Использование архитектуры TESLA при обработке экспериментальных данных диагностики ИТЭР ВНК

Скопинцев Д.А., Семенов И.Б., Портоне С.С., Кузьминов С.А., Ларионов А.С.

Частное учреждение ГК Росатом «Проектный центр ИТЭР», г. Москва, Россия.

e-mail: d.scopintcev@iterrf.ru, i.semenov@iterrf.ru, s.portone@iterrf.ru, s.kuzminov@iterrf.ru, a.larionov@iterrf.ru

В докладе представлена предлагаемая конфигурация предварительной обработки данных разрабатываемой диагностики ИТЭР Вертикальная Нейтронная Камера на основе локального кластера Nvidia TESLA S1070 имеющего 4 вычислительных ядра.

Вертикальная Нейтронная Камера должна обеспечивать измерение интенсивности потоков и фиксацию спектров быстрых нейтронов от 48 измерительных трактов каждые 10 миллисекунд в течение всего разряда. Длина посылки составляет 1044 слова (unsigned32), итого 16 Гбит / с.

Конфигурация предназначена для сжатия данных и уменьшения потока передаваемого в систему CODAC до ширины полосы пропускания сетей ИТЭР ~ 10 Гбит / с.

Вычислительное ядро GT200B имеет индекс архитектуры 1.3 и работает с 64-разрядными числами двойной точности по стандарту IEEE 754. Поддерживает асинхронную передачу данных для заполнения 4 ГБ собственной памяти. 512-разрядный интерфейс **о**беспечивает пропускную способность памяти GDDR3 до 100 ГБ/с. Передача данных с промышленного PC осуществляется через MXI интерфейс шины PCIe 2.0 с пропускной способностью до 6,4 ГБ/с. Один интерфейс обслуживает два ядра. 240 универсальных процессоров собраны в 10 потоковых мультипроцессоров.

Загрузкой мультипроцессоров занимается драйвер (Release 331 Quadro, NVS, Tesla, & GRID Driver Version 332.21 | January 6, 2014). Он поддерживает следующие API: Open Computing Language (OpenCL) 1.1; OpenGL 4.4; DirectX 11 и CUDA 5.5 [1], среду программирования интегрированную в Visual Studio Professional 2012.

Алгоритмы обработки данных организуются в параллельные потоки (thread), которые объединяются по 32 в группы (warp) выполняющиеся одновременно. Эти группы собираются в блоки (block), которые заполняют сетки (grid) [2]. Каждый блок выполняется на одном мультипроцессоре.

Непрерывный поток входных данных разбивается на окна, каждое из которых обрабатывается по одному и тому же алгоритму, но с разными входными данными параллельно. Тем самым достигается ускорение процесса обработки. Проведённые тесты показали, что ускорение времени обработки, включая передачу данных по шинам, составляет не менее одного порядка. Пропускная способность шин справляется с расчётными загрузками.

Литература.

1. CUDA Compute Unified Device Architecture: CUDA 5.5 Programming Guide // NVIDIA Corporation, **2014**.
2. Параллельные вычисления на GPU. Архитектура и программная модель CUDА: Учеб.пособие / А.В.Боресков и др. Предисл.: В. А. Садовничий. - М.: Издательство Московского университета, **2012**.