

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего
профессионального образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Институт – Физико-технический
Направление – Ядерная физика и технологии
Кафедра – Прикладная физика
Специальность – Физика ядра и частиц

Отчет по летней практике

Калибровка вето счетчиков

Выполнили: Егоя А.С.
Огребо А.В.

Руководитель: Хохлов Ю.А.

Томск, 2014

Содержание

1	Теоретическая часть	3
1.1	Ионизация	3
1.2	Формула Бете-Блоха	3
1.3	Распределение Ландау	4
1.4	Сцинтилляционные детекторы	5
1.5	ФЭУ	5
1.6	Разрешение ФЭУ	5
1.7	Вето счетчики	6
1.8	Логическая схема	6
1.9	Пьедестальные значения спектров	7
1.10	Осциллограф	7
2	Практическая часть	8
2.1	Детектирование космических мюонов	8
2.2	Измерения с LED	11
2.3	Обработка экспериментальных данных	13
2.4	Измерение порога срабатывания каналов схемы антисовпадений с помощью сигналов космических мюонов	16
3	Приложение	20
3.1	Журнал лабораторных наблюдений	20
3.2	Спектры амплитуд космических мюонов	21
3.3	Листинги программ	23
3.3.1	Гаусс	23
3.3.2	Ландау	25
3.3.3	ЛанГаусс	27

Введение

Институт Физики Высоких Энергий в г. Протвино занимается широким спектром задач связанным с физикой элементарных частиц. В институте имеется ускоритель заряженных частиц "У-70" на энергию до 70 ГэВ. После достижения требуемой энергии частицы попадают на различные установки для экспериментов. В частности, наша практика проходила на установке ВЕС(Вершинный Спектрометр). Команда работающая на установке в данный момент занимается проектом по созданию активной мишени для адронов. Наша работа была связана с вето счетчиками. Это детекторы сцинтилляционного типа, данные с которых поступают на логические схемы совпадений, с помощью которых определяются сигналы, которые не нужно учитывать в ходе эксперимента. При плановой замене вето счетчиков выяснилось, что отсутствуют конкретные данные о параметрах счетчиков, таких как чувствительность, качество и однородность вещества сцинтиллятора.

Постановка задачи

1. Калибровка вето счетчиков космическими мюонами
2. Определение вклада статистики фотоэлектронов в мюонный пик с помощью измерений с LED.
3. Определение порога чувствительности схемы антисовпадений триггерных счетчиков с помощью космических мюонов.

1 Теоретическая часть

1.1 Ионизация

Ионизационные потери - потери частицы на столкновения с атомными электронами при движении в веществе. Частица испытывает кулоновское взаимодействие с электронами и теряет часть энергии на возбуждение и ионизацию атомов среды. Для тяжелых заряженных частиц это является основным механизмом потери энергии.

1.2 Формула Бете-Блоха

Удельные ионизационные потери частицы описываются формулой Бете-Блоха, которая является функцией скорости частицы.

$$\frac{dE}{dx} = \frac{4\pi N_A Z z^2 e^4}{m_e c^2 \beta^2 A} \left[\ln \frac{2m_e c^2 \beta^2}{I(1 - \beta^2)} - \beta^2 \right] \quad (1)$$

Здесь Z и A - порядковый номер и массовое число вещества, I - средний потенциал ионизации, z - заряд (в единицах e) ионизирующей частицы, β - её скорость (в единицах скорости света), e и m_e - заряд и масса электрона. Основные закономерности формулы Бёте-Блоха:

1. Потери не зависят от массы, квадратично зависят от заряда частицы
2. Подходит для всех частиц в области малых скоростей $\beta \ll 1$ (но больших по сравнению с "внутриатомными скоростями" электронов) изменяет как $\frac{1}{\beta^2}$. При $\gamma = 34$ функция достигает минимума, а частица называется минимально ионизирующей.
3. Зависимость потерь от среды определяется отношением Z/A близким к $1/2$ для большинства веществ. Влияние среднего потенциала ионизации I , стоящего под логарифмом мало.

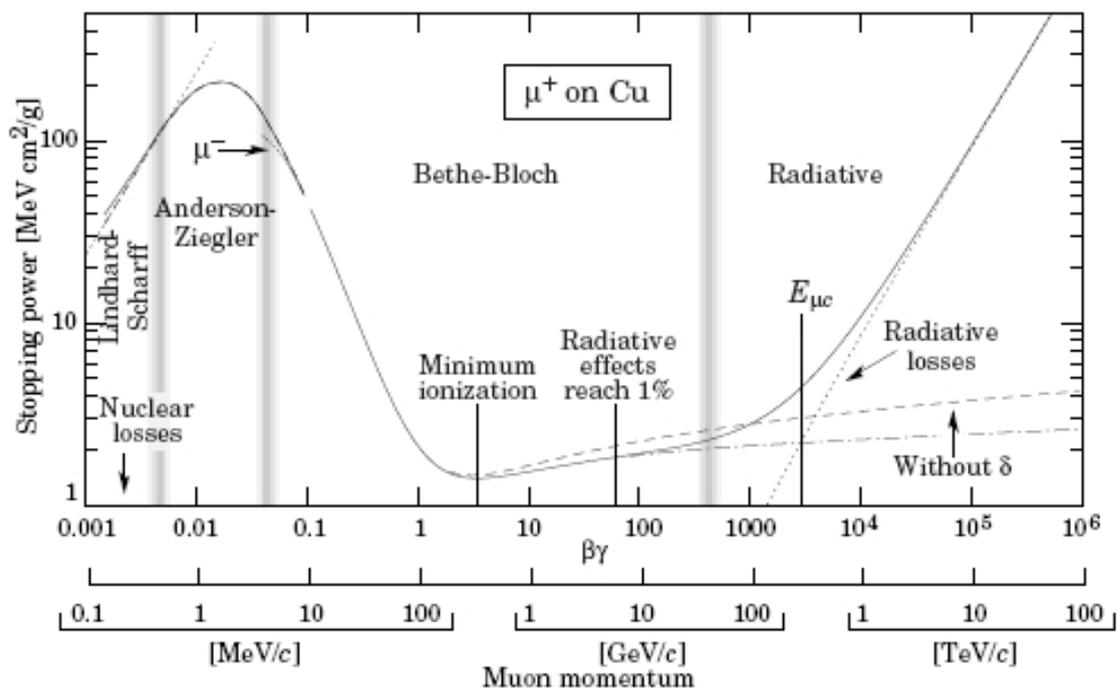


Рис. 1: Потери энергии мюонами в медном поглотителе

1.3 Распределение Ландау

Формула Бете-Блоха описывает средние потери энергии, а распределение потерь энергии вокруг среднего значения описываются распределением Ландау. Данное распределение описывает флуктуации ионизационных потерь заряженных частиц только для тонких мишеней.

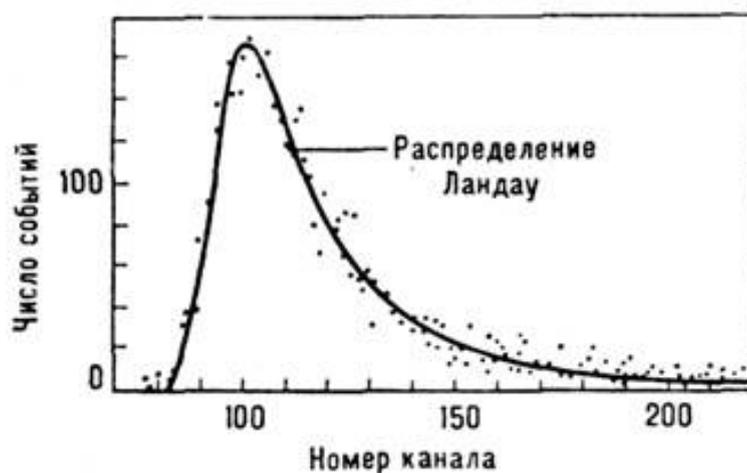


Рис. 2: Вид распределения Ландау

1.4 Сцинтилляционные детекторы

Сцинтилляция - свечение в веществе, возникающее в результате ионизации и возбуждения атомов вещества, частицей проходящей через сцинтиллирующее вещество. В качестве сцинтиллятора используются различные кристаллы, пластик, жидкости. В сцинтилляционном детекторе свет, излученный при сцинтилляции, собирается на фотоприёмнике (как правило, это фотокатод фотоэлектронного умножителя — ФЭУ, значительно реже используются фотодиоды и другие фотоприёмники), преобразуется в импульс тока, усиливается и записывается той или иной регистрирующей системой.

1.5 ФЭУ

ФЭУ(фотоэлектронный умножитель) - устройство для конвертации фотонов в электронные импульсы. Состоит из фотокатода - фоточувствительного элемента, испускающего, за счет фотоэффекта, электроны со своей поверхности, диодной системы, размножающей первичные электроны вылетевшие из фотокатода, за счет вторичной эмиссии электронов. В нашем распоряжении оказались детекторы с пластиковыми сцинтилляторами и ФЭУ-85.

