

# Introduction aux Transmissions Numériques à l'aide de blocs fonctionnels Temps Réel sur processeur DSP

Jean-Marie ORY

ESSTIN Université Henri Poincaré, 2 rue Jean Lamour F-54519 Vandœuvre les Nancy  
jean-marie.ory@esstin.uhp-nancy.fr

**RESUME** Nous avons présenté lors du CETSI'S03 une plate-forme Processeur de Signal (DSP) "MuPsi - FIBULA". Elle permet de créer très simplement des applications Temps Réel, en interconnectant des blocs fonctionnels. Dans ce document, nous montrons l'utilisation de cette plate-forme dans notre enseignement en Télécommunications. Nous y décrivons plusieurs manipulations simples sur les principes de la transmission en bande de base, des modulations numériques, du multiplexage et du codage de canal. Nous rassemblons plusieurs de ces notions dans une maquette que nous présentons lors de ce CETSI'S, un modem acoustique avec code correcteur d'erreur.

**Mots clés :** DSP, blocs fonctionnels, télécommunications, modulation, multiplexage, codage.

## 1. INTRODUCTION

Lors du dernier CETSI'S nous avons présenté une plate-forme Processeur de Signal (DSP) pour l'enseignement et l'industrie [1]. Elle se compose d'une carte DSP "MuPsi" Fig. 1.1 et d'un environnement de développement intégré basé sur un compilateur de blocs fonctionnels "FIBULA" Fig. 1.2.



Fig. 1-1 Carte DSP MuPsi

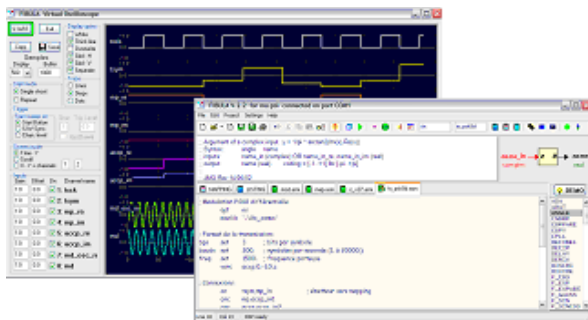


Fig. 1-2 Environnement de développement FIBULA

Nous présentons ici un module d'enseignement "Introduction aux Transmissions Numériques" pour lequel existe un grand nombre de démonstrations Temps Réel sur notre plate-forme DSP.

L'ensemble constitué de la plate-forme DSP MuPsi - FIBULA associé à la bibliothèque Télécom et du manuel "Introduction aux Transmissions Numériques" est maintenant distribué par les Stés PIERRON Technique ([www.pierron.fr](http://www.pierron.fr)), et ADAPTECH ([www.adaptech.com](http://www.adaptech.com)).

## 1.1 Historique

Nous avons développé la carte Mu.Psi en 98-99 avec l'aide de la Sté Arnatronic ([www.arnatronic.com](http://www.arnatronic.com)) pour répondre à un besoin de carte DSP industrielle performante pour le contrôle de processus rapides [1]. En enseignement, cette carte nous a d'abord servi comme support du cours sur les Processeurs de Signaux. Puis, elle nous a servi à monter quelques TP du cours de Signaux (en particulier, filtres numériques et modulations). Afin de ne pas obliger les étudiants à programmer en assembleur, nous avons écrit un ensemble de blocs fonctionnels prêts à l'emploi, la programmation ne se ramenant qu'à définir les connexions entre ces blocs, d'où le concept FIBULA.

Mais nous avons surtout perçu l'intérêt de ce support en enseignement des télécommunications. La rapidité du processeur permet en effet de réaliser en temps réel la plupart des fonctions de traitement de l'information. Cette rapidité permet également de traiter rapidement un très grand nombre de données conduisant à une bonne statistique qui permet de vérifier la concordance des lois de la Théorie de l'Information et du Codage.

## 2. COMMUNICATIONS EN BANDE DE BASE

### 2.1 Transmission idéale, régénération de l'horloge

Toutes les manipulations que nous présentons consistent à transmettre un texte ASCII d'un émetteur vers un récepteur, et d'afficher le texte reçu sur l'écran d'un terminal. Pour cela, on réalise la concaténation des codes ASCII du texte, et le résultat est découpé en une suite de symboles binaires de *bps* bits émis au rythme d'une horloge qui définit la vitesse de transmission en Bauds. Pour la transmission des signaux en bande de base, nous adoptons le codage multi-niveaux qui, dans sa forme la plus simple, consiste à envoyer les symboles binaires vers un convertisseur NA (Fig. 2-1). Les niveaux transmis et les seuils de décision en réception pour sont donnés dans le tableau de la fig. 2.2.

