# Пояснительная записка

к курсовой работе по дисциплине "Цифровые устройства и микропроцессоры"

Студент Сумнительнов С. Н.

Группа 912 специальности 2007

2003

**Введение**

В наши дни, развитие цифровых устройств происходит гигантскими шагами. Очевидно и преимущество применения цифровой обработки сигнала наряду с аналоговым: улучшается помехозащищенность канала связи, бесконечные возможности кодирования информации. Применение микропроцессоров в радиотехнических системах существенно улучшает их массогабаритные, технические и экономические показатели, открывает широкие возможности реализации сложных алгоритмов цифровой обработки сигналов.

Микропроцессоры находят применение при решении широкого круга радиотехнических задач, таких как построение радиотехнических измерителей координат, сглаживающих и экстраполирующих фильтров, устройств вторичной обработки сигналов, специализированных вычислительных устройств бортовых навигационных комплексов, устройств кодирования и декодирования сигналов, весовой обработки пачечных сигналов в радиолокации, различного рода измерительных устройств и т.п. К таким устройствам относятся и цифровые фильтры, для которых стало возможным построение разнообразных частотных характеристик путём их аналитической задачи. При этом реализуемы и фильтры традиционных типов: нижних частот, верхних частот , полосовые и режекторные.

Цифровой фильтр относится к особому типу фильтров и призван убирать из сигнала импульсные проявления (пики и щелчки), сглаживая их. Импульс имеет широкий (в идеале бесконечный) частотный спектр, однако острота его формы определяется именно высокочастотными составляющими.

Фильтр должен быть выполнен на основе МП комплекта К1821 при использовании ЦАП 1108ПА1. МП комплект К1821 состоит из микросхем: К1821ВМ85 - микропроцессор, К1821РФ55 – ПЗУ (емкость – 2 Кб; два 8-разрядных порта ввода-вывода), К1821РУ55 – ОЗУ (емкость –256 байт; два 8-разрядных и один 6-разрядный порты ввода-вывода, встроенный счетчик-таймер).

Входной сигнал цифровой, преобразуется в аналоговый, ЦАП на микросхеме 1108ПА1А.

После прихода сигнала с периферийного устройства (ПУ) на порт ввода в дополнительном цифровом коде на ПУ выдаётся сигнал квитирования. Частота дискретизации FД = 1.2 кГц, разрядность входного сигнала 8. Обработка должна происходить в реальном масштабе времени.

Проектируемое устройство, его базовая конфигурация должны содержать минимальные аппаратные и программные средства, достаточные для выполнения поставленной выше задачи обработки.

**Анализ и формализация задачи**

Согласно техническому заданию разностное уравнение имеет следующий вид:



Тогда обобщённая структура, описывающая работу фильтра выглядит следующим образом:



где - входные отчёты , - выходные отчёты, - задержка на , и - коэффициенты.



=1.0 =-0.1 = 1.0 =0.64



=1.4 = 0.5 = 0.5



Коэффициенты нужны, для того чтобы не было переполнения в восьми разрядах двоичного числа, с которым оперирует процессор. Расчёт производится следующим образом:



Как видно, фильтр должен держать в памяти семь отсчётов одновременно и оперировать с ними, производя арифметические операции.

Произведём внесение коэффициентов в скобки и получим следующие выражения:



Из формул видно что, для формирования выходного отчёта необходимо использовать текущее значение входного отчёта и его предыдущие значения. Для хранения входных, выходных, предыдущих значений отсчетов и промежуточных результатов вычисления необходимо выделить область памяти в ОЗУ.

После программы инициализации, т.е. после настройки работы, таймер начинает формировать импульсы следующие с частотой дискретизации, которые поступают на вывод STB A (PC2) порта С работающего в информационном режиме. В этом режиме через выводы PC0, PC1, PC2 происходит обмен управляющими сигналами для порта А .

По переднему фронту импульса на линии STB A формируется сигнал BF (PC1) "буфер полон" который используется в качестве сигнала квитирования. По заднему фронту импульса STB A формируется сигнал INTR (PC0) который является сигналом прерывания.

