



# **EXPLOITATION DE LABVIEW DANS L'ENSEIGNEMENT : UN LANGAGE DE PROGRAMMATION MULTI-PLATEFORMES**

5 décembre 2008

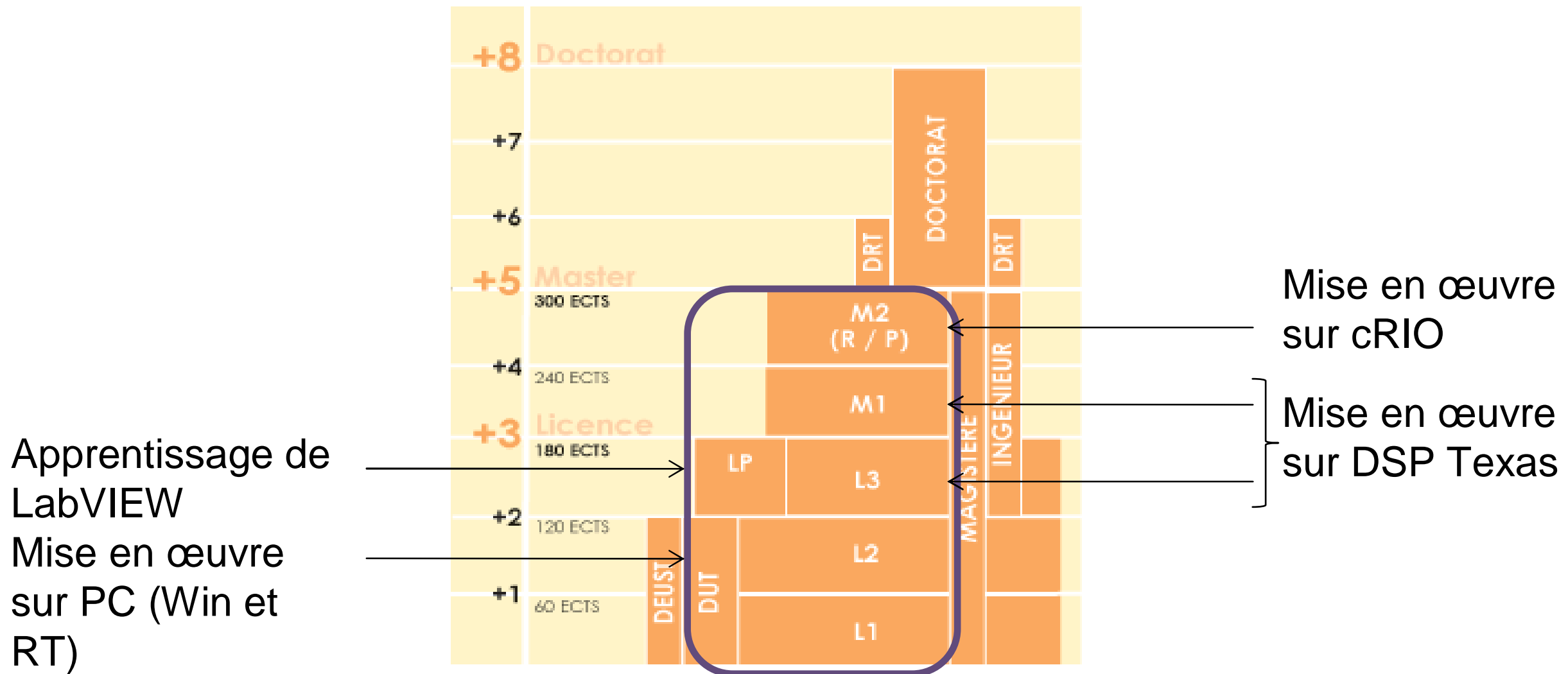
David FREY, Pierre-Armand DEGRYSE

Futurview 2008

- Introduction
- LabVIEW et les différents niveaux d'études
- Un exemple parmi d'autres
- Bilan de l'utilisation
- Et dans l'avenir ???

# Introduction

## Contexte : Formation en Génie Electrique du BAC à BA



Formations dispensées au département GEI11 de l'IUT de Grenoble  
et en formation EEATS de l'UFR de Physique.

# Introduction

Nos disciplines : l'informatique industrielle et l'automatisme

Cibles utilisées :

- Microcontrôleurs;
- Systèmes G64 et VME sous OS RT;
- DSP.

L'informatique industrielle vu par nos étudiants c'était quoi???

- Des langages de programmation avec pleins de termes anglais et un peu obscurs;
- Des suites de développements différentes pour chaque cible à laquelle il faut s'adapter.

En conclusion, il faudrait :

- Un langage simple d'approche pour nos étudiants;
- Un langage qui permettrait rapidement de développer des applications pour pouvoir les « accrocher » et pour avoir plus de temps sur les aspect essentiels;
- Une interface graphique et des fonctionnalités intéressantes;
- Un langage et des bibliothèques qui peuvent s'adapter à des cibles différentes.

