

Programme

- 2011.12.06 : Conférence
- 2011.12.14 : TP1, Représentation numérique
- 2012.01.06 : TP2, Acquisition de signaux
- 2010.01.12 : TP3, A définir

Objectifs

- Démystifier LabVIEW
- Faire des protos structurés
- Faire un peu d'acquisition de signaux
- Si possible un peu de RT / embarqué



LabVIEW pour quoi?

LabVIEW pour qui?

Pro ou Proto?

- Le vieux débat...
 - La porte ouverte à l'informatique pour tous?
 - Un environnement pour professionnel du développement de logiciel?
- Deux axes complémentaires
 - « Easy to use » >> Express VI
 - Une suite professionnelle >> projet, OOP, SCC

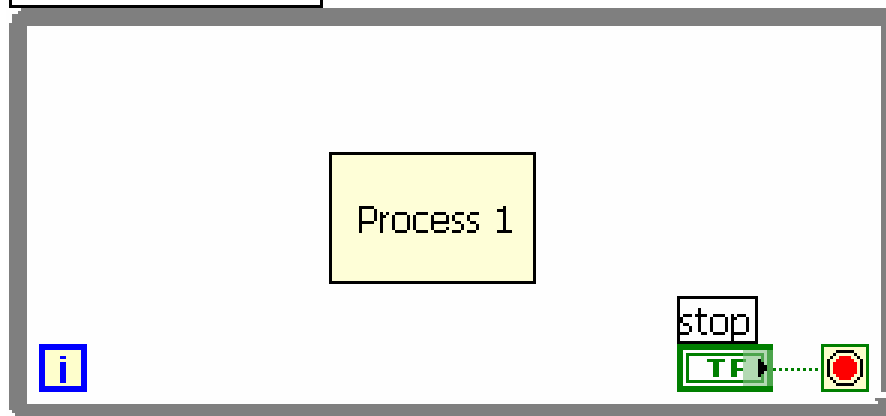
Pas faux mais injuste

- Les critiques (trop) faciles
 - « c'est un vrai paquet de nouille »
 - pieuvre géante
 - pas de visibilité d'ensemble
- >> La faute ne revient pas à l'outil
- >> L'outil met en évidence le mauvais usage qu'on en fait, c'est une qualité!

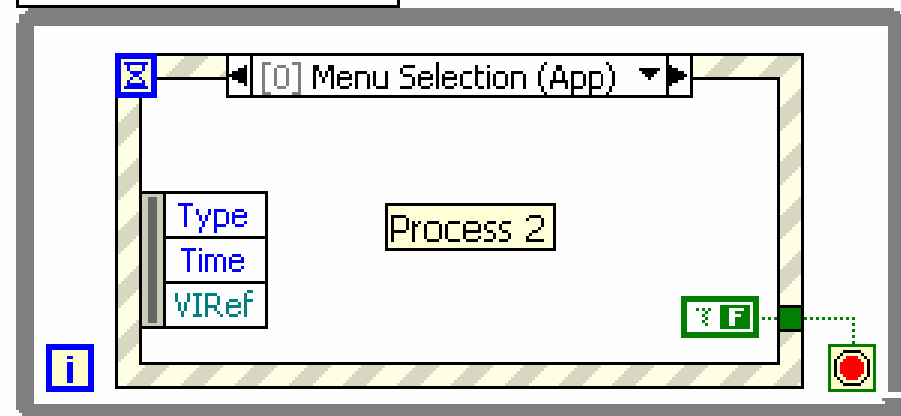
Ingratitude

- LabVIEW, outil de prototypage d'IHM?
- >> Définir l'IHM est un bon moyen de définir le CDC d'une application
- Quoi de plus naturel qu'une représentation graphique?

boucle de services



boucle événementielle



Respect mutuel

- Amateur ≠ Professionnel
 - leur tâches ne sont pas les mêmes
 - l'amateur mesure ce qui le sépare du « Pro »
 - le « Pro » sait se comporter en amateur quand il le faut
- Et chacun adapte l'effort à l'enjeu!

Nominal / Singulier

- Le nominal : ce que doit faire le logiciel
- Le singulier : ce qu'il ne devrait pas faire (planter, ...)
- La loi du 80/20
 - 80% du temps pour faire 80% du travail
 - et encore 80 % pour faire le reste!

C'est en forgeant...