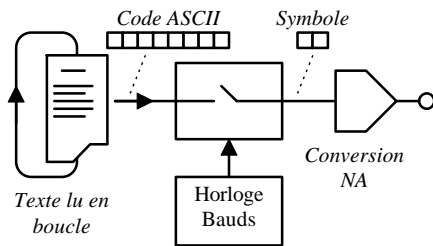


Fig. 2-1 Emetteur en bande de base

Bits par symbole	Symboles binaires à transmettre	Signaux émis (unités machine)	Seuils en réception (unités machine)
1	0 1	-0.5 0.5	0
2	10 11 00 01	-0.75 -0.25 .25 0.75	-0.5 0 0.5
3	100 101 . . .	-0.875 -0.625 . . .	-0.75 -0.5 . . .

Fig. 2-2 Codage multiniveaux des symboles

En réception, le décodage des symboles consiste simplement en une conversion AN dont on tronque le résultat à *bps* bits. On doit régénérer l'horloge et procéder à un cadrage lors de la concaténation des symboles afin de reconstruire correctement les codes ASCII Fig. 2-3.

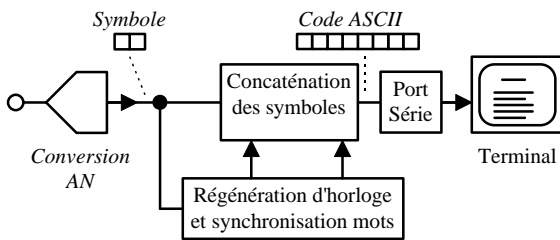


Fig. 2-3 Récepteur en bande de base

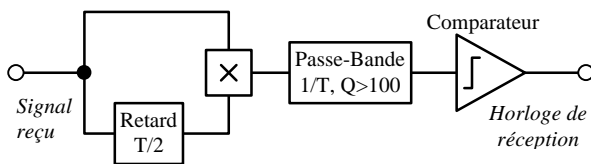


Fig. 2-4 Circuit de régénération de l'horloge

Le signal multiniveaux transmis ne contient aucune raie à la fréquence du rythme (Bauds). Pour la régénération d'horloge, nous devons donc reconstituer une composante spectrale à cette fréquence, et qui possède la même phase que l'horloge d'origine. La figure 2-4 montre le circuit de régénération d'horloge. Le filtre passe-bande étroit du 2<sup>nd</sup> ordre extrait cette composante et introduit un déphasage nul à la résonance.

L'opération de cadrage se fait en injectant à l'émission des codes ASCII "NUL" qui sont reconnus à la

réception et provoquent la remise à 0 du compteur de bits.

### 2.2 Canal à bande limitée

Bien entendu, la transmission réalisée comme ci-dessus n'est limitée que par le débit du port série connecté au terminal (38400 Bauds). Pour simuler un canal à bande limitée, nous introduisons un filtre RIF passe-bas entre émetteur et récepteur. Le canal est maintenant virtuel, c'est à dire logiciel. En synchronisant l'oscilloscope sur un front d'horloge et en superposant les traces (signal reçu), on obtient le diagramme de l'œil qui montre les interférences inter symboles (ISI). On vérifie que la limite supérieure du rythme de transmission est de 2 fois la bande passante. On minimise les ISI en émettant des impulsions en forme de Sinus Cardinal fenêtré (impulsions et filtre "Root Raised Cosine").

### 2.3 Suppression d'échos

Lors d'une transmission bidirectionnelle, il arrive qu'une diaphonie entre le canal montant et le canal descendant produise une pollution du signal reçu par une somme pondérée de versions retardées du signal émis. On élimine ce phénomène à l'aide d'un filtre RIF autoadaptatif dont le rôle est de synthétiser un écho identique à celui que l'on observe, puis de le retrancher. La simulation de ceci est représentée figure 2-5.

Le RIF utilise l'algorithme LMS, la constante de temps de convergence est réglée à environ 10 secondes, de manière à ce que l'on puisse bien observer le phénomène. On peut alors visualiser la réponse impulsionnelle du RIF et déterminer les retards des échos.

