

Министерство образования Российской Федерации

Рыбинская государственная авиационная технологическая академия

В. А. ВИШНЯКОВ

СИСТЕМЫ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ

УЧЕБНОЕ ПОСОБИЕ

Рыбинск 2000

## ВВЕДЕНИЕ

Цифровая обработка сигналов охватывает широкий спектр практических приложений: цифровая фильтрация, кодирование и декодирование информации, распознавание звуков, обработка изображений, спектральный анализ, цифровая звукотехника, медицина, измерительная техника, управляющие системы. Для решения этих задач необходимо иметь аппаратно-программные системы, реализующие обработку сигналов в соответствии с современными методами и алгоритмами обработки информации. Такие системы и методы рассматриваются в пособии.

Пособие дает основные сведения о наиболее важных способах анализа, разработки и проектирования цифровых систем в реальном времени. Оно рассчитано на студентов, подготовленных в областях электроники, программирования и математики.

### 1. СИГНАЛЫ И ПРОЦЕССОРЫ ИХ ОБРАБОТКИ

Сигналы применяются для передачи информации. Существует большое многообразие сигналов. Они бывают непрерывными и дискретными. Если сигнал непрерывен и по величине, и по времени, то он является аналоговым. Дискретный по величине и по времени сигнал называется цифровым. Чтобы извлечь информацию из сигнала, его нужно обработать, предварительно преобразовав из аналоговой формы в цифровую.

Сигналы бывают случайными (стохастическими) или детерминированными. Детерминированные сигналы описываются неслучайными функциями в любой момент времени  $t$ . Промежуточным классом сигналов являются квазидетерминированные.

На практике используются модулированные и немодулированные виды сигналов. Широко применяются сигналы гармонического вида (рис.1).

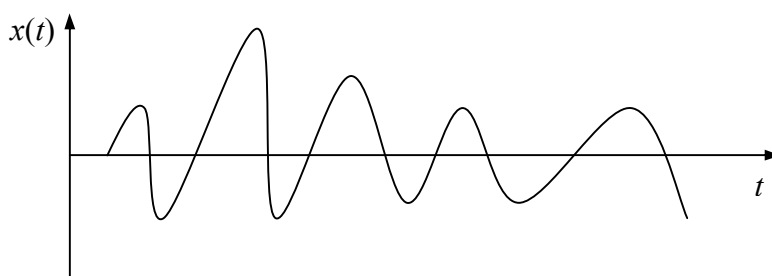


Рис.1

Модуляции подвергаются такие параметры, как амплитуда, частота и фаза или комбинация этих параметров. Для описания сигналов применяются две

формы: прямая и косвенная. При прямой форме сигнал задается как функция времени:  $x = S(v, t)$ , где  $v$ -информационный параметр сигнала.

При косвенной форме сигнал может описываться законом распределения вероятностей его величины, векторным определением сигнала, комплексным заданием, дифференциальным уравнением и др.

Задачи, решаемые с использованием сигналов:

- кодирование информации при ее передаче параметром сигнала;
- декодирование информации из принятого сигнала;
- обнаружение сигнала на фоне помехи;
- распознавание сигналов;
- измерение параметров сигнала;
- синтез звуковых сигналов;
- синтез видеосигналов.

Типовые устройства обработки сигналов можно представить в виде рис.2.



Рис.2

Цифровой сигнальный процессор (ЦСП) лучше адаптирован, чем универсальный, к типовым вычислениям большого объема в реальном времени.

Отличительные особенности ЦСП:

- использование гарвардской архитектуры (память делится на два блока: для хранения программ и хранения данных; используются независимые шины; увеличивается производительность);
- сокращенная длительность командного цикла;
- конвейерный принцип обработки (двух- или трехкаскадный конвейер);
- применение аппаратного умножителя (операция умножения производится за один машинный цикл);
- включение в систему специальных комплексных команд (например, умножение с накоплением);
- наличие преобразователей информации АЦП и ЦАП (не у всех ЦСП).

В настоящее время выпускаются сигнальные процессоры 16-, 24-, 32-разрядные с фиксированной и плавающей точкой.

ЦСП фирмы Texas Instruments приведены в таблице.

Тип	Время цикла, нс	Представление чисел, разрядность	Память слов		
			ОЗУ	ПЗУ	Внешняя
TMS32010	200	Фиксированная точка, 16	144 × 16 данных	1,5К×16 программ, данных	4К × 16 программ, данных
TMS320C25	100	Фиксированная точка, 16	544 × 16 данных	4К × 16 программ, данных	64К × 16 данных
TMS320C30	60	Плавающая точка, 32	2×(1К×32) данных 64 × 32-КЭШ	4К × 16 программ, данных	16М
TMS320C40-50	40	Плавающая точка, 32	2К	4К	4G

Новая разработка фирмы TMS320C80 имеет следующие параметры: содержит 4 миллиона транзисторов, производительность – 2 млрд. операций в секунду, включает четыре 32-разрядных ЦСП и центральный RISK процессор, 50 Кбайт внутреннее ОЗУ, 2 программируемых видеоконтроллера, обладает возможностью линейной и двухкоординатной адресации.

## 2. ЦИФРОВОЙ СИГНАЛЬНЫЙ ПРОЦЕССОР TMS32010

### Технические данные микроЭВМ

- Длительность машинного (командного) цикла – 200нс
- Встроенная память данных – 144 × 16
- Встроенная память программ (ПЗУ) – 536 × 16.
- Возможность расширения памяти программ (вне кристалла) – до 4К × 16
- Разрядность команд и данных – 16
- Разрядность АЛУ и аккумулятора – 32 разряда
- Время умножения (16 × 16 разрядов) – 200 нс
- Количество каналов I/O – 8
- Скорость передачи по двунаправленной 16-разрядной шине данных 40Мбит/с
- Арифметика чисел – с фиксированной точкой в дополнительном коде с учетом знака

### Структурная особенность ЦСП (Приложение 1)

МикроЭВМ построена по гарвардской архитектуре. Каждая память имеет свою шину данных. Память программ - типа ROM (ПЗУ). Память данных - типа RAM (ОЗУ). Шины данных связаны между собой, и имеется возможность пере-сылки данных (констант) из ПЗУ в ОЗУ.

Программный счетчик используется для адресации команд (РС). Этот счетчик команд 12-разрядный, т.е. максимальное количество команд, которое можно реализовать на данной микроЭВМ –  $2^{12} = 4096$ . Вне микроЭВМ можно использовать  $4096 - 1536 = 2560$  ячеек. Имеется стек емкостью 4 слова. В него временно помещается адрес команды.

*Команды*, которые записаны в ПЗУ последовательно, выводятся на исполнение.

*Операционная часть* содержит умножитель и АЛУ. Умножитель (производит одно умножение за один машинный цикл) имеет вход и выход. По входу расположен 16-разрядный регистр Т. Сюда записывается один из сомножителей. Второй сомножитель поступает на умножитель из ОЗУ. Выходной результат сохраняется в 32-разрядном регистре Р. Результат умножения через мультиплексор MUX может быть переписан в АЛУ (ALU). АЛУ универсального типа выполняет операции «+», «-», а также циклические операции.

Данные в АЛУ подаются либо с мультиплексора MUX, либо со сдвигателя SHIFTER. Сдвиг происходит на 0-15 разрядов влево. Второй вход АЛУ всегда связан с аккумулятором АСС – регистром временного хранения результата. АСС является источником операнда для АЛУ.

*Память данных* имеет 8-разрядную шину адреса. Некоторые команды, хранящиеся в ПЗУ, являются командами с прямой адресацией с адресным полем, величиной в 7 разрядов. При этом можно адресоваться только к  $2^7 = 128$  ячейкам. Поэтому реализована постраничная адресация памяти:

0 страница – адресация к ячейкам 0-127,

1 страница – адресация к ячейкам 128-143.

Для постраничной адресации необходимо выбрать номер страницы. Это осуществляется через триггер DP : 0 – выбирается нулевая страница, 1 – первая страница.

Регистры AR0 и AR1 – вспомогательные 16-разрядные, могут выполнять 3 функции:

- 1) временное хранение информации;
- 2) используются при косвенной адресации;
- 3) организация счётчиков циклов.

Содержимое регистров может инкрементироваться и декрементироваться. Регистр ARP производит коммутации (выбор) регистров AR0 и AR1. В текущий момент времени может использоваться один регистр.

В качестве внешних устройств выступают внешняя память и порты ввода/вывода. С помощью шины D15-D0 данные могут передаваться и приниматься. Для адресации используется шина A11-A0, три младшие линии которой применяются для адресации портов PA2-PA0.

Внешними устройствами управляет Controller. Назначение выводов контроллера:

X2/CLK IN – вход при реализации встроенного или внешнего генератора;

X1- вход при реализации встроенного генератора;

CLKOUT – выход синхронизации;

WE – (активен при низком потенциале) сигнал означает, что микроЭВМ находится в режиме выполнения команды OUT;

DEN – сигнал означает, что микроЭВМ находится в режиме выполнения команды IN;

MEN – сигнал означает, что ЭВМ выполняет выборку команды из внутренней или внешней памяти;

BIO – вход проверки готовности к обмену с портами;

MC/MP – вход задания режима: либо микроЭВМ, либо микропроцессор (в режиме микропроцессор используется только внешняя память);

INT – вход прерывания программ и команд;

RS – вход сброса в ноль счётчика команд.

Регистр состояния ЦСП пятиразрядный со следующими полями:

DP	ARP	INTM	OV	OVM
----	-----	------	----	-----

где DP – указатель номера страницы;

ARP – указатель вспомогательных регистров;

INTM – бит маскирования прерывания;

OV – разряд переполнения аккумулятора;

OVM – бит управления режимом переполнения.

Система команд ЦСП (Приложение 2) включает 60 команд. Используются команды длиной 1 и 2 слова. Команды длиной 2 слова – это команды перехода. Во втором слове записывается адрес перехода, в первом – код операции. Применяются безадресные, одноадресные и двухадресные команды. Система команд использует 3 метода адресации: непосредственная, прямая, косвенная.

При непосредственной адресации (безадресные команды) в команде фигурирует только код операции.

### Формат прямой адресации одноадресной команды

15							8	7	6						0
								0							

Биты:

0-6 – адрес операнда;

7 – признак прямой адресации;

8-15 – код операции.