## **Pourquoi LabVIEW? : Un logiciel de mise en œuvre « aisée**

- Interface d'utilisation « intuitive » pour les étudiants
- Nombreuses bibliothèques de fonctions adaptées à notre utilisation
  - Gestion de fichiers
  - Acquisition de données
  - Communications (Série, USB, Ethernet, Mail, ...)
  - ...
- Nombreux exemples disponibles
- Possibilité de développer rapidement des applications « complexes »
- Nécessité de mener obligatoirement une réflexion et une structuration préalable à toute programmation

## Pourquoi LabVIEW? : Un langage multi-cible

- Nombreux supports de développements
  - Plateformes PC
  - Modules cRIO
  - DSP Texas Instruments
  - PDA
  - processeurs ARM...
- Possibilité de programmer
  - Sur PC Windows
  - Sur cibles « classiques » (DSP)
  - Sur cibles RT (PC, cRIO, ...)
  - Sur FPGA

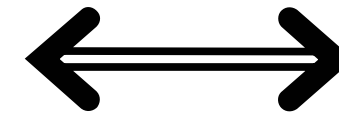
# Equipements et objectifs pédagogiques (DUT)

- Apprendre un langage de programmation graphique
- Savoir piloter et communiquer entre PCs et avec des appa



M6221

Acquisition  
de données



Communication : série, USB, Ethernet, ...