С приходом сигнала запроса прерывания RST 7,5 процессор переходит в режим обслуживания прерываний. Соответственно программе обслуживания прерываний 7,5 производится считывание данных из порта ввода А.Затем процессор начинает обрабатывать данные в соответствии с заданным алгоритмом. По окончанию вычислений процессор выдаёт данные в порт В. И процесс повторяется вновь.

Входные отсчеты поступают на шину данных микропроцессора в виде 8-разрядного параллельного дополнительного кода с частотой Fд=1.2 кГц, которая формируется **таймером**. На выходе устройства стоит ЦАП, преобразующий код в аналоговый сигнал-ток.

Микросхема умножающего ЦАП К572ПА1Б предназначена для преобразования 12-разрядного двоичного кода на цифровых входах в ток на аналоговом выходе пропорциональный значениям кода.

Анализируя выше приведённые уравнения видно что в нашем алгоритме должны использоваться операции умножения на константу и сложения. Где операция умножения реализуется путём сдвига значения переменной и суммирования.

Техническое задание требует чтобы на выходе устройства действовало напряжение (-3…+3) В. Следовательно, для преобразования тока на выходе ЦАП в напряжение (-3…+3) В необходимо дополнительное устройство. Таким устройством может послужить операционный усилитель (ОУ).

Обработка отсчетов производится программой, хранящейся в ПЗУ. Размер одного цикла программы должен быть таким, чтобы к моменту прихода следующего отсчета устройство закончило обработку предыдущего и находилось в состоянии готовности.

**Общий алгоритм функционирования устройства и его описание**

При включении питания фильтр должен быть настроен на соответствующую работу. Для этого используется программа инициализации. В ПЗУ должны быть заложены данные о конфигурации и синхронизации устройства. При включении питания на адресной шине микропроцессора устанавливается начальный адрес 0000, поэтому программу инициализации следует располагать, начиная с этого адреса.

Вторым этапом работы обобщённого алгоритма является основная программа- алгоритм фильтрации. Таким образом, обобщённый алгоритм функционирования устройства имеет вид:

**ИНИЦИАЛИЗАЦИЯ:**

Настройка портов,

установка маски прерываний, организация стека,

настройка таймера

**ОСНОВНАЯ ПРОГРАММА:**

Алгоритм фильтрации

Загрузка данных из порта ввода А.

Вывод данных в порт вывода В.

Инициализация устройства включает в себя следующие этапы:

1. Установка маски прерываний
2. Настройка портов ввода-вывода
3. Организация стека
4. Настройка таймера

После программы инициализации, т.е. после настройки работы, таймер начинает формировать импульсы следующие с частотой дискретизации, которые поступают на вывод STB A (PC2) порта С работающего в информационном режиме. В этом режиме через выводы PC0, PC1, PC2 происходит обмен управляющими сигналами для порта А .

По переднему фронту импульса на линии STB A формируется сигнал BF (PC1) "буфер полон" который используется в качестве сигнала квитирования. По заднему фронту импульса STB A формируется сигнал INTR (PC0) который является сигналом прерывания. С приходом сигнала запроса прерывания RST 7,5 процессор переходит в режим обслуживания прерываний. Соответственно программе обслуживания прерываний 7,5 производится считывание данных из порта ввода А.Затем процессор начинает выполнение основной программы. Основная программа, содержащая в себе алгоритм фильтрации производит операции над выборками, поступающих с внешнего устройства на порт ввода, в соответствии с заданным алгоритмом. По окончанию вычислений процессор выдаёт данные в порт вывода В. И процесс повторяется вновь.

**Режим прерываний**

Методология построения системы прерываний ВМ85 подчинена архитектуре ВМ80, однако число возможных источников прерываний на аппаратном уровне увеличено с одного до пяти. Наряду с типовым векторным запросом INTR (INT для ВМ80) введены ещё четыре, имеющие фиксированные векторы прерываний. Это означает, что при появлении соответствующего запроса управление передаётся на ячейку с фиксированным адресом приведённом в таблице 1:

**Таблица 1: Описание прерываний процессора ВМ85**

|  |  |  |  |
| --- | --- | --- | --- |
| Имя | Приоритет | Стартовый адрес | Вид сигнала |
| TRAP | 1 | 24CH | Переход из 0 в 1, затем в 1 |
| RST 7.5 | 2 | 3CH | Переход из 0 в 1 |
| RST 7.5 | 3 | 34CH | 1 |
| RST 7.5 | 4 | 2CH | 1 |
| INTR | 5 | Вводится при подтверждении прерывания | 1 |

Из таблицы видно, что стартовые адреса подпрограмм обслуживания прерываний находятся в области точек входа по команде RST n, n=0-7, но расположены посередине между ними. Это разъясняет ряд наименований , принятых для запросов RST n.5, n=5-7.