### Формат косвенной адресации одноадресной команды

15							8	7	6	5	4	3	2	1	0
								1	0				0	0	

Биты:

0, 3 – управляют триггером ARP. Если в 3 разряде 0, то в ARP помещается то, что записано в 0 разряде, иначе состояние ARP не меняется;

4-5 – определяют операции над вспомогательными регистрами AR0 и AR1 (INC, DEC);

8-15 – код операции.

## 3. ПРОЕКТИРОВАНИЕ СИСТЕМ НА БАЗЕ ПРОЦЕССОРОВ ОБРАБОТКИ СИГНАЛОВ

В одном корпусе на одном кристалле не укладываются все элементы, необходимые для организации ЭВМ. Требуются внешние устройства для надлежащей работы ЭВМ. Процессоры TMS не содержат АЦП и ЦАП, возникает задача подключения к микроЭВМ АЦП и ЦАП.

Схема подключения АЦП и ЦАП к ЦСП показана на рис.3.

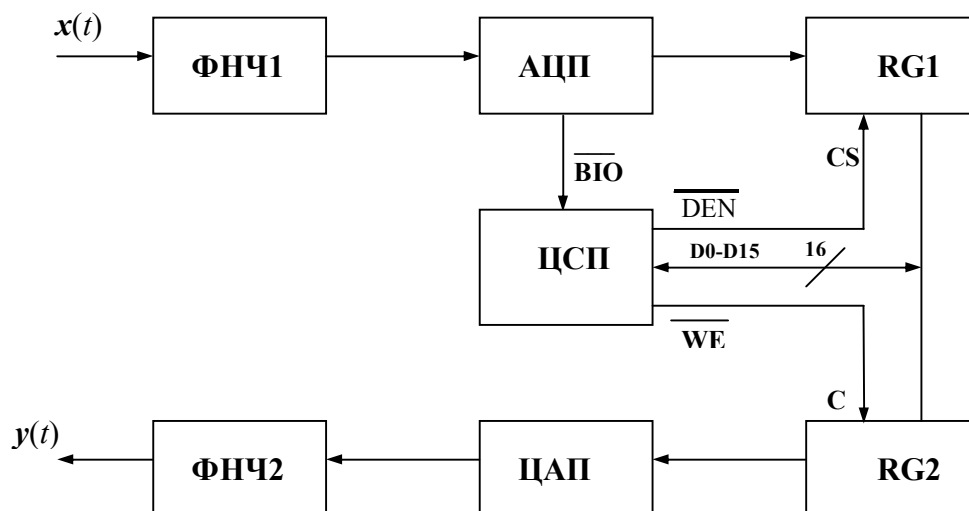


Рис.3

Для устранения эффектов наложения спектр входного сигнала ограничивается полосой пропускания ФНЧ1, меньшей, чем половина частоты дискретизации АЦП. Вход ЦСП  $\overline{\text{BIO}}$  используется для определения ЦСП момента завершения аналого-цифрового преобразования АЦП. Регистр RG1 необходим для согласования скоростей работы АЦП и ЦСП. Сигнал  $\overline{\text{DEN}}$  возникает при выполнении команды IN и служит для ввода данных из RG1 в ЦСП. При выполнении ЦСП команды OUT возникает сигнал  $\overline{\text{WE}}$ , по которому данные из ЦСП переносятся для временного хранения в регистр RG2, а из RG2 – в ЦАП. ФНЧ2 обеспечивает сглаживание выходного сигнала с ЦАП.

На практике приходится использовать и внешнюю память, так как внутренней памяти не хватает. В TMS 32010 возможна адресация к 4К ячейкам. Объем внешней памяти можно организовать значительно большим с постраничной структурой. Предположим надо организовать внешнюю память объемом 16К×16. Имеются микросхемы 8К×8. Необходимы 4 микросхемы. Схема подключения микросхем к сигнальному процессору показана на рис.4.



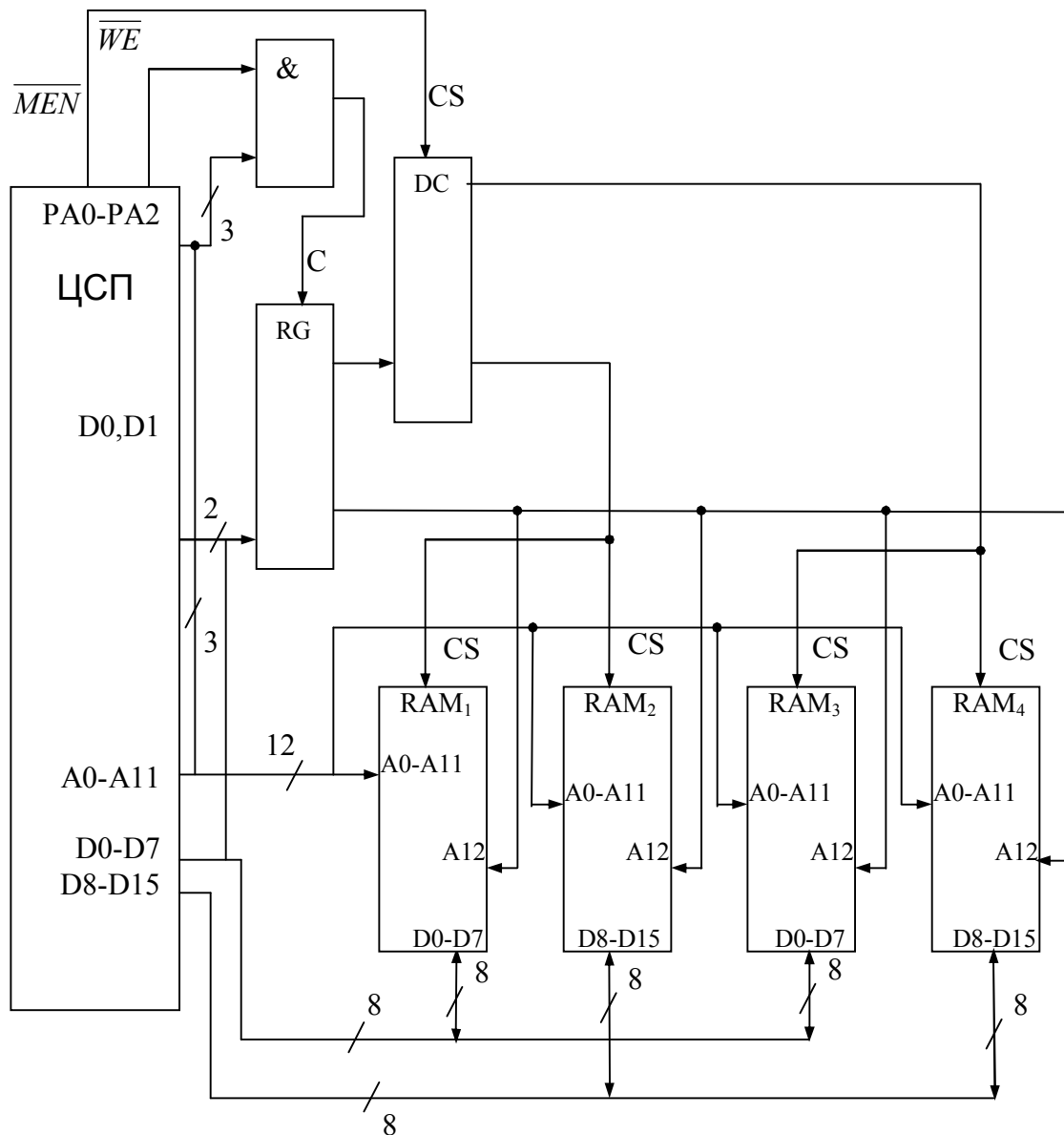


Рис.4

По команде OUT N, PA7 возбуждаются сигналы  $\overline{MEN}$  и  $\overline{WE}$  и номер страницы N записывается в регистр RG, где 7 – номер порта (RG). Номер страницы определяет дешифратор, выбирающий пары микросхем.

Существует задача дополнения аппаратной части микроЭВМ с тем, чтобы была возможность передачи в ЭВМ сигнала прерывания программы и определения номера внешнего устройства, сделавшего запрос.

Одна из возможных схем прерываний  $n$  внешними устройствами может иметь вид, показанный на рис.5 .

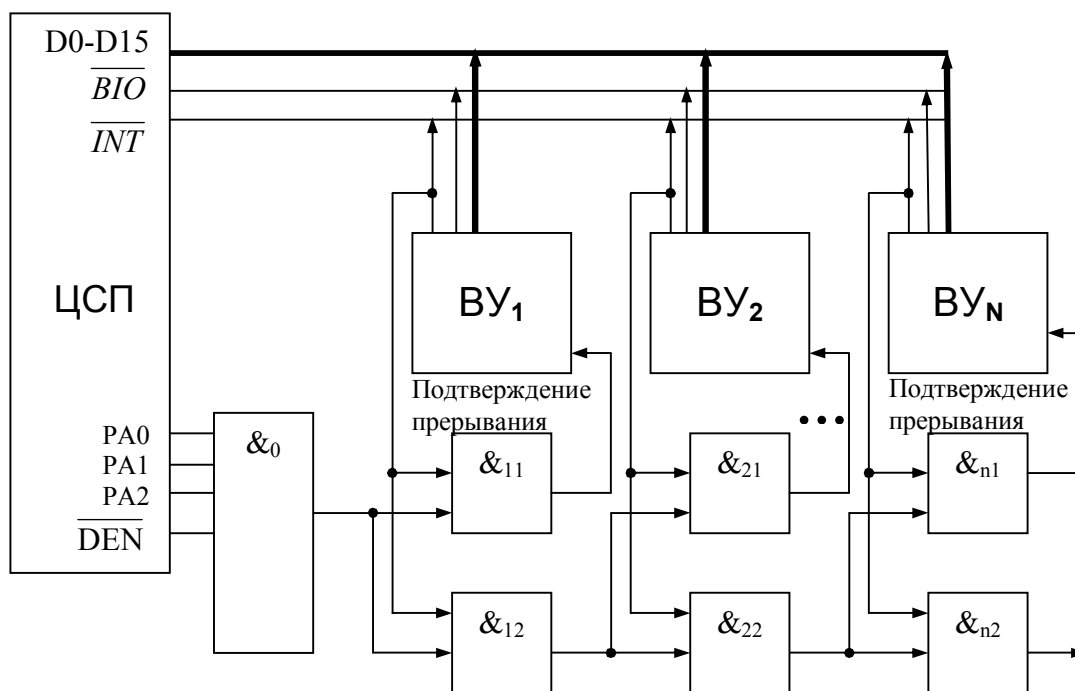


Рис.5

Внешнее устройство ВУ<sub>1</sub> имеет самый высокий приоритет. Приоритет убывает с увеличением номера ВУ. ВУ, которое хочет провести обмен информацией, посылает сигнал  $\overline{INT}$ . Процессор заканчивает выполнение текущей команды, засылает содержимое счетчика команд в стек. В счетчик команд загружается 2. В ячейке с номером 2 хранится первая команда обработки прерывания. В ячейке 0 хранится первая команда (В – безусловный переход), в ячейке 1 записан адрес перехода (W). ЭВМ начинает выполнение подпрограмм обработки прерывания. При этом она обязательно должна выполнить команду IN A, PA7, по которой возбуждается сигнал на выходе элемента &0 и с помощью последующих элементов & производится определение ВУ, сделавшего запрос прерывания. ВУ, пославшее запрос, формирует сигнал  $\overline{BIO}$  и выставляет свой номер на шину данных. ЭВМ считывает номер с шины и на основании его определяет номер подпрограммы, которую ей необходимо выполнить.

