Министерство образования и науки Украины

Харьковский национальный университет радиоэлектроники

Кафедра ТКС

Курсовая работа

пояснительная записка

“Разработка микропроцессорного устройства цифрового фильтра”

по курсу “Цифровые устройства и микропроцессоры ”

Выполнил: ст. гр. ИСС – 01 – 3 Егурнов К. В. \_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

« »\_\_\_\_\_\_\_\_\_\_\_2003г.

Проверила: Нетикова Л. И. \_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

« »\_\_\_\_\_\_\_\_\_\_\_2003г.

Харьков 2003

**СОДЕРЖАНИЕ**

Реферат…………………………………………………….…………....4

Введение………………………………………………………………...5

## 1. Обзор аналогичных устройств………………….……...……………7

2. Выбранный вариант технического решения…….………………...11

3. Синтез проектируемого устройства………………………………..19

4. Разработка алгоритма программы проектируемого устройства…20

5. Программа на языке ассемблер…………...………………….…….25

6. Разработка и описание принципиальной схемы…………………..30

Вывод……....…………………………………………………….……..33

Список литературы……………………………………………….……34

Приложение А………………………………………………………….35

Приложение Б………………………………………………………….36**Р Е Ф Е Р А Т**

Пояснительная записка: 34 с., 12 рис.

# Цель работы – построение цифрового фильтра с применением современных микропроцессорных систем, освоение методов автоматического проектирования и совершенствование навыков программирования на языке Ассемблер.

В работе использован микропроцессор фирмы MICROCHIP серии PIC16C711, имеющей возможности аналоговой обработки сигналов. ЦАП не входит в состав микроконтроллера и вынесен отдельно, его входы подключены к порту B.

В ходе проектирования были использованы не все ресурсы микропроцессорной системы, поэтому имеется возможность совершенствования устройства. К примеру, можно построить алгоритм цифровой обработки нескольких аналоговых сигналов, используя для каждого сигнала свой канал и программу вычислений.

Программа, написанная на языке Ассемблер, является универсальной для всех микроконтроллеров фирмы MICROCHIP, имеющих возможность аналоговой обработки сигналов. Необходимо лишь сконфигурировать порты и учесть спецификацию РОН и специальных регистров в соответствии с документацией на требуемую модель.

Благодаря современным технологиям разработчик устройств цифровой обработки сигналов может в кратчайшие сроки составлять высокопроизводительные системы с дальнейшим доведением их до производства.

Наличие большого выбора недорогих высокопроизводительных микропроцессорных устройств зарубежного производства на рынке Украины позволяет студенту, радиолюбителю реализовывать свои схемы и алгоритмы в виде технически готового устройства.

**ВВЕДЕНИЕ**

Цифровая обработка сигнала – это арифметическая обработка в реальном масштабе времени последовательности значений амплитуды сигнала, определяемых через равные временные промежутки. Примерами цифровой обработки являются:

* фильтрация сигнала;
* свертка двух сигналов;
* вычисление значений корреляционной функции двух

сигналов;

* усиление, ограничение или трансформация сигнала;
* прямое/обратное преобразование Фурье.

Аналоговая обработка сигнала, традиционно используемая во всех радиотехнических устройствах, является во многих случаях более дешевым способом достижения требуемого результата. Однако тогда, когда требуется высокая точность обработки, миниатюрность устройства, стабильность его характеристик в различных температурных условиях функционирования, цифровая обработка оказывается единственным приемлемым решением.

R2

x(t) x(t) C y(t)

R1

y(t)

t t

входной сигнал выходной сигнал

Рисунок 1.1 - Аналоговая фильтрация сигнала

Пример аналоговой фильтрации приведен на рисунке 1. Используемый в фильтре операционный усилитель позволяет расширить динамический диапазон обрабатываемых сигналов. Форма амплитудно-частотной характеристики фильтра определяется величинами R2, C. Для аналогового фильтра сложно обеспечить высокое значение добротности, характеристики фильтра сильно зависят от температурного режима. Компоненты фильтра вносят дополнительный шум в результирующий сигнал. Аналоговые фильтры трудно перестраивать в широком диапазоне частот.

Важнейшая особенность цифрового фильтра заключается в том, что подбором определенных значений коэффициентов разностного уравнения, описывающего выходной сигнал, можно выполнить преобразование входного сигнала, которое будет нереализуемо при использовании аналогового фильтра. Само устройство, реализующее цифровую фильтрацию, будет иметь компактные размеры и не будет требовать особой технической настройки.

**1. ОБЗОР АНАЛОГИЧНЫХ УСТРОЙСТВ**

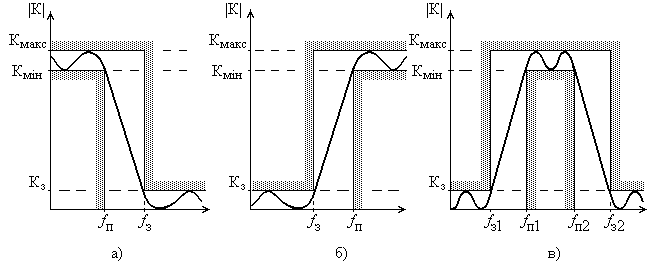
Фильтры – это частотно-избирательные устройства, которые пропускают или задерживают сигналы, лежащие в определенных полосах частот. До 60-х годов для реализации фильтров применялись, в основном, пассивные элементы, т. е. индуктивности, конденсаторы и резисторы. Основной проблемой при реализации таких фильтров оказывается размер катушек индуктивности (на низких частотах они становятся слишком громоздкими). С разработкой в 60-х годах интегральных ОУ появилось новое направление проектирования активных фильтров на базе ОУ. В активных фильтрах применяются резисторы, конденсаторы и усилители (активные компоненты), но в них нет катушек индуктивности. В дальнейшем активные фильтры почти полностью заменили пассивные. Сейчас пассивные фильтры применяются только на высоких частотах (выше 1МГц), за пределами частотного диапазона большинства ОУ широкого применения.

Сейчас во многих случаях аналоговые фильтры заменяются цифровыми. Работа цифровых фильтров обеспечивается, в основном программными средствами, поэтому они оказываются значительно более гибкими в применении по сравнению с аналоговыми. С помощью цифровых фильтров можно реализовать такие передаточные функции, которые очень трудно получить обычными методами. Тем не менее, цифровые фильтры пока не могут заменить аналоговые во всех ситуациях, поэтому сохраняется потребность в наиболее популярных аналоговых фильтрах – активных RC –фильтрах.