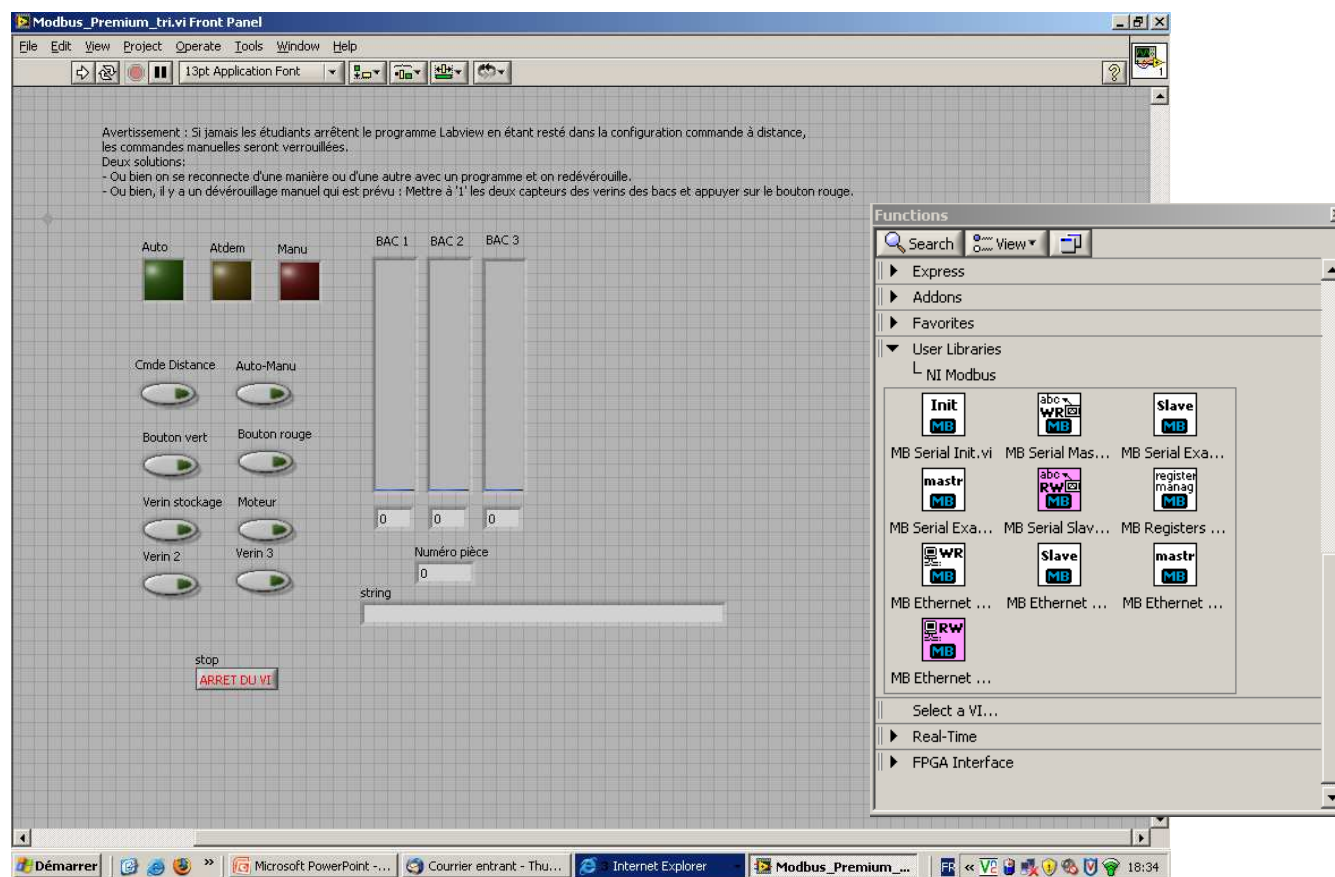


# Equipements et objectifs pédagogiques (LP DEAI)

- Interfaçage
- Supervision de systèmes industriels



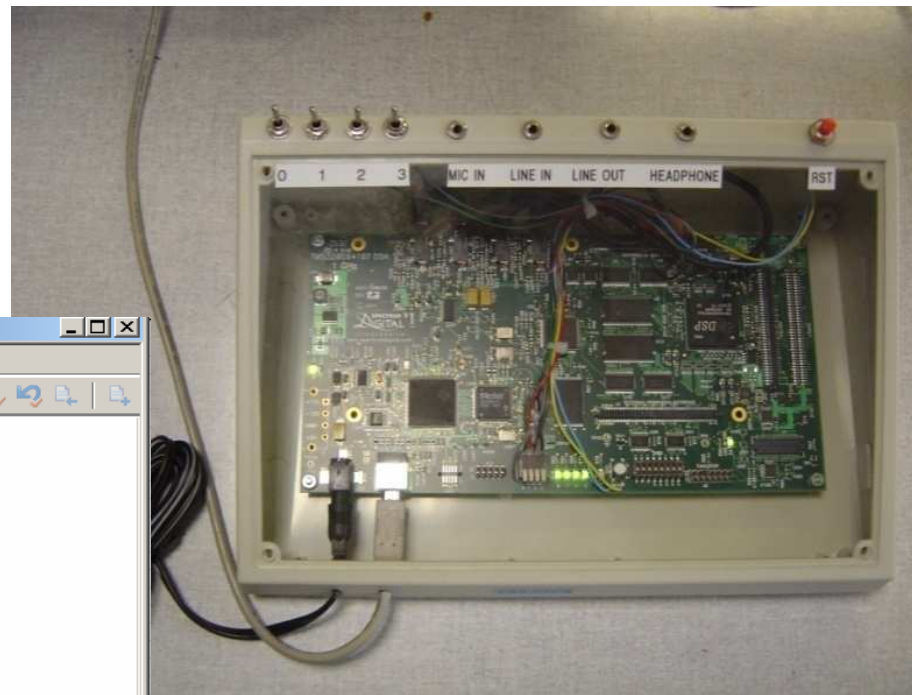
Modbus Ethernet



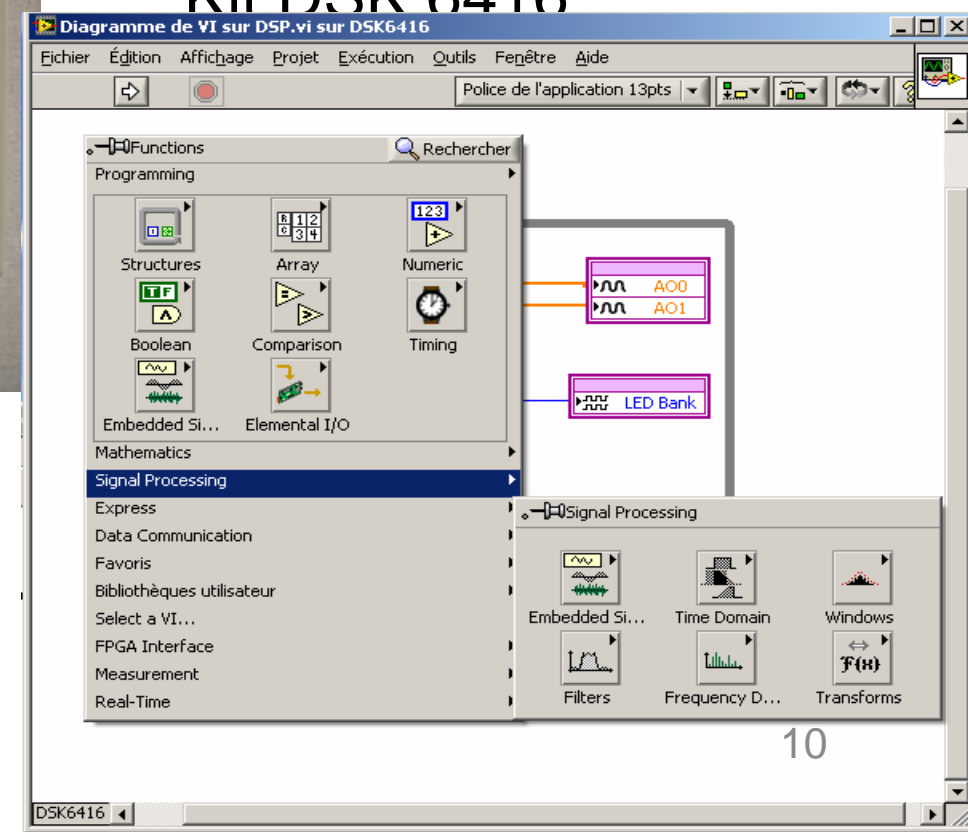
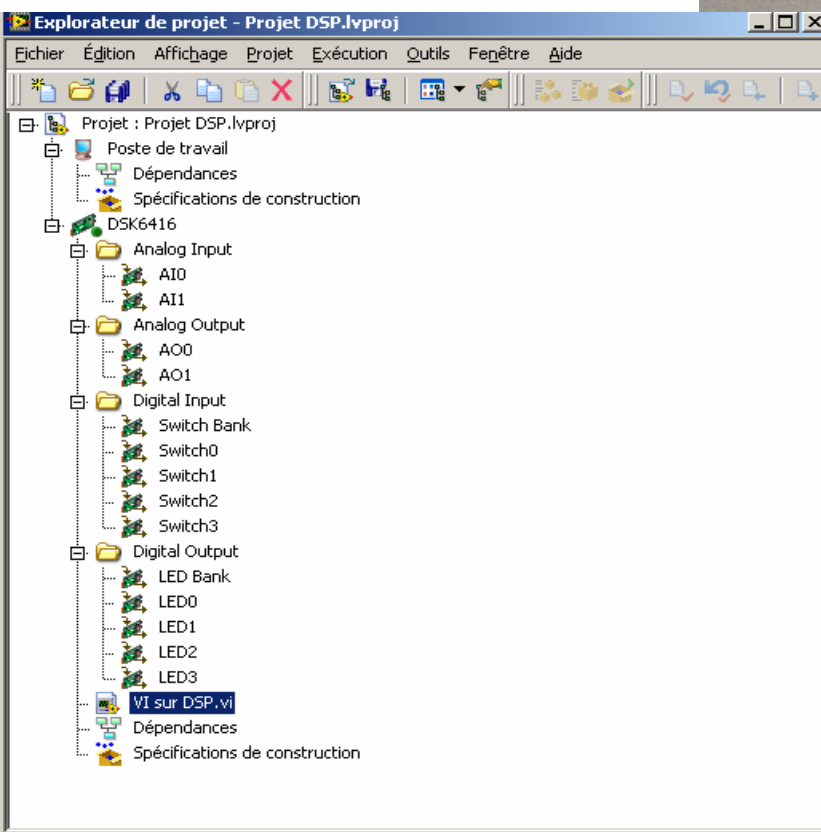
# Equipements et objectifs pédagogiques (L3 - M1)

- Avoir une approche rapide des applications sur DSP
- Servir de « motivant » pour aller creuser le code 'C' sur ce type de cibles.

