

# Réunion Pôle ITT

## FPGA: accélérateur de calcul

19.10.2018

Bogdan Vulpescu

LPC+



# Le contexte

- IN2P3 master-projet Decalog, Gilles Grasseau (LLR) et David Chamont (LAL)

<https://gitlab.in2p3.fr/CodeursIntensifs/DecaLog/wikis/home>

- projets:

- ComputOps (calcul intensif et conteneurs)

- **Reprises** (calcul reproductible)

<https://gitlab.in2p3.fr/CodeursIntensifs/Reprises/wikis/home>

- Reprises: 14.5 % ETP (4C+7IT du LAL, LLR, IPHC, LUPM, LAPP, LPC, IPNO)  
+ 7 *observateurs*
- AMS, ATLAS, BELLE, CMS, CTA, FERMI, GEANT4 & GATE, LHCb, LSST...
- 3 ans: 2018, 2019, 2020

# FPGA: dev kit Cyclone V GT, 5CGTFD9E5F35(C7N)



PCIe x4



câble JTAG pour la programmation  
du FPGA (pas possible via PCIe)

# L'environnement de programmation

- Board Support Package (BSP) avec le SDK OpenCL

[http://www.alterawiki.com/wiki/The\\_board\\_support\\_package\\_of\\_Cyclone\\_V\\_GT\\_Development\\_Kit\\_for\\_Intel\\_FPGA\\_SDK\\_OpenCL](http://www.alterawiki.com/wiki/The_board_support_package_of_Cyclone_V_GT_Development_Kit_for_Intel_FPGA_SDK_OpenCL)

(23.05.2017, beta)

- Dell Precision T3610
- Linux CentOS 7.4
- Quartus Prime Standard Edition 17.0 + Intel FPGA SDK for OpenCL 17.0
- clralicepc11 (groupes *alice*, *calcul* et *local*)
- mais aussi:
  - CPU + OpenCL (Intel Xeon E5-1607 3000 MHz)
  - GPU + OpenCL (Nvidia Quadro 2000M 1100 MHz)

## Une application:

### La Transformée de Fourier (analyse du signal)

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-i\omega t} dt \quad \longleftarrow \text{directe}$$

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega)e^{i\omega t} d\omega \quad \longleftarrow \text{inverse}$$

$$X_k = \sum_{n=0}^{N-1} x_n e^{-i2\pi kn/N} \quad k = 0, \dots, N-1$$

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{i2\pi kn/N} \quad n = 0, \dots, N-1$$

discrète (échantillonnée, DFT)

# La Transformation de Fourier Rapide (FFT)

The discrete Fourier transform (DFT) is defined by the formula:

