

IUT GEII NICE

# TP ELEN4

TRANSMISSION NUMERIQUE

BUT2 GEII - Semestre 4  
2022-2023



***TP1 : TRANSMISSION EN BANDE DE BASE :  
CODAGE DE SIGNAUX***



## 1. Introduction

La transmission en bande de base consiste à transmettre un signal dans sa bande originelle sans transposition de fréquence (sans porteuse) par l'intermédiaire d'un support physique (par essence sur support en cuivre) et sur de courtes distances (car atténuation sur le support proportionnel à sa longueur) ou donc être régénérés périodiquement sur une longue distance.

Si le signal d'origine est un signal analogique (infinité de valeurs), celui-ci doit être numérisé en passant par différentes étapes :

- **Echantillonnage** : qui consiste à transformer le signal analogique (infinité de valeurs) en un signal discret (N valeurs) conditionné par la période d'échantillonnage  $T_e$ . Si  $T_e$  très petit, N sera très grand (donc trop de valeurs à stocker et à traiter) ; si  $T_e$  grand, N petit et le signal discret ne sera pas fidèle au signal analogique d'où des pertes d'information. C'est pourquoi il existe un critère optimal pour choisir  $T_e$  qui est le critère de Shannon.
- **Quantification** : Consiste à faire correspondre à chaque signal discret une suite binaire. Ceci vient du fait qu'en informatique, les supports d'information ne sont pas analogiques. Le choix du nombre de bits  $n$  sur lequel chaque signal discret est codé détermine la perte ou non d'informations et qui peut être estimée par l'erreur de quantification.

Après ces différentes étapes (transformation d'un signal analogique en une suite binaire), le signal binaire n'est généralement pas transmis directement sur la ligne et différents **codages** numériques sont utilisés pour différentes raisons :

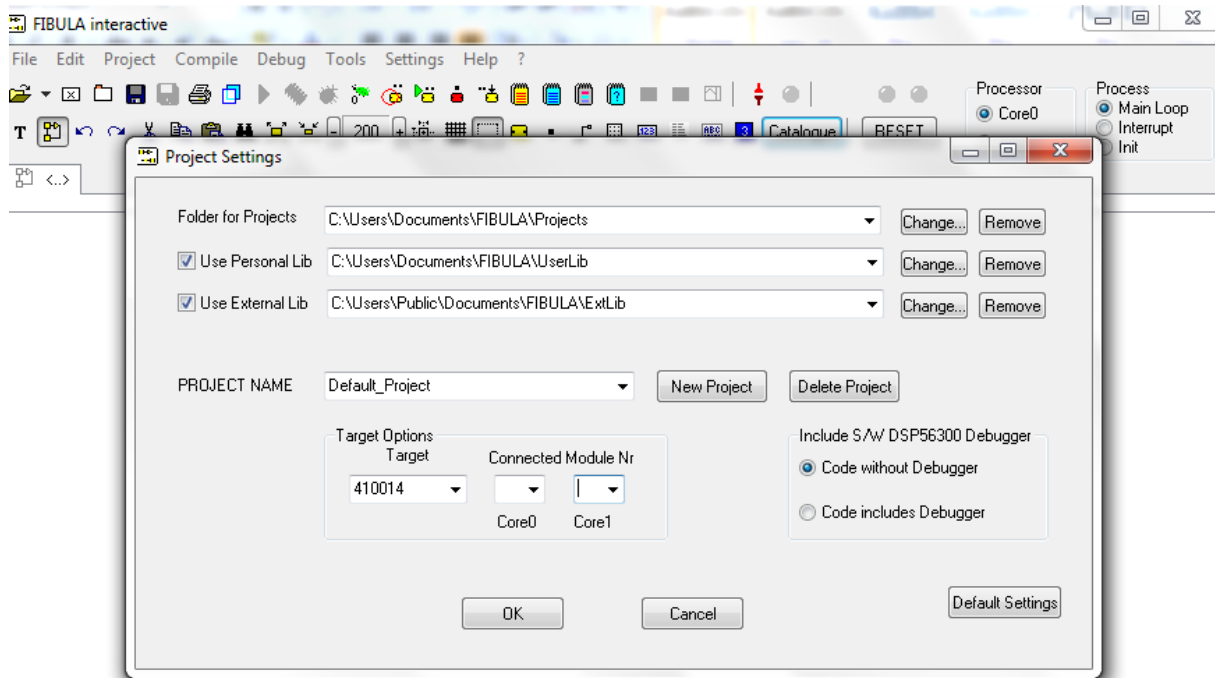
- La récupération de l'horloge nécessaire en transmission synchrone est facilitée par des séquences qui présentent des changements d'états fréquents et évitent ainsi les longues suites de 1 ou de 0 et ne possède pas une raie non nulle à fréquence de l'horloge.
- Les perturbations subies par un signal sont proportionnelles à la largeur de sa bande de fréquence.
- La présence d'une valeur moyenne importante peut entraîner une mauvaise propagation dans les lignes.

*L'objectif de ce TP est de comparer (avantages et inconvénients) différents types de codage numérique. Les différents types de codage qui seront vus dans ce TP sont les codages NRZ, RZ et Manchester.*

## 2. Manipulation

**NB :** la prise en main de Fibula se trouve en annexe, donc vous pouvez vous référer à cette partie parfois pour mieux comprendre l'utilisation du logiciel



- Créer un dossier TPELEN4 puis ouvrir fibula et cliquer sur le menu « **Project** », et faire le réglage suivant :



- Choisir un répertoire personnel (TPELEN4) pour « **Folder to Projects** » et « **Use Personnel Lib** » ; et laisser le répertoire par défaut pour « **Use external Lib** ».
- Choisir comme nom de projet **TPELEN4\_prj**
- Aller dans document partagé dans le réseau/ELEN4/Fibula, puis copier les 2 dossiers (Blocklibs et Programs) et les mettre dans le dossier TPELEN4\_prj que vous venez de créer (les 2 dossiers portant les même noms seront écrasés).

Pour ce 1<sup>er</sup> TP, nous allons utiliser le fichier « **Codage\_bande\_base** » se trouvant dans programme.

Dans ce fichier vous voyez apparaître plusieurs blocs (Message, NRZ, RZ, Manchester) représentant les schémas en annexe. Le débit binaire  $D=R=f_H$  définit en externe est  $D = 4000$  bauds.

- 1- Exécutez le programme avec l'icône , puis lancez l'oscilloscope virtuel avec l'icône , et cliquez sur START.
- 2- Pour chaque type de codage visualiser à chaque fois l'horloge, le signal bande de base et le signal codé en unipolaire et en bipolaire en plaçant des sondes. Expliquer pour chaque type de codage comment sont codés les bits.
- 3- Pour les codages en bipolaire, relever les spectres en amplitude et en dB et faites un tableau comparatif (présence raie à  $f = 0$ , à  $f = f_H$ , BP).
- 4- Donner les avantages et les inconvénients de chaque type de codage.

Le codage de Manchester bipolaire consiste à coder les états de base par des transitions et non par des niveaux permettant de décaler le spectre du signal vers les fréquences plus élevées.

Mais le problème avec ce type de codage est que si on inverse les fils à l'arrivée, le signal décodé n'est plus du tout le même qu'initialement. Pour remédier à ce problème, on utilise le codage Manchester différentiel qui peut correspondre aux règles suivantes :

- Écart entre donnée  $i$  et  $i-1$  égal à 0 -> front montant
  - Écart entre donnée  $i$  et  $i-1$  égal à 1 -> front descendant
- 5- Modifier la liaison entre bande\_base et l'entrée du codage Manchester pour obtenir en sortie du bipolaire un codage Manchester différentiel. La donnée  $i-1$  peut être obtenue en introduisant un retard de  $T_H$  sur le signal bande\_base avec le bloc **DELAY**.

**Attention :** Ce bloc ne travaille qu'avec des données de type *FRACT* de structure *WORD*. Donc le passage de *TF* à 0,1 se fait avec « *booltof* » et le passage de 0,1 à *TF* se fait avec « *frtobool* ».

## Annexe

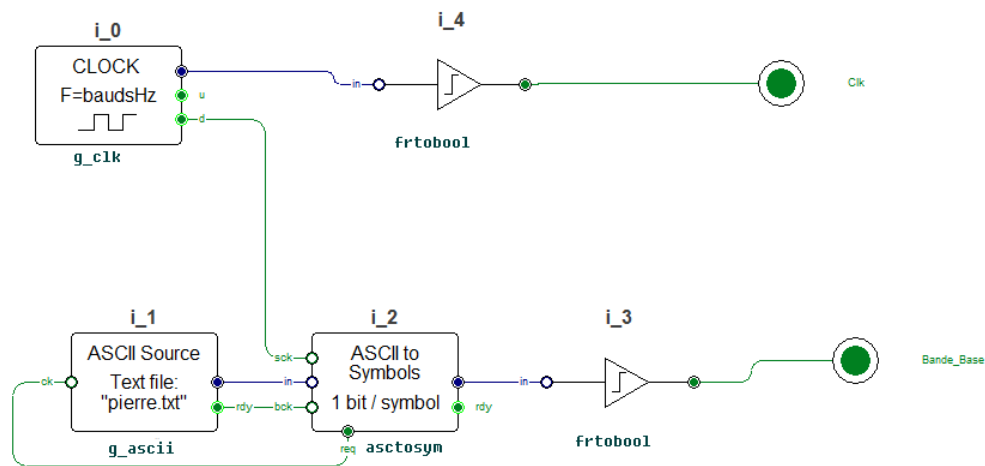


Figure 1: Bloc bande de base avec 2 sorties : clk et bande\_base (type : bool, struct : bit)

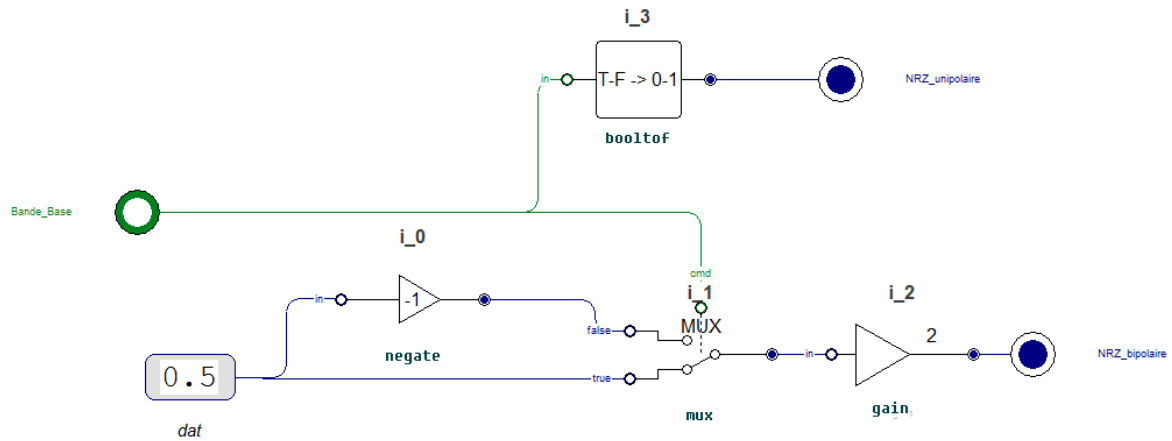


Figure 2 : Bloc NRZ avec 1 entrée bande\_base et 2 sorties NRZ\_uni et NRZ\_bip (type : FRACT, struct : WORD)

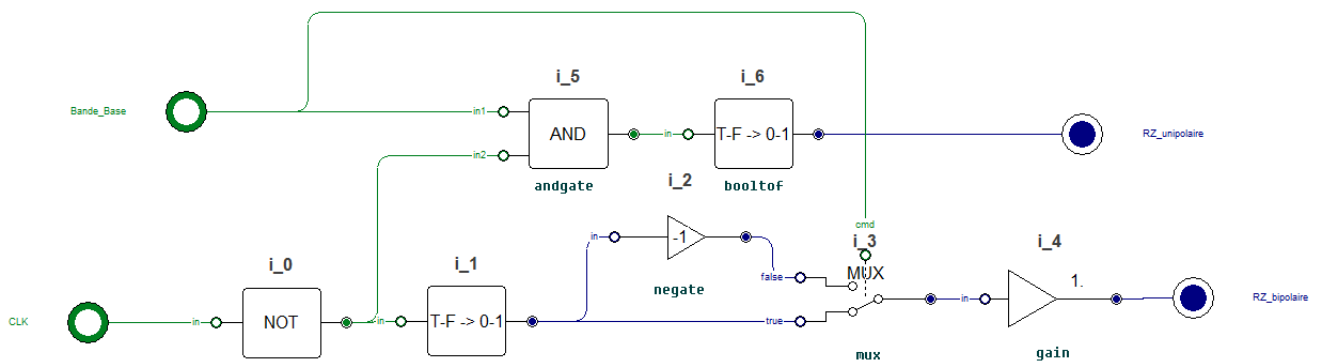


Figure 3 : Bloc RZ avec 2 entrées : bande\_base et CLK et 2 sorties : RZ\_uni et RZ\_bip



