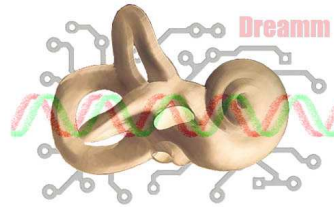


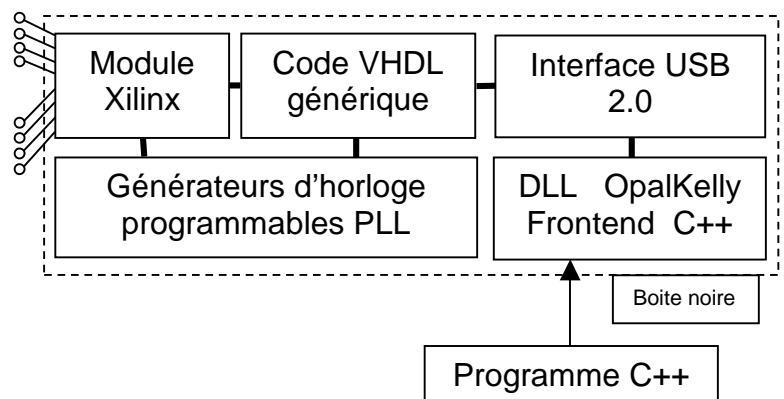
Projet DREAMM



Le projet DREAMM (aide au Diagnostic Rapide Et Automatisé de la Maladie de Ménière) consiste à mettre en place le premier appareil permettant de mesurer les variations de pression à l'intérieur de l'oreille interne. Cette technologie serait à même de diagnostiquer la maladie de Ménière.

Le contexte :

Plusieurs périphériques électroniques numériques (DAC,ADC,PGA,Memory) sont commandés à partir d'un logiciel d'analyse médicale. Actuellement, ce système étant en cours de développement, il n'est pas possible de savoir quelles vont être les différents composants retenus pour la production. Plusieurs choix sont disponibles et peuvent être utilisés indifféremment. Pour cela, nous utilisons un module xilinx qui reçoit un flux continu d'informations depuis le programme :



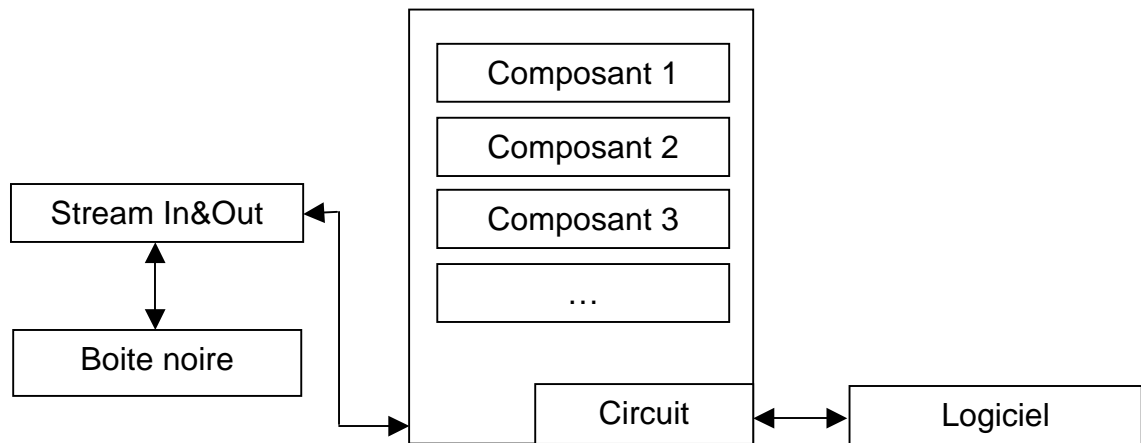
Chacune des pins du circuit est identifiée par un numéro. De plus nous fixons dès le commencement l'horloge à laquelle l'information doit être débitée sur chaque pin. Ainsi, l'information circule sous forme de paquets de 8ko par le port USB. Chaque paquet contient un ensemble de blocs dont le format est le suivant :

| | |
|-----------------|---------------------------------|
| 7 bits : ID pin | 121 bits : données de cette pin |
|-----------------|---------------------------------|

Le but du projet est de supprimer la création de code VHDL pour chaque circuit utilisé. Cette modification devant être de toute façon refaite aussi dans le logiciel, nous préférons utiliser un code VHDL générique qui gère deux streams (in & out) puis un logiciel qui manipule tous les composants.

Réalisation :

Il s'agit purement de conception logicielle objet. Actuellement considérons que nous disposons de quatre fonctions : deux qui nous permettent d'écrire ou lire un paquet et deux autres avec lesquelles on sait si on peut écrire ou lire un paquet. Il s'agit en fait de notre boite noire. Chaque objet-composant regroupe un ensemble de propriétés communes. Par exemple, l'ensemble des signaux qui servent à son initialisation, la tâche qu'il doit effectuer durant le processus de traitement, la gestion de la fin du processus et ses fonctions particulières. Les fonctions communes seront regroupées dans une super-classe. Le logiciel lui va dialoguer avec un circuit (virtuel). Un circuit est décrit par l'ensemble des composants qu'il contient et par les connexions entre les pins de sortie du montage et certains composants. Le logiciel doit pouvoir demander le lancement d'un scénario. Dans une première phase, l'ensemble des chronogrammes de sortie sont pré-calculés. Dans une deuxième phase, un processus va gérer les deux streams in&out avec les contraintes propres. Puis, dans la troisième phase, les informations retournées seront assemblées pour reconstituer les signaux acquis.



Le développement sera effectué sous C++ Builder 5. Une présentation de l'environnement sera effectuée en début de projet. Une interface simple devra être implantée pour spécifier facilement les chronogrammes d'initialisation de certains circuits.

Renseignement : buzerl@esiee.fr