$$(1) \quad X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} nk},$$

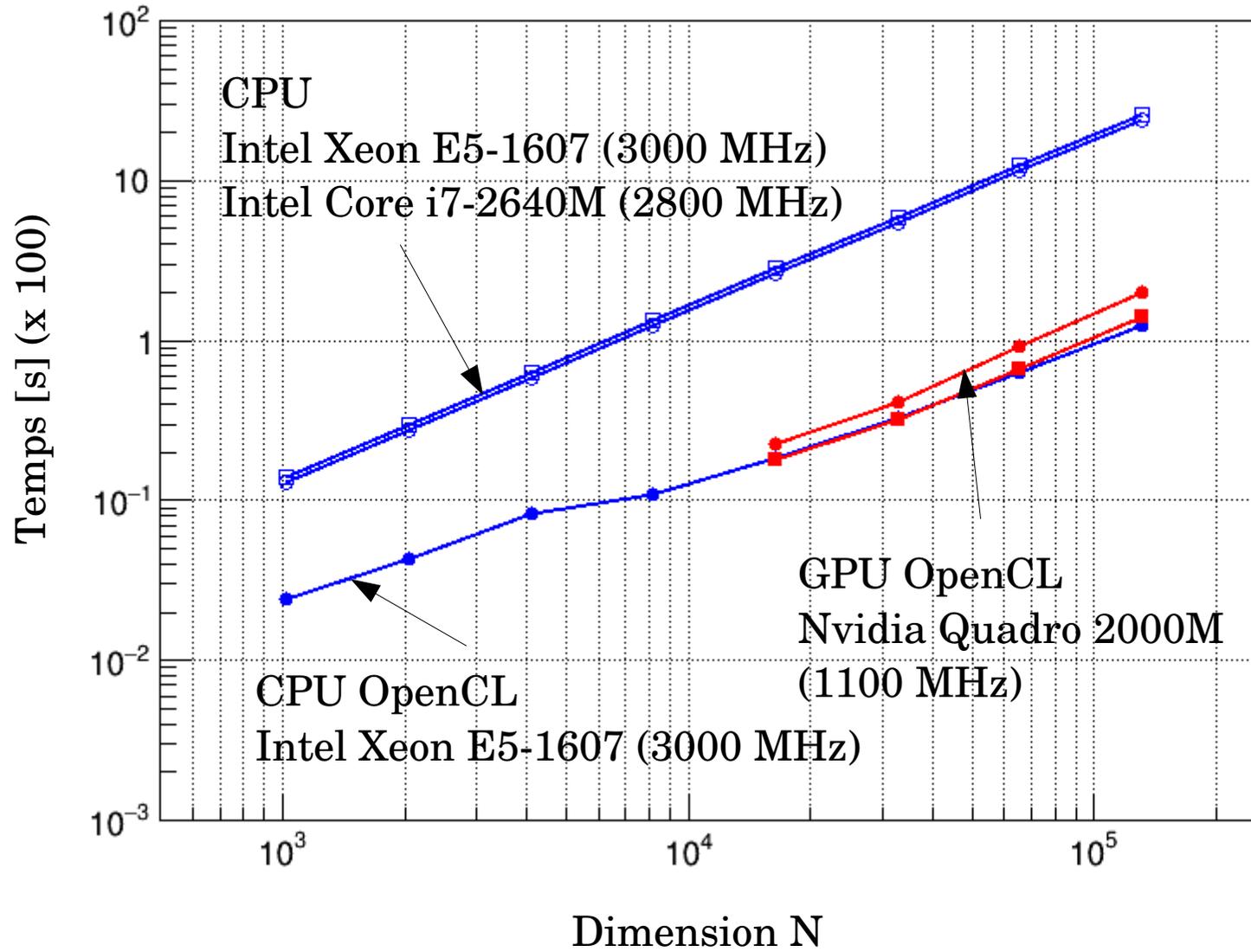
where  $k$  is an integer ranging from 0 to  $N - 1$ .

$$(2) \quad X_k = \sum_{m=0}^{N/2-1} x_{2m} e^{-\frac{2\pi i}{N} (2m)k} + \sum_{m=0}^{N/2-1} x_{2m+1} e^{-\frac{2\pi i}{N} (2m+1)k}$$

$$(3) \quad X_k = \underbrace{\sum_{m=0}^{N/2-1} x_{2m} e^{-\frac{2\pi i}{N/2} mk}}_{\text{DFT of even-indexed part of } x_n} + e^{-\frac{2\pi i}{N} k} \underbrace{\sum_{m=0}^{N/2-1} x_{2m+1} e^{-\frac{2\pi i}{N/2} mk}}_{\text{DFT of odd-indexed part of } x_n} = E_k + e^{-\frac{2\pi i}{N} k} O_k.$$

$$(4) \quad \begin{aligned} X_k &= E_k + e^{-\frac{2\pi i}{N} k} O_k \\ X_{k+\frac{N}{2}} &= E_k - e^{-\frac{2\pi i}{N} k} O_k \end{aligned}$$