1.6 Разрешение ФЭУ

Рассмотрим разрешение сцинтилляционного детектора в импульсном режиме. Пусть сцинтилляционным детектором регистрируются акты полного поглощения ионизирующих частиц фиксированной энергии. Полный заряд Q в анодном импульсе ФЭУ представляется в упрощенном виде как

$$\bar{Q} = \bar{N}_\gamma \bar{p} \bar{M} \quad (2)$$

В дальнейшем знак усреднения будем опускать. Здесь N_γ — число фотонов в сцинтилляционной вспышке, M — коэффициент усиления ФЭУ, p — эффективность передачи фотонов, равная отношению числа первичных электронов, возникших на входе ФЭУ, к числу фотонов: $p = N_{el}/N_\gamma$. По формуле распространения ошибок дисперсия(в этом разделе все дисперсии подразумеваются относительными) заряда просто складываются из дисперсий предполагаемых независимых сомножителей:

$$\sigma^2(Q) = \sigma^2(Q) + \sigma^2(p) + \sigma^2(M) \quad (3)$$

Можно показать, что $\sigma^2(N_\gamma) + \sigma^2(p) = \sigma^2(N_{el}) = 1/N_{el}$, а флуктуации усиления в простой модели размножения $\sigma(M) = 1/\delta$. В реальном детекторе различные факторы вносят свой вклад в дисперсию. Во-первых, дисперсия числа фотонов не является чисто пуассоновской: $\sigma^2(N_\gamma) = 1/N_\gamma + \sigma_{sc}^2$. В нашей установке добавочный член σ_{sc}^2 обусловлен в основном большим размером запускающего счетчика и следующими из этого изменениями светосбора в зависимости от места пересечения мюонного трека с детектором. Во-вторых, флуктуации анодного тока ФЭУ не ограничиваются одними статистическими флуктуациями размножения. Дисперсию увеличивают такие факторы, как междинодный пролет электронов, темновой ток, и другие. поэтому вклад флуктуаций усиления часто заменяют совокупной величиной так называемым коэффициентом избыточного шума ФЭУ B . Окончательно, имеем

$$\sigma^2(Q) = \sigma_{sc}^2 + \frac{1 + B}{N_{el}} \quad (4)$$

1.7 Вето счетчики

Вето счетчиками называют детекторы расположенные вокруг мишени, которая взаимодействует с частицами летящими из ускорителя. Они предназначены для обеспечения корректности параметров, попадаемых частиц на мишень. Соединенные в логические схемы, вето счетчики или как их еще называют счетчики охраны, исключают из подсчета частицы с некорректными характеристиками. В качестве вето счетчиков выступают детекторы сцинтилляционного типа. На установке использовалась система из 6 детекторов с чистым сцинтиллятором и 4 "сэндвичей". Детектор типа "сэндвич" отличается по конструкции от обычного сцинтилляционного детектора тем, что сцинтилятор, в нашем случае пластик, идет слоями чередуясь со свинцом.

1.8 Логическая схема

В нашем распоряжении на установке оказались сумматор и инвертер. Первое устройство имело на входе 4 разъема, на выходе 1, и выполняло функцию логического сложения сигналов. Инвертер - устройство преобразующее сигнал в обратный. В качестве инвертера использовался формироваель. Была реализована формула $\overline{(\bar{A} + \bar{B})}$.

1.9 Пьедестальные значения спектров

При каждом измерении нами были сняты амплитудные спектры пьедесталов. Пьедестал — амплитудный спектр шумов систем детектирования, передачи и обработки сигналов. Измерение пьедестальных значений проводилось при подключенных счетчиках, но без подачи высокого напряжения и напряжения подпитки ФЭУ. При обработке данных пьедестальные спектры вычитаются из основного пика, для определения нуля измерения.

1.10 Осциллограф

Осциллограф - электронное устройство, позволяющее наблюдать график зависимости амплитуды электрического сигнала от времени. У нас в распоряжении оказался очень продвинутый осциллограф Lecroy Waverunner Z606i. Помимо зависимости амплитуды от времени данный аппарат позволяет собирать статистику, строить гистограммы, выполнять различные измерения. Осциллограф имеет 4 входных канала, 8-битный АЦП, что означает чувствительность канала осциллографа в 97.6 мкВ на деление при чувствительности 200 мВ на клетку. Данная модель обладает тачскрином и работает на системе Windows 7.

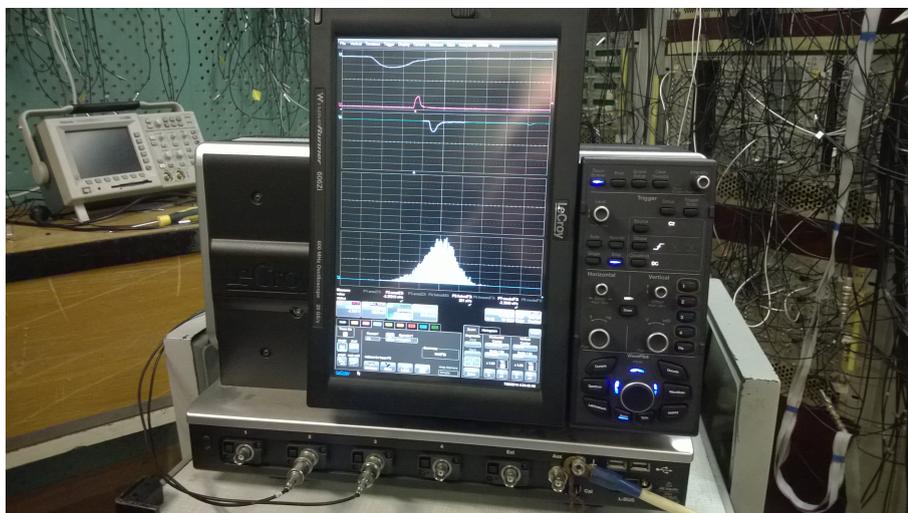


Рис. 3: Осциллограф

2 Практическая часть

2.1 Детектирование космических мюонов

В данных измерениях амплитудных спектров (зарядовых спектров, получаемых интегрированием сигнала в области 200 нс), совместно с детекторами мюонов, использовалась логическая схема совпадений для более точного подсчета мюонов прошедших через изучаемый детектор. Таким образом, если частица прошла через красные счетчики, вырабатывается триггерный (запускающий сигнал) и запускается для детектирования изучаемый (зеленый) счетчик. Это сделано для детектирования частиц, летящих под определенным углом, так как пролетая под разными углами мюоны проходят разный путь в сцинтиляторе и соответственно оставляют там разную энергию, что влияет на форму наблюдаемого спектра. При площади детектирования 144 см^2 расстояние до "наблюдателя" составляло 35 см ($\Omega = 0.117 \text{ ср}$). На рис. 4 изображена блок-схема данной системы. На рис. 5 изображена геометрия блока детектирования. В этом ряде измерений были задействованы детекторы с сцинтилляторами двух типов: "сэндвичи" (sa1 и sa4 - большие, sa7 и sa8 - маленькие) и с обычным сцинтиллятором (sci2, sci3, sci4, sci5, sci6). Для sa1, sa4, sci2 были сняты измерения в трех точках на рис.6 и 7 показано расположение точек.

