

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Физический факультет

Кафедра радиофизики

Изучение быстропротекающих
процессов с помощью
цифровых осциллографов
Лабораторная работа №5

(Обучающий комплекс для выполнения новых
лабораторных работ)

Новосибирск, 2012

Лабораторная работа посвящена изучению быстропротекающих процессов с помощью цифровых осциллографов на примере диагностики пучков заряженных частиц. В данном пособии рассмотрены принципы работы и основные характеристики цифровых осциллографов, описаны методики диагностики пучков в ускорителях, приведены сведения, необходимые для начала работы.

В процессе выполнения данной работы студенту нужно будет написать программу для регистрации формы сигнала в имитаторе пучка, определить положение пучка относительно центра датчика, а также изучить колебания пучка при внешнем воздействии. Полученные результаты представить в виде графиков.

Составители: Дементьев Е.Н., Бехтенов Е.А.

Обучающий комплекс для выполнения новых лабораторных работ подготовлен в рамках реализации Программы развития НИУ-НГУ на 2009–2018 г.г.

© Новосибирский государственный университет, 2012

1. Введение

Проведение экспериментов на современных физических установках требуют создания сложных систем сбора и обработки информации, поступающей с периферийных устройств. Во многих экспериментах существенную информацию о происходящих процессах несет форма сигнала, то есть поведение измеряемой величины во времени. Для регистрации и наблюдения формы сигналов применяются цифровые осциллографы, которые преобразуют аналоговый сигнал в цифровой вид. Полученные данные затем могут отправляться в компьютер для дальнейшей обработки и архивирования, либо визуализируются сразу. Кроме этого данные могут предварительно обрабатываться в режиме реального времени с помощью программируемой логики FPGA или DSP, это позволяет существенно уменьшить объём данных передаваемых в компьютер, отфильтровать полезный сигнал от шумов или организовать обратную связь с временем обработки сигнала порядка единиц наносекунд.

Целью данной работы является изучение принципов работы цифровых осциллографов, их основных характеристик и получение практических навыков работы с этими приборами. Для этого в данной работе будет использоваться стенд включающий в себя имитатор пучка заряженных частиц, датчик положения пучка (называемый пикапом – от английского pickup – датчик) с электроникой обрабатывающей данные с датчика, и кейт типа PXIe с осциллографом NI 5105 фирмы National Instruments.

В задачу студента входит написание в среде LabWindows/CVI программы, измеряющей форму сигнала с пластин датчика, положение пучка относительно центра пикапа, а также колебания пучка в результате внешнего воздействия. Полученные результаты необходимо представить в виде графиков.

2. Принцип работы и характеристики цифровых осциллографов

2.1. Принцип работы

На рис. 1 показано внутреннее устройство цифрового осциллографа. Он состоит из аналогового тракта (**А**), аналого-цифрового преобразователя (**АЦП**), оперативного запоминающего устройства (**ОЗУ**), схемы таймирования и управления (**Триггер&Таймер**), встроенного генератора тактовой частоты (**Генератор тактов**), интерфейса с магистралью системы (**Интерфейс**). Назначение аналогового тракта – приведение амплитуды

входного сигнала к шкале аналого-цифрового преобразователя (усиление или ослабление). Далее, АЦП производит выборки сигнала через равные промежутки времени и преобразует их в цифровой код. Полученные таким образом цифровые отсчёты запоминаются в ОЗУ. Схема таймирования задаёт частоту (частоту дискретизации), с которой выполняются выборки сигнала, и определяет режим работы (в простейшем случае - запись/чтение) устройства. Нетрудно догадаться, что данные, последовательно записываемые в ячейки памяти, представляют собой полученный с некоторой точностью цифровой эквивалент формы входного сигнала.

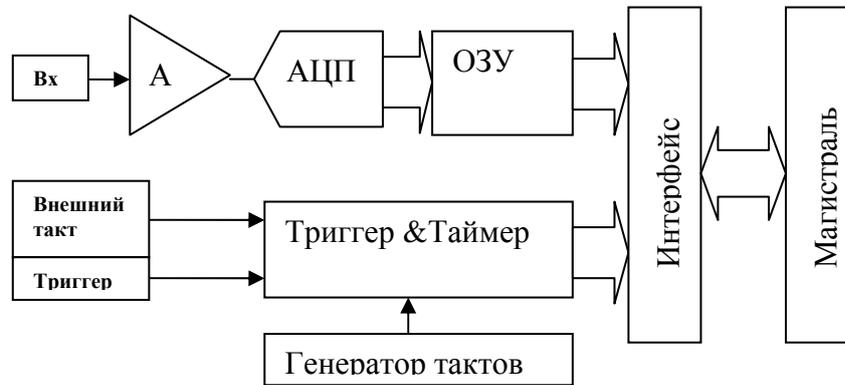


Рис.1. Структурная схема цифрового осциллографа

2.2. Характеристики

Важнейшие характеристики цифрового осциллографа определяются в первую очередь аналого-цифровым преобразователем. Характеристики АЦП детально рассматривались в «*Работе 4*», поэтому здесь будут описаны только те из них, что существенны в цифровых осциллографах. К ним относятся:

- производительность или частота выдачи отсчётов, измеряемые в MSPS (megasamples per second или миллионов выборок в секунду);
- эффективная (реальная) разрядность и её частотная зависимость;
- погрешности масштаба преобразования.

Цифровая часть прибора характеризуется параметрами и режимами работы памяти, а также возможностями триггерного узла. Эти характеристики будут рассмотрены в последующих разделах.

Производительность преобразователя, или частота дискретизации (Sample rate) – это количество отсчётов в секунду, обеспечиваемое им на выходе. Это понятие отражает быстродействие осциллографа. Величина, обратная производительности, задаёт интервал между отсчётами. Допустимая величина интервала (и производительность) определяется способами обработки и восстановления сигнала из произведённых аналого-цифровым преобразователем отсчётов. Чем сложнее алгоритм восстановления исходного сигнала по дискретным отсчётам, тем больший интервал между ними возможен. В соответствии с теоремой Котельникова сигнал можно однозначно восстановить по дискретным равноотстоящим отсчетам, идущим с интервалом не больше $0.5/F_{max}$, где F_{max} - максимальная частота в спектре сигнала (см. далее раздел 2.3).

