

# Potentiels de la technologie FPGA dans la conception des systèmes

Avantages des FPGAs pour la  
conception de systèmes  
optimisés

Gérard FLORENCE  
Lotfi Guedria

1. Le CETIC en quelques mots
2. Générateur quantique de nombres aléatoires
3. Porting Software On Parallel Processors
4. Conclusion

- **Centre de recherche collective agréé la Région Wallonne**
  - Innovation & transfert de technologie au bénéfice des entreprises
  - Fondé en 2001



## Software & Services Technologies

Distributed systems, open source technologies, search engine

ACADEMICS

## Software & Systems Engineering

Methodology of development, Safety(Security), Quality

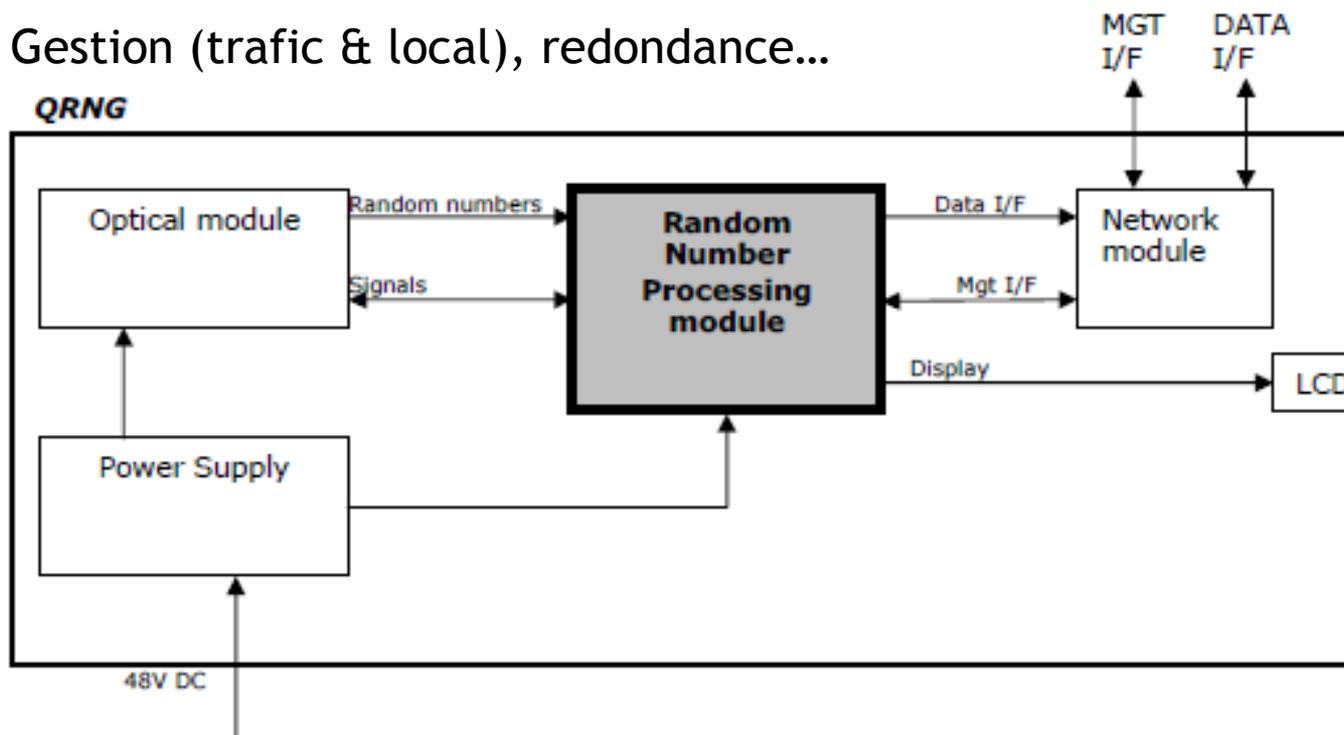
INDUSTRY

## Embedded & Communicating Systems

Embedded systems prototyping, wireless communications

## □ Générateur haut débit (4Gb/s) pour la cryptographie

- Architecture
  - Module optique: laser, photodiode, numérisation du signal
  - Tests statistiques (NIST) des séquences aléatoires
  - Distribution 10Gb Ethernet
  - Gestion (trafic & local), redondance...