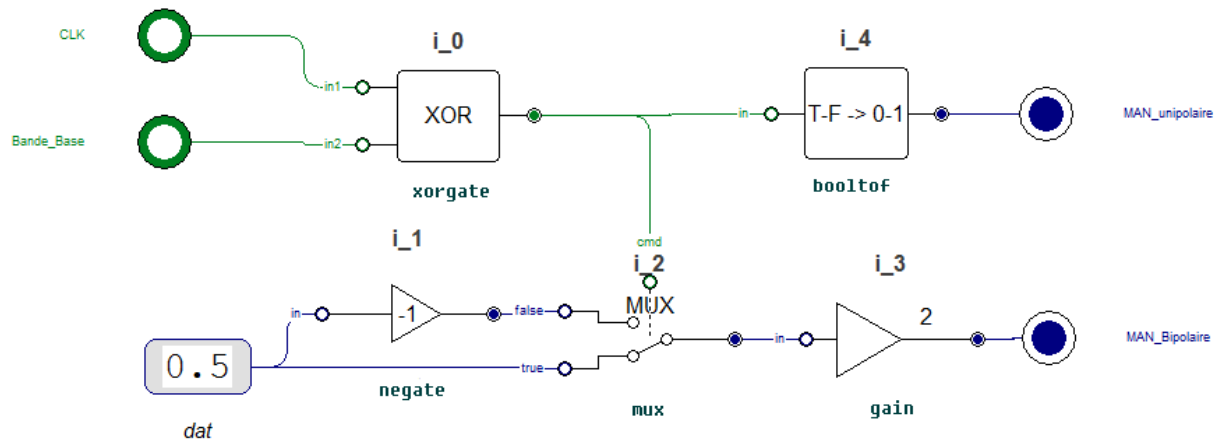


Figure 4 : Bloc MANCHESTER avec 2 entrées : bande\_base et CLK et 2 sorties : Man\_uni et Man\_bip

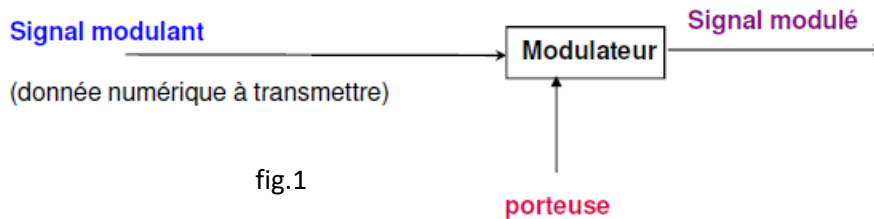


## ***TP2 : MODULATION/DEMODULATION NUMERIQUE ASK-FSK***



## INTRODUCTION

Le schéma fonctionnel est donné par la figure 1 ci-dessous.



- Signal modulant : représente l'information binaire
- Porteuse : signal de haute fréquence le **plus souvent sinusoïdal**
- Signal modulé : signal dont un des paramètres suivants varie en fonction de l'amplitude du signal modulant selon le **type de modulation**.

## Types de modulation

- Modifier l'amplitude de la porteuse :
  - Analogique : MA ou MAPS (Modulation d'Amplitude à Porteuse Supprimée)
  - Numérique : MDA (Modulation à Déplacement d'Amplitude)  
**ASK** (Amplitude Shift Keying)
- Modifier la fréquence de la porteuse :
  - Analogique : MF (Modulation de Fréquence)
  - Numérique : **FSK** (Frequency Shift Keying)
- Modifier la phase de la porteuse :
  - Analogique : MP (Modulation de Phase)
  - Numérique : **PSK** (Phase Shift Keying)

Les modulations étudiées dans ce TP sont les suivantes : ASK et FSK.

# 1. MODULATION NUMERIQUE ASK

Le signal modulant sera un signal binaire NRZ unipolaire et les modulations étudiées seront à une dimension 1 bit/symbole.

## 1.1 MODULATION

Le chronogramme est donné figure 2 et le schéma fonctionnel figure 3

Le modulant est numérique, de la forme :

$$m(t) = \sum a_k g(t - kT)$$

L'amplitude de la porteuse variera en TOUT OU RIEN (OOK).

Le signal modulé est donné par l'expression :

$$s_{mod}(t) = \sum a_k g(t - kT) \cos(\omega_0 t + \phi_0)$$

Où  $a_k$  est la valeur **0** ou **1** des bits du signal binaire à transmettre.

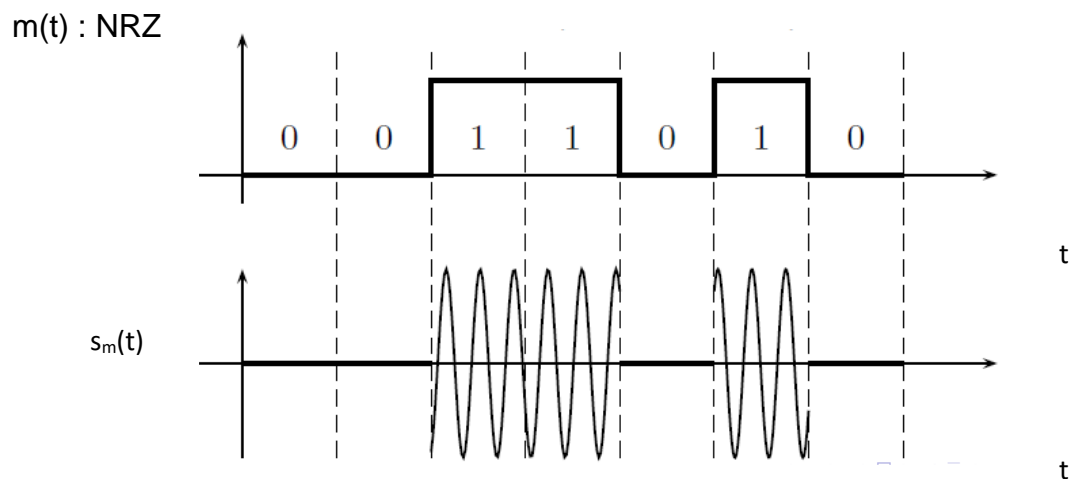


Fig.2

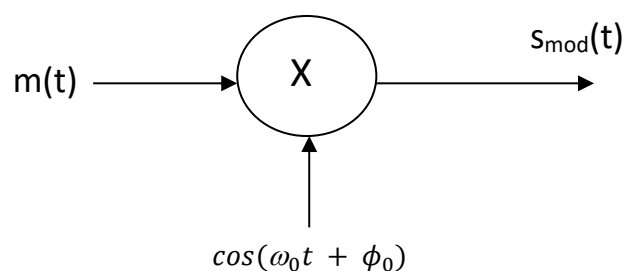


fig.3

La densité spectrale reflète les composantes de la porteuse du signal NRZ.

## 1.2 DEMODULATION

La méthode utilisée est la **démodulation cohérente** figure 4.

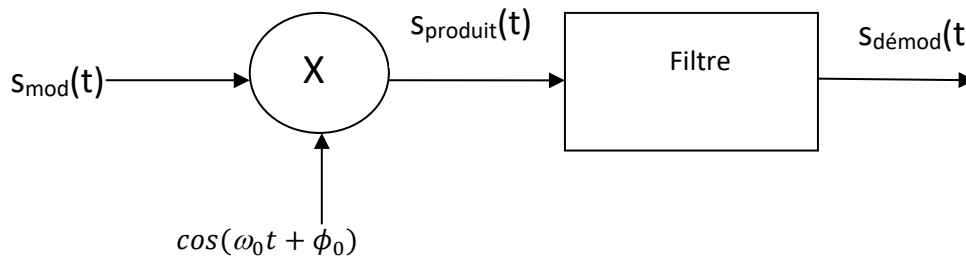


fig.4

Le signal émis après réception est multiplié par la **porteuse reconstituée** comme suit :

$$s_{produit}(t) = s_{mod}(t) \cos(\omega_0 t + \phi_0)$$

$$s_{produit}(t) = \cos(\omega_0 t + \phi_0) \sum a_k g(t - kT) \cos(\omega_0 t + \phi_0)$$

Le terme  $a_k g(t - kT)$  étant indépendant de k peut être sorti du signe somme ; cela donne :

$$s_{produit}(t) = \cos^2(\omega_0 t + \phi_0) \sum a_k g(t - kT)$$

$$s_{dem}(t) = \frac{1 + \cos 2(\omega_0 t + \phi_0)}{2} \sum a_k g(t - kT)$$

Le signal  $s_{produit}(t)$  peut se mettre sous la forme :

$$s_{produit}(t) = \frac{1}{2} \sum a_k g(t - kT) + \cos 2(\omega_0 t + \phi_0) \sum a_k g(t - kT)$$

Le terme  $\cos(2(\omega_0 t + \phi_0)) \sum a_k g(t - kT)$  contient la pulsation  $2\omega_0$  facile à éliminer par un filtre passe-bas.

Il reste le terme :  $s_{produit}(t) = \frac{1}{2} \sum a_k g(t - kT)$ .

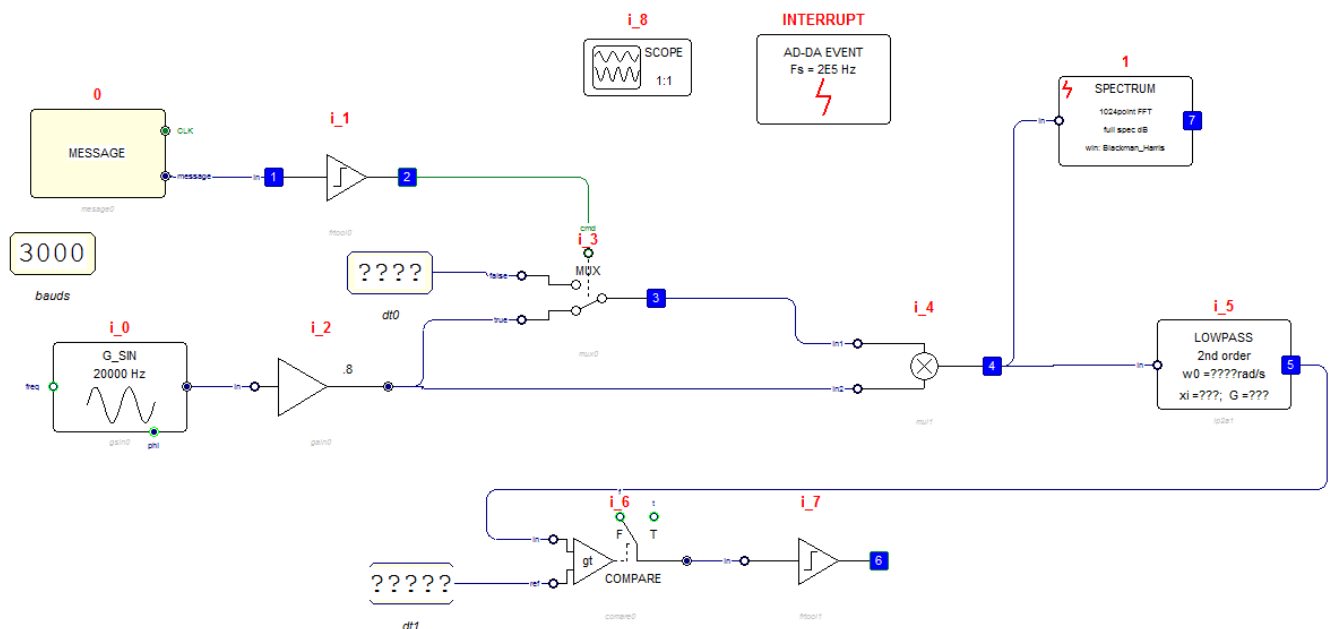
### Remarque

Dans le TP, la porteuse reconstituée est tout simplement la porteuse d'origine pour des raisons de simplification des schémas.