- « Easy to use » & « ready to go »... mon œil!
- LabVIEW n'est pas « magique »
- Les « wizards » SONT pertinents...
- ... il faut un « Pro » pour faire du travail de « Pro »

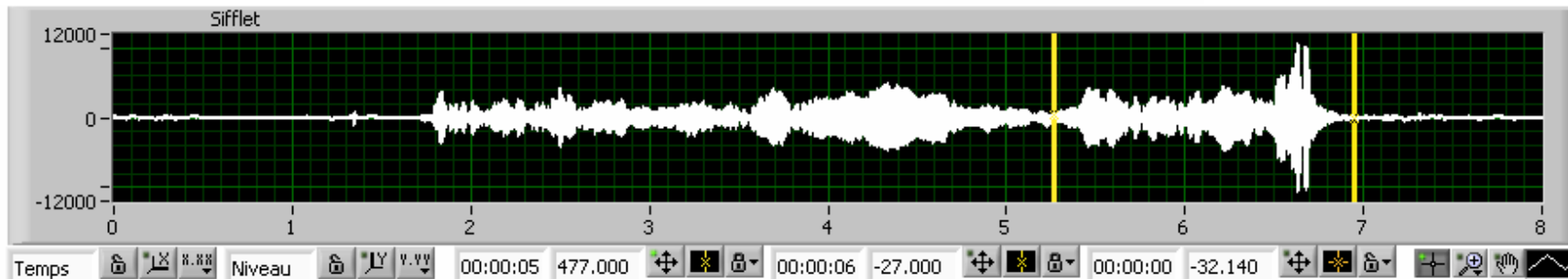
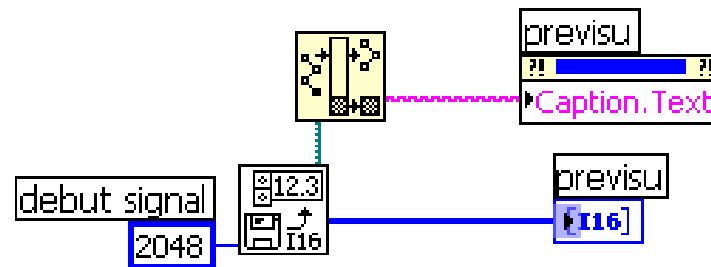
Quelle surprise!

Un exemple

Prenons un objectif simple (besoin courant) :

