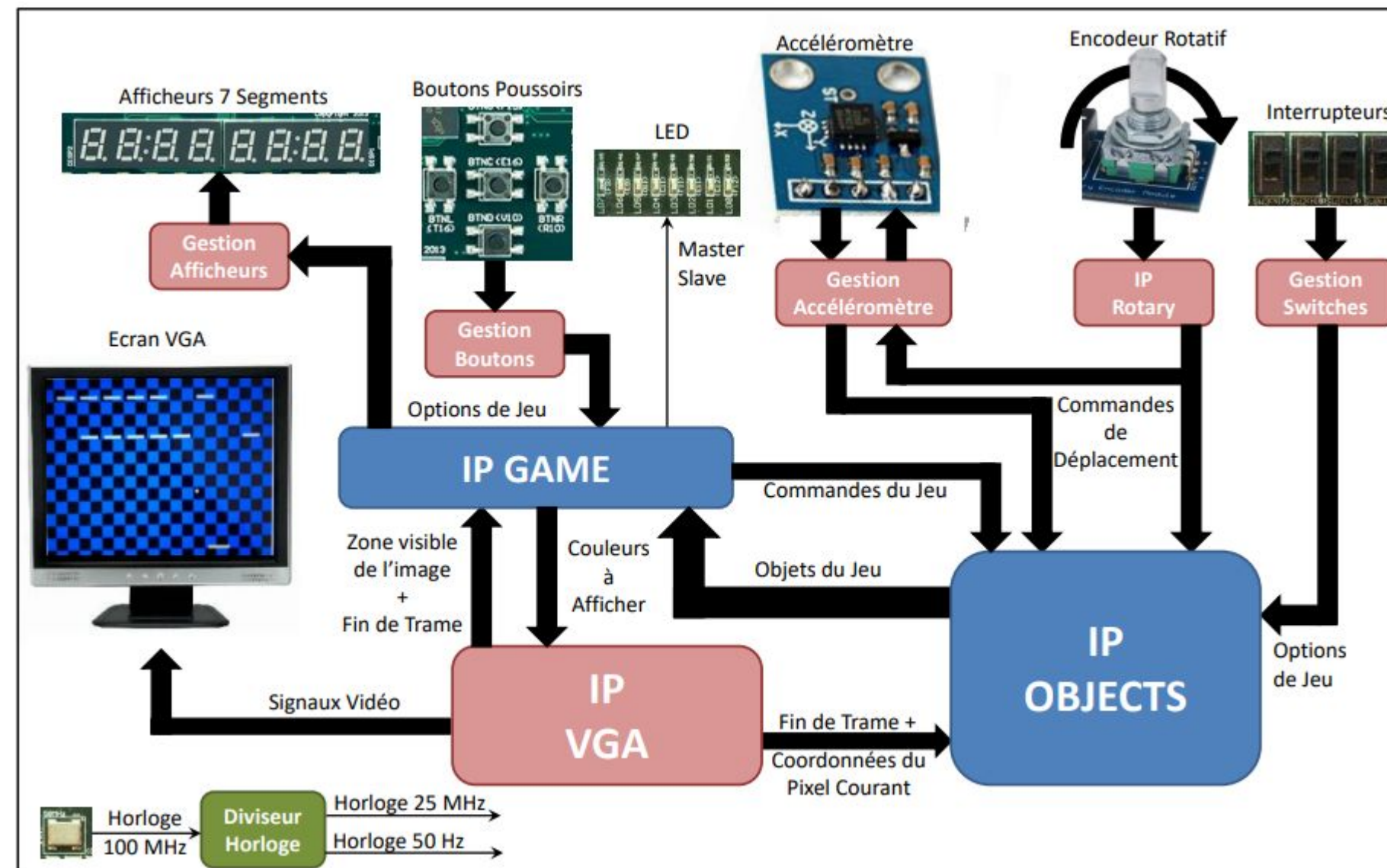
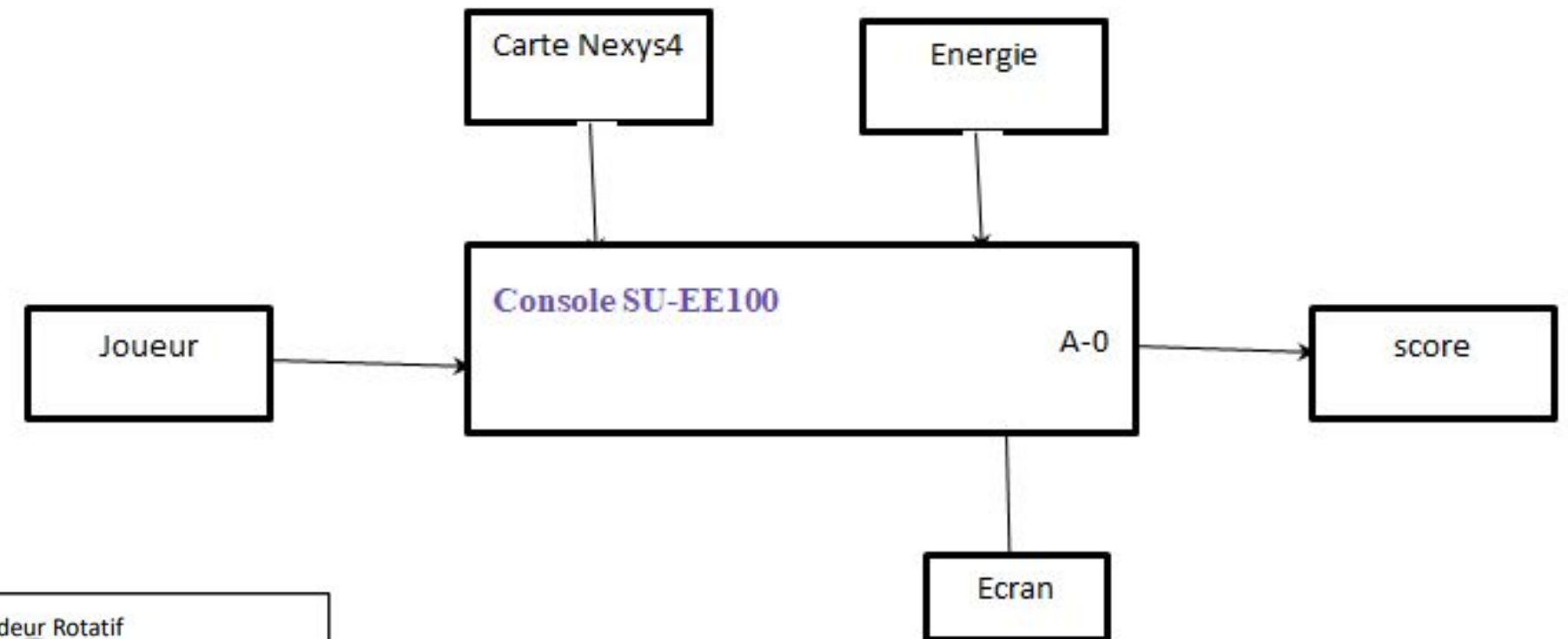
# Sommaire De La Présentation

- Architecture globale de la console.
- Gestion de l'encodeur rotatif.
  - Machine à état du module IP Rotary.
  - chronogramme du IP rotary.
- Gestion de l'état de la partie (Pause, gagné, perdu).
  - Module Mode.
- Bilan du TP.
- Conclusion.