#### 4. ПРОГРАММИРОВАНИЕ СИГНАЛЬНЫХ ПРОЦЕССОРОВ

Рассмотрим одну из характерных задач, решаемых ЦСП. Это задача фильтрации полезной информации, передаваемой сигналом. Типовой алгоритм обработки определяется выражением

$$y_k = \sum_{i=0}^{20} A_i x_{k-i},$$

где  $A_i$  – постоянные коэффициенты,  $x_j$  ( $j=0,1,\dots$ ) – отсчеты входного сигнала в момент времени  $t = jT_0$ ,  $T_0$  – период дискретизации сигнала,  $y_k$  – выходное значение ( $k=0,1,\dots$ ).

Для упрощения составления программы рассмотрим гибридную функционально-алгоритмическую схему обработки (рис.6).

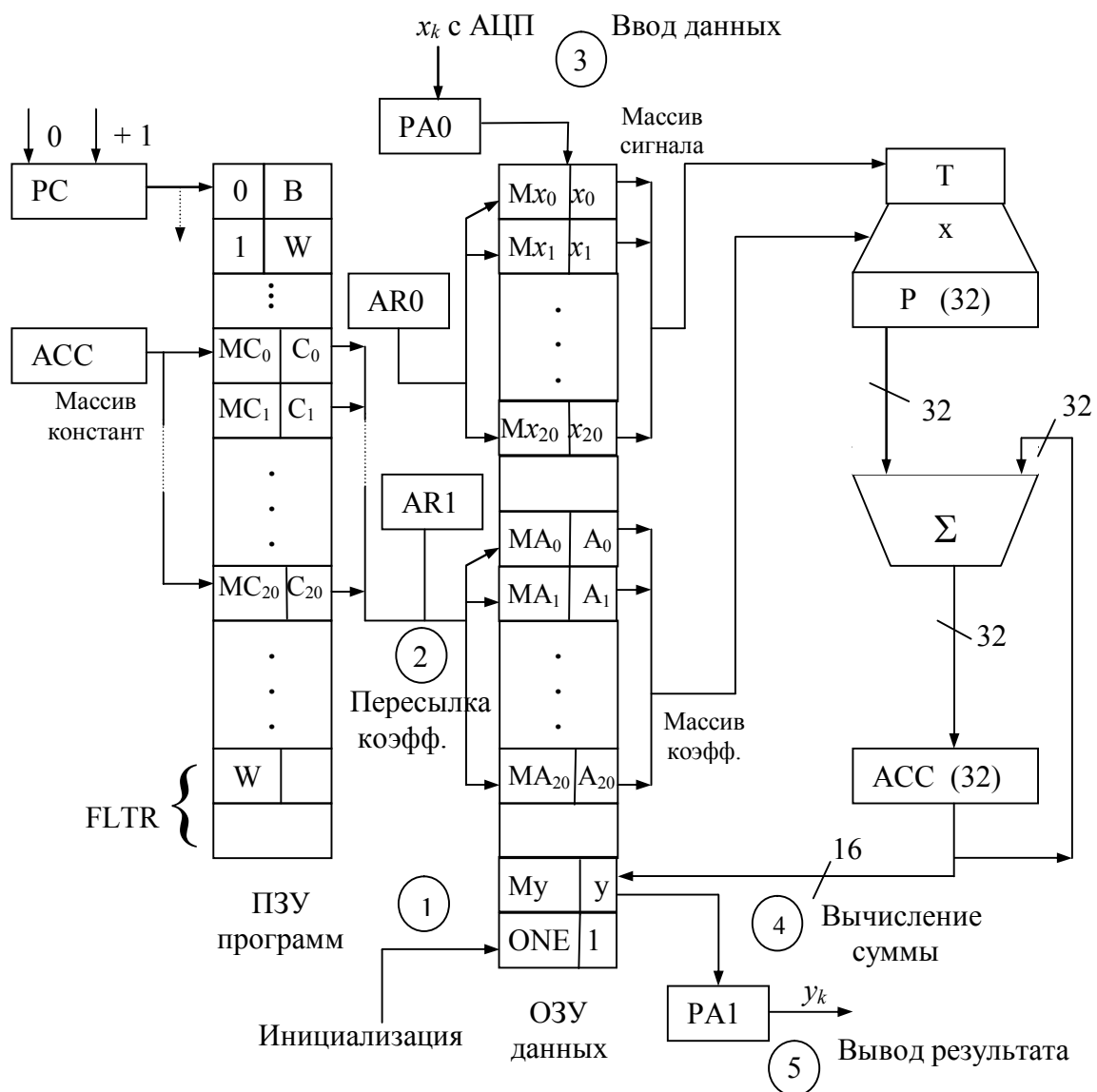


Рис.6

Разделим программу фильтрации FLTR на пять фрагментов.

## 1. Инициализация

W	LDPK 0	;установка нулевой страницы ;памяти данных $0 \rightarrow DP$
	LACK ONE	; $1 \rightarrow ACC$
	SACL ONE	; $1 \rightarrow ONE$

2. Пересылка коэффициентов из ПЗУ в ОЗУ ( $C_i = A_i$ )

	LACK MC0	; $MC0 \rightarrow ACC$ , загрузка адреса коэфф. C0
	LARK AR0, 20	; $20 \rightarrow AR0$ , организация счетчика
	LARK AR1, MA0	; $MA0 \rightarrow AR1$ , загрузка адреса коэфф. A0
RCONST	LARP 1	; $1 \rightarrow ARP$ , выбор регистра AR1
	TBLR *+, AR0	;пересылка из ПЗУ в ОЗУ: ;(MC0) $\rightarrow$ MA0, ; $AR1 = AR1 + 1$ ; выбор AR0
	ADD ONE	; $(ACC) = (ACC) + 1$
	BANZ RCONST	;если $(AR0) \neq 0$ , ;то переход на метку RCONST ;и $(AR0) = (AR0) - 1$ , иначе $(PC) = (PC) + 1$

## 3. Ввод данных

F	BIOZ F	;переход на метку F, если ;внешний сигнал BIO=0; организуется ;цикл ожидания, процессор ждет, пока ;АЦП будет готово
	IN MX0, PA0	;ввод данных $(PA0) \rightarrow MX0$

## 4. Вычисление суммы

	LARK AR0, 20	;загрузка $20 \rightarrow AR0$ , организация счетчика
	LARK AR1, MA20	; $MA20 \rightarrow AR1$ , загрузка адреса коэфф. A20
	ZAC	; $0 \rightarrow ACC$
	LARP 0	; $0 \rightarrow ARP$ , выбор регистра ARO
	MPYK 0	; $0 \rightarrow P$
LOOP	LTD *, AR1	; $((AR0)) \rightarrow T$ , т.е. $x20 \rightarrow T$ , $((AR0)) \rightarrow (AR0) + 1$ , ; т.е. $x21 = x20$ , ; $(ACC) + (P) \rightarrow ACC$ , ; $1 \rightarrow ARP$ , т.е. выбор AR1
	MPY *-, AR0	; $(T) \times ((AR1)) \rightarrow P$ , т.е. $x20 \times A20 \rightarrow P$ , ; $(AR1) - 1 \rightarrow AR1$ , ; $0 \rightarrow ARP$

BANZ LOOP	;если (AR0)≠0, то переход к
	;метке LOOP, (AR0)-1→AR0
APAC	;(ACC)+(P) →ACC, т.е. вычисление у

## 5. Вывод результата

SACL MY	; y→MY
OUT MY, PA1	;(y) →PA1
B F	;переход на F

## 5. ДИСКРЕТИЗАЦИЯ СИГНАЛОВ

Сигналы, которые обрабатываются в ЭВМ, представляются в дискретном виде. Дискретизацию производит АЦП.

Пусть имеется некий сигнал  $x(t)$ , где  $t$  – время (рис.7). Введем дискретное время  $t=kT_0$ ,  $k=t/T_0$ ,  $k=0,1,2,\dots$ ,  $T_0$  – период дискретизации. Дискретный сигнал обозначают  $x[kT_0]=x_k$ .

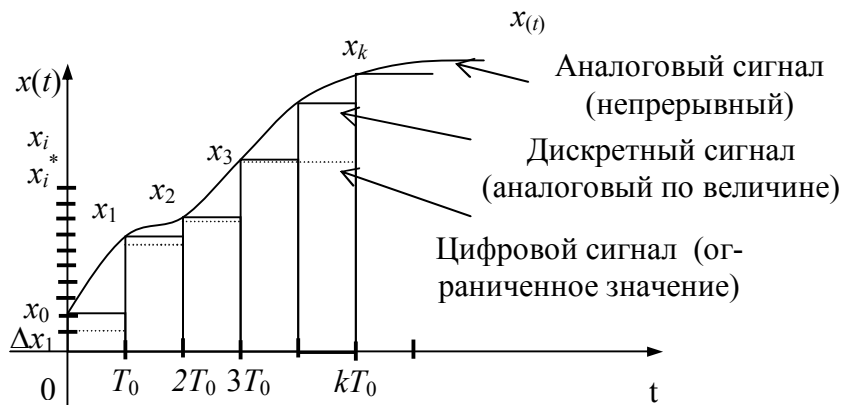


Рис.7

Кроме дискретизации по времени сигнал подвергается дискретизации (квантованию) по величине. Обычно это округление с шагом  $\Delta x$  значения  $x_k$  к ближайшему меньшему уровню. Получается цифровой сигнал

$$x^*[kT_0] = x_k^*.$$

Если  $P \gg 1$ , где  $P$  – разрядность АЦП, то можно считать  $x_k^* \approx x_k$ . В этом случае АЦП можно рассматривать как линейный элемент.