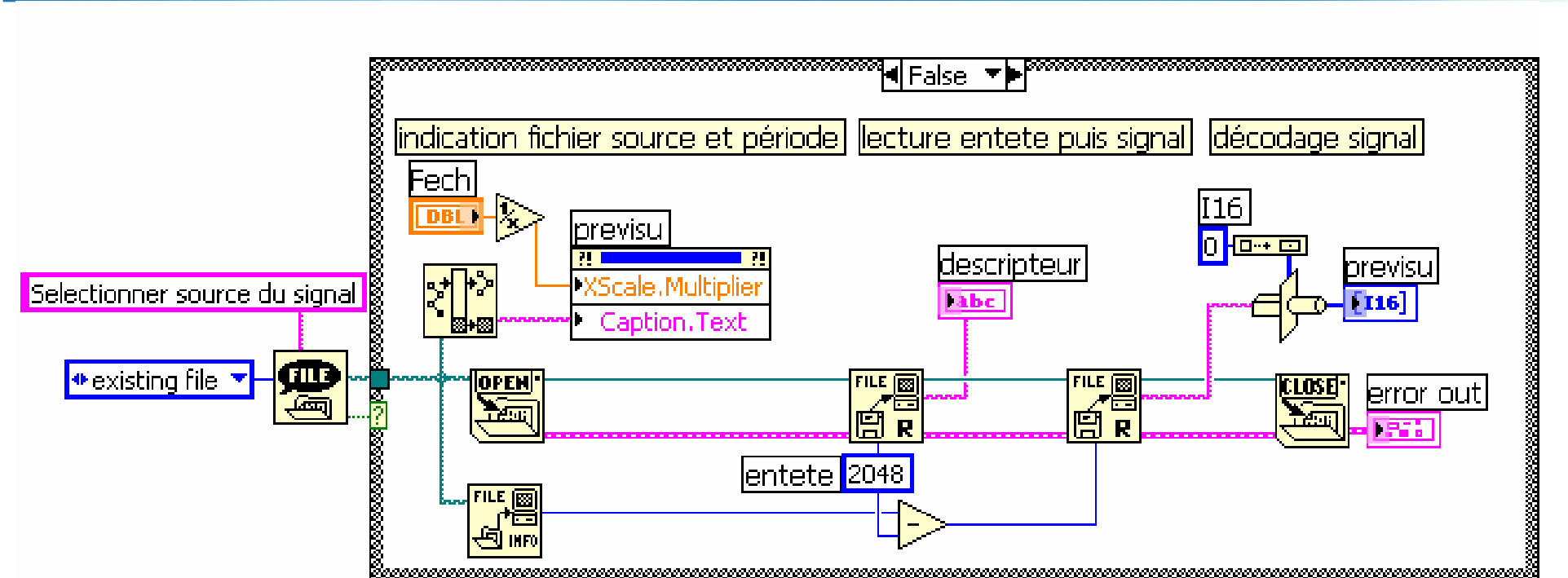
Visualiser un signal échantillonné

Le débutant



- 30 secondes de développement
- Un signal à 11KHz durant 8 secondes soit 88k points dans un graph de 1000 pixel de large.

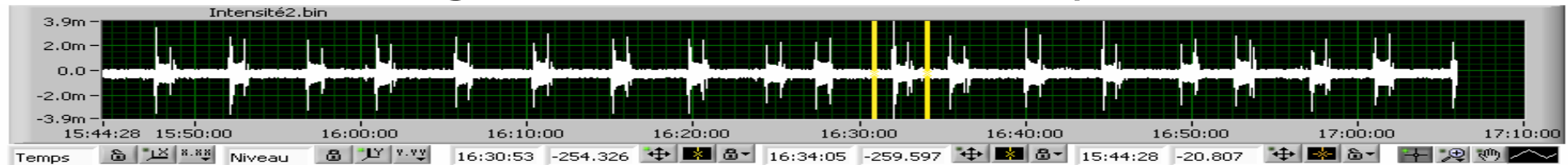
L'amateur



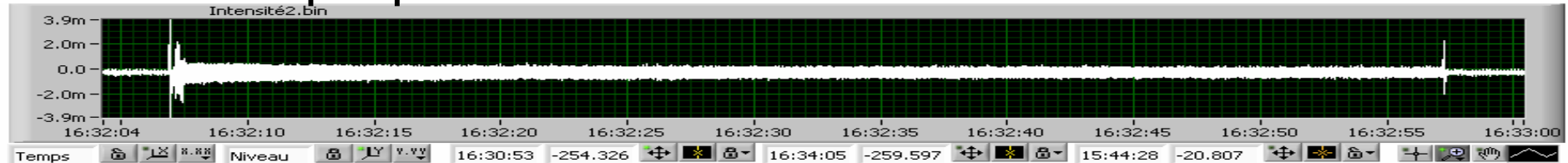
- 3 minutes de développement
- Un code légèrement enrichi

Et très vite les limites

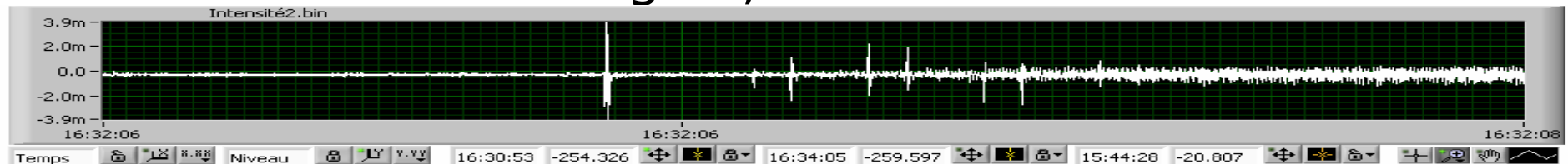
- 75min de signal à 3kHz >> 13.5M pts >> 27Mo



