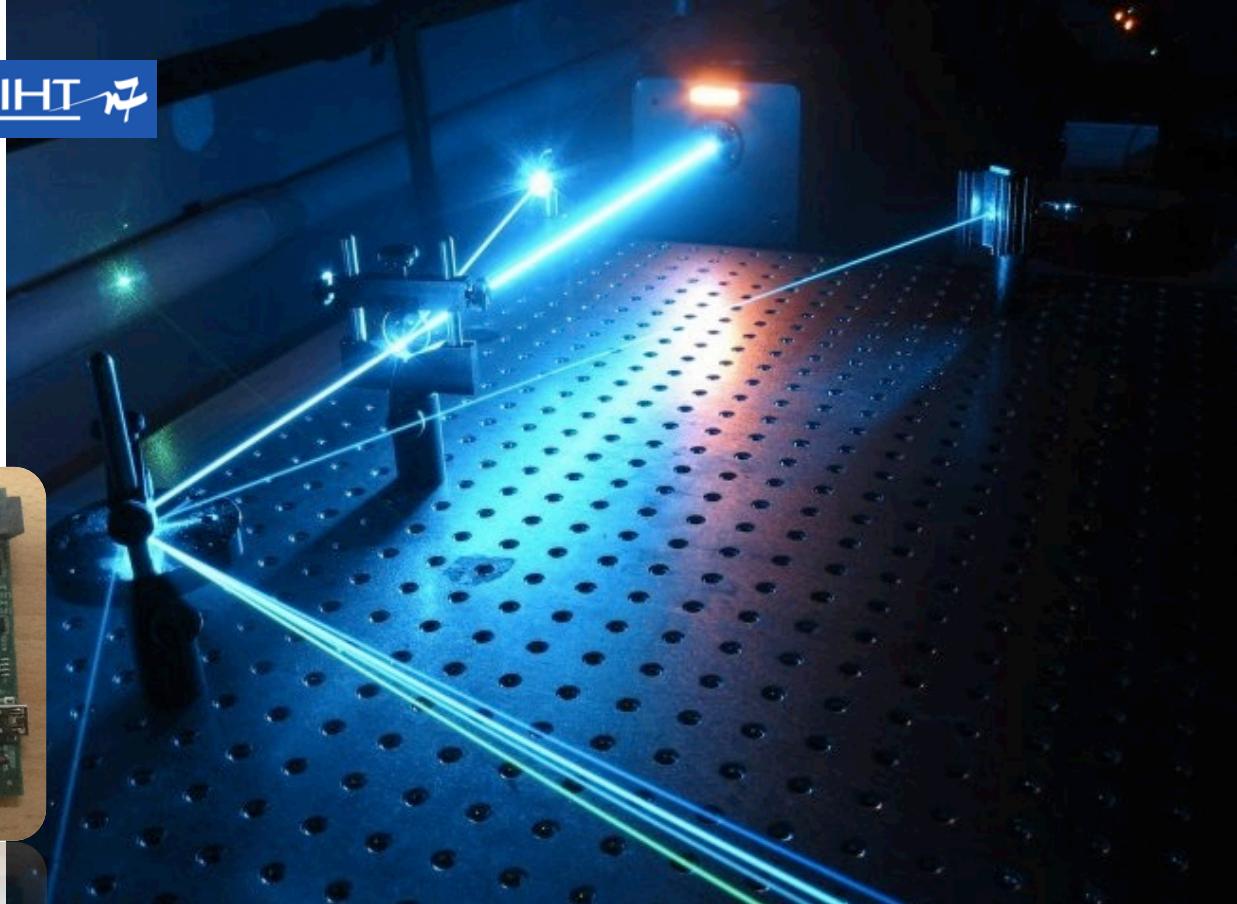




Optoélectronique pour les Systèmes
Embarqués -OSE



*Conception d'un démonstrateur électronique versatile
pour la mesure de déplacements par réinjection optique
dans une diode laser, avec contrôle du faisceau émis*

Master MIGESE-CAS

Contexte

- **LAAS-OSE**
 - Optoélectronique pour les Systèmes Embarqués
- **Objectif**
 - Prototype embarqué en temps réel
 - Balayage de faisceau

Le programme

- Introduction
 - *Self-Mixing (SM)*
 - *Algorithme Détection de Franges*
- Développement
 - Outils logiciels
 - Outils matériel
 - Implémentation des fonctions
- Expériences
 - ✓ Mise en œuvre
- Scanning de surface
- Conclusion

Introduction

Développement

Expériences

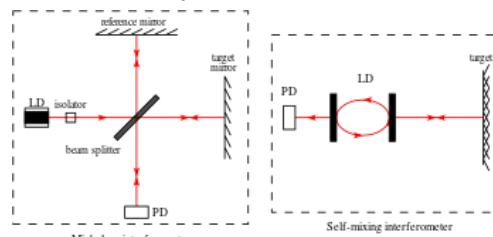
Scanning de surface

Conclusion

Introduction

LAAS-OSE

Lieu de Stage...?