## □ 14 tests statistiques (National Institute of Standards and Technology)

- Normalité, spectral, entropie, Maurer ...
- Probabilité d'avoir un résultat moins bon sur une séquence fournie par un générateur parfait
- Qualité estimée sur 100 calculs (matrice 14 x 100)
- Exécution non temps réel, vecteurs d'entrée [ $2^{16}$  -  $2^{20}$ ]

## □ Benchmark

- dual core, 32 bits, 2.4GHz
- 13Mbyte

## → Pseudo-RNG → True-RNG

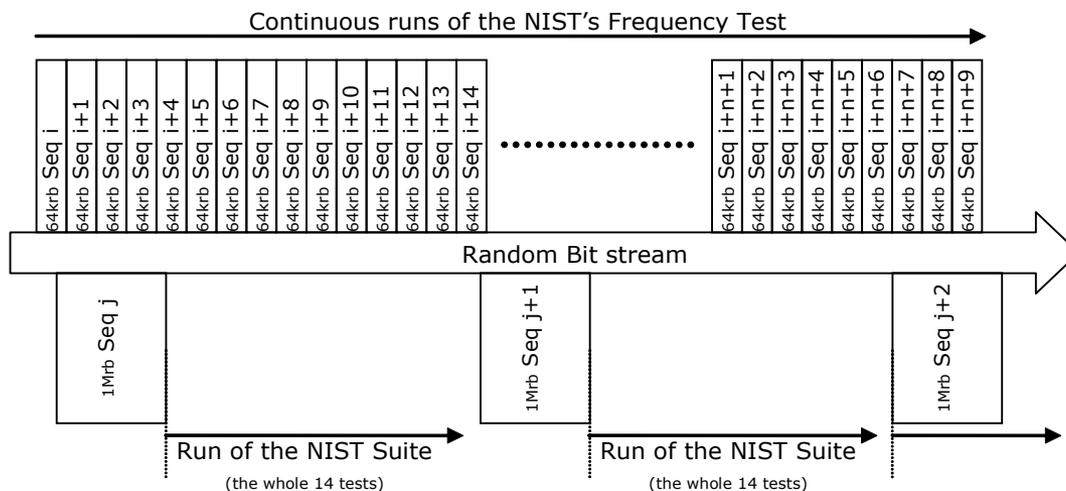
- Interruption du flot de bits en cas d'erreur
- Performance: # secondes, 14 tests

N°	Test	Input vector size (bits)	Indicative execution time PC dual core, 2.4GHz CETIC
1	The Frequency Test	$2^{16}$	0,014s
2	Frequency Test within a Block	$2^{16}$	0,0137s
3	The Runs Test	$2^{16}$	0,0146s
4	Test for the Longest-Run-of-Ones in a Block	$2^{16}$	0,0146s
5	The Binary Matrix Rank Test	$2^{16}$	0,0233s
6	The Non-overlapping Template Matching Test	$2^{20}$	2,096s
7	The Overlapping Template Matching Test	$2^{20}$	0,3166s
8	Maurer's "Universal Statistical" Test	$2^{20}$	0,307s
9	The Linear Complexity Test	$2^{20}$	6,57s

## ❑ Contraintes

- Frequency test: nombre de '0' et '1', exécuté en continu sur le flot de bit → défauts évidents
  - 13 tests exécutés séquentiellement → précision de la mesure de la qualité
- Temps réel, volume de données traité, traitements nombreux, débit élevé