Графики, показанные на рис.2, наглядно демонстрируют, что в случае использования линейной интерполяции при восстановлении сигнала, частота выборок для левого графика явно недостаточна, в то время как для правого может быть уменьшена.

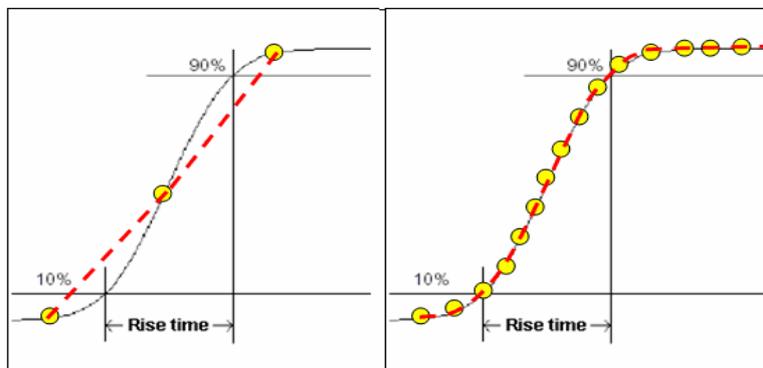


Рис 2. Влияние частоты выдачи отсчётов на точность восстановления сигнала

Величина **эффективной разрядности** является мерой шума, т.е. случайной ошибки, которую вносит АЦП цифрового осциллографа при преобразовании сигнала в код. В «Работе 4» приводилась формула, связывающая величину шума и значение эффективной разрядности:

$$SNR(dB) = 6.02 \cdot ENOB + 1.76,$$

$$\text{где } SNR = \frac{Signal}{Noise},$$

выраженное в децибелах, а *ENOB* – Effective Number of Bits. Эта обобщённая характеристика отражает влияние сразу нескольких источников ошибок: шума квантования, статической и динамической дифференциальной нелинейности, шумов внутренних узлов тракта преобразования, джиттера. Таким образом, знание эффективной разрядности позволяет просто определить ошибку оцифровки,

$$\text{как: } \pm \frac{1}{2} \cdot \frac{1}{2^{ENOB}}$$

Масштаб преобразования в общем случае определяется как:

$$G = \frac{DATA_{OUT} \cdot q}{U_{FS}},$$

где *G* – масштаб преобразования, *DATA_{OUT}* – выходной код при напряжении на входе *U_{FS}* (FS- Full Scale – полная шкала) равном максимальному, а *q* – вес младшего разряда преобразователя. В идеальном случае *G* не может отличаться от 1 больше, чем на $\pm \frac{1}{2} \cdot \frac{1}{2^N}$, где *N*

разрядность преобразования. Таким образом, масштаб преобразования – это наклон характеристики преобразования. В реальных условиях, за счёт многих факторов, ошибка масштаба превышает половину младшего разряда. В цифровых осциллографах статическая ошибка масштаба не приводит к искажению формы зарегистрированного сигнала и поэтому не столь важна. Она может быть очень просто учтена при последующей обработке умножением данных на необходимый коэффициент. В то же время частотная зависимость масштаба, приводящая к искажению формы вследствие искажения спектра сигнала, гораздо существеннее.

Для описания частотной зависимости в цифровой осциллографии, также как и в аналоговой, принято использовать термин «полоса пропускания».

Напомним, что полоса пропускания (Bandwidth) описывает частотный диапазон входного сигнала, в котором сигнал проходит без ослабления. Границу полосы пропускания определяют как частоту, на которой входной

сигнал ослабляется на 3 дБ или до уровня 0.7. Влияние полосы пропускания на передаваемый сигнал представлено на рис. 3.

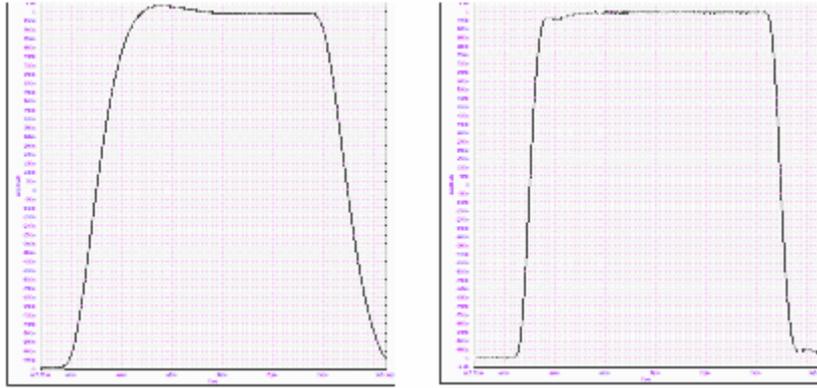


Рис. 3. Влияние полосы пропускания осциллографа на форму сигнала. На левой осциллограмме полоса пропускания 20 МГц, на правой - 300 МГц

В общем случае рекомендуется использовать осциллограф с полосой пропускания в несколько раз больше, чем спектр измеряемого сигнала. Например, сигналы с малыми временами нарастания, такие как цифровые импульсы или сигналы сложной формы с короткими фронтами имеют частотный спектр с заметной долей высокочастотных компонентов. Иногда вместо «полосы пропускания» используют понятие «время нарастания» (rise time), определяемое как интервал, необходимый для перехода от 10 до 90% амплитуды сигнала (см. рис.4).

Время нарастания сигнала и полоса пропускания непосредственно связаны по формуле

$$t_r = \frac{0.35}{F_{07}},$$

где F_{07} – полоса пропускания осциллографа, t_r – время нарастания.

При известных характеристиках осциллографа, измеренное время нарастания связано с реальным соотношением:

$$t_m = \sqrt{t_s^2 + t_d^2},$$

где t_m – измеренное время, t_s – время нарастания фактического сигнала, t_d – собственное время нарастания осциллографа.