Все запросы, за исключением TRAP, могут быть запрещены или разрешены одновременно с помощью команд EI, DI, управляющих общим флагом разрешения прерываний IEN. Существует также возможность раздельного маскирования запросов типа RST независимо др. от друга, которое выполняется с помощью команды SIM. По команде SIM обеспечивается установка нового состояния маски в соответствии с содержимым аккумулятора. При выполнении этой команды содержимое аккумулятора интерпретируется следующим образом.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| А7 | А6 | А5 | А4 | А3 | А2 | А1 | А0 |
| SOD | SDEN | -- | R 7.5 | MSEN | M 7.5 | M 6.5 | M 5.5 |

M 7.5, M 6.5, M 5.5 – маски соответствующего прерывания

MSEN – разрешение установки маски прерываний

R 7.5 – бит, сбрасывающий триггер, где фиксируются запрос внешнего прерывания по входу RST 7.5

SDEN – Разрешение вывода данных, которые находятся в разряде A7

SOD – данные предназначенные для вывода через линию SOD

Установленная маска запрещает соответствующее прерывание. Смена маски в соответствии с А0-А2 выполняется только при разрешении её установки: MSEN=1. В противном случае функция установки маски подавляется.

Текущее состояние масок прерываний может быть прочитано по команде RIM.

При включении источника питания или пуске МП все индивидуальные маски устанавливаются, а флаг разрешения прерывания IEN сбрасывается, что приводит к запрету прерываний. Приём какого – либо запроса на прерывание тоже вызывает общее запрещение всех маскируемых прерываний, но при сохранении состояния индивидуальных масок. Кроме этого запрос TRAP сохраняет и состояние флага общего разрешения, которое может быть прочитано только после первой команды RIM. Вторая команда даёт результат текущего состояния маски, показывающего общий запрет прерываний.

В нашем случае маскирование с помощью режимного слова имеет вид:



Установка маски осуществляется по команде SIM. При этом режимное слово находится в аккумуляторе.

**Настройка портов ввода-вывода**

В качестве порта ввода и порта вывода в данной конфигурации используются соответственно порты А и В микросхемы РУ55. Порты А и В одинаковы, и каждый из них содержит 8-битовый регистр для буферного запоминания пересылаемого байта данных. Все разряды портов А и В используются для ввода и вывода параллельно, т.е. невозможна установка направления передачи для каждого разряда.

В зависимости от содержания РУС каждый порт (А и В) может работать в двух режимах: простого ввода – вывода ( без обмена управляющими сигналами) или ввода – вывода с квитированием сообщения ( с обменом управляющими сигналами). В режиме 0-го программируемого параллельного интерфейса (ППИ) порт С может осуществлять ввод- вывод параллельно по шести линиям.

Режим с квитированием сообщений соответствует режиму 1 ППИ. В этом случае линии порта С разбиваются на две группы , одна из которых обеспечивает обмен управляющими сигналами для порта А, а другая для порта В.Варианты использования линий порта С и соответственно организация работы всех портов представлены в следующей таблице:

|  |  |  |
| --- | --- | --- |
| Линии порта С | Варианты использования линии | |
| ALT 3 | ALT4 |
| РС 0 | INTR A | INTR A |
| РС1 | BF A | BF A |
| РС2 | STB A | STB A |
| РС3 | Порт вывода | INTR B |
| РС4 | То же | BF B |
| РС5 | >> | STB B |

**Примечание:** Вариант ALT1 соответствует использованию всех линий порта С в качестве портов ввода, а вариант ALT2 – в качестве портов вывода.