Introduction

Développement

Expériences

Scanning de surface

Conclusion

Le programme

- Introduction
 - *Self-Mixing (SM)*
 - *Algorithme Détection de Franges*
- Développement
 - Outils logiciels
 - Outils matériel
 - Implémentation des fonctions
- Expériences
 - ✓ Mise en œuvre
- Scanning de surface
- Conclusion

Introduction

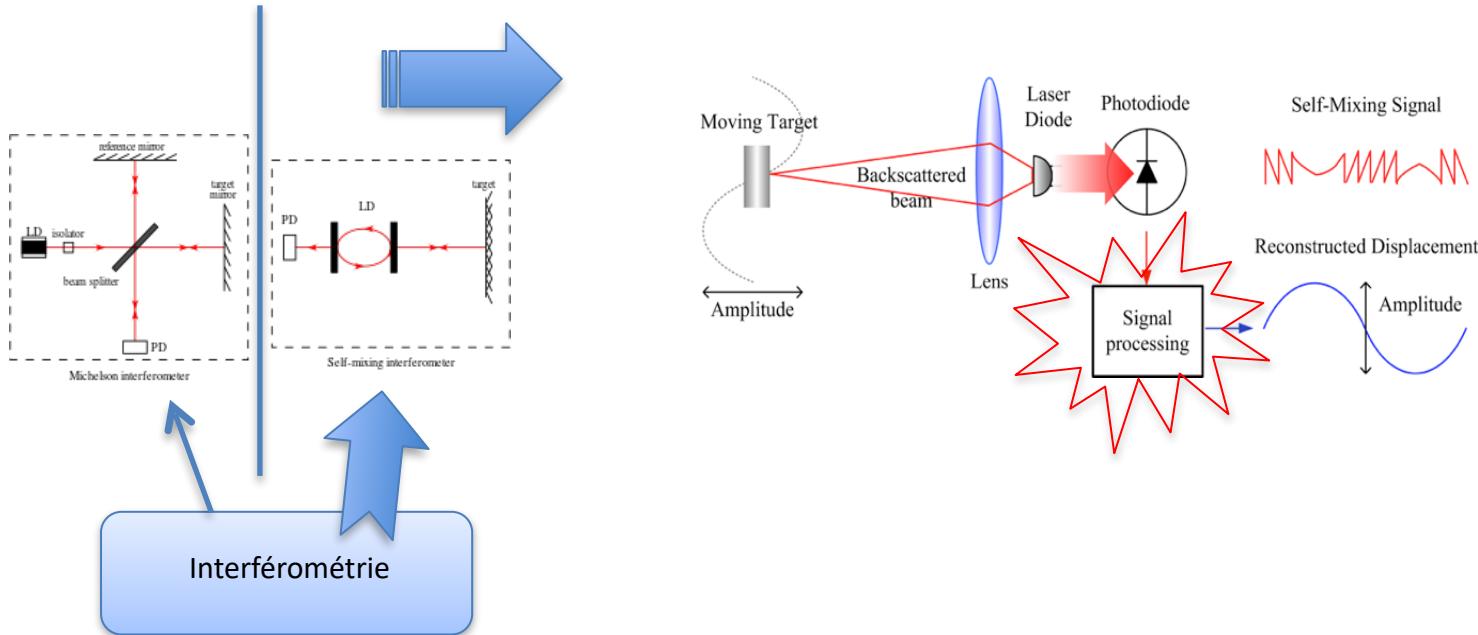
Développement

Expériences

Scanning de surface

Conclusion

Self-Mixing (SM)



Introduction

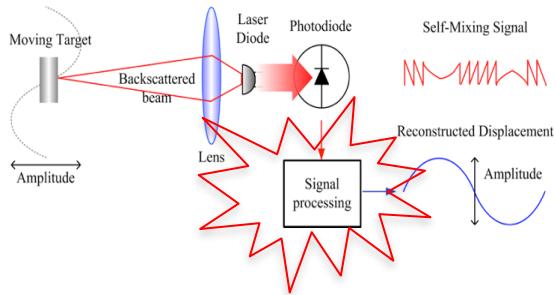
Développement

Expériences

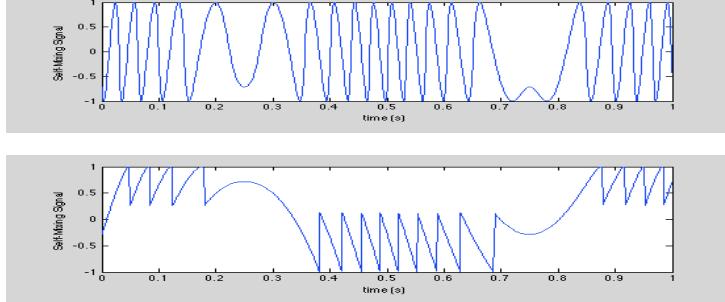
Scanning de surface

Conclusion

Self-Mixing (SM)



Paramètre C



Représentation de la signal SM

Introduction

Développement

Expériences

Scanning de surface

Conclusion

Le programme

- Introduction
 - *Self-Mixing (SM)*
 - *Algorithme Détection de Franges*
- Développement
 - Outils logiciels
 - Outils matériel
 - Implémentation des fonctions
- Expériences
 - ✓ Mise en œuvre
- Scanning de surface
- Conclusion