Пусть  $x^{\min} \leq x \leq x^{\max}$ , тогда количество уровней преобразования можно выразить следующим образом:

$$N = \frac{x^{\max} - x^{\min}}{\Delta x},$$

где  $\Delta x$  определяет ошибку цифрового преобразования. Отсюда разрядность АЦП будет

$$P = \lceil \log_2 N \rceil + 1.$$

Как выбрать  $T_0$ ? Верхнее значение  $T_0$  задается неравенством Котельникова:

$$T_0 \leq \frac{T_{\text{верхн.}}}{2} = \frac{1}{2F_{\text{верхн.}}},$$

где  $F_{\text{с}} -$  наивысшая частота в спектре сигнала  $x$ . Практически  $T_0 = \frac{1}{mF_{\text{с}}}$ , где  $m \geq 5$ .

Задача восстановления аналогового сигнала из дискретного решается путем пропускания дискретного сигнала через фильтр низкой частоты (ФНЧ) (рис.8).



Рис.8

Когда  $T_0 = \frac{T_{\text{с}}}{2}$ , то это ФНЧ – идеальный фильтр с прямоугольной амплитудно-частотной характеристикой. Он физически не реализуем. Если  $T_0 \ll \frac{T_{\text{с}}}{2}$ , то в качестве ФНЧ может использоваться реальный фильтр низких частот.

## 6. ЦИФРОВЫЕ ФИЛЬТРЫ

Цифровые фильтры (ЦФ) производят обработку сигналов. Большинство рассмотренных в первом разделе задач решается с применением ЦФ. Фильтрация означает выделение полезной информации на фоне прочей, которая является лишней или мешающей.

Цифровые фильтры могут быть реализованы программно, аппаратно или комбинированно. Наиболее легко они могут быть построены на основе ЦСП. В дальнейшем остановимся лишь на линейных цифровых фильтрах. При этом будем полагать, что разрядность фильтра достаточно велика, чтобы не влияли эффекты квантования сигналов.

Если на вход линейного цифрового фильтра действует сумма сигналов, то его выходной сигнал оказывается равным сумме реакций на действие каждого элементарного входного сигнала (рис.9).



Рис.9

Практически используются две основные разновидности ЦФ – нерекурсивные и рекурсивные фильтры.

## 7. НЕРЕКУРСИВНЫЕ ФИЛЬТРЫ

Нерекурсивное описание фильтра заключается в определении по известному входному сигналу  $x[kT_0]$  на любом шаге  $k$  выходного сигнала  $y[kT_0]$  с использованием импульсной реакции фильтра. Любой сигнал  $x_k$  можно представить как сумму импульсов одинаковой длительности  $T_0$  и разной величины (рис.10).

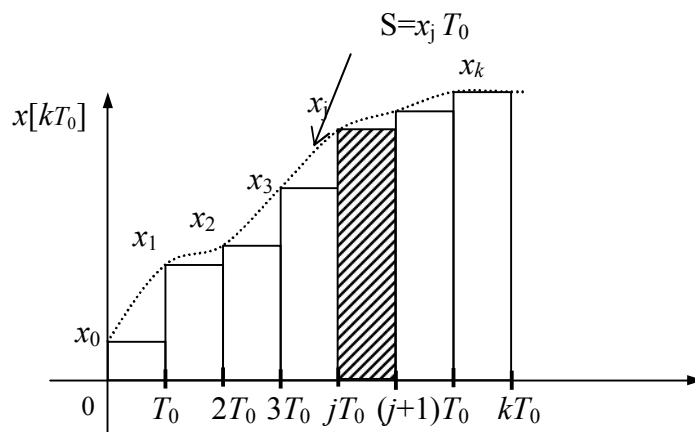


Рис.10

На основании принципа линейности выходной сигнал фильтра равен сумме элементарных выходных сигналов (реакций) на действие каждого входного импульса. В качестве элементарного входного сигнала используют нормированный импульс  $x(t) = \delta(t)$  с единичной площадью (рис.11), который называется дельта-импульс.

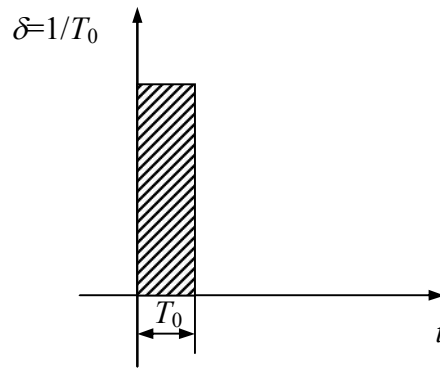


Рис.11

Если на вход ЦФ действует дельта-импульс  $\delta[kT_0]$ , то выходная импульсная реакция будет  $y[kT_0] = g[kT_0]$ . Каждый фильтр имеет единственный специфичный отклик  $g$  (рис.12). Таким образом, фильтры можно идентифицировать и сравнивать. Причем в пределе при  $T_0 \rightarrow 0$  получим импульсную реакцию  $g(t)$  для аналогового фильтра.

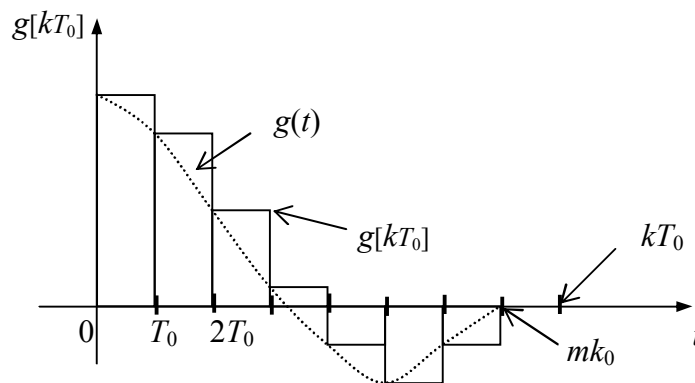


Рис.12

Импульсы, составляющие входной сигнал (рис.10), являются ненормированными с разной величиной. Поэтому для  $j$ -го импульса реакция будет равна  $x_j T_0 g(t - jT_0)$ . Вычислим суммарную реакцию  $y(t)$  в момент времени  $kT_0$ . Для этого нужно учитывать действия импульсов  $x_0, x_1, \dots, x_k$ , следовавших до момента времени  $t = kT_0$ . Пусть при  $k < 0$   $y = 0$ , тогда

$$y[kT_0] = x_0 T_0 g[kT_0] + x_1 T_0 g[(k-1)T_0] + x_2 T_0 g[(k-2)T_0] + \dots + x_k T_0 g[0] \quad \text{или}$$

$$y[kT_0] = \sum_{j=0}^k x[jT_0] T_0 g[(k-j)T_0].$$

Обозначим  $T_0 g = c$  – реакция цифрового фильтра, получим

$$y[kT_0] = \sum_{j=0}^k x[jT_0] c[(k-j)T_0], \quad \text{или} \quad y_k = \sum_{j=0}^k x_j c_{k-j}.$$



Этой формулой описывается нерекурсивный фильтр. Она называется “свёртка”. Входной сигнал свертывается с импульсной реакцией. Это цифровая свертка. От нее можно перейти к аналоговой свёртке при следующих преобразованиях:  $T_0 \approx d\theta$ ,  $jT_0 = \theta$ ,  $kT_0 = t$ , при  $T_0 \rightarrow 0$ , получим:

$$y(t) = \int_0^t x(\theta)g(t-\theta)d\theta \text{ - интеграл Дюамеля.}$$

Отметим, что  $c_j$  – постоянные коэффициенты, свойственные определенному фильтру. По виду коэффициентов можно определить фильтр. Возможно второе представление для формулы “свертки”, которое получается путем замены переменной  $k-j=n$ , имеем:

$$y_k = \sum_{j=0}^k x_{k-j}c_j \text{ .}$$

С ростом  $k$  увеличивается количество слагаемых в свертке и т.о. требуется на каждом шаге  $k$  производить большое количество операций умножения и сложения. Практически импульсную реакцию можно считать равной нулю с требуемой точностью, начиная с некоторого шага  $m$  (рис.12), тогда

$$y_k = \sum_{j=k-m}^k x_j c_{k-j} = \sum_{j=0}^m x_{k-j} c_j \text{ .}$$

Структурная схема, реализующая цифровой фильтр показана на рис.13 .

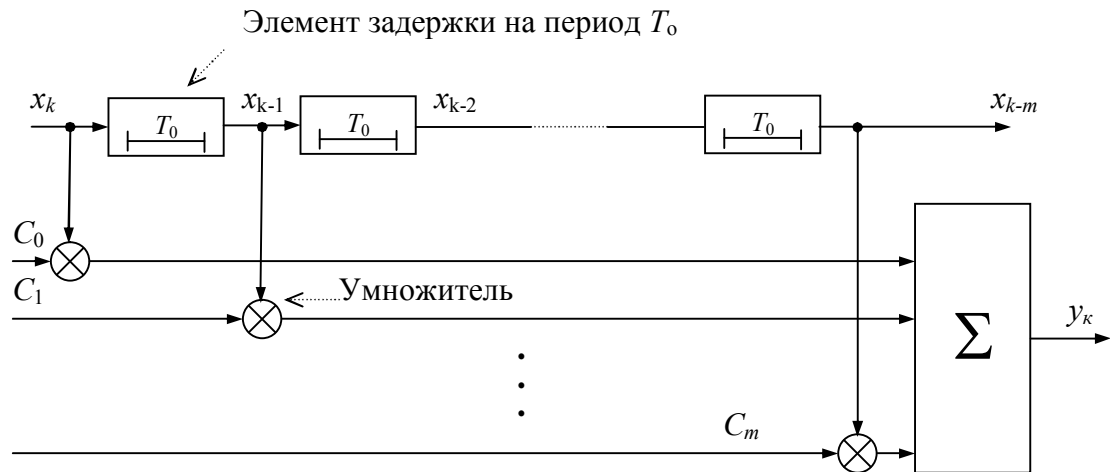


Рис.13

*Пример*

Пусть в качестве прототипа взят аналоговый фильтр Баттерворта первого порядка (рис.14). Найдем цифровой аналог этого фильтра.

