

Architecture de l'Amd2901 Nous décomposons l'Amd2901 en 2 blocs : la partie contrôle qui regroupe la "glu" logique et la partie opérative (chemin de données). Le chemin de données contient les parties régulières de l'Amd2901 c'est à dire les registres et l'unité arithmétique et logique. La partie contrôle contient la logique irrégulière, c'est à dire le décodage des instructions et le calcul des "drapeaux" (indicateurs, ou "ags"). Les fichiers fournis sont les suivants : Répertoire : ~trncomun/TP/2006/TP2/Fichiers/Fournis/amd2901/ amd2901_ctl.vbe, description comportementale de la partie contrôle amd2901_dpt.vbe, description comportementale de la partie chemin de données amd2901_dpt.py, fichier PYTHON de la partie chemin de données, à compléter amd2901_core.vst, description structurelle du coeur amd2901_chip.vst, description structurelle du circuit contenant les plots pattern.pat, fichier de test CATAL, fichier listant les fichiers comportementaux, à modifier Makefile, pour automatiser la génération Réalisation de la partie contrôle 6.1 Vérification des fichiers fournis Récupérez le fichier CATAL dans votre répertoire de simulation. Il doit contenir les lignes suivantes : amd2901_ctl C amd2901_dpt C Cela a pour effet d'indiquer au simulateur qu'il faut prendre les fichiers comportementaux (.vbe) de "amd2901_ctl" et de "amd2901_dpt". Faire :

```
asimut amd2901_chip pattern resultat
```

Vous pouvez contrôler le résultat en utilisant XPAT sur le fichier "resultat". 6.2 Génération de la vue structurelle On souhaite réaliser la vue structurelle de la partie contrôle de l'amd à l'aide de la vue comportementale fournie. Pour ce faire on utilise de nouveau BOOG qui réalise la synthèse logique avec les cellules pre-caractérisées de SXLIB.

```
boog amd2901_ctl
```

Valider ensuite la vue structurelle obtenue en simulant le circuit complet avec les vecteurs de test qui vous sont fournis. On remplacera la vue comportementale de la partie contrôle par sa vue structurelle en ôtant le nom amd2901_ctl du fichier CATAL.

```
asimut -zerodelay amd2901_chip pattern resultat
```

Notez que l'on réalise une simulation "sans délai" de la netlist. En cas de problème(s), n'hésitez pas à utiliser XPAT.