- Le Graph permet de zoomer sur 1 minute

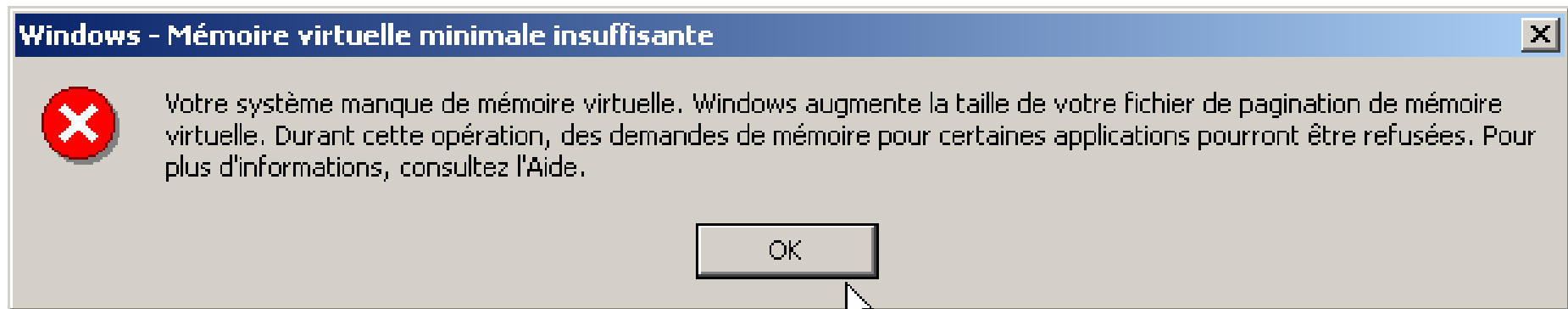


- Et 2 secondes de signal, l'observation est facile!



Pas de miracle...

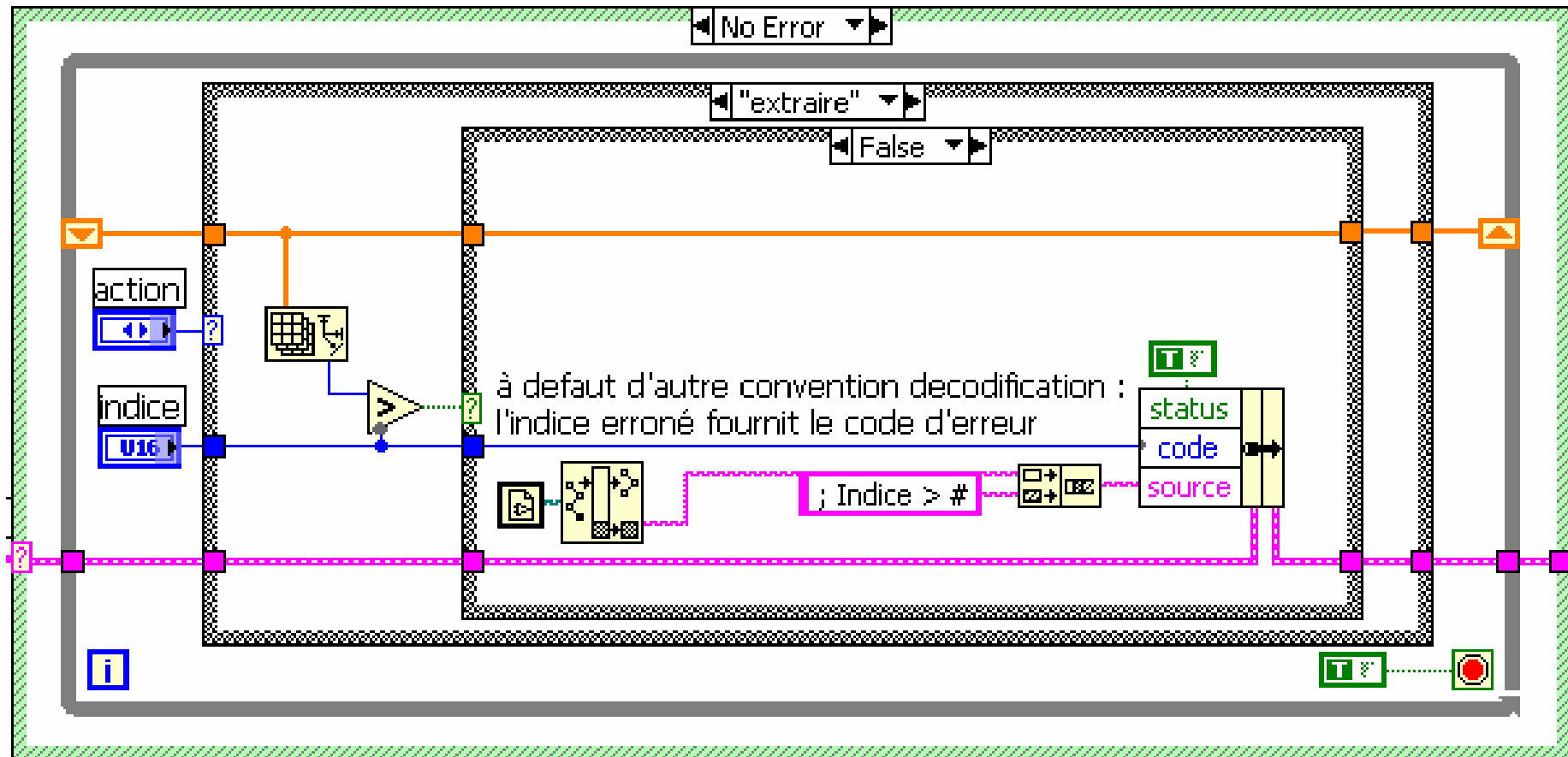
- Si le fichier est un peu gros...