## Diagramme SADT

# STRUCTURED ANALYSIS AND DESIGN TECHNIQUE

Le niveau A-0 du jeu.



Architecture globale de la console



# IP rotary

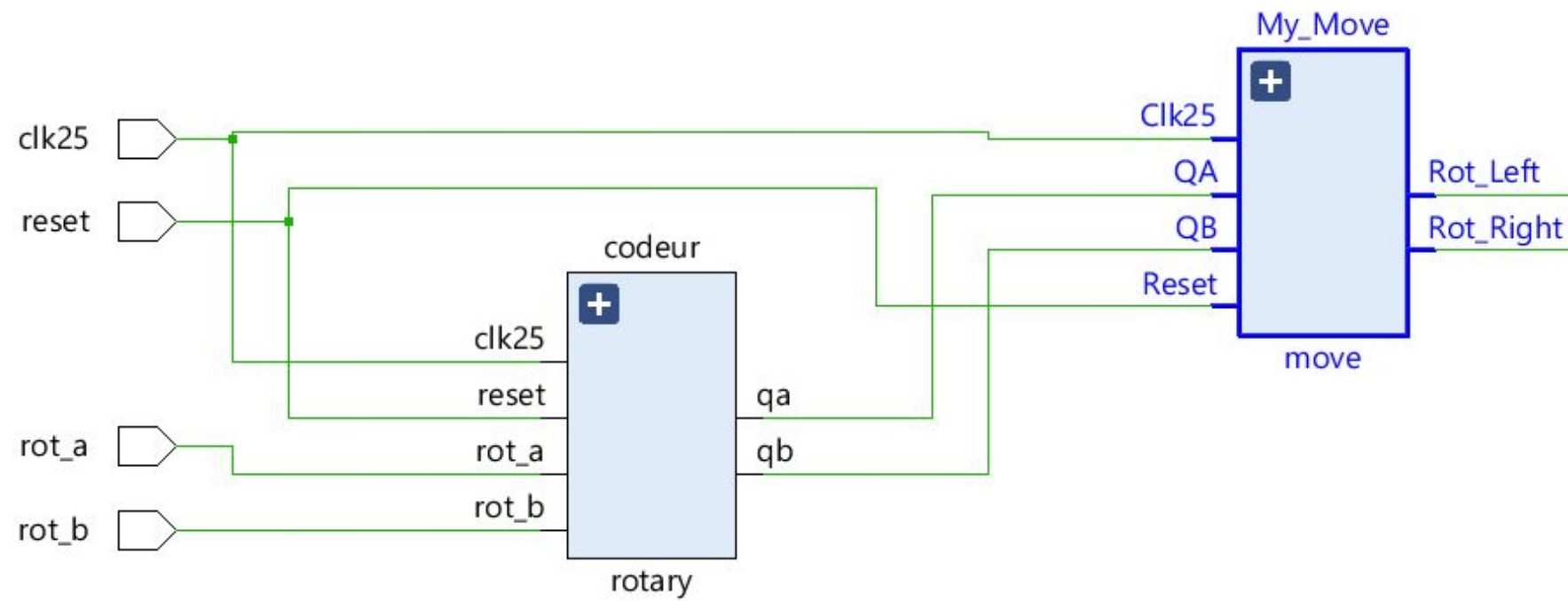
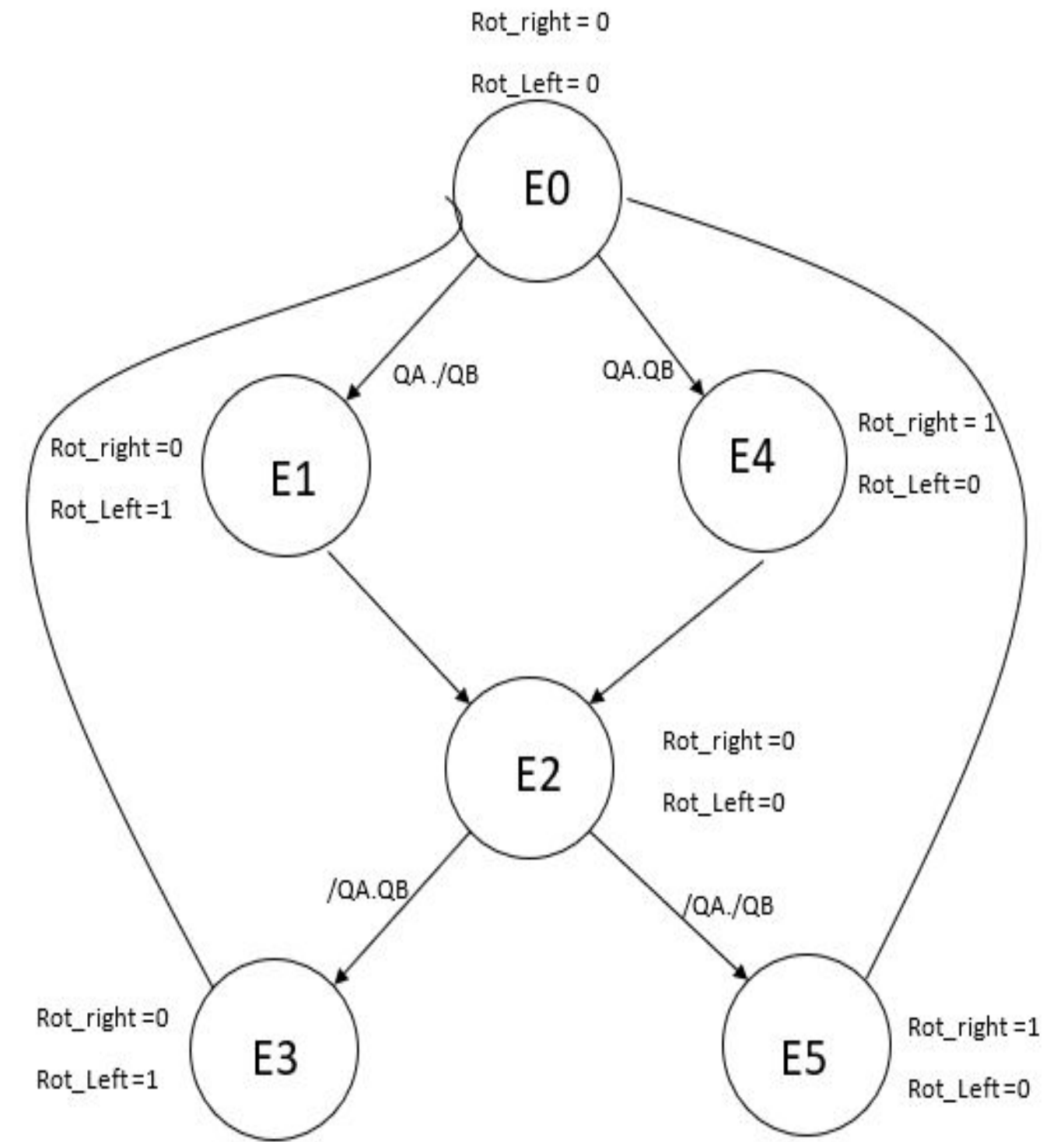