Exemple: Acquisition et traitement simple de son.

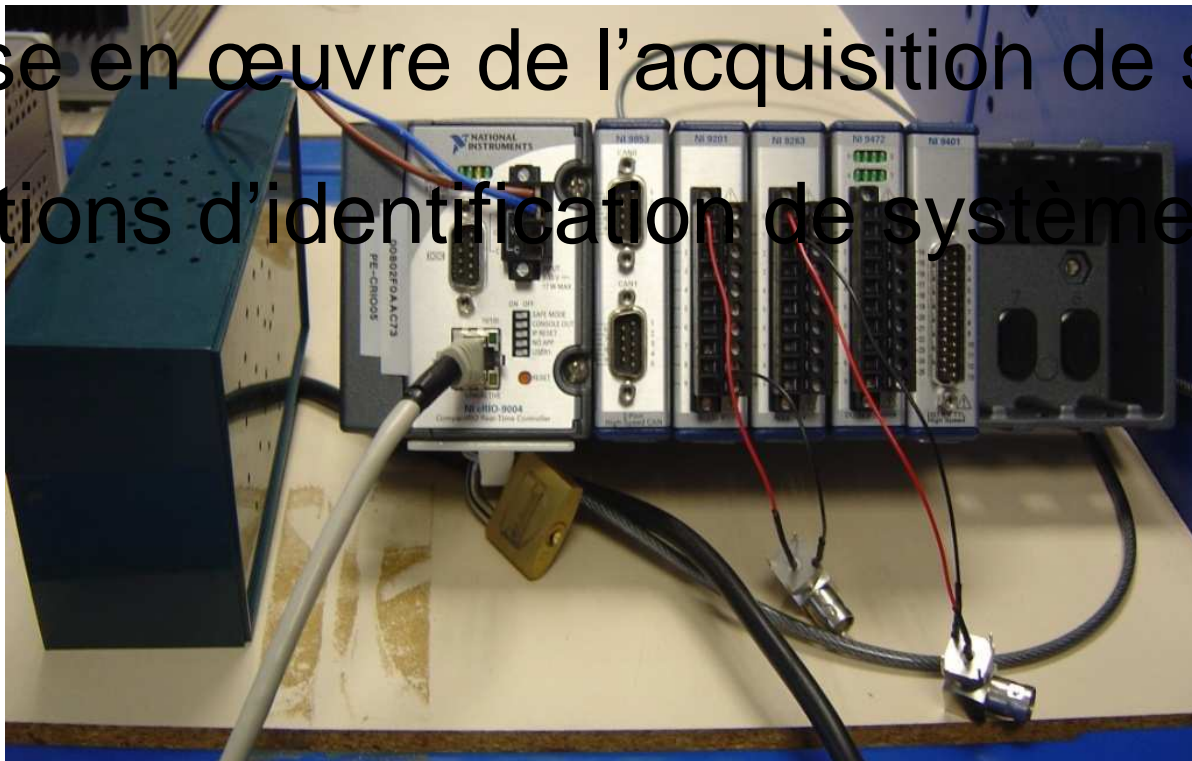


DSP Texas  
Instrument TMS320  
6416  
Kit DSK 6416



## PSPI : Pilotage et Surveillance des Procédés Industriels

- Support de cours aux enseignements de systèmes embarqués;
- Développer une architecture basée sur la mise en œuvre de processeurs sous OS RT associés à des FPGA;
- Mise en œuvre de l'acquisition de signaux et des fonctions d'identification de systèmes.