Обычно скорости, с которыми АЦП регистрирует сигнал, превосходят возможности канала связи с компьютером. Буферизация данных в оперативной памяти осциллографа и позволяет накапливать информацию

синхронно с работой АЦП. **Важнейшей характеристикой ОЗУ** является его ёмкость, определяемая как максимальное количество слов, которое может быть записано. Разрядность слова данных памяти в большинстве случаев совпадает с разрядностью АЦП. Произведение емкости памяти на интервал между отсчетами определяет максимальную длительность регистрируемого процесса. Большой объем памяти в сочетании с высокой частотой выборки даёт возможность подробно исследовать временную структуру длительных сигналов.



Рис. 4. Определение времени нарастания сигнала

Триггерный узел позволяет синхронизировать исследуемые сигналы с процессом регистрации в цифровом осциллографе. В этой связи необходимо обратить внимание на два момента: что является источником триггерного сигнала и как расположен во времени этот сигнал относительно регистрируемого процесса. Под триггерным сигналом понимается импульс (или команда от компьютера), вызывающий начало/окончание регистрации. Существуют три источника триггерного сигнала. Первый – это сам исследуемый сигнал, когда *триггерный импульс образуется непосредственно из этого сигнала внутри устройства*. Вторым – *внешний импульс*, связанный с регистрируемым сигналом. Третий – команда от компьютера, запускающая осциллограф и одновременно исследуемый процесс.

Во времени триггерный сигнал может подаваться до начала исследуемого процесса (pre-trigger mode) и после его окончания (post-

trigger mode). Первая мода применяется тогда, когда точно известно, что исследуемый сигнал возникнет в течение времени, определяемым длиной памяти, и соответственно, будет зарегистрирован. Вторая мода применима тогда, когда время возникновения интересующего процесса неизвестно (например, сбой в работе или авария). ОЗУ в этой моде переводится в такой режим работы, когда запись происходит «по кругу», т.е. после записи данных в последнюю ячейку, выполняется запись в первую, вторую и т.д. Возникновение интересующего процесса, вызывает появление триггерного сигнала, который останавливает запись, т.е. функция триггера в данном случае – остановка, а не старт. Таким способом удаётся зарегистрировать предисторию процесса и наблюдать развитие аварийных ситуаций.

2.3. Особенности цифровых осциллографов

Возможность организации многоканальных измерений (от 2-х каналов и более) является одним из важных качеств цифровых осциллографов. Во многих задачах необходимо исследовать ряд параллельно протекающих процессов. Существуют два основных метода увеличения количества каналов – это мультиплексирование и построение параллельных автономных каналов.

В первом случае используется один аналого-цифровой преобразователь, на вход которого при помощи коммутатора поочередно подключаются различные источники сигналов. Коммутатор работает синхронно с тактовой частотой АЦП, и в память осциллографа поочередно заносятся результаты измерения от различных источников. Достоинством такого метода является простота реализации, недостатком – потеря эффективной частоты выборок и объема памяти для каждого канала.

Во втором методе в состав осциллографа входят несколько одинаковых и синхронно работающих трактов преобразования, каждый из которых обрабатывает свой сигнал. Достоинством этого метода является полная синхронность отсчетов, а недостатком – возрастание сложности и объема электроники. Вследствие этого, количество каналов в «синхронных» осциллографах как правило не превышает восьми.

Следует обратить внимание ещё на одно свойство цифровых осциллографов. Полоса пропускания в распространённых моделях находится в диапазоне от нескольких десятков, до нескольких сотен мегагерц. В то же время частота взятия отсчетов может устанавливаться пользователем, и вполне возможна ситуация, когда для достаточно высокочастотного сигнала выбран темп постановки отсчетов ниже, чем F_{07} . К чему это приводит, можно продемонстрировать на примере. Если в качестве входного сигнала взять периодический сигнал, а частоту выборки сделать очень близкой к частоте входного сигнала, как это показано на

рис. 5. Из рисунка видно, что по отсчётам, записанным в память, будет восстановлен сигнал, совпадающий по форме с синусоидальным, но с частотой равной разности между входной и частотой выборки. Такой режим иногда называют стробоскопическим.

Если же входной сигнал не периодичен, а темп выборок ниже, чем максимальные частоты в спектре сигнала, то это приведёт к искажению формы при восстановлении, хотя АЦП и не внесёт ошибок при преобразовании в код. Вопросам соотношения спектра сигнала и частот дискретизации и связанным с ними ошибками посвящены классические работы, заложившие основы теории информации [3].

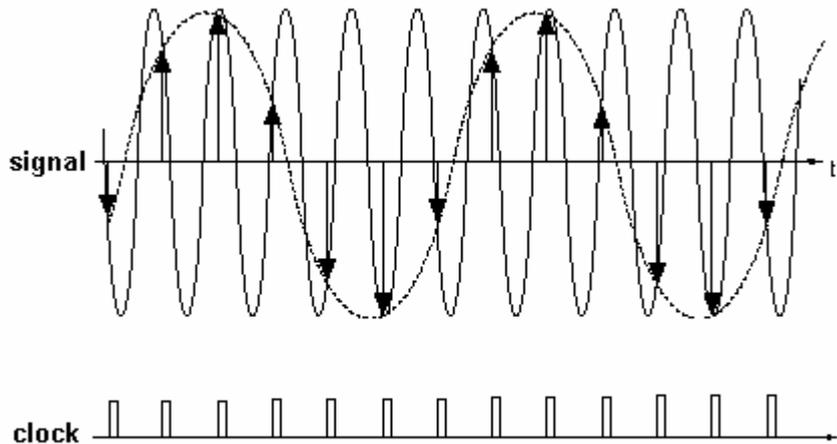


Рис. 5. Стробоскопическое осцилографирование

3. Диагностика пучков в циклических ускорителях

Одной из задач, решаемых при помощи цифровых осциллографов, является измерение параметров пучка в циклических ускорителях заряженных частиц. Сигналы с датчиков положения пучка представляют собой короткие импульсы, следующие с частотой обращения пучка. Из этих сигналов получают информацию о токе пучка, вертикальной и горизонтальной координатах, фазе и различного типа колебаниях. Обычно для измерения параметров пучка применяется система из четырёх электродов, называемых электростатическим пикапом (рис 6).