Фильтры можно классифицировать по их частотным характеристикам. На рисунке 1.2 изображены характеристики фильтра нижних частот (а), фильтра верхних частот (б), полосового фильтра (в). Основная функция любого фильтра заключается в том, чтобы ослабить сигналы, лежащие в определенных полосах частот, внести в них различные фазовые сдвиги или ввести временную задержку между входным и выходным сигналами.

С помощью активных RC – фильтров нельзя получить идеальные формы частотных характеристик в виде прямоугольников со строго постоянным коэффициентом передачи в полосе пропускания, бесконечным ослаблением в полосе подавления и бесконечной крутизной спада при переходе от полосы пропускания к полосе подавления. Проектирование активного фильтра всегда представляет собой поиск компромисса между идеальной формой характеристики и сложностью её реализации. Это называется «проблемой аппроксимации». Во многих случаях требования к качеству фильтрации позволяют обойтись простейшими фильтрами первого или второго порядка. Набор таких схем приведен ниже. Проектирование фильтра в этом случаи сводится к выбору схемы с наиболее подходящей конфигурацией и последующему расчету значений номиналов элементов для конкретных частот.

Рисунок 1.2 – Типовые характеристики фильтров.



Однако бывают ситуации, когда требование к фильтрации сигнала могут оказаться гораздо более жесткими, и могут потребоваться схемы фильтра с характеристиками более высоких порядков, чем первых или второй. Проектирование фильтра высоких порядков является более сложной задачей.

Анализ состояния и тенденций развития теории и практики цифровой фильтрации показывает, что основными используемыми методами являются частотная селекция сигналов и оптимальная (адаптивная) фильтрация.

Классификация цифровых фильтров (ЦФ) была предложена в докладе Д.А. Губанова и В.Б. Сташенко В ее основу положен функциональный признак (т. е. используемые алгоритмы цифровой фильтрации, а не схемотехнические решения), согласно которому ЦФ подразделяются на 4 группы: фильтры частотной селекции, оптимальные (квазиоптимальные), адаптивные и эвристические.

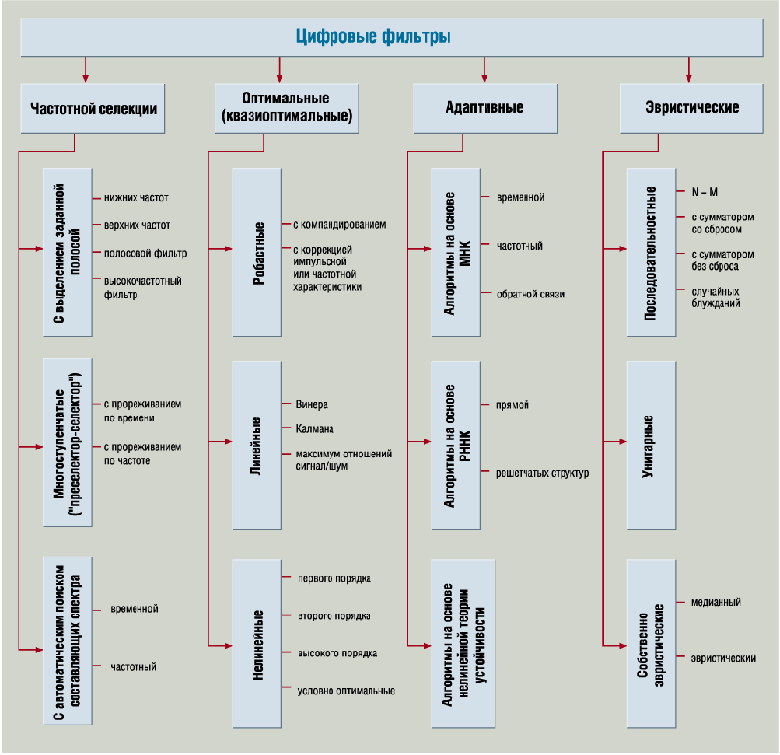


Рисунок 1.3 – Классификация цифровых фильтров

Наиболее изученными и опробованными на практике являются ЦФ частотной селекции. Они почти всегда представляют собой реализованные на новой элементной базе традиционные аналоговые фильтры частотной селекции. Выступления участников конференции показали, что развитие цифровых средств частотной селекции происходит в следующих направлениях:

* создание пакетов прикладных программ для структурного синтеза, анализа качества фильтрации, обеспечения схемотехнической реализации и тестирования устройств;
* совершенствование существующих методов оптимального проектирования многоступенчатых структур с целью их полной формализации и включения в состав ПО;
* разработка новых подходов к проектированию ЦФ частотной селекции с улучшенными качественными показателями.

Новейшие технические реализации как традиционных, так и нетрадиционных задач цифровой обработки сигналов чаще всего используют разные схемотехнические решения. Наибольшее внимание участников секции цифровой фильтрации привлекли алгоритмы многоскоростной обработки сложных сигналов, которые содержат как быстро, так и медленно меняющиеся составляющие. Такой алгоритм должен предусматривать предварительное разделение «быстрых» и «медленных» компонентов с понижением частоты дискретизации «медленных» составляющих и последующей их обработкой. Современные вычислительные средства позволяют решать в режиме реального времени и задачи многомерной фильтрации, существенно более сложные, чем цифровая фильтрация одномерных сигналов, выполняемая с помощью сигнальных процессоров или многопроцессорных систем.

**2. ВЫБОР ВАРИАНТА ТЕХНИЧЕСКОГО РЕШЕНИЯ**

Мир современных технологий наполнен разнообразием микропроцессорных устройств. Десятки крупнейших фирм производителей конкурируют между собой, предлагая каждый день новую более совершенную продукцию. В курсовом проекте используется цифровая обработка сигналов. Для цифровой обработки сигналов используются так называемые сигнальные микропроцессоры. Рассмотрим некоторые микроконтроллеры, выпускаемые современной промышленностью, наиболее подходящие для реализации курсовой работы.

2.1 МИКРОПРОЦЕССОРЫ КОМПАНИИ ANLOG DEVICES.

Микропроцессоры компании Analog Devices образуют два семейства: ADSP21xx и ADSP210xx.