### 1.3 TRAVAUX PRATIQUES

- 1- Ouvrir le fichier **OOK**, compiler et visualiser les signaux CLK, Bande\_base, et le signal à la sortie de Mux en temporel et en fréquentiel. Analyser et interpréter les résultats.
- 2- Ouvrir le fichier **ASK** et visualiser les mêmes signaux qu'à la question précédente. Quelle est la différence par rapport à la modulation OOK ?
- 3- Pour la partie démodulation, visualiser en temporel et en fréquentiel le signal à la sortie du multiplieur, et paramétrer le filtre passe-bande comme il faut pour récupérer le message.
- 4- A l'aide de l'oscilloscope, faire le diagramme de l'œil (CLK voie 1 et sortie LOWPASS voie 2) et en déduire le seuil du bloc COMPARE.
- 5- Visualiser le signal démodulé et comparer au message.



## 2. DEFINITION DU SIGNAL FM

Le signal est modulé en fréquence à une enveloppe constante.

Le message est toujours de la forme :

En transmission numérique le message  $m(t)$  est de la forme :

$$m(t) = \sum a_k g(t - kT)$$

Le signal modulé dans un cas général est de la forme :

$$s_m(t) = \cos(\omega_0 t + \varphi(t) + \phi_0)$$

La fréquence d'un signal périodique est la dérivée de l'angle ou de la phase totale :

$$\Phi(t) = \omega_0 t + \varphi(t) + \phi_0$$

Soit :

$$f(t) = \frac{1}{2\pi} \frac{d\Phi(t)}{dt} = f_0 + \frac{1}{2\pi} \frac{d\varphi(t)}{dt}$$

Dans cette expression on définit :

- La fréquence centrale  $f_0$ ,
- La déviation de fréquence  $df = \frac{1}{2\pi} \frac{d\varphi(t)}{dt}$

**Modulation analogique continue** : la déviation df est une fonction continue du temps.

**Modulation numérique** : la déviation df est une fonction du message numérique variant en Tout Ou Rien (TOR), c'est à dire fonction des bits 0 et 1 du message numérique ; donc fonction des coefficients  $a_k$  du message.

**Modulation FSK** : on aura par exemple :  $df = +\Delta f$  pour un 1 logique

$df = -\Delta f$  pour un 0 logique

Cela ;

$$f_1 = f_0 + \Delta f \text{ pour un 1 logique}$$

$$f_2 = f_0 - \Delta f \text{ pour un 0 logique}$$

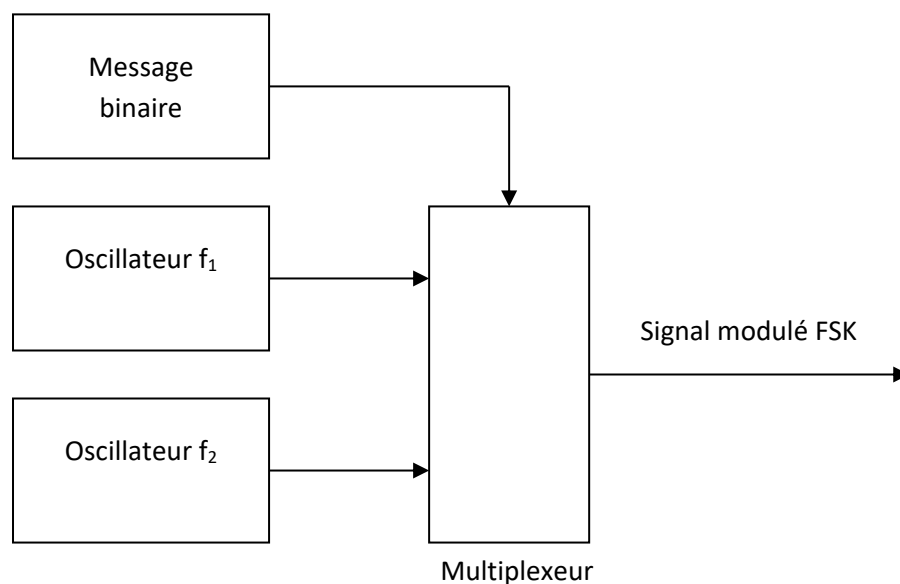
$$f_2 - f_1 = 2\Delta f$$

$$f_0 = \frac{f_1 + f_2}{2}$$

## 2.1 MODULATION FSK à PHASE DISCONTINUE

Les 2 fréquences  **$f_0 + \Delta f$  et  $f_0 - \Delta f$**  sont fournies par 2 oscillateurs différents, donc possédant un déphase quelconque, aléatoire ; ce sont des oscillateurs asynchrones.

On constate lors des changements de niveau logique, les **sauts de tension** du signal modulé.

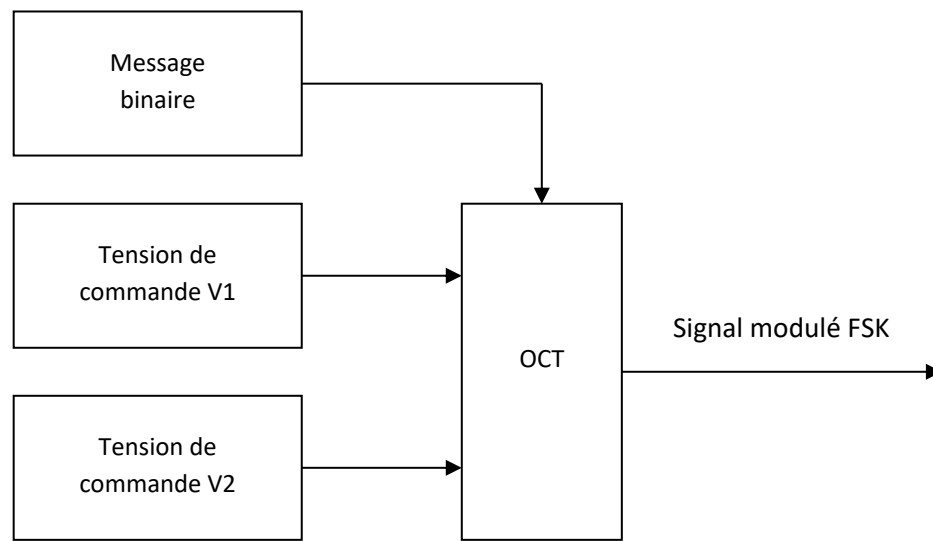


Modulation FSK à Phase Discontinue

## 2.2 MODULATION FSK à PHASE CONTINUE

Les 2 fréquences  $f_1$  et  $f_2$  sont fournies par le même oscillateur contrôlé en tension (OCT ou VCO) ; le signal modulé ne subit **pas de saut de tension** lors du changement de la tension de commande en fonction des niveaux logiques 0 et 1.

Son encombrement spectral est donc moins important.



**Modulation FSK à Phase Continue**

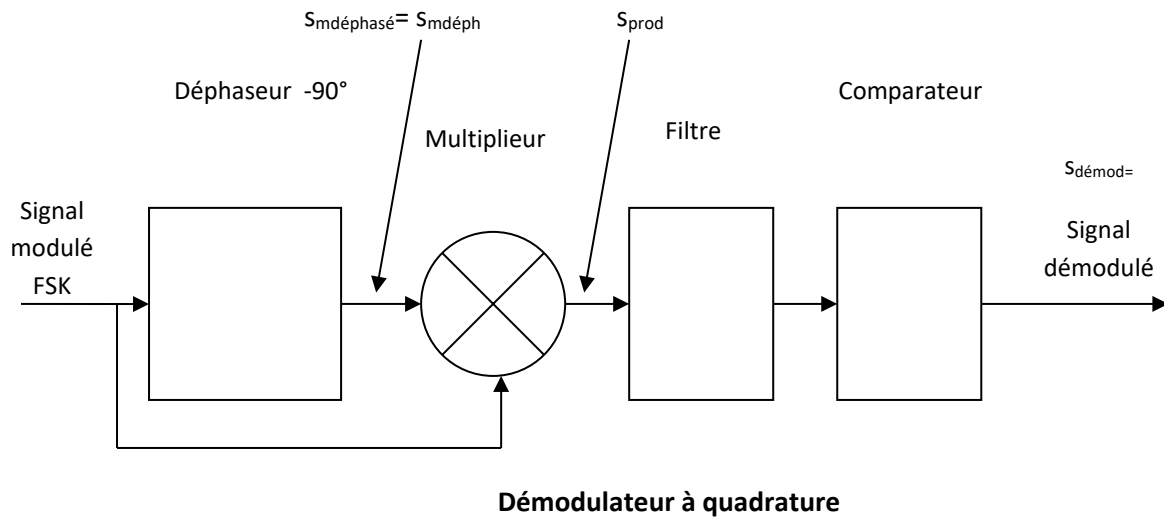
On constate lors des changements de niveau logique, l'**absence de saut de tension** du signal modulé.

## 2.3 DEMODULATION FSK

Plusieurs principes peuvent être utilisés :

- Un filtre passe-bande sur une des fréquences, suivi d'un filtre passe-bas et d'un comparateur de niveau,
- Une PLL,
- Un démodulateur à quadrature.

C'est le dernier qui sera utilisé dans ce TP et pour les 2 types de modulation FSK



Le signal modulé en fréquence est de la forme :

$$s_m(t) = \cos(\omega_0 t + \varphi(t) + \phi_0)$$

Pour simplifier les calculs, supposons **nul le déphasage initial  $\phi_0$** .

Cela donne pour le signal modulé, l'expression :

$$s_m(t) = \cos(\omega_0 t + \varphi(t))$$

A la sortie du déphaseur on trouve :

$$s_{mdéph}(t) = \cos\left(\omega_0 t + \varphi(t) - \frac{\pi}{2} + d\Psi(t)\right) = \sin(\omega_0 t + \varphi(t) + d\Psi(t))$$

Dans cette expression  $d\Psi(t)$  est la variation de phase autour  $\frac{\pi}{2}$  **pour de faible variation de fréquence autour de  $f_0$**  à la sortie du filtre déphaseur.

A la sortie du multiplieur on trouve :

$$s_{prod} = \sin(\omega_0 t + \varphi(t) - d\Psi(t))\cos(\omega_0 t + \varphi(t))$$

Cette expression de la forme :  $\sin a \cdot \cos b$  donne

$$\frac{\sin(a+b) + \sin(a-b)}{2}$$

La somme des angles donne :  $2(\omega_0 t + \varphi(t)) - d\psi(t)$

La différence des angles donne :  $d\psi(t)$

Cela donne :

$$s_{prod} = \frac{1}{2} \left[ \sin \left( 2(\omega_0 t + \varphi(t)) + d\psi(t) \right) + \sin (d\psi(t)) \right]$$

Le terme  $\sin \left( 2(\omega_0 t + \varphi(t)) + d\psi(t) \right)$  sera éliminé par le filtre passe-bas.

Il reste le terme  $\sin(d\psi(t))$  *qui peut être assimilé à  $d\psi(t)$*

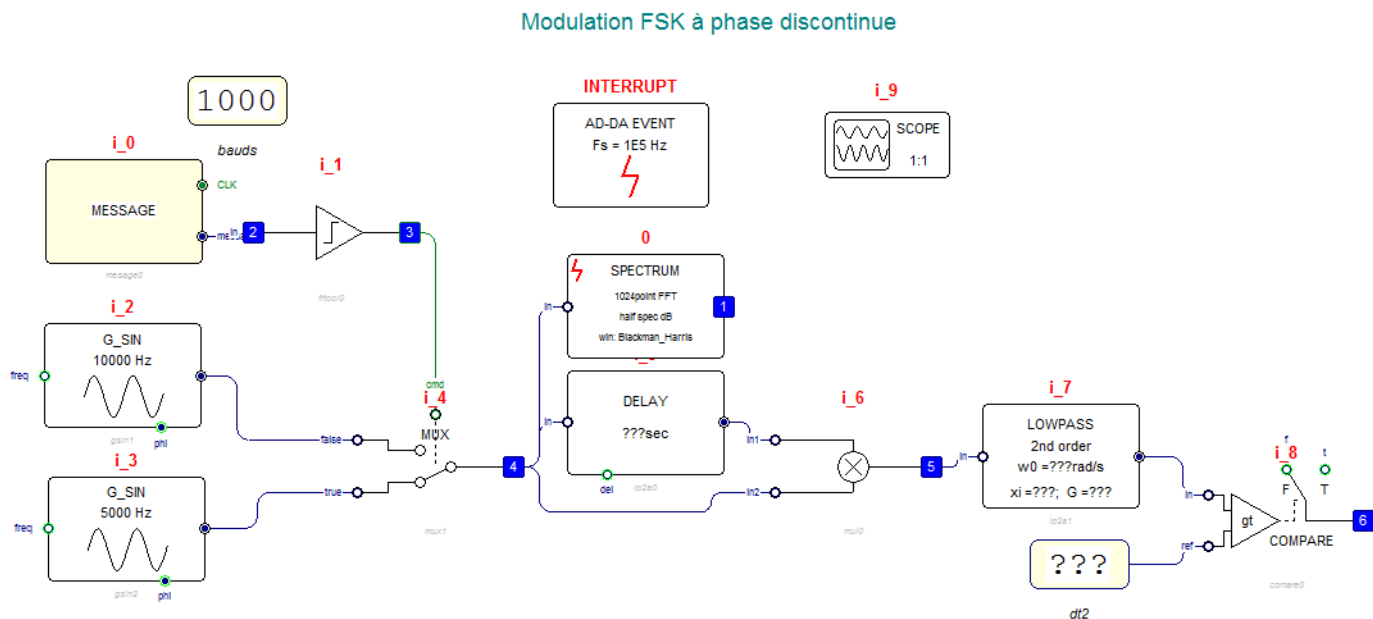
Ce terme est proportionnel à l'amplitude du message modulant.