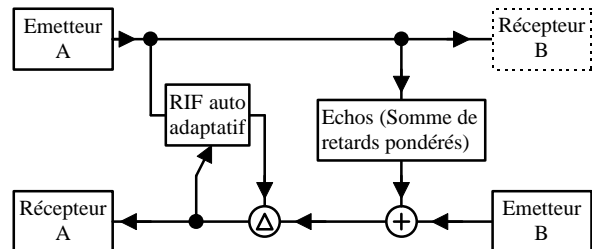


Fig. 2-5 Suppression des échos de l'émission A reçus en A

## 3. MODULATIONS

### 3.1 Modulateur généralisé

Lorsque le canal est de nature passe-bande, on procède à une modulation du signal de manière à déplacer son spectre vers la bande passante du canal. Une modulation numérique se met généralement sous la forme:

$$m(t) = \text{Re} \left[ \sum_k c_k(t - kT) e^{(j\omega_0 t + \phi)} \right]$$

où  $c_k(t)$  représente l'impulsion complexe associée au symbole d'ordre  $k$ ,  $T$ , la période de l'horloge de

transmission, et  $\omega_0$ , la pulsation de la fréquence porteuse. Dans le cas général où les  $c_k$  sont de simples constantes complexes, le modulateur se ramène au schéma représenté figure 3-1, et les différents types de modulation sont déterminés par l'emplacement (mapping) des points  $c_k$  dans le plan complexe. L'ensemble de points constitue une constellation (figure 3-2).

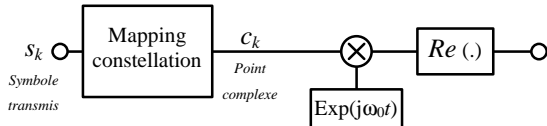


Fig. 3-1 Principe général d'une modulation numérique

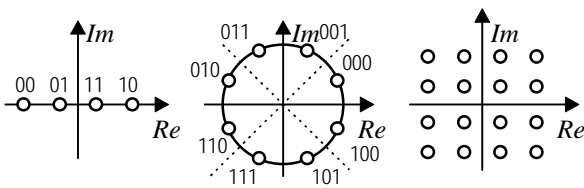


Fig. 3-2 Constellations des modulations ASK-4, PSK-8, QAM-16

### 3.2 Modulations ASK, PSK, QAM

Les modulations d'amplitude (ASK), de phase (PSK), et de phase et d'amplitude (QAM) s'obtiennent très simplement avec le schéma de la figure 3-1. Lors de la démodulation, en revanche, on doit lever l'ambiguïté de la phase. Pour cela, on code en différentiel, c'est à dire que les phases des points de la constellation représentent l'écart entre la phase du symbole actuel et la phase du symbole précédent. Pour cela, à l'émission, on accumule les phases, tandis qu'à la réception, on différencie les phases. Ces notions sont décrites dans la démonstration "Modem Acoustique" présentée lors de ce CETIS.

## 4. MULTIPLEXAGE

Le multiplexage consiste à transmettre plusieurs messages simultanément sur un même canal. Pour cela, il suffit à l'émission de rendre les signaux orthogonaux entre eux et de les superposer. A la réception, on effectue la sélection du message désiré par projection du signal composite sur un axe déterminé.

Un procédé trivial de multiplexage consiste à prendre comme base de signaux orthogonaux des rectangles décalés, soit dans le domaine temporel, soit dans le domaine fréquentiel.

### 4.1 Multiplexage temporel et fréquentiel

La figure 4-1 représente le schéma d'une manipulation de transmission multiplexée au travers d'un câble coaxial. Dans le domaine temporel, on crée des trames subdivisées en tranches. Les trames sont émises au

rythme de la transmission, chaque tranche correspond à une voie. Une tranche particulière sert à la synchronisation de trame. Dans le domaine fréquentiel, les différentes voies sont associées à des bandes distinctes. Le multiplexage consiste à moduler en amplitude sur différentes porteuses, et le démultiplexage revient à filtrer la bande désirée, puis à démoduler le signal obtenu.

Dans la manipulation présentée ci-dessous, nous transmettons des messages en 4 langues sur un câble coaxial à l'aide d'un multiplexage soit temporel, soit fréquentiel.

