



Программная реализация разделения гамма-квантов и нейтронов по форме импульса в режиме реального времени на платформе CUDA

*Колбасин В.А., Иванов А.И., Педаш В.Ю.,
НАНУ ИСМА*



Раздельное детектирование нейтронов и гамма-квантов

- От систем радиационной безопасности требуется, чтобы они по-разному реагировали на гамма и нейтронное излучение

- Гамма-излучение проверить уровень и спектр

- Нейтронное – сразу выдавать сигнал тревоги

- Для этого обычно используют два детектора: один чувствителен к гамма-излучению, а второй – к нейтронному

- для детектирования нейтронов хорошо подходят детекторы на основе He^3 , но он крайне дефицитен

- Нейтроны детектируют и некоторые сцинтилляционные материалы (например, стильбен)

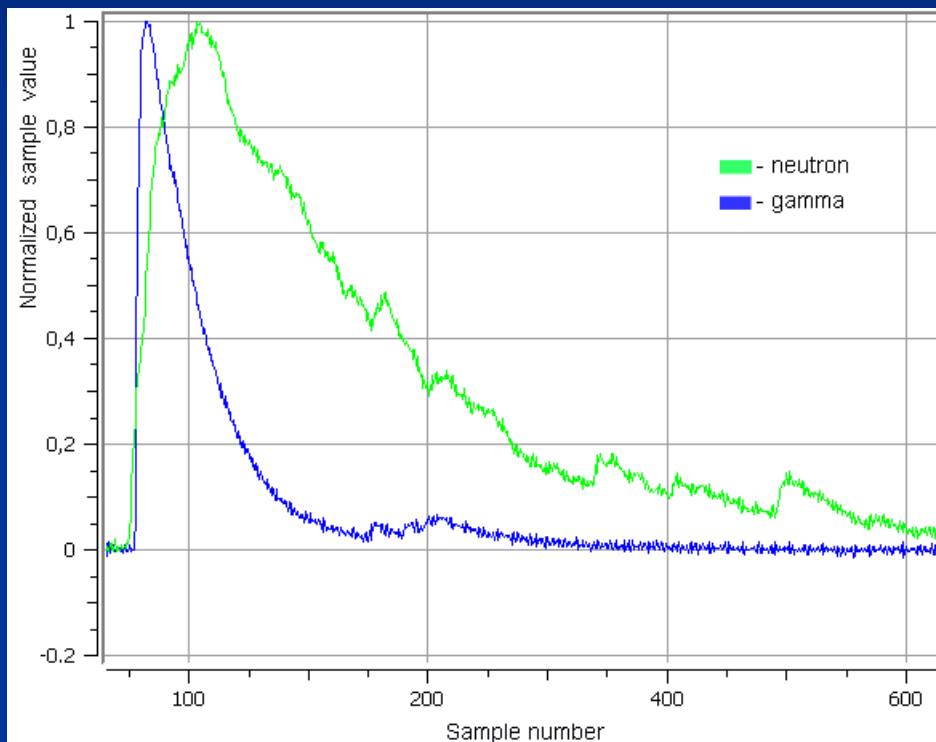
- Но они реагируют и на гамма-кванты

- Форма сцинтилляционных импульсов отличается для гамма-квантов и нейтронов – по ней можно определить, что вызвало срабатывание