Introduction

Développement

Expériences

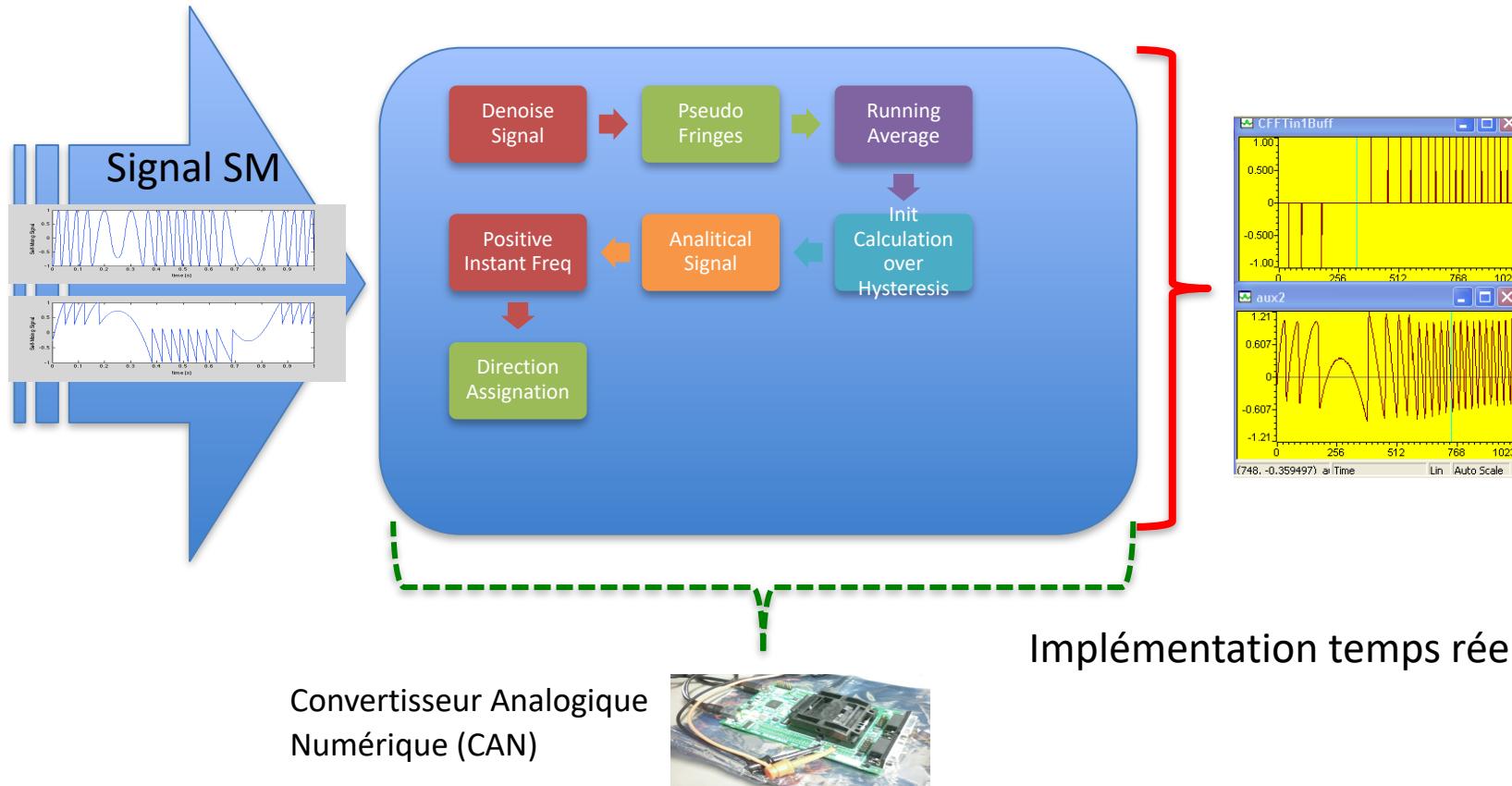
Scanning de surface

Conclusion

Algorithme

Thèse de Antonio LUNA

Algorithme de détection de franges



Le programme

- Introduction
 - *Self-Mixing (SM)*
 - *Algorithme Détection de Franges*
- Développement
 - Outils logiciels
 - Outils matériel
 - Implémentation des fonctions
- Expériences
 - ✓ Mise en œuvre
- Scanning de surface
- Conclusion