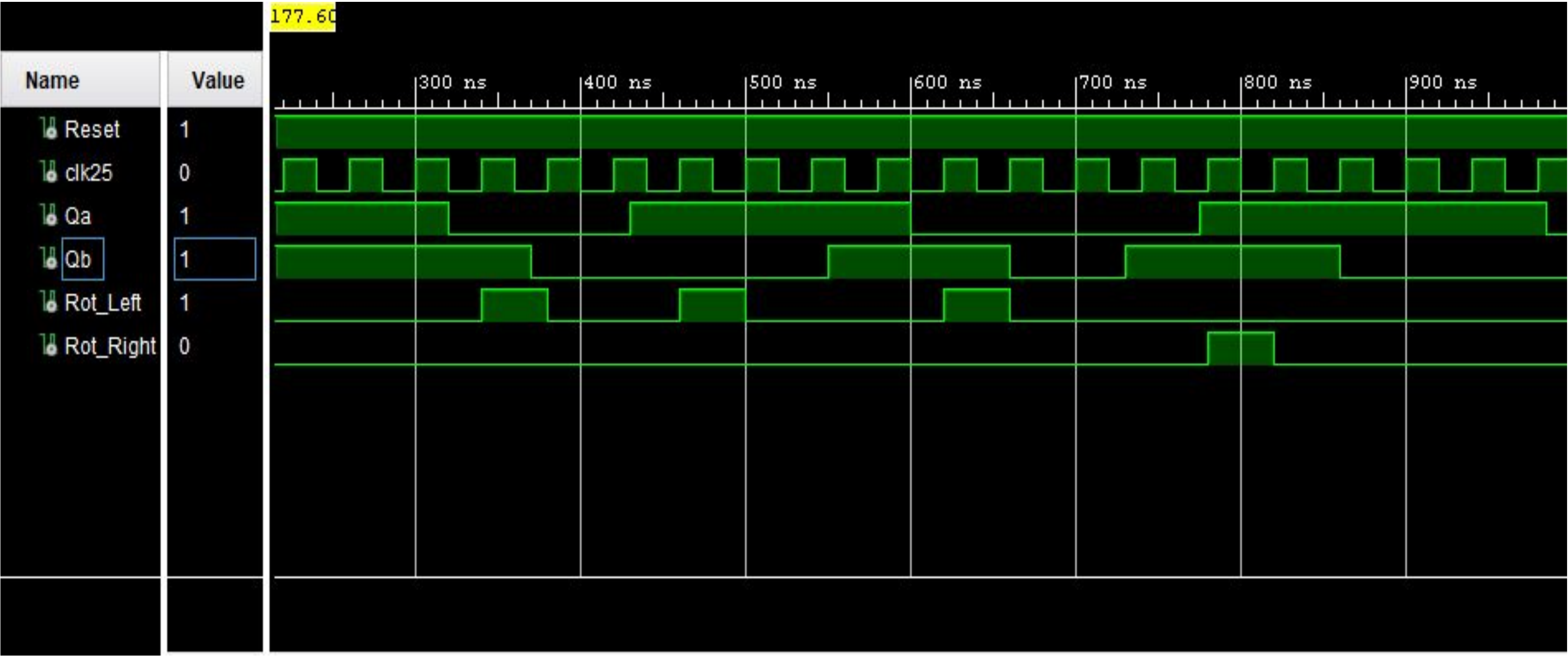
schéma pour illustrer IP rotary via Vivado