Tester, Tester, TESTER!

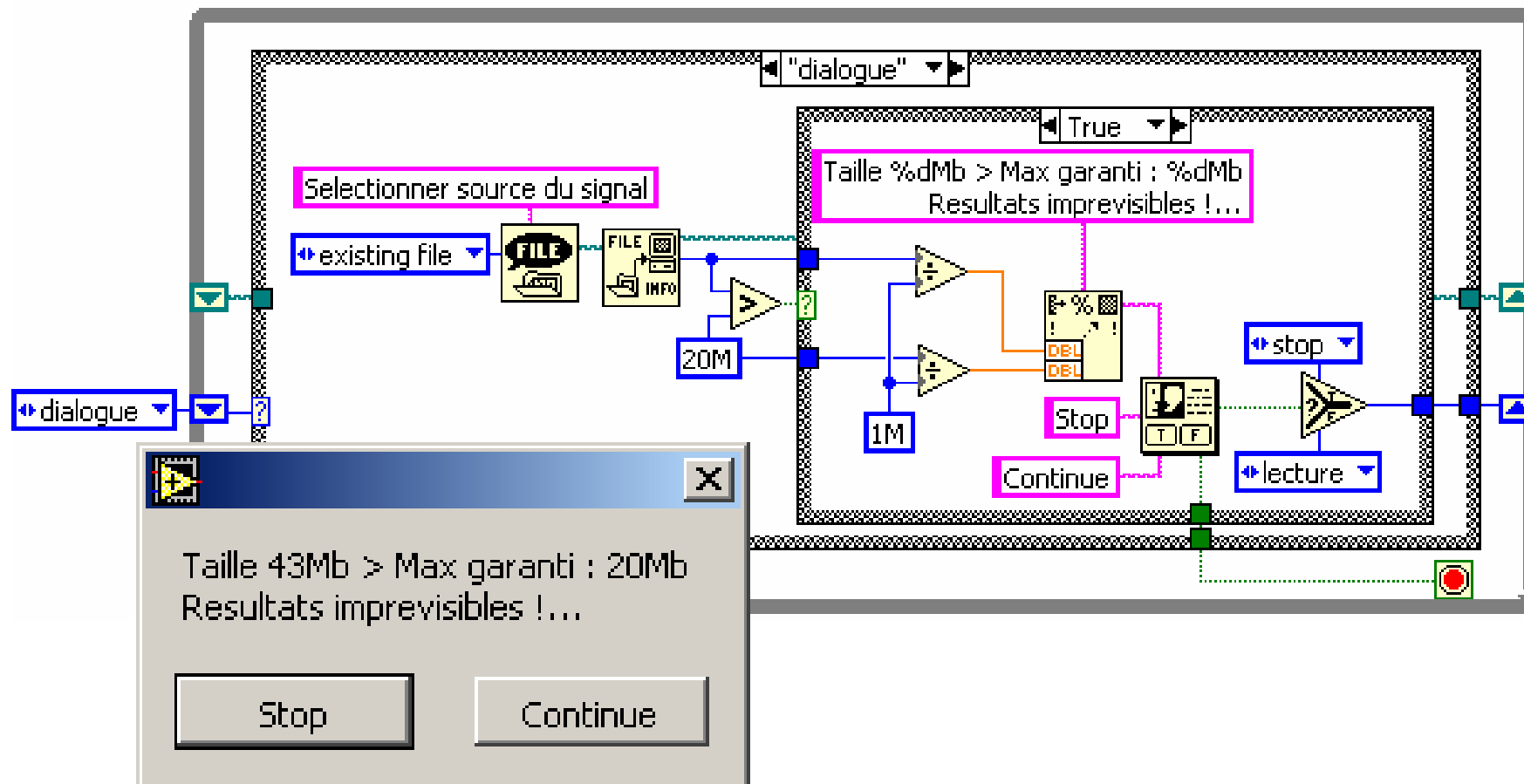
- Dès la conception, prévoir des tests :
 - Test dans les bornes du CDC, c'est le minimum
 - Test hors des bornes, pour voir où ça casse
 - Faire tester