Introduction

Développement

Expériences

Scanning de surface

Conclusion

Développement

Introduction

Développement

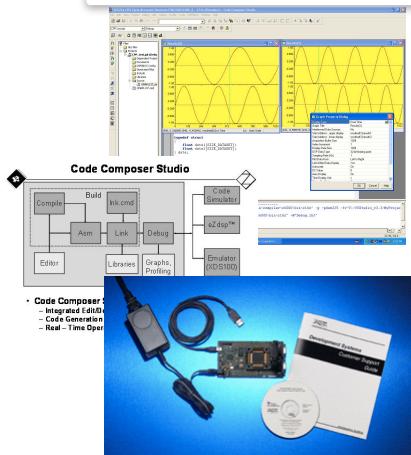
Expériences

Scanning de surface

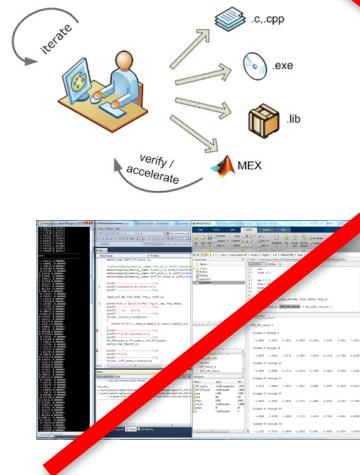
Conclusion

Software utilisées

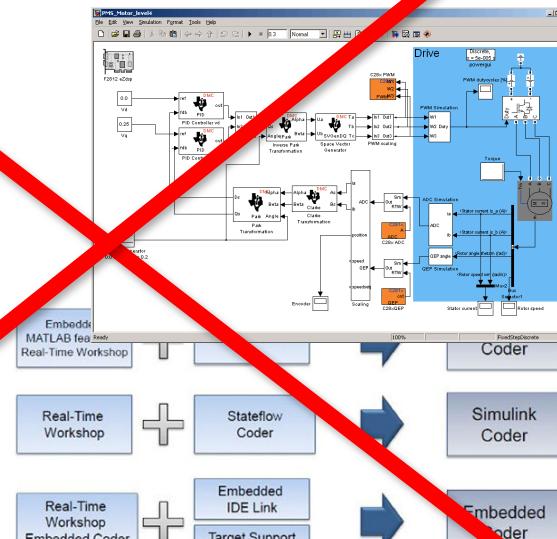
Code Composer
Studio v3.3 (CCS)



MATLAB Coder



SIMULINK Coder



Développement

Introduction

Développement

Expériences

Scanning de surface

Conclusion

Code Composer Studio v3.3

- ✓ TI F28xx Code Composer Studio Integrated Development Environment
- ✓ Texas Instruments' F28335 header files and example software



Software utilisées

MATLAB Coder

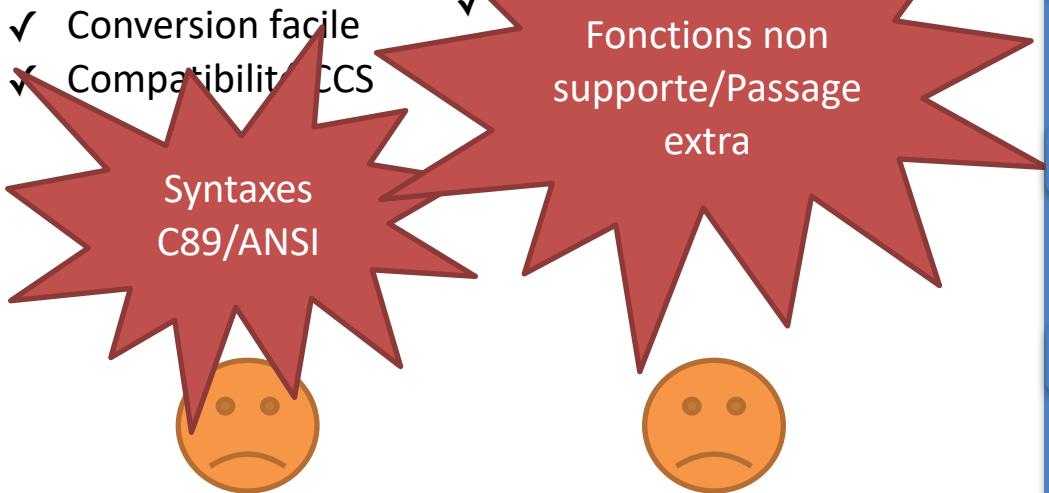
- ✓ Conversion facile
- ✓ Compatibilité CCS

Syntaxes C89/ANSI

SIMULINK Coder

Programmation par blocks

Fonctions non supportées/Passage extra



Le programme

- Introduction
 - *Self-Mixing (SM)*
 - *Algorithme Détection de Franges*
- Développement
 - Outils logiciels
 - Outils matériel
 - Implémentation des fonctions
- Expériences
 - ✓ Mise en œuvre
- Scanning de surface
- Conclusion

Introduction

Développement

Expériences

Scanning de surface

Conclusion

Développement

Introduction

Développement

Expériences

Scanning de surface

Conclusion

Hardware utilisé

TMS320F28335 eZdsp

- ✓ Vitesse de fonctionnement
150 Mhz
- ✓ Virgule flottante de 32 bits
- ✓ CAN de 12 bit avec 16 entrées
- ✓ Programmation en C



Le programme

- Introduction
 - *Self-Mixing (SM)*
 - *Algorithme Détection de Franges*
- Développement
 - Outils logiciels
 - Outils matériel
 - Implémentation des fonctions
- Expériences
 - ✓ Mise en œuvre
- Scanning de surface
- Conclusion