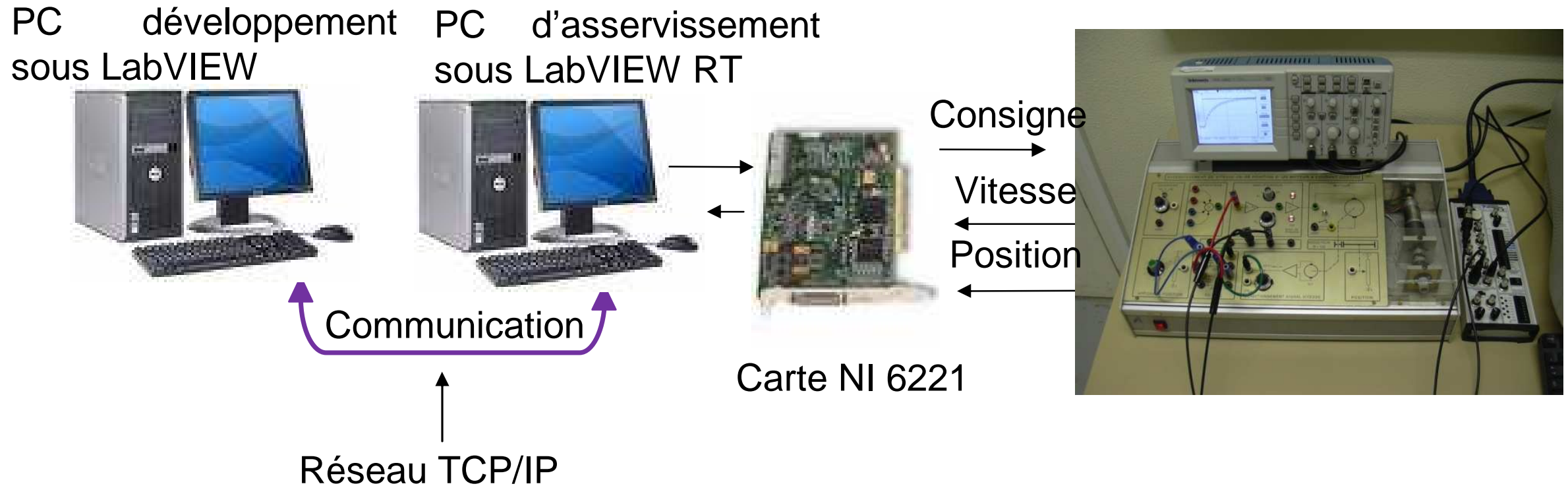


NI cRIO 9004

### TP d'automatisme échantillonnée au niveau DUT 2<sup>ème</sup> année

- Permettre aux étudiants de mettre en œuvre les connaissances acquises en cours LabVIEW et d'automatique continue et échantillonnée.
- Pouvoir avoir accès aux principales grandeurs de l'asservissement pour identifier les contraintes / limites qui peuvent apparaître et pouvoir les stocker dans un fichier au format Excel pour traitement.
- Pouvoir constater facilement l'influence de la fréquence d'échantillonnage sur la stabilité.
- Pouvoir se concentrer sur l'aspect régulation numérique en s'affranchissant des aspects plus matériels et programmation.

## Architecture du système



2 machines :

- Un PC de développement sous Windows XP (développement / modification du code, test local et stockage des données)
- Un PC cible sous LabVIEW RT (téléchargement du code et pilotage de l'application)



## Pourquoi cette architecture?

Mise en œuvre analogue à ce que les étudiants voient par ailleurs (exemple : les API)

- Développement du code sur une machine
- Téléchargement et exécution du code sur une cible

## Pourquoi LabVIEW RT ?

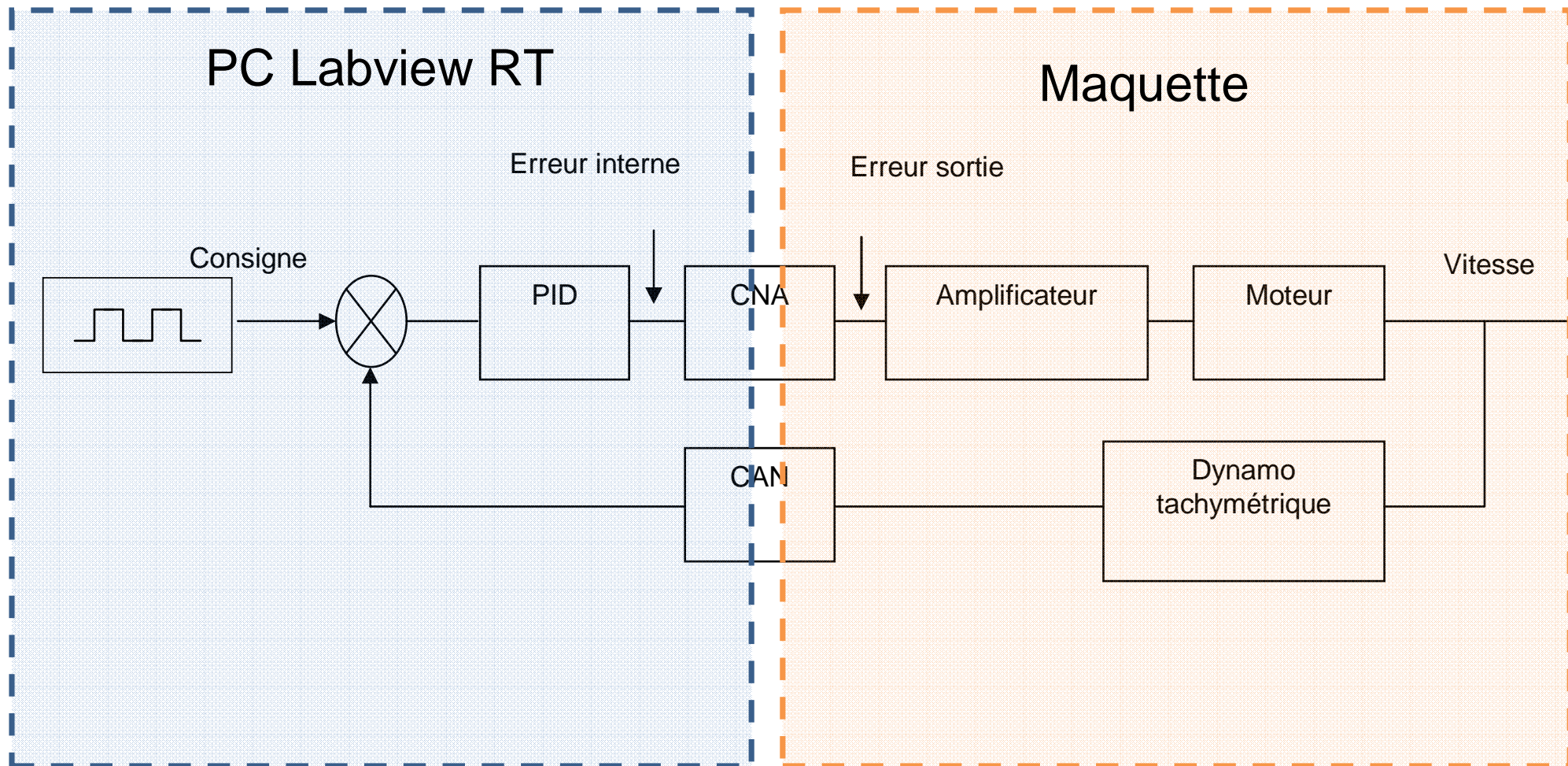
	Programme « simple »	Programme « complet »
Labview Windows XP	12,5 kHz	4,25 kHz
Labview RT	18,5 kHz	6,75 kHz
Différence	+ 48%	+ 58%

Même programme dans les deux cas : Seule la cible change.