где INTR A/INTR B (Interrupt Request)- запрос прерывания по порту A/B;

BF A/BF B (Buffer Full) – буфер порта А/В полон/заняты

STA A/STA B (Strobe)- строб записи, соответствующий порту А/В

Временные диаграммы , поясняющие обмен управляющими сигналами при работе портов А и В в режиме с квитированием сообщений, представлены на рисунке:

BF

INTR

RD

PA7-PA0

STB

**Временная диаграмма работы порта А(В) со стробированием в режиме ввода**

По переднему фронту импульса на линии STB A, который формируется таймером, формируется сигнал BF (PC1) "буфер полон" который используется в качестве сигнала квитирования. По заднему фронту импульса STB A формируется сигнал INTR (PC0) который является сигналом прерывания.

С приходом сигнала запроса прерывания RST 7,5 процессор переходит в режим обслуживания прерываний. Соответственно программе обслуживания прерываний 7,5 производится считывание данных из порта ввода А.Затем процессор начинает обрабатывать данные в соответствии с заданным алгоритмом. По окончанию вычислений процессор выдаёт данные в порт В. И процесс повторяется вновь.

Сигналы BF и INTR остаются активными до выполнения микропроцессором команды IN PORT, по которой содержимое буферного регистра порта А вводится МП по сигналу RD. С началом импульса RD сбрасывается сигнал INTR , а с его окончанием – сигнал BF. Таким образом , сигнал BF =1 сохраняется в течении всего времени обращения к буферному регистру и хранению данных в нём. Сигнал INTR=1 сохраняется, пока записанный в регистр байт данных ожидает ввода в МП.

Если в МПС используется система прерываний, то выход INTR соединяется с соответствующим входом прерывания. Если она не используется, то МП может получить информацию о наличии сигнала прерывания и о состоянии других управляющих сигналов путём чтения слова состояния из РСС и анализа его содержимого.

Таким образом, использование управляющих сигналов при вводе или выводе данных позволяет не терять байты из-за того, что МП не ввёл вовремя байт из порта или же вывел очередной байт в порт до того, как внешнее устройство считало предыдущий байт.

Режим работы портов, а также управление таймером состоит в том, что в регистр управляющего слова записывается байт формат которого имеет следующий вид:



Биты D0, D1 управляющего слова задают направление передачи данных через порты А и В;

Биты D2,D3 определяют вариант использования порта С;

Биты D4,D5 разрешают или запрещают выработку сигналов прерывания INTR;

Биты D6,D7 содержат код TM2,TM1 команды управления таймером.

В нашем случае этот байт будет равен:



Запись в управляющий регистр производится записью режимного слова в порт с адресом 08H.

Пространство ввода-вывода я распределил следующим образом:

08Н – РУС (Регистр управляющего слова) и РСС (Регистр словосостояния) ;

09Н – Порт А (Порт ввода)

0АН –Порт В (Порт вывода)

0ВН – Порт С (Порт обработки сигнала квитирования)

0СН – Регистр хранения таймера (младший байт)

0DH - Регистр хранения таймера (старший байт)

Адресное пространство памяти, исходя из её объёма, распределено следующим образом:



**Настройка таймера**

Таймер используется для формирования последовательности импульсов следующих с частотой дискретизации. В качестве входного сигнала счётчика используется выход тактовых импульсов микропроцессора.

Для обеспечения частоты дискретизации 1.2 КГц коэффициент счёта таймера должен равняться:



Запись управляющего слова осуществляется в два регистра таймера старший и младший, микросхемы РУ55.

Загрузка управляющего слова производится в порты

C9H Порт ОD- старший байт

C4H Порт ОС- младший байт

**Функционирование таймера.** Блок таймера содержит 14-битовый вычитающий счетчик, 16-битовый регистр хранения, схему формирования выходного сигнала и схему управления таймером. Счетчик выполняет основную функцию данного блока, заключающуюся в подсчете импульсов, поступающих извне на вход TIMER IN. Регистр хранения в 14 младших битах содержит коэффициент счета N, который загружается в него при подготовке таймера к работе и задает начальное состояние счетчика. Число N может иметь любое значение от 2Н до 3FFFN. Два старших бита этого регистра составляют регистр режима (RR), в котором хранится код ТМ2ТМ1 заданного режима работы таймера. Схема формирования обеспечивает форму сигнала TIMER OUT в соответствии с режимом работы таймера. Этот сигнал обязательно изменяет свое значение в момент окончания счета. Схема управления реализует заданный режим работы таймера и, в частности, осуществляет перезагрузку числа N из регистра в счетчик при организации циклической (непрерывной) работы таймера. Программирование таймера состоит в загрузке двухбайтового слова в регистр хранения, после чего осуществляется запуск счетчика с помощью специальной команды (предусмотрен только программный запуск таймера). Во время работы счетчик уменьшает свое содержимое по фронту каждого импульса, поступающего на вход TIMER IN. Когда требуется производить отсчет реального времени, этот вход обычно соединяется с выходом PCLK генератора. При обращении к таймеру используются два адреса: младший байт N загружается по адресу А2А1АО = 101, старший — по адресу А2А1АО=101. Это позволяет загружать указанные байты в любом порядке.