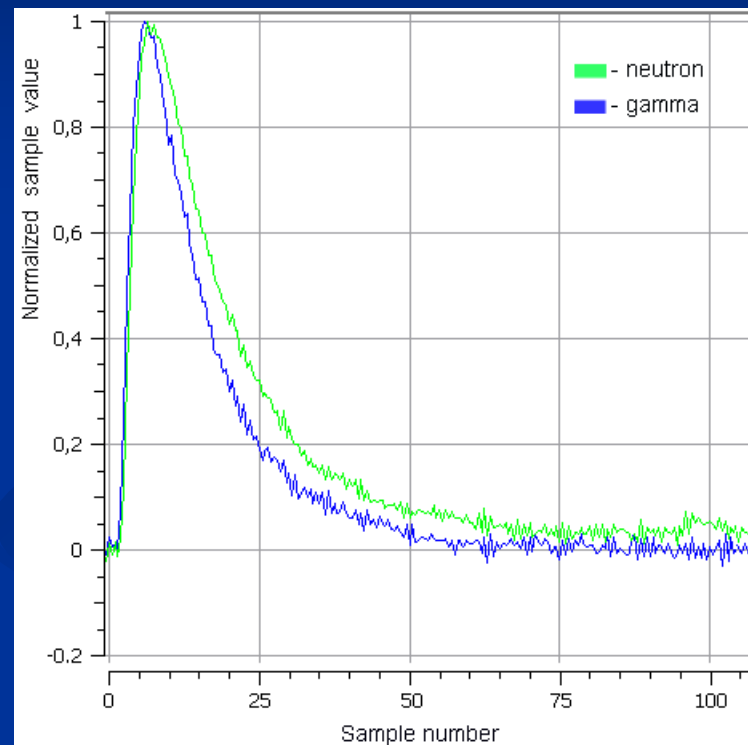


Форма сцинтилляционных импульсов

LiF



Стильбен



Здесь все распознается

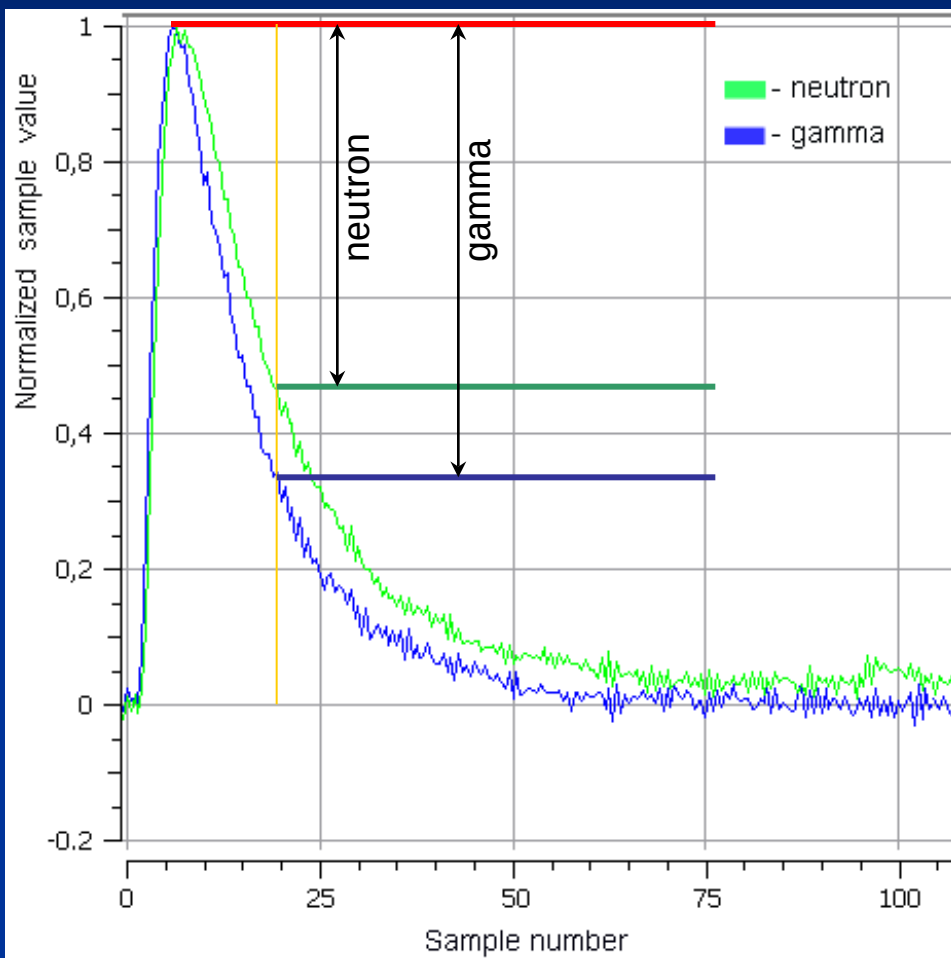
Но какова максимальная скорость счета ?

А как отличить эти импульсы ?

А что будет при повышении
уровня шума ?