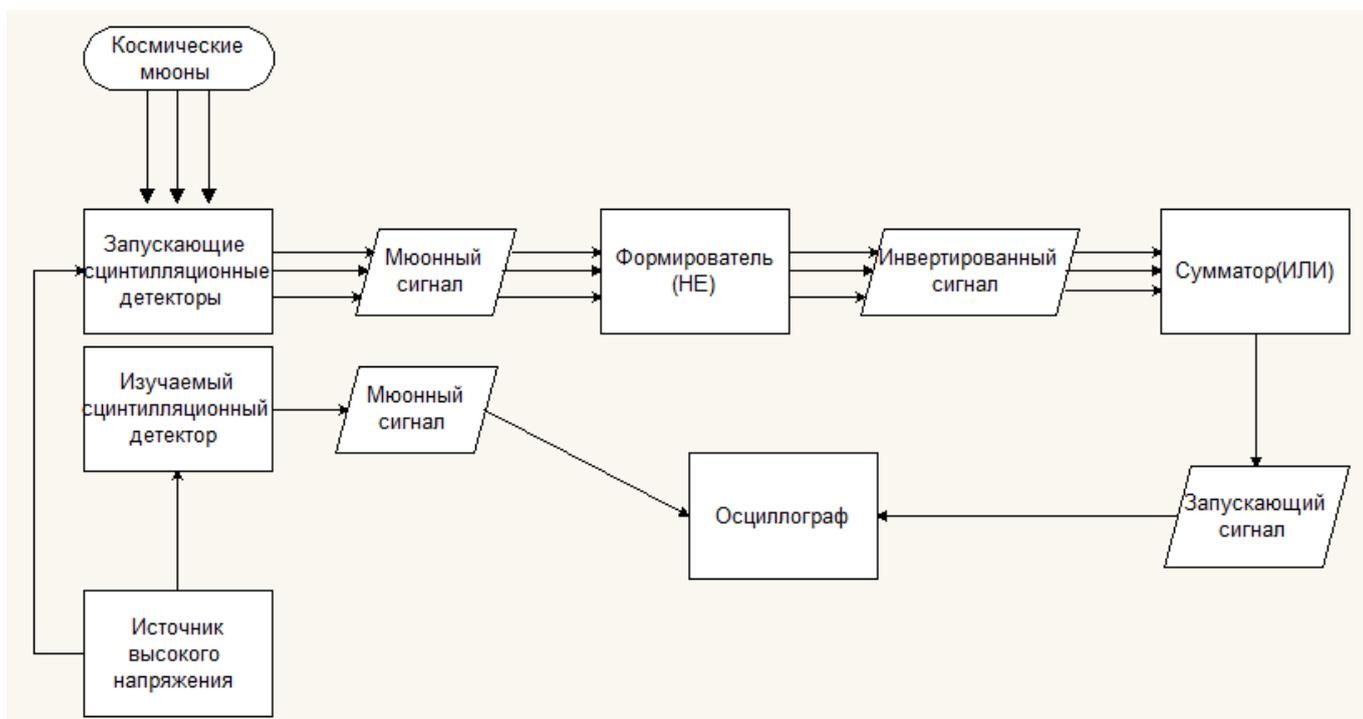


Рис. 4: Блок-схема эксперимента

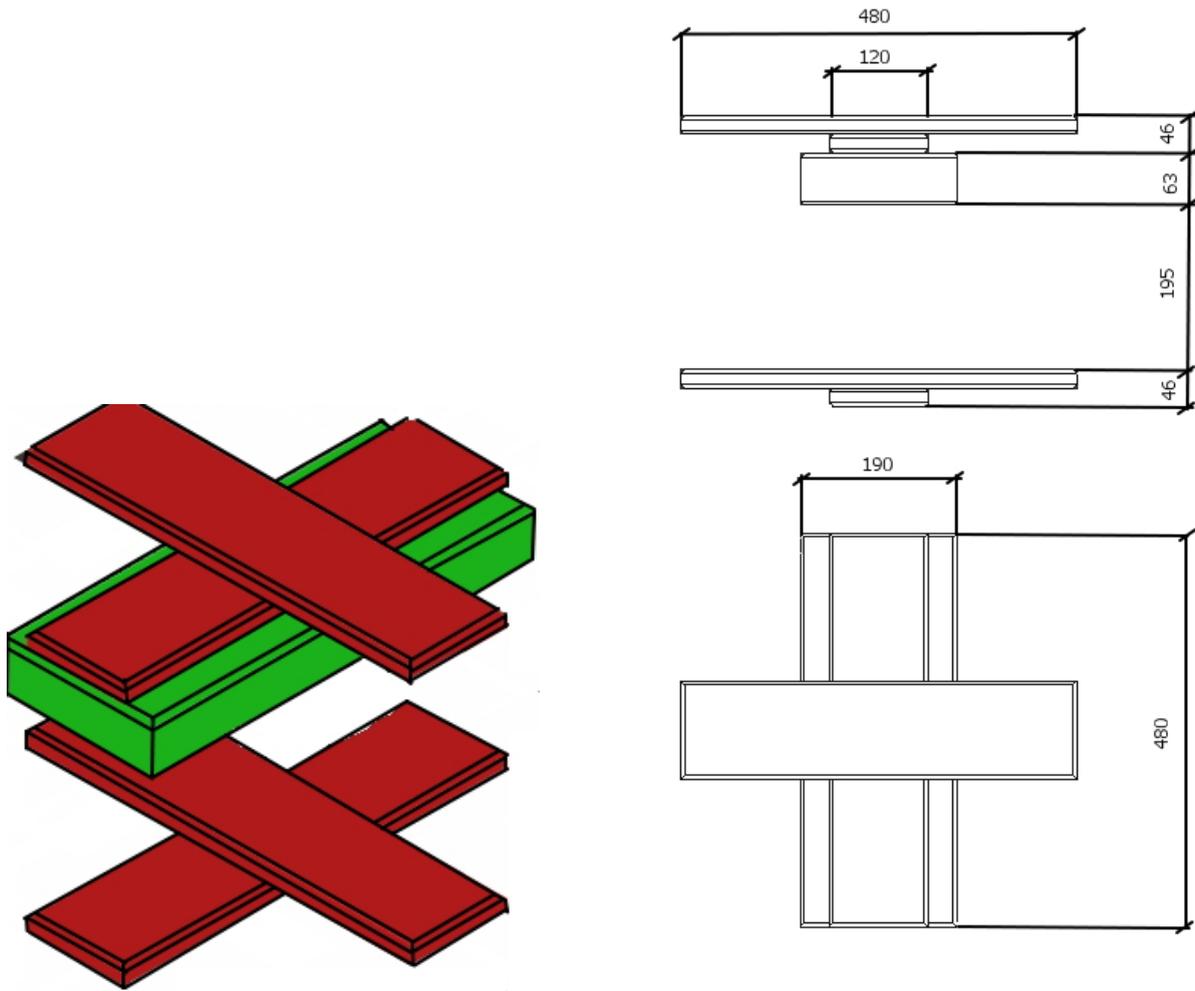


Рис. 5: Геометрия эксперимента

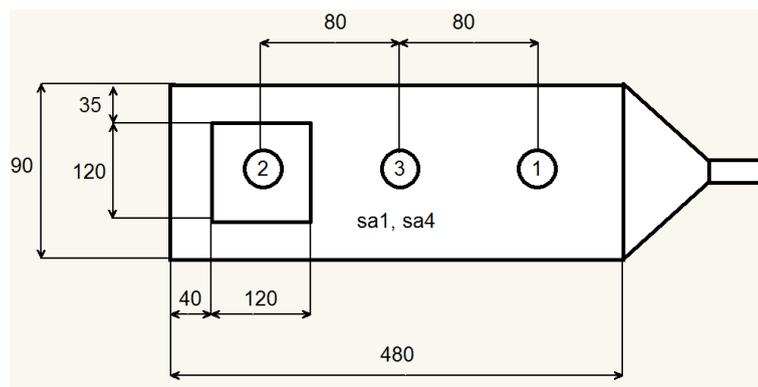


Рис. 6: Точки на сэндвичах

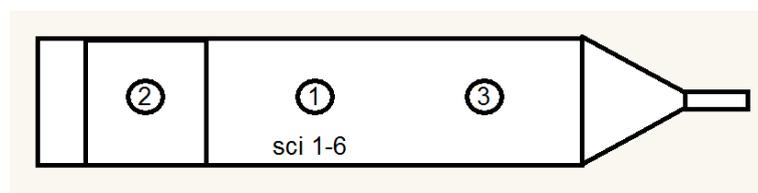


Рис. 7: Точки на сцинтилляторах

В таблице 1 приведены данные и параметры полученные с помощью осциллографа. Выходными данными являются гистораммы амплитудных спектров. Scale, mV — чувствительность шкалы измерения амплитуды входящего сигнала, в окне 8 клеток. Entries — число событий.

Таблица 1: Космические мюоны

№	Измер.объект	Scale,mV	Entries	Файл .txt
1	sa4(точка1)	500	10448	sa4point1
2	sa1(точка1)	500	10448	sa1point1
3	sa4(точка2)	500	1728	sa4point2
4	sa1(точка2)	500	1728	sa1point2
5	sa4(точка3)	500	>2000	sa4point3
6	sa1(точка3)	500	>2000	sa1point3
7	sa4(пьедестал)	500	>2000	-
8	sa1(пьедестал)	500	>2000	-
9	sa7	200	5000	sa7scale200
10	sa8	200	>2000	sa8scale200
11	sa7(пьедестал)	200	5000	sa7pedscale200
12	sa8(пьедестал)	200	5000	sa8pedscale200
13	sa7(пьедестал)	100	5000	sa7pedscale200
14	sa8(пьедестал)	100	5000	sa8pedscale200
15	sa7	100	855	sa7scale100
16	sa8	100	855	sa8scale100
17	sa7(расст.55см)	100	>2000	sa7scale100R55
18	sa8(расст.55см)	100	>2000	sa8scale100R55
19	sci2(4зап.,точка1)	100	1092	sci2fourstart
20	sci2(3зап.,точка1)	100	15925	sci2threestart
21	sci2(точка2)	100	>2000	sci2point2
22	sci5(точка1)	100	<2000	sci5point1
23	sci2(точка3)	100	1668	sci2point3
24	sci2(пьедестал)	100	>2000	sci2ped
25	sci3(точка1)	100	1668	sci3point1
26	sci3(пьедестал)	100	>2000	sci3ped
27	sci4(точка1)	100	2067	sci4point1
28	sci4(пьедестал)	100	>2000	sci4ped
29	sci6(точка1)	100	2163	sci6point1
30	sci6(пьедестал)	100	>2000	sci6ped

2.2 Измерения с LED

Данный ряд измерений проводился для выяснения значения вклада распределения фотоэлектронов в амплитудный спектр мюонного пика. В разделе В вещество сцинтиллятора был вмонтирован светодиод. Место соединения светодиода со сцинтиллятором светоизолировано. При снятии данных с счетчика sa1 использовался делитель для канала генератора, т. е. запускающий сигнал и сигнал на светодиод шли из канала №1, поэтому напряжение пришлось увеличить вдвое. При измерении sci2 светодиод был подключен к генератору электрических импульсов к каналу №1, к каналу 2 был подключен сигнальный кабель А7 идущий к осциллографу. Импульсы на обоих каналах генератора были синхронизированы по выдаче напряжения, частоте, ширине и форме сигнала. Импульс, исходивший из канала №2 являлся запускающим. Изменяя напряжение на светодиоде, можно регулировать интенсивность свечения, т. е. с увеличением напряжения амплитуда сигнала ФЭУ увеличивается. В таблицах 8 и 4 при различных напряжениях были сняты амплитудные спектры: LED — при включенном светодиоде, Ped1 - пьедестал, при выключенном HV, Ped2 — шумы осциллографа при данном Scale(чувствительность).

Таблица 2: Сигнал генератора

Параметр	Значение
Форма	Pulse
Частота	1 kHz
Ширина	25 ns
Нижний порог	0 V

Таблица 3: Измерение sa1 с LED

№	U_0 , В	Scale,mV	Сигнал	MPV,nVs	Файл данных .txt
1	6.8	50	LED	-5.9	sa1led68
		50	Ped1	-0.26	sa1ped68
		50	Ped2	-0.25	sa1no50mv
2	6.9	50	LED	-11.3	sa1led69
		50	Ped1	-0.27	sa1ped69
3	6.95	100	LED	-15.7	sa1led695
		100	Ped1	-0.56	sa1ped695
		100	Ped2	-0.51	sa1no100mv
4	7	100	LED	-20.5	sa1led7
		100	Ped1	-0.55	sa1ped7

Продолжение таблицы 3

№	U_0 , В	Scale,mV	Сигнал	MPV,nVs	Файл данных .txt
5	7.05	100	LED	-25.6	sa1led705
		100	Ped1	-0.540	sa1ped705

Таблица 4: Измерение sci2 с LED

№	U_0 , В	Scale,mV	Сигнал	MPV,nVs	Файл данных .txt
1	6.5	50	LED	-1.8	sci2led65
		50	Ped1	-0.27	sci2ped65
		50	Ped2	-0.26	sci2no50mv
2	6.55	50	LED	-2	sci2led655
		50	Ped1	-0.25	sci2ped655
	6.65	100	LED	-4	sci2led665
		100	Ped1	-0.21	sci2ped665
		100	Ped2	-0.13	sci2no100mv
4	6.7	100	LED	-5.2	sci2led67
		100	Ped1	-0.13	sci2ped67
5	6.75	100	LED	-10.1	sci2led675
		100	Ped1	-0.21	sci2ped675
6	6.8	100	LED	-12	sci2led68
		100	Ped1	-0.13	sci2ped68
7	6.85	100	LED	-15.9	sci2led685
		100	Ped1	-0.13	sci2ped685
8	6.87	100	LED	-16.6	sci2led687
		100	Ped1	-0.13	sci2ped687
9	6.9	100	LED	-21	sci2led69
		100	Ped1	-0.13	sci2ped69

В разделе "Разрешение ФЭУ" объясняется, что статистика фотоэлектронов подчиняется распределению Гаусса. При стабильных параметрах излучения от LED, можно установить корневую зависимость относительного разрешения от количества фотоэлектронов возникающих в ФЭУ.

$$\frac{\sigma_Q}{Q} = \frac{1+B}{\sqrt{N_{el}}} \quad (5)$$

, где N - количество фотоэлектронов, Q - заряд на аноде ФЭУ, B - коэффициент избыточного шума. В результате измерений была установлена корневая зависимость только для одного счетчика sa1(график 8), из-за нестабильности работы светодио-

да. Было сделано предположение, что неправильно выбрано напряжение питания, т.е. при низких напряжениях возникают флуктуации в свечении. Впоследствии, был проведен эксперимент с затемнением светодиода, с целью повышения его рабочего напряжения, результатов он не дал.