Семейство ADSP21xx – набор однокристальных 16-разрядных микропроцессоров с общей базовой архитектурой, оптимизированной для выполнения алгоритмов цифровой обработки сигналов и других приложений, требующих высокоскоростных вычислений с фиксированной точкой.

Второе семейство микропроцессоров ADSP210xx объединяет 32-х разрядные микропроцессоры, ориентированные на сигнальные алгоритмы, требующие вычислений с плавающей точкой.

В рамках каждого семейства микропроцессоров обеспечивается совместимость снизу вверх по системе команд. Старшие представители семейства обладают большими функциональными возможностями и содержат на кристалле дополнительные функциональные блоки.

Рассмотрим подробнее микропроцессоры первого семейства, так как их возможности наиболее подходят для реализации проекта.

Микропроцессоры семейства ADSP21xx успешно конкурируют с аналогичной продукцией других компаний производителей сигнальных процессоров благодаря сравнимой производительности при более низкой цене, а также развитой системе технических и программных средств разработки прикладных систем. Основными конкурентами данного семейства являются микропроцессоры DSP56xxx (Motorola) и TMS320C1x, TMS320C2xx, TMS320C5x (Texas Instruments).

Высокая производительность процессоров на сигнальных алгоритмах достигается благодаря многофункциональной и гибкой системе команд, аппаратной реализации большинства типичных для данных приложений операций, высокой степени параллелизма процессов в микропроцессоре, сокращения командного такта. Микропроцессоры ADSP21xx имеют модифицированную Гарвардскую архитектуру, в рамках которой предусматривается возможность доступа в память команд, при ее физическом разделении с памятью данных.

Каждый микропроцессор семейства содержит три независимых полнофункциональных устройства: АЛУ, МАС – умножитель с накоплением, устройство барабанного сдвига. Каждое устройство непосредственно оперирует с 16-ти разрядными данными и обеспечивает аппаратную поддержку вычислений с различной точностью.

Микропроцессор содержит генератор адресов команд и два генератора адресов данных, обеспечивающие адресацию к данным и командам, расположенным как во внутренней, так и во внешней памяти. Параллельное функционирование генераторов сокращает длительность выполнения команды, позволяя за один такт выбирать из памяти команду и два операнда.

Таймер/счетчик микропроцессора обеспечивает периодическую генерацию прерываний.

Последовательные порты (SPORTs) обеспечивают последовательный интерфейс с большинством стандартных последовательных устройств, а также с аппаратными средствами сжатия-восстановления данных, использующими A- и μ- законы компандирования.

Порт интерфейса с хост-процессором позволяет без дополнительных интерфейсных схем взаимодействовать с главным микропроцессором системы, в качестве которого может использоваться как процессор данного семейства, так и другой микропроцессор, например Motorola 68000 или Intel 8051.

Микропроцессор ADSP – 21msp5x отличается наличием аналогового интерфейса, позволяющего совмещать аналоговую и цифровую обработку. В состав интерфейсных средств входят АЦП, ЦАП, цифровой и аналоговый фильтры, параллельный интерфейс к процессорному ядру.

Система команд микропроцессоров семейства оптимизирована для алгоритмов цифровой обработки сигналов. По системе команд все микропроцессоры совместимы снизу вверх. Совершенствование данного семейства идет в направлении повышения тактовой частоты, снижения энергопотребления и расширения коммуникационных возможностей процессора.

2.2 МИКРОПРОЦЕССОРЫ КОМПАНИИ MOTOROLA

Сигнальные микропроцессоры компании Motorola. Подразделяются на семейства 16- и 24-разрядных микропроцессоров с фиксированной точкой – DSP – 560xx, - 561xx, - 563xx, -566xx, 568xx и микропроцессоры с плавающей точкой – DSP – 960xx.

Рассмотрим 24-х разрядные микропроцессоры с фиксированной точкой семейства DSP 560xx. Эти микропроцессоры являются первыми представителями сигнальных микропроцессоров компании Motorola. Архитектура микропроцессоров ориентирована на максимизацию пропускной способности в приложениях DSP с интенсивным обменом данными. Это обеспечивается благодаря расширяемой архитектуре со сложной встроенной периферией и универсальной подсистеме ввода/вывода. Данные свойства, а также низкое энергопотребление минимизируют сложность, стоимость и сроки разработки прикладных систем на базе микропроцессоров DSP56000/DSP56001.

Микропроцессоры работают на частоте 33МГц и обеспечивают производительность около 16 MIPS, что позволяет выполнять быстрое преобразование Фурье по 1024 отсчетам за 3,23мс.

Дальнейшее развитие семейства микропроцессоров осуществляется в рамках концепции процессорного ядра, общего для всех представителей семейства, в состав которого входят 24-разрядные микропроцессоры с фиксированной точкой.

Процессоры данного семейства характеризуются высокой пропускной способностью, расширенной разрядностью, обеспечивающей высокую точность вычисления и широким динамическим диапазоном обрабатываемых данных, поддержкой энергосберегающего режима работы. Представители семейства отличаются друг от друга конфигурациями памяти и периферийными устройствами.

2.3 МИКРОПРОЦЕССОРЫ КОМПАНИИ TEXAS INSTRUMENTS

Сигнальные микропроцессоры компании Texas Instruments разделяются на два класса: это процессоры для обработки чисел с фиксированной точкой и процессоры для обработки чисел с плавающей точкой. Первый класс представлен тремя семействами процессоров, базовыми моделями которых являются соответственно TMS320.10, .20, .50. Второй класс включает процессоры TMS320.30, . 40, TMS320С80, которые поддерживают операции с плавающей точкой и представляют собой мультипроцессорную систему, выполненную в одном кристалле, а семейство TMS320C6x включает процессоры как с фиксированной, так и с плавающей точкой.

Процессоры старших поколений одного семейства наследует основные архитектурные особенности и совместимы “снизу вверх” по системе команд (чего нельзя сказать о процессорах, входящих в разные семейства).

Перечислим некоторые микропроцессоры, оптимально подходящие для нашей системы.

1.Микропроцессоры семейства TMS320C1x

Первый процесор семейства – TMS320C10 был выпущен в 1982г. и благодаря ряду удачных технических решений получил широкую распространенность.

В основу микропроцессоров данного семейства положена модифицированная Гарвардская архитектура, отличием которой от традиционной Гарвардской архитектуры является возможность обмена данными между памятью программ и памятью данных, что повышает гибкость устройства.

