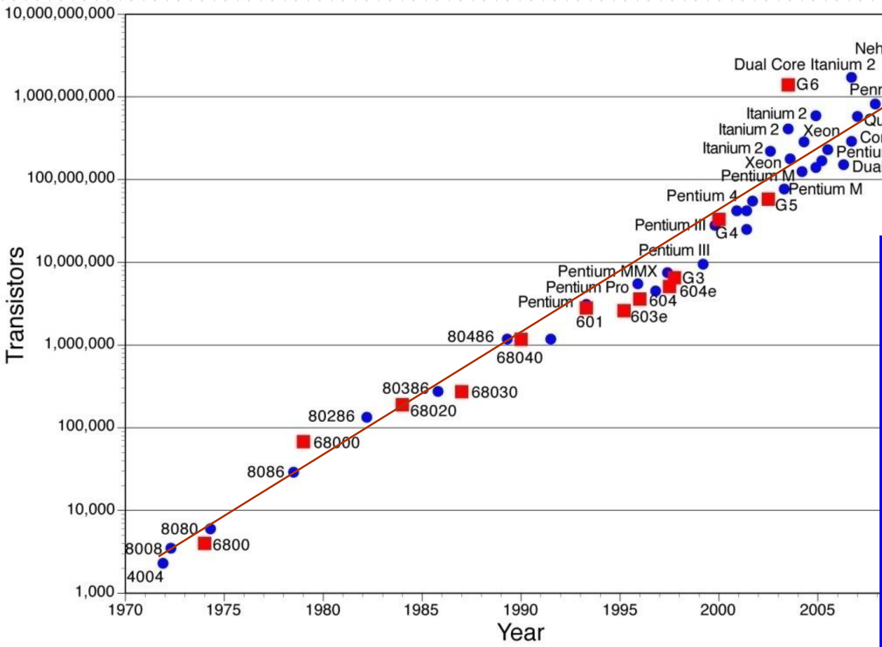
Test on simulated data of cuInspiral, the GPU high arithmetic intensity prototype library for coalescing binaries detection.

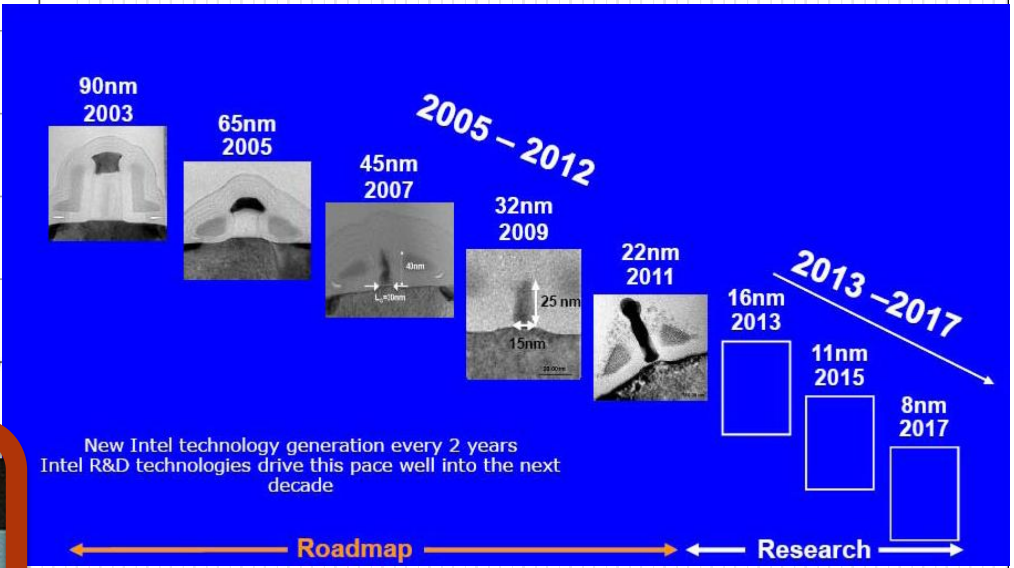
Dr. Leone B. Bosi – INFN Perugia, ET Project