Introduction

Développement

Expériences

Scanning de surface

Conclusion

Implémentation des fonctions

Introduction

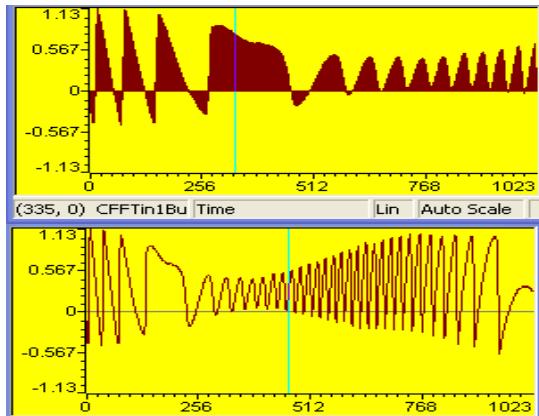
Développement

Expériences

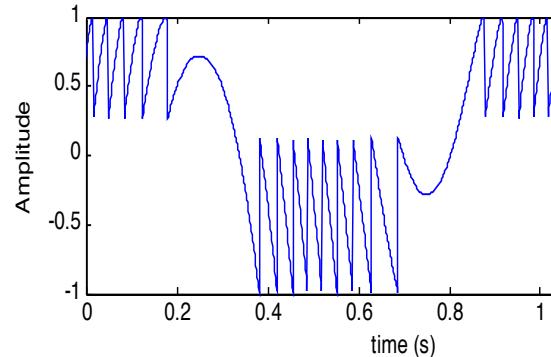
Scanning de surface

Conclusion

En pratique
CCS



En théorie
MATLAB

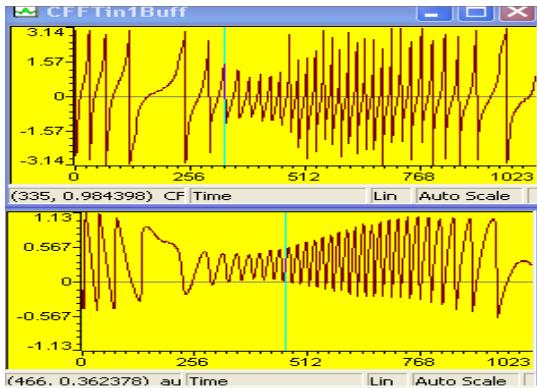


- CAN de 12 bits
- Buffer de 2048 éléments représentation complexe

- Vecteur de 1024 éléments

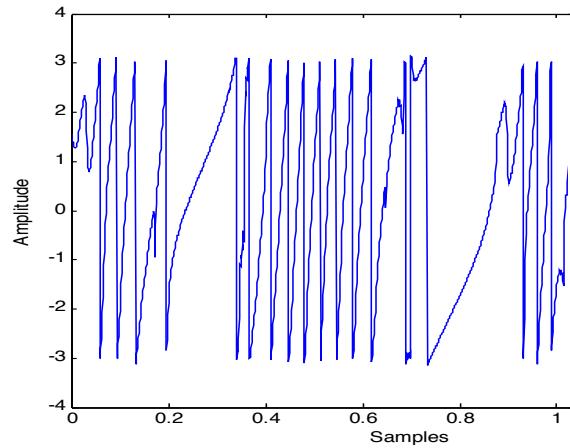
Implémentation des fonctions

En pratique
CCS



- Transformée de Hilbert
- Vecteur de 1024 éléments réels

En théorie
MATLAB



- Transformée de Hilbert
- Eléments réels

Introduction

Développement

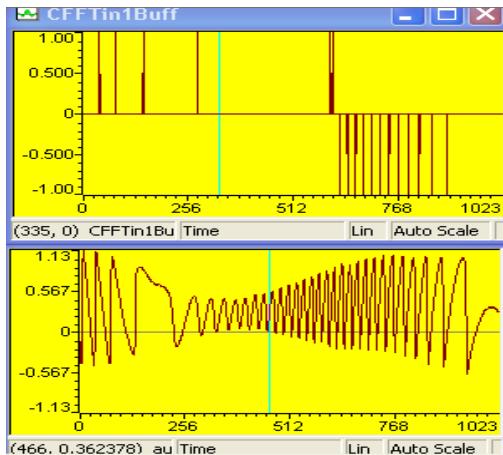
Expériences

Scanning de surface

Conclusion

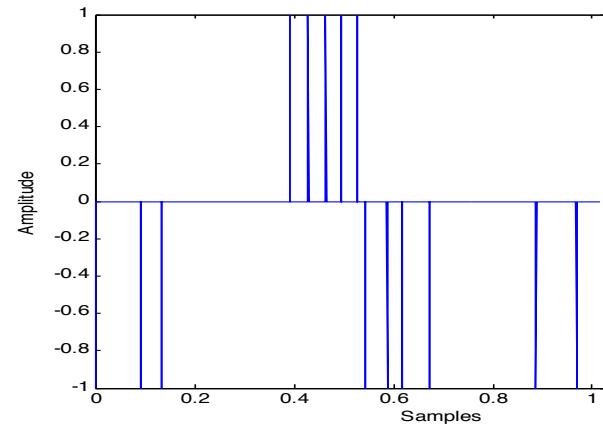
Implémentation des fonctions

En pratique
CCS



- Estimation de franges
- Buffer de 1024 éléments représentation réel

En théorie
MATLAB



- Vecteur de 1024 éléments