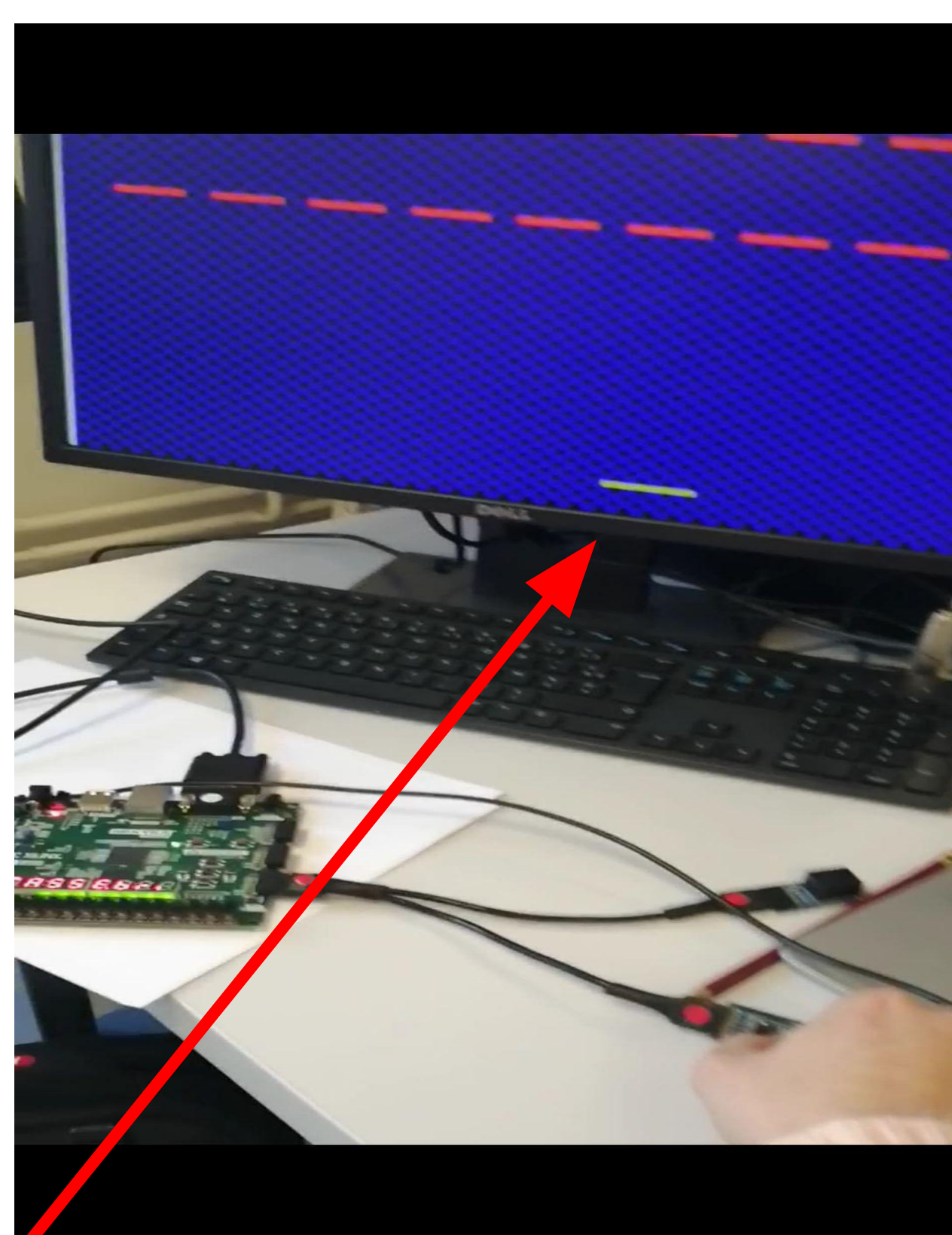
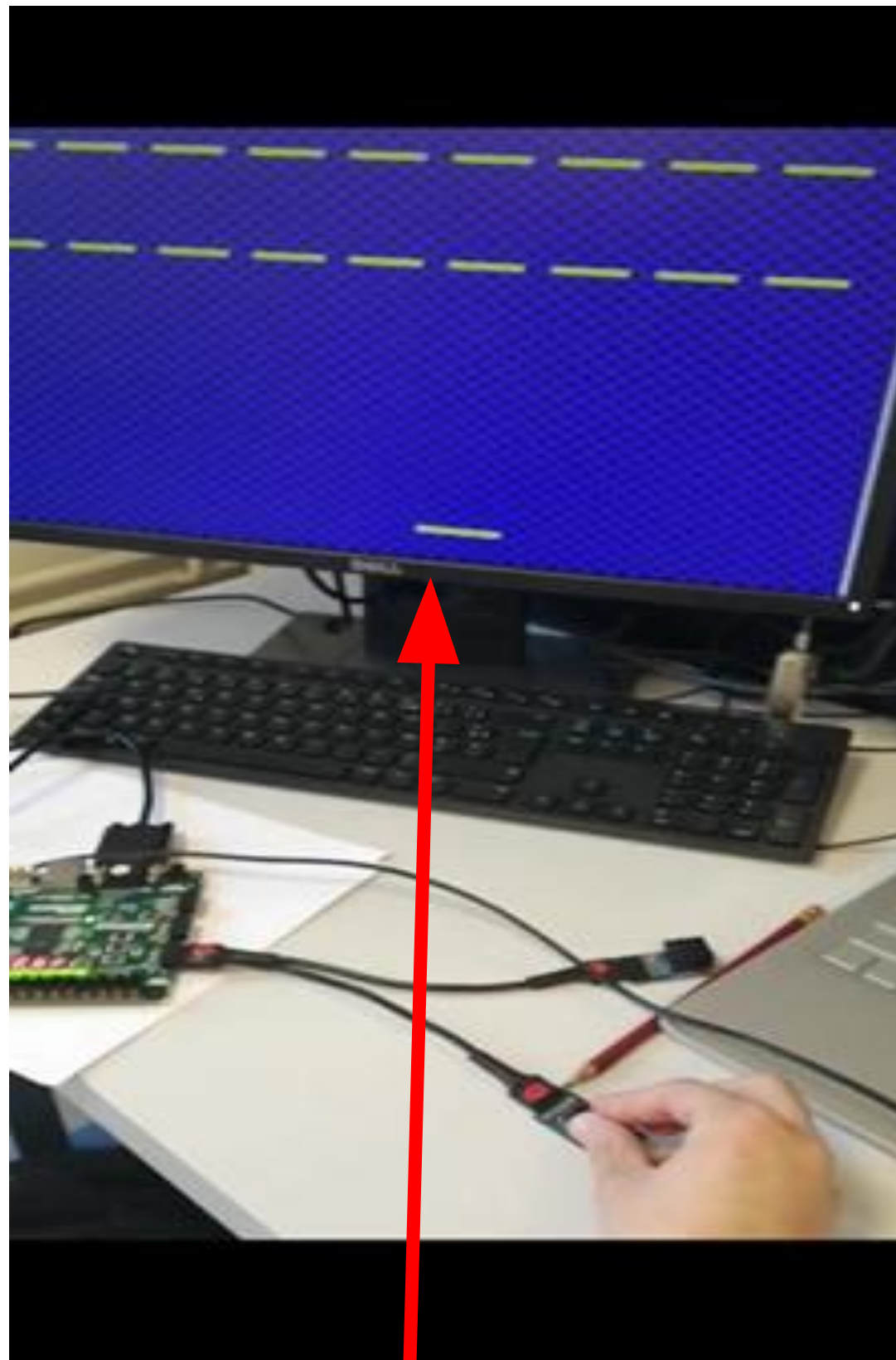
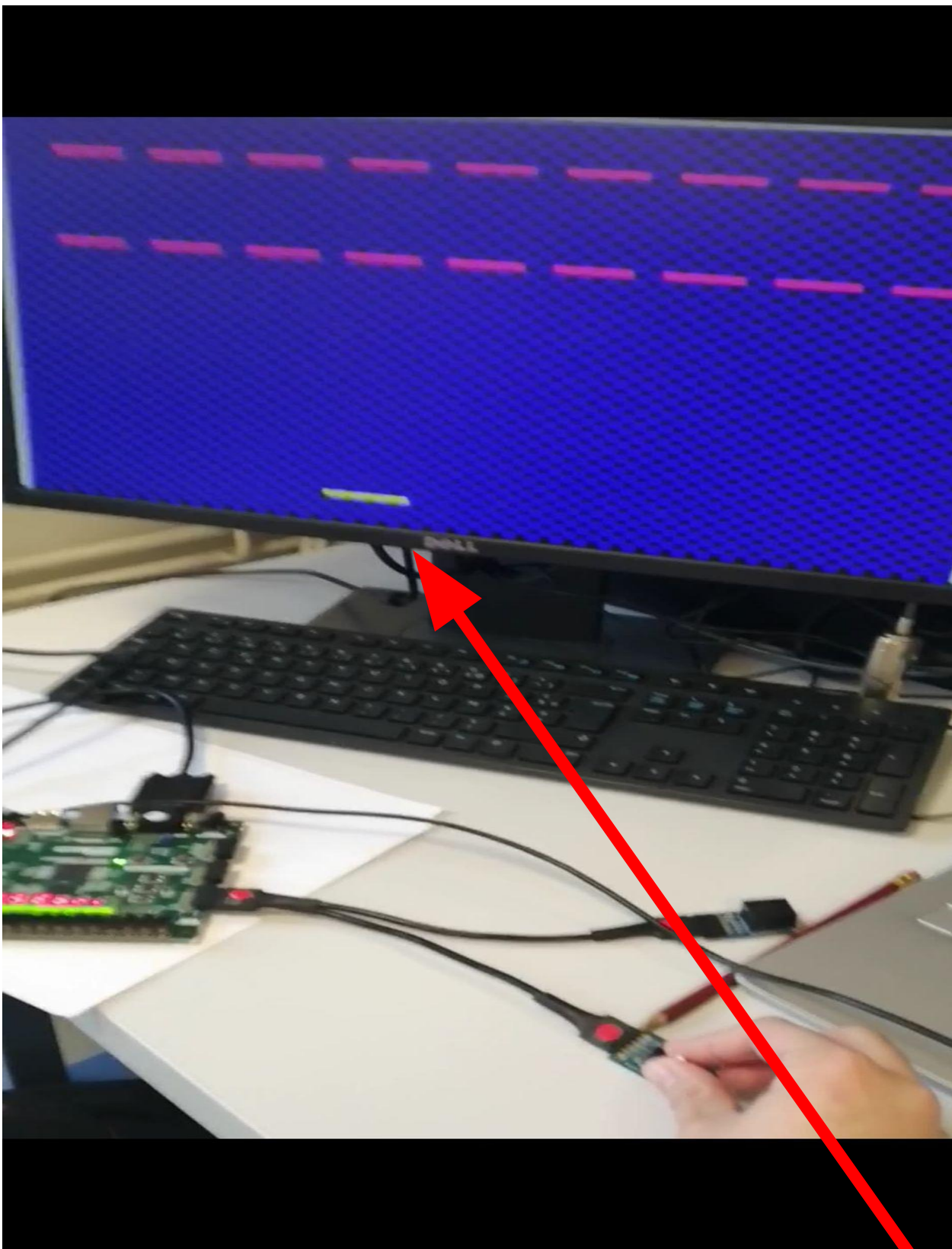
machine à état du module move