GWDAW 14 – Rome (Italy)
25-29 January 2010

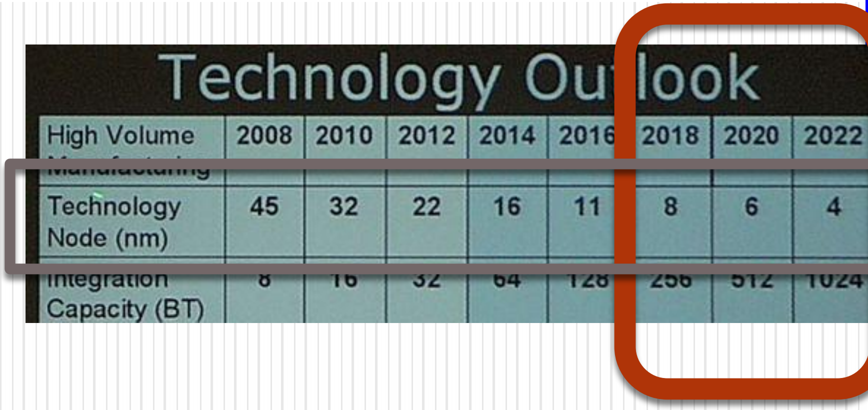
Technological outlook:



Moore’s law has been respected for several decades, increasing exponentially the number of transistors per CPU and computing power.



New Intel technology generation every 2 years
Intel R&D technologies drive this pace well into the next decade



Technology Outlook

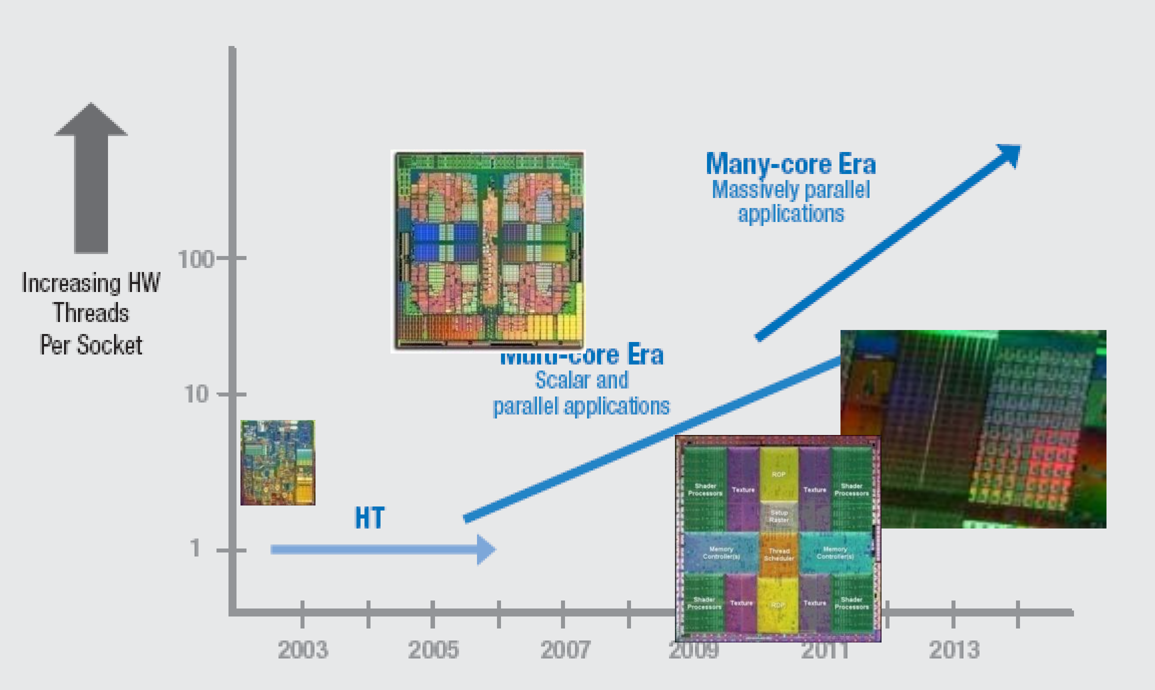
High Volume Manufacturing	2008	2010	2012	2014	2016	2018	2020	2022
Technology Node (nm)	45	32	22	16	11	8	6	4
Integration Capacity (BT)	8	16	32	64	128	256	512	1024

CPU production process is approaching the physical limits of the integration scale.

Technological outlook:

Most important chip semiconductor makers are working in order to solve problems related to integration scale. Several deep changes in processors architectures have been applied to the CPU generations of last 10 years, introducing several level of parallelism, from SSE to multi core.

During next years, that evolutive process will continue in a even more deep manner, moving CPU processor architectures to the so called “Many-core” era



Many-core era is characterized by processors containing hundreds-thousands computing cores, permitting thousands HW threads per socket, running massive parallel applications.