Если на вход регистра подать единичный ступенчатый сигнал  $x(t)=1(t)$ , то на его выходе появляется реакция  $y(t)=1-e^{-t/\tau}$  (рис.15).

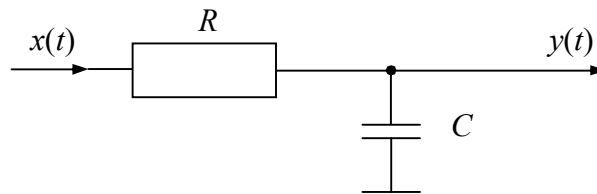


Рис.14

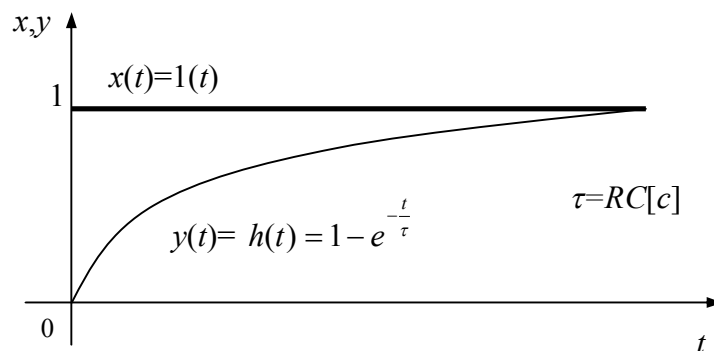


Рис.15

Так как фильтр линейный и справедливы соотношения  $\delta(t) = \frac{dl(t)}{d\tau}$ ,  $g(t) = \frac{dh(t)}{dt}$ , то отсюда следует

$$g(t) = \frac{1}{\tau} e^{-\frac{t}{\tau}}.$$

Перейдем к дискретному времени  $t = kT_0$ ,  $k=0,1,2,\dots$ , получим:

$$g[kT] = g_k = \frac{1}{\tau} e^{-\frac{kT_0}{\tau}}, \quad c_k = T_0 g_k.$$

Следовательно, выражение, определяющее работу ЦФ, имеет вид

$$y_k = \sum_{j=k-m}^k x_j c_{k-j} \quad \text{или} \\ y_k = \frac{T_0}{\tau} \sum_{j=k-m}^k x_j e^{-\frac{(k-j)T_0}{\tau}}.$$

Недостатком нерекурсивных фильтров является значительное количество операций сложения и умножения, если реакция фильтра на действие дельта-импульса затухает медленно.

## 8. РЕКУРСИВНЫЕ ФИЛЬТРЫ

Аналоговые системы описываются дифференциальными уравнениями. Дискретные системы такими уравнениями описать нельзя, потому что сигналы в них скачкообразные и производных от них не существует. Поэтому дискретные системы описывают разностными уравнениями, являющимися аналогами дифференциальных.

Пусть имеется некоторая произвольная функция  $f(t)$ ,  $t \geq 0$ . Перейдем к дискретному времени:  $t = kT_0$ ,  $k = 0,1,2,\dots$ , имеем  $f[kT_0] = f_k$ . Аналогом первой производной является первая разность:

$$\Delta f[kT_0] = f[kT_0] - f[(k-1)T_0].$$

Аналогично определяется вторая разность:

$$\Delta^2 f[kT_0] = \Delta f[kT_0] - \Delta f[(k-1)T_0] = f[kT_0] - f[(k-1)T_0] - f[(k-1)T_0] + f[(k-2)T_0] = f[kT_0] - 2f[(k-1)T_0] + f[(k-2)T_0].$$

Так же можно найти разность  $i$ -го порядка  $\Delta^i f[kT_0]$ .

Таким образом вводятся разности и составляются разностные уравнения для ЦФ. В этом разностном уравнении связь входных и выходных разностей устанавливается через некоторые коэффициенты (параметры системы). Каждую из разностей можно раскрыть, выражая через значение функции в соответствующие моменты времени. Таким образом получается рекуррентное уравнение, которое связывает значение сигнала  $y$  на  $k$ -том шаге и в более ранние моменты времени в зависимости от входного сигнала  $x$  в текущий и ранние моменты времени. Общий вид подобного уравнения:

$$a_0 y[(k-n)T_0] + a_1 y[(k-n+1)T_0] + \dots + a_{n-1} y[(k-1)T_0] + a_n y[kT_0] = b_0 x[(k-m)T_0] + b_1 x[(k-m+1)T_0] + \dots + b_{m-1} x[(k-1)T_0] + b_m x[kT_0] .$$

Поделив все составляющие уравнения на  $a_n$ , либо полагая  $a_n=1$ , после преобразования получим:

$$y[kT_0] = -\sum_{i=0}^{n-1} a_i y[(k-n+i)T_0] + \sum_{i=0}^m b_i x[(k-m+i)T_0] .$$

Входной сигнал должен быть известен  $x[kT_0]$ ,  $k=0,1,2,\dots$ . Тогда можно определить  $y$  на  $k$ -ом шаге, если заданы начальные значения  $y[0], y[-T_0], \dots, y[-(n-1)T_0]$ .

Замечательным свойством разностных уравнений является то, что их можно решить последовательно, используя рекурсию. Но необходимо иметь ввиду, что при рекуррентных вычислениях накапливается ошибка.

### Пример 1

Рассмотрим аналоговый фильтр первого порядка (рис.14), описываемый дифференциальным уравнением

$$\tau \frac{dy(t)}{dt} + y(t) = x(t) , \quad \tau = RC .$$

Введем дискретное время  $t = kT_0$ ,  $dt \approx T_0$ . Имеем:  $x(t=kT_0) = x_k$ ,  $y(t=kT_0) = y_k$ ,  $dy(t) \approx \Delta y(t) = \Delta y(t=kT_0) = \Delta y_k = y_k - y_{k-1}$ . Переходя от дифференциального уравнения к рекуррентному, получим:

$$y_k = (1 - T_0/\tau) y_{k-1} + x_k T_0/\tau .$$

Структура ЦФ, описываемая этим уравнением, приведена на рис.16 .

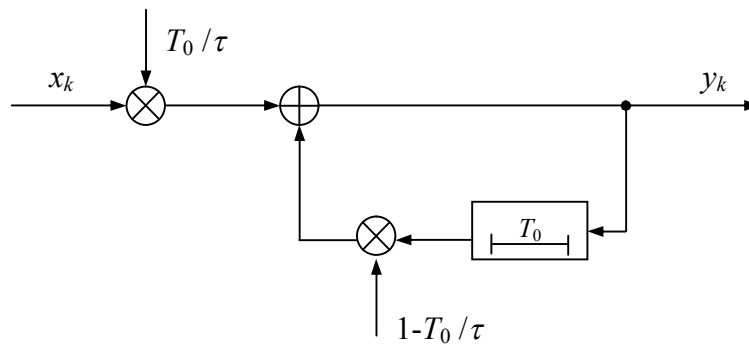


Рис.16

## Пример 2

Цифровой фильтр задан структурной схемой (рис.17).

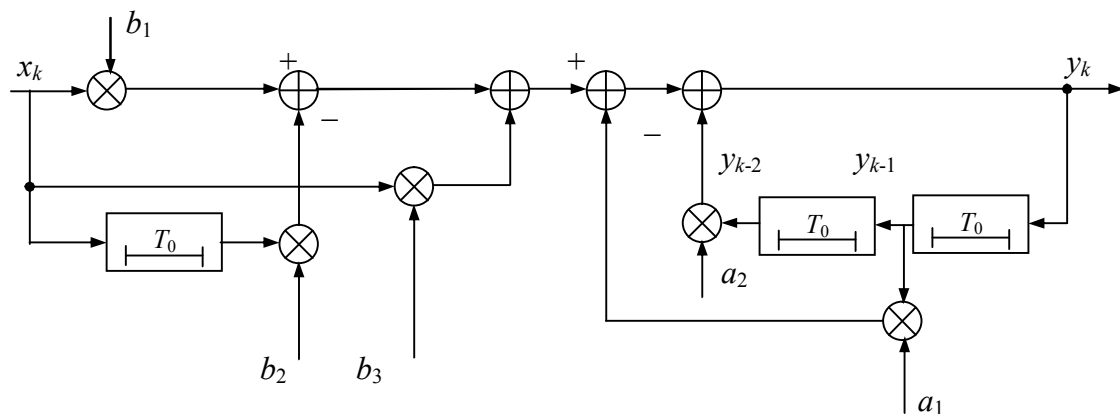


Рис.17

Найти для него рекуррентное уравнение.

Непосредственно из анализа схемы следует:

$$y_k = -a_1 y_{k-1} + a_2 y_{k-2} + (b_1 + b_3)x_k - b_2 x_{k-1}.$$

## 9. ОПИСАНИЕ ФИЛЬТРОВ В ЧАСТОТНОЙ ОБЛАСТИ. ДИСКРЕТНОЕ ПРЕОБРАЗОВАНИЕ ФУРЬЕ, Z –ПРЕОБРАЗОВАНИЕ

До сих пор рассматривалось временное описание фильтров. При проектировании цифровых фильтров важное значение имеет их описание в частотной области. Кроме того, на ЭВМ можно реализовать обработку сигналов и в частотной области. Для этого производится преобразование временной функции в частотную область, выполняется обработка в частотной области, а затем делается обратный переход во временную область. Частотные преобразования могут использоваться также для решения рекуррентных уравнений.

При переходе в частотную область для непрерывных сигналов применяются преобразование Лапласа и как частный случай его – преобразование Фурье. Для дискретных сигналов их использовать нельзя, так как сигналы скачкообразные. Применяют аналоги преобразований.

Дискретное преобразование Лапласа для функции  $f[kT_0]$  имеет вид:

$$F_D(p) = \sum_{k=0}^{\infty} f[kT_0] \cdot e^{-kpT_0},$$

где  $p = c + j\omega$  - комплексное число.

В частном случае, когда  $p = j\omega$ , имеем дискретное преобразование Фурье:

$$F_D(j\omega) = \sum_{k=0}^{\infty} f[kT_0] \cdot e^{-j\omega kT_0}.$$

Можно показать, что справедливо следующее соотношение :

$F_H(p) = \lim_{T_0 \rightarrow 0} F_D(p) = \int_0^{\infty} f[t] e^{-pt} dt$  – непрерывное преобразование Лапласа.