On obtient donc :

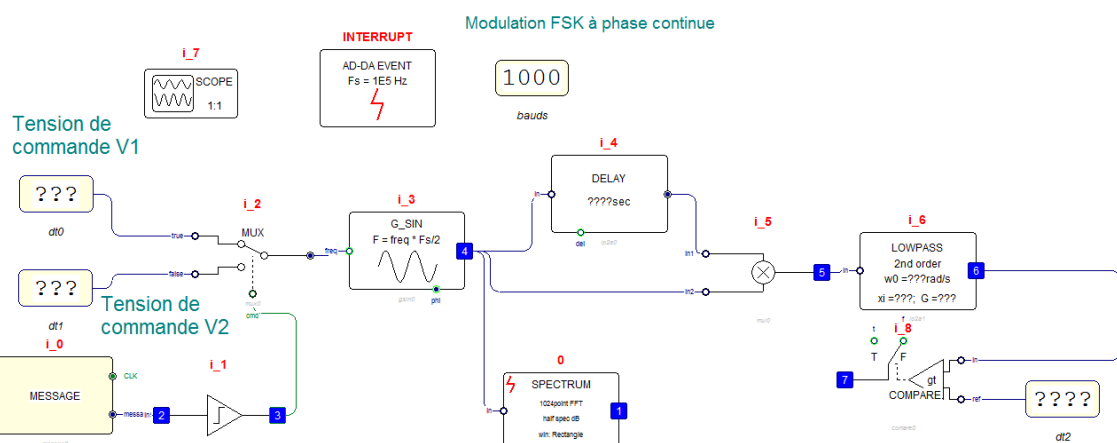
$$s_{demod} \cong \frac{1}{2} d\psi(t)$$

## 2.4 TRAVAUX PRATIQUES

- 1- Ouvrir le fichier **FSK1**, compiler et visualiser les signaux CLK, Bande\_base, et le signal à la sortie de Mux en temporel et en fréquentiel. Analyser et interpréter les résultats.



- 2- Ouvrir le fichier **FSK2**, compiler et visualiser les signaux CLK, Bande\_base, et le signal à la sortie de Mux en temporel et en fréquentiel. Comparer aux résultats obtenus dans FSK1?



- 3- Pour la partie démodulation, visualiser en temporel et en fréquentiel le signal à la sortie du multiplieur, et paramétrer le filtre passe-bande pour récupérer le message.

- 4- A l'aide de l'oscilloscope, faire le diagramme de l'œil et en déduire le seuil du bloc COMPARE.
- 5- Visualiser le signal démodulé et comparer au message.



## ***TP3 : MODULATION NUMERIQUE PSK-QAM***



*L'objectif de ce TP est d'étudier les diagrammes de constellation, de l'œil et l'encombrement spectral des modulations PSK et QAM et leur sensibilité au bruit.*

## 1- RAPPELS THEORIQUES

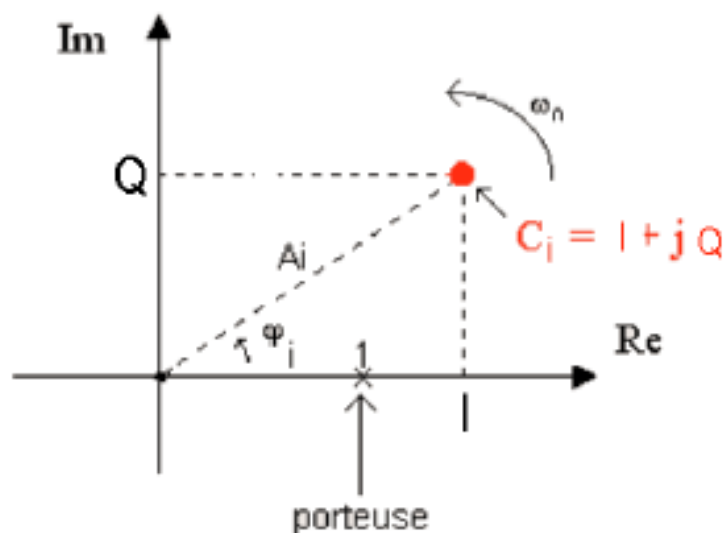
### 1.1 Paramètres IQ

- L'expression du symbole  $m(t) = I \cos(\omega_c t) + Q \cos(\omega_c t + \phi)$

I est le paramètre en phase avec la porteuse,

Q le paramètre en quadrature.

- La représentation graphique du symbole dans le plan complexe est la suivante :

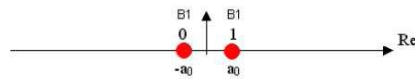


### 1.2 Constellation

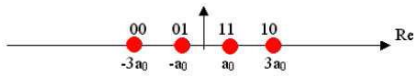
C'est la représentation graphique de l'ensemble des symboles.

Quelques exemples de constellation :

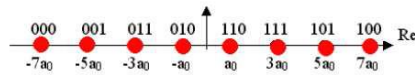
ASK-2



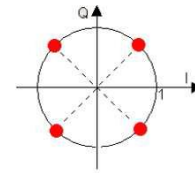
ASK-4



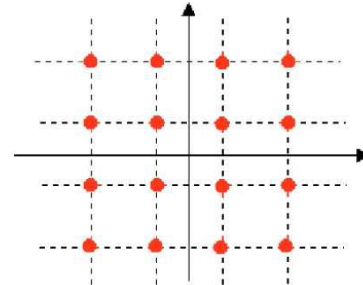
ASK-8



PSK4



MAQ16



### 1.3 Spectre

- En bande de base (pour un signal NRZ):

Le spectre est autour de 0 et l'encombrement est  $1/T_B = D$

- En modulation numérique :

C'est la durée du symbole  $T_S$  ( $1/R$ ) qui importe ( $R$  étant la rapidité de modulation).

On rappelle que  $T_S = n T_B$ ,  $n$  étant le nombre de bits par symbole.

Le spectre est transposé autour de la porteuse et l'encombrement est :  $2 \times 1/T_S = 2 R$

### 1.4 Valence – Rapidité de modulation - Débit

La rapidité de modulation est limitée par la bande passante du canal.

Pour augmenter le débit on est alors obligé d'augmenter la valence en construisant chaque symbole sur  $n$  bits :

- Si  $n = 1$  Valence  $V = 2$   $D = n \times R = 1 \times R = R$

- Si  $n = 2$  Valence  $V = 4$   $D = n \times R = 2 R$

- Si  $n = 4$  Valence  $V = 16$   $D = n \times R = 4 R$

Remarquons que l'augmentation de la valence nous permet à la fois d'augmenter le débit et à la fois de diminuer l'encombrement spectral :

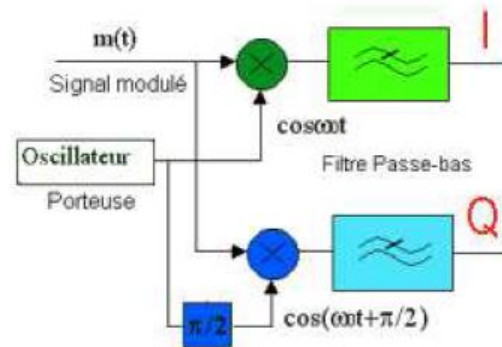
- Si  $n = 1$  Valence  $V = 2$   $R = D / 1 = D$  BP =  $2 R = 2 D$

- Si  $n = 2$  Valence  $V = 4$   $R = D / 2$   $BP = 2 R = 2 D / 2 = D$
- Si  $n = 4$  Valence  $V = 16$   $R = D / 4$   $BP = 2 R = 2 D / 4 = D / 2$

### 1.5 Diagramme de l'œil

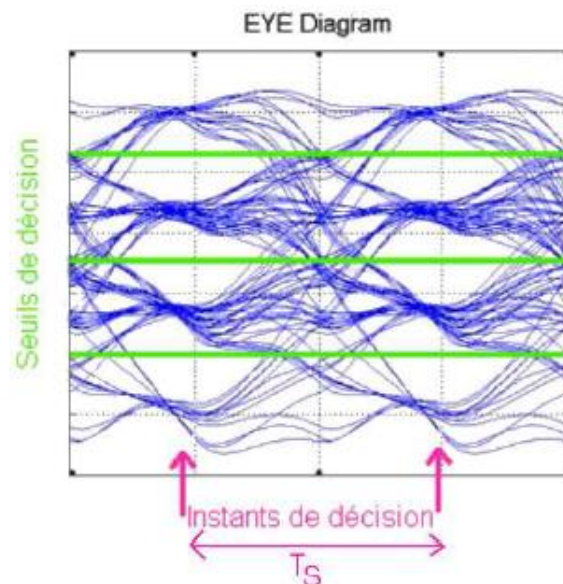
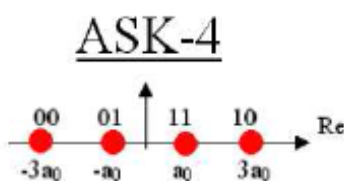
Le premier étage en réception est de démoduler le signal afin de régénérer les paramètres I et Q.

Le diagramme de l'œil de I (puis de Q) va nous permettre de qualifier la transmission (œil ouvert  $\Rightarrow$  pas d'erreurs), de mesurer la rapidité de modulation ( $T_s$  = temps entre 2 instants de décision), de calculer le ou les comparateurs permettant de régénérer les signaux I et Q (seuils de détection).



Par exemple, pour ASK4, le diagramme de l'œil peut avoir l'allure ci-contre.

On voit apparaître les 4 niveaux (puisque'il y a 4 valeurs de I possibles) et la durée du symbole  $T_s$ .



## 2. Manipulation

### 2.1 Diagramme de constellation

Ouvrir le fichier **constellation** et observer  $I(t)$  ( $\text{Re}(\text{MAP})$ ),  $Q(t)$  ( $\text{Im}(\text{MAP})$ ), et les diagrammes de constellation (mode XY avec  $I(t)$  en X et  $Q(t)$  en Y) des modulations BPSK, QPSK et QAM16 sur l'oscilloscope virtuel et sur l'oscilloscope réel.

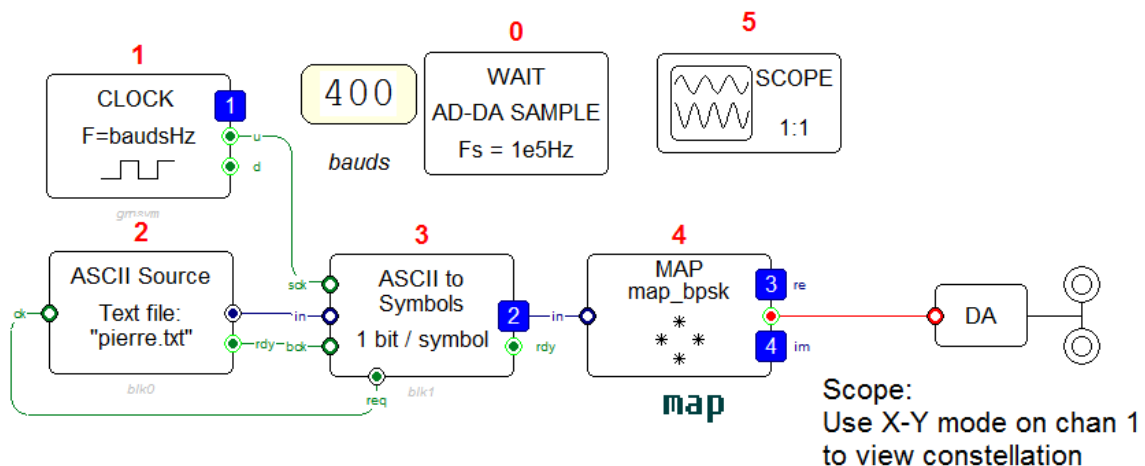


Figure 5

Sur le diagramme de constellation, sur chaque point mettre le symbole binaire correspondant.