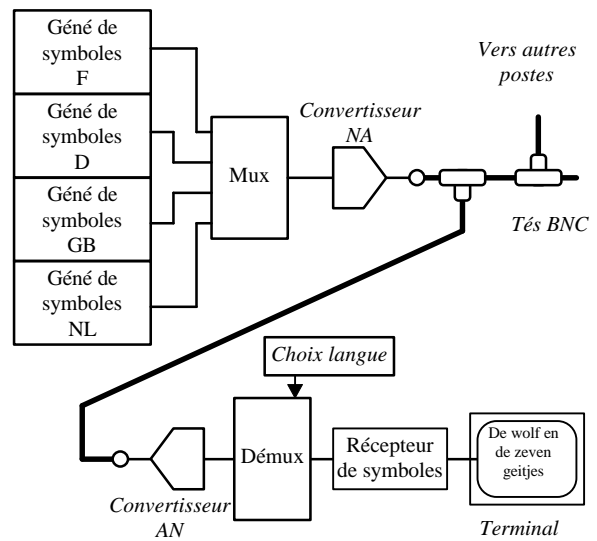


Figure 4-1 Transmission simultanée de 4 messages en 4 langues entre postes distants reliés par un câble coaxial.

### 4.2 Multiplexage par codes orthogonaux

On peut également rendre les signaux orthogonaux en les multipliant par des séquences pseudo aléatoires. La figure 4-2 représente le principe habituel, dit "implémentation de Fibonacci", qui utilise un registre à décalage et des portes OU exclusif, et où les  $g_i$  représentent le code de rétroaction qui produit la séquence de longueur maximale.

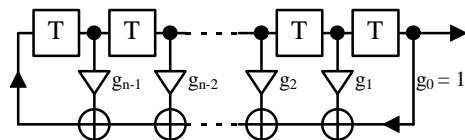


Figure 4-2 Générateur Pseudo Aléatoire

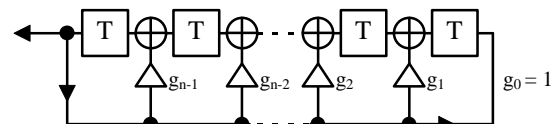


Figure 4-3 Générateur PA, version transposée du précédent

Si l'on applique le théorème de la Transposition à ce schéma (inversion des sens, permutation entre noeuds et additionneurs), on obtient le schéma équivalent de la

figure 4-3 appelé "implémentation de Galois", laquelle est particulièrement facile à programmer sur un DSP, puisque son code assembleur ne se ramène qu'à 2 instructions:

```
lsl a y:g,x0 ; décalage à G. et chargement g_i
eor x0,a ifcs ; OU exclusif si Carry = 1
```

Un registre de N bits produit une suite de  $2^N-1$  états, laquelle possède une fonction d'autocovariance proche d'une impulsion de Dirac (fig. 4-4), ce qui permet de considérer 2 suites identiques décalées comme orthogonales.

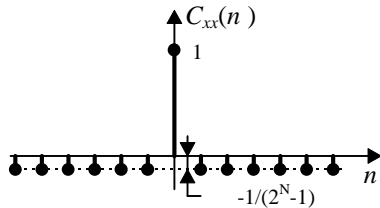


Figure 4-4 Fonction d'autocovariance des suites pseudo-aléatoires

Pour montrer le principe du multiplexage CDMA, nous multiplions 4 formes d'onde (sinus, rectangle, triangle, et dent de scie) par des codes PA identiques mais décalés, et sommes les signaux. La démodulation consiste à multiplier le signal obtenu par la séquence appropriée et à intégrer le résultat sur 1 période du code PA.

### 5. THEORIE DE L'INFORMATION ET CODAGE

L'apport du processeur DSP en Théorie de l'Information est de permettre de réaliser des expérimentations statistiques de bonne qualité. En effet, la vitesse de la machine permet d'atteindre rapidement de très grands nombres d'expériences, conduisant à de bonnes estimations statistiques.

#### 5.1 Estimation en TR de l'entropie d'une source et de l'information mutuelle transmise sur un canal

Pour obtenir expérimentalement l'entropie d'une source

$$H(X) = - \sum_{i=1..n} p_i \log_2(p_i)$$

nous réalisons un histogramme sur un grand nombre de symboles (ordre de  $10^6$ ) et nous ramenons la somme des contenus des classes  $\Sigma c_i$  à 1.0, de sorte que les valeurs correspondent aux densités de probabilités  $p_i$ . Il suffit alors de sommer les valeurs pondérées par la fonction  $-x \log_2(x)$  pour obtenir  $H$ . Fig. 5-1.