Обратное дискретное преобразование Лапласа выглядит следующим образом:

$$f[kT_0] = F_D^{-1}(p) = \frac{T_0}{2j\omega} \int_{c_1 - j\frac{\pi}{T_0}}^{c_1 + j\frac{\pi}{T_0}} F_D(p) e^{kpT_0} dp, \quad c_1 > c.$$

Функция  $F_D(j\omega)$  определяет частотный спектр сигнала, из которого можно выделить амплитудно-частотный и фазо-частотный спектры. Покажем, что дискретный спектр является периодическим, с периодом  $2\pi/T_0$ . Действительно, для любого целого числа  $n$  имеем:

$$F_D\left(j\left(\omega + n\frac{2\pi}{T_0}\right)\right) = \sum_{k=0}^{\infty} f_k e^{-j\left(\omega + n\frac{2\pi}{T_0}\right)kT_0} = \sum_{k=0}^{\infty} f_k e^{-j\omega kT_0} e^{-jn\frac{2\pi}{T_0}kT_0} = \sum_{k=0}^{\infty} f_k e^{-j\omega kT_0} e^{-jnk2\pi} = F_D(j\omega)$$

Следовательно, если амплитудный спектр  $|F_H(j\omega)|$  непрерывного сигнала  $f(t)$  имеет вид показанный на рис.18, то спектр  $|F_D(j\omega)|$  дискретного сигнала  $f_k$  (рис.19) – периодический.

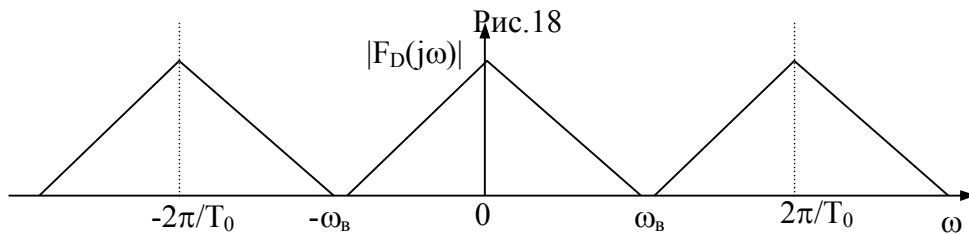
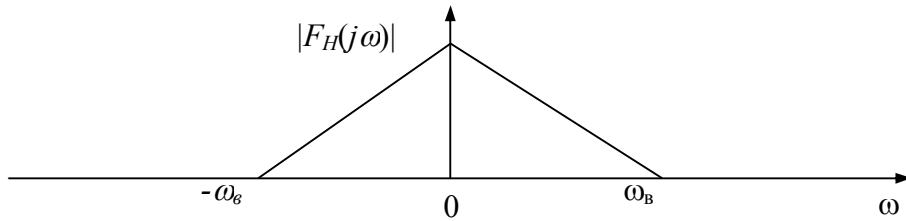


Рис.19

При увеличении  $T_0$  “лепестки” спектра сближаются и они могут сомкнуться при  $T_0 = \frac{\pi}{\omega_s} = \frac{1}{2f_s}$ . Если  $T_0 > \frac{1}{2f_s}$ , то лепестки перекроются. Это приводит к искажению сигнала. Поэтому период дискретизации должен быть равен  $T_0 \leq \frac{1}{2f_s}$ , как это следует из неравенства Котельникова.

Часто вместо дискретного преобразования Фурье используют Z– преобразование :

$$Z\{f_k\} = F(z) = \sum_{k=0}^{\infty} f_k z^{-k},$$

где  $z = e^{j\omega T_0}$ . Из функции  $F(z)$  легко получить функцию  $F_D(j\omega)$ :

$$F_D(j\omega) = F(-z = e^{j\omega T_0}).$$

На основе использования  $Z$ -преобразований разработана теория проектирования цифровых фильтров. Поэтому рассмотрим 3 основные свойства  $Z$ -преобразования.

### 1. Линейность

Пусть имеются две дискретные функции  $f_{1k}, f_{2k}$ , изображение которых  $F_1(z)$  и  $F_2(z)$ . Тогда

$$Z\{f_{1k} + f_{2k}\} = F_1(z) + F_2(z).$$

### 2. Теорема сдвига (запаздывания)

Пусть есть дискретная функция  $f_k$  изображение которой  $F(z)$ . Найдем  $Z$ -изображение функции  $f_{k-i}$ , которая запаздывает на  $i$  тактов.

$$Z\{f_{k-i}\} = \sum_{k=0}^{\infty} f_{k-i} z^{-k}.$$

Заменим переменную  $k-i=n$ , имеем

$$Z\{f_{k-i}\} = \sum_{n=-i}^{\infty} f_n z^{-n-i} = z^{-i} \left[ \sum_{n=0}^{\infty} f_n z^{-n} + \sum_{n=-i}^{-1} f_n z^{-n} \right] = z^{-i} \left[ F(z) + \sum_{n=-i}^{-1} f_n z^{-n} \right].$$

Если до момента  $k=0$  значения функции  $f$  нулевые, т.е.  $f_n = f_{(n-1)} = \dots = f_{-1} = 0$ , то получим

$$Z\{f_{k-i}\} = z^{-i} F(z).$$

### 3. Изображение свертки

Изображение свертки  $y_k = \sum_{i=0}^k x_i c_{k-i}$  есть  $Y(z) = X(z)W(z)$ , где  $X, Y$  - изображения входного и выходного сигналов,  $W(z) = Z(C_k)$  - изображение импульсной реакции фильтра (рис.20). Причем  $W(z) = Y(z)/X(z)$  - есть ничто иное как передаточная функция фильтра.

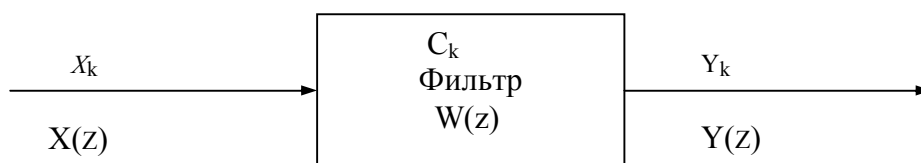


Рис.20

## **10. Вычисление реакций фильтров (Решение рекуррентных уравнений)**

Пусть имеется цифровой фильтр, который описывается рекуррентным уравнением следующего вида:

$$a_0 y_{k-n} + \dots + a_{n-1} y_{k-1} + a_n y_k = b_0 x_{k-m} + \dots + b_{m-1} x_{k-1} + b_m x_k.$$

Конечно, можно вычислить  $y_k$ , последовательно используя это уравнения. Но при больших значениях  $k$  это не эффективно. Существует свернутая форма  $y_k$  (решение уравнения). Применим к левой и правой частям уравнение  $Z$  – преобразования при нулевых начальных значениях, с применением свойства линейности и теоремы сдвига, получаем:

$$a_0 z^{-n} Y(z) + \dots + a_{n-1} z^{-1} Y(z) + a_n Y(z) = b_0 z^{-m} X(z) + \dots + b_{m-1} z^{-1} X(z) + b_m X(z),$$

отсюда 
$$Y(z) = \frac{b_0 z^{-m} + \dots + b_{m-1} z^{-1} + b_m}{a_0 z^{-n} + \dots + a_{n-1} z^{-1} + a_n} X(z).$$

Очевидно, что дробь в последнем выражении есть передаточная функция ЦФ, которую можно представить в виде отношения полиномов  $B(z)$  и  $A(z)$  с положительными показателями:

$$W(z) = \frac{B(z)}{A(z)} = \frac{b_m z^n + \dots + b_1 z^{n-m+1} + b_0 z^{n-m}}{a_n z^n + \dots + a_1 z + a_0},$$

т. к. обычно  $n \geq m$ .

Реакция фильтра  $y_k$  находится путем применения обратного  $Z^{-1}$  преобразования:

$$y_k = Z^{-1} \{ Y(z) = W(z)X(z) \}.$$

При этом пользуются различными способами:

- 1) формальное вычисление обратного преобразования;
- 2) использование справочных таблиц соответствия  $y_k$  и  $Y(z)$ ;
- 3) метод разложения.

Последний метод применим, если изображение реакции можно представить как  $Y(z) = \frac{B'(z)}{A'(z)}$ , т.е. в виде дробно-рациональной функции аргумента  $z$ , где  $B'(z)$ ,  $A'(z)$  – полиномы. Тогда, если степень полинома числителя меньше степени полинома знаменателя, то  $Y$  можно разложить на сумму простых дробей

$$Y(z) = \frac{B'(z)}{A'(z)} = \sum_{i=1}^n \frac{d_i}{z - z_i},$$

где  $z_i$  – корни полинома  $A'(z)$ , которые находятся из решения уравнения  $A'(z=z_i)=0$ ,  $d_i$  – постоянные коэффициенты. Они находятся путем решения



системы  $n$  уравнений, вытекающей из равенства  $\frac{B'(z)}{A'(z)} = \sum_{i=1}^n \frac{d_i}{z - z_i}$ , приведенного к общему знаменателю.

Изображение простой дроби  $\frac{d_i}{z - z_i}$  равно  $d_i z_i^{k-1}$ ,  $k \geq 1$ . Действительно, возьмем  $Z$ -преобразование от  $d_i z_i^{k-1}$ ,  $k \geq 1$ , имеем

$$Z\{d_i z_i^{k-1}\} = \sum_{k=1}^{\infty} d_i \cdot z_i^{k-1} z^{-k} = d_i z^{-1} \sum_{k=0}^{\infty} (z_i z^{-1})^k.$$

Так как последняя сумма есть сумма членов убывающей геометрической прогрессии равная  $\frac{1}{1 - z_i z^{-1}}$ , то исходное утверждение верно. Следовательно,

$y_k = \sum_{i=1}^n d_i \cdot z_i^{k-1}$  - реакция фильтра или решение рекуррентного уравнения.