# Résultats FFT



# La transformée FFT sur FPGA

- problème avec la taille de l'implémentation du noyau, trop large pour le modèle de FPGA, Cyclone V GT, 5CGTFD9E5F35(C7N)
  - 301k LE (*logic element*)
  - 113,560 ALM (*adaptive logic module*)
- une contribution importante apporte l'implémentation des fonctions trigonométriques (et le calcul de leur argument)
- idée: les remplacer par des tableaux pré-calculés (*look-up-table*, LUT)

```
Compiler Command: aoc test_c5gt.cl -o test_c5gt.aocx --board c5gt
```

```
+-----+
; Estimated Resource Usage Summary
+-----+-----+
; Resource                + Usage
+-----+-----+
; Logic utilization        ; 212%
; ALUTs                    ; 124%
; Dedicated logic registers ; 98%
; Memory blocks            ; 71%
; DSP blocks                ; 75%
+-----+-----+
```

# Rapport de compilation noyau FFT généré par Quartus (>= 17.0) (HTML)

HLD FPGA Reports (Beta) [View reports...](#)

Area analysis of system  
(area utilization values are estimated)

Notation *file:X > file:Y* indicates a function call on line X was inlined using code on line Y.

[^ Collapse All](#)  
[v Expand All](#)

	ALUTs	FFs	RAMs	DSPs	Details
▼ Block12	36447 (16%)	59733 (13%)	92 (8%)	75 (22%)	
Cluster logic	235	278	5	0	• Logic requ...
► State	8128	23293	6	0	• Resources ...
▼ Computation	28084	36162	81	75	
► test_c5gt.cl:70	56	0	0	0	
▼ test_c5gt.cl:71	13263	18591	56	43	
'fused sin/cos' Function Call	2848	2031	17	25	
Call	206	2	0	0	
Floating Point Divide	6680	11474	39	14	
Floating Point Multiply	3366	4933	0	4	
Fptrunc	163	151	0	0	