TMS320C10 является 16-разрядным процессором. Его адресное пространство составляет 4K 16-разрядных слов памяти данных. Длительность такта процессора составляет 160-200 нс.

Арифметические функции в процессоре реализованы аппаратно. Он имеет аппаратные умножители, устройство сдвига, аппаратную поддержку автоинкремента/декремента адресных регистров данных.

С внешними устройствами процессор взаимодействует через 8 16-разрядных портов ввода/вывода. Предусмотрена возможность внешнего прерывания.

Остальные микропроцессоры данного семейства имеют аналогичную архитектуру и отличаются длительностью командного такта, конфигурацией памяти, наличием (или отсутствием) дополнительных периферийных устройств.

2.Микропроцессоры семейства TMS320C2x.

Микропроцессоры семейства TMS320C2x имеют анлогичную архитектуру, но обладают повышенной производительностью и более широкими функциональными возможностями. Все процессоры семейства поколения могут использовать по 64K слов памяти программ и данных, имеют 16 16-разрядных портов ввода/вывода и последовательный порт.

Процессоры семейства TMS320C2x имеют возможность использования внешнего контроллера ПДП. Умножитель микропроцессоров, помимо операций умножения, позволяет выполнять за один такт возведение в квадрат. В процессоры включена аппаратная поддержка кратного выполнения команды, реализован режим двоичной инверсно-косвенной адресации, предназначенный для эффективной реализации быстрого преобразования Фурье.

2.4 ОБОСНОВАНИЕ ВЫБРАННОГО ВАРИАНТА

После длительного обзора микропроцесорных устройств, возникла задача выбора наиболее подходящего микропроцессора. Решение данной задачи было найдено при рассмотрении микропроцессоров компании Microchip. Для реализации проекта было принято решение использовать микропроцессор компании Microchip PIC16C711. Приведем преимущества использования данного микропроцессора в курсовой работе, а также общие его характеристики.

PIC16C711 это дешевый, высокопроизводительный, изготовленный по КМОП технологии 8-битный микроконтроллер. В основу процессора положена RISC архитектура, он имеет улучшенные характеристики ядра, внутренние и внешние прерывания. Устройство имеет два двунаправленных порта ввода/вывода, один из которых может быть использован как вход для аналого-цифрового преобразования, другой - как выход для ЦАП. Тактовая частота составляет 20MHz.

На рисунке 2.1 представлена структурная схема PIC16C711. Перечислим основные элементы схемы: АЛУ (ALU), где выполняются простейшие арифметические операции, а также логические операции; аккумулятор (W), где хранятся данные результатов вычислений; регистр состояния, в котором хранятся флаги результатов операций, а также биты выбора банка памяти; регистр косвенной адресации (FSR), через который непрямым образом можно обращаться к ОЗУ; память (RAM), организованная 8-битными специальными регистрами и регистрами общего назначения, рабочая область пользователя составляет 68x8; 13-битный счетчик команд (Program Counter), младшие 8 бит приходят из регистра PCL, старшие 5 из регистра PCLATH; ППЗУ (EPROM), составляет 1K 14- битных ячеек памяти; стек 8-ми уровневый, длина слова 13-бит, во время прерывания содержимое счетчика команд сохраняется в стек автоматически; два двунаправленных 8-битных порта А и В; 4-х канальный 8-битный АЦП (A/D); счетчик Timer0.

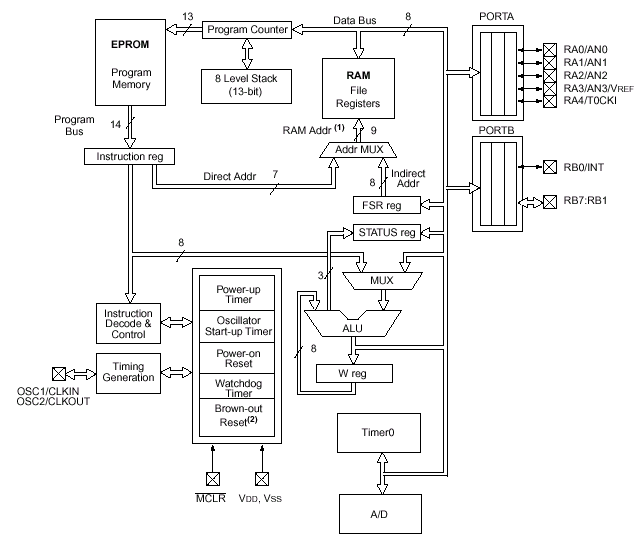


Рис. 2.1 – Структурная схема PIC16C711

К сожалению устройство не имеет внутреннего ЦАП. ЦАП будет внешним, его вход будет подсоединен к порту В. Применение внешнего ЦАП не ухудшит работу цифрового фильтра и не усложнит его структуру, так как ЦАП не нужно синхронизировать с временем выполнения программы, в отличие от АЦП.

Выбранный вариант технического решения полностью удовлетворяет техническому заданию. Микроконтроллер + внешний ЦАП позволяют создать гибкую, высокопроизводительную систему, удовлетворяющую требованиям микроминиатюрности, быстродействия, качества и простоты исполнения. Сравнивая данный микроконтроллер с большинством современных микроконтроллеров, можно сказать, что он обладает сравнительно низкой стоимостью и доступностью на рынке Украины. В современных системах обработки сигналов требуется высокая разрядность шины данных и АЦП для увеличения точности вычислений, также для этих целей необходимо, чтобы АЛУ могло вычислять операции с плавающей точкой. Выбранный микроконтроллер не имеет данных свойств, но он благодаря своему быстродействию и гибкости может вполне справиться с целью, поставленной в техническом задании на курсовой проект.

Архитектура процессора построена таким образом, что он способен выполнять команду за один цикл, кроме операций условных и безусловных переходов. Это означает, что время выполнения программы будет небольшим, и частота дискретизации АЦП будет удовлетворять условию обработки сигналов качественной телефонии. Ассемблер очень простой: пользователю необходимо выучить 35 команд. Последнее достоинство позволяет снизить время на подготовку к выполнению проекта.

**3 СИНТЕЗ ПРОЕКТИРУЕМОГО УСТРОЙСТВА**

Структурная схема типа реализации ЦФ приведена на рис.3.1. Параметры звеньев ЦФ следующие :

параметры звена : ;



параметры звена : ;



параметры звена : .