Когда степени полиномов числителя и знаменателя  $B'(z)$  и  $A'(z)$  равны, возможны две ситуации:

1.  $B'(z) = z B_0(z)$ , тогда

$$Y(z) = z \frac{B_0(z)}{A'(z)} = z \cdot \sum_{i=1}^n \frac{d_i}{z - z_i} = \sum_{i=1}^n \frac{d_i z}{z - z_i}.$$

Отсюда на основании предыдущего соотношения и теоремы сдвига имеем

$$y_k = \sum_{i=1}^n d_i \cdot z_i^k.$$

2. Степени одинаковы, но нельзя представить полином числителя, как в первом случае. В этом случае делят полином  $B'(z)$  на полином  $A'(z)$ , выделяя значение  $y_0$ , получают  $\frac{B'(z)}{A'(z)} = y_0 + \frac{B_1(z)}{A'(z)}$ .  $\frac{B_1(z)}{A'(z)}$  разлагают на сумму простых дробей.

Реакция фильтра будет равна  $y_k = \sum_{i=1}^n d_i z_i^{k-1}$ ,  $k \geq 1$ .

Замечание: при бесконечном делении полинома  $B'(z)$  на полином  $A'(z)$  получается ряд Лорана  $\frac{B'(z)}{A'(z)} = \sum_{k=0}^{\infty} y_k z^{-k}$ , в котором  $y_k$  - реакция ЦФ на шаге  $k$ .

*Пример*

Найдем реакцию фильтра Баттерворта (пример 1 раздел 8), для которого рекуррентное уравнение имеет вид

$$y_k = (1 - \frac{T_0}{\tau}) y_{k-1} + \frac{T_0}{\tau} x_k.$$

Применим  $Z$ -преобразование к левым и правым частям уравнения, получим

$$Y(z) = (1 - \frac{T_0}{\tau}) z^{-1} Y(z) + \frac{T_0}{\tau} X(z);$$

тогда

$$Y(z) = \frac{\frac{T_0}{\tau} \cdot X(z)}{1 - (1 - \frac{T_0}{\tau})z^{-1}} = \frac{\frac{T_0}{\tau} \cdot X(z) \cdot z}{z - (1 - \frac{T_0}{\tau})} .$$

Пусть входной сигнал является дельта-импульсом  $x_k = \begin{cases} 1, k=0, \\ 0, k>0, \end{cases}$  изображение

которого  $X(z) = \sum_{k=0}^{\infty} x_k z^{-k} = x_0 z_0^{-0} + x_1 z^{-1} + x_2 z^{-2} + \dots = 1 \cdot 1 + 0 \cdot z^{-1} + \dots + 0 = 1$  .

Отсюда имеем  $Y(z) = \frac{z \cdot \frac{T_0}{\tau}}{z - (1 - \frac{T_0}{\tau})}$  .

На основании рассмотренного выше имеет место первая ситуация и реакция ЦФ будет

$$y_k = \frac{T_0}{\tau} \cdot (1 - \frac{T_0}{\tau})^k = c_k .$$

## 11. Передаточная функция фильтра

Передаточная функция определяет все свойства фильтра, также, как и импульсная реакция. На практике может возникнуть ситуация, когда необходимо экспериментально найти функцию  $W(z)$  либо функцию  $W_D(j\omega)$ . Так как (см. р.9)  $z=e^{j\omega T_0}$  , то  $W_D(j\omega) = W(z=e^{j\omega T_0})$  . Причем передаточную функцию можно представить в одном из следующих видов

$$W_D(j\omega) = |W(j\omega)|e^{j\varphi(\omega)} = A(\omega) \cos \varphi(\omega) + jA(\omega) \sin \varphi(\omega) ,$$

где  $A(\omega)$  и  $\varphi(\omega)$  - амплитудо- и фазочастотные характеристики (рис.21) .

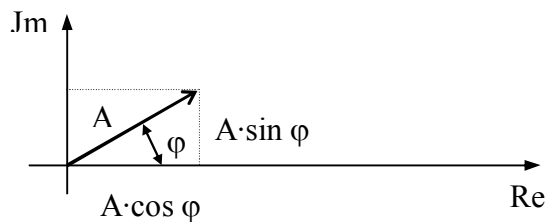


Рис.21

Задача заключается по сути дела в измерении  $A(\omega)$  и  $\varphi(\omega)$ . Для этого на вход ЦФ надо подать синусоидальный сигнал  $x$  частоты  $\omega$ . На выходе фильтра возникнет реакция  $y$  той же частоты.

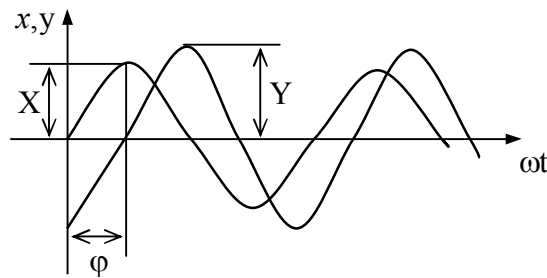


Рис.22

Далее измеряем амплитуды сигналов  $Y$ ,  $X$  и фазу  $\varphi$ . Вычисляем  $A(\omega) = \frac{Y}{X} \Big|_{\omega}$ ,  $\varphi \Big|_{\omega}$ . Задавая разные частоты  $\omega = \omega_1, \omega_2, \dots$  можно снять зависимости модуля и фазы от частоты, тем самым получить передаточную функцию.

Рассмотрим некоторые соединения цифровых фильтров и их эквивалентные передаточные функции.

1. Последовательное соединение (рис.23).

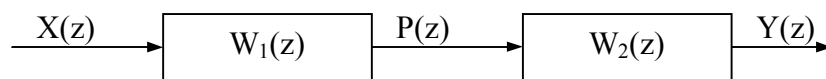


Рис.23

$$W(z) = \frac{Y(z)}{X(z)} = \frac{Y(z)}{P(z)} \cdot \frac{P(z)}{X(z)} = W_1(z) \cdot W_2(z) .$$

3. Параллельное соединение (рис.24).

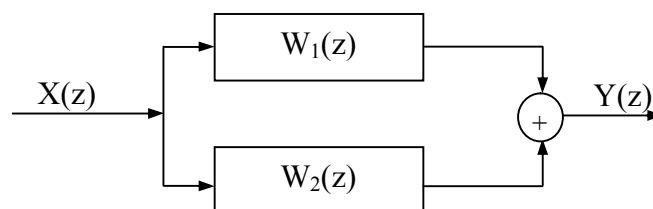


Рис.24

$$W(z) = W_1(z) + W_2(z) .$$

3. Соединение с обратной связью (рис.25).

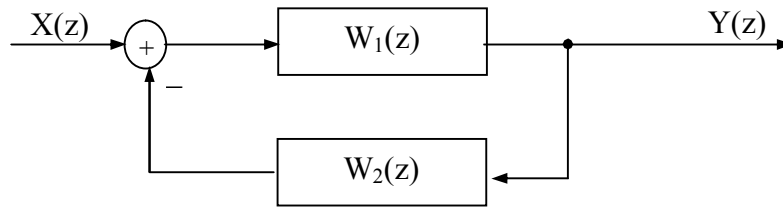


Рис.25

$$W(z) = \frac{W_1(z)}{1 \pm W_1(z) \cdot W_2(z)} .$$

Пусть фильтр описывается рекуррентным уравнением. Если перейти к Z-преобразованию этого рекуррентного уравнения, то можно найти передаточную функцию  $W(z) = \frac{Y(z)}{X(z)}$  которая является дробно-рациональной функцией.

Если поделить  $Y(z)$  на  $X(z)$ , то получится бесконечный ряд (ряд Лорана)

$$W(z) = \sum_{k=0}^{\infty} c_k z^{-k} . \text{ Изображение реакции ЦФ будет равно } Y(z) = \sum_{i=0}^{\infty} c_i z^{-i} X(z) .$$

Осуществляя обратный переход во временную область с учетом теоремы сдвига, получим

$$y_k = \sum_{i=0}^{\infty} c_i x_{k-i} .$$

Это выражение является сверткой для ЦФ, описанного в разделе 7. Так от рекурсивного фильтра можно перейти к нерекурсивному.

## 12. Устойчивость цифровых фильтров

Важнейшим при проектировании фильтра является вопрос устойчивости. Одна и та же структура фильтра при одних параметрах может быть устойчивой (фильтром), а при других - генератором. Для оценки устойчивости на вход ЦФ необходимо подать некоторый сигнал—"толчок", например дельта-импульс. На входе получим реакцию  $c_k$ . Необходимым условием устойчивости является  $c_k \rightarrow 0$  при  $k \rightarrow \infty$ . Достаточное условие устойчивости:  $\sum_{k=0}^{\infty} |c_k| < \infty$ . Как показано в разделе 10, для любой передаточной функции которая описывается как дробно-рациональная, справедливо соотношение  $c_k = \sum_{i=1}^n d_i z_i^k$ , где  $d_i$  - постоянный коэффициент,  $z_i$  - корни полинома знаменателя передаточной функции,  $n$  - общее число корней. Из анализа этого выражения следует, что для устойчивости необходимо, чтобы  $|z_i| < 1$ ,  $i=1, \dots, n$ . В общем случае  $z_i$  - это комплексные числа (рис.26).

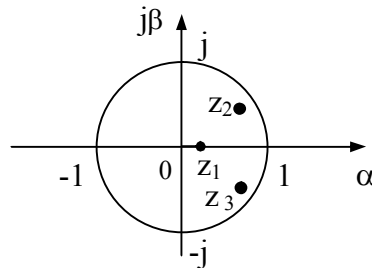


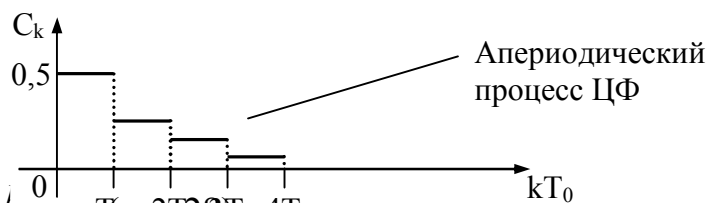
Рис.26

Для устойчивой системы корни должны лежать внутри окружности с единичным радиусом. Если хотя бы один корень окажется на границе или вне окружности, то фильтр неустойчивый.

### Пример

Исследуем на устойчивость ЦФ первого порядка, аналогичный рассмотренному в примере раздела 10, заданный уравнением  $y_k = (1 - K_0)y_{k-1} + K_0 \cdot x_k$ , где  $K_0$  - параметр фильтра имеющий передаточную функцию  $W(z) = \frac{Y(z)}{X(z)} = \frac{K_0 \cdot z}{z - (1 - K_0)}$ . Поскольку  $z_1 = 1 - K_0$  - корень знаменателя функции  $W(z)$ , то  $C_k = K_0(1 - K_0)^k$ . Условие устойчивости ЦФ:  $|z_1| = |1 - K_0| < 1$  или  $0 < K_0 < 2$ . Рассмотрим реакции  $C_k$  для различных значений  $K_0$ .