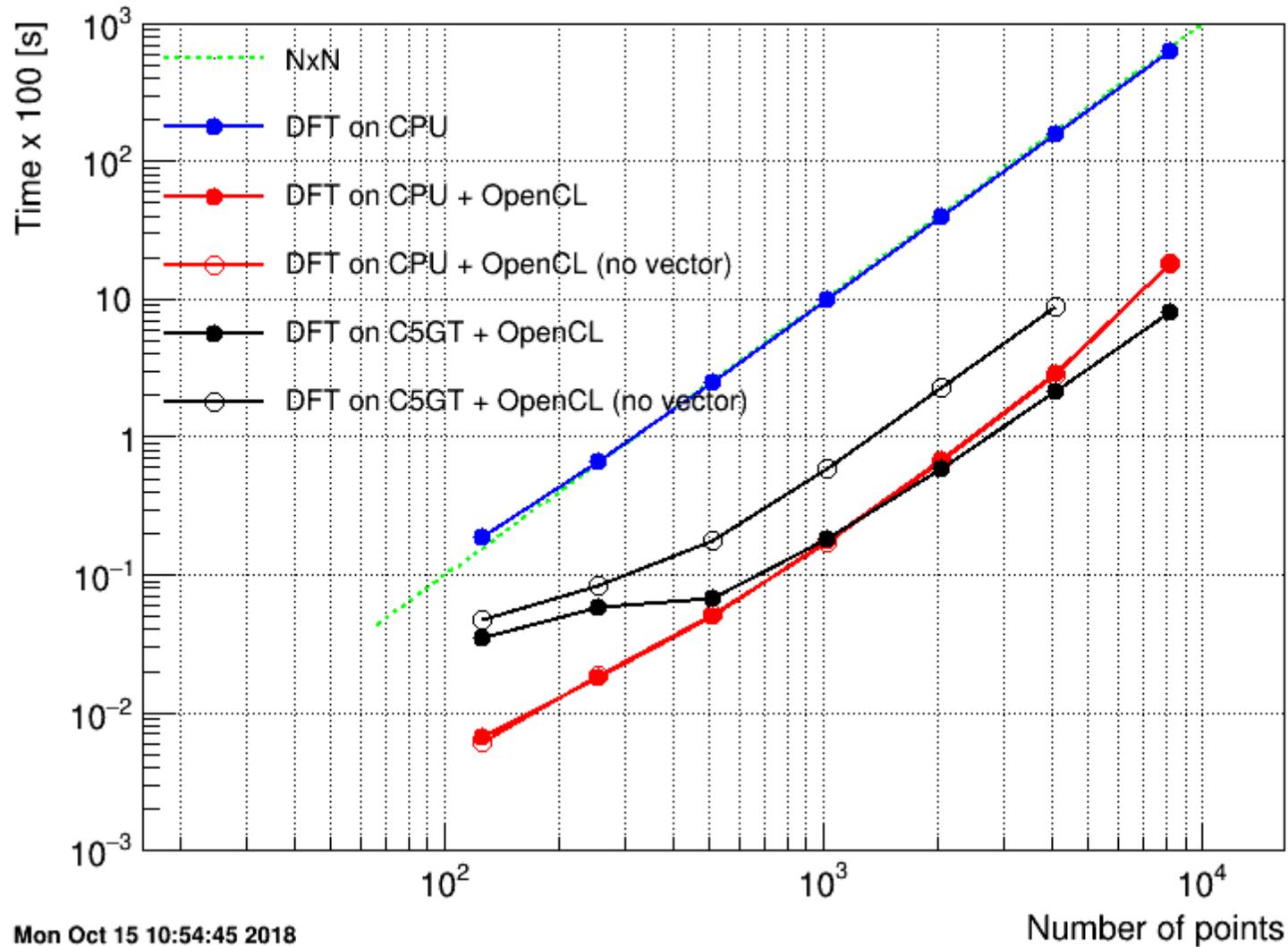
test\_c5gt.cl

```

52     diff12 = x1 - x2;
53     sum34 = x3 + x4;
54     diff34 = (float2)(x3.s1 - x4.s1, x4.s0 - x3.s0) *
        dir;
55     l_data[l_addr] = sum12 + sum34;
56     l_data[l_addr+1] = diff12 + diff34;
57     l_data[l_addr+2] = sum12 - sum34;
58     l_data[l_addr+3] = diff12 - diff34;
59     l_addr += 4;
60     g_addr += 4;
61 }
62
63 /* Perform initial stages of the FFT - each of length
64    N2*2 */
64 for(N2 = 4; N2 < points_per_item; N2 <= 1) {
65     l_addr = get_local_id(0) * points_per_item;
66     for(fft_index = 0; fft_index < points_per_item;
67         fft_index += 2*N2) {
67         x1 = l_data[l_addr];
68         l_data[l_addr] += l_data[l_addr + N2];
69         l_data[l_addr + N2] = x1 - l_data[l_addr + N2];
70         for(i=1; i<N2; i++) {
71             cosine = cos(M_PI*i/N2);
72             sine = dir * sin(M_PI*i/N2);
73             wk = (float2)(l_data[l_addr+N2+i].s0*cosine +
74                         l_data[l_addr+N2+i].s1*sine,
75                         l_data[l_addr+N2+i].s1*cosine -
76                         l_data[l_addr+N2+i].s0*sine);