Next 15 years and Einstein Telescope computing

- In the manycore era, computing power will be distributed across thousands cores, interconnected by several hierarchical cache levels and complex bus subsystem.
- Express the expected performances of these new architectures require a complete different programming models and algorithms.
- Performances achievable are not a priori predictable, depending on:
  - The specific problem algorithms
  - Memory/registries architecture model
  - Inter communication
  - Serial portion of the algorithm (ref. Amdahl's law)
- Third generation of gravitational wave detectors, like ET, need to deal with this technology in order to address their scientific goals
- Currently, Graphics Processing Unit or GPU are the firsts devices that map the manycore definition with interesting performance characteristics. For that, has been defined the concept of General Purpose computing (GPGPU)

cuInspiral:
GPU CB library prototype

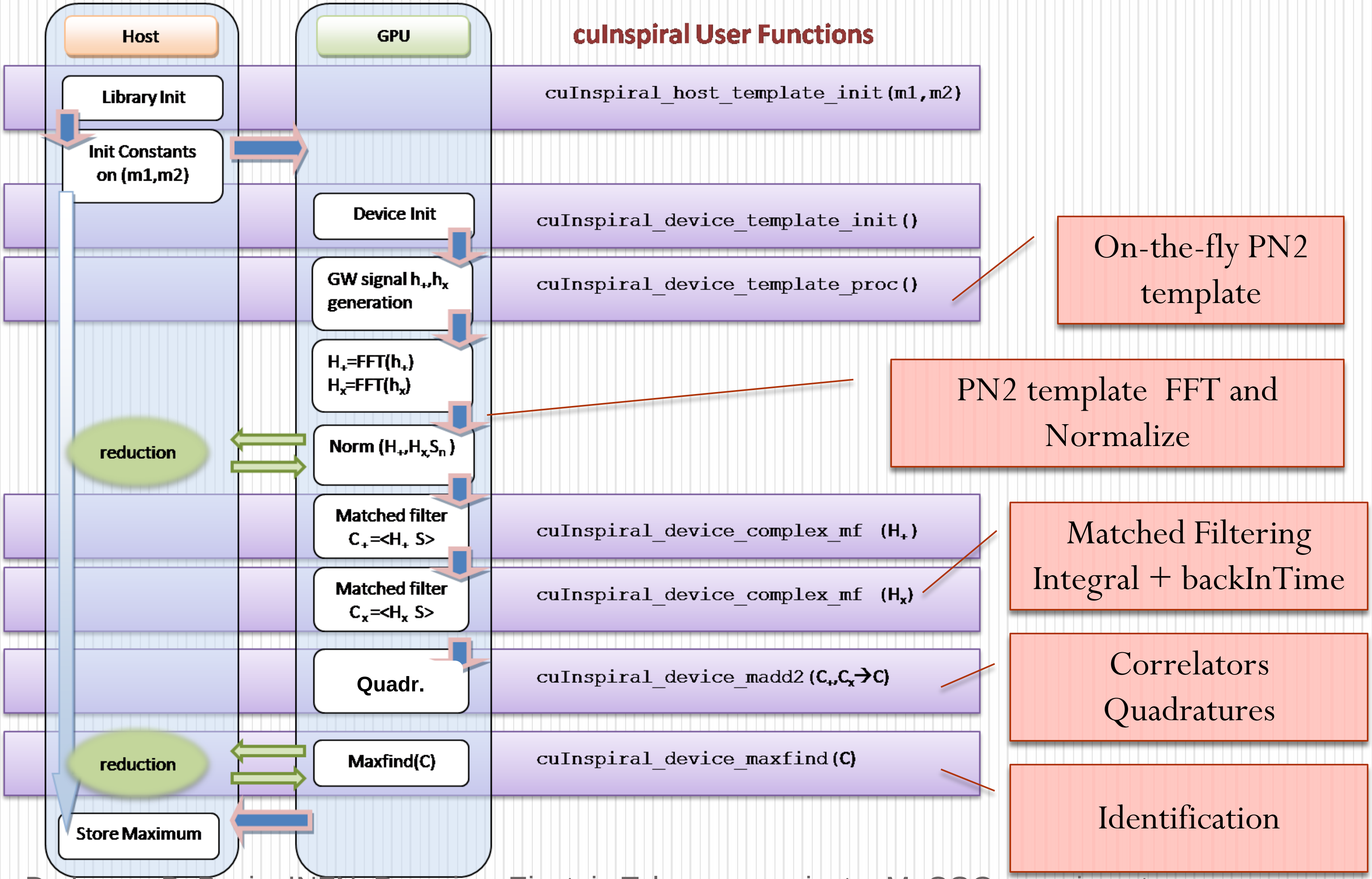
❑ cuInspiral was under develop in Virgo/ET INFN Perugia in 2009, using CUDA framework of NVIDIA, now merged in **MaCGO** (***Manycore Computing for future Gravitational Observatories***) INFN experiment.