Формат двухбайтового слова, загружаемого в таймер, представлен на рисунке.



Рис. 1 Формат команды управления таймером

Два старших бита М2М1 задают один из четырех режимов работы таймера, при которых вырабатываются соответствующие сигналы на выходе TIMER OUT (рис. 6.56), В режиме 0 поддерживается выходной сигнал TIMER OUT = 1 в течение первой половины счета и TIMER OUT —0 в течение второй. При нечетном N первый полупериод счета, соответствующий высокому уровню выходного сигнала, оказывается на единицу больше второго. Режим 1 аналогичен режиму 0 с той разницей, что каждый раз по окончании счета производится автоматическая перезагрузка N из регистра хранения в счетчик и повторный запуск счетчика. Таким образом, таймер работает в непрерывном режиме деления на N частоты повторения входных импульсов и генерирует выходной сигнал симметричной формы (скважность равна двум). В режиме 2 после окончания счета формируется одиночный импульс напряжения низкого уровня, длительность которого равна одному периоду входных (счетных) импульсов. Режим 3 отличается от режима 2 автоматической перезагрузкой и перезапуском счетчика, что обеспечивает непрерывность работы таймера. Это режим деления частоты с выработкой выходного сигнала несимметричной формы (скважность зависит от значения N).

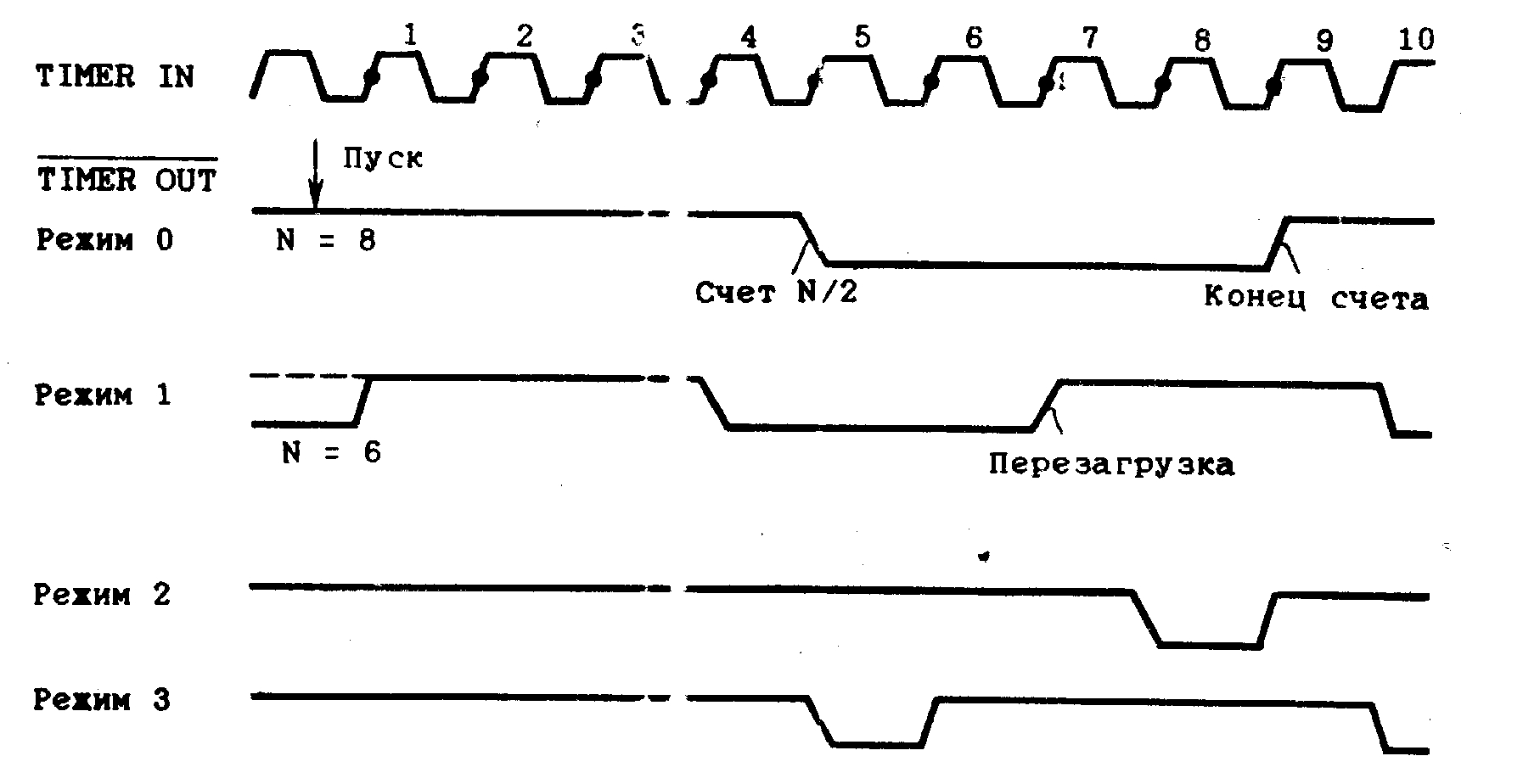


Рис .2 Временная диаграмма работы таймера

**Таблица 3: Режимы работы таймера**

|  |  |
| --- | --- |
| Режим работы таймера | НАИМЕНОВАНИЕ РЕЖИМА |
| 0 | Ждущий мультивибратор с программным запуском |
| 1 | Делитель частоты, генератор меандра |