# Chronogramme de Move







05

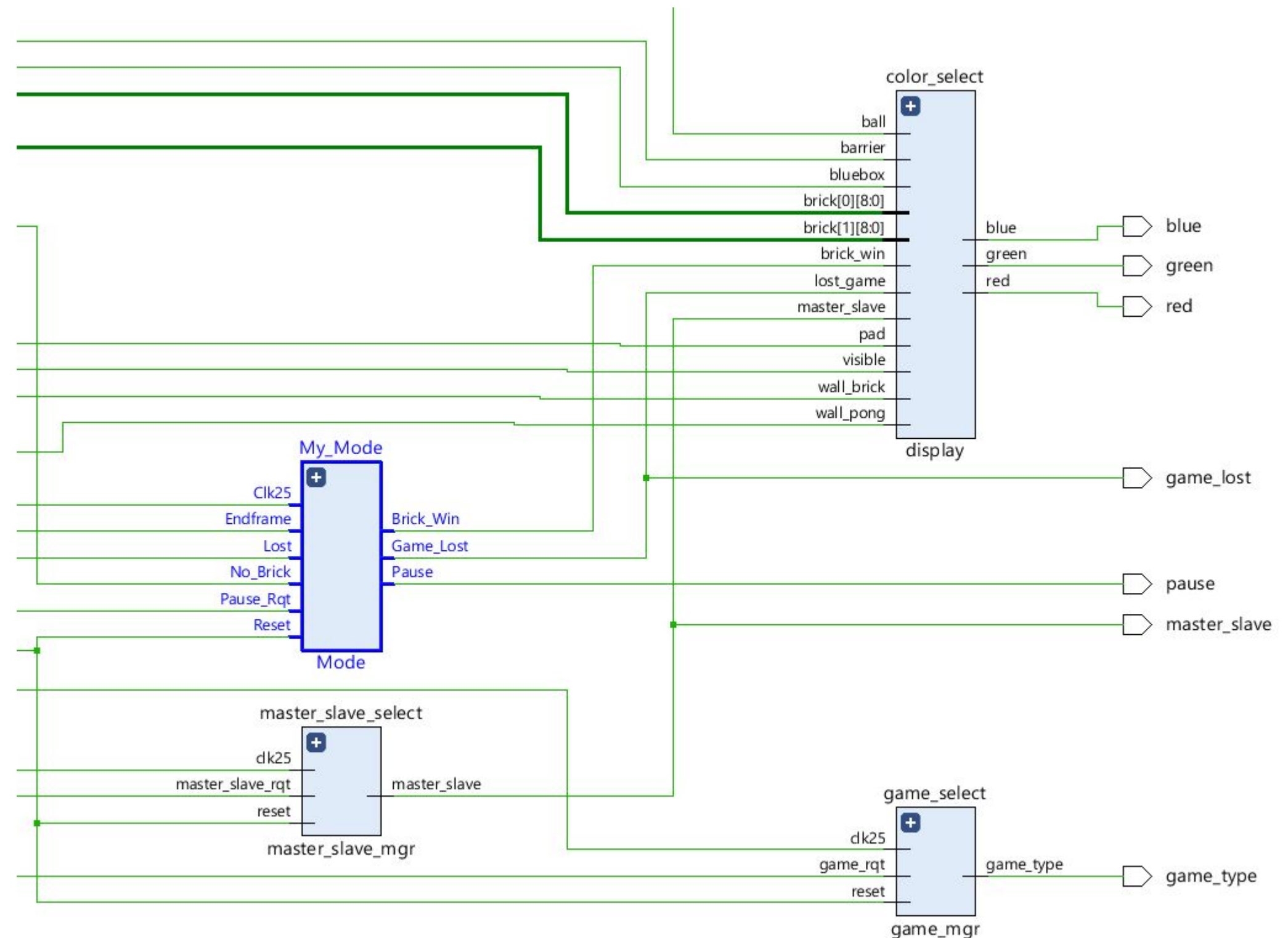
déplacement de la raquette à l'aide de l'encodeur rotatif



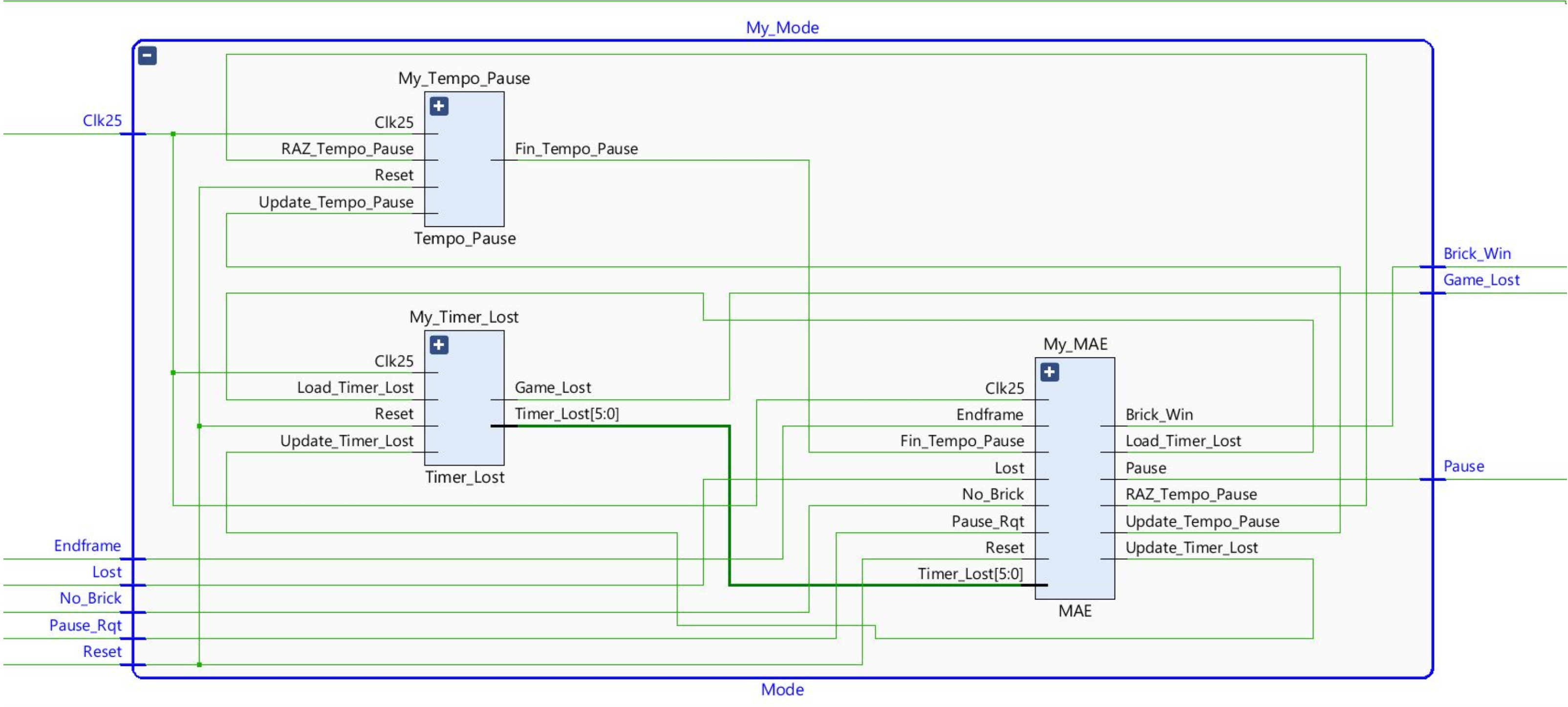
# Gestion du Jeu & Gestion de l'état de la partie