## ❑ Optimisation des performances: Exécution parallèle → 2 ressources

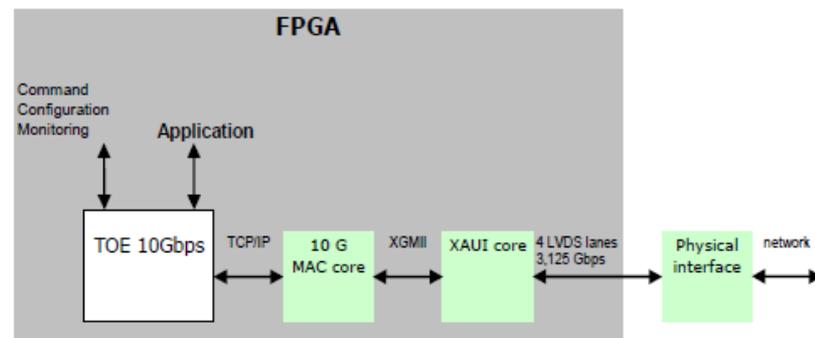
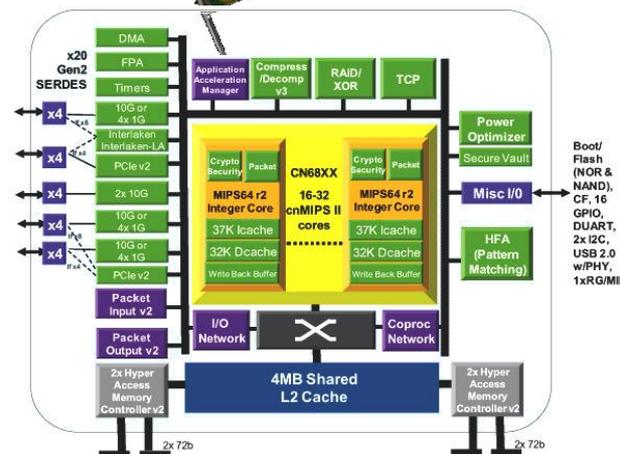


## Network module

- 4096 TCP& UDP connexions,
- 10 Gbit/s 802.3x flow control,
- IPV4, IPV6
- Jusqu'à 4096 VLAN
- Jumbo frames

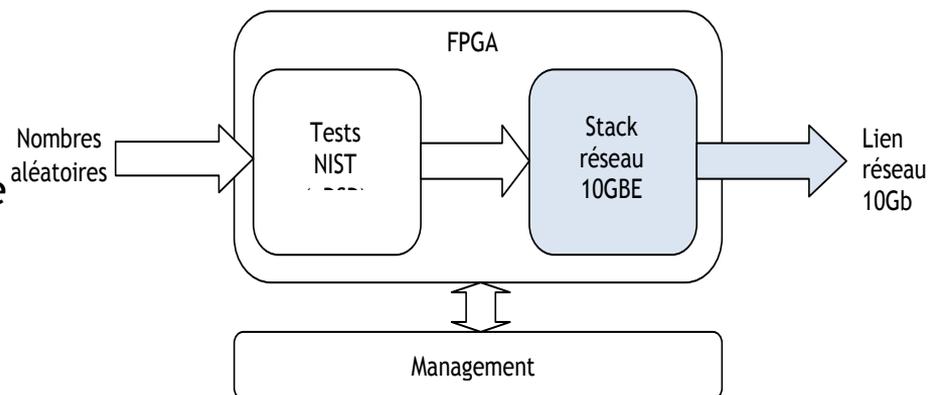
## Comparatif des solutions

- Carte accélérateur réseau (Chelsio ...) HW simplifié
  - 700\$, compatible avec nos spécifications
  - Développement : Interface avec l'équipement, driver
- Processeur réseau (multi-coeurs), performance
  - Développement : firmware, interface
- "TCP Offload Engine", IP FPGA
  - Fonctionnalité incomplète
  - Prix # k€ (licence projet)
  - Evolutions possibles



## ❑ IP TCP Offload Engine:

- Meilleur compromis coût/performances
- Coût licence amorti par la quantité
- Développement limité: interface simple (fifo) avec l'application
- **Optimisation des ressources: IP intégrée en FPGA avec les tests statistiques**