❑The first library implementing a high arithmetic intensity strategy on GPU for coalescing binaries DA., where all computation is made completely inside GPU space.

❑Some functions currently implemented:

- Taylor PN2 generator
- Normalization
- Matched filtering
- Maximum identification
- Other complex vector operations

Layout of the cuInspiral CB pipeline



Host

GPU

cuInspiral User Functions

Library Init

`cuInspiral_host_template_init(m1,m2)`

Init Constants on (m1,m2)

Device Init

`cuInspiral_device_template_init()`

GW signal h_+,h_x generation

`cuInspiral_device_template_proc()`

On-the-fly PN2 template

H_+=FFT(h_+)
H_x=FFT(h_x)

PN2 template FFT and Normalize

reduction

Norm (H_+,H_x,S_n)

Matched filter C_+=<H_+ S>

`cuInspiral_device_complex_mf` (H_+)

Matched filter C_x=<H_x S>

`cuInspiral_device_complex_mf` (H_x)

Matched Filtering Integral + backInTime

Quadr.

`cuInspiral_device_madd2` (C_+,C_x→C)

Correlators Quadratures

reduction

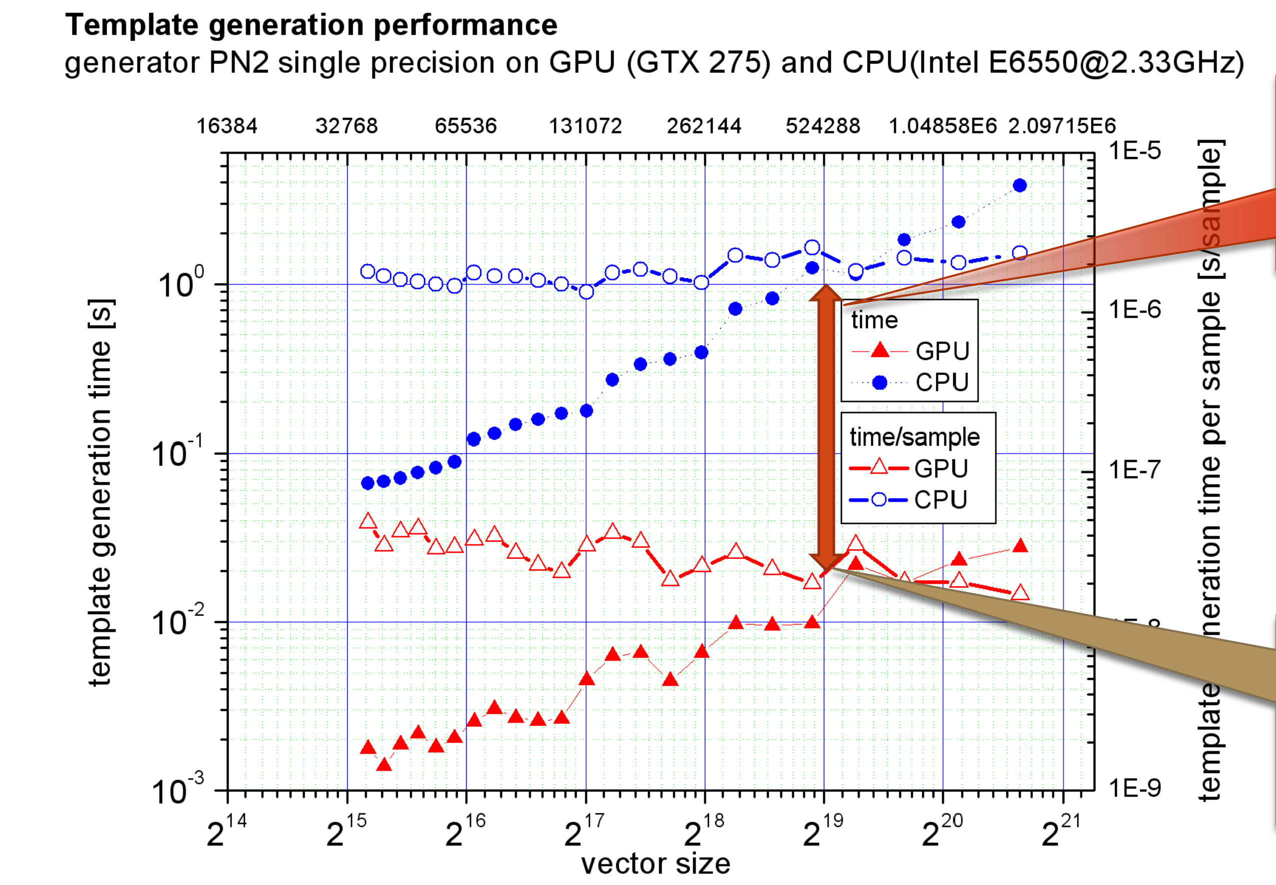
Maxfind(C)