## Gestion du Jeu : Game

- Sélectionner le mode Console ou Manette pour la SU-EE100
- Sélectionner le jeu actif (Pong ou Casse Briques)
- Gestion du mode Pause, gagné et perdu.
- Transmission des signaux couleurs l'IP VGA



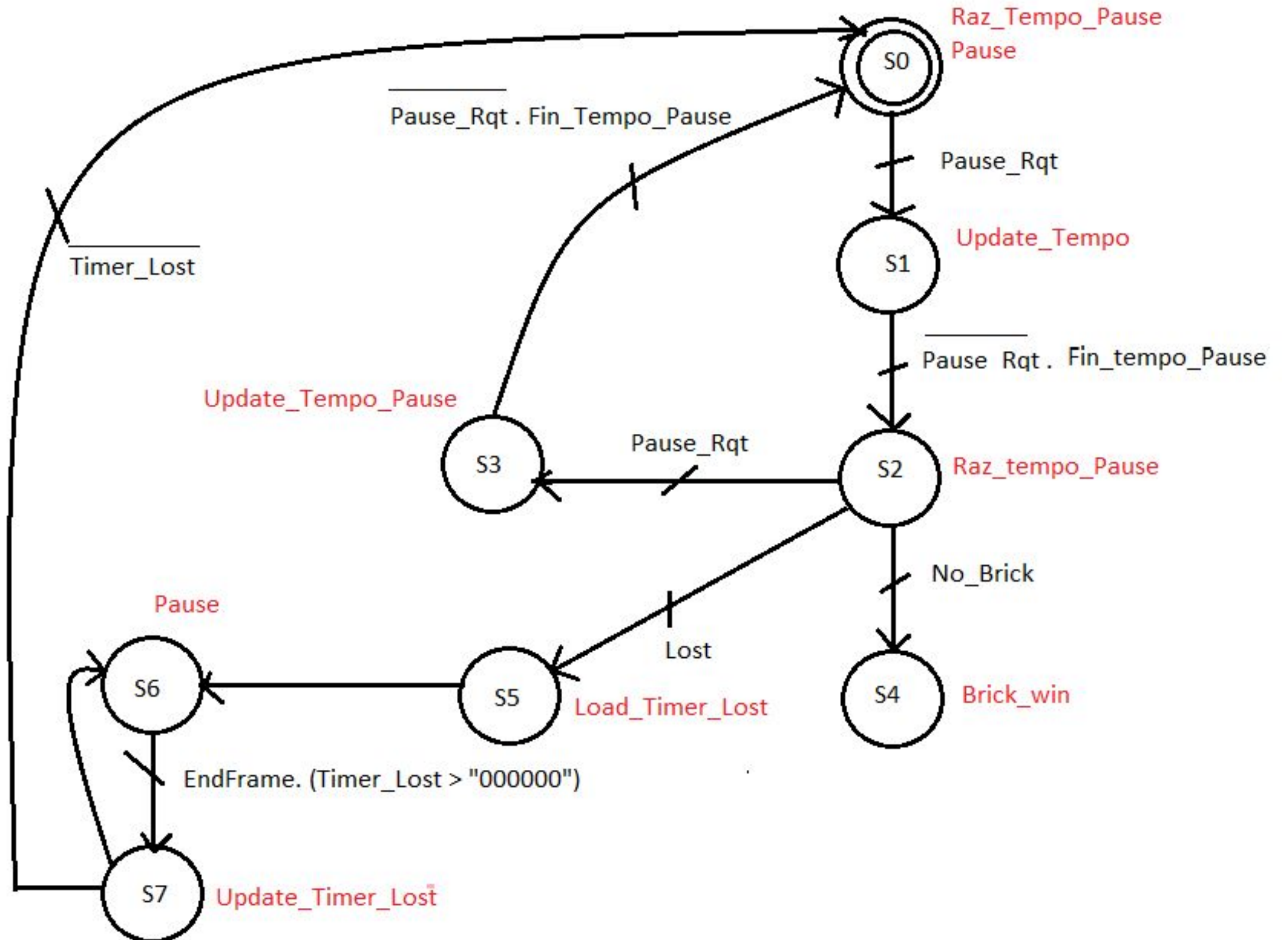
# Schéma élaboré par vivado



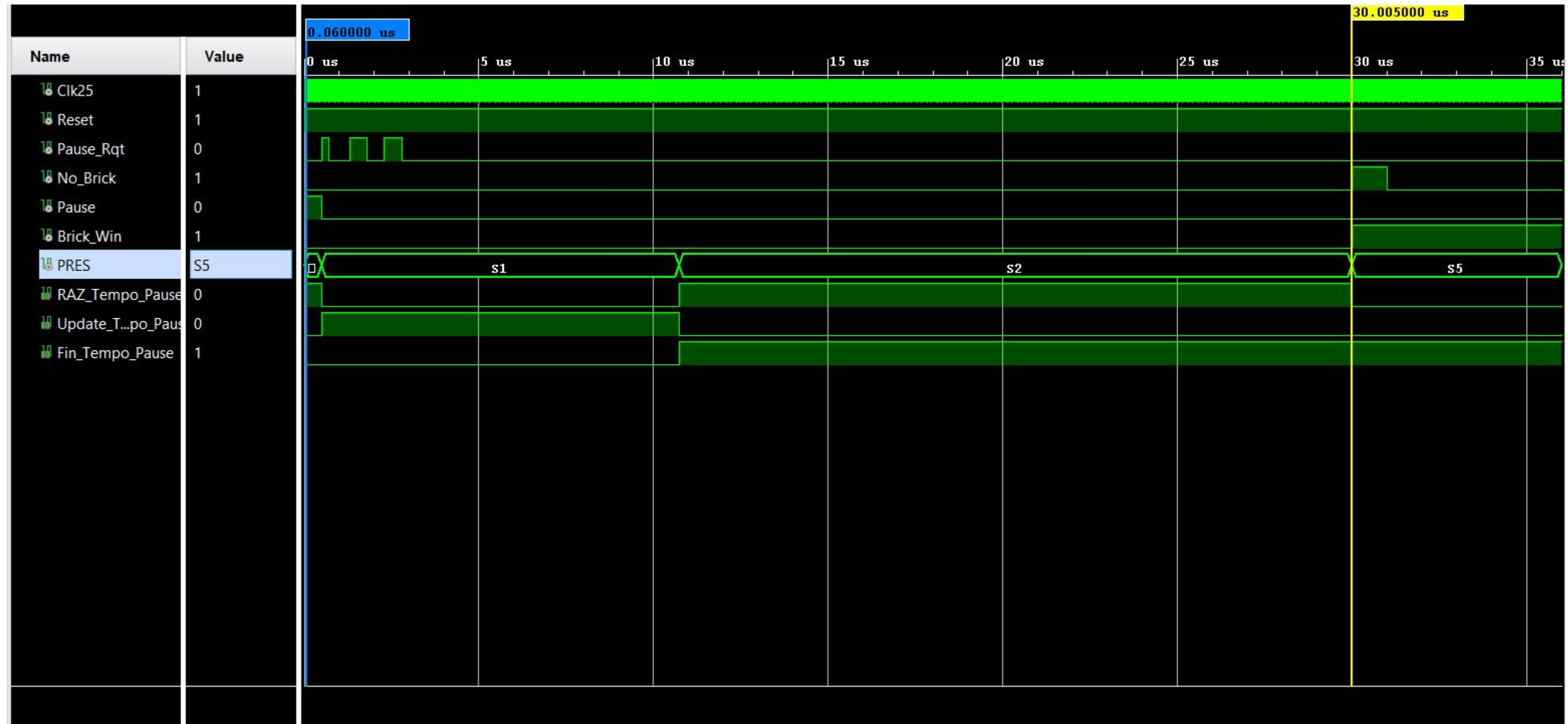


## Machine à état : Mode

- Génération des deux signaux Brick\_Win et Pause
- Gestion des deux compteurs Tempo Pause et Timer Lost