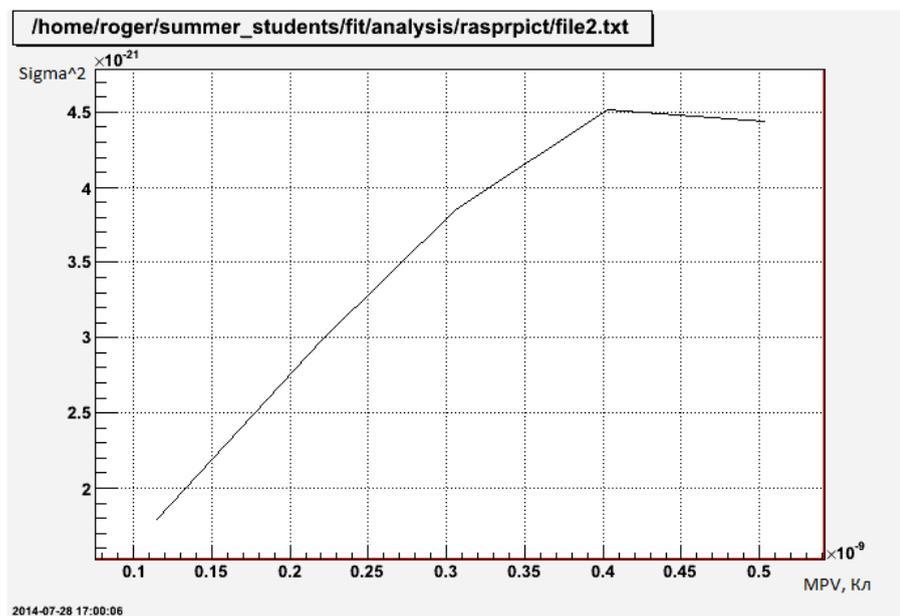


Рис. 8: salled, $\sigma_Q^2 = f(N)$

2.3 Обработка экспериментальных данных

Данные, которые поступили с детекторов необходимо фитировать. Первым предположением было, что мы наблюдаем кривую Гаусса. При помощи библиотеки для C++ Root была написана терминальная программа для фитирования полученных данных функцией Гаусса и вывода результатов на экран в виде гистограммы, с наложенной на него кривой функции фитирования. Результат фитирования также выводится на экран. После нескольких попыток было установлено, что данная гипотеза неверна, так как отношение χ^2 к ndf (степеням свободы) был равен для всех графиков приблизительно 20-30. Было выдвинуто новое предположение - фитировать данные кривой Ландау, так как процессы в счетчике идут ионизационного типа. Написана программа для фитирования данных, как и в предыдущем случае, но гауссиан был заменен на функцию Ландау. Полученные результаты так же не удовлетворяли нашей гипотезе и отношение критерия Пирса на степени свободы были порядка 20 -30. Было выдвинуто третье предположение, полученные гистограммы являются результатом свертки функции Гаусса с Ландау. Полученная задача является по своей сути математически некорректной и решается с помощью методов численного интегрирования. Была написана программа, которая сворачивает две функции Гаусса и Ландау, производит фит гистограммы, выводит на экран гистограмму и полученную кривую. Так же

на экран выводятся результаты фита, а именно: ширина Ландау, MPV , подинтегральная площадь без учета ширины бина, ширина Гаусса. Результат получился очень хороший. Отношение χ^2 на ndf (степени свободы) получался порядка единицы для всех данных, что говорит о правдоподобии нашей гипотезы. Написанная программа очень плохо сходится, для этого кроме начальных приближений были введены границы. Примером плохой процедуры конволюции является 13я строчка во второй таблице. Проверка программы была выполнена следующим образом. Был создан массив который был наполнен данными о кривой Гаусса с известными параметрами. Результат фитирования функцией ЛанГауса дал ожидаемый результат: пренебрежимо малый Ландау по сравнению с очень большим Гауссом.

Таблица 5: Космические мюоны: результаты фита(все в пКл)

№	Объект	MPV	ΔMPV	W_{Land}	ΔW_{Land}	$SGau$	$\Delta SGau$	MP-Ped	SGau-RPED
1	sa1p1	351	1	18	0.8	58	1	262	54
2	sa1p2	306	3	18	2	54	4	211	49
3	sa1p3	310	3	14	2	54	3	230	50
4	sa4p1	227	0.8	10	0.45	37	1	150	30
5	sa4p2	219	2	7	1.2	36	2	149	34
6	sa4p3	205	2	12	2.2	31	2.5	141	28
7	sa8s100	247	2.2	11.3	2	29	2.9	226	27
8	sa8s200	250	1	12.4	1	30	1.4	220	25
9	sa7s200	283	1	16	1	44	1.3	245	41
10	sa7s100	269	3.4	15	3	42	3.9	244	41.2
11	sa7s100R55	256	6.9	10.5	5.9	41.7	6.6	226	37.6
12	sa8s100R55	231	2.5	13.7	2.8	25	5	206	22
13	sci2fst	307	1.4	20	2	9.5	4.3	292	9.2
14	sci2ts	311	0.4	16	0.4	22.8	0.6	293	21
15	sci2p2	293	1.3	13.8	1.5	24.4	1.9	280	24.5
16	sci2p3	317.9	1.5	20	2.4	21	3.7	304	21
17	sci3p1	289	2.4	17	5	28.3	6.1	270	26
18	sci4p1	306	2.7	21.5	5	22.8	5.7	287	20
19	sci5p1	151	2.5	5.3	3.5	18.1	3.5	135	17.1
20	sci6p1	300	2.4	20.1	7.4	18.7	14	281	17
21	sci1RE	284	4	10	3.9	25.1	5.3	262	25.1
22	sci5RE	192	2.4	14	3.5	13.7	6.1	168	12

Величины в таблице 5:

W_{land} – ширина Ландау;

$SGau$ – сигма Гаусса;

Ped – наиболее вероятное значение пьедестала;

RPED – среднее значение пьедестала;

$$SG_{\text{Gau-RPED}} = \sqrt{SG_{\text{Gau}}^2 - RPED^2}.$$

Таблица 6: Относительные величины измерений

№	Объект	WL/Mp-Ped	SG/Mp-Ped	SG-RPed/Mp-Ped
1	sa1point1	0.067	0.220	0.210
2	sa1point2	0.079	0.258	0.230
3	sa1point3	0.060	0.230	0.220
4	sa4point1	0.066	0.217	0.202
5	sa4point2	0.044	0.241	0.221
6	sa4point3	0.085	0.218	0.195
7	sa8scale200	0.056	0.138	0.114
8	sa8scale100	0.050	0.128	0.121
9	sa7scale200	0.064	0.179	0.169
10	sa7scale100	0.061	0.172	0.168
11	sa7scale100R55	0.046	0.184	0.166
12	sa8scale100R55	0.066	0.121	0.106
13	sci2fourstart	0.066	0.032	0.031
14	sci2threestart	0.054	0.077	0.070
15	sci2point2	0.049	0.087	0.087
16	sci2point3	0.065	0.068	0.068
17	sci3point1	0.065	0.105	0.095
18	sci4point1	0.075	0.079	0.070
19	sci5point1	0.039	0.134	0.126
20	sci6point1	0.071	0.067	0.062
21	Sci1	0.069	0.116	0.077
22	Sci1RE	0.038	0.096	0.096
23	Sci5RE	0.083	0.082	0.071

Описание таблицы 6:

WL – ширина Ландау;

Mp – наиболее вероятное значение;

Ped – наиболее вероятное значение пьедестала;

SG – сигма Гауса;

RPed – среднее значение пьедестала;

$$SG\text{-RPed} = \sqrt{SG^2 - RPed^2}.$$

2.4 Измерение порога срабатывания каналов схемы антисовпадений с помощью сигналов космических мюонов

Последней задачей, поставленной перед нами, являлась проверка чувствительности схемы антисовпадений при характерном мюонном сигнале. Данное упражнение позволяет оценить идентичность каналов. Основная идея реализации данного эксперимента состоит в том, что лучшим сигналом для проверки порога является тот сигнал, который будет измеряться в последующем. В ином случае, можно было бы воспользоваться генератором сигналов. Но так как сигнал от мюона имеет характерные флуктуации по амплитуде и форме решено было с помощью делителей напряжения понижать уровень сигнала и с помощью счетчика срабатываний определять порог срабатывания канала схемы антисовпадений.