| 2 | Делитель частоты, генератор импульсов несимметричной формы |
| 3 | Генератор задержанного строба |

Для нашего случая таймер использует **третий** режим работы.

Таким образом, управляющее слово таймера будет иметь следующий вид



Алгоритм основной программы

Запрет IRQ

Порт А→A(Xn)

0.4\*A(Xn)

А→[0819]

[0800](Xn-1)→A

A(Xn)→[0801]

0.56\*A(Xn-1)

A→[0819]

A→[081A]

0.04\*A(Vn-1)

[0806](Vn-1)→A

[081A]→A

A→B

[0818]→A

[0819]→A

A→C

A+B→A

A-C→A

A(Vn)→[0805]

A(Vn)\*0.189

A→[081B]

[0806](Vn-1)→A

0.378\*A(Vn-1)

A→[081C]

[0807](Vn-2)→A

[0807](Vn-2)→A

[0807](Vn-2)→A

0.189\*A(Vn-2)

A→[081D]

[080C](Yn-2)→A

0.250\*A(Yn-2)

A→[081E]

A→B

[081B]→A

[081C]→A

[081D]→A

A→C

A→D

[081E]→A

A+C→A

A+D→A

A+B→A

A(Yn)→[080A]

[0801]→[0802]

[0806]→[0807]

[0805]→[0806]

[080B]→[080C]

[080A]→[080B]

A(Yn)→

порт В

Разрешение

IRQ

**Обоснование аппаратной части устройства**

Набор К1821 определяет типовой состав аппаратных средств, образующих структуру вычислительного ядра системы. В его состав входят МП, ОЗУ, ПЗУ, схемы формирования сигналов синхронизации, микросхемы формирования сигналов управления системой. Полная структурная схема МП- устройства получается при объединении структуры вычислительного ядра и дополнительных аппаратных узлов.

Аппаратный состав фильтра в целом уже определен, незатронутым остается преобразователь ток-напряжение. Также следует произвести согласование адресов ОЗУ, ПЗУ, портов ввода-вывода с адресами МП.

Вспомогательные схемы целесообразно выполнять с наименьшими аппаратными затратами, т.е. следует стремиться к сокращению количества микросхем.