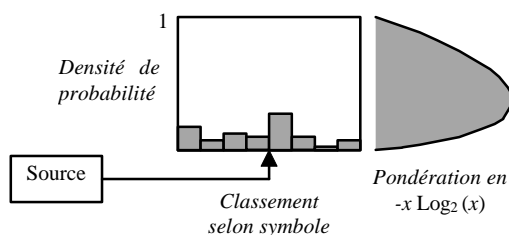


Figure 5-1 Estimation de l'entropie d'une source

Une autre notion qui semble bien abstraite à nos étudiants est celle d'Information Mutuelle qui est fondamentale, puisque son maximum représente la capacité d'un canal.

Soient X et Y les variables d'entrée et de sortie d'un canal, en reprenant la définition de l'information mutuelle moyenne pour un canal sans mémoire,

$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$$

$$= \sum_{x \in X} \sum_{y \in Y} p(x,y) \log \frac{p(x,y)}{p(x)} = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

nous voyons qu'elle peut s'exprimer en fonction de la densité de probabilité conjointe  $p(x,y)$  et des densités marginales  $p(x)$  et  $p(y)$ . En obtenant un histogramme à 2 dimensions (entrée X, sortie Y), nous calculons en temps réel la valeur de l'information mutuelle moyenne exprimée en bits par utilisation du canal. Fig 5-2.

Expérimentalement, on vérifie parfaitement que pour un canal complètement bruité (X et Y indépendants) la valeur de  $I(X;Y)$  obtenue est nulle, ce qui est normal, puisque dans ce cas,  $p(x,y)=p(x)p(y)$ .

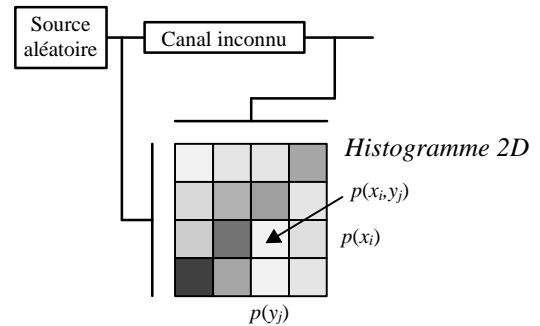


Figure 5-2 Estimation de l'information mutuelle moyenne transmise à chaque utilisation du canal

#### 5.2 Codes par blocs et codes cycliques

Nous avons inclus récemment les opérations matricielles dans la bibliothèque de blocs fonctionnels FIBULA. Cela permet, entre autres, de réaliser très facilement des codeurs par blocs avec décodage et correction d'erreurs par le syndrome. Pour les codes cycliques, nous préférons toutefois utiliser des techniques à base de registre à décalage du genre de la Fig. 4-2; il en est de même pour le calcul de CRC.

#### 5.3 Mesure des performances d'un codeur de canal

Bien que l'on sache parfaitement calculer les performances d'un codeur en matière de correction d'erreurs, il est toujours agréable de vérifier la théorie par l'expérience. Pour cela, il nous faut injecter dans le canal un taux d'erreur par bit TEB parfaitement connu. La figure 5-3 montre comment nous nous y prenons: à chaque bit transmis, on réalise un tirage d'une variable aléatoire X de densité parfaitement uniforme entre 0 et 1. La probabilité que celle-ci soit inférieure à la valeur TEB est donc égale à TEB. Donc, si  $X < TEB$ , nous complétons le bit transmis.

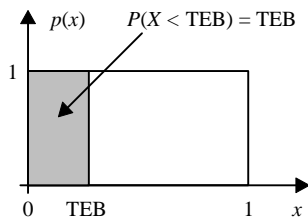


Figure 5-3 Simulation d'un taux d'erreur TEB défini à l'aide d'un générateur aléatoire uniforme

La figure 5.4 montre un banc de test pour tous types de codeurs correcteurs d'erreurs. La source est par exemple un texte ASCII assez volumineux tel que la Bible ou le "Guide du Rootard". Les blocs  $p \rightarrow q$  sont des fonctions de découpage, concaténation et FIFO qui transforment un flux de symboles à  $p$  bits en un flux de symboles à  $q$  bits. Ces blocs communiquent entre eux

en se passant des variables booléennes (jetons) symbolisés par les flèches. Les codes ASCII sont d'abord découpés en symboles  $k$  bits, lesquels sont codés en symboles  $n$  bits. On repasse à 1 bit afin de pouvoir procéder à l'injection d'erreur. Une horloge limite le débit du flux 1 bit afin qu'en sortie, l'affichage du texte sur écran terminal soit facilement lisible. Après injection d'erreur, on reconstitue les symboles codés à  $n$  bits, on décode, puis on concatène les symboles  $k$  bits pour reconstruire des codes ASCII. Une deuxième branche de la chaîne de traitement ne comporte pas l'injection d'erreur et sert de référence de comparaison. On compte le nombre total de bits transmis, le nombre d'erreurs injectées. Le décodeur, les erreurs détectées et les erreurs qu'il corrige. La comparaison entre les 2 branches fournit un comptage objectif des erreurs qui subsistent après décodage et correction. On visualise ces erreurs sur l'écran en colorant en rouge les codes ASCII erronés.