`cuInspiral_device_maxfind` (C)

Identification

Store Maximum

cuInspiral:
PN2 template generation [performance]



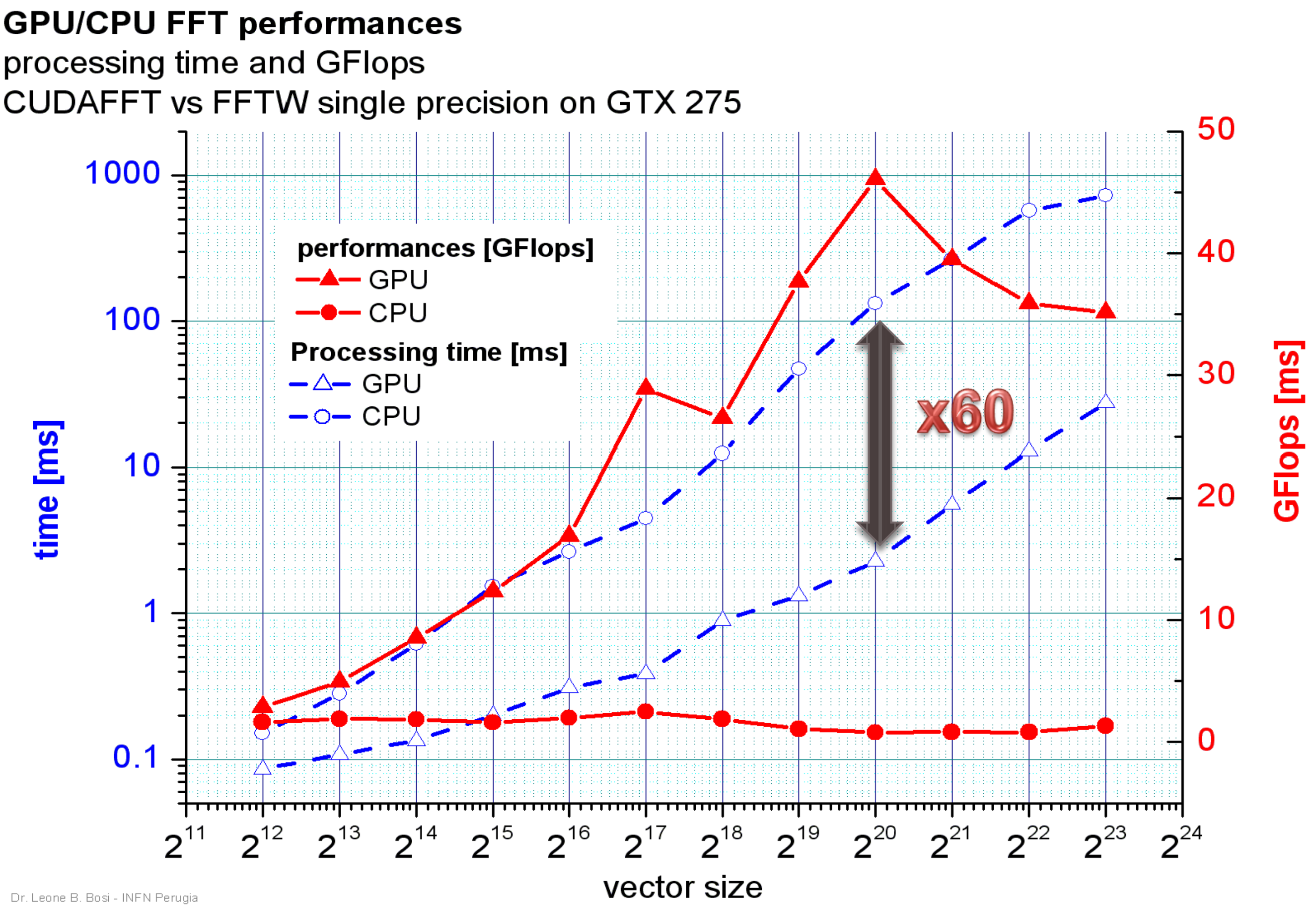
Template generation performance

generator PN2 single precision on GPU (GTX 275) and CPU(Intel E6550@2.33GHz)

gain = X100

2-3 10^{-8} s/sample

CUDAFFT vs FFTW: proc time | GFlops (single precision)