Необходимо также обеспечить соответствие адресных пространств ВМ85, РФ55 и РУ55. Так как для адресации ПЗУ необходимо 11 адресных линий, а для адресации ОЗУ – 8, то у МП остается еще 5 свободных адресных линий, которые можно использовать для выбора микросхем в процессе работы фильтра. Кроме того, необходимо учесть адресацию внутренних объектов РФ55 и РУ55.

Адресация портов и регистров направления передачи данных РФ55 осуществляется в соответствии с таблицей:

|  |  |  |
| --- | --- | --- |
| AD1 | AD0 | Адресуемый объект |
| 0 | 0 | Порт А |
| 0 | 1 | Порт В |
| 1 | 0 | Регистр направления передачи данных порта А |
| 1 | 1 | Регистр направления передачи данных порта В |

Адресация внутренних узлов РУ55:

|  |  |  |  |
| --- | --- | --- | --- |
| А2 | А1 | А0 | Адресуемый объект |
| 0 | 0 | 0 | Регистру РУС и РСС |
| 0 | 0 | 1 | Порт А |
| 0 | 1 | 0 | Порт В |
| 0 | 1 | 1 | Порт С |
| 1 | 0 | 0 | Таймер (младший байт) |
| 1 | 0 | 1 | Таймер (старший байт) |

В микросхемах РФ55 и РУ55 предусмотрены изолированные адресные пространства памяти и ввода-вывода. Для включения РФ55 в работу существует еще два входа – CS1 и CS2 (РУ55 – CS).

**Разработка и отладка программы на языке команд микропроцессора**

Как было показано ранее общий алгоритм программы состоит из двух основных частей: программы инициализации и основной программы (программы обработки отсчетов).

Алгоритм фильтрации:



Для хранения промежуточных результатов вычисления в ОЗУ выделяется область памяти. Она распределяется, как показано в таблице

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес ОЗУ | Данные | Адрес ОЗУ | Данные |
| 0801 |  | 0818 |  |
| 0802 |  | 0819 |  |
| 0803 |  | 081A |  |
| 0804 |  | 081B |  |
| 0805 |  | 081C |  |
| 0806 |  | 081D |  |
| 0807 |  | 081E |  |
| 0808 |  |  |  |
| 0809 |  |  |  |
| 080A |  |  |  |
| 080B |  |  |  |
| 080C |  |  |  |

Подготовка данных к следующим циклам обработки производится по правилу прибавления "-1" к индексам отсчетов т.е. отсчет становится, -> и т.д.



**Отладка программы**

Отладка программы производилась с использованием специального набора программ. В его состав входит программа-транслятор и отладчик. Транслятор преобразует исходную программу, написанную на языке ассемблера, в результирующую – так называемую объектную программу на языке команд микропроцессорной системы. Кроме объектной программы транслятор выдает листинг программы, содержащий распечатку исходной и объектной программ, таблицы использованных идентификаторов, сообщения об обнаруженных ошибках и другие виды диагностической информации, необходимые для отладки и документирования программы.

Также необходима программа-отладчик, позволяющая проводить весь цикл работ по отладке. В его состав входит система моделирования и отладки, состоящая из имитатора МП системы и программы управления отладкой.

Листинг программы приведен в приложении.

**Цель отладки**: в случае успешной отладки части программы сделать заключение о том что вся программа будет успешно работать.

**Средства отладки:** программное обеспечение выполненное руководителем курсового проекта ***Сальниковым Николай Ивановичем*** в виде программ DEB80.EXE, ASM80.EXE

**Ручной просчёт:**

Отладка программы в данной курсовой работе произведена следующим образом, т.е. произведена отладка подпрограммы PP6 умножения на 0.25. С помощью отладчика я выхожу на нужную мне строку программы и ввожу значения аккумулятора в данном случае на выходе подпрограммы получаем значение аккумулятора , теперь умножим



конечно о полученном результате можно говорить с определённой точностью, т.к. очевидно что при представлении 10-го числа 2-м с ограниченным числом разрядов точность уменьшается. Все эти проанализированные данные сохраняются в файл musor.trs (см. приложение)

Если выполнять ручной просчёт для критических значений то нужно оговориться, что от переполнения мы избавились ещё в начале программы при умножении на соответствующие коэффициенты. Т.е. можно смело говорить о том, что при выполнении данной подпрограммы, в процессе выполнения всей программы в целом, переполнения быть **не может**.