Introduction

Développement

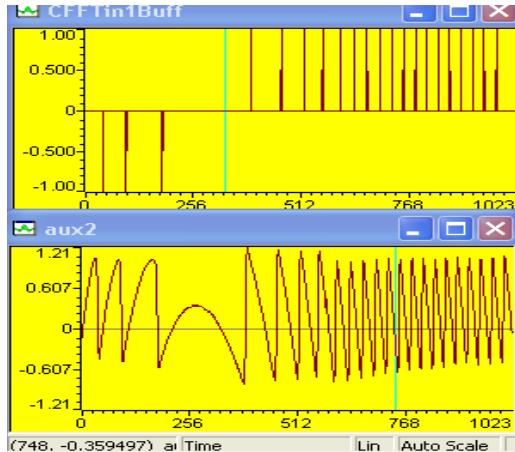
Expériences

Scanning de surface

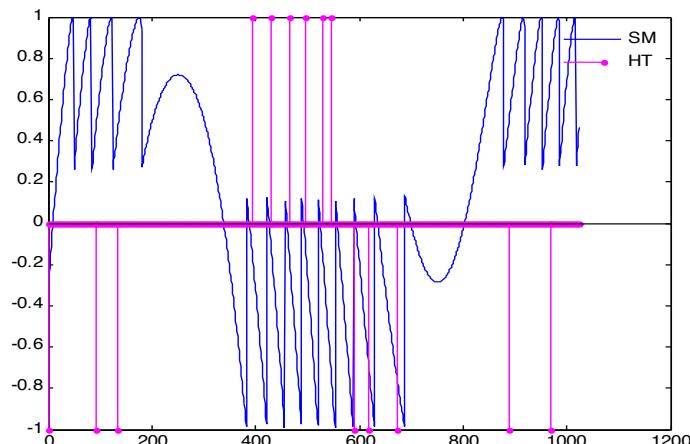
Conclusion

Implémentation des fonctions

En pratique
CCS



En théorie
MATLAB



Introduction

Développement

Expériences

Scanning de surface

Conclusion

- Résultat du Algorithme
- Vecteur de 1024 éléments réels

- Résultat dans la simulation

Implémentation des fonctions

Introduction

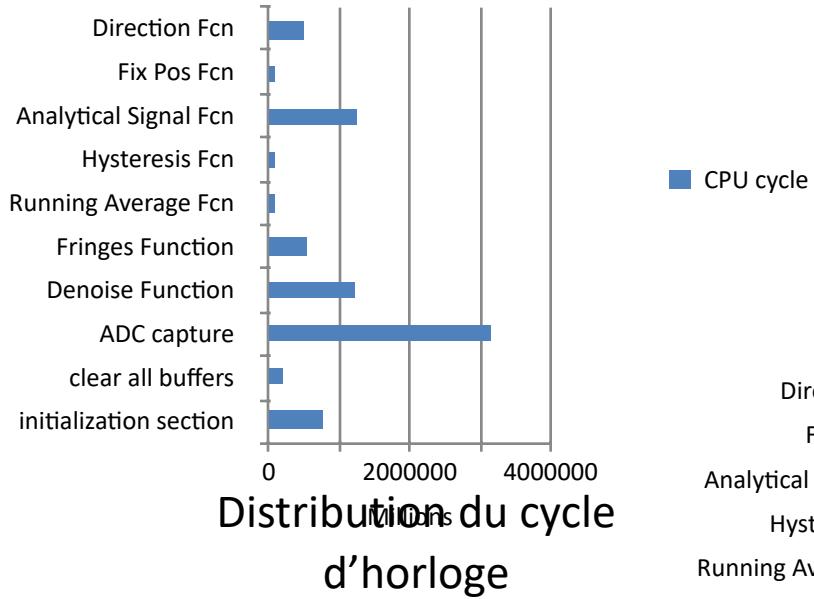
Développement

Expériences

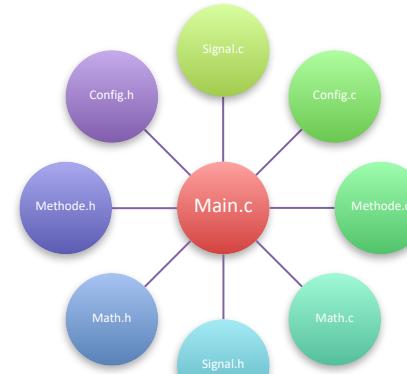
Scanning de surface

Conclusion

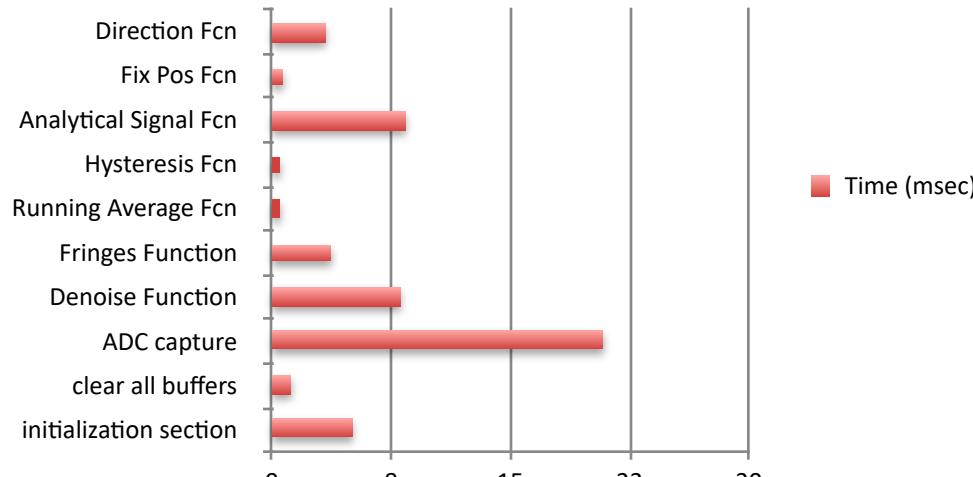
Cycle d'horloge



Caractéristiques principaux



Temps des Fonctions



Distribution du temps du DSP

Le programme

- Introduction
 - *Self-Mixing (SM)*
 - *Algorithme Détection de Franges*
- Développement
 - Outils logiciels
 - Outils matériel
 - Implémentation des fonctions
- Expériences
 - ✓ Mise en œuvre
- Scanning de surface
- Conclusion