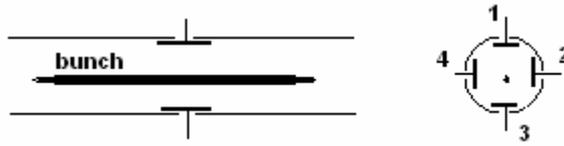


Рис. 6. Электростатический пикап

Каждый электрод пикапа представляет собой металлическую пластину, изолированную от камеры вакуумным вводом. Пролетающий сгусток заряженных частиц на этих электродах индуцирует сигналы, амплитуда и длительность которых, зависит от тока пучка, его смещения относительно центра вакуумной камеры и параметров пикапа. Сигнал с каждой пластины представляет собой короткий импульс с длительностью, зависящей от геометрических размеров пикапа и длины сгустка. В случае, когда длина сгустка намного больше размера пластины, длительность сигнала можно оценить соотношением:

$$t_b = \frac{l_b}{V},$$

где t_b – длительность сигнала, l_b – длина сгустка, V – скорость сгустка, для релятивистских частиц это скорость света.

В большинстве случаев, это время лежит в наносекундном и субнаносекундном диапазоне, это означает, что спектр сигналов простирается до десятков гигагерц. Для согласования амплитудного и частотного диапазонов регистрирующей аппаратуры и измеряемого сигнала применяются полосовые фильтры-усилители. Фильтрация сигнала позволяет также оптимизировать соотношение сигнал-шум.

Ускоритель - это большая и сложная физическая установка, чтобы она правильно работала, необходима синхронная работа всей аппаратуры. Для этого формируется сетка опорных частот, для создания которой используется один эталонный генератор. Чаще всего таким источником эталонной частоты выступает генератор, питающий ускоряющую структуру, для краткости именуемый генератором Высокой Частоты (ВЧ генератор). Поделив его частоту в нужное количество раз можно получать набор опорных частот для всех нужд ускорительного комплекса. Одна из таких опорных частот это частота равная частоте обращения пучка. На протяжении периода частоты обращения укладывается целое количество периодов частоты ВЧ. Каждый период ВЧ соответствует возможному устойчивому положению пучка, называемому сепаратрисой. Для удобства сепаратрисы нумеруют, начиная с 0. Момент прихода импульса частоты

обращения соответствует началу оборота и сбросу счётчика периодов ВЧ (сепаратрис) в 0, поэтому часто этот импульс называют 0-сепаратриса. В больших физических установках отношение частоты 0-сепаратрисы к частоте ВЧ может достигать от нескольких сотен до десятков тысяч. Изготовить задержку, которая работала бы с необходимой точностью на всем диапазоне периода обращения, просто невозможно. Поэтому задержку делают двухэтапной. На первом этапе (грубая задержка) счётчик осчитывает заданное количество импульсов ВЧ относительно импульса 0-сепаратрисы и дает сигнал для второго этапа задержки. На втором этапе (точная задержка) генерируется задержанный импульс в пределах одного периода ВЧ. Дискретность задержки точного этапа составляет от одной десятой до тысячных долей периода ВЧ.

3.1. Стационарная орбита.

Циркулирующий по орбите ускорителя, электронный пучок наводит на стенки вакуумной камеры заряд, представляющий собой зеркальное отображение самого пучка. При пролете пучка сквозь электростатический пикап, часть этого заряда попадает на его электроды, тем самым инициируя изменение их потенциала. Напряжение на электроде пикапа зависит от емкости пластины и наведенного заряда.

$$U_n = \frac{Q_n}{C}$$

Q_n – заряд на электроде, C – ёмкость электростатического датчика, n – номер пластины. В свою очередь заряд электрода пропорционален току пучка и зависит от его положения относительно геометрического центра как

$$Q_n = K(r) * Q$$

Q – заряд пучка в пределах пикапа, $K(r)$ – коэффициент, описывающий зависимость распределение наведенного заряда от положения пучка относительно геометрического центра камеры, r – смещение пучка. При малых смещениях, функция $K(r)$ близка к линейной. Таким образом, смещение пучка по радиусу и вертикали может быть вычислено из уравнений

$$X_r = K(0) * R_0 \frac{U_2 - U_4}{U_2 + U_4}$$

$$Z_r = K(0) * R_0 \frac{U_1 - U_3}{U_1 + U_3}$$

В ряде случаев пластины пикапов располагаются смещенными относительно горизонтальной плоскости на 45 градусов. В этом случае обычно применяется соотношения:

$$X_r = K(0) * R_0 \frac{(U_1 + U_2) - (U_3 + U_4)}{U_1 + U_2 + U_3 + U_4}$$

$$Z_r = K(0) * R_0 \frac{(U_1 + U_4) - (U_2 + U_3)}{U_1 + U_2 + U_3 + U_4},$$

где X_r, Z_r – смещение пучка по горизонтали и вертикали, R_0 -полуапертура вакуумной камеры, U_1, U_2, U_3, U_4 – напряжения с вертикальных и горизонтальных пластин. При больших отклонениях пучка от центра, необходимо учитывать нелинейность функции $K(r)$. Задача определения координат электронного сгустка сводится к измерению амплитуд сигналов на электродах пикапа с последующей обработкой результатов. Для этого необходимо усилить и отфильтровать сигнал аналоговыми цепями и преобразовать его в цифровую форму для компьютерной обработки. Аналоговая обработка приводит частотный и амплитудный спектры сигналов в соответствие с техническими характеристиками цифровых осциллографов.

3.2. Динамика пучка

Воздействие внешних возмущений, а также взаимодействие пучка с магнитной и высокочастотной ускоряющей структурами, приводят к колебаниям его положения относительно равновесной орбиты, т.е. усредненной за много оборотов траектории частиц. Устойчивое движение электронного сгустка возможно только при прохождении его сквозь ускоряющую структуру в узком диапазоне фаз одного из периодов напряжения на резонаторе. Флуктуации фазы пучка относительно равновесной приводят к изменению его энергии, а это в свою очередь влияет на его траекторию и приводит к синхротронным колебаниям.