В результате отладки было установлено, что подпрограмма работает в соответствии с заданным алгоритмом.

**Оценка быстродействия устройства**

Оценка быстродействия необходима для проверки возможности работы устройства в реальном масштабе времени. Расчёт времени обработки производится по самому длинному пути алгоритма вычислений. Для этого необходимо определить число команд, а по ним число машинных тактов МП, длительность которых известна.

Быстродействие:



максимальное время вычислений:



интервал дискретизации



Так как, tmax <Tд, то быстродействие устройства вполне достаточно для обработки выборки в реальном масштабе времени .

**Составление принципиальной схемы устройства и ее описание**

Принципиальная схема цифрового фильтра содержит след. микросхемы:

DD1 – МП К1821ВМ85

DD2 – ПЗУ К1821РФ55

DD3 – ОЗУ К1821РУ55

DD4 – ЦАП К1108ПА1Б

DА5 - ОУ К574УД1

DD6- К174ЛЕ5

Выходной узел работает в режиме биполярного выходного напряжения. Для наиболее полного использования динамических свойств ЦАП на выходе включён быстродействующий ОУ К574УД1, обладающий , выходным напряжением 10В и скоростью его нарастания 50-100 В/мкс. В цепи стабилизации включен стабилитрон КС 191Ф который задаёт опорное напряжение . В результате на выходе схемы формируется напряжение от -Uref до Uref. ЦАП К1108ПА1Б допускает изменение опорного напряжения от –10.24 В до +10.24 В.



В данном курсовом проекте использована типовая схема подключения ОУ к ЦАП, где коэффициент передачи регулируется резистором R8. Т. е. для обеспечения заданного напряжения на выходе (-3 …+3)В необходимо подстроить R8 и подать на ЦАП опорное напряжение 3 В.

**Заключение**

В данной курсовой работе была построена схема цифрового устройства и разработана программа, обеспечивающая работу данного устройства как цифрового сглаживающего фильтра. Устройство имеет высокое быстродействие и в полной мере удовлетворяет требованиям технического задания. Необходимость в высоком быстродействии связана со стремлением обрабатывать в реальном масштабе времени широкополосные сигналы.

Цифровые фильтры имеют свои преимущества и недостатки перед аналоговыми.

Аналоговые фильтры физически реализуемы, если в их передаточных функциях степень полинома числителя не выше степени полинома знаменателя. Цифровые фильтры не предъявляют таких ограничений, и, таким образом, они могут иметь характеристики, добиться которых в аналоговых фильтрах невозможно.

К недостаткам цифрового фильтра можно отнести неточность представления коэффициентов вследствие ограниченной разрядности процессора.

**Список использованной литературы**

1. Балашов Е.П. и др. Микро- и мини-ЭВМ / Е.П. Балашов, В.Л. Григорьев, Г.А. Петров: Учебное пособие для вузов. – Л.: Энергоатомиздат. Ленингр. отд-ние, 1984
2. Калабеков Б.А. Микропоцессоры и их применение в системах передачи и обработки сигналов: Учеб. пособие для вузов. –М.: Радио и связь, 1988.
3. Микропроцессорный комплект К1810: Структура, прграммирование, применение: Справочная книга/ Ю.М. Казаринов, В.Н. Номоконов, Г.С. Подклетнов, Ф.В. Филиппов; Под ред. Ю.М. Казаринова.- М.: Высш. шк., 1900.
4. Микропроцессоры: системы программирования и отладки / В.А. Мясников, М.Б. Игнатьев, А.А. Кочкин, Ю.Е. Шейнин; Под ред. В.А. Мясникова, М.Б. Игнатьева. – М.: Энергоатомиздат, 1985.
5. Проектирование импульсных и цифровых устройств радиотехнических систем: Учеб. Пособие для радиотехнич. спец. вузов/Гришин Ю.П., Катаков В.М. и др.; Под ред. Ю.М. Казаринова. – М.: Высш. шк., 1985.
6. Рафикузаман М. Микропрцессоры и машинное проектирование микропроцессорных систем: В 2-х кн. Пер. С англ.-М.: Мир, 1988.