### 2.2 Diagramme de l'œil

- 1- Ouvrir le fichier **Oeil** et observer sur l'oscilloscope pour un canal sans et avec bruit le diagramme de l'œil des modulations BSK, QPSK, QAM16.
- 2- Pour chaque cas, donner les instants de décision, le nombre de niveaux et le ou les seuils de décision.

**Procédure pour observer le diagramme de l'œil sur l'oscilloscope :**

**Sur la voie 1 envoyer l'horloge et sur la voie 2, le signal  $I(t)$  démodulé (partie réelle de la sortie du démodulateur) ; et choisir persistance infinie.**

- 3- Observer sur l'oscilloscope virtuel les spectres en dB des signaux modulés avec et sans bruit.
- 4- Conclure sur la bande passante et la sensibilité au bruit de chaque type de modulation.

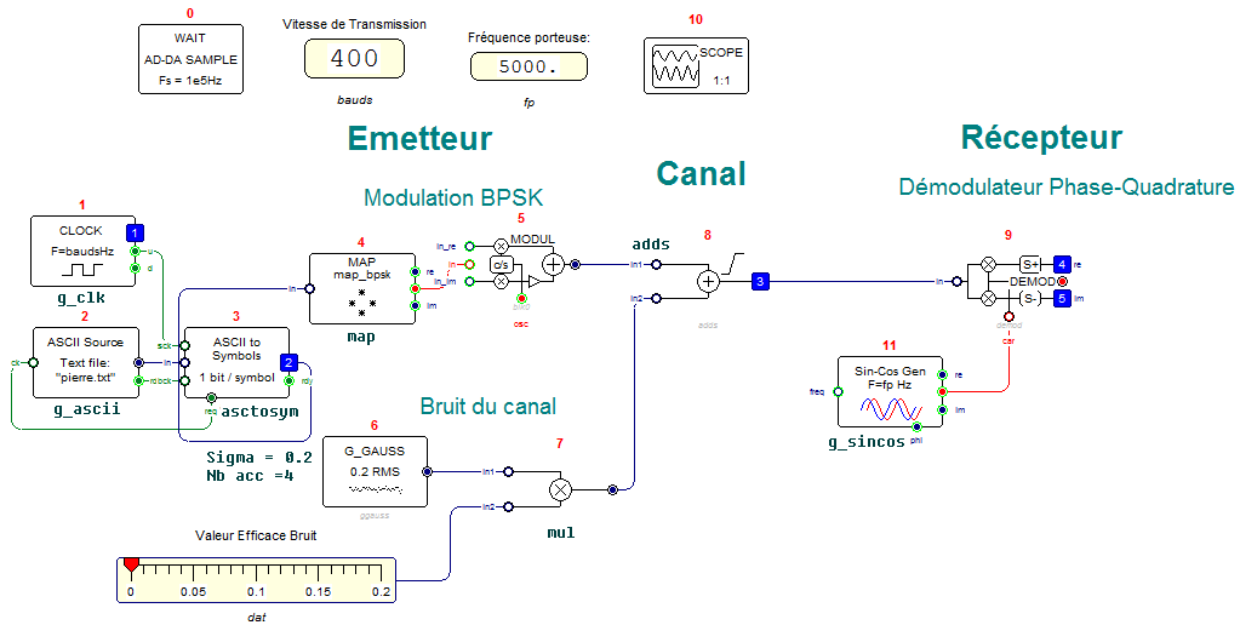


Figure 6

## 2.3 MODULATION ET DEMODULATION BPSK

Ouvrir le fichier **modulation\_BPSK** compiler et :

- 5- Visualiser le signal modulé et son spectre. Expliquer comment est modulé le signal.
- 6- Compléter les blocs filtres et surtout comparateur de la démodulation après avoir observé le diagramme de l'œil. On pourra aussi rajouter du bruit dans le canal pour voir l'influence sur le niveau du comparateur.

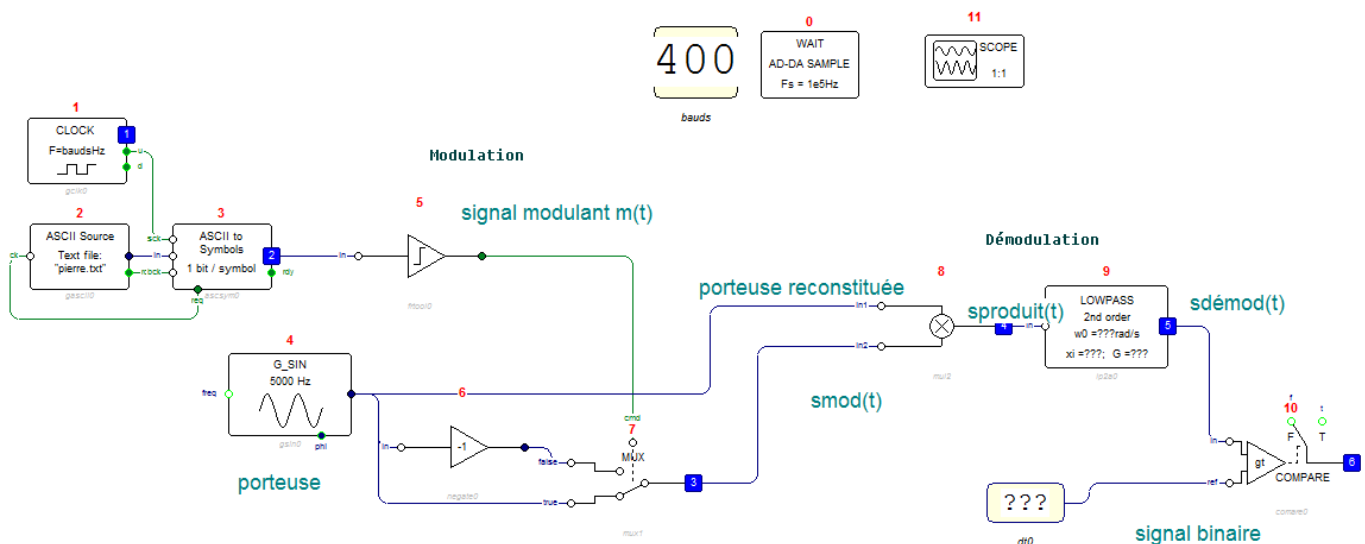


Figure 7





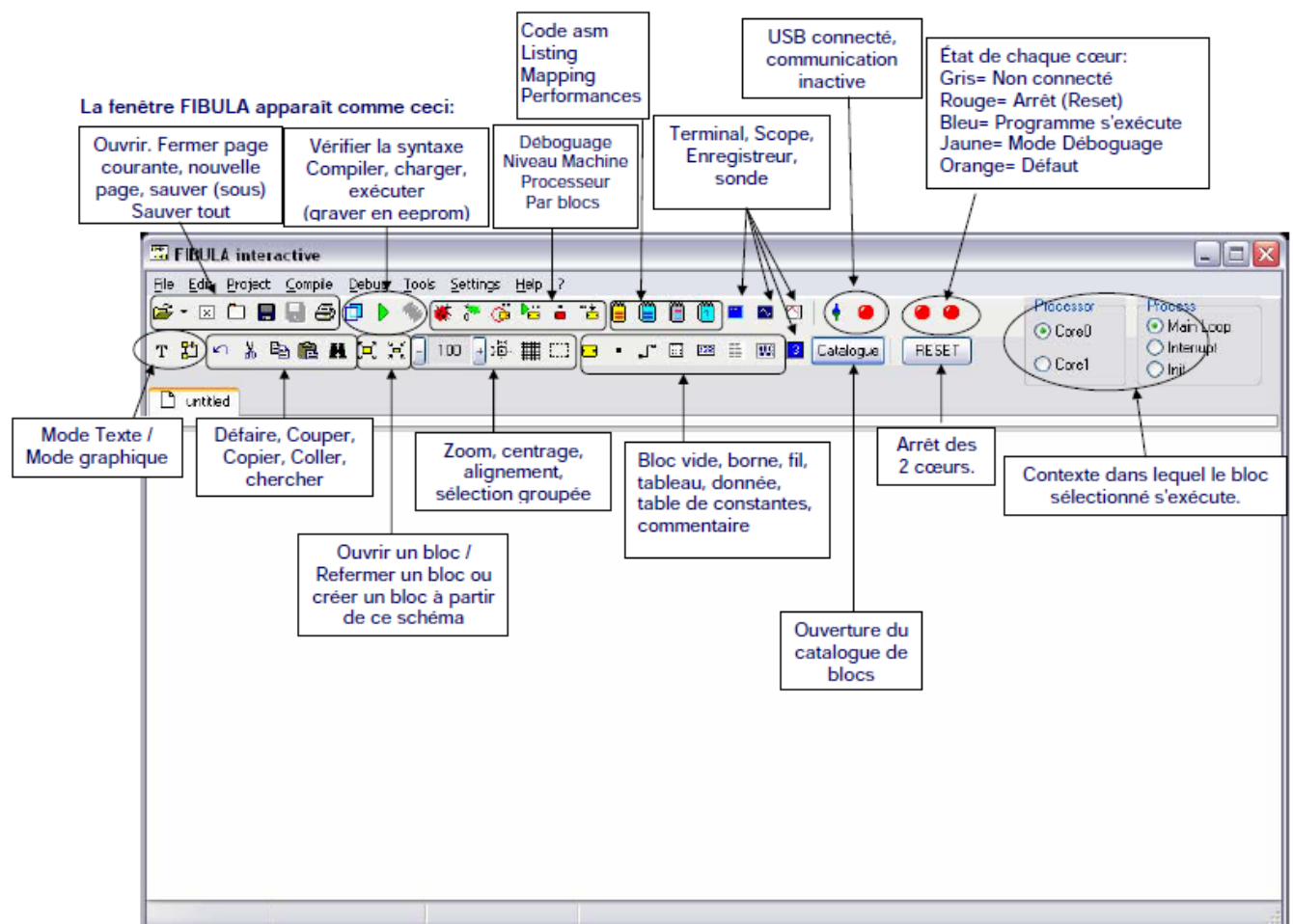
## ANNEXE : PRISE EN MAIN DU MODULE ETD410000 SOUS FIBULA GRAHIQUE

### 1- Utilisation de FIBULAG

Raccorder le module ETD41000 à l'ordinateur PC.

Mettre le module ETD410000 sous tension.

Lancer le logiciel FIBULA et vous verrez apparaître l'environnement suivant.



Normalement le logiciel détecte le matériel tout seul s'il n'y a aucun défaut.

Il faut vérifier que le module ETD410000 est bien détecté :

	ETD410000 non connecté ou hors tension
	ETD410000 détecté et sous tension

Si ce n'est pas le cas, vérifier votre câblage.

La version logicielle de Fibula-Graphic v3 est compatible avec toutes les versions matérielles ETD410.

