

ENSSAT - EII3

TP de Systèmes sur Silicium (SOC)

Traduction et simulation de la DCT inverse en SystemC

Installation du TP

Modifiez votre fichier `.cshrc` en y écrivant la ligne suivante afin de configurer les outils Synopsys et BSS utilisés dans la suite du TP.

```
source /usr/local/synopsys/environ.csh
```

```
source /users/local/gaut/environ.csh
```

- Copiez le fichier DCT.tar à partir de <http://r2d2.enssat.fr/enseignements/Cao/DCT.tar>
- Décompressez le avec la commande : `tar xfv DCT.tar`
- Vous obtenez un répertoire DCT dans lequel vous vous placerez pour la suite de cette partie du TP.

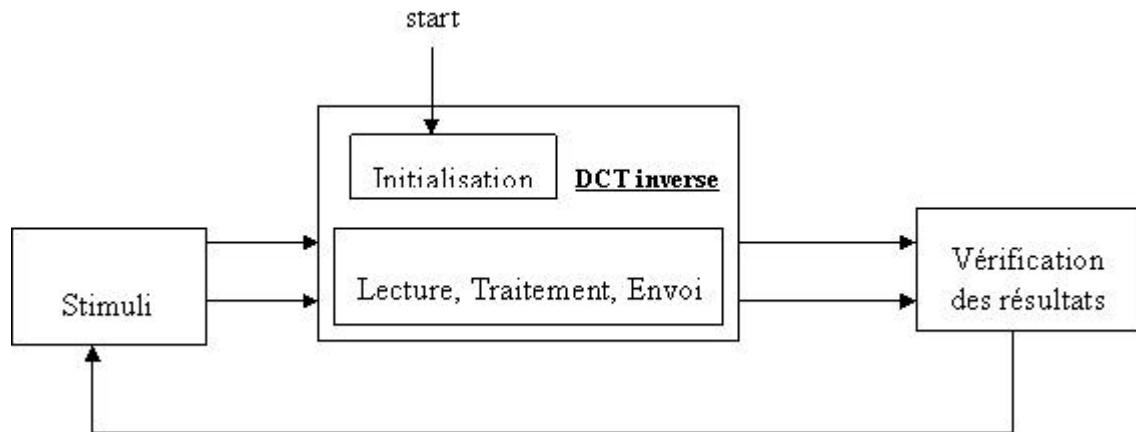
1. Objectifs

La DCT inverse ayant été identifiée par le profiling comme un traitement critique du décodeur MPEG-2, il s'agit dans cette partie de l'extraire du code source en vue de le synthétiser. Pour cela, ce code doit être encapsulé dans un module SystemC et validé avant d'être transmis aux outils de synthèse.

Cette partie se concentre uniquement sur l'aspect encapsulation et simulation SystemC et occulte l'aspect synthèse, ce module étant trop complexe pour être synthétisé dans le cadre d'un TP de 4 heures.

2. Méthodologie

L'encapsulation de l'IDCT nécessite l'introduction de méthodes permettant de lire les Macroblocs, d'exécuter la transformation et de renvoyer les Macrobloc transformés. A ceci s'ajoute également une méthode chargée de réaliser les différentes initialisations utiles au traitement ainsi que deux nouvelles entités permettant d'envoyer des stimuli (données et contrôle) et de vérifier les résultats. Le système peut être modélisé de la manière représentée ci dessous.



Les fichiers nécessaires à cette modélisation sont inclus dans le fichier `DCT.tar`. Extrayez ces fichiers par la commande `tar xvf DCT.tar`. Vous allez générer les fichiers suivants :

- `Block_Entree.txt`, `Block_Sortie.txt` : Ces fichiers contiennent les données de test. Le fichier `Block_Entree.txt` contient les valeurs des Macroblocs et le fichier `Block_Sortie.txt` les données transformées. Ces données ont été extraites par l'intermédiaire du décodeur MPEG décrit en c (en rajoutant des commandes `fprintf(...)`) et correspond aux Macroblocs de la séquence de test `test.m2v`.
- `Stimuli.cc`, `Stimuli.h` : Ces fichiers décrivent le module chargé de lire les données de test dans le fichier `Block_Entree.txt` et de les envoyer au bloc traitant la DCT inverse.
- `Idct_fix.cc`, `Idct_fix.h` : Ces fichiers décrivent le module traitant la DCT inverse.
- `Check_res.cc`, `CheckRes.h` : Ces fichiers décrivent le module chargé de lire le résultat de la transformation des Macroblocs dans le fichier `Block_Sortie.txt` et de les comparer à celles issues du module `idct`.
- `Main_idct_fix.cc` : Ce fichier contient la fonction `main`. Il instancie les différents modules et lance les fonctions de simulation.
- `Makefile`, `Makefile.defs` : Le fichier `Makefile.defs` est appelé par le `makefile` et permet d'intégrer les bibliothèques systemC lors de la compilation.

3. Travail à réaliser

1. Etudiez les fichiers évoqués précédemment et en particulier, décrivez le protocole de communication mis en place entre les différents modules. Le fichier `Idct_fix.h`, décrivant l'interface du module `Idct` est incomplet. Vous devez le modifier de manière à spécifier ces entrées/ sorties et les méthodes utilisées. Certaines d'entre elles devront être implémentées sous la forme de `SC_CTHREAD` et `SC_METHOD`. Justifiez les choix mis en œuvre, en tenant compte de la description des méthodes décrites dans le fichier `Idct_fix.cc`.
2. Compilez la description (`make`) et lancez la simulation par la commande `IDCT.exe`. L'exécution renvoie ces résultats sous deux formes.
 - Une sortie textuelle s'affichant dans le terminal.
 - Une sortie graphique sous forme de *waveformes* (`IDCT.awif`) pouvant être visualisée via le waveform viewer de Synopsys.
Pour exploiter cette sortie graphique vous devez tout d'abord la rendre compatible avec l'outil Synopsys par la commande `a2wif IDCT.awif` qui génère le fichier `IDCT.awif.b`. Renommer ce fichier en `IDCT.ow` (`mv IDCT.awif.b IDCT.ow`). Lancer alors l'outil par la commande `waves` et ouvrez le fichier `IDCT.ow`, ou lancer la commande `waves -wif IDCT.ow &`
3. Vérifiez la fonctionnalité de votre design par l'intermédiaire de ces deux méthodes.