*Um*



Рис. 3.1 – Структурная схема цифрового фильтра

Исходя из этой структуры, а также коэффициентов, для каждого звена составим разностные уравнения:

1. первое звено:

;



1. второе звено:

;



1. третье звено:

.



Результирующее разностное уравнение для цифрового фильтра будет иметь вид:

.



**4 РАЗРАБОТКА АЛГОРИТМА ПРОГРАММЫ**

**ПРОЕКТИРУЕМОГО УСТРОЙСТВА**

Прежде чем приступить к программированию устройства необходимо основательно изучить его внутреннюю структуру и возможности ресурсов. Простейшая блок-схема проектируемого устройства приведена на рисунке 4.1 В ней учтены конфигурация портов и АЦП, вычисление разностного уравнения фильтра, результат вычислений выдается на порт В.

Начало

Порт А –

вход

Порт В –

выход

Запуск

АЦП

Вычисление

разностного

уравнения

Выдача

Результата

На порт В

Рисунок 4.1 – блок-схема работы цифрового фильтраОпираясь на данную блок схему, начнем более тщательное

рассмотрение вопроса. Конфигурация портов и АЦП задается исходя из документации предлагаемой производителем. Данная документация содержит примеры программ для настройки портов и АЦП. Оговорим некоторые аспекты этих программ применительно к нашему курсовому проекту. Контакты порта А зададим входами установкой битов в регистре TRISA. Данное условие необходимо для осуществления аналого-цифровой обработки сигнала. Контакты порта В зададим как выходные сбросом битов в регистре TRISB. Для АЦП выберем канал 0, то есть вывод микропроцессора RA0. В PIC16C711 на одно преобразование АЦП необходим минимальный интервал времени 10Тad, где Тad =1,6 мкс выбирается программно. Также одной из важных частей программы запуска АЦП является организация разрешения прерываний.

После конфигурации портов и запуска АЦП необходимо приступить к выполнению программы вычисления разностного уравнения, описывающего фильтр.

Просчитаем количество переменных, входящих в уравнения. Это количество будет определять необходимое число байт памяти. В нашем случае оно составляет 17. Нетрудно заметить, что коэффициенты разностных уравнений подобраны таким образом, что умножение переменной на них будет делением на 2,4,8…, а эта операция выполняется путем сдвига регистра вправо. Для хранения промежуточных данных сдвиговых операций и операций сложения нам понадобятся дополнительные ячейки памяти в количестве 17. При начальном запуске программы или при сбросе микропроцессорной системы все ячейки памяти необходимо сбросить в нуль.

На обработку данных АЦП требуется, как указывалось ранее, время. Чтобы повысить эффективность и качество системы, необходимо во время просчета АЦП, вместо простоя и ожидания системой результата просчета, произвести вычисления без участия текущего значения отсчета. Вычисления с участием текущего значения отсчета должны производиться через время, равное 10Tad.

Результат вычисления АЦП будет обрабатывать программа прерывания.

Исходя из всех вышеприведенных соображений, составим окончательный алгоритм программы:

Начало 1

Установка

порта А на COUNT-1

ввод

Установка да

порта В на Z=0

вывод

нет

6

0x17→W Сброс ADCON1

W→COUNT 0xC1→W

W→ADCON0

0x0C→W

Разрешить

прерывание

W→FSR АЦП

Разрешить

глобальное

Сброс INDF прерывание

FSR + 1 Пуск АЦП

1 2

2 3

Вычисление

А11UM1с сох- Вычисление

ранением зна- A32YN2YK2=

чения =A32(YN2+YK2)

с сохранением

значения

Вычисление

B11YN1 с сох-

ранением зна- Вычисление

чения B31YL1

Вычисление

В12YN2

ВычислениеYLL

= A31YN1YK1+

Вычисление + A32YN2YK2-

YNN=А11UM1- - B31YL1 с сох-

- B11YN1- ранением зна-

- В12YN2 с сох- чения

ранением зна-

чения

Вычисление YK1 → W

A21UM1 с сох-

ранением зна-

чения

Вычисление W → YK2

B21YK1

Вычисление YK → W

YK=А21UM1-

- B21YK1 с сох-

ранением зна-

чения

W → YK1

Вычисление

A31YN1YK1 =

=A31(YN1+YK1) Вычисление

с сохранением A10UM

значения

3 4 4 6 5

Вычисление

YN =A10UM+ UM → W

+YNN

с сохранением

значения

W → UM1

Вычисление

YL=A30(YN+

+YK)+YLL

с сохранением

значения

ОСТАНОВ

Передача знач-

чения YL на

порт B

YN1 → W

W → YN2

YN → W

W → YN1

YL → W

5

Рисунок 4.2 - Алгоритм программы проектируемого устройства

**5 РАЗРАБОТКА ПРОГРАММЫ НА ЯЗЫКЕ АССЕМБЛЕР**

Прежде чем приступить к написанию программы необходимо учесть спецификацию ПЗУ микроконтроллера. Организация ПЗУ изображена на рисунке 5.1

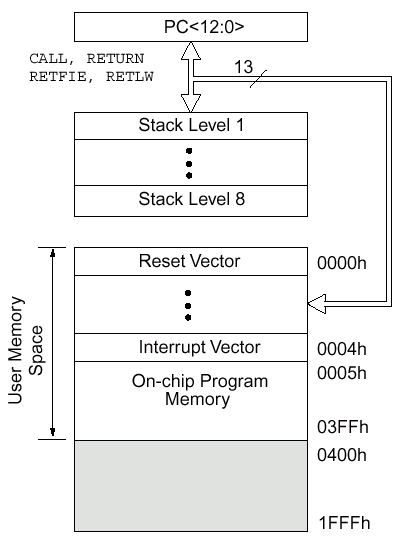


Рисунок 5.1 – Организация ПЗУ PIC16C71

Из рисунка видно, что начало программы пользователя может быть расположено ниже адреса 0004h. Адрес 0000h – вектор сброса, 0004h – вектор прерывания. По этим адресам перед началом программирования, как правило, прописывают безусловный переход на программу пользователя и безусловный переход на программу обработки прерывания соответственно. Так как в случае нашей системы устройство, осуществляющее запрос прерывания, одно, то по адресу 0004h можно прописывать начало программы обработки прерывания.