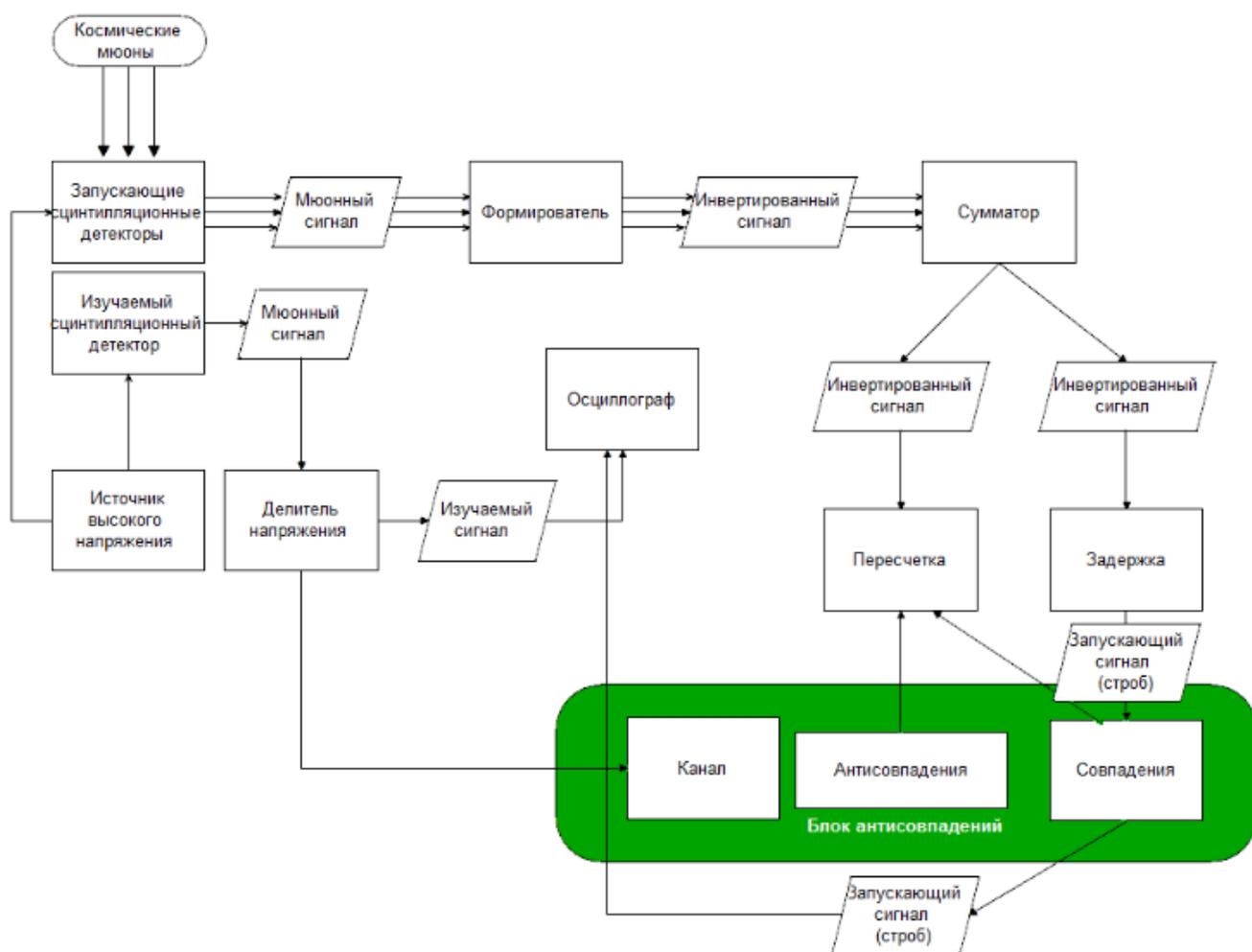


Рис. 9: Блок-схема измерения

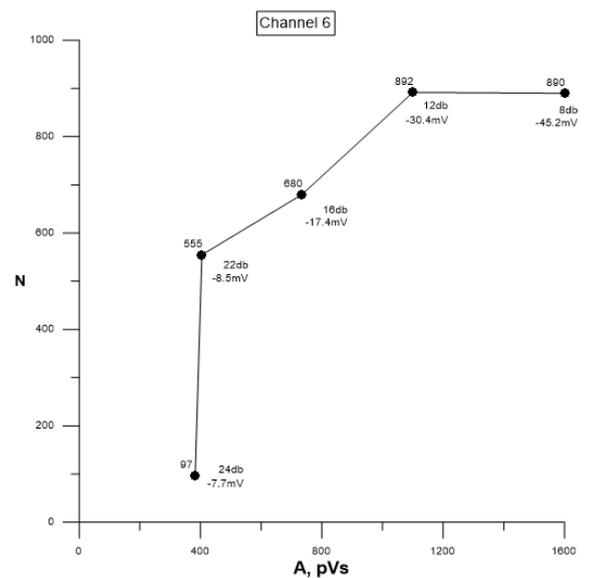
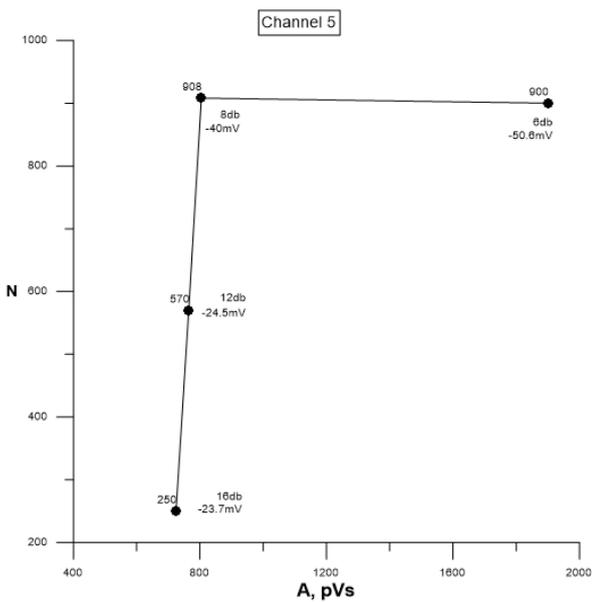
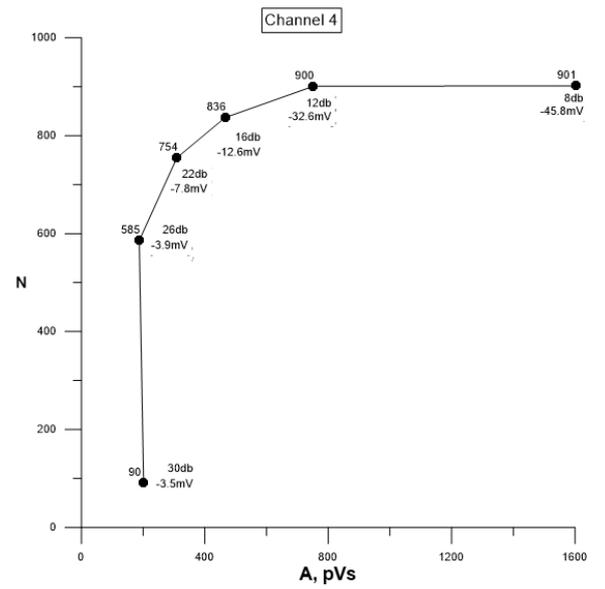
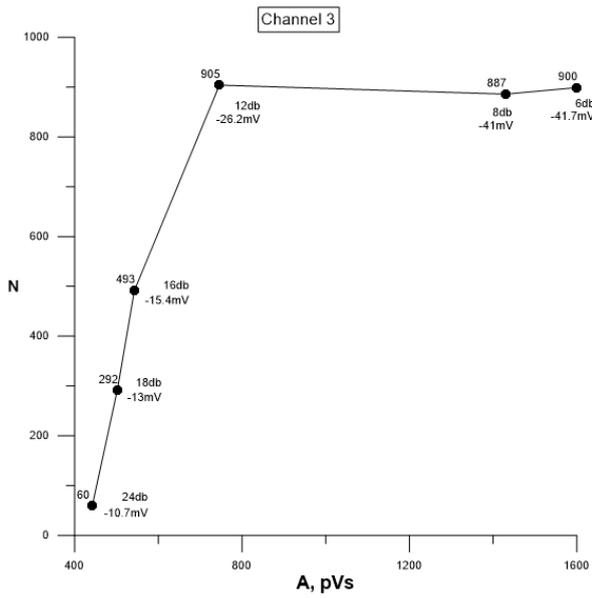
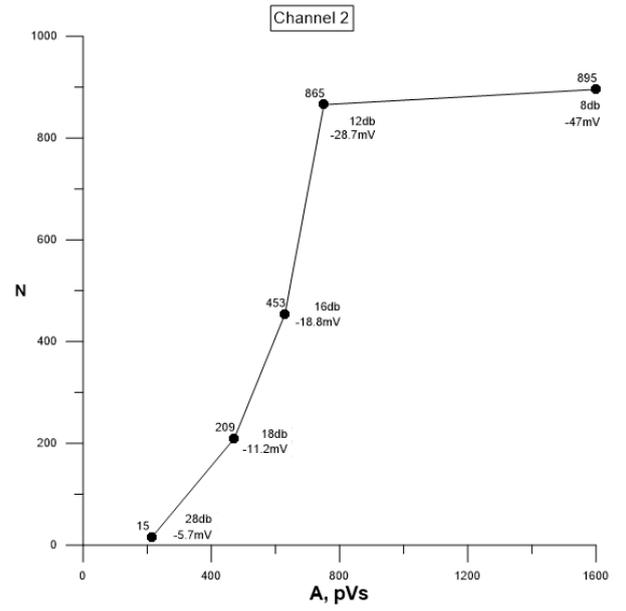
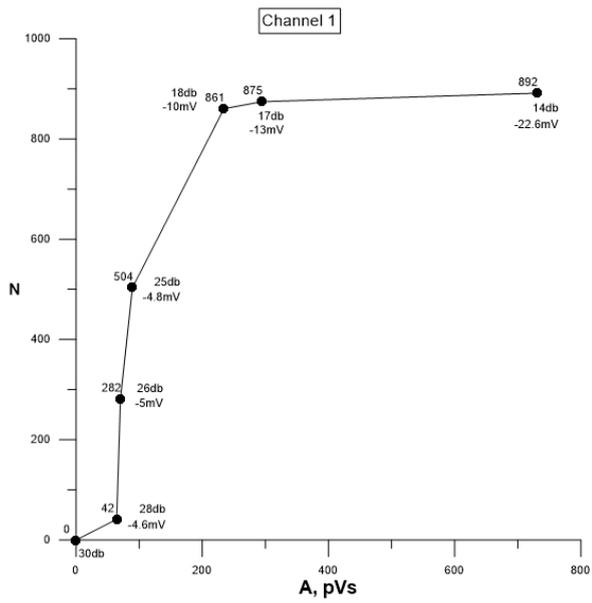