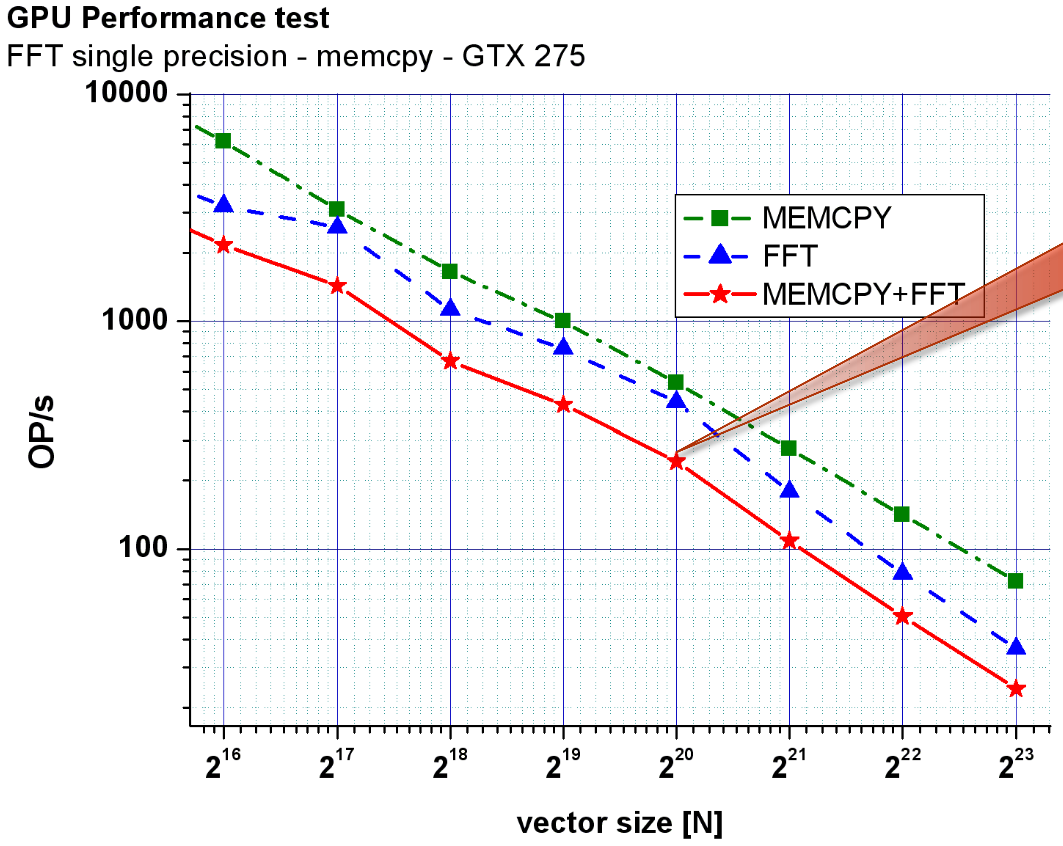


GPU/CPU FFT performances

processing time and GFlops

CUDAFFT vs FFTW single precision on GTX 275

Host←→Device Memory I/O overhead

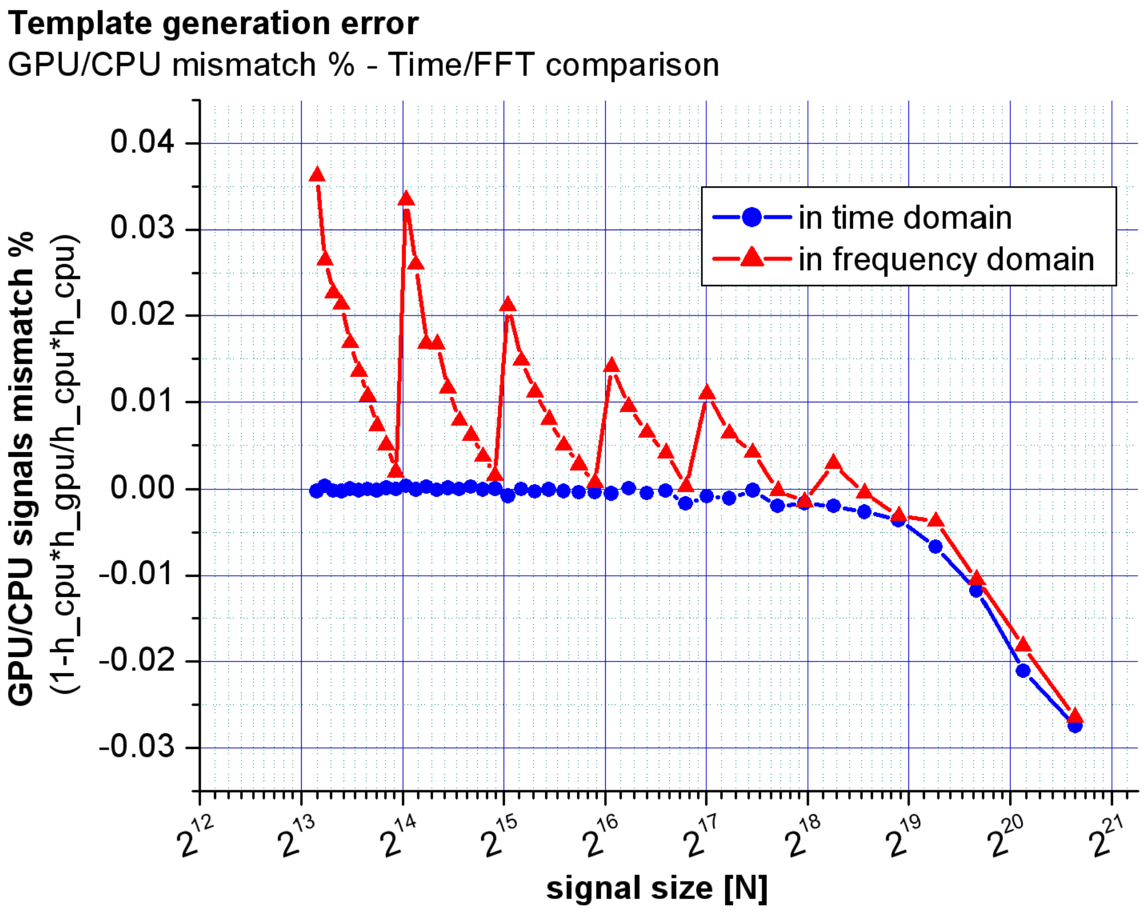


cuFFT wrapping approach looses performances due to memory IO operations

Host-Device memory IO kills performances

GPU express best performances in high arithmetic intensity conditions, as in cuInspiral design.

cuInspiral:
PN2 template and FFT Accuracy



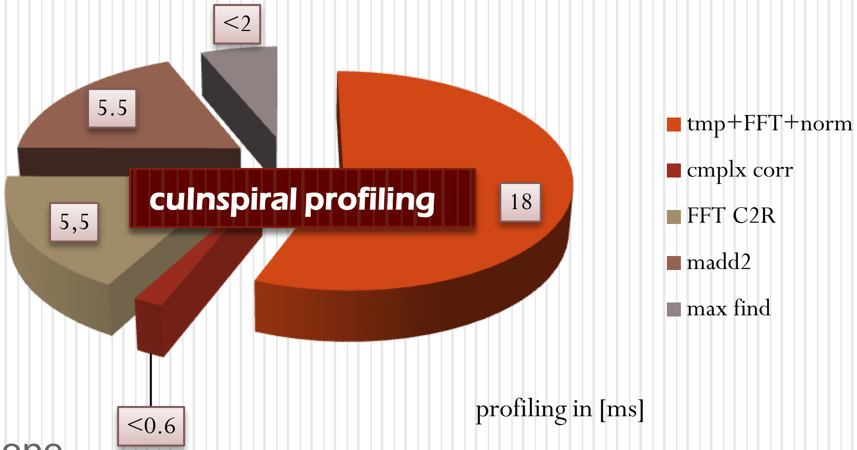
Pipeline profiling:

- If we consider analysis parameters of :
  - low.cutof.freq:24Hz,
  - vector length 2^20, fs=4kHz the
- The cuInspiral processing rate on GTX 275 and Tesla is roughly of:

  30 templates/sec (lower limit)
- If we consider the online constrain processing with 6000 templates, we can estimate detection phase analysis can be performed with a couple of GPU Class GTX280.