Также при программировании обязательно нужно учесть спецификацию банков ОЗУ. Структура ОЗУ изображена на

рисунке 5. При обращении к регистрам ОЗУ каждый раз необходимо программно переключаться между банками ОЗУ, в зависимости от того в каком банке находится регистр. Ассемблер, поставляемый производителем, позволяет вначале всем регистрам присвоить вместо адреса определенное имя. Это удобно для составления программы.

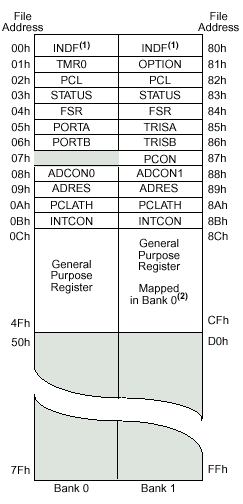


Рисунок 5.2 – Организация ОЗУ PIC16C711

Сначала напишем программу обработки прерывания. Занесем программу в таблицу - 1 с адресами ПЗУ и комментариями.

Таблица 5.1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Адрес | Метка | Мнемонический код | Комментарии | Циклы |
|  |  | ORG 0x004 | Директива |  |
| 0004h |  | MOVWF W\_TEMP | Сохранение содержимого W в W\_TEMP | 1 |
| 0005h |  | MOVF STATUS, 0 | Пересылка содержимого STATUS в W | 1 |
| 0006h |  | MOVWF STATUS\_TEMP | Сохранение W в STATUS \_TEMP | 1 |
| Продолжение таблицы 5.1 | | | | |
| 0007h |  | BCF STATUS, RP0 | Выбор банка 0 | 1 |
| 0008h |  | MOVF ADRES, 0 | Пересылка результата АЦП в W | 1 |
| 0009h |  | MOVWF UM | Пересылка W в UM |  |
| 000Ah |  | MOVF STATUS\_TEMP, 0 | Пересылка STATUS\_TEMP в W | 1 |
| 000Bh |  | MOVWF STATUS | Пересылка W в STATUS | 1 |
| 000Ch |  | MOVF W\_TEMP, 0 | Пересылка W\_TEMP в W | 1 |
| 000Dh |  | RETFIE | Возврат прерывания | 2 |

Программу, реализующую математическую модель фильтра начнем с адреса 000Eh и разместим в таблицу 5, 2.

Таблица 5.2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Адрес | Метка | Мнемонический код | Комментарии | Циклы |
|  |  | ORG 0x00E | Директива |  |
| 000Eh |  | MOVLW 0xFF | Пересылка 0xFF в аккумулятор | 1 |
| 000Fh |  | BSF STATUS, RP0 | Выбор банка 1 | 1 |
| 0010h |  | MOVWF TRISA | Порт А – Вход | 1 |
| 0011h |  | CLRF TRISB | Порт В – Выход | 1 |
| 0012h |  | MOVLW 0x18 | Организация счетчика СOUNT | 1 |
| 0013h |  | BCF STATUS, RP0 | Выбор банка 0 | 1 |
| 0014h |  | MOVWF COUNT | Организация счетчика | 1 |
| 0015h |  | MOVLW 0x0C | Адрес начальной ячейки в ОЗУ | 1 |
| 0016h |  | MOVWF FSR | Пересылка адреса начальной ячейки в FSR | 1 |
| 0017h | M1: | CLRF INDF | Обнуление текущей ячейки памяти | 1 |
| 0018h |  | INCF FSR, 1 | Адрес + 1 | 1 |
| 0019h |  | DECF COUNT | Счетчик – 1 | 1 |
| 001Ah |  | BTFSS STATUS, Z | Условный переход по 1 значению флага нулевого результата | 1(2) |
| 001Bh |  | GOTO M1 |  | 2 |
| 001Ch | M2: | BSF STATUS,RP0 | Выбор банка 1 | 1 |
| 001Dh |  | CLRF ADCON1 | RA0-RA3 Входы АЦП | 1 |
| 001Eh |  | MOVLW 0xC1 |  | 1 |
| 001Fh |  | BCF STATUS,RP0 | Выбор банка 0 | 1 |
| 0020h |  | MOVWF ADCON0 | АЦП включен, работает канал0 (RA0), частота 32Tosc. | 1 |
| 0021h |  | BSF INTCON ADIE | Разрешить прерывание АЦП | 1 |
| 0022h |  | BSF INTCON GIE | Разрешить глобальное прерывание | 1 |
| 0023h |  | BSF ADCON0, GO | Запустить АЦП | 1 |
| 0024h |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 0025h |  | RRF UM1, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 0026h |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 0027h |  | RRF W, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 0028h |  | MOVWF A11UM1 | W→ A11UM1 | 1 |
| 0029h |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 002Ah |  | RRF YN1, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 002Bh |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 002Ch |  | RRF W, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 002Dh |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 002Eh |  | RRF W, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| Продолжение таблицы 5.2 | | | | |
| 002Fh |  | BCF STATUS, C |  | 1 |
| 0030h |  | RRF W, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 0031h |  | MOVWF B11YN1 | W→ B11YN1 | 1 |
| 0032h |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 0033h |  | RRF YN1, 0 | Циклический сдвиг вправо через триггер переноса |  |
| 0034h |  | ADDWF W, 0 | W=W+W | 1 |
| 0035h |  | ADDWF W, 0 | W=W+W | 1 |
| 0036h |  | ADDWF B11YN1 | W=W+ B11YN1 | 1 |
| 0037h |  | SUBWF A11UM1, 0 | W= A11UM1- W | 1 |
| 0038h |  | MOVWF YNN | W→ YNN | 1 |
| 0039h |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 003Ah |  | RRF UM1, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 003Bh |  | MOVWF A21UM1 | W→ A21UM1 | 1 |
| 003Ch |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 003Dh |  | RRF YK1, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 003Eh |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 003Fh |  | RRF W, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 0040h |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 0041h |  | RRF W, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 0042h |  | SUBWF A21UM1, 0 | W= A21UM1- W | 1 |
| 0043h |  | MOVWF YK | W→ YK | 1 |
| 0044h |  | MOVF YN1, 0 | YN1→W | 1 |
| 0045h |  | ADDWF YK1 | W=W+ YK1 | 1 |
| 0046h |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 0047h |  | RRF W, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 0048h |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 0049h |  | RRF W, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 004Ah |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 004Bh |  | RRF W, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 004Ch |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 004Dh |  | RRF W, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 004Eh |  | MOVWF A31YN1YK1 | W→ A31YN1YK1 | 1 |
| 004Fh |  | MOVF YN2, 0 | YN2→W | 1 |
| 0050h |  | ADDWF YK2 | W=W+ YK2 | 1 |
| 0051h |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 0052h |  | RRF W, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 0053h |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 0054h |  | RRF W, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 0055h |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 0056h |  | RRF W, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 0057h |  | MOVWF A32YN2YK2 | W→ A32YN2YK2 | 1 |
| 0058h |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 0059h |  | RRF YL1, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| Продолжение таблицы 5.2 | | | | |
| 005Ah |  | ADDWF W, 0 | W=W+W | 1 |
| 005Bh |  | ADDWF W, 0 | W=W+W | 1 |
| 005Ch |  | SUBWF A32YN2YK2, 0 | W= A32YN2YK2- W | 1 |
| 005Dh |  | ADDWF A31YN1YK1 | W=W+ A31YN1YK1 | 1 |
| 005Eh |  | MOVWF YLL | W→ YLL | 1 |
| 005Fh |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 0060h |  | NOP | Ничего не выполнять | 1 |
| 0061h |  | NOP |  | 1 |
| 0062h |  | NOP |  | 1 |
| 0063h |  | NOP |  | 1 |
| 0064h |  | NOP |  | 1 |
| 0065h |  | NOP |  | 1 |
| 0066h |  | NOP |  | 1 |
| 0067h |  | NOP |  | 1 |
| 0068h |  | NOP |  | 1 |
| 0069h |  | NOP |  | 1 |
| 006Ah |  | NOP |  | 1 |
| 006Bh |  | NOP |  | 1 |
| 006Ch |  | NOP |  | 1 |
| 006Dh |  | NOP |  | 1 |
| 006Eh |  | NOP |  | 1 |
| 006Fh |  | NOP |  | 1 |
| 0070h |  | NOP |  | 1 |
| 0071h |  | NOP |  | 1 |
| 0072h |  | NOP |  | 1 |
| 0073h |  | NOP |  | 1 |
| 0074h |  | MOVF YK1, 0 | YK1→W | 1 |
| 0075h |  | MOVWF YK2 | W→ YK2 | 1 |
| 0076h |  | MOVF YK, 0 | YK →W | 1 |
| 0077h |  | MOVWF YK1 | W→ YK1 | 1 |
| 0078h |  | RRF UM, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 0079h |  | ADDWF YNN | W=W+ YNN | 1 |
| 007Ah |  | MOVWF YN | W→ YN | 1 |
| 007Bh |  | ADDWF YK | W=W+ YK | 1 |
| 007Ch |  | BCF STATUS, C | Очистка триггера переноса | 1 |
| 007Dh |  | RRF W, 0 | Циклический сдвиг вправо через триггер переноса | 1 |
| 007Eh |  | ADDWF W, 0 | W=W+W | 1 |
| 007Fh |  | ADDWF YLL | W=W+ YLL | 1 |
| 0080h |  | MOVWF YL | W→ YL | 1 |
| 0081h |  | MOVWF PORTB | W→ PORTB | 1 |
| 0082h |  | MOVF YN1, 0 | YN1→W | 1 |
| 0083h |  | MOVWF YN2 | W→ YN2 | 1 |
| 0084h |  | MOVF YN, 0 | YN →W | 1 |
| 0085h |  | MOVWF YN1 | W→ YN1 | 1 |
| 0086h |  | MOVF YL, 0 | YL →W | 1 |
| 0087h |  | MOVWF YL1 | W→ YL1 | 1 |
| 0088h |  | MOVF UM, 0 | UM →W | 1 |
| 0089h |  | MOVWF UM1 | W→ UM1 | 1 |
| 008Ah |  | GOTO M2 | Безусловный переход на метку | 1 |
|  | END | |  |  |
| Всего вместе с программой обработки прерывания | | | | 283 |