1.  $K_0 = 0,5$ ;  $C_k = (0,5)^{k+1}$ , (рис.27).



2.  $K_0 = 1$ ;  $C_k = \begin{cases} 1, & k=0 \\ 0, & k \geq 1 \end{cases}$ , (рис.28)

Рис.27

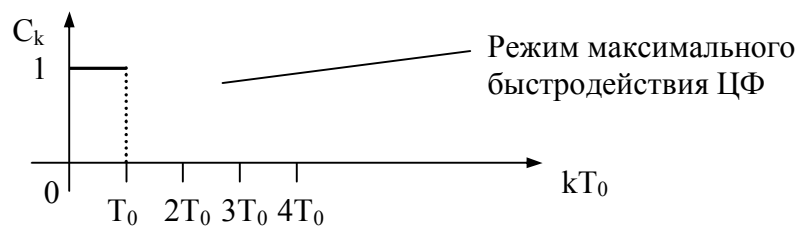


Рис.28

3.  $K_0=1,5$  ;  $C_k=1,5(-0,5)^k$  , (рис.29).

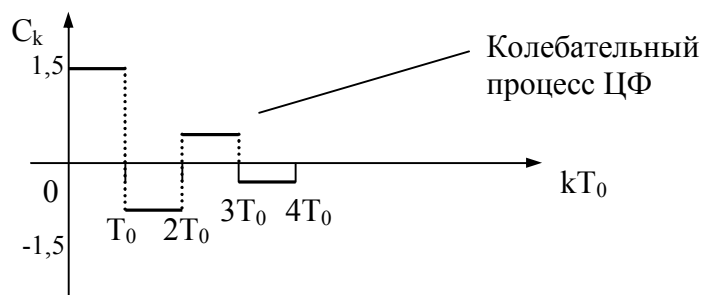


Рис.29

4.  $K_0=2$  ;  $C_k=2(-1)^k$  , (рис.30).

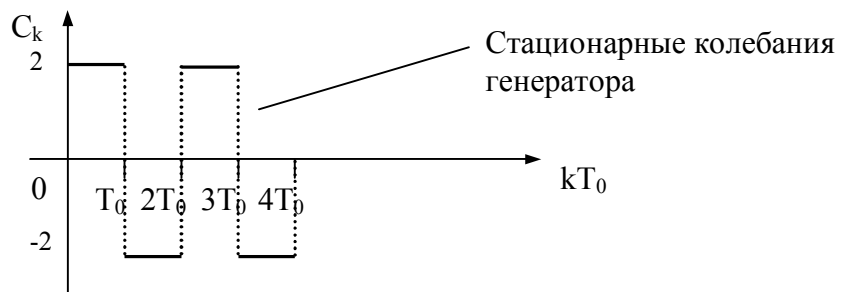


Рис.30

5.  $K_0=2,5$ ;  $C_k=2,5(-1,5)^k$  , (рис.31).

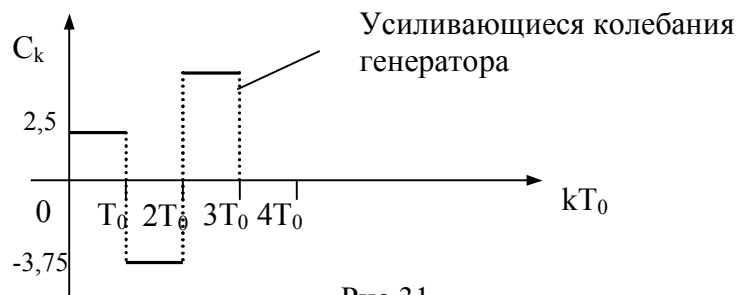


Рис.31

### 13. Фильтр Калмана

Особенность фильтра Калмана в том, что его некоторые параметры есть функции времени, т.е. меняются по определенному (оптимальному с точки зрения ошибки фильтрации) закону. Если в фильтре Калмана зафиксировать параметры, то получим обычный фильтр. Рассмотрим переход от обычного фильтра первого порядка, определенного уравнением  $y_k = (1 - K_0) \cdot y_{k-1} + K_0 x_k$  к фильтру Калмана. Уравнение может быть записано в виде  $y_k = y_{k-1} + K_0 (x_k - y_{k-1})$ . Его структурная реализация с возможностью преобразования в фильтр Калмана показана на рис.32.

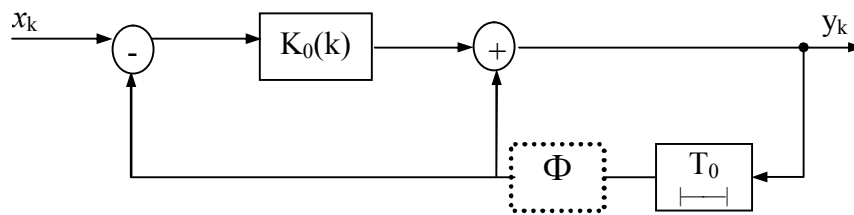


Рис.32

В фильтре Калмана параметр  $K_0$  зависит от времени  $k$  и кроме того в общем случае включается блок передачи  $\Phi = e^{-\frac{T_0}{\tau}}$ , где  $\tau$  - постоянная времени.

Если  $\tau \gg T_0$ , то  $\Phi = 1 - \frac{T_0}{\tau}$  и в пределе равен 1. Уравнение Калмана, определяющее работу фильтра записывается в следующем виде :

$$y_k = \Phi y_{k-1} + K_{0k} (x_k - \Phi y_{k-1}) .$$

Фильтр Калмана используется для выделения полезной информации на фоне помех, т.е. когда  $x_k = S_k + n_k$ , где  $S_k$  - полезный сигнал,  $n_k$  - помеха, (рис.33).

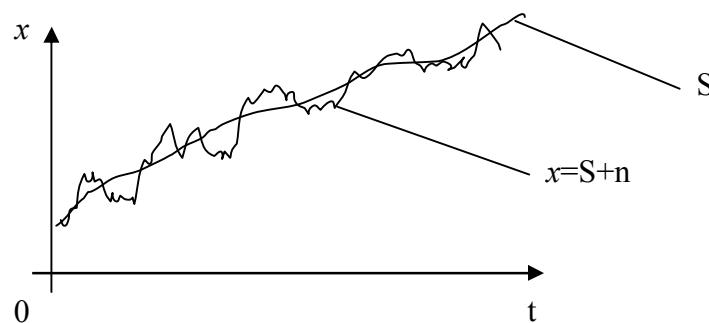


Рис.33

Сигнал отличается от помехи обычно тем, что он меняется медленнее чем помеха, поэтому его спектр узкополосен (рис.34).

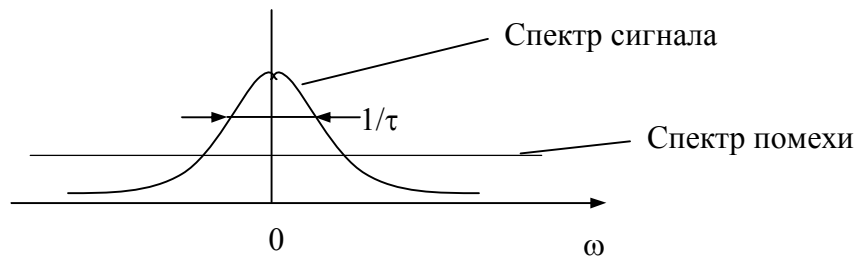


Рис.34

Задачей фильтра является оценка  $\hat{S}_k = y_k$ . Фильтр Калмана минимизирует ошибку фильтрации на любом шаге. Эта ошибка определяется дисперсией  $D_k = (S_k - \hat{S}_k)^2 = \min$ .

Как управлять коэффициентом  $K_{0k}$ ? Начало фильтрации относится к моменту времени  $t=0$ . Поэтому, когда  $k$  мало, необходимо доверять входной информации надо полностью. Значит  $K_{0k}$  должно быть большим. А затем, когда оценка уточняется,  $K_{0k}$  уменьшается (рис.35).

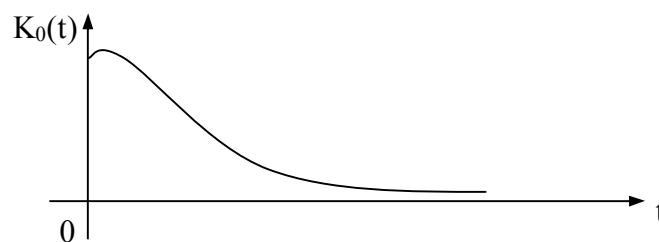


Рис.35

$K_{0k}$  зависит от  $D_k$ , которое вычисляется на основании уравнения Риккати, с уменьшением  $K_{0k}$  сужается полоса пропускания фильтра, тем самым отфильтровывается широкополосная помеха. Величина  $\Phi y_{k-1} = \Phi \hat{S}_{k-1}$  представляет собой предсказанное значение или оценку экстраполяции сигнала, разность  $x_k - \Phi y_{k-1}$  - обновляющая входная информация фильтра.

На основе одномерного фильтра Калмана (рис.32) можно построить многомерный фильтр. Для этого рекуррентное уравнение порядка  $n$ , описывающее работу фильтра преобразуют в одномерную систему  $n$  - уравнений первого порядка. Покажем возможность перехода от рекуррентного порядка 2 к рекуррентному порядку 1. Исходное уравнение равно

$$a_0 y_{k-2} + a_1 y_{k-1} + a_2 y_k = b_0 x_k.$$



Обозначим 
$$\begin{cases} y_{1k} = y_k, \\ y_{2k} = y_k - y_{1k-1}. \end{cases}$$

Тогда 
$$\begin{cases} y_{1k-1} = y_{k-1} = y_{1k} - y_{2k}, \\ y_{1k-2} = y_{k-2} = y_{1k-1} - y_{2k-1}. \end{cases}$$

Используя эти соотношения и исходное уравнение, получим:

$$\begin{cases} y_{1k} = -\frac{a_0 + a_1}{a_2} y_{1k-1} + \frac{a_0}{a_2} y_{2k-1} + \frac{b_0}{a_2} x_k, \\ y_{2k} = -\frac{a_0 + a_1 + a