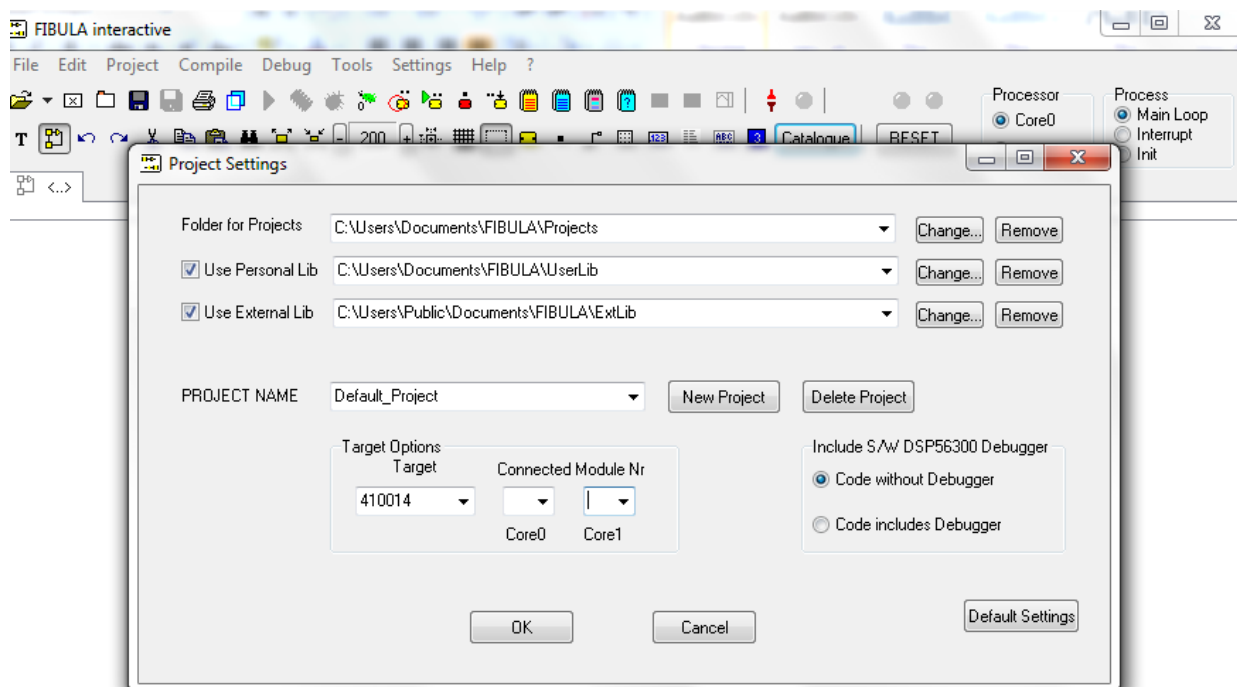
La sélection de la version du matériel par le logiciel se fait souvent de manière automatique.

Afin de permettre à chaque utilisateur de développer ses propres fonctions, il est possible de renseigner un espace de travail qui lui est propre en cliquant sur le menu « **Project** ».

Choisir un répertoire personnel pour « **Folder to Projects** » et « **Use Personal Lib** » ; et laisser le répertoire par défaut pour « **Use external Lib** ».

Choisir un nom de projet (exemple TP1).

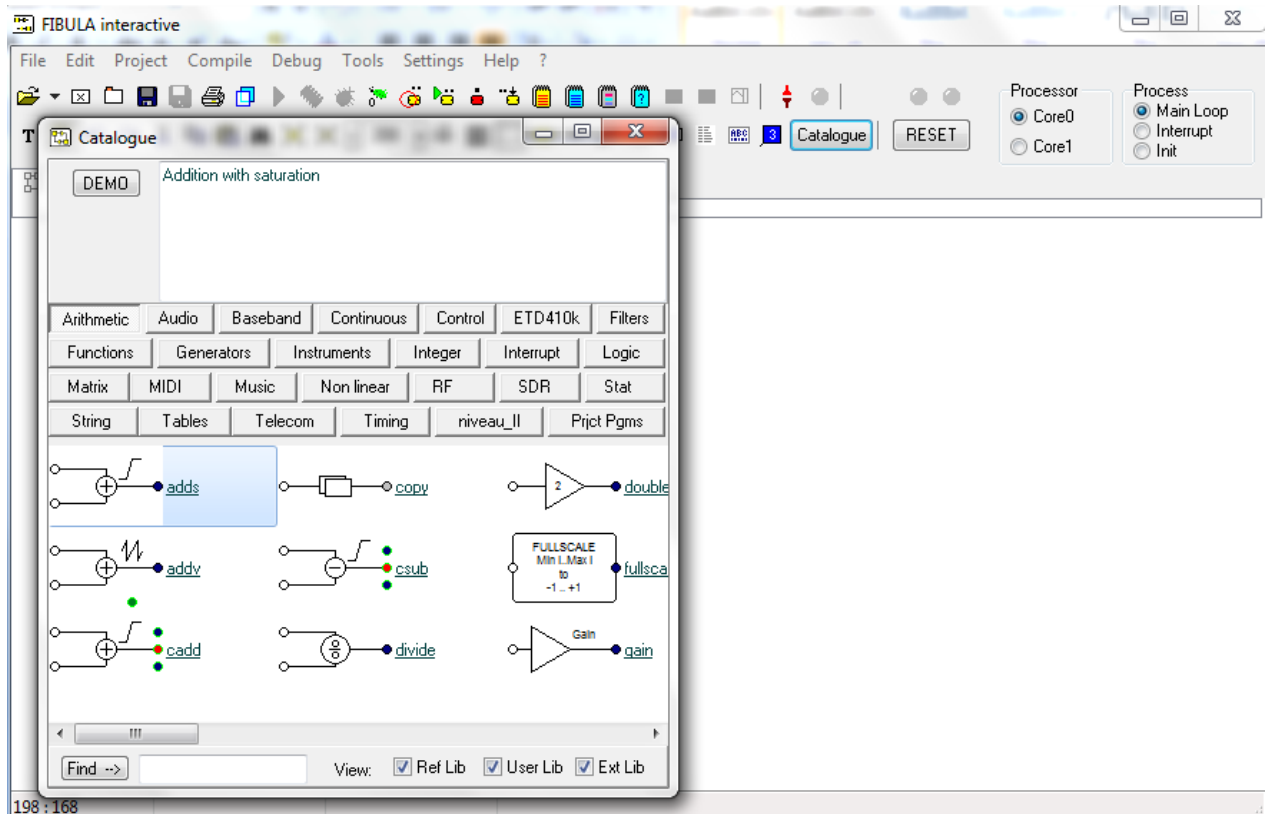
Vous pouvez aussi voir sur cette fenêtre la version exacte du matériel.




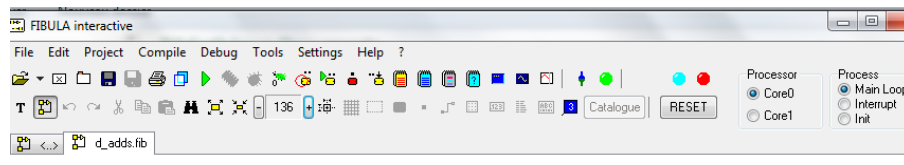
## 2-Chargement d'une DEMO

Cliquer sur le bouton **catalogue** et sélectionner la catégorie Arithmétique.

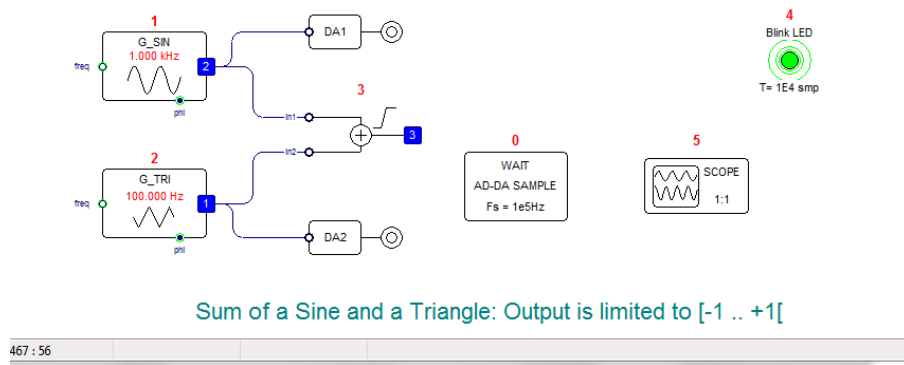
Cliquer sur le block Adds, puis sur le bouton **DEMO**



La démo de l'additionneur apparaît. Enregistrer la dans votre répertoire projet et cliquer sur  pour construire le programme et le télécharger dans la carte ETD410000.



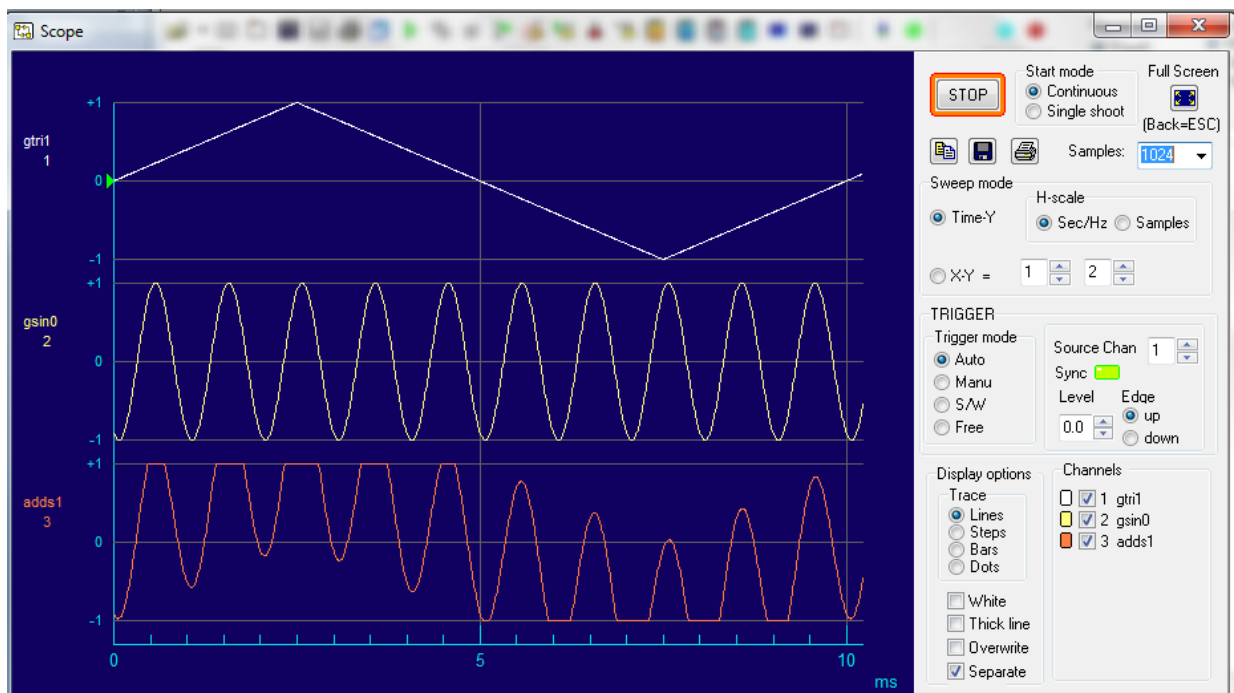
### Demo Add with Saturation



Rouge	Etat Reset
Orange	Problème de téléchargement
Vert	Programme en cours d'exécution

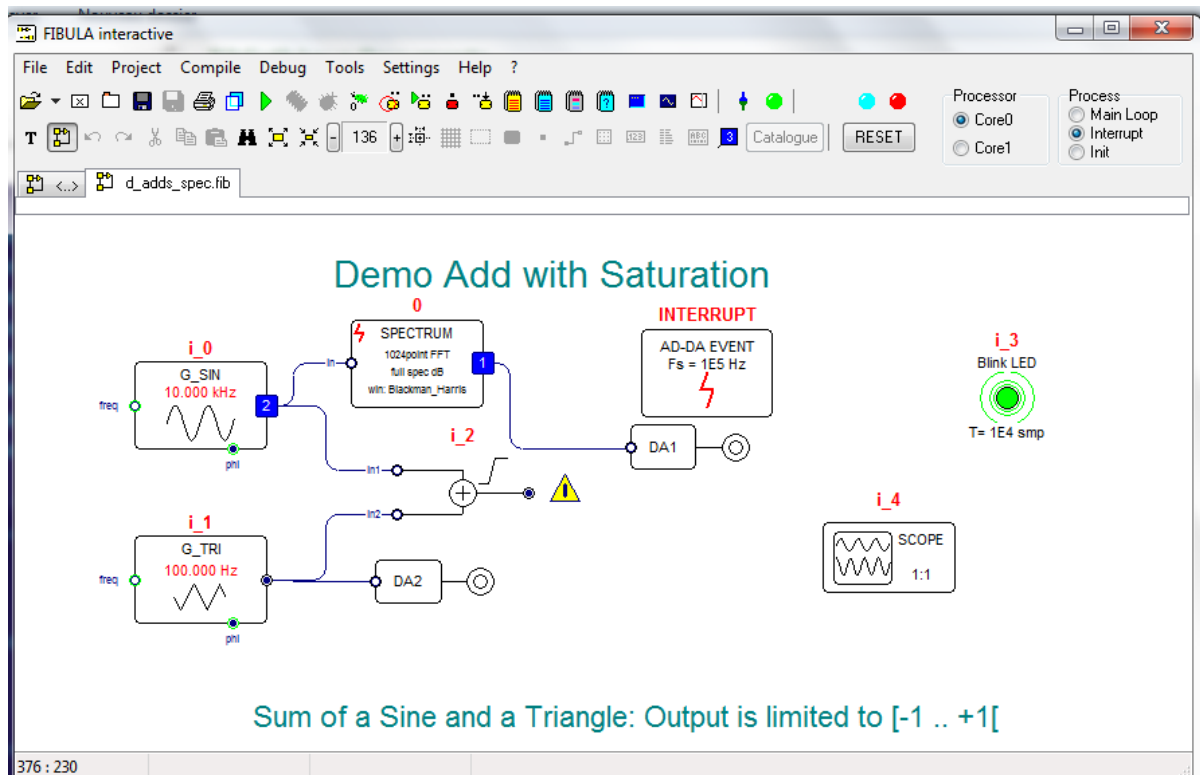
Exemple sur la figure ci-dessus : cœur 0 est en cours d'exécution, cœur 1 est à l'arrêt

