

# EN202 Projet Blackjack

## VHDL - FPGA

Besson Thomas



Année Universitaire 2025-2026

# Contexte & objectifs

## But du projet

Implémenter une partie de **Blackjack** sur FPGA (Nexys A7-100T) en **VHDL**.

## Rappel des règles

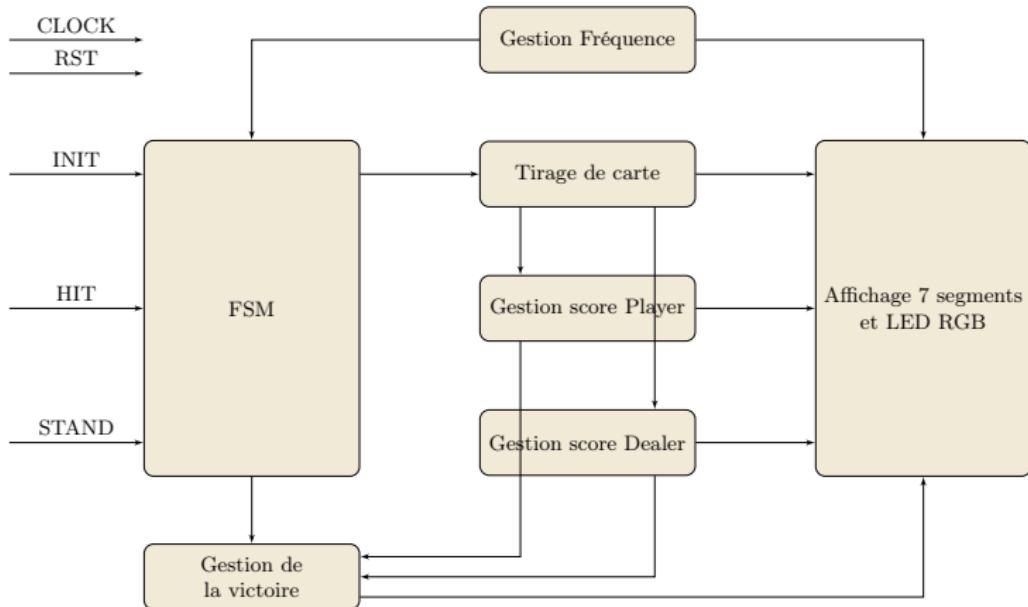
- Objectif : atteindre **21** sans dépasser (bust).
- Le joueur choisit **HIT** (tirer) ou **STAND** (rester).
- Le croupier tire jusqu'à **score  $\geq 17$** .



## Exigences principales

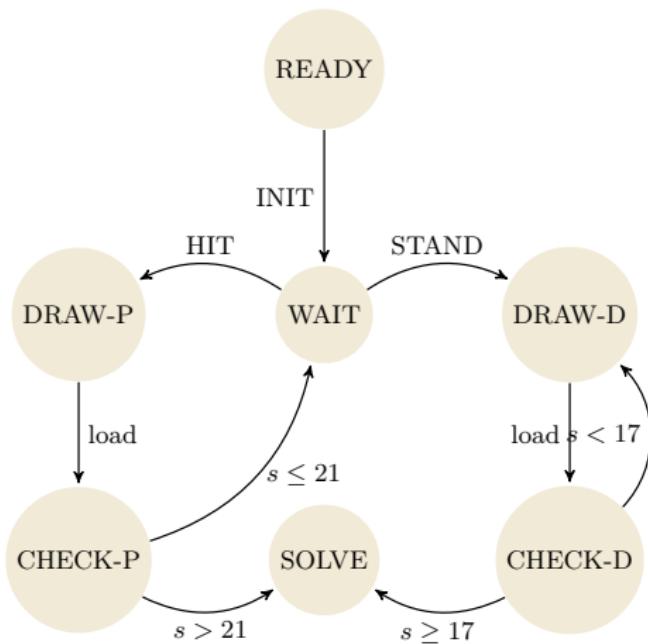
- 1 joueur vs 1 croupier sans mises, interface par **boutons** (INIT / HIT / STAND)
- Tirage **pseudo-aléatoire** à chaque HIT
- Calcul et affichage des **scores** (gestion des As)
- Résultat de la donne
- Affichage sur **8 digits 7-seg + LED RGB**

# Architecture générale



CLOCK et RST distribués aux blocs séquentiels (non détaillés sur chaque liaison).