```

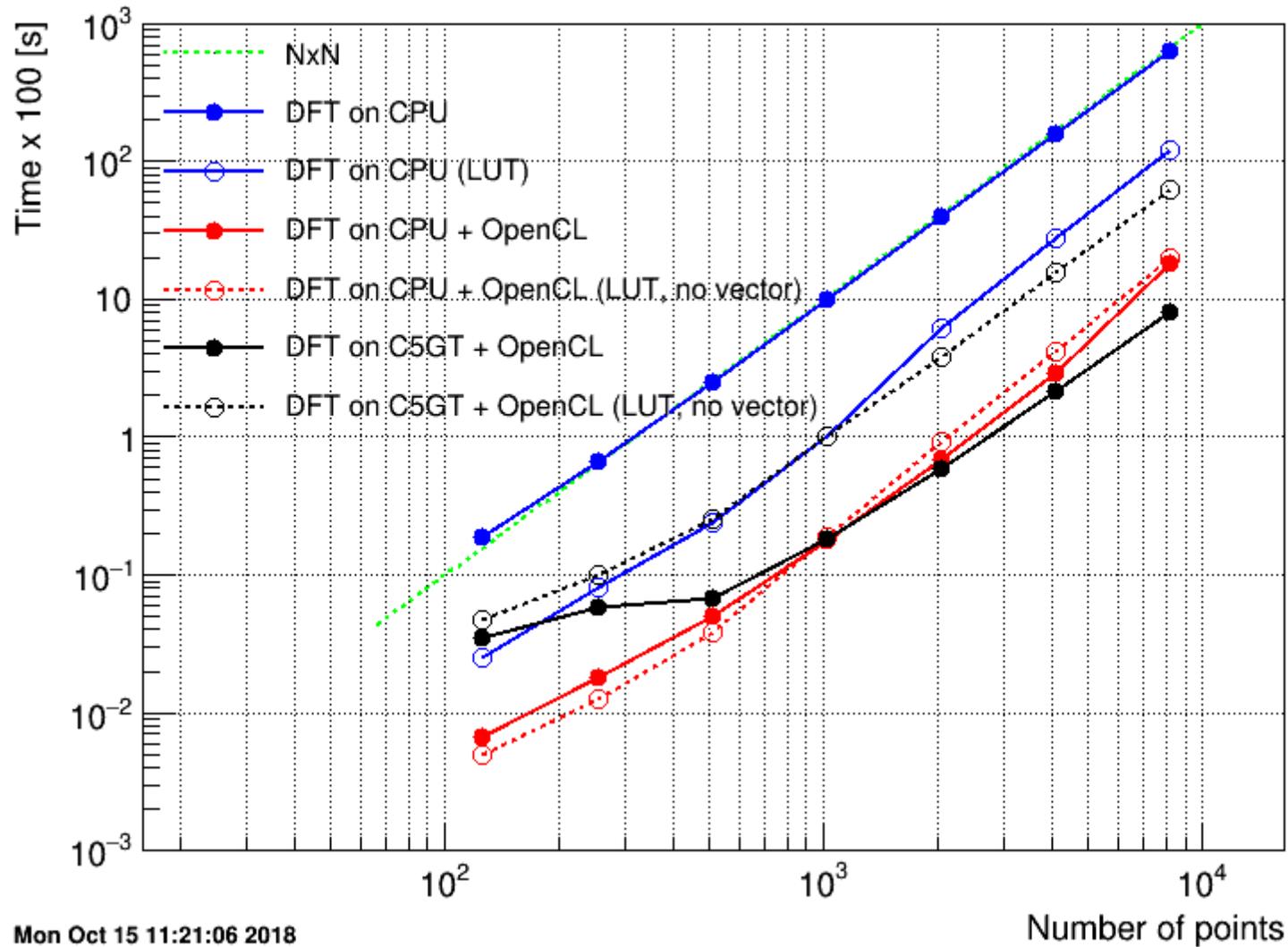
# Résultats DFT: CPU et FPGA



problèmes  
d'exécution  
pour le GPU...