Tester, Tester, TESTER!



Définir le champs d'application

- Afficher clairement ses limites

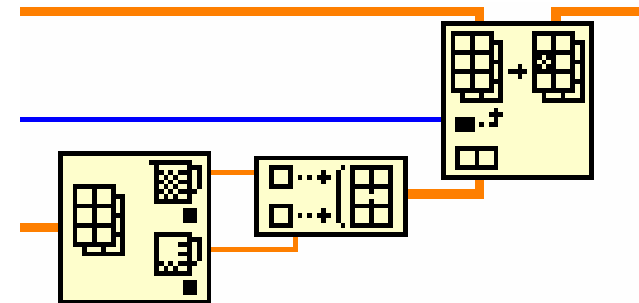


Aller plus loin

- Nouvelles exigences :
 - pas de limite de durée du signal à observer
 - fonctionnel quel que soit la RAM du PC
 - permet un zoom x100 sans perte d'info

Un peu de calcul...

- pix : largeur du graph en pixel
- nbE : nb de points
- Fe : Fréquence d'échantillonnage



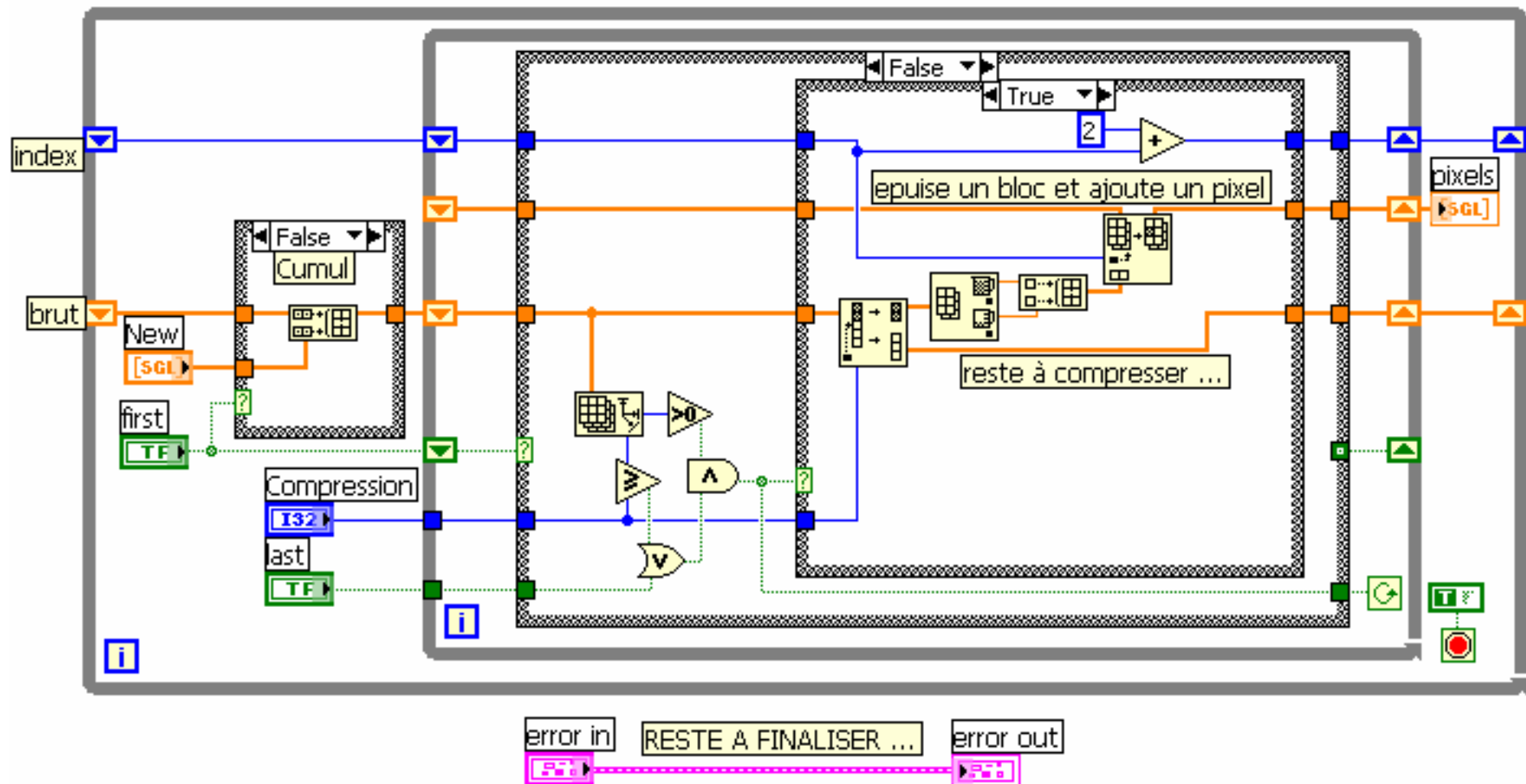
```
Kp=nbE/pix;  
Kp=ceil(Kp);  
nmM=nbE/Kp;  
nmM=ceil(nmM)*2;  
duree=nbE/Fe;  
dtmM=Kp/Fe/2;
```

