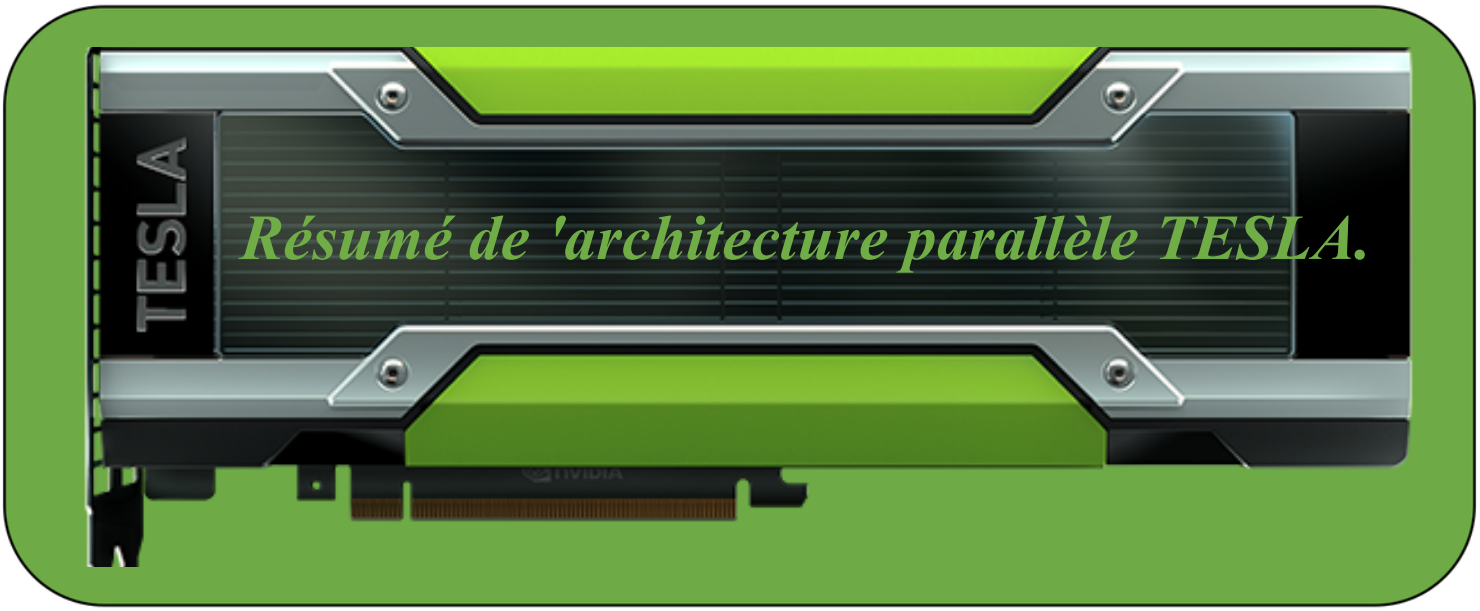
République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université des Sciences et de la Technologie
- Houari Boumediene –
Faculté d'Informatique

Département Des Systèmes Informatique



Résumé de 'architecture parallèle TESLA.

Module :*Programmation multi-cœurs.* ***Niveau :*** *Master 2 HPC.*

Préparer par:

BOULAHIA Yasmine.

ZERKOUK K haoula.

SAIDANI Cerine.

Enseignant:

Mr.H.SAADI.

Année universitaire : 2025/2026

Résume de l’architecture NVIDIA Tesla 2007 :

L’architecture NVIDIA Tesla, introduite en 2007 avec le GPU G80, constitue la première architecture graphique explicitement conçue pour le calcul haute performance (HPC) et le calcul général sur GPU (GPGPU). Elle marque une rupture majeure avec les GPU préexistants en abandonnant le pipeline graphique fixe au profit d’une architecture unifiée, massivement parallèle et programmable via le modèle CUDA.

Sur le plan matériel, Tesla repose sur une organisation hiérarchique composée de **Streaming Multiprocessors (SM)**. Le GPU G80 intègre **16 SM**, chacun contenant **8 unités de calcul scalaires (Streaming Processors, SP)**, pour un total de **128 cœurs CUDA**. Chaque SM dispose d’un ordonnanceur matériel, d’unités arithmétiques (ALU), d’unités de fonctions spéciales (SFU), d’un fichier de registres massif (≈ 8 000 registres) et d’une **mémoire partagée de 16 KB**, essentielle pour la coopération rapide entre threads. Cette architecture permet l’exécution simultanée de plusieurs milliers de threads afin de masquer les latences mémoire.

Tesla adopte un modèle d’exécution **SIMT (Single Instruction, Multiple Threads)**, dans lequel les threads sont regroupés en **warps de 32 threads** exécutant la même instruction de manière synchrone. Bien que le matériel fonctionne selon un paradigme SIMD, le modèle CUDA expose une abstraction de threads indépendants, simplifiant la programmation parallèle. En cas de divergence de contrôle à l’intérieur d’un warp (branchements conditionnels), l’exécution devient séquentielle par masquage, ce qui peut dégrader les performances pour les algorithmes irréguliers.

La hiérarchie mémoire de Tesla est volontairement simple mais exigeante en termes d’optimisation. La **mémoire globale**, de grande capacité (jusqu’à ~768 MB), présente une latence élevée et n’est pas cachée dans cette première génération. Les performances reposent donc sur des **accès mémoire coalescés** et une utilisation intensive de la mémoire partagée. Tesla ne dispose ni de cache L1 ni de cache L2, ce qui constitue une limite importante pour les applications à accès mémoire non réguliers. Des mémoires spécialisées (constante et texture) sont également disponibles pour optimiser certains schémas d’accès.

En termes de performances, le GPU G80 atteint environ **345 GFLOPS en simple précision (FP32)**, offrant une accélération significative (×10 à ×40) par rapport aux CPU contemporains pour les applications massivement parallèles telles que l’algèbre linéaire, la simulation numérique, le traitement d’images et les premières approches d’apprentissage automatique. En revanche, Tesla souffre d’une **faible efficacité en double précision (FP64)** et d’une dépendance forte aux transferts CPU–GPU via PCIe, limitant certaines applications HPC critiques.

L’impact de Tesla est fondamental : elle établit les bases architecturales du calcul GPU moderne. Les concepts de SM, warps, hiérarchie mémoire explicite et programmation CUDA seront repris et améliorés dans les générations suivantes (Fermi, Kepler, Pascal, Volta). Tesla ne représente pas seulement une évolution technologique, mais le point de départ d’un nouveau paradigme de calcul parallèle qui domine aujourd’hui le HPC et l’intelligence artificielle.