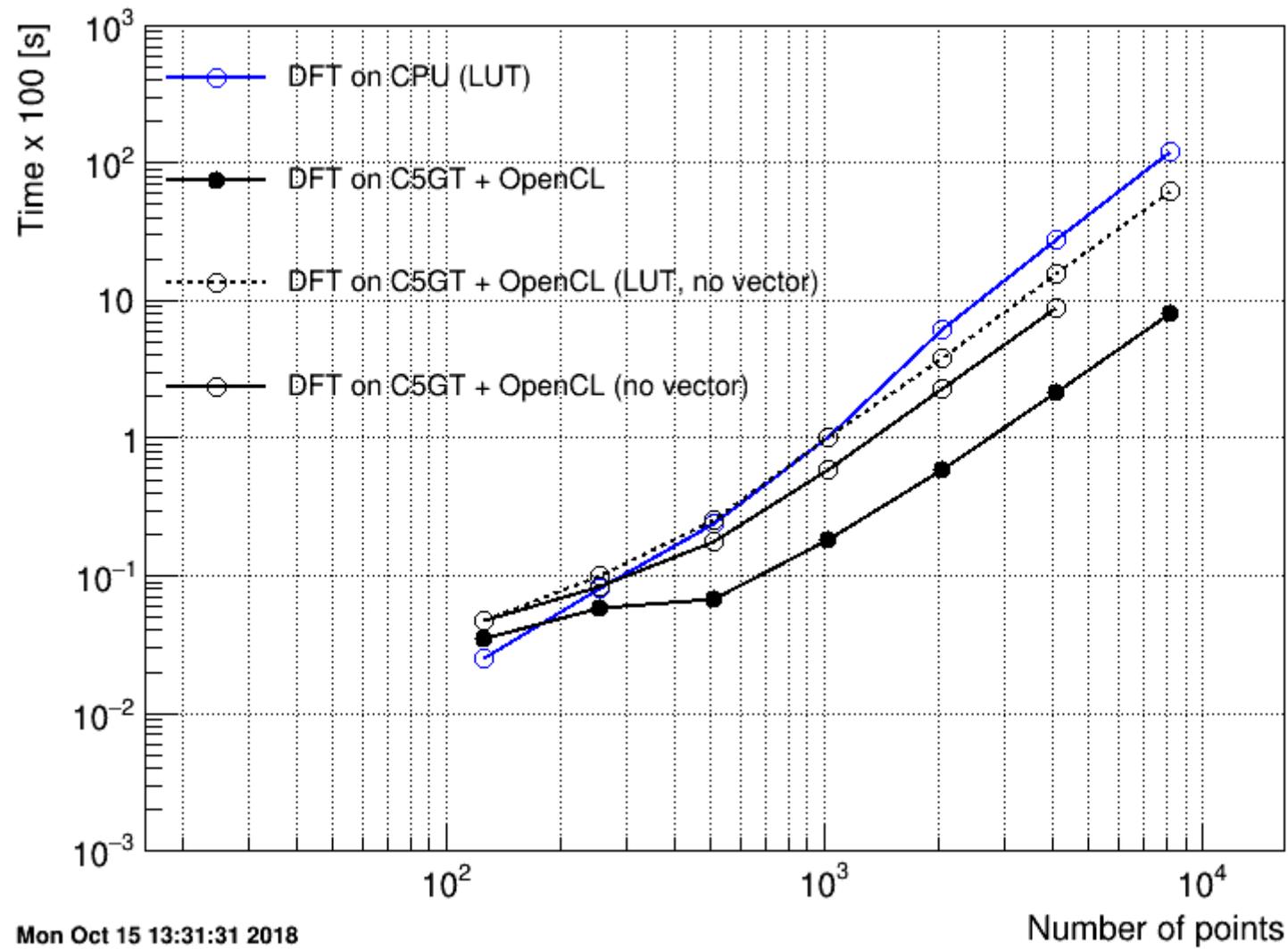
Mon Oct 15 10:54:45 2018

# Résultats DFT avec LUT pour *sin* et *cos*



Mon Oct 15 11:21:06 2018

# Résumé DFT sur FPGA



## Pour la suite

- proposition de Gilles Grasseau (LLR): carte FPGA pour le calcul au LLR (spécialisés en calcul sur GPU), visite pour l'installation et accès à distance
  - budget  $\geq$  10 k€
  - <http://www.terasic.com.tw/cgi-bin/page/archive.pl?Language=English&CategoryNo=11&No=911>  
Altera Stratix® V GX FPGA (5SGXEA7N2F45C2), voir la page suivante
- besoin d'une plateforme commune (tutoriel Reprises) avec un accélérateur FPGA
- BSP pour d'autres cartes ? appel aux spécialistes: comment créer un projet Quartus pour un BSP (apparemment il y a de la doc) ?  
... venez analyser celui que j'utilise ...

# OpenCL for Terasic DE5-Net Board

 Like Sign Up to see what your friends like.

July 21st , 2014 【 Terasic Technologies 】



BSP avec OpenCL SDK pour les versions : 13.1, 14.0/14.1, 16.0/16.1, 18.0

Prix Terasic : \$6,250 + \$200 licence OpenCL (versions < 17.1)

# Sur ma liste pour le Père Noël

Aria 10 FPGA dev kit 10AX115S2F45I1SG

1,150k LE

427,200 ALM