Variables used in the code:

- pix
- nbE
- Fe
- duree
- nmM
- Kp
- dtmM

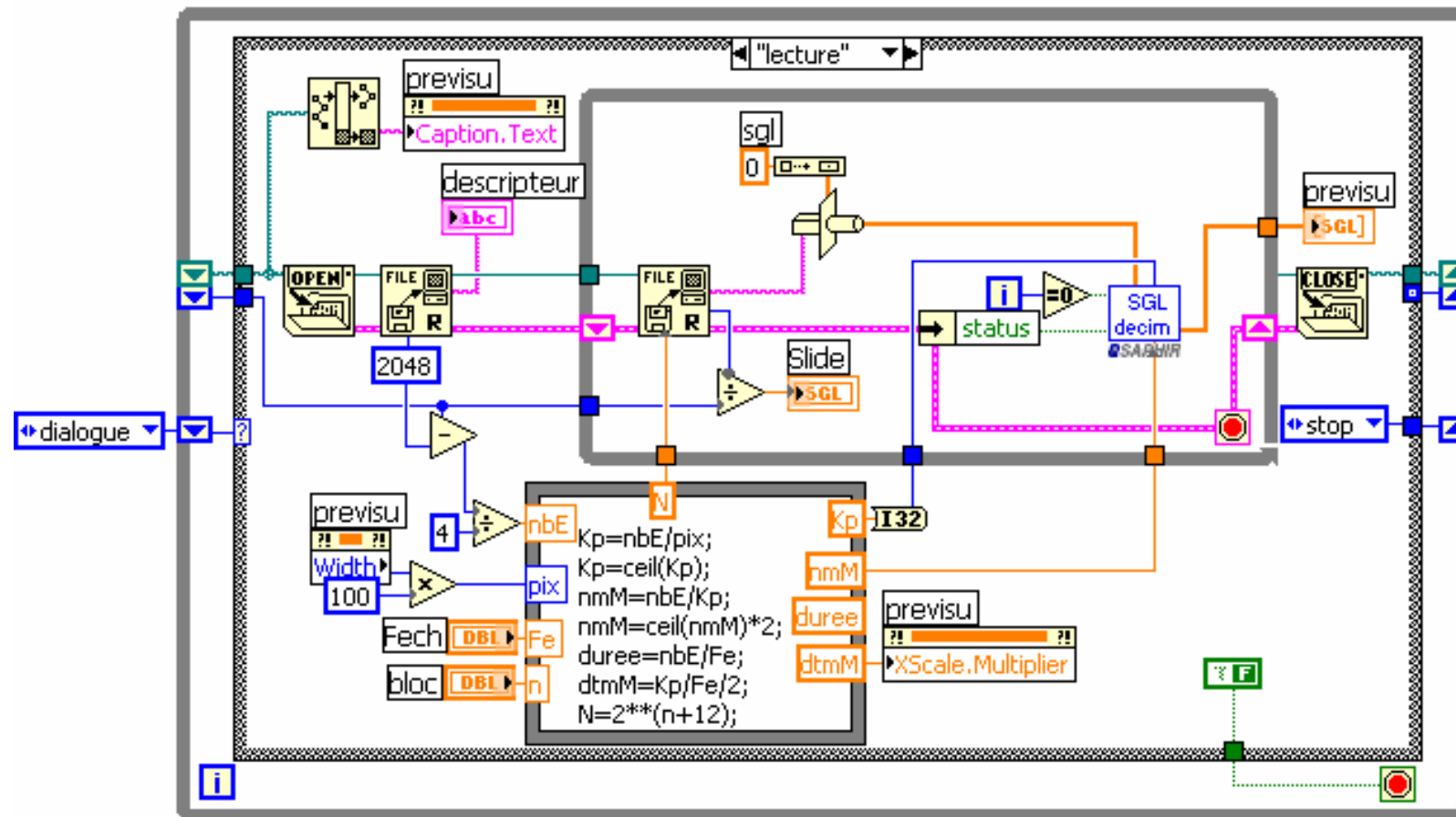
- Durée de l'enregistrement
- Nb de min et Max après réduction
- Rapport de réduction (ou taille d'un bloc à réduire à un couple min/max)
- Durée d'un bloc réduit