Рис. 10: Чувствительность каналов схемы антисовпадений

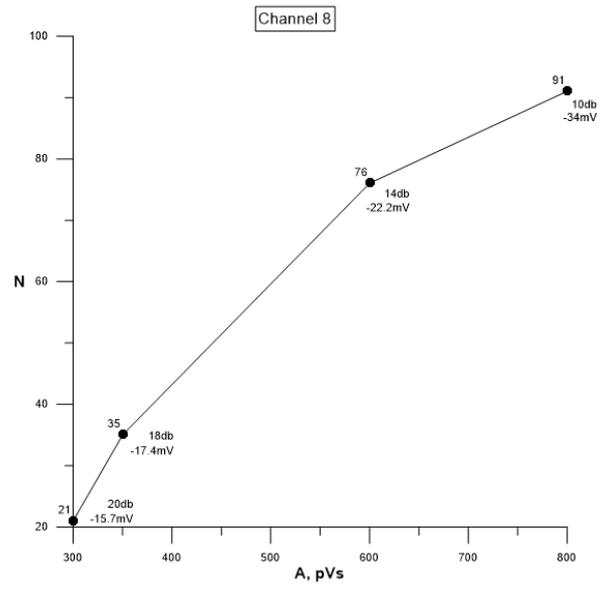
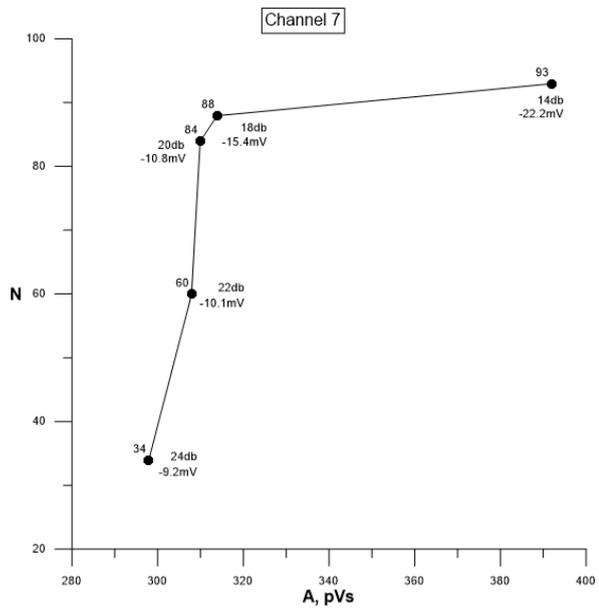


Рис. 11: Чувствительность каналов схемы антисовпадений

Заключение

В ходе прохождения летней практики были выполнены следующие задачи. А именно:

1. Откалиброваны вето счетчики на космических мюонах
2. Опытным путем было установлено, что светодиоды ведут себя некорректно и определить вклад статистики фотоэлектронов в мюонный пик определить удалось только на одном счетчике(sa1). Эти данные послужат для проектирования новых детекторов, которые будут оборудованы штатными светодиодами.
3. Выполнено измерение порога срабатывания схемы антисовпадений триггерных счетчиков с помощью космических мюонов.

3 Приложение

3.1 Журнал лабораторных наблюдений

1. При проверке сцинтилляторов выяснилось, что сцинтиллятор №1 не работает.
2. При проверке соответствия на сигнальных кабелях и кабелях высокого напряжения было установлено, что метка кабелей HV на sci2 и sci6 не соответствует действительности. Исправлено: были приклеены метки с верными значениями.
3. При использовании осциллографа Lecroy Waverunner Z606i выяснилось, что при настройке более высокой чувствительности (Scale) измерения снижается ошибка битности.
4. Снова используется sci1, был заменен ФЭУ. Также было соответственно изменено высокое напряжение на канале A1(обозначение в CAEN) с 1250 В на 1150 В, так как характерная амплитуда мюонных сигналов в два раза превышала норму, а также была замечены повышенное количество шумов большой амплитуды. Результат: детектор sci1 работает нормально