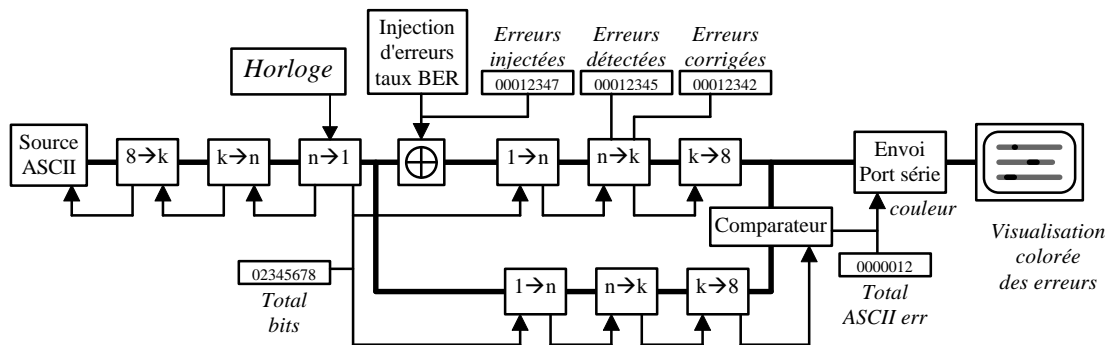


Figure 5-4 Plate-forme pour le test des performances des codeurs  
NB: Les fonctions Codage et Décodage sont représentées par les blocs  $k \rightarrow n$  et  $n \rightarrow k$

## 6. CONCLUSIONS

La plate-forme DSP MuPsi-FIBULA propose environ une centaine de blocs fonctionnels orientés télécommunications, et une cinquantaine de démonstrations prêtes à l'emploi. Celles-ci constituent un support expérimental permettant de vérifier et confirmer lors de TD l'étude théorique vue en cours. Pour beaucoup d'étudiants, le fait de manipuler des signaux, de les visualiser sur un écran d'oscilloscope, ou de voir le message transmis s'afficher sur un écran terminal constitue une véritable révélation. Il convient néanmoins prendre garde à ce que les séances de TD ne se ramènent pas à un simple jeu consistant à tester les démonstrations sans comprendre, puis à éditer un magnifique compte rendu rempli de diagrammes et d'oscillogrammes obtenus par copier-coller !

Cette plate-forme peut également servir à la réalisation de projets très intéressants comme, par exemple, le modem acoustique présenté comme démonstration de ce CETIS.

Elle peut servir en recherche, à mettre très rapidement en œuvre une maquette qui permettra de tester tel type de modulation ou tel algorithme de codage en temps réel.

Depuis peu, nous avons mis au point une version graphique du compilateur FIBULA. Celle-ci permet de visualiser les blocs et leurs connexions, d'ouvrir un bloc particulier, voir son schéma interne, etc. ..., ceci en descendant la hiérarchie jusqu'au niveau atomique qui est constitué de code assembleur. Elle permet aussi de créer facilement de nouveaux blocs, soit en montant, soit en descendant la hiérarchie. Cette version est actuellement au stade de l'évaluation et nous espérons bientôt la mettre sur le marché.

Nous pensons ainsi proposer un outil pédagogique très simple à mettre en œuvre, et bon marché, face à des produits industriels beaucoup plus lourds, souvent inabordable.

## Bibliographie

1. J.M. ORY *Mu.Psi - FIBULA, un kit DSP simple et efficace pour l'enseignement et l'industrie.*  
CETSIIS 03 Toulouse
2. A.B. CARLSON *Communication systems*  
Mc Graw-Hill ISBN 0-07-100560-9
3. A. SPATARU *Fondements de la théorie de la transmission de l'information*  
Presses Polytechniques Romandes ISBN 2-88074-133-0
4. P. LECOY *Technologie des Télécoms* Hermes  
ISBN 2-7462-0002-3
5. C. SHANNON *A mathematical theory of communication*  
The Bell System Technical Journal 1948
6. G. BATAIL *Théorie de l'Information, Application aux techniques de communication.* Masson 1997