- Globalement 50% de performance en plus
- Performances de la machine sous XP fortement dépendante des applications en tâche de fond et de la mémoire RAM.

# Un exemple parmi d'autres

## Schéma bloc de l'architecture



Principaux paramètres réglables :

- Signaux créneaux de consigne (amplitude, offset, fréquence)
- Paramètres du correcteur PID
- Fréquence d'échantillonnage

# Un exemple parmi d'autres

Face-avant de PID Control-Advance.vi sur Proj1.lvproj/IUTGE1-T40-Vx

Fichier Édition Affichage Projet Exécution Outils Fenêtre Aide

Police de l'application 13pts

## PRINCIPE DE FONCTIONNEMENT

Ce programme permet de piloter la maquette moteur à courant continu. L'échelon de consigne nécessaire pour l'identification ou pour voir le comportement du système est généré de manière interne par une fonction Labview. La mesure de la grandeur est configurable par le menu Entrée analogique. La sortie du calculateur assurant le pilotage du système est par contre fixe sur ao0. Vous pourrez obtenir une image de l'échelon de consigne sur la sortie ao1 du bornier.

Vous aurez le choix entre un correcteur de type P et un correcteur PID. Un bouton marche-arrêt permettra d'arrêter le moteur. Pensez à l'arrêter à chaque fois que vous stoppez le programme. Sans cela, le moteur continuera de tourner tant qu'il sera alimenté.

La plupart des paramètres sont réglables

- Fréquence d'échantillonnage (ne peut être changée que programme arrêté!!!)
- Gain des correcteurs P et PID
- Caractéristiques de l'échelon (gain, amplitude, période)
- Caractéristiques des signaux en entrée et en sortie.

(Par défauts ceux-ci sont limités à +/- 10V)

**PC sous Labview RT**

**Oscilloscope**

**Maquette**

**Bornier BNC 2110**

### Configuration

#### Parametrage des canaux d'acquisition

Entrée analogique:  $\frac{1}{\%}$  Dev1/ai0

Sortie analogique:  $\frac{1}{\%}$  Dev1/ao0

Valeur d'entrée mini: -10,00

Valeur de sortie mini: -10,00

Valeur d'entrée maxi: 10,00

Valeur de sortie maxi: 10,00

#### Parametrage de l'échantillonnage

Fréquence d'échantillonnage: 1000,00 ( $\leq 15000$  kHz)

#### Réglage correcteur

Gain Proportionnel: 1

#### Réglage de l'échelon

Amplitude: 0,2

Valeur moyenne: 0

Période du signal (s): 0

#### Réglage correcteur PID

gain proportionnel (K): 1,000

integrateur (Ti, min): 0,01000

derivativé (Td, min): 0,000

### Contrôle du programme et de la maquette

Arrêt <---> Marche

STOP

Echantillon raté

Correcteur P <-----> PID

Commande onglet

Proj1.lvproj/IUTGE1-T40-Vx

Démarrer Explorateur de projet - P... Face-avant de PID Co... Courrier :: Boîte de réce...