Pour visualiser les courbes à l'oscilloscope virtuelle, il faut cliquer sur 

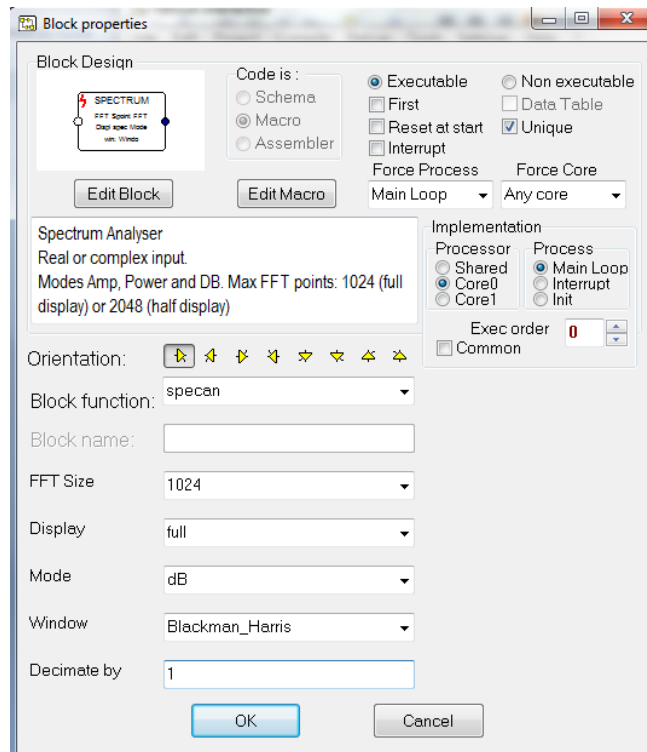


Pour visualiser les courbes sur l'oscilloscope, connecter les câbles BNC sur les sorties DA1 ou DA2 de la maquette.

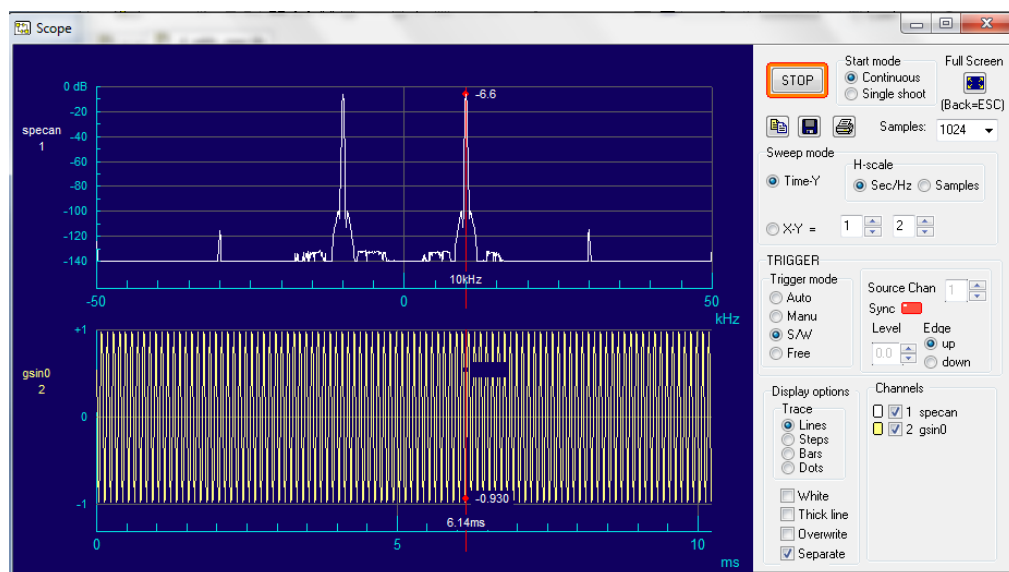
Les signaux visualisés sont des signaux temporels ; pour visualiser le signal sinusoïdal en spectrale, il faut réaliser le montage suivant :



- Remplacer le block **WAIT** par le **block INTERRUPT** (catalogue puis interrupt)
- Placer un analyseur de spectre **SPECTRUM** (catalogue puis instrument) puis après un click droit sur ce bloc, faire les réglages suivants :

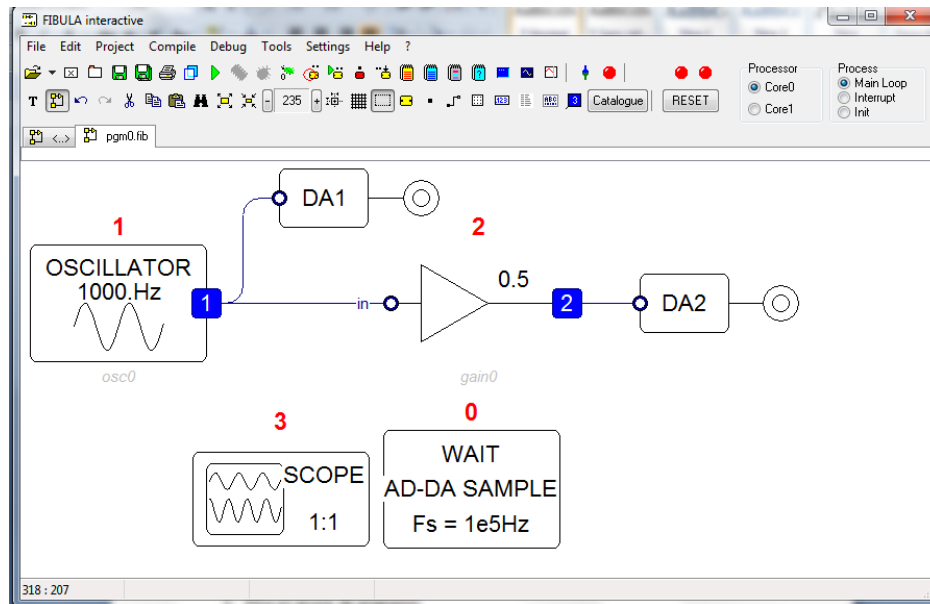


Mettre tous les blocs en interrupt sauf les blocs SPECTRUM et INTERRUPT (main loop)



### 3- Création d'un programme

Réaliser le montage suivant :



- Oscillator (catalogue, Generators)
- Gain (catalogue, Arithmetic)
- DA1 et DA2 (catalogue, ETD410K)
- WAIT AD-DA échantillonneur qui permet de définir la base de temps (catalogue, ETD410K)
- SCOPE (catalogue, instruments)

Régler le gain à 0,5 puis enregistrer et compilez le programme.


## 4- Création d'un Block


Nous venons de créer un programme, puis de l'enregistrer en tant que programme de référence.

Sur le même principe, il est possible de créer un Block, une fonction, propre.

Sous Fibula, tous les générateurs de fonctions sortent un signal borné entre -1 et +1.

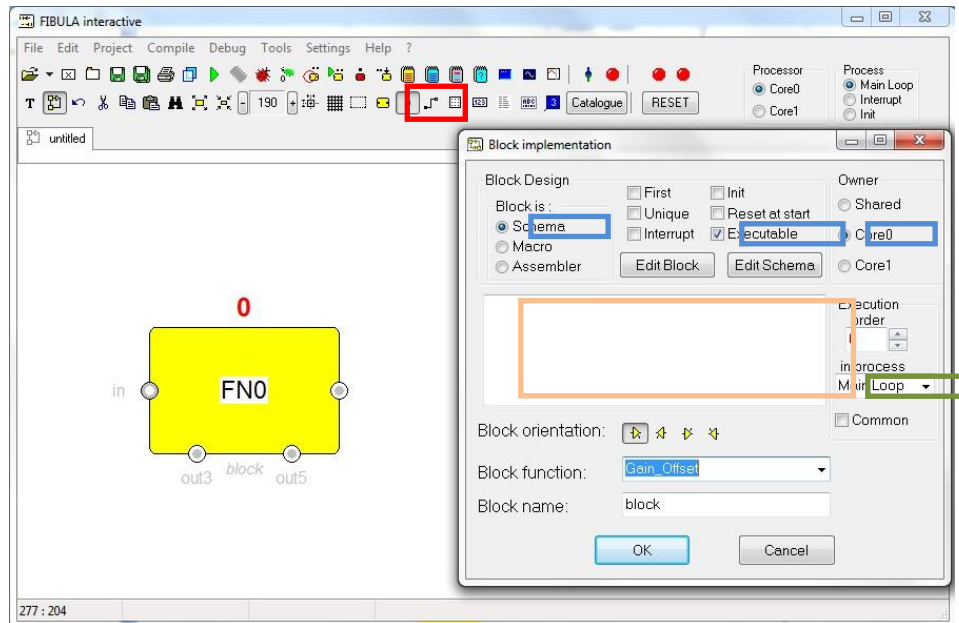
Il peut donc être utile de réaliser un Block qui permet d'avoir un générateur avec gain et offset variable.

Pour créer un block, cliquez sur l'icône : , puis cliquez dans la fenêtre de FibulaG.

Cliquez ensuite sur l'icône  pour ajouter des points d'entrées / sorties.

Dans notre exemple, nous avons besoin de 3 entrées :

- Le signal entrant,
  - Le réglage de gain,
  - Le réglage d'offset,
- et d'une sortie ; le signal de sortie



Faites un clic droit sur le block, et attribuez lui un nom dans le champ « block function ». Par exemple Gain\_Offset.

**Avant de cliquer sur OK**, vérifiez que ;

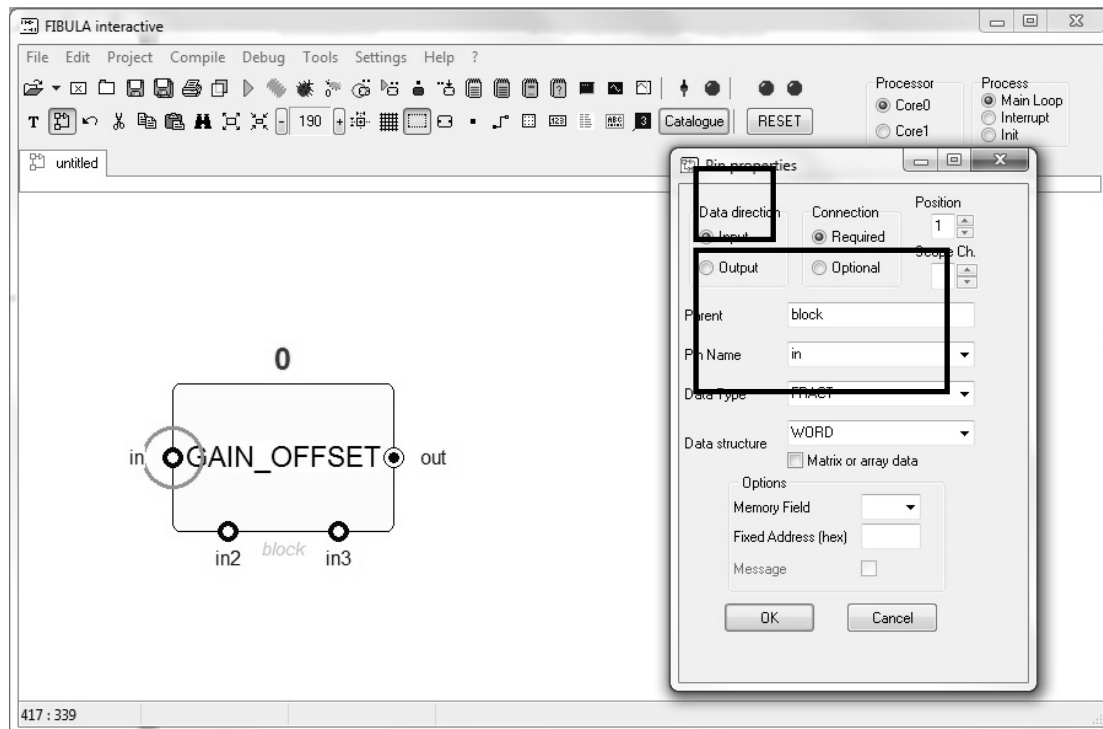
- Les options Schéma, Executable, Core0 soient cochés,
- Que « Main Loop » soit sélectionné dans « in process »
- Ajouter si vous le souhaitez un commentaire explicatif dans le champ au-dessus de block orientation.