3.2 Спектры амплитуд космических мюонов

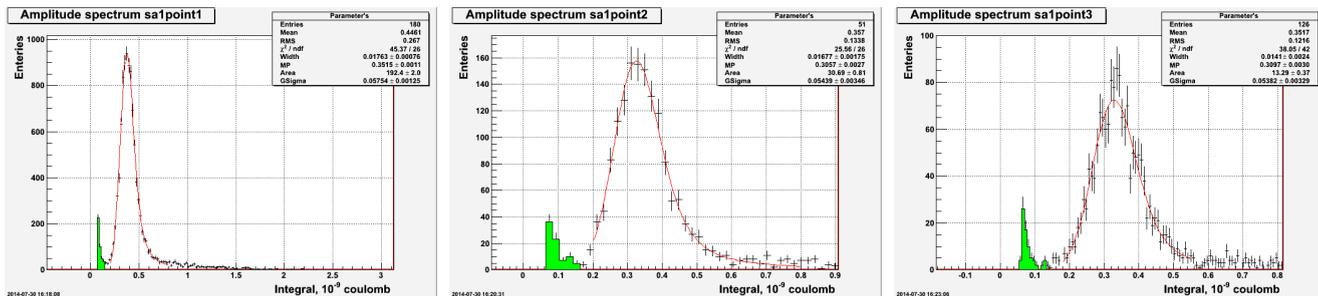


Рис. 12: sa1cosmic

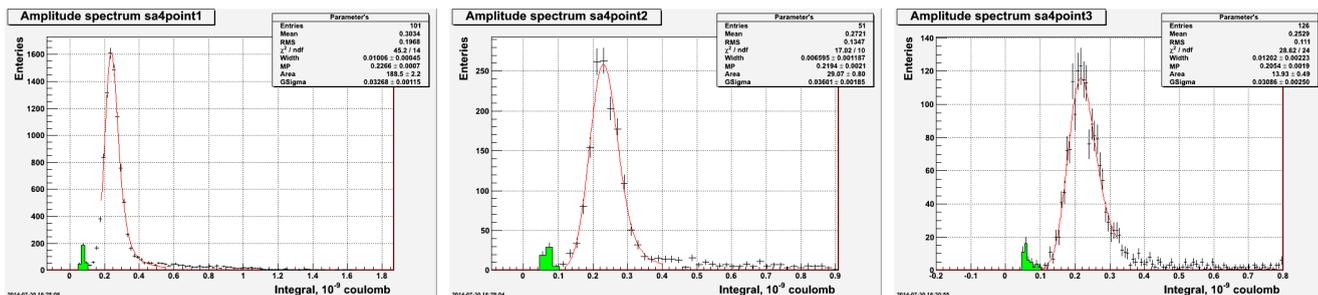


Рис. 13: sa4cosmic

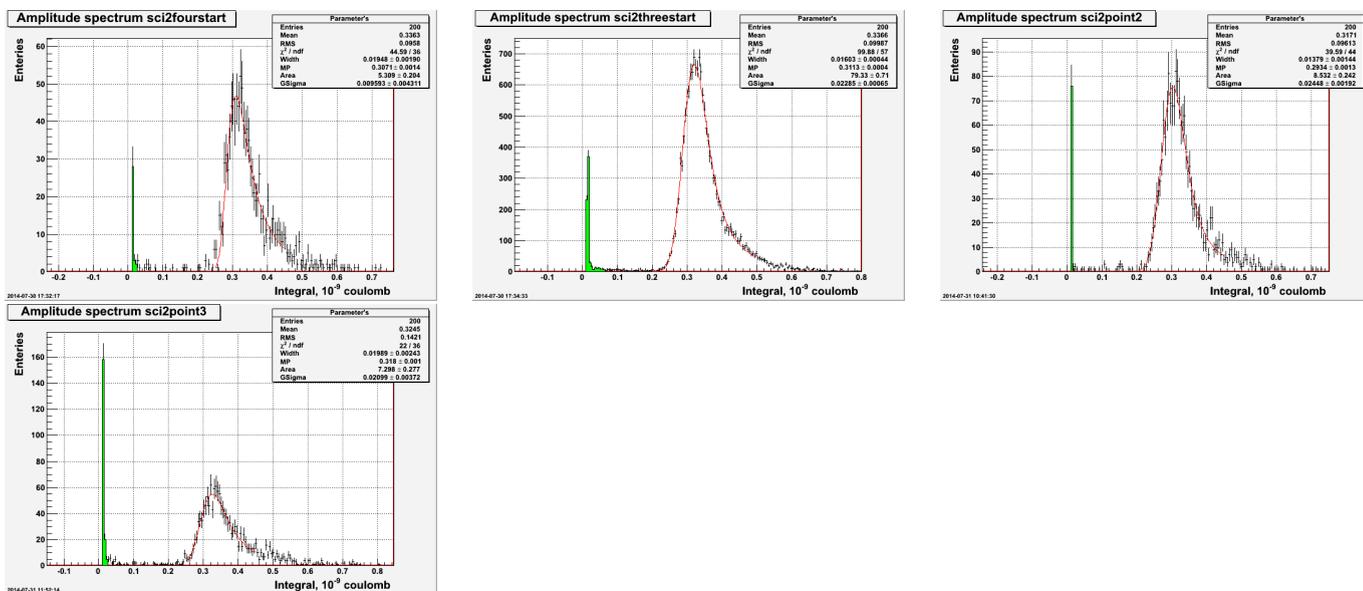


Рис. 14: sci2cosmic

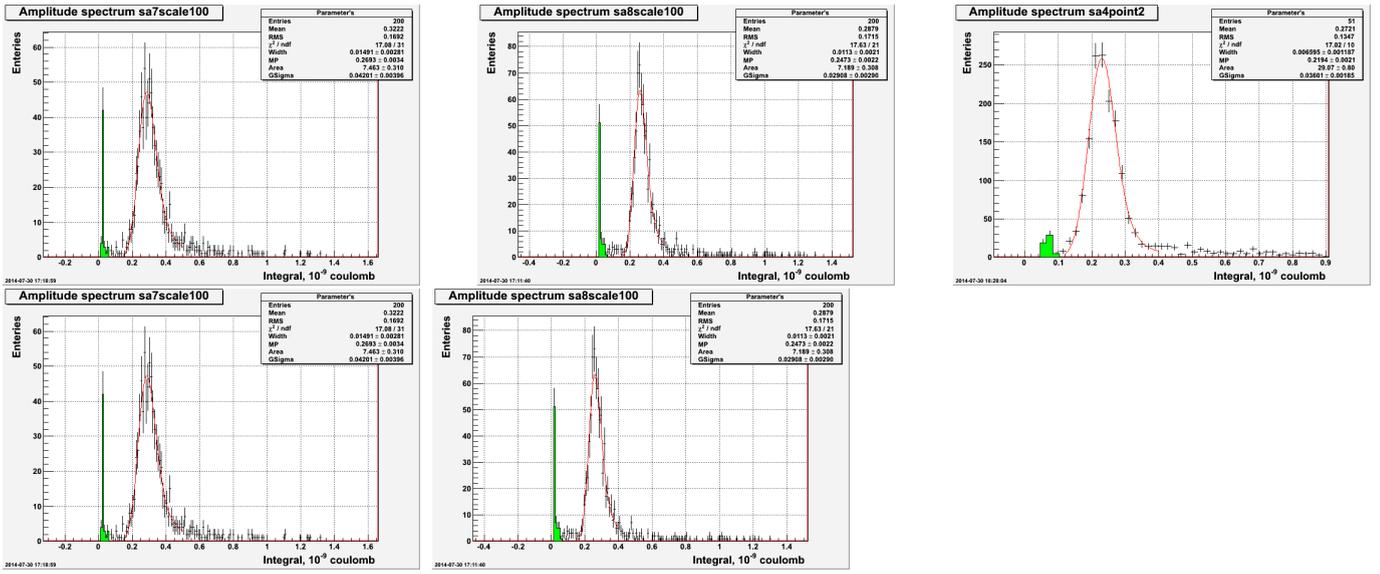


Рис. 15: sa7, sa8 cosmic

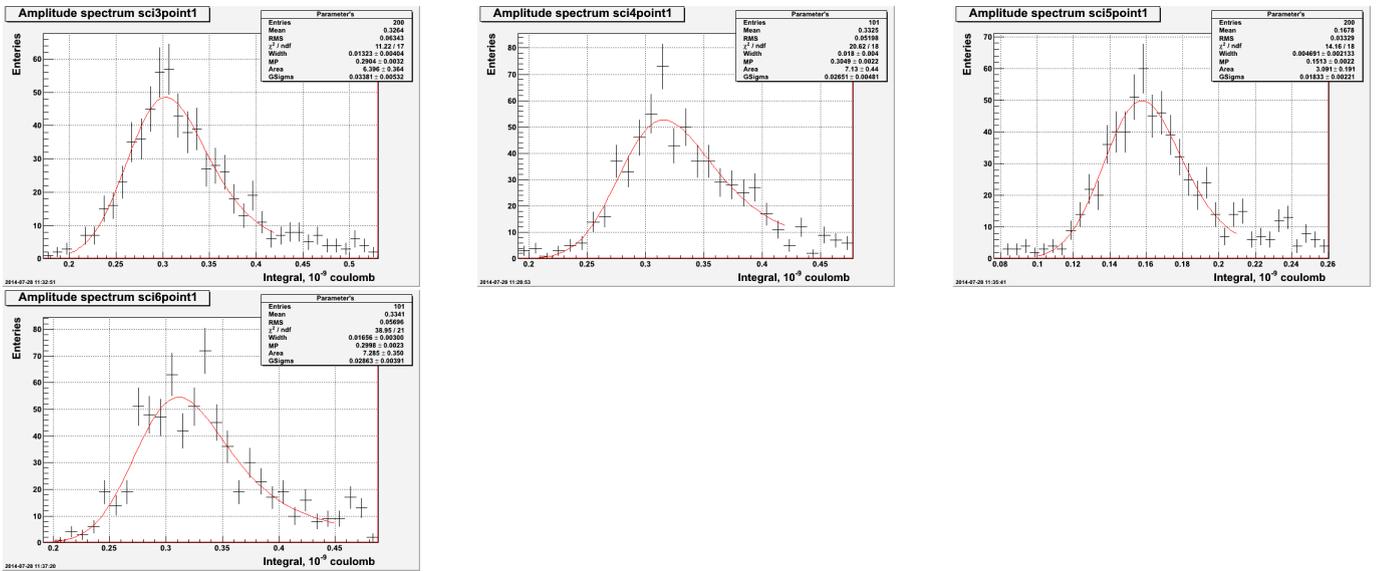


Рис. 16: sci3, sci4, sci5, sci6

3.3 Листинги программ

3.3.1 Гаусс

```
#include <iostream>
#include <TF1.h>
#include <TH1D.h>
#include <TGraph.h>
#include <TCanvas.h>
#include <TR00T.h>

int triv_fit_gaus2(const char *path = "/home/roger/summer_students/raspr/s

int triv_fit_gaus2(const char *path, double r1, double r2)
{
    gROOT->SetStyle("plain");
    TGraph *Fh= new TGraph(path);
    double *Xi= Fh->GetX();
    double *Yi= Fh->GetY();
    const int Nbin = Fh->GetN();
    double Xi2[Nbin];
    double Yi2[Nbin];
    for(int i=0;i<Fh->GetN();i++)
        {
            Xi[i]/=-50e-12;
        }
    //----- flip arrays -----
    for(int k=0;k<Fh->GetN();k++)
        {
            Xi2[k]=Xi[Fh->GetN()-1-k];
            Yi2[k]=Yi[Fh->GetN()-1-k];
        }
    double entries = 0;
    for(int l=0;l<Fh->GetN();l++)
        {
            Xi[l]=Xi2[l];
```

```

        Yi[1]=Yi2[1];
        entries += Yi[1];
    }
    //----- filling hist -----
    TH1D *Hist = new TH1D("HIST","Amplitude spectrum SA7pedscale200(GAUS);In
    for(int j=1; j<=Fh->GetN();j++)
{
Hist->SetBinContent(j,Yi[j-1]);
    Hist->SetBinError(j,sqrt(Yi[j-1]));
}
    //----- fit -----
TF1 *Gau2s = new TF1("Gaus","gaus(0)",r1,r2);
    Gau2s->SetParameter(0,Yi[Hist->GetMaximumBin()-1]);
    Gau2s->SetParameter(1,Hist->GetMean(1));
    Gau2s->SetParameter(2,Hist->GetRMS(1));
    /*TF1 *land2 = new TF1("landeu","[2]*TMath::Landau(x,[0],[1],0)",r1,r2);
    land2->SetParameter(0,Hist->GetMean(1));
    land2->SetParameter(1,Hist->GetRMS(1));
    land2->SetParameter(2,entries/5.0);*/

    //Fh->Fit(land2,"R");
    //Fh->Draw("apl");
    //Hist->Draw("error");
    Hist->Fit(Gau2s,"R");
    double_t A;
    A=Gau2s->GetParameter(0);
    double_t Ae;
    Ae=Gau2s->GetParError(0);
    cout<<"A = "<<A<< " +- "<<Ae<< endl;
    double_t mpv;
    mpv=Gau2s->GetParameter(1);
    double_t mpve;
    mpve=Gau2s->GetParError(1);
    cout<<"mpv = "<<mpv<< " +- "<<mpve<< endl;
    double_t sigma;

```

```

sigma=Gau2s->GetParameter(2);
double_t sigmae;
sigmae=Gau2s->GetParError(2);
cout<<"sigma = "<<sigma<< " +- "<<sigmae<< endl;
return 0;
}

```

3.3.2 Ландау

```
#include <iostream>
```

```
#include <TF1.h>
```

```
#include <TH1D.h>
```

```
#include <TGraph.h>
```

```
#include <TCanvas.h>
```

```
#include <TRROOT.h>
```

```
int triv_fit_landau2(const char *path = "/home/roger/summer_students/fit/F
```

```

int triv_fit_landau2(const char *path, double r1, double r2) {
    gROOT->SetStyle("plain");

```

```
    TGraph *Fh= new TGraph(path);
```

```
    double *Xi= Fh->GetX();
```

```
    double *Yi= Fh->GetY();
```

```
    const int Nbin = Fh->GetN();
```

```
    double Xi2[Nbin];
```

```
    double Yi2[Nbin];
```

```
    for(int i=0;i<Fh->GetN();i++)
```

```
    {
```

```
        Xi[i]/=-50;
```

```
    }
```

```
//----- flip arrays -----
```

```
    for(int k=0;k<Fh->GetN();k++)
```

```
    {
```

```
        Xi2[k]=Xi[Fh->GetN()-1-k];
```

```

        Yi2[k]=Yi[Fh->GetN()-1-k];
    }
double entries = 0;
for(int l=0;l<Fh->GetN();l++)
    {
        Xi[l]=Xi2[l];
        Yi[l]=Yi2[l];
        entries += Yi[l];
    }
cout << entries << endl;
//----- filling hist -----
TH1D *Hist = new TH1D("HIST","hist",Fh->GetN(),Xi[0],Xi[Nbin-1]);
for(int j=1; j<=Fh->GetN();j++) {
    Hist->SetBinContent(j,Yi[j-1]);
    Hist->SetBinError(j,sqrt(Yi[j-1]));
}
//----- fit -----

/*TF1 *Gau2s = new TF1("Gaus","gaus(0)",r1,r2);
Gau2s->SetParameter(0,Yi[Hist->GetMaximumBin()-1]);
Gau2s->SetParameter(1,Hist->GetMean(1));
Gau2s->SetParameter(2,Hist->GetRMS(1));*/

TF1 *land2 = new TF1("landeu","[2]*TMath::Landau(x,[0],[1],0)",r1,r2);
land2->SetParameter(0,Hist->GetMean(1));
land2->SetParameter(1,Hist->GetRMS(1));
land2->SetParameter(2,Yi[Hist->GetMaximumBin()-1]);

//Fh->Fit(land2,"R");
//Fh->Draw("apl");
//Hist->Draw("error");
Hist->Fit(land2,"R");
double_t o0= land2->GetParameter(0);
cout<<o0<< endl;
return 0;

```

```
}
```

3.3.3 ЛанГаусс

```
#include "TH1.h"
#include "TF1.h"
#include "TROOT.h"
#include "TStyle.h"
#include "TMath.h"
#include "TGraph.h"
#include "TCanvas.h"
#include <fstream>
#include <iostream>
#include <TFile.h>
#include <TObject.h>
//using namespace std;

Double_t langaufun(Double_t *x, Double_t *par) {

    //Параметры фита
    //par[0]=ландгау
    //par[1]=наиболее вероятное
    //par[2]=, .!..=(^_^)=..!.,
    //par[3]=Width (sigma) of convoluted Gaussian function

    Double_t invsq2pi = 0.3989422804014;    // (2 pi)^(-1/2)
    Double_t mpshift = -0.22278298;

    // ???
    Double_t np = 100.0;    // количество шагов
    Double_t sc = 5.0;

    // Переменные
    Double_t xx;
    Double_t mpc;
```

```

Double_t fland;
Double_t sum = 0.0;
Double_t xlow,xupp;
Double_t step;
Double_t i;

mpc = par[1] - mpshift * par[0];

// Граница для интеграла
xlow = x[0] - sc * par[3];
xupp = x[0] + sc * par[3];

step = (xupp-xlow) / np;

// Свертка
for(i=1.0; i<=np/2; i++) {
    xx = xlow + (i-.5) * step;
    fland = TMath::Landau(xx,mpc,par[0]) / par[0];
    sum += fland * TMath::Gaus(x[0],xx,par[3]);

    xx = xupp - (i-.5) * step;
    fland = TMath::Landau(xx,mpc,par[0]) / par[0];
    sum += fland * TMath::Gaus(x[0],xx,par[3]);
}

return (par[2] * step * sum * invsq2pi / par[3]);
}