Отклонение поперечной координаты или импульса частиц от равновесной орбиты приводит к возникновению бетатронных колебаний, когда пучок циркулирует совершая многократные колебания относительно равновесной орбиты заданной фокусирующей структурой ускорителя.

Особый интерес представляет режим инжекции, когда в результате импульсного воздействия со стороны системы впуска, пучок начинает совершать когерентные т.е. одновременные для всех частиц, затухающие колебания. Для изучения свойств ускорителя в таких случаях используется режим многооборотного наблюдения динамики пучка. Для этого при помощи кикера возбуждаются колебания пучка, а затем его координаты регистрируются на каждом обороте с последующим вычислением спектра колебаний.

4. Описание стенда

В данной работе используется стенд, имитирующий систему измерения параметров электронного пучка в ускорителе заряженных частиц.

Стенд состоит из четырехэлектродного пикапа, иммитатора пучка, четырехканального фильтра – усилителя и схемы синхронизации.

Функциональная схема стенда изображена на рис. 7. общий вид рис. 9.

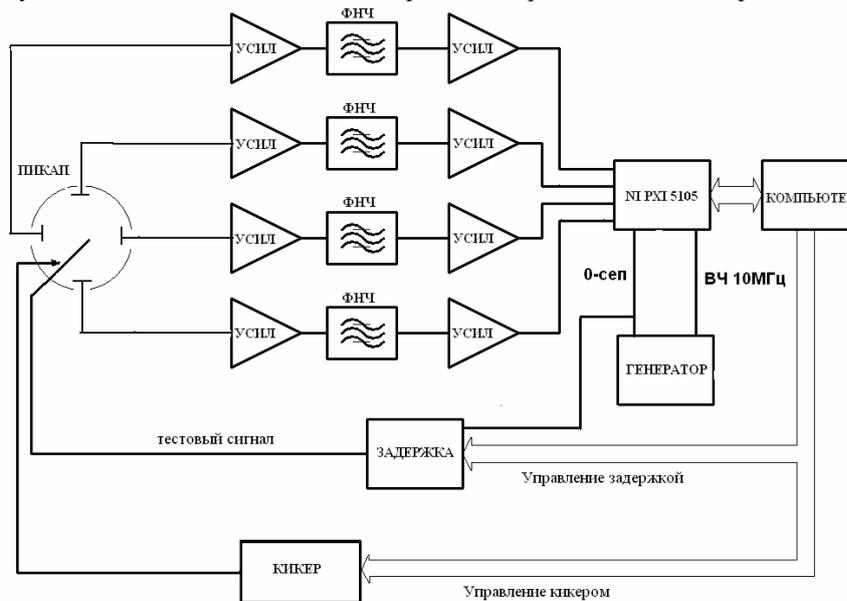


Рис. 7. Функциональная схема стенда

Имитатор пучка состоит из импульсного генератора, проводника, формирующего электрическое поле на пластинах пикапа и схемы стимулирующей колебания пучка при внешнем воздействии (кикер). Сигналы с пластин пикап – электродов поступают на входы фильтров – усилителей, а с их выходов – на четыре идентичных входа цифрового

осциллографа. Схема синхронизации формирует два внешних тактовых сигнала осциллографа, сигнал высокой частоты ВЧ – 10 МГц, и пооборотный сигнал, являющийся сигналом 0-сепаратрисы. Импульс 0-сепаратрисы имеет частоту в 16 раз меньше частоты ВЧ. Эти сигналы сдвинуты по времени относительно иммитатора пучка на величину, определяемую программно-управляемой линией задержки. Сигнал ВЧ подается на вход опорной частоты PFI1 модуля NI PXI-5105. Сигнал 0-сепаратрисы подается на вход СН7, и используется в качестве импульса внешнего запуска.

В структуре стенда отсутствует грубая задержка, есть только точная задержка, которая обеспечивает задержку только в пределах одного периода ВЧ. Но благодаря небольшому отношению частот ВЧ и импульса 0-сепаратрисы существует возможность захватить не одну сепаратрису, а все 16. Затем программно выбрав нужную сепаратрису, в которой находится полезный сигнал.

Управление линией задержки осуществляется через порт 0.

Младшие 8 бит регистра управляют задержкой синхроимпульса. Вес младшего бита – 5 наносекунд. Для кода задержки используется инверсный код. Нулевой бит порта 1 инициализирует кикер. Удар по проволоке происходит при изменении значения бита с 0 на 1. Для запрета работы устройств подключенных к шине АВАЛОН в биты 0-2 порта 2 необходимо записать 0, см. Рис. 8.

	line7	line6	line5	line4	line3	line2	line1	line0
port0	инвертированный код задержки							
port1								удар
port2						0	0	0

Рис. 8. Схема портов управления

5. Задания

1. Изучить форму сигнала с пикапа в полосе частот 60 МГц, сначала используя внутренний запуск, а затем используя внешний запуск (сигнал 0-сепаратриса).
2. Получить сигнал с пикапов, используя частоту синхронизации и импульс запуска от блока электроники пикапа.
3. Найти номер сепаратрисы, в которой присутствует полезный сигнал, и настроить линию задержки.
4. Вычислить координаты X, Y и ток пучка.
5. Определить геометрический множитель для обеих осей координат, выполнив сдвиг пикапа на некоторое известное расстояние – 10-20 мм.
6. Ударьте по пучку с помощью кикера и получите спектр колебаний.

Дополнительные задания

* Отфильтруйте цифровым образом координаты X, Y от шума.

***** Проведите измерение переходного процесса колебаний пучка в непрерывном режиме. Для этого: установите коэффициент деления частоты синхронизации равным 3. В режиме непрерывного захвата триггер запуска по фронту сигнала не доступен, поэтому установите режим запуска от «программного» триггера. Изучите сигнал 0-сепаратрисы, определите смещения до полезного сигнала. Отфильтруйте полезные данные и вычислите координаты пучка и спектр колебаний. Определите время затухания колебаний.