Pour définir le type d'entrée/sortie, effectuer un clic droit sur le point,

Sélectionner « Input » dans « Data direction » puisqu'il s'agit d'une entrée (Pour une sortie, sélectionnez output).

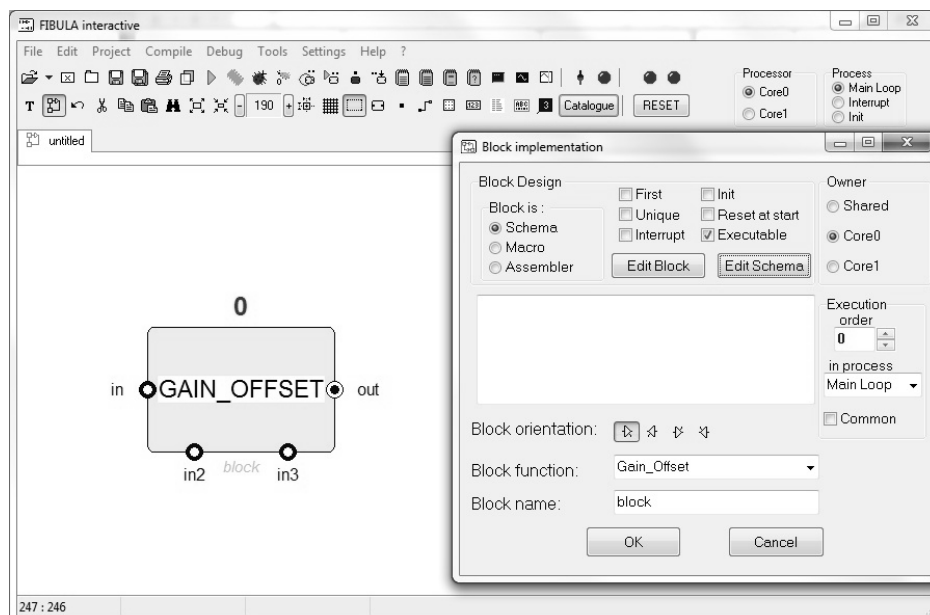
Dans « Pin Name » entrer le nom souhaité, puis le type de données « FRATC », une data structure « WORD ».





Configurer chacune des entrées et la sortie sur le même principe.

Pour créer votre structure interne, cliquer droit sur le block puis sur « Edit Schéma ».

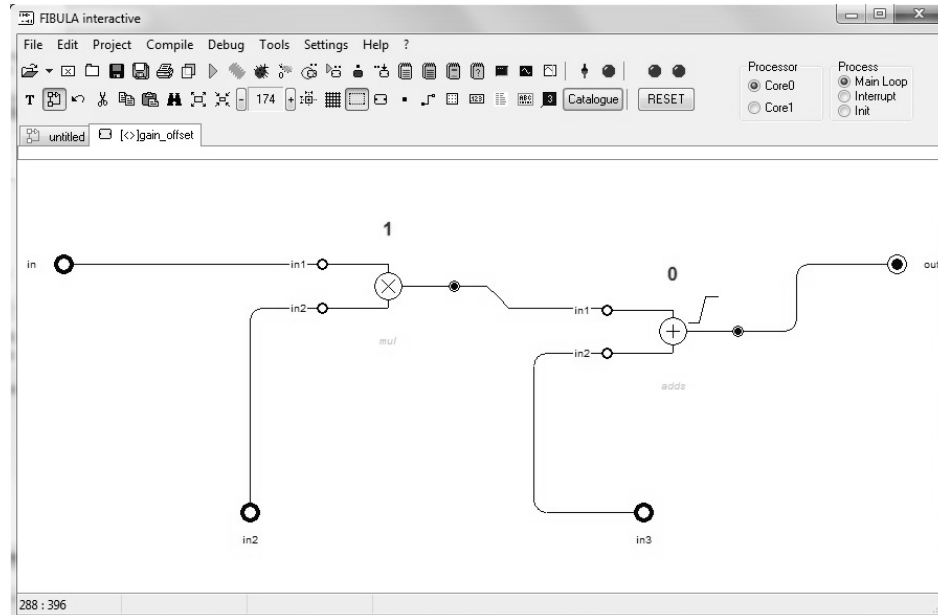


Un nouvel onglet s'ouvre.

Vous y retrouvez les points d'entrées / sortie définis préalablement.

Vous pouvez créer la fonction souhaitée avec les blocks existant dans le catalogue.

Puis effectuer les liaisons internes.




Le multiplieur nous permet de régler le niveau de notre signal.

L'additionneur permet d'ajouter un offset.

Pour enregistrer votre block, cliquez sur « File » puis « Save As ... »,

FibulaG se place automatiquement dans le sous dossier Blocklib de votre répertoire personnel.

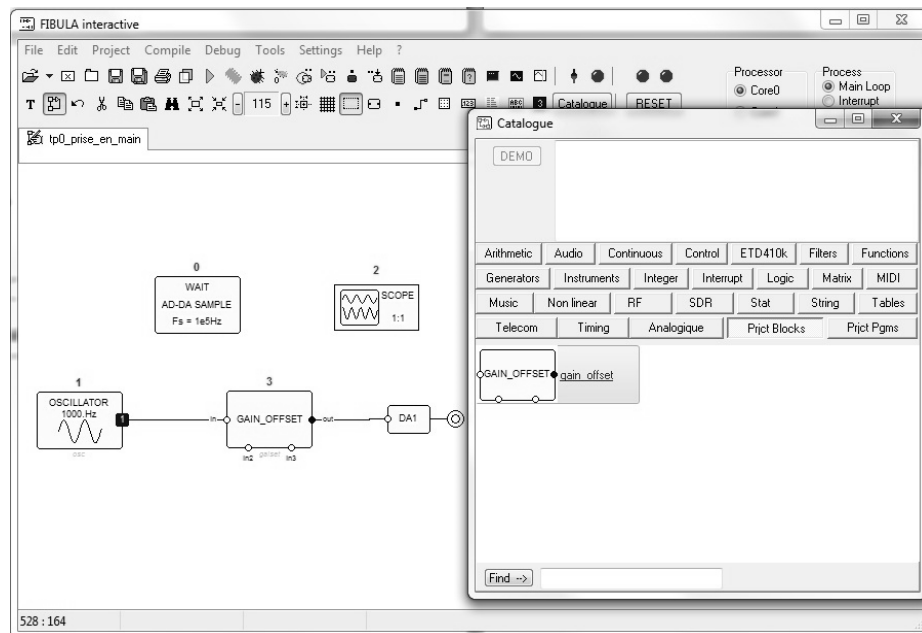
Vérifier que le nom du fichier est bien Gain\_offset, et cliquez sur Enregistrer.

Ouvrez un nouvel onglet sous Fibula avec l'icône , puis ouvrez le **Catalogue** et dans la catégorie **Prjct Block** vous retrouverez le block créé.

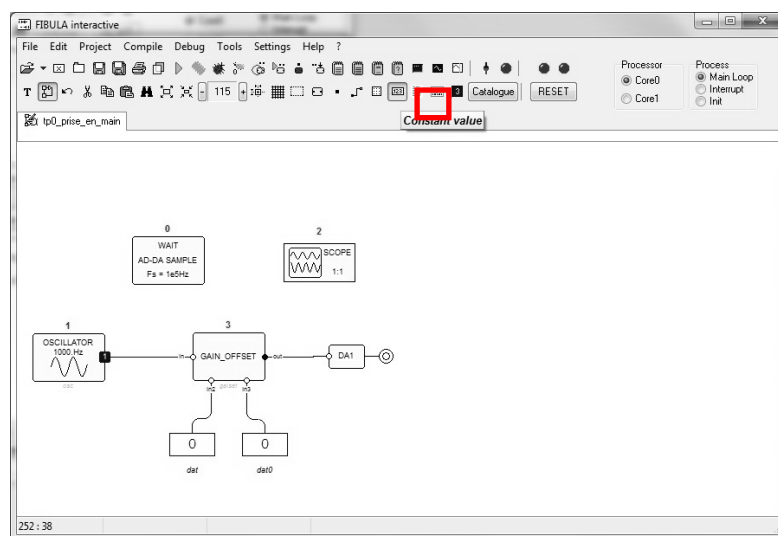
## 5- Exploitation des Programmes et Blocks Personnels

Nous avons créé et enregistré un programme et un block précédemment, nous allons nous en servir en exploitant les fonctions interactives de Fibula.

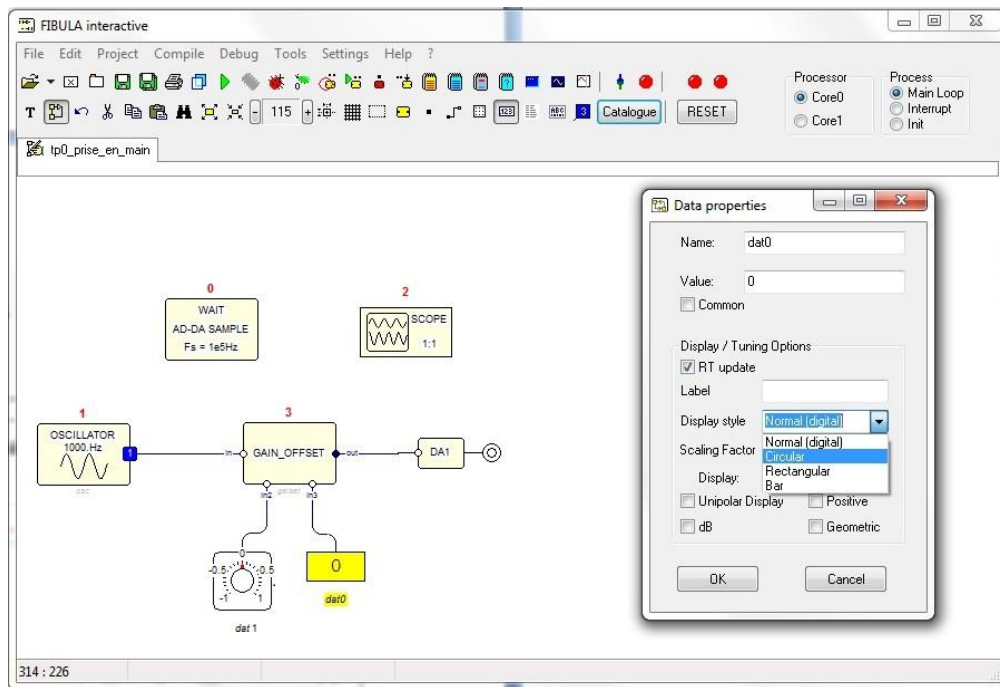
Reprenez votre 1<sup>er</sup> programme créé et remplacez le bloc gain par le bloc Gain\_Offset que vous avez créé.





Ajouter ensuite deux valeurs constantes, et reliez-les sur in1 et in2.



Faites un clic droit sur l'un des blocks « constante value », et sélectionnez l'option « Circular » dans le « display style »



Ajoutez une sonde d'oscilloscope en sortie du block Gain\_Offset.

Exécutez le programme avec l'icône , puis lancez l'oscilloscope virtuel avec l'icône , et cliquez sur START.