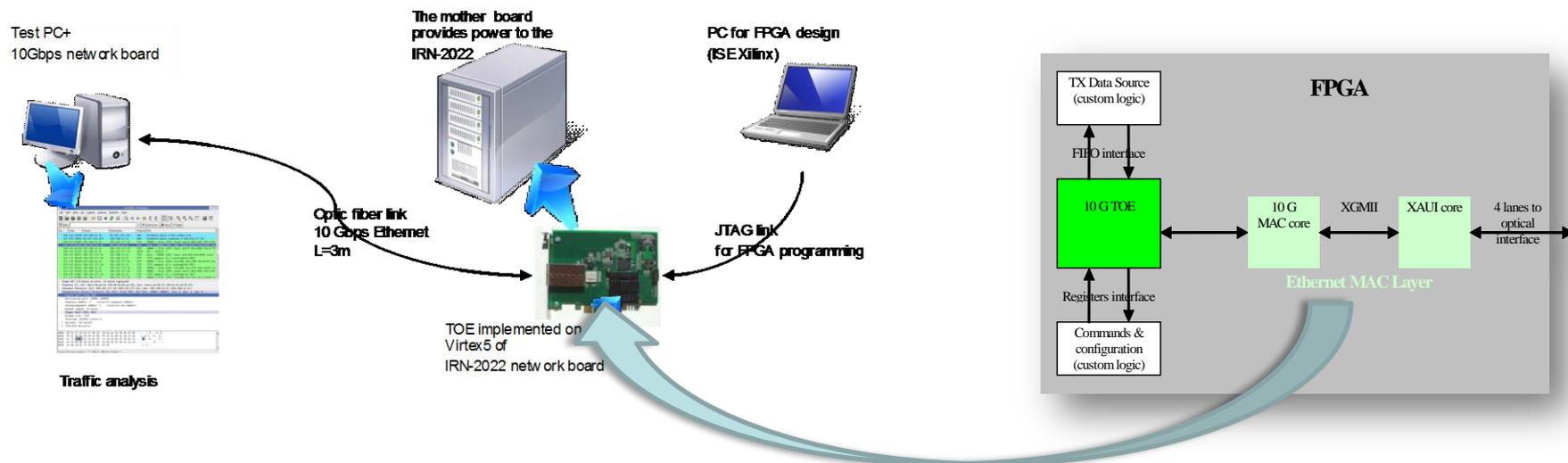
## ❑ Fournisseurs

- IP Blaze, fournisseur de technologies Gigabit Ethernet → solution à base de CPU
- Fraunhofer, laboratoire allemand de recherche appliquée → architecture hardware

## ❑ Modification importante: TOE → Super TOE

- 1 connexion TCP/UDP → 4096 connexions
- Sauvegarde des données et du contexte pour chaque connexion
- Espace mémoire externe (DDR) requis
- Evaluation de l'impact avec le fournisseur

## ❑ Mesure des performances de chaque source



## ❑ Fiabilité du fournisseur: "QIP Metric" VSIA

- **Evaluation du fournisseur:** Application d'un plan qualité, gestion des évolutions, support au client, stabilité de la société ...
- **Méthodologie de conception:** facilité de réutiliser l'IP, règles de conception détaillées et les tests de vérification.
- **Facilité d'intégration:** documentation utilisateur
- Impératifs, règles, directives

## ❑ Optimisation des ressources: Test NIST + IP TCP Offload Engine

- IP TCP Offload Engine:  $\approx 20\%$  des ressources (XC5Vlx110)
- Test NIST: Portage compliqué  $\rightarrow$  FPGA grosse capacité, développement inutilement long (code source existant)