Introduction

Développement

Expériences

Scanning de surface

Conclusion

Expériences

Introduction

Développement

Expériences

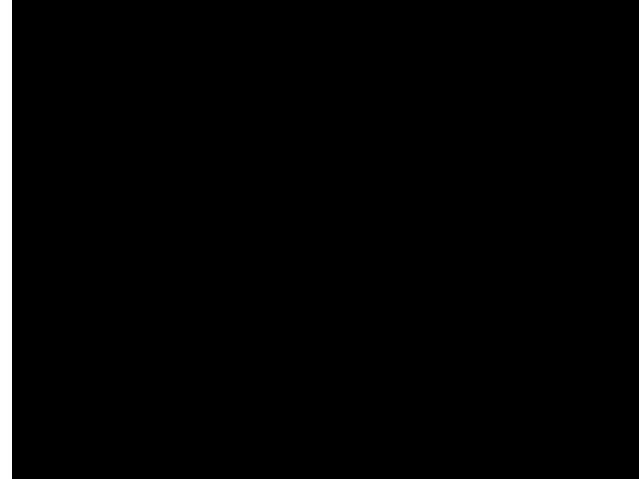
Scanning de surface

Conclusion

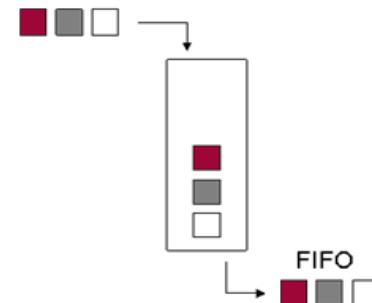
Caractéristiques principaux



Détection de franges



Méthode FIFO



	Cycle d'horloge	Temps (msec)
Temps de l'Algorithmhe	7085612	47.23741
Temps total	7839855	52.26570

Le programme

- Introduction
 - *Self-Mixing (SM)*
 - *Algorithme Détection de Franges*
- Développement
 - Outils logiciels
 - Outils matériel
 - Implémentation des fonctions
- Expériences
 - ✓ Mise en œuvre
- Scanning de surface
- Conclusion

Introduction

Développement

Expériences

Scanning de surface

Conclusion

Scanning de surface

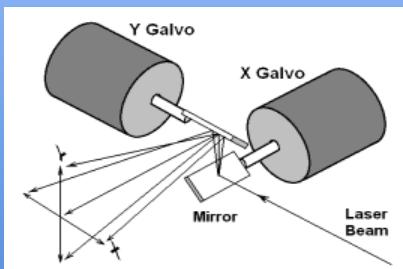
Introduction

Développement

Expériences

Scanning de surface

Conclusion



Problématique

2 heures/patient



Objectif:

Scanning surface optimal
(Fluides)