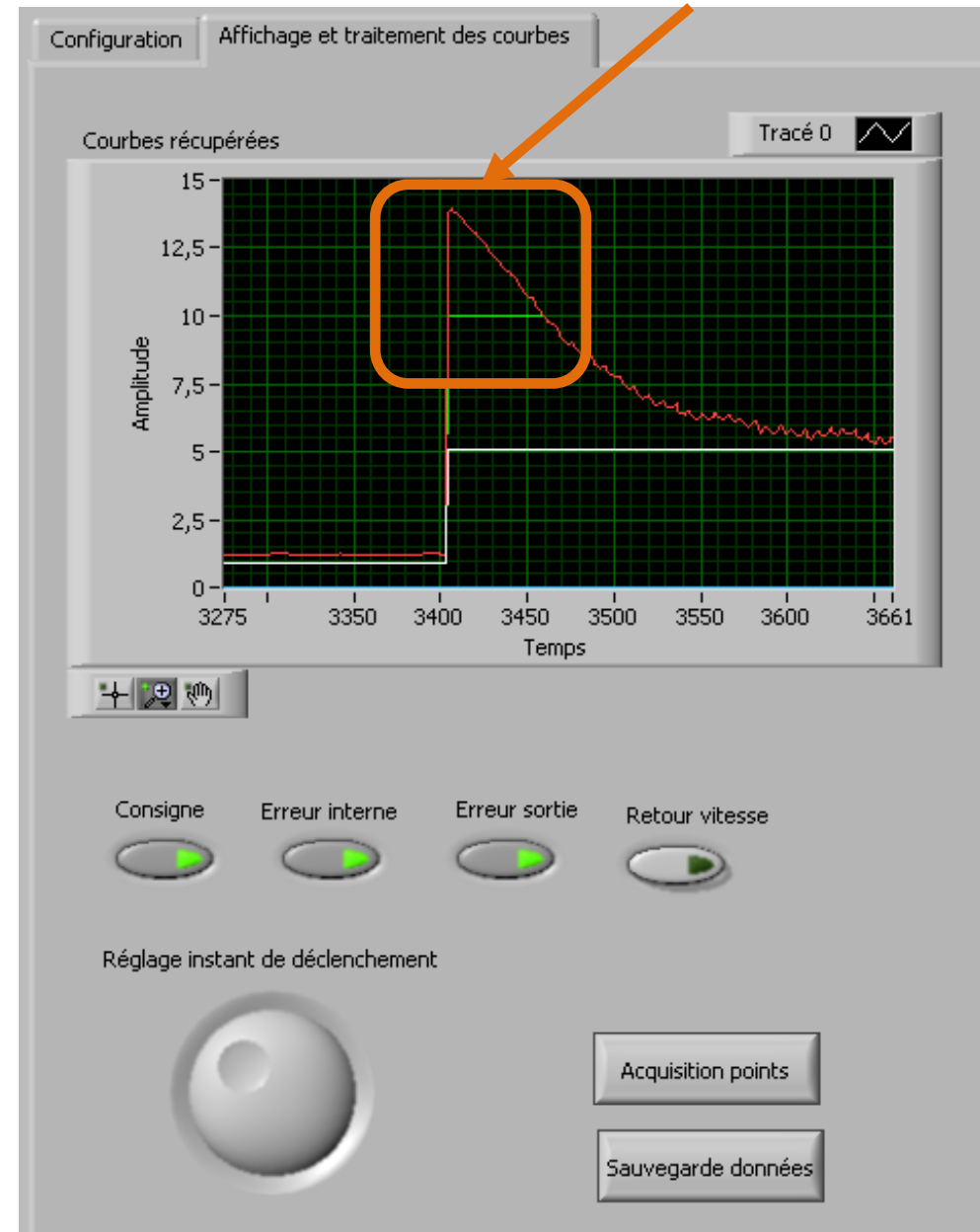
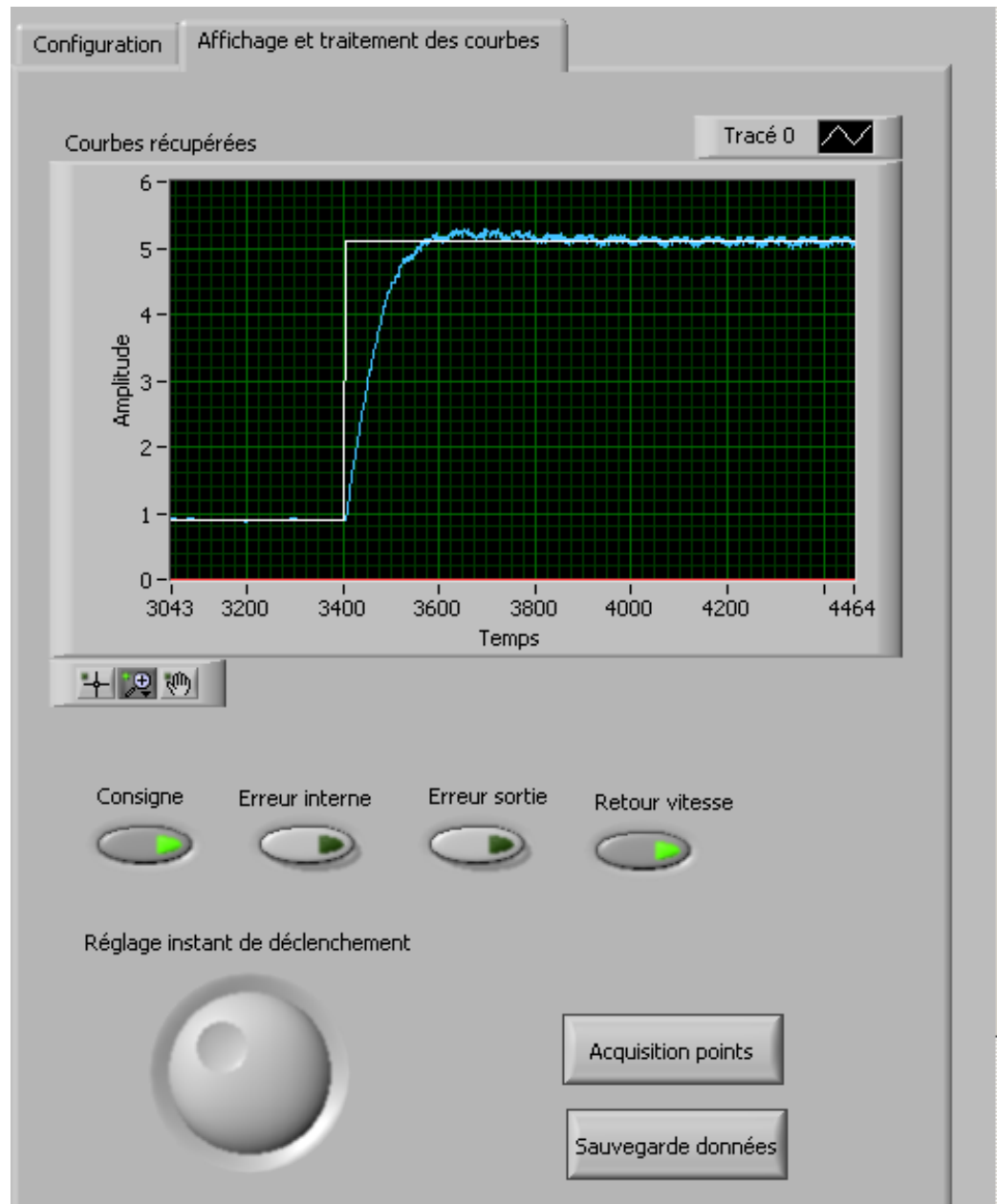
16 09:10



# Un exemple parmi d'autres

## Principaux signaux retournés

Effet de la saturation du CNA



# Un petit bilan de LabVIEW...

Un environnement graphique convivial bien adapté à nos étudiants (notamment au niveau DUT).