```

```

TF1 *langaufit(TH1D *his, Double_t *fitrange, Double_t *startvalues, Double_t *par)
{
    // параметры ЛанГауса

```

```

//  par[0]=ширина Ландау
//  par[1]=МПВ
//  par[2]=Подинтегральная площадь
//  par[3]=сигма Гауса
//
//  переменные для ландауфит:
//  his          фитируемая гистограмма
//  fitrange[2]  границы фита
//  startvalues[4] начальные приближения
//  parllimitslo[4] нижняя граница для нач
//  parlimitshi[4] верхняя граница для нач
//  fitparams[4]  вышло
//  fiterrors[4]  вышло ошибки
//  ChiSqr       хиквадрат
//  NDF          ndf

```

```
Int_t i;
```

```
Char_t FunName[100];
```

```
sprintf(FunName,"Fitfcn_%s",his->GetName());
```

```
TF1 *ffitold = (TF1*)gROOT->GetListOfFunctions()->FindObject(FunName);
if (ffitold) delete ffitold;
```

```
TF1 *ffit = new TF1(FunName,langaufun,fitrange[0],fitrange[1],4);
ffit->SetParameters(startvalues);
ffit->SetParNames("Width","MP","Area","GSigma");
```

```
for (i=0; i<4; i++) {
    ffit->SetParLimits(i, parlimitslo[i], parlimitshi[i]);
}

```

```
his->Fit(FunName,"RBO");
```

```
ffit->GetParameters(fitparams);
```

```

for (i=0; i<4; i++) {
    fiterrors[i] = ffit->GetParError(i);
}
ChiSqr[0] = ffit->GetChisquare();
NDF[0] = ffit->GetNDF();

return (ffit);          // возвращает функцию для фитирования
}

```

```

Int_t langaupro(Double_t *params, Double_t &maxx, Double_t &FWHM) {

```

```

    Double_t p,x,fy,fxr,fxl;
    Double_t step;
    Double_t l,lold;
    Int_t i = 0;
    Int_t MAXCALLS = 10000;

```

```

    // поиск максимума

```

```

    p = params[1] - 0.1 * params[0];
    step = 0.05 * params[0];
    lold = -2.0;
    l = -1.0;

```

```

    while ( (l != lold) && (i < MAXCALLS) ) {
        i++;

        lold = l;
        x = p + step;
        l = langaufun(&x,params);
    }

```

```

    if (l < lold)
        step = -step/10;

    p += step;
}

if (i == MAXCALLS)
    return (-1);

maxx = x;

fy = 1/2;

// поиск x справа fy

p = maxx + params[0];
step = params[0];
lold = -2.0;
l    = -1e300;
i    = 0;

while ( (l != lold) && (i < MAXCALLS) ) {
    i++;

    lold = l;
    x = p + step;
    l = TMath::Abs(langaufun(&x,params) - fy);

    if (l > lold)
        step = -step/10;

    p += step;
}

```

```

}

if (i == MAXCALLS)
    return (-2);

fxr = x;

// поиск x с лева fy

p = maxx - 0.5 * params[0];
step = -params[0];
lold = -2.0;
l    = -1e300;
i    = 0;

while ( (l != lold) && (i < MAXCALLS) ) {
    i++;

    lold = l;
    x = p + step;
    l = TMath::Abs(langaufun(&x,params) - fy);

    if (l > lold)
        step = -step/10;

    p += step;
}

if (i == MAXCALLS)
    return (-3);

fxl = x;

```

```

    FWHM = fxr - fxl;
    return (0);
}

void langaus() {

    TGraph *Fh= new TGraph("/home/roger/summer_students/raspr/sci2fourstart.

    int Nbins=Fh->GetN();
    double MaxPointInHist;
    char *name="sa7sale100R55";
    char *note="note:";

    double *Ampl = Fh->GetY();
    double *Xi = Fh->GetX();
    double Yi2[Nbins];
    double Xi2[Nbins];
    double StartX;
    double EndX;
    int Nhelp=12;
    double Endhhelp=0;

    //заполнение и переверот исходных данных
    for(int gk=0;gk<Nbins;gk++)
        {
            Yi2[gk]=Ampl [Nbins-1-gk] ;
            Xi2[gk]=Xi [Nbins-1-gk]/(-50);
        }

    StartX= Xi2[0]*1e+9;
    EndX = Xi2[Nbins-1]*1e+9;
    Endhhelp=Xi2[Nhelp-1]*1e+9;
    double_t data[Nbins];

```

```

for(int gh=0; gh<Nbins; gh++)
{
    data[gh]=Yi2[gh];
    MaxPointInHist+=Yi2[gh];
}

/*
TH1F *hSNR = new TH1F("snr","Signal-to-noise",Nbins,0,Nbins);

    for (Int_t i=0; i<Nbins; i++) hSNR->Fill(i,data[i]);*/
//TFile f("Sci5RE.root","recreate");

    TH1D *hSNR = new TH1D("Parameter's","Amplitude spectrum sa7sale100R55;In
TH1D *helper = new TH1D("NOTH","noth",Nbins,StartX,EndX);
for(int y=1; y<=Nbins; y++)
    {helper->SetBinContent(y,0);
}
for(int j=1; j<=Nbins;j++) {
    hSNR->SetBinContent(j,Yi2[j-1]);
    hSNR->SetBinError(j,sqrt(Yi2[j-1]));}
for(int y=1; y<=Nhelp; y++)
    {helper->SetBinContent(y,Yi2[y-1]);
}

    double MVP;
MVP =hSNR->GetMean();
double RMS;
RMS=hSNR->GetRMS();
double MPVhelp;
MPVhelp =helper->GetMean();
double RMShelp;
RMShelp =helper->GetRMS();

```

```

// думаем
printf("Fitting...\n");

// начальные приближения
Double_t fr[2];
Double_t sv[4], pllo[4], plhi[4], fp[4], fpe[4];
//-----Тут вводить границы фитирования-----
fr[0]=0.25;//0.0*hSNR->GetMean();
fr[1]=0.45;//2.5*hSNR->GetMean();
//-----Заканчиваем вводить границы -----
//-----
//минимальные значения которые могут принимать ширина Ландау---
//МВП, Интеграл(Без учета ширины бина), Ширина Гауса-----
pllo[0]=0.0; pllo[1]=MVP/2; pllo[2]=0.0; pllo[3]=0.0;
//те же самые значения но уже максимальные
plhi[0]=1; plhi[1]=EndX; plhi[2]=MaxPointInHist*2; plhi[3]=1;
//те же самые значения но уже которые мы думаем что подойдут наилучшим
sv[0]= 0.016; sv[1]=MVP; sv[2]=MaxPointInHist; sv[3]=0.022;
cout << MaxPointInHist<< endl;

Double_t chisqr;
Int_t ndf;
//fp это 4 значения фита которые получились
//fpe это 4 значения ошибок фита
TF1 *fitsnr = langaufit(hSNR,fr,sv,pllo,plhi,fp,fpe,&chisqr,&ndf);

Double_t SNRPeak, SNRFWHM;
langapro(fp,SNRPeak,SNRFWHM);

printf("Fitting done\nPlotting results...\n");

// Global style settings
gStyle->SetOptStat(1111);

```

```

gStyle->SetOptFit(111);
gStyle->SetLabelSize(0.03,"x");
gStyle->SetLabelSize(0.03,"y");

//hSNR->GetXaxis()->SetRange(0,EndX);
//TCanvas *c1 = new TCanvas("c1");
hSNR->Draw();
helper->SetFillColor(kGreen);
helper->Draw("same");
// c1->SaveAs("c1.root");
// hSNR->Write();
//helper->Write();
// f.Close();

//-----какое гавнище-----
/* cout<<MVP<< endl;
cout<<RMS<< endl;
cout<<MPVhelp<< endl;
cout<<RMShelp<< endl;*/
//-----конец гавнища-----
fitsnr->Draw("lsame");
//-----???-----
double difLan;
difLan=sqrt(fp[0]*fp[0]-RMShelp*RMShelp);
double difGau;
difGau=sqrt(fp[3]*fp[3]-RMShelp*RMShelp);
double WG=sqrt(fp[0]*fp[0]+fp[3]*fp[3]);
double difMPV;
difMPV=fp[1]-MPVhelp;
//-----profit-----
//-----ВЫВОД-----
cout<<name<< endl;
cout<<note<< endl;
cout<<"fit range: "<< fr[0]<<" - "<<fr[1]<<endl;

```

```

cout<<"from hist"<< endl;
cout<<"Nhelp= "<<Nhelp<< endl;
cout<<"MEAN "<<MVP<< endl;
cout<<"RMS "<<RMS<< endl;
cout<<"MEANPED "<<MPVhelp<< endl;
cout<<"RMSPED "<<RMShelp<< endl;
cout<<"fit"<< endl;
cout<< "Width "<< fp[0]<<" +- "<<fpe[0]<< endl;
cout<< "MP "<< fp[1]<<" +- "<<fpe[1]<<endl;
cout<< "Area "<< fp[2]<<" +- "<<fpe[2]<<endl;
cout<<"GSigma "<< fp[3]<<" +- "<<fpe[3]<<endl;
cout<<"difference"<< endl;
cout<<"MP-MEANPED "<<difMPV<< " +- "<<fpe[1]<< endl;
cout<< "GSigma-RMSPED "<< difGau<<" +- "<<fpe[3]<< endl;
}

```