## ❑ "Frequency Test" : FPGA + mémoire externe

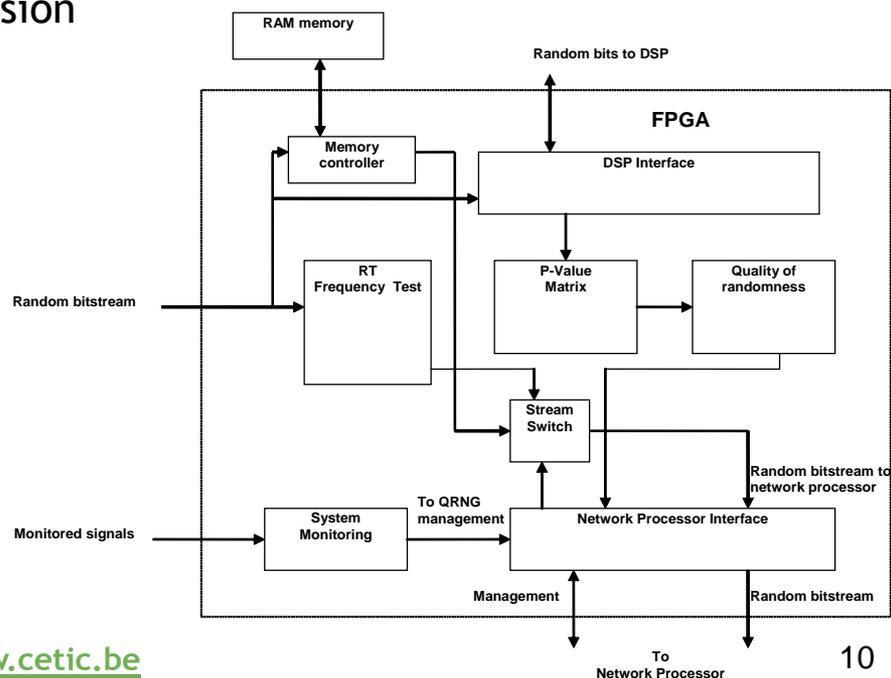
- Traitement bit à bit en continu

## ❑ "Tests séquentiels: DSP, double précision

- Traitement sur des blocs prélevés
- Interface avec "Frequency test"

## ❑ Composants couramment utilisés

- Virtex XC5VL110T,
- ADSP-TS201S (Analog Device) 32 bit fixed or floating point, 600MHz core

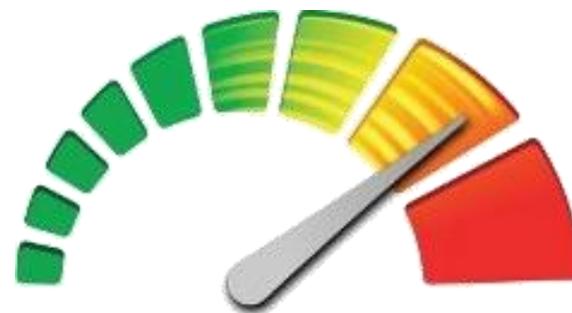


## □ Une méthodologie de parallélisation des applications logicielles

- Pour conseiller et assister les entreprises dans leur démarche d'amélioration des performances de leurs applications

## □ Pourquoi une méthodologie:

- Comment optimiser le code existant
- Identifier les points à améliorer, profiling
- Choisir les technologies, les outils, évaluer le facteur d'accélération...
- Rendre le code parallélisable
- Evaluer la charge de développement, coûts
- Portage du code et tests



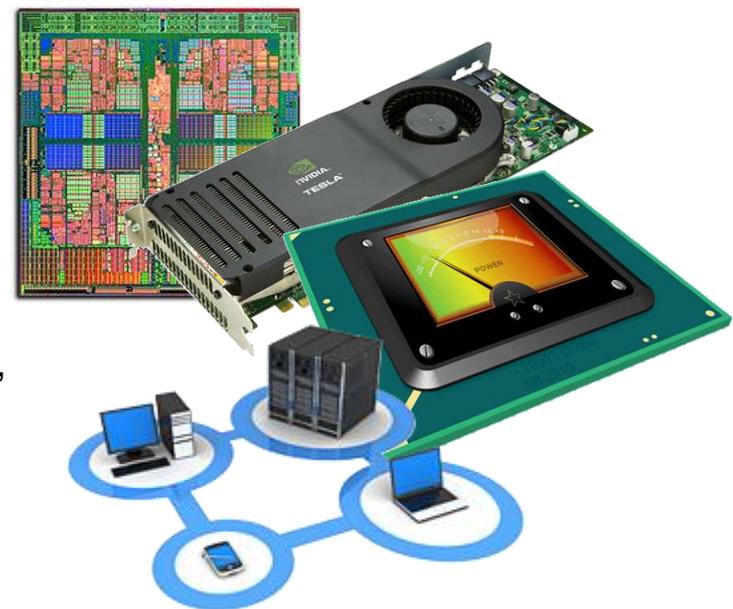
## □ Définir une approche standardisée pour l'adaptation des logiciels

## □ Les applications

- Tous secteurs : Télécommunications, traitement d'images, bioinformatique, simulation, modélisation ...
- Installations diverses: 1 machine, architecture distribuée