Pipeline Gain respect CPU**>x50**
with GTX 275/Tesla
→expected with Fermi GPU:
→**x100-150**

Multi-GPU configuration: x DevN

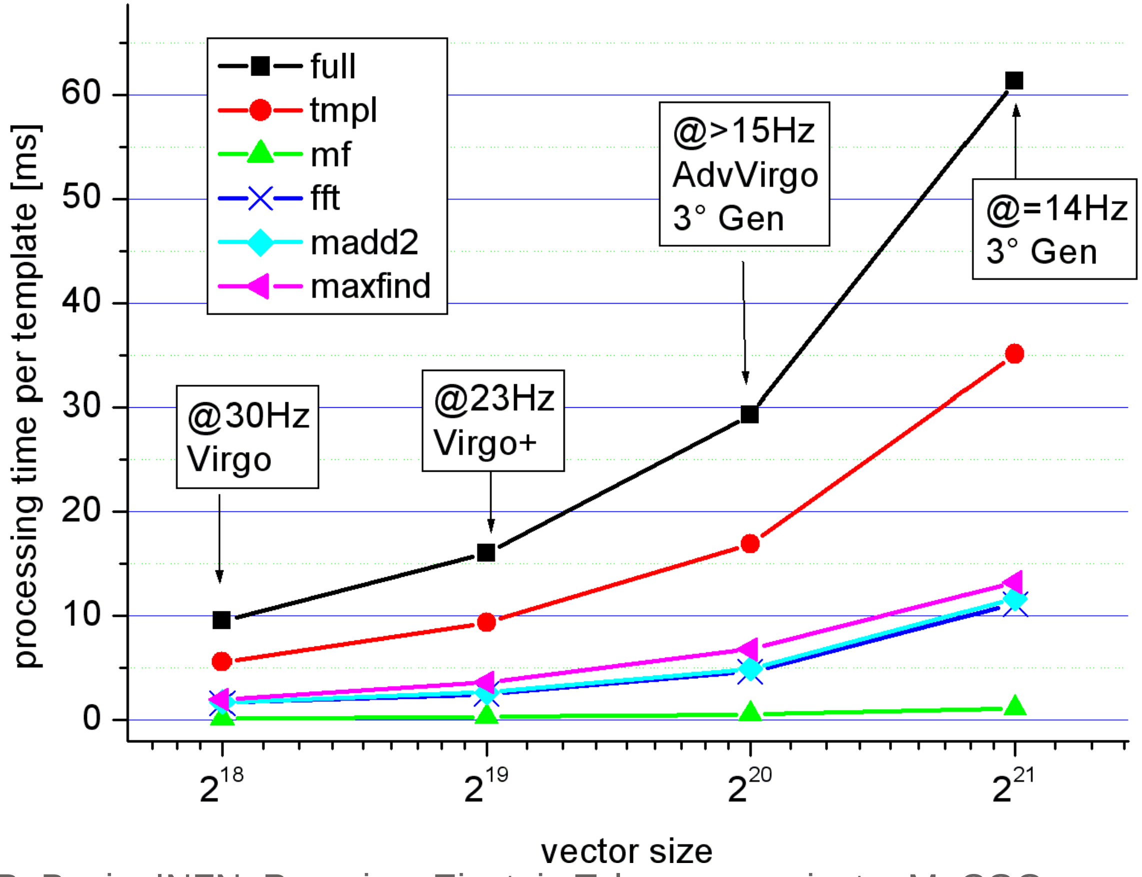


profiling in [ms]

cuInspiral:
pipeline performance vs GW detectors generations

cuInspiral benchmark

processing time per template vs vector size



ET era , computing power requirements and manycore solutions.

- Given a template bank for ET, we can define to truncate critical long inspiral by chosing properly the low frequency cut-off (e.g.4Hz)
- With this choice we can reduce the max template length for this analysis step → 3600s
- **2000000** templates (**length=2hours**@4kHz), today using a cuInspiral GPU like library to process a single timeslice we require: 2000000 x 0.8 s=**18 days, with a normal CPU 1.5 years**!!
- If we renormalize respect to estimated gain factors, such as: FFT code optimization, Moore's Law and loss factor (@2020 forecast) **3x100*0.4=120** we obtain

  →**4 hours to process ET like data**
- It seems plausible that by 2020 computing innovation we will be able to pursue ET requirements for this task.

Conclusions

- **cuInspiral** is a prototype library developed in INFN Perugia to perform and evaluate high arithmetic intensity computation on GPU about CB data analysis.
- This work is in the perspective of the so called "manycore era", important for ET purposes.
- With this version we have reported preliminary performances on a full CB detection pipeline on these new architecture (on nvidia GTX275). In particular:
  - Gain factor **X100** about templates generation respect to identical CPU implementation. (This factor is expected to be roughly constant also for higher PN approximation or others generators).
  - Gain factor **X60** about FFT, using cuFFT library, but in the close future, new cuFFT versions promise to have **X120-180** or more.
  - Number of **X30-35** templates processed per seconds with vector size= 2^20
- It has been reported how the cuInspiral approach permit to obtain very impressive gain from these manycore architectures.