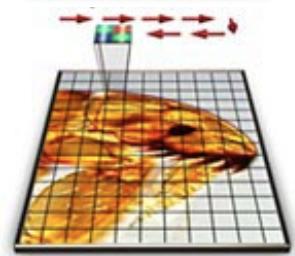
✓ Applications Médicaux

- Solides
- Fluides

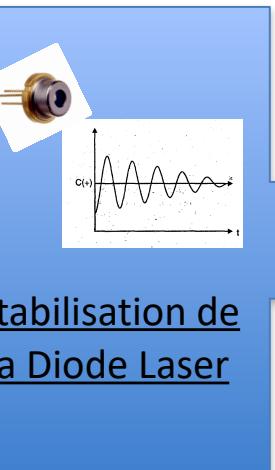
Scanning de surface

Problématique

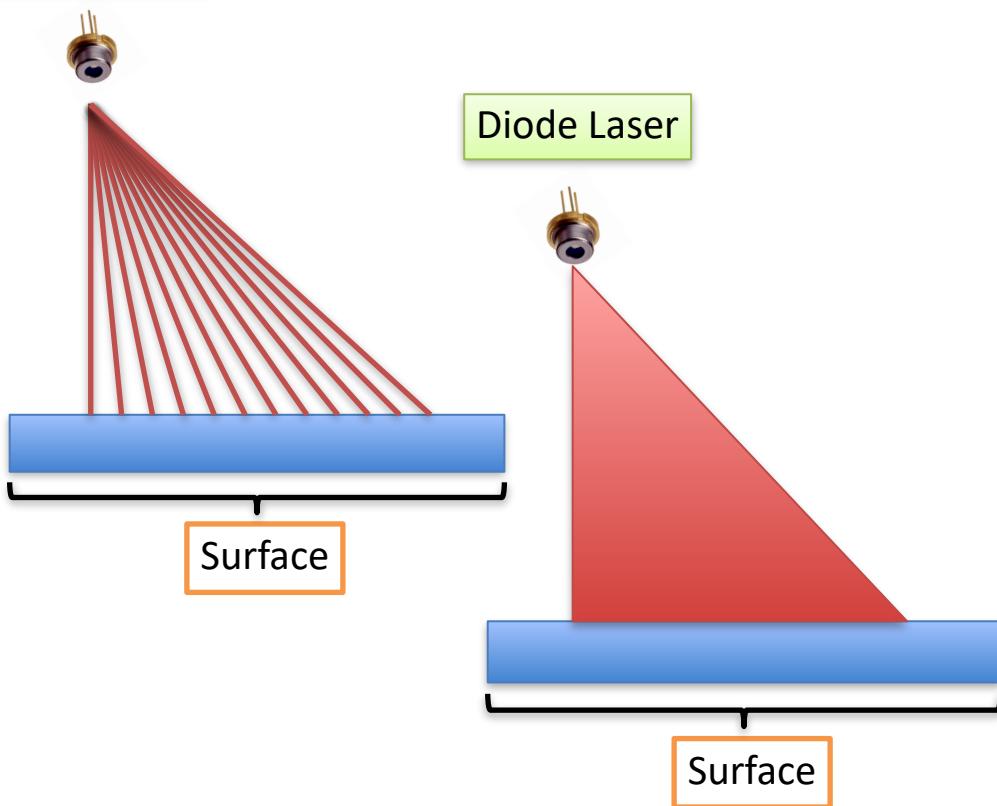
Méthode
point à point



Diode Laser



Méthode
de scanning proposé



Introduction

Développement

Expériences

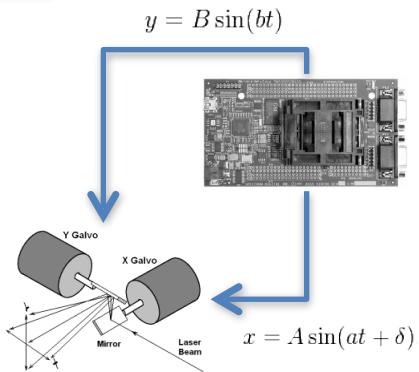
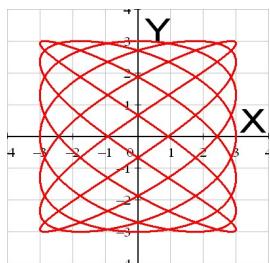
Scanning de surface

Conclusion

Scanning de surface

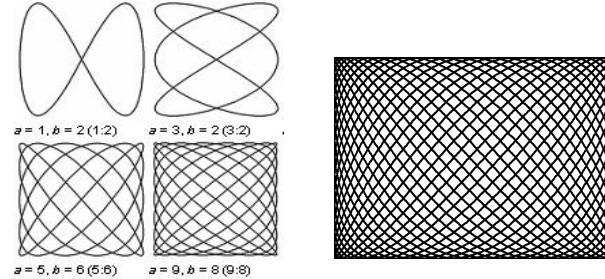
Solution

Méthode proposée



Courbes de Lissajous

Changement des paramètres



- ✓ Codage facile
- ✓ Deux paramètres variables (a,b)
- ✓ Deux sorties du DSP

Introduction

Développement

Expériences

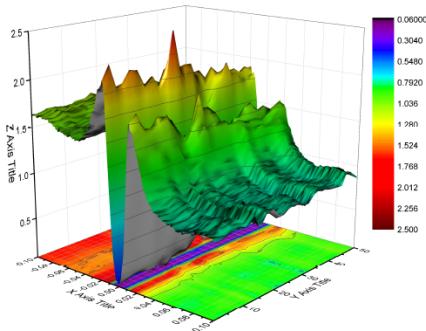
Scanning de surface

Conclusion

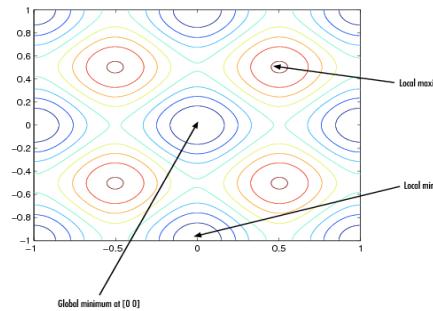
Scanning de surface

Solution

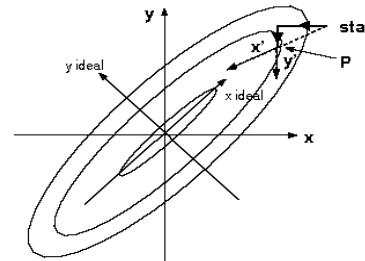
Comment trouver l'information d'intérêt?



Représentation en 3D des données



Représentation en 2D des données



Méthode de Powell

Introduction

Développement

Expériences

Scanning de surface

Conclusion

Le programme

- Introduction
 - *Self-Mixing (SM)*
 - *Algorithme Détection de Franges*
- Développement
 - Outils logiciels
 - Outils matériel
 - Implémentation des fonctions
- Expériences
 - ✓ Mise en œuvre
- Scanning de surface
- Conclusion

Introduction

Développement

Expériences

Scanning de surface

Conclusion

Conclusion

- ✓ Connaissances de SM de base et des outils
- ✓ Développement de prototype de mesure de déplacement en temps réel
- ✓ Méthode proposé pour le scanning de surface en diminuant le temps de capture de données

Introduction

Développement

Expériences

Scanning de surface

Conclusion

Perspectives

- Mise en œuvre le scanning de surface
- Système embarqué en temps réel

Merci de votre attention...