## □ Différentes technologies

- **Multi-coeurs:** Multitâches asynchrones, indépendantes
- **GPU:** Traitements massifs synchrones
- **FPGA:** Traitements parallèles
- **Architecture distribuée:** Gros volumes de données, flux d'entrée nombreux
- **Environnements hétérogènes**



## □ Une méthodologie basée sur des études de cas industriels

## ❑ Comparaison des technologies

	Multi-cœur	GPU	FPGA
Architecture	Cœurs indépendants	Blocs de cœurs	Matrice de ressources programmables
Exécution parallèle	Tâches multiples Processus, dizaine de threads asynchrones	Traitement massivement parallèle Très nombreux (millions) threads synchrones	Duplication des blocs fonctionnels, pipelining
Synchronisation des traitements	Signaux, mutex, files d'attentes	Fonctions de synchronisation des threads	Synchronisation hardware: Fifo, flags, timers ...
Formats des données	Fixe (8/16/32/64)	Fixe (8/16/32/64/128/256)	Libre (1...n) mais limité en ressources
Mémoire interne	Grands cache	Cache petits	Bloc mémoire, mémoire distribuée, petite capacité
consommation	Consommation faible à moyenne. Consommation rapportée à un cœur : moyenne à élevée (jusqu'à 25W/cœur)	Consommation globale élevée. Consommation rapportée à un cœur : faible (<1W/cœur)	Technologie récentes faibles consommation Consommation proportionnelle à la complexité du design, fréquence
Prix	Peu couteux jusqu'à 4 cœurs	Peux couteux	Couteux

## ❑ Le GPU complété par le FPGA

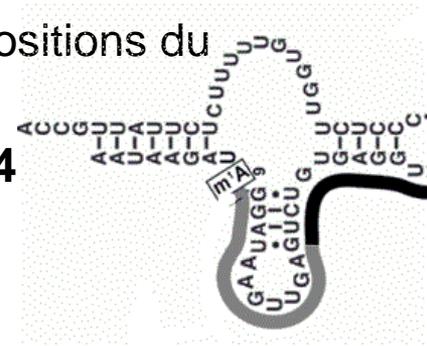
- Limité par la logique de décision
- Branchements dans la logique d'exécution → invalidations de pipeline
- Architecture figée, pas d'opération bit par bit ...

## ❑ Applications pour FPGA

- Traitement de signal, d'image : filtrage, corrélation, codeur JPEG, scaler ...
- Calcul arithmétique, matriciel...
- Cryptographie ....

## □ Bioinformatique: algorithme de Zuker

- Correspondance entre une amorce ADN (courte séquence utilisée pour le séquençage ADN) et des séquences connues.
- Calcul de l'énergie de liaison des nucléotides, pour chaque positions du primer dans la séquence, dans les 2 sens, sur les 2 brins
  - **Matrice (taille de la séquence x taille du primer) X 4**
- Plusieurs traitements:
  - Calcul en fonction du voisinage de chaque nucléotide
  - Prise en compte des boucles

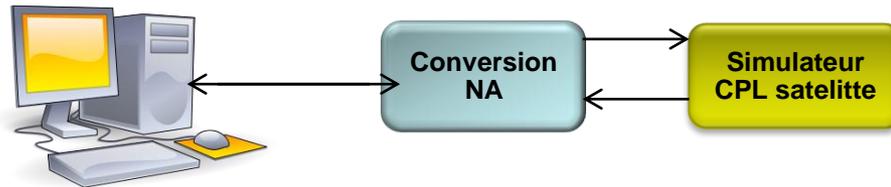


## □ Mise en œuvre sur la plateforme FPGA

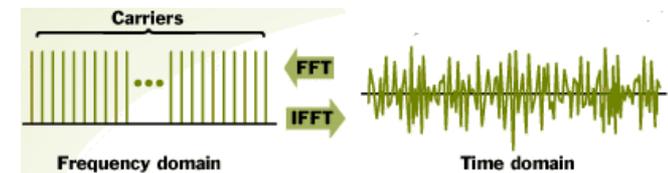
- Transfert de la matrice dans la mémoire de la carte FPGA
- Portage des fonctions de calcul d'énergie dans le FPGA
- 10 calculs en parallèle

## ❑ Modem CPL pour environnement satellite

- Démonstrateur implémenté sous Matlab (non temps réel)