6. Литература

1. Б.В. Керниган, Д.М. Ричи. Язык С.
2. Н. Nyquist. Certain Factors Affecting Telegraph Speed. // Bell System Technical Journal, Vol. 3, April 1924, pp. 324-346.
<http://bstj.bell-labs.com/BSTJ/images/Vol03/bstj3-2-324.pdf>
3. Котельников В.А. О пропускной способности эфира и проволоки в электросвязи. // Всесоюзный энергетический комитет. Материалы к I Всесоюзному съезду по вопросам технической реконструкции дела связи и развития слаботочной промышленности, 1933г.
4. С.Е. Shannon. A Mathematical Theory of Communication. // Bell System Technical Journal, Vol. 27, July 1948, pp. 379-423. Русский перевод в кн.: К.Шеннон. Работы по теории информации и кибернетике. М.: ИЛ, 1963, стр. 243-332.

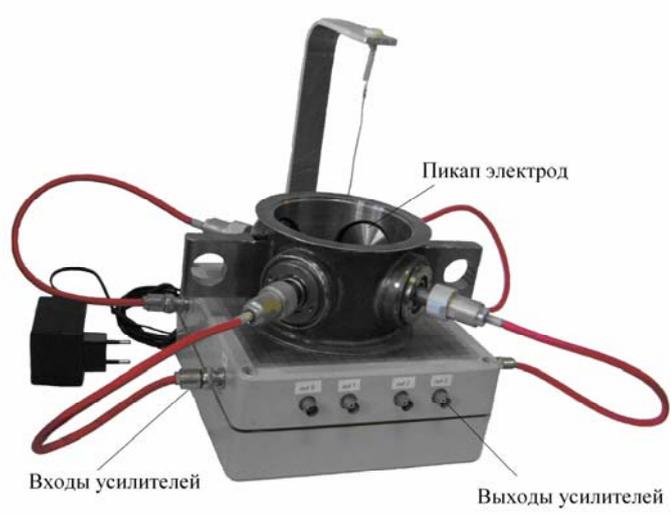


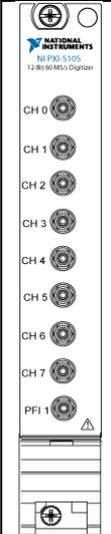
Рис. 9. Стенд Пикап

7. Приложение: Цифровой осциллограф NI PXI-5105

основные особенности NI PXI-5105

- 8 входных каналов, измеряемых одновременно
- вертикальное разрешение 12 бит
- максимальная частота дискретизации 60 МГц
- частота дискретизации задается от внутреннего или внешнего источника
- старт измерения от внутреннего источника, внешнего источника, программный
- аналоговая полоса входного фильтра 60 МГц или 25 МГц
- 16 Мбайт внутренней памяти
- входное сопротивление 50 Ω или 1 М Ω .
- для входного сопротивления 1 М Ω можно выбрать режим по постоянному DC или переменному AC току
- диапазон входных напряжений от пика до пика 0.05В, 0.2В, 1В, 6В для входных сопротивлений 50 Ω и 1 М Ω и 30В для 1 М Ω .
- шина данных NI PXI-5105 – PC 66МГц

Вид передней панели NI PXI-5105

 <p>NI PXI-5105 12-BIT DIGITAL OSCILLOSCOPE</p> <p>CH 0 CH 1 CH 2 CH 3 CH 4 CH 5 CH 6 CH 7 PFI 1</p>	<table border="1"> <thead> <tr> <th>назначение входов</th> </tr> </thead> <tbody> <tr> <td>входы с СНО по СН7 исследуемый сигнал и/или триггер запуска</td> </tr> <tr> <td>PFI1 вход частоты дискретизации</td> </tr> </tbody> </table>	назначение входов	входы с СНО по СН7 исследуемый сигнал и/или триггер запуска	PFI1 вход частоты дискретизации
назначение входов				
входы с СНО по СН7 исследуемый сигнал и/или триггер запуска				
PFI1 вход частоты дискретизации				

Концепция работы с NI PXI-5105:

- инициализация
- настройка диапазона измеряемых входов
- настройка типа входа
- настройка частоты и источника синхронизации
- настройка триггера старта
- резервирование памяти для данных
- измерение
- освобождение ресурсов

Список функций библиотеки niScope для работы с NI PXI-5105

Прототип Функции	назначение	параметры
инициализация и освобождение		
ViStatus niScope_init (ViRsrc resourceName , ViBoolean IDQuery , ViBoolean resetDevice , ViSession* vi);	Создание новой сессии для работы с устройством	resourceName – имя ресурса IDQuery – выполнять идентификационный запрос resetDevice – сброс устройства vi – адрес дескриптора сессии
ViStatus niScope_close (ViSession vi);	освободить ресурсы связанные с дескриптором и закрыть дескриптор	vi – дескриптор сессии
настройка измерительных каналов		
ViStatus niScope_ConfigureVertical (ViSession vi , ViConstString channelList , ViReal64 range , ViReal64 offset , ViInt32 coupling , ViReal64 probeAttenuation , ViBoolean enabled);	настройка диапазона входных напряжений	vi – дескриптор сессии channelList – список настраиваемых каналов, номер канала цифра от 0 до 7, чтобы задать несколько каналов перечислите их разделяя запятыми – "3,1" range – диапазон входных напряжений в вольтах, например для входных напряжений $\pm 5\text{В}$ диапазон 10 offset – смещение входных напряжений в вольтах coupling – тип входа по постоянному или переменному току probeAttenuation – коэффициент ослабления внешней цепи, если подключено напрямую - 1 enabled – разрешение для захвата данных