Время выполнения программы 56 мкс.

Операции NOP предназначены для синхронизации работы АЦП с программой. Также для этой цели было добавлено 4 команды перед операциями над текущим значением отсчета. В составленном алгоритме данные коррекции небыли предусмотрены. Время выполнения программы, реализующей цифровой фильтр, равно 49 мкс. Интервал дискретизации сигнала равен 24,6 мкс, соответственно частота дискретизации равна 41000Гц. В программе используются два перехода: один для обнуления ОЗУ, другой – для возврата программы на место запуска АЦП. При использовании более современных PIC процессоров возможно исключение операций NOP.

**6 РАЗРАБОТКА И ОПИСАНИЕ ПРИНЦИПИАЛЬНОЙ СХЕМЫ**

Для технической реализации ЦФ необходимы следующие элементы:

* процессор;
* ЦАП;
* Блок питания от сети 220 В, 50 Гц.

**Описание процессора:**

Основным блоком ЦФ является процессор PIC16C711 (рис.6.1). Приведём краткую характеристику интересующих для данной задачи выводов процессора.

*RA4/RTCC* - Вход через триггер Шмита. Ножка порта ввода/вывода с открытым стоком или вход частоты для таймера/счетчика RTCC. *RA0/AIN0* -Двунаправленная линия ввода/вывода. Аналоговый вход канала 0. Как цифровой вход имеет уровни ТТЛ.

*RA1/AIN1* - Двунаправленная линия ввода/вывода. Аналоговый вход канала 1. Как цифровой вход имеет уровни ТТЛ.

*RA2/AIN2* - Двунаправленная линия ввода/вывода.Аналоговый вход канала 2. Как цифровой вход имеет уровни ТТЛ.

*RA3/AIN3/Vref* - Двунаправленная линия ввода/вывода. Аналоговый вход канала 3. Как цифровой вход имеет уровни ТТЛ.

*RB0/INT* - Двунаправленная линия порта вывода или внешний вход прерывания. Уровни ТТЛ.

*RB1 - RB5* - Двунаправленные линии ввода/ вывода. Уровни ТТЛ.

*RB6* - Двунаправленные линии ввода/ вывода. Уровни ТТЛ.

*RB7* - Двунаправленные линии ввода/ вывода. Уровни ТТЛ.

*/MCLR/Vpp* - Низкий уровень на этом входе генерирует сигнал сброса для контроллера. Активный низкий. Вход через триггер Шмитта.

*OSC1* - Для подключения кварца, RC или вход внешней тактовой частоты.