- Transmission multi-porteuses
- Portage de fonctions de traitement de signal (FFT, auto corrélation, filtrage...) pour la réception et la transmission



## ❑ Mise en œuvre

- Optimisation des communications entre host et accélérateur FPGA
- Implémentation à base d'IP (Coregen ISE)
- Fonctions MEX

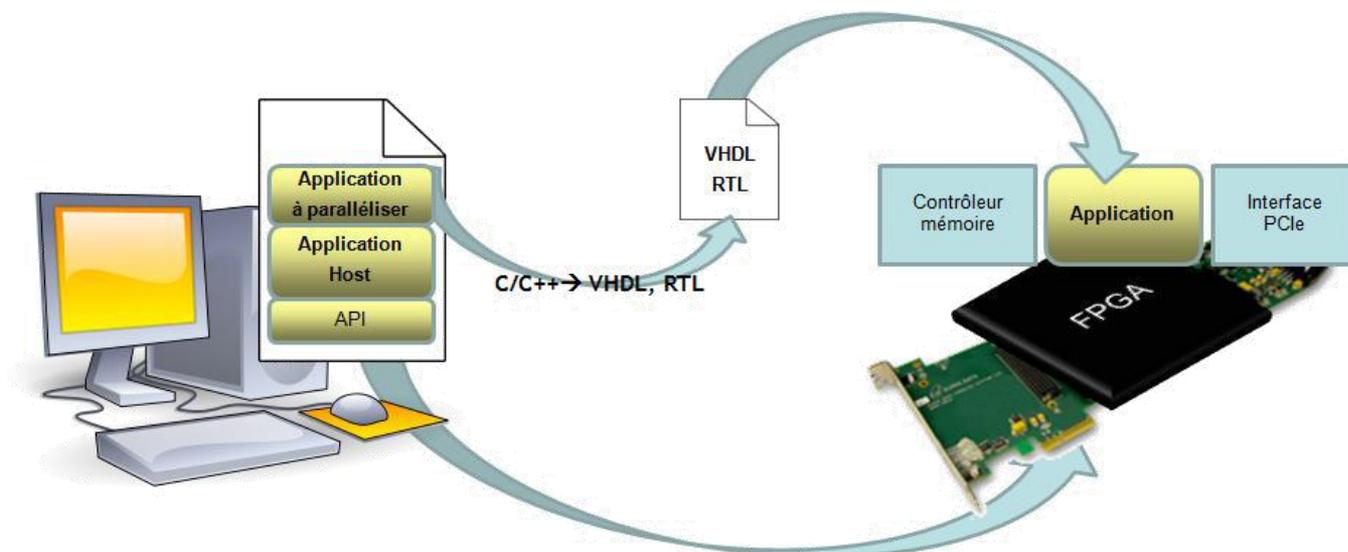
## ❑ Facteur d'accélération visé $\approx 10$

## ❑ Plateforme d'essais

- Alpha Data: Virtex VI, DDR3, PCIe
- SDK : couche basses interface PCIe, contrôleur DDR3, accès DMA, API
- Outil : ROCCC, CoDeveloper, HDL Coder, Open CL

## ❑ Maîtrise de l'environnement de développement

## ❑ Interfaçage de l'application avec les couches basses



### ❑ Critères de sélection de la technologie FPGA

- Traitement parallèles, performances
- Coût de développement, coût de production

### ❑ Technologie actuellement dédiée aux traitements intensifs, temps réel

- Secteurs traditionnels: audio, électronique grand publique, communications, traitement d'image, médical ...

### ❑ Technologie moins adaptée aux petits systèmes embarqués ...

- Microcontrôleur:
  - Connectivité (UART, USB, CAN, IOs...)
  - Traitements séquentiels, interruption, peu de calcul
- **Mais potentiellement candidate**
  - Intégration, consommation, familles low cost
  - Petite plateforme ARM, Cyclone ...





Your Connection to **ICT** Research

*Thank you*

## **CETIC**

Aéroport de Charleroi-Gosselies  
Rue des Frères Wright, 29/3

B-6041 Gosselies

[info@cetic.be](mailto:info@cetic.be)

**[www.cetic.be](http://www.cetic.be)**