ViStatus niScope_ConfigureChanCharacteristics (ViSession vi, ViConstString channelList , ViReal64 inputImpedance , ViReal64 maxInputFrequency);	настройка входного сопротивления заданных каналов	vi – дескриптор сессии channelList – список каналов inputImpedance – входное сопротивление, 50 Ом или 1 МОм maxInputFrequency – максимальная входная частота, Гц
настройка триггера старта		
ViStatus niScope_ConfigureTriggerImmediate (ViSession vi);	программный запуск измерения	vi – дескриптор сессии
ViStatus niScope_ConfigureTriggerEdge (ViSession vi, ViConstString triggerSource , ViReal64 level , ViInt32 slope , ViInt32 triggerCoupling , ViReal64 holdoff , ViReal64 delay);	настройка запуска измерений по фронту сигнала на заданном входе	vi – дескриптор сессии triggerSource – канал используемый для старта измерений level – уровень срабатывания триггера в вольтах slope – выбор фронта triggerCoupling – тип входа по постоянному или переменному току holdoff – мертвое время, время ожидания после срабатывания прежде чем произойдет следующее срабатывание, сек delay – задержка начала измерения после срабатывания триггера в сек
настройка частоты измерения и источника синхронизации		
ViStatus niScope_ConfigureHorizontalTiming (ViSession vi, ViReal64 minSampleRate , ViInt32 minNumPts , ViReal64 refPosition , ViInt32 numRecords , ViBoolean enforceRealtime);	установить частоту дискретизации, положение сигнала запуска и длину данных.	vi – дескриптор сессии minSampleRate – частота дискретизации, Гц minNumPts – длина записи refPosition – положение триггера запуска в процентах от длительности записи numRecords – количество записей enforceRealtime – измерять в режиме реального времени (NISCOPE_VAL_FALSE)
ViStatus niScope_SetAttributeViString (ViSession vi, ViConstString channelList , ViAttr attributeID , ViConstString value);	установка канала внешней синхронизации	vi – дескриптор сессии channelList – не используется, установить VI_NULL attributeID – для установки внешней синхронизации установить равным NISCOPE_ATTR_SAMP_CLK_TIMEBASE_SRC value – вход внешней синхронизации

ViStatus niScope_SetAttributeViReal64 (ViSession vi , ViConstString channelList , ViAttr attributeID , ViReal64 value);	установка при- близительной частоты внешней синхронизации	vi – дескриптор сессии channelList – не используется, установить VI_NULL attributeID – для установки частоты внешней синхронизации установить равным NISCOPE_ATTR_SAMP_CLK_TIMEB ASE_RATE value – частота, Гц
запрос фактических значений		
ViStatus niScope_ActualNumWfms (ViSession vi , ViConstString channelList , ViInt32* numWfms);	прочитать количество каналов измерения.	vi – дескриптор сессии channelList – строка со списком каналов numWfms – количество каналов
ViStatus niScope_ActualRecordLength (ViSession vi , ViInt32* recordLength);	прочитать длину данных.	vi – дескриптор сессии recordLength – адрес для записи длины данных
ViStatus niScope_SampleRate (ViSession vi , ViReal64* actualSampleRate);	прочитать установленную частоту измерений.	vi – дескриптор сессии actualSampleRate – адрес для записи частоты дискретизации
получение данных		
ViStatus niScope_Read (ViSession vi , ViConstString channelList , ViReal64 timeout , ViInt32 numSamples , ViReal64* wfm , struct niScope_wfmInfo* wfmInfo);	запустить измерение, дождаться окончания измерения, прочитать данные.	vi – дескриптор сессии channelList – строка со списком каналов timeout – таймаут измерения, сек numSamples – количество данных wfm – массив для записи данных wfmInfo – массив структур с информацией о измерении
struct niScope_wfmInfo	структура описания прочитанных данных	relativeInitialX – время в секундах от момента срабатывания триггера запуска absoluteInitialX – время в секундах для первого отсчета выборки от начала измерения, в случае непрерывных измерений xIncrement – шаг по времени между последовательными отсчетами actualSamples – длина выборки данных gain и offset - множитель и смещение для вычисления сигнала по формуле: $voltage = binary\ data \times gain\ factor + offset$
ViReal64* wfm	полученные данные	данные представляют собой numWfms массивов длиной actualSamples точек типа ViReal64

получение данных в непрерывном режиме		
ViStatus niScope_SetAttributeViInt32 (ViSession vi , ViConstString channelList , ViAttr attributeID , ViInt32 value);	задать режим указателя данных модуля для работы в непрерывном режиме	vi – дескриптор сессии channelList – не используется, установить VI_NULL attributeID – для установки режима работы указателя установить равным NISCOPE_ATTR_FETCH_RELATIVE_TO value – режим работы указателя - NISCOPE_VAL_READ_POINTER
ViStatus niScope_SetAttributeViInt32 (ViSession vi , ViConstString channelList , ViAttr attributeID , ViInt32 value);	установка делителя частоты дискретизации	vi – дескриптор сессии channelList – не используется, установить VI_NULL attributeID – для установки частоты дискретизации установить равным NISCOPE_ATTR_SAMP_CLK_TIMEBASE_DIV value – коэффициент деления
ViStatus niScope_ConfigureTriggerSoftware (ViSession vi , ViReal64 holdoff , ViReal64 delay);	задать триггер запуска непрерывного режима (программный)	vi – дескриптор сессии holdoff – мертвое время, время ожидания после срабатывания прежде чем произойдет следующее срабатывание, сек delay – задержка начала измерения после срабатывания триггера в сек
ViStatus niScope_InitiateAcquisition (ViSession vi);	начать захват данных	vi – дескриптор сессии
ViStatus niScope_Fetch (ViSession vi , ViConstString channelList , ViReal64 timeout , ViInt32 numSamples , ViReal64* wfm , struct niScope_wfmInfo* wfmInfo);	прочитать захваченные данные	vi – дескриптор сессии channelList – строка со списком каналов timeout – таймаут измерения, сек numSamples – количество данных wfm – массив для записи данных wfmInfo – массив структур с информацией о измерении
ViStatus niScope_FetchBinary16 (ViSession vi , ViConstString channelList , ViReal64 timeout , ViInt32 numSamples , ViInt16* wfm , struct niScope_wfmInfo* wfmInfo);	прочитать захваченные данные в формате ViInt16	vi – дескриптор сессии channelList – строка со списком каналов timeout – таймаут измерения, сек numSamples – количество данных wfm – массив для записи кода данных, используя поля gain и offset структуры wfmInfo можно перевести код в вольты wfmInfo – массив структур с информацией о измерении
ViStatus niScope_Abort (ViSession vi);	остановить захват данных	vi – дескриптор сессии

Примеры

Обработка ошибок

Обработка ошибок важная составляющая любой надежно работающей компьютерной программы. Особенно в случае не тривиального и громоздкого библиотечного интерфейса, когда важно понять в каком месте и по какой причине что-то в исполнении программы пошло не так. Библиотека **NI niScope** предлагает следующие средства для удобной обработки ошибок.