*OSC2 /CLKOUT* - Генератор, выход тактовой частоты в режиме RC генератора, в остальных случаях - для подключения кварца.

*Vdd* - Напряжение питания.

*Vss* – Общий (земля).

RA2/AIN2 1 18 RA1/AIN1

RA3/AIN3/Vref RA0/AIN0

RA4/RTCC OSC1/CLKIN

/MCLR/Vpp OSC2/CLKOUT

Vss PIC16C711 Vdd

RB0/INT RB7

RB1 RB6

RB2 RB5

RB3 9 10 RB4

Рисунок 6.1 – Выводы процессора PIC16C711

**Описание ЦАП:**

В качестве ЦАП используется микросхема фирмы Analog Devices AD7302. Данный ЦАП имеет следующие основные характеристики:

AD7302 двойной, 8-битный ЦАП, который работает от однополярного источника питания с напряжением +2.7 - +5.5 В. AD7302 имеет интерфейс, совместимый параллельным выходом микропроцессоров и цифровых сигнальных процессоров (DSP). Интерфейс обладает высокоскоростными регистрами и двойной буферизированной логикой. Данные загружаются при повышенном уровне на CS или WR. С помощью пина ~A/B выбирают какой из 2 встроенных ЦАП будет использоваться.

Приведём краткую характеристику интересующих для данной задачи выводов ЦАП.

*D7-D0* - Параллельные информационные входы. 8-битные данные загружаються во входной регистр AD7302 под управлением CS и WR.

*CS* - Выбор кристалла. Активный низкоуровневый логический ввод.

*WR* - Ввод записи. Активный низкоуровневый логический ввод. Используеться вместе CS и A/B, чтобы писать данные в регистр выбранного ЦАП.

*A/B* – Выбор ЦАП.

*PD* – Активный низкоуровневый ввод, используемый для перехода в режим потребления малой мощности.

*LDAC* – Логический ввод загрузки ЦАП. Когда на этот ввод подаеться низкий уровень, оба ЦАП синхронно обновляются со своими регистрами.

*CLR* – Асинхронный вход сброса. Когда на него подаеться низкий уровень, все регистры ЦАП обнуляються и на выход напряжение не поступает.

*Vdd* – Напряжение питания.

*REFIN* – Внешний ввод ссылки. Может использоваться как ссылка для обоих ЦАП. Диапазон этого ввода 1 В – Vdd/2.

*AGND* – Земля для аналоговой части микросхемы.

*VoutB* – Аналоговый вывод ЦАП B.

*VoutA* – Аналоговый вывод ЦАП А.

*DGND* – Земля для цифровой части схемы.

DB7 1 20 DGND

DB6 VoutA

DB5 VoutB

DB4 AGND

DB3 AD7302 REFIN

DB2 Vdd

DB1 CLR

DB0 LDAK

CS PD

WR 10 11 A/B

Рисунок 6.2 – Выводы ЦАП AD7302

**Блок питания от сети 220 В, 50 Гц :**

На рисунке 6.3 приведена простая схема блока питания на 3 В (ток в нагрузкеке 200 мА) с автоматической электронной защитой от перегрузки (Iз = 250 мА). Уровень пульсации выходного напряжения не превышает 1 мВ.В схеме в качестве источника опорного напряжения используется светодиод HL1. Трансформатор Т1 можно приобрести из унифицированной серии ТН любой, но лучше использовать самые малогабаритные ТИ1-127/220-50 или ТН2-127/220-50. Подойдут также и многие другие типы трансформаторов со вторичной обмоткой на 5...6 В. Конденсаторы С1...СЗ типа К50-35. Схема использует интегральный стабилизатор DA1, для нормальной работы микросхемы необходимо, чтобы входное напряжение превышало выходное не менее чем на 3,5 В. Это снижает КПД стабилизатора за счет тепловыделения на микросхеме — при низком выходном напряжении мощность, теряемая в блоке питания, будет превышать отдаваемую в нагрузку. Необходимое выходное напряжение устанавливается подстроечным резистором R2. Микросхема устанавливается на радиатор.

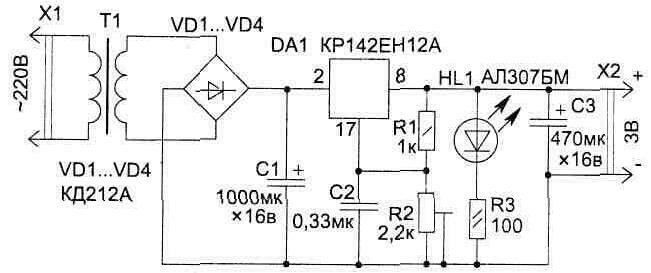


Рисунок 6.2 – Схема генератора

В**ЫВОД**

Спроектированная система является достаточно простой. Схема электрическая принципиальная составлялась на ПК с использованием пакета ACCEL. С помощью утилит, входящих в данный пакет, можно провести трассировку печатной платы по готовой схеме электрической принципиальной и довести проект до готового изделия.

По результатам курсового проекта можно оценить погрешности связанные с оцифровкой сигнала. Основную погрешность будет вносить ограниченная восемью разрядами шина АЦП, чем больше уровней квантования - тем лучше. Также, ввиду отсутствия операций с плавающей точкой, деление на два нечетных чисел будет давать неточный результат.

Для оцифровки аналоговых сигналов должно выполняться условие . На практике и выше для улучшения качества передаваемого сигнала. Спектр речевого сигнала качественной телефонии составляет 0,3 – 3,4 кГц в соответствии с установленным международным стандартом. Цифровой фильтр, спроектированный в ходе курсовой работы, удовлетворяет требованиям обработки не только сигналов качественной речи, но и сигналов с верхней частотой спектра 15..20кГц (музыка, каналы ШК и т.д.).



СПИСОК ЛИТЕРАТУРЫ

1. Калабеков Б. А. Микропроцессоры и их применение в системах

передачи и обработки сигналов: Учеб. Пособие для вузов. – М.:

Радио и связь, 1988. – 368с.: ил.

1. Корнеев В. В., Киселев А. В. Современные микропроцессоры. – М.:НОЛИДЖ, 1998. – 240 с.; ил.
2. MICROCHIP PIC16/17 MICROCONTROLLER DATA BOOK.
3. Документация на CD, поставляемая компанией MICROCHIP.