

TP4 : Débogage de l'AMD2901

1. Objectifs
2. Compte-Rendu

Objectifs

Le but de cette séance est de vous apprendre à déboguer un circuit qui sera réutilisé dans les TP suivants. L'intérêt est donc double:

- Au niveau de l'état d'esprit: vous apprendrez, par la pratique, la méthodologie du débogage. Cela vous resservira toujours étant donné que la découverte des erreurs est universelle et touche également des domaines autres que l'informatique ou l'électronique.
- Au niveau des connaissances techniques: vous ferez découvrir (ou revoir) un circuit ayant été déployé dans l'industrie qui sera réutilisé dans les TP suivants.

Pour explorer toutes les fonctionnalités du circuit, vous devrez, au cours de cette séance, valider la vue comportementale qui vous sera fournie. (Deux fichiers VBE par binôme). Vous trouverez la DATA SHEET de ce circuit en bas de la page.

La validation devra être réalisée à l'aide de vecteurs de tests générés avec genpat. Les vecteurs doivent être soigneusement écrits pour vous permettre de déceler un bug insidieusement inséré dans votre fichier comportemental .vbe. Environ 500 patterns suffiront pour debugger votre AMD2901. Pour obtenir la description d'un AMD2901, tapez la commande suivante :

```
/users/enseig/trncomun/TP/2006/TP1/Fichiers/Fournis/get2901.Linux > amd.vbe
```

Remarque : L'Amd2901 que vous allez récupérer présente quelques différences par rapport à la spécification fournie :

- La RAM est modélisée par un banc de registres à triple accès (un accès en écriture, deux accès en lecture). Ceci implique que les lectures et les écritures mémoire se font sur front montant du signal d'horloge ck.
- Les drapeaux np et ng sortent la valeur '1' pour les opérations logiques.

Compte-Rendu

Il vous est demandé un rapport au format .pdf. Vous joindrez tous vos tests en annexe. Votre méthodologie sera évalué en priorité sur les résultats. Par exemple, un groupe ayant identifié le bogue d'un seul des deux AMD2901 mais de manière intelligible recevra bien plus de points qu'un groupe ayant trouvé les erreurs au hasard sans la moindre méthode.