Un langage de programmation multi-cible qui permet de s'affranchir plus aisément des spécificités matérielles et permettre une première approche simplifiée.

Des bibliothèques de fonctions et des toolkits qui couvrent la majeure partie des besoins de nos applications et adaptées aux différentes cibles.

Un formidable outil pour inciter les étudiants à développer des programmes sur PC dans le cadre de leurs projets:

- GPS\_VIEW (Interfaçage d'un PC avec un récepteur GPS)
- Pilotage / Configuration de variateurs de vitesse Leroy

Somer par MODBUS RTU.

- ...

# Et pour l'avenir ???

Actuellement nos cibles sont :

- PC XP et LabVIEW RT
- DSP Texas
- cRIO
- PDA

Dans l'avenir :

- Microcontrôleurs ARM
- LEGO « Mindstorm »
- Mise en œuvre conjointe de MATLAB (SCILAB?) avec LabVIEW...

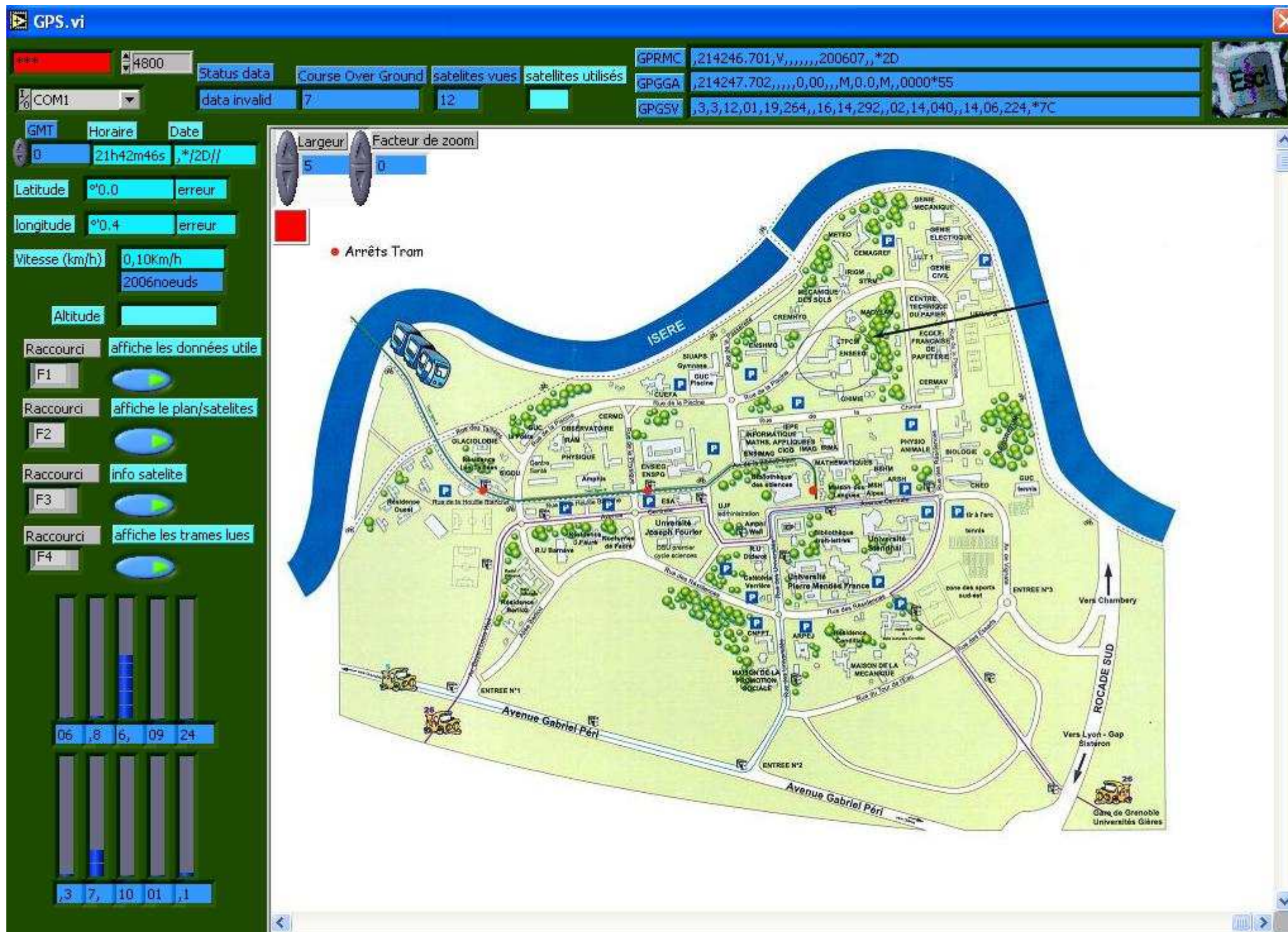


**MERCI DE VOTRE  
ATTENTION**





# GPS\_VIEW



Circuit SIRF III