Для этого служит макрос `handleErr`. Данный макрос определен в заголовочном файле `<niScope.h>`,

```
#define handleErr(fCall) { int code = (fCall);           \
    if (code < 0) {                                     \
        error = code;                                  \
        strncpy( errorSource, #fCall, MAX_FUNCTION_NAME_SIZE ); \
        errorSource[MAX_FUNCTION_NAME_SIZE - 1] = '\0'; \
        strncpy( errorSource, strtok( errorSource, " (" ) ); \
        goto Error; }                                   \
    else if ((error == 0) && (code > 0)) {              \
        error = code;                                  \
        strncpy( errorSource, #fCall, MAX_FUNCTION_NAME_SIZE ); \
        errorSource[MAX_FUNCTION_NAME_SIZE - 1] = '\0'; \
        strncpy( errorSource, strtok( errorSource, " (" ) ); } }
```

Макрос работает следующим образом: все функции из состава библиотеки **niScope** возвращают целочисленный статус завершения (см. таб. 1.), если статус завершения обозначает что во время вызова функции произошла ошибка, то макрос передает управление программой на метку **Error**, предварительно заполнив переменную `error` кодом ошибки и переменную `errorSource` подробным описанием ошибки. После метки **Error** вам нужно разместить код, осуществляющий корректное завершение вашей функции – освободить все занятые ресурсы.

Таблица 1 Значение статуса завершения

статус завершения	значение
меньше 0	ошибка
равно 0	успех
больше 0	предупреждение

При написании программы также может быть полезна функция `ProcessSystemEvents()`, которая позволяет обработать события из очереди сообщений программы, будучи вызвана из функции обратного вызова.

```
void имяФункции()
{
    ViStatus error = VI_SUCCESS;
    ViChar  errorMessage[MAX_ERROR_DESCRIPTION] = " ";
    ViChar  errorSource[MAX_FUNCTION_NAME_SIZE];
    handleErr (niScope_init (.....&vi));           // открытие дескрипторов
    handleErr (niScope_somefunction (...));        // вызов библиотечной функции
    mem = malloc(...);                             // резервирование памяти
    ....                                           // выполнение полезной работы
    while(run) {                                   // локальный рабочий цикл
        ....                                       // выполнение полезной работы
        ProcessSystemEvents();                    // обработка очереди сообщений
    }
    Error:
    if(mem)
        free(mem);                               // освобождение выделенной памяти
    if (error != VI_SUCCESS) {                    // вывод диагностики ошибки
        niScope_errorHandler(vi, error, errorSource, errorMessage);
        MessagePopup("error", errorMessage);
    }
    if (vi)                                       // закрытие открытых дескрипторов
        niScope_close (vi);
}
```

Работа с NI PXI-5105

Создаем и закрываем дескриптор

```
ViSession vi;
niScope_init ("PXI1Slot3", NISCOPE_VAL_FALSE, NISCOPE_VAL_FALSE, &vi);
niScope_close (vi);
```

Конфигурируем входы

```
niScope_ConfigureVertical (vi, "0,1,2", 10, 0, NISCOPE_VAL_DC, 1.0, NISCOPE_VAL_TRUE);
```

Конфигурируем параметры дискретизации

```
niScope_ConfigureHorizontalTiming (vi, 1e6, 1000, 5, 1, NISCOPE_VAL_FALSE);
```

Вслучае использования внешней синхронизации задаем источник и частоту сигнала синхронизации.

```
niScope_SetAttributeViString (vi, VI_NULL, NISCOPE_ATTR_SAMP_CLK_TIMEBASE_SRC,  
timebaseSource);  
niScope_SetAttributeViReal64 (vi, VI_NULL,  
NISCOPE_ATTR_SAMP_CLK_TIMEBASE_RATE, timebaseRate));
```

Настраиваем триггер внешнего старта по фронту

```
niScope_ConfigureTriggerEdge (vi, triggerSource, 1, triggerSlope, triggerCoupling,  
triggerHoldoff, triggerDelay);
```

Получаем количество измеряемых каналов, длину выборки, реально установленную частоту дискретизации

```
niScope_ActualNumWfms (vi, channelName, &numWaveform);  
niScope_ActualRecordLength (vi, &actualRecordLength);  
niScope_SampleRate (vi, &actualSampleRate);
```

Резервируем память для данных

```
wfmInfoPtr = malloc (sizeof (struct niScope_wfmInfo) * numWaveform);  
scaledWfmPtr = malloc (sizeof (ViReal64) * actualRecordLength * numWaveform);
```

Запускаем измерения и читаем данные

```
niScope_Read (vi, "0,1,2", 10, actualRecordLength, scaledWfmPtr, wfmInfoPtr);
```

Исправляем ошибки.

Оглавление

1.	Введение	3
2.	Принцип работы и характеристики цифровых осциллографов.....	3
2.1.	Принцип работы	3
2.2.	Характеристики	4
2.3.	Особенности цифровых осциллографов	9
3.	Диагностика пучков в циклических ускорителях.....	10
3.1.	Стационарная орбита.	12
3.2.	Динамика пучка	13
4.	Описание стенда	14
5.	Задания	15
6.	Литература	16
7.	Приложение: Цифровой осциллограф NI PXI-5105	18
	Примеры.....	23
	Работа с NI PXI-5105.....	24