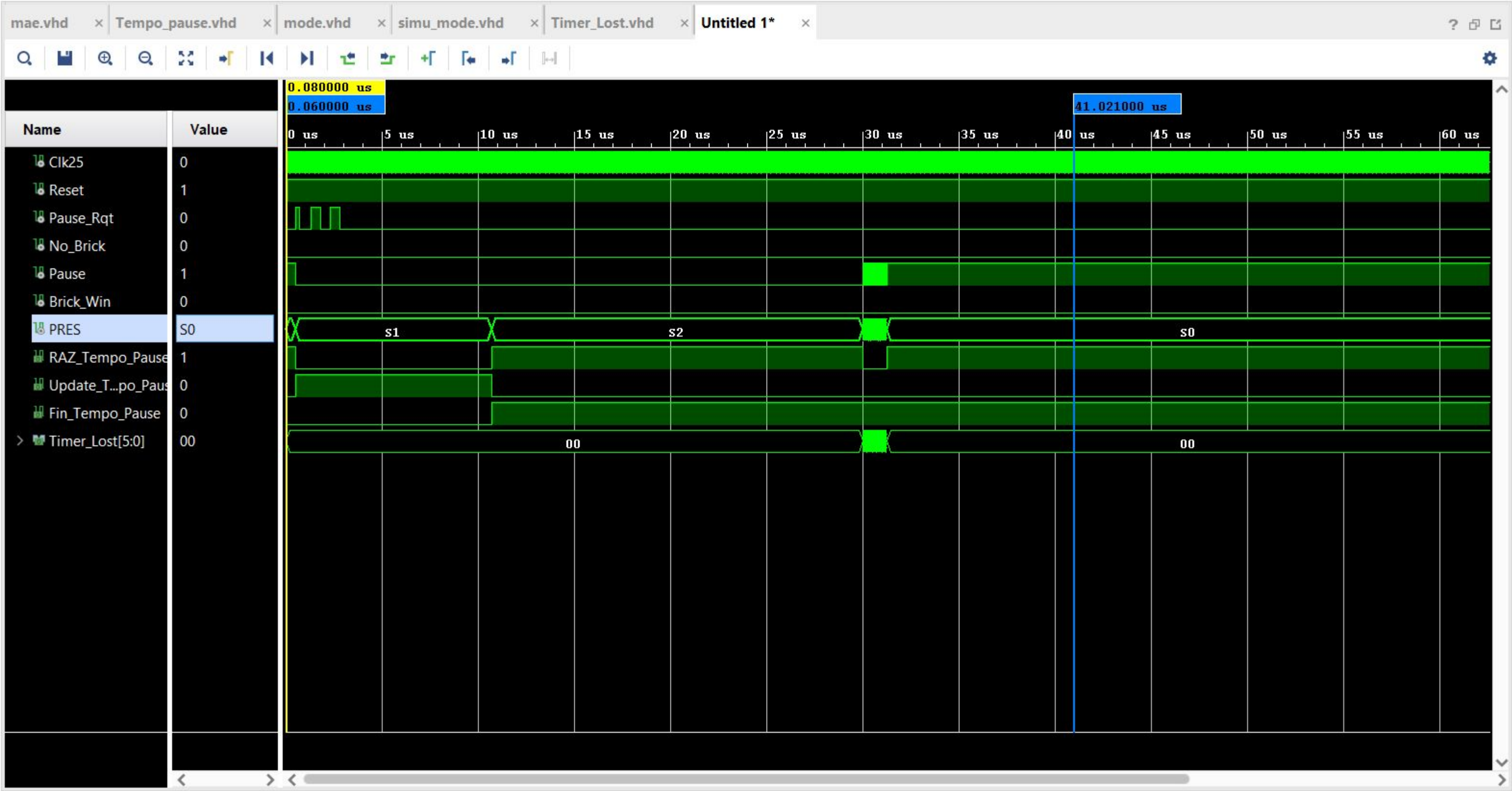


## Simulation d'une partie gagnée : Affichage d'un écran vert pour une durée indéterminée



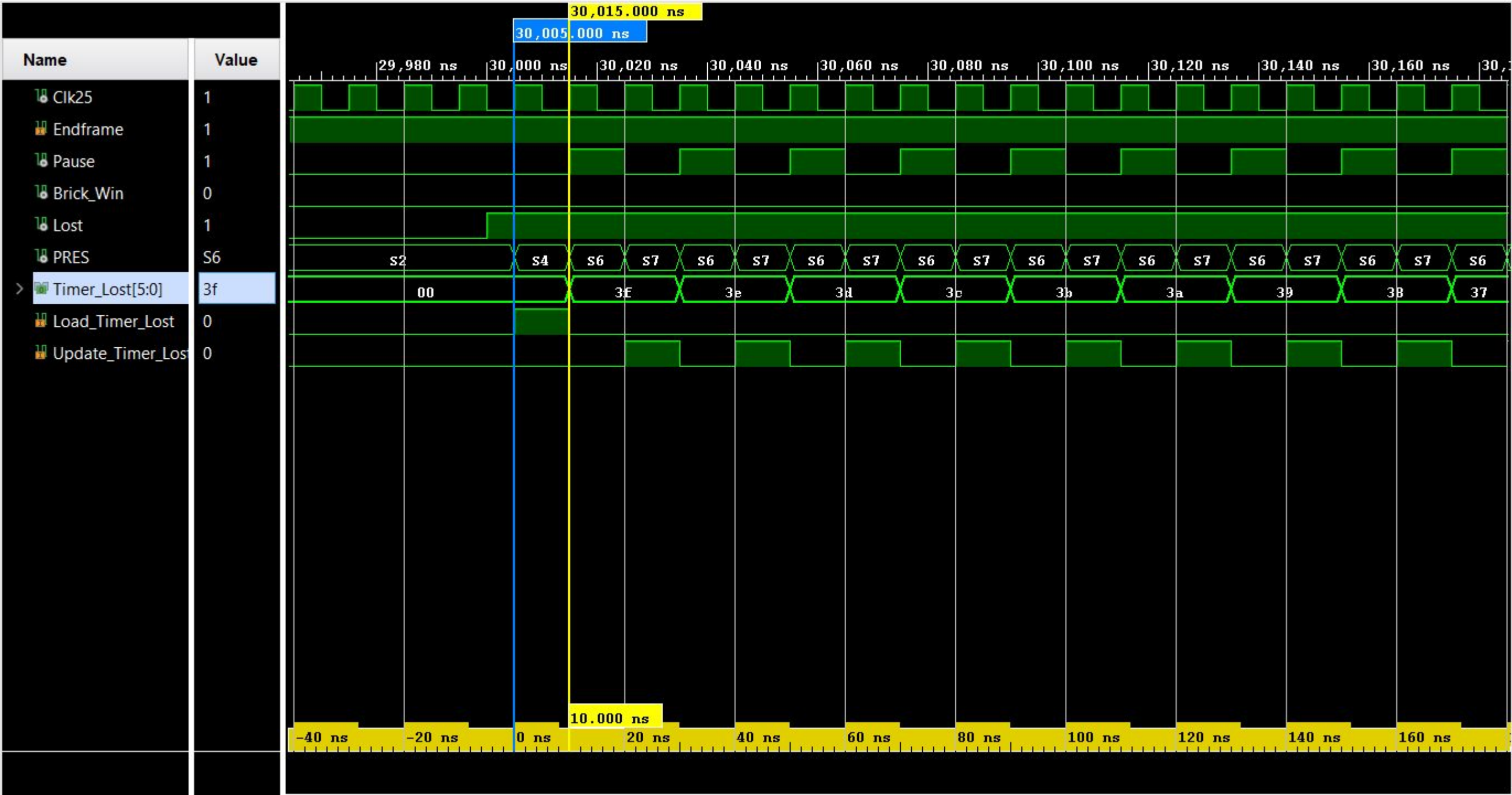
# Simulation d'une partie perdue

10

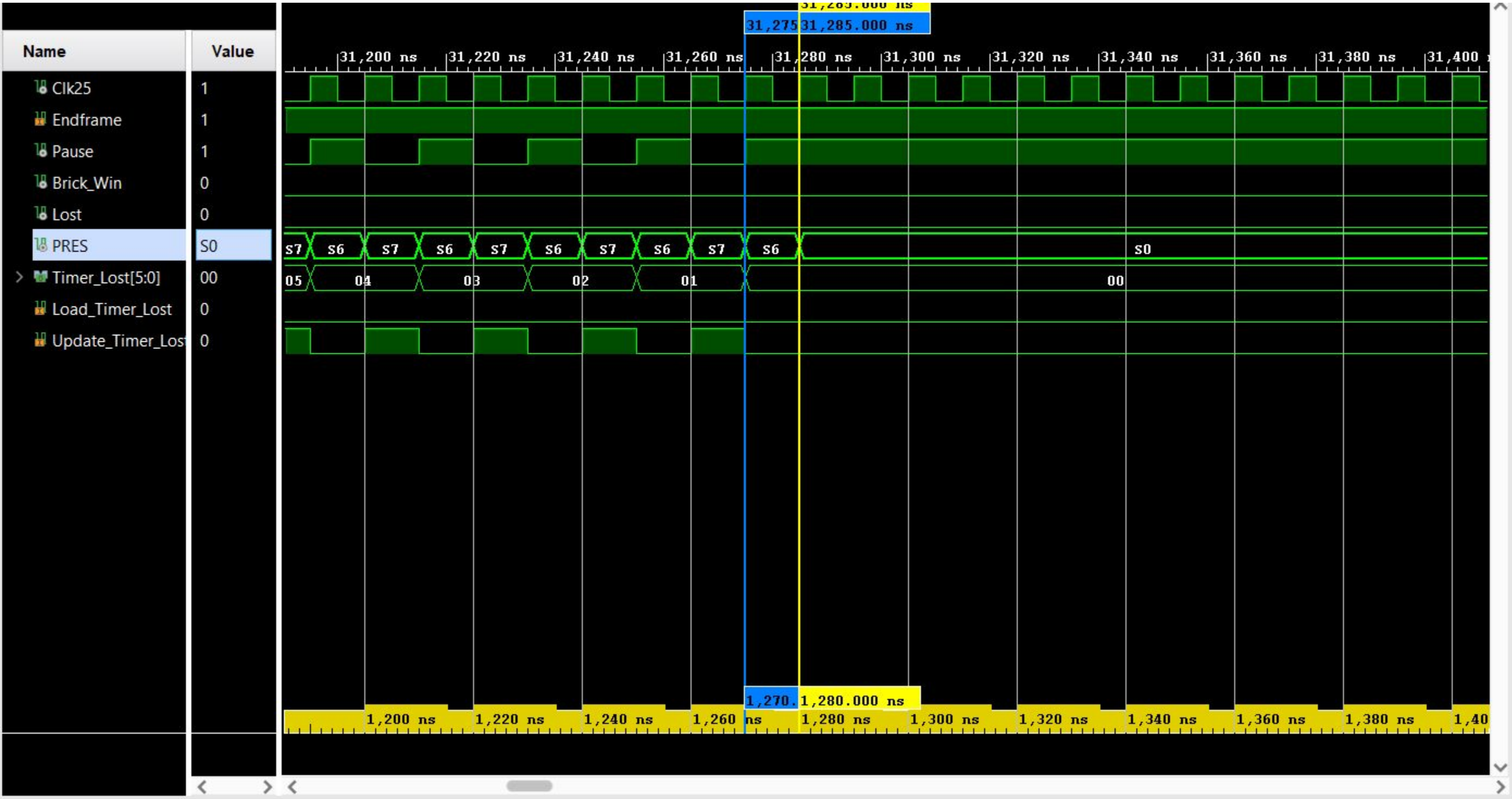




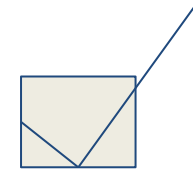
Affichage d'un écran rouge pour une durée de 64 images.



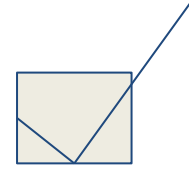
# Retour à l'état initial S0



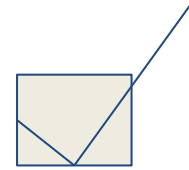
# Bilan & Conclusions



Tache 01 : MovingColors



Tache 02 : IP Rotary



Tache 03 : Mode [ gestion du jeu ]



MERCI DE VOTRE ATTENTION

A series of thin, dark blue, concentric wavy lines that curve from the top right towards the bottom right, creating a sense of motion or a stylized wave pattern.