Pulse Gradient Analysis (PGA)



- Для различия используется значение:

$$d = \frac{s_{peak+offset}}{s_{peak}}$$

- если d меньше порога – гамма-квант, больше – нейтрон

+ Очень простой метод

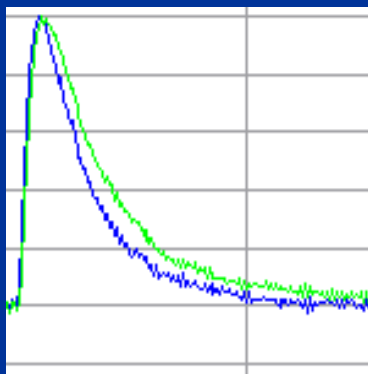
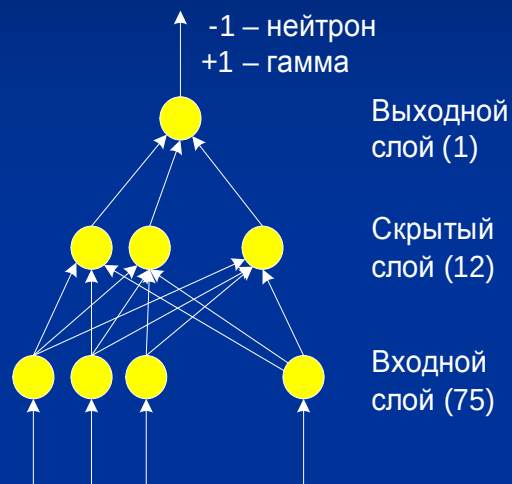
— Чувствительный к шуму

✓ Вместо второго отсчета возьмем среднее:

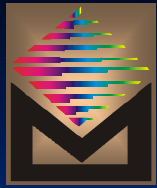
$$d = \frac{\frac{1}{2N+1} \cdot \sum_{i=-N}^N s_{peak+offset+i}}{s_{peak}}$$



Метод на основе нейронной сети



- Для различения типа частицы используется трехслойный перцептрон
- + Низкая чувствительность к шуму
- Перед использованием нейронную сеть надо тренировать
- Высокая вычислительная сложность
- ✓ Для программной реализации можно использовать параллельные вычисления



Подходы к реализации алгоритмов различения частиц по форме импульсов

■ Аппаратная реализация

- ✓ Высокая скорость обработки данных
- Можно реализовать только самые простые алгоритмы
- Нет возможности изменить используемый алгоритм, если этого требуют обстоятельства

■ Программная реализация

- ✓ Можно менять алгоритм
- ✓ Можно использовать сложные схемы распознавания
- На типовых вычислительных устройствах сложно обработать программно поток данных в 10^6 импульсов в секунду (~ 120 МБ/сек)

➤ Для программной реализации подходят технологии массовых параллельных вычислений CUDA и Open CL