...Et de manipulation d'outil de base



Et l'intégration

- Pour arriver à un modeste proto, fait en 1h

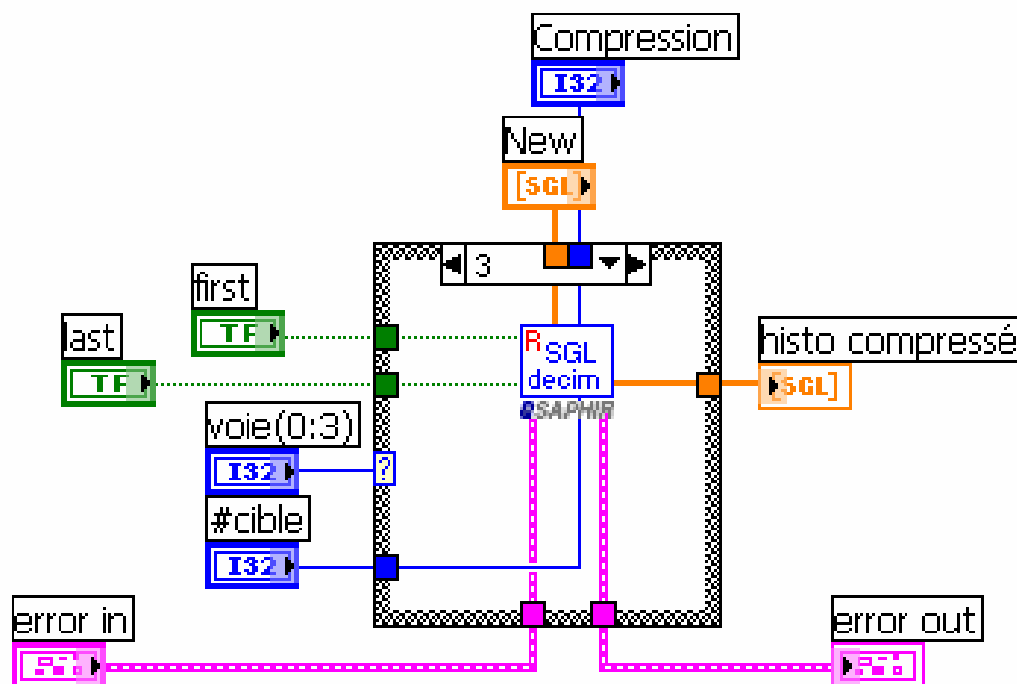


La multiplication des pains

- Dès la conception, se poser les bonnes questions :
 - champs d'application
 - durée de vie
 - Instantiable?
 - possibilité de ré-utilisation ailleurs

Une médiocre multiplication

- Un peu de paramétrage (vi ré-entrant)
- Des bornes serrées
- Et voilà 4 voies pour le prix d'une...



Pas de conclusion, une ouverture!