# Diagramme de transition de la machine d'état



Le diagramme est simplifié ; le code détaille le tour croupier en sous-états (**DEALER\_DRAW**, **DEALER\_WAIT**) pour intégrer la temporeisation.

## Principe du random\_hit

Un compteur tourne en continu à 100 MHz. Lors d'un appui **HIT**, la valeur est **capturée** puis mappée vers une valeur de carte.

Le moment d'appui sur HIT dépend de l'utilisateur  $\Rightarrow$  variation temporelle suffisante pour un tirage pseudo-aléatoire exploitable.

## Fonction du score\_manager

Accumulateur de score et gestion des As :

- As tiré : ajouté comme 11
- Si dépassement de 21 : conversion d'un As de 11 à 1 (soustraction de 10) tant que possible

## Bloc combinatoire

Activé uniquement quand `is_final=1`.

### Priorité

- Bust joueur  $\Rightarrow$  victoire croupier
- Bust croupier  $\Rightarrow$  victoire joueur
- Sinon comparaison des scores ; égalité  $\Rightarrow$  draw

# Affichage 7 segments

## Multiplexage

- **CE\_perception**  $\approx 2$  kHz pour balayer les 8 digits (persistance rétinienne), provenant du bloc de gestion de fréquence.
- Répartition : cartes et scores joueur/croupier sur **8 digits**

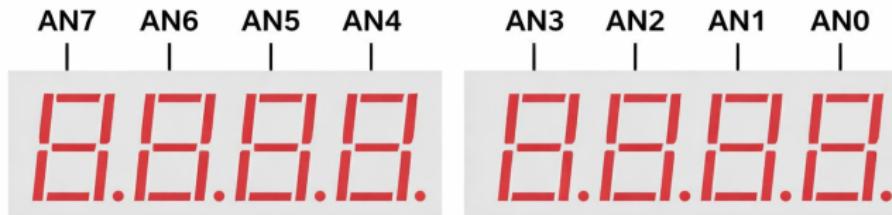


Figure – Afficheur 8 digit, 7 segments

AN7 - AN6 Carte tirée Joueur

AN5 - AN4 Main Joueur

AN1 - AN0 Main Croupier

# Affichage des résultats de la partie

## Objectif

Rendre le verdict **immédiatement lisible** sur carte. **clignotement du perdant** et **LED RGB en PWM**.

### Clignotement du perdant

- Principe : **masquer** uniquement les digits du perdant.
- Implémentation : un **bit d'un compteur** joue le rôle de signal **blink\_sig**.
- Lorsque **blink\_sig=0** et joueur perdant alors on éteint les segments au lieu de la valeur du digit.

### LED RGB en PWM

- Un compteur 8 bits génère **pwm\_on**.
- Couleur selon le verdict :
  - **Victoire joueur** : vert
  - **Défaite joueur** : rouge
  - **Égalité** : blanc (R+G+B)

# Conclusion

## Synthèse cahier des charges

- FSM joueur et croupier
- Hit pseudo-aléatoire
- Scores indépendants et gestion As
- Règle croupier
- Affichage 8 digits + LED RGB

## Bilan

Blackjack fonctionnel sur FPGA avec :

- FSM de contrôle
- Modules dédiés (tirage, score, verdict)
- Affichage multiplexé et LED RGB PWM

## Perspectives

- Affichage sur VGA pour les figures et le jeu.
- Interface utilisateur enrichie (mise, split, etc.)

- [1] Le blackjack, règles et histoire, wikipedia.org
- [2] Nexys A7 Reference Manual, Digilent Inc.