Схема модуля разделения нейтронов/гамма-квантов

Детектор

- Стильбен

АЦП

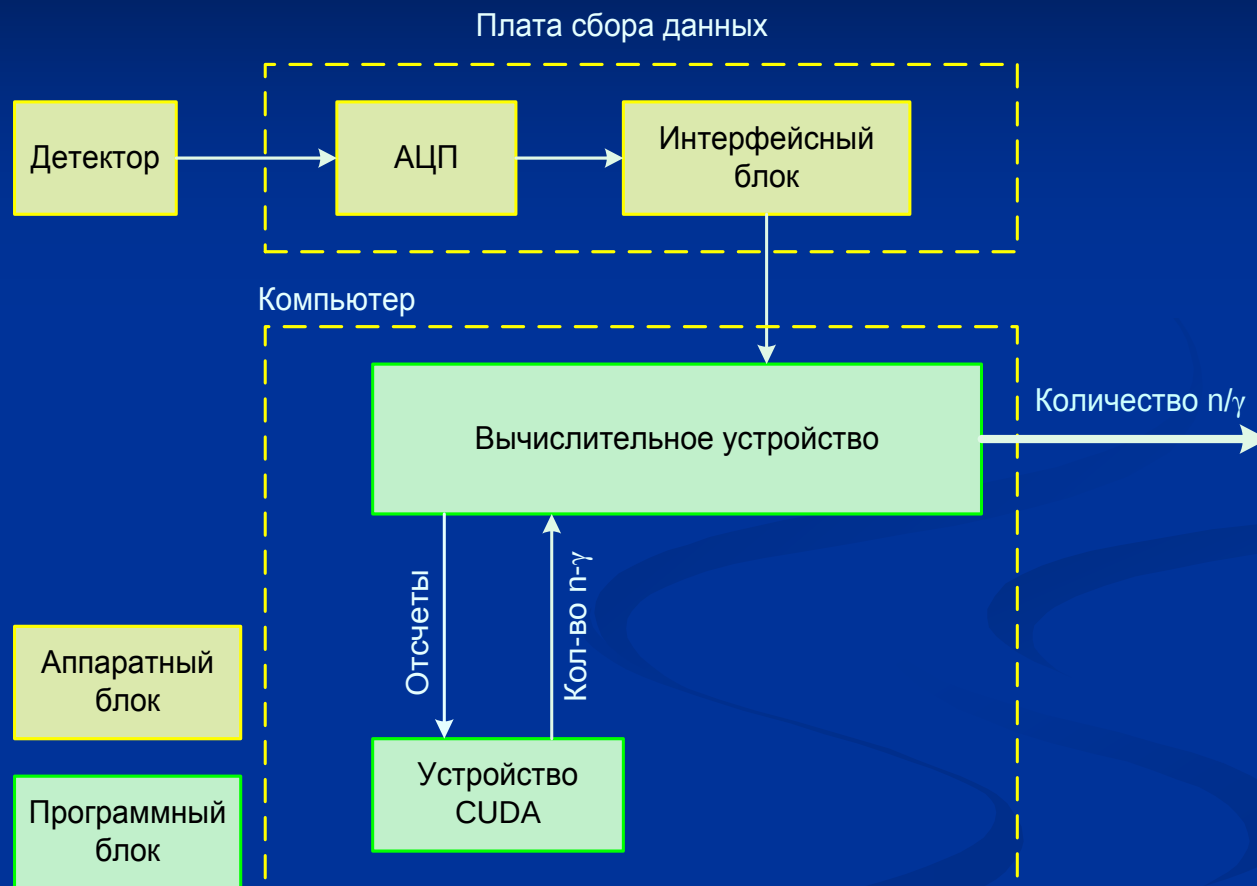
- 10 бит
- 100 MHz

CPU

- Core 2 Duo
- 3,2 MHz

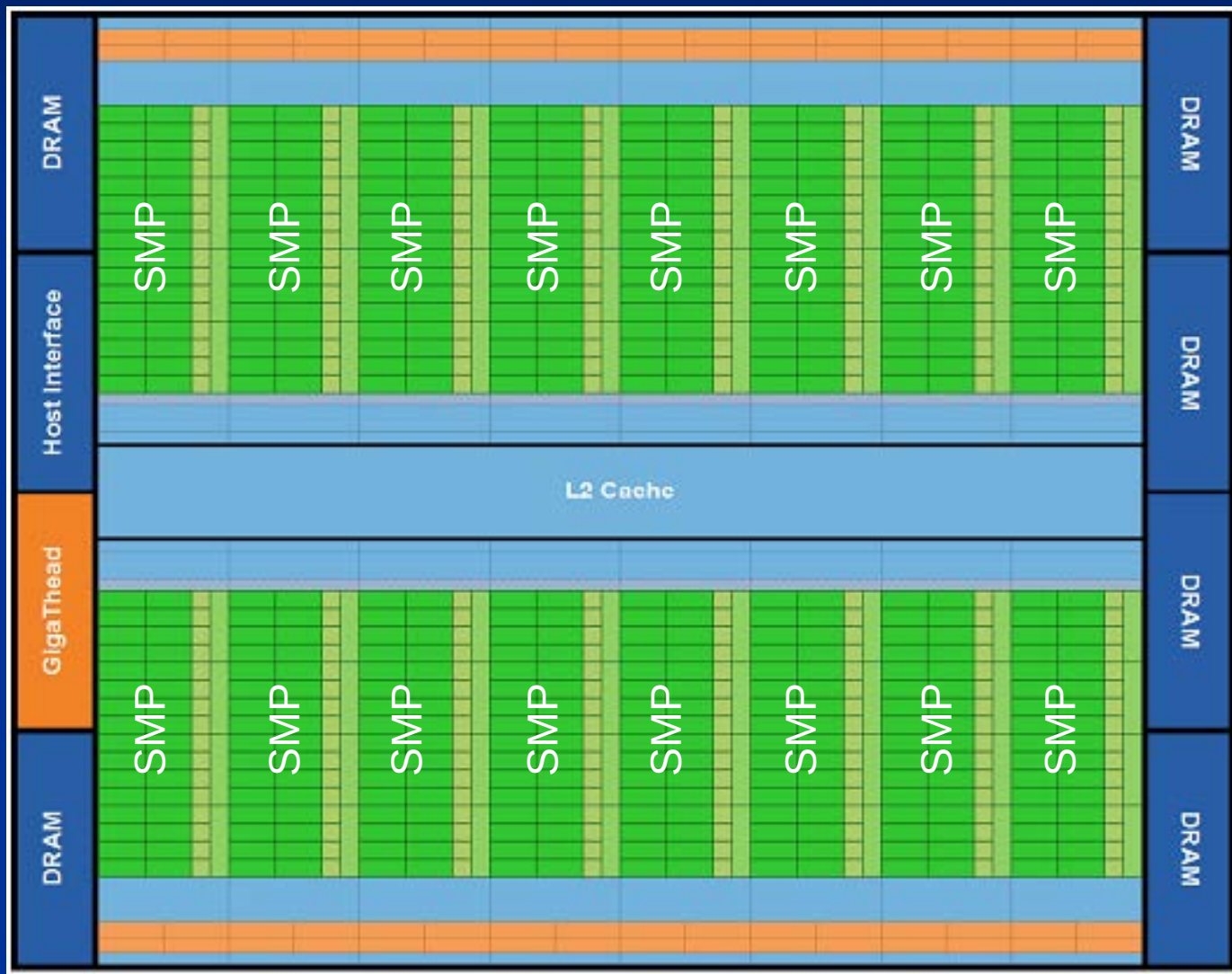
GPU

- GeForce 560 Ti



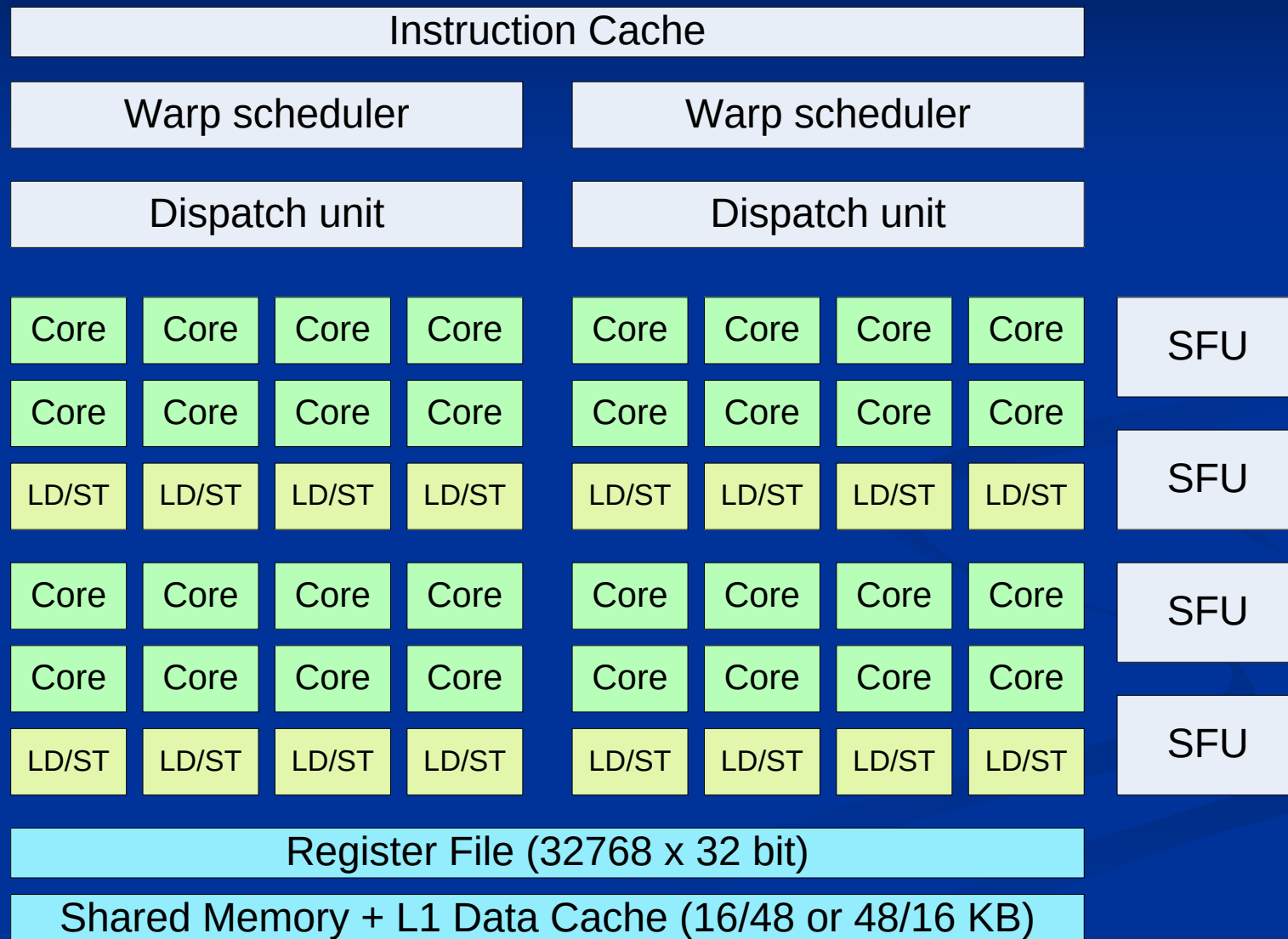


Платформа CUDA (1)





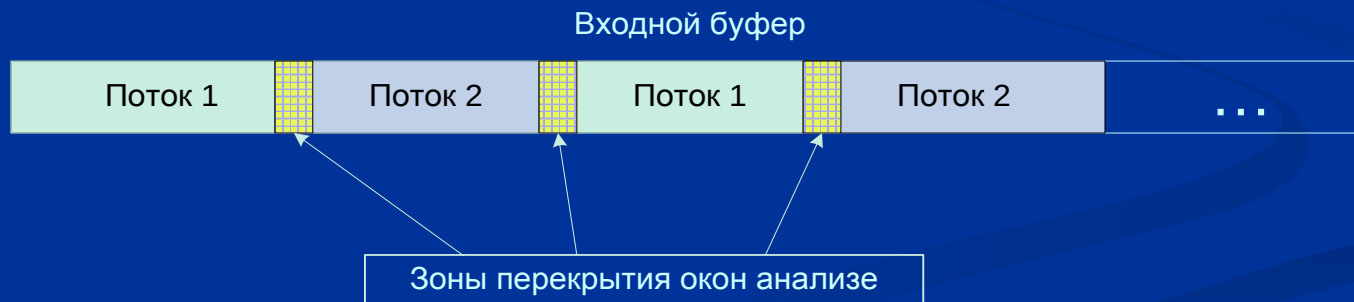
Потоковый мультипроцессор

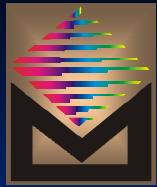




Параллельная реализация метода PGA

- Чтобы вычислить какое-то значение требуется нужна лишь небольшая часть данных импульса
 - ➔ Для обработки на CUDA-устройстве надо данные скопировать в память устройства
 - ➔ За счет этого весь прирост производительности будет потерян
 - Алгоритм реализуется многопоточно на центральном процессоре
 - Данные обрабатываются по схеме:





Параллельная реализация метода на основе нейронной сети

- Чтобы достичь полной загрузки потокового мультипроцессора (SMP) надо чтобы он выполнял не менее 32 потоков
 - 12 SMP → надо использовать больше чем 384 потоков
 - Но окно анализа содержит только 75 отсчетов →
 - Распараллеливаем по принципу: один SMP – одно окно анализа
- Все отсчеты одного окна используются много раз
 - Будем их хранить в разделяемой памяти
- Одни и те же весовые коэффициенты нейронов используются при обработке всех окон анализа
 - Будем их хранить в разделяемой памяти
- Копирование данных в память карты занимает ощутимое время
 - Используем для этого CUDA streams



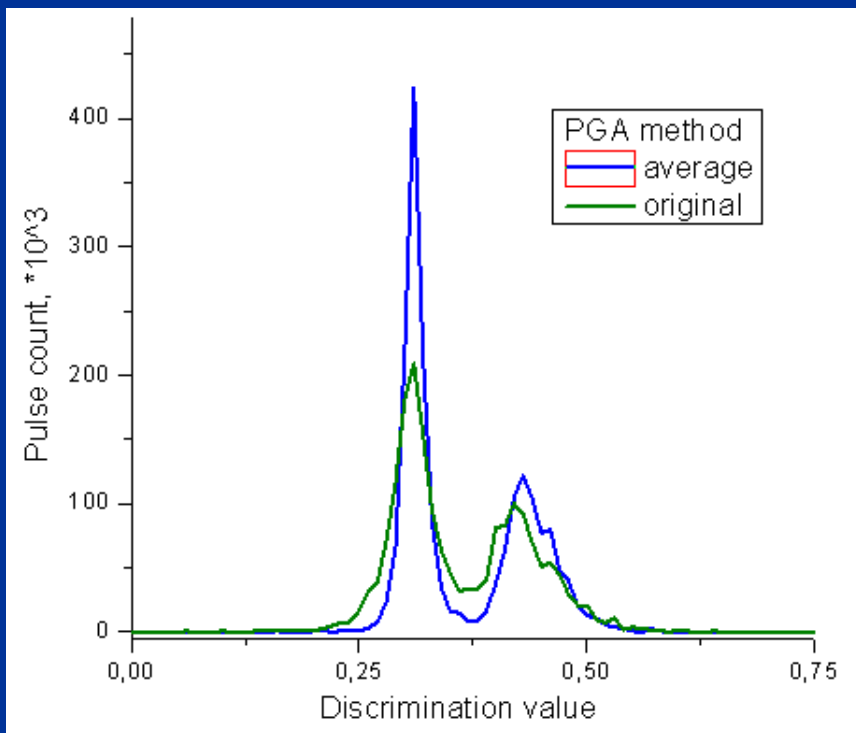
Вычислительная производительность

- Метод Pulse Gradient Analysis
 - Процессор: Core 2 Duo 3 GHz
 - Пиковая скорость обработки: $1.5 \cdot 10^6$ импульсов в секунду
- Метод на основе нейросети (CPU)
 - Процессор: Core 2 Duo 3 GHz
 - Пиковая скорость обработки : $0.07 \cdot 10^6$ импульсов в секунду
- Метод на основе нейросети (CUDA)
 - Процессор: GeForce 560 Ti
 - Пиковая скорость обработки : $1.4 \cdot 10^6$ импульсов в секунду

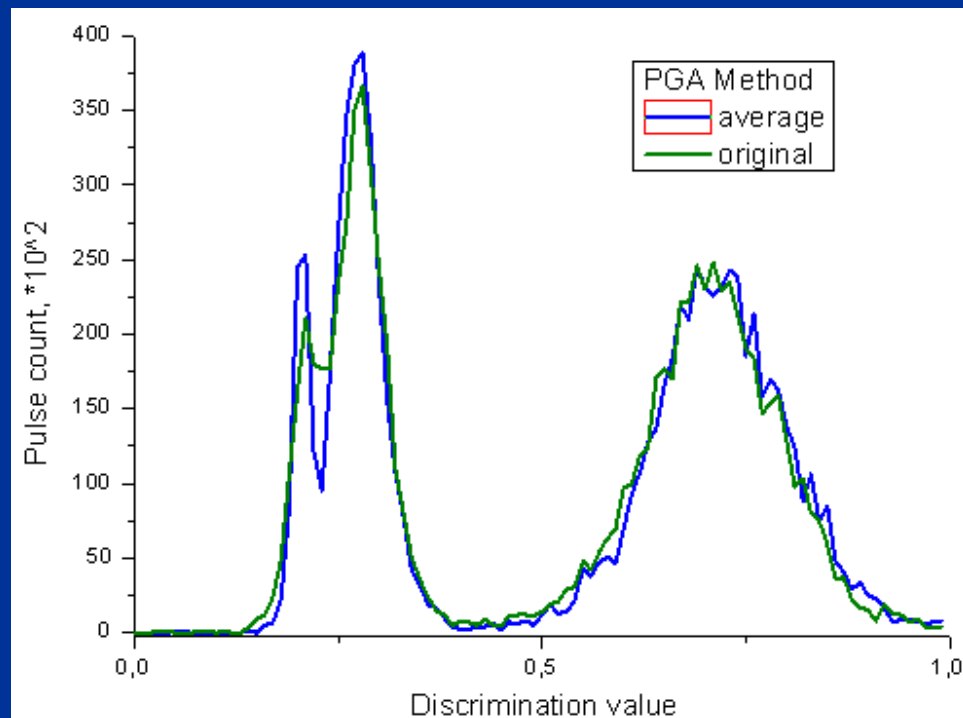


Результаты применения метод PGA для стильбена и LiF

- Стильбен
- Источник PuBe
- Параметр PGA *offset* = 30



- LiF
- Источник PuBe
- Параметр PGA *offset* = 60

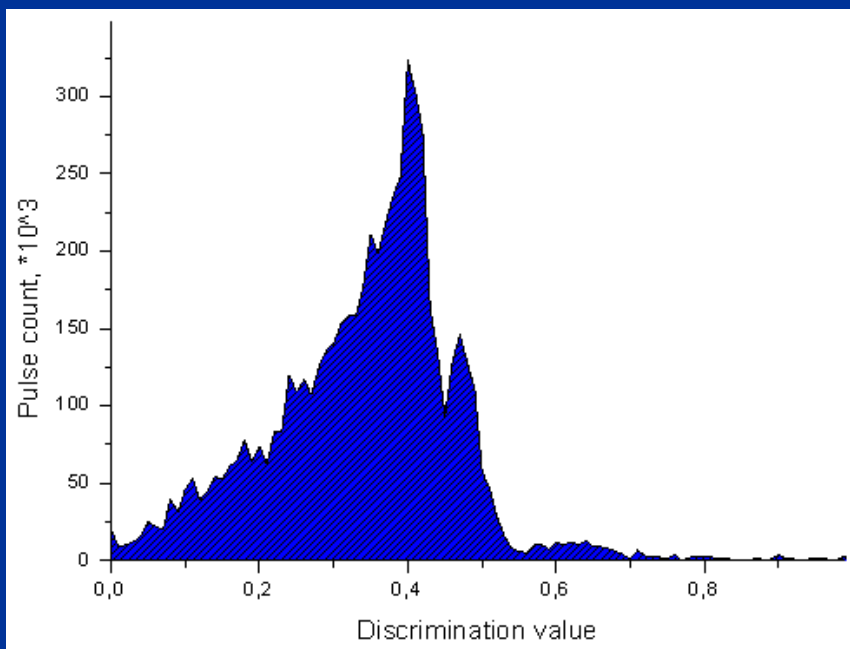




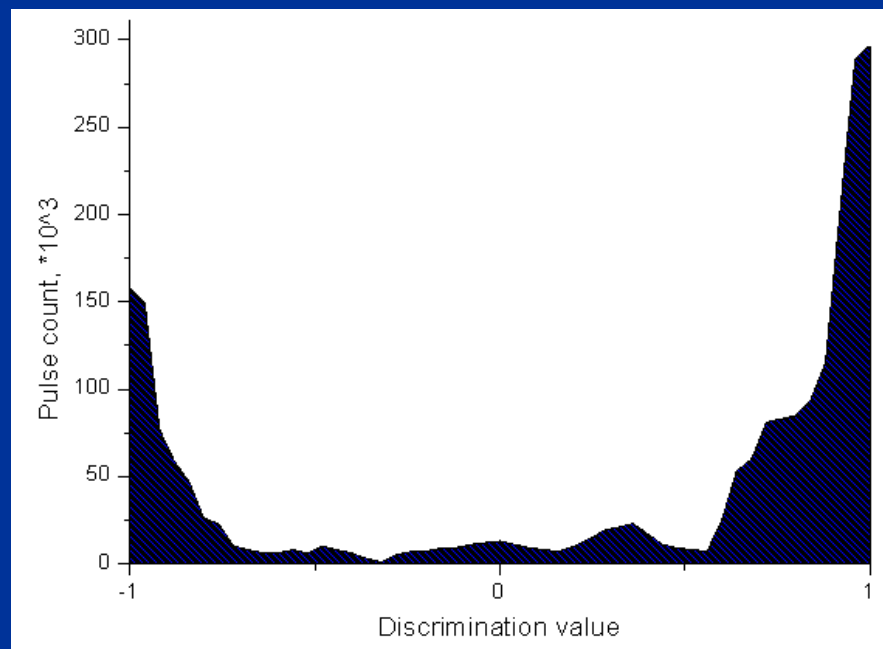
Результаты применения метода на основе нейронной сети для стильбена

- Стильбен, кристалл хуже качества, выше уровень помех
- Источник PuBe

Метод PGA



Метод на основе нейронной сети





Выводы

- При программной реализации метода Pulse Gradient Analysis обеспечивается скорость обработки данных порядка $1,5 \cdot 10^6$ импульсов в секунду
 - Дополнительный аппаратные ускорители не требуются
- При программной реализации метода на основе нейронной сети на платформе CUDA обеспечивается скорость обработки порядка $1.4 \cdot 10^6$ импульсов в секунду
 - Требуется одна карта CUDA среднего ценового диапазона (порядка 250– 300 \$)
- Предложенные программные реализации обеспечивают скорость обработки данных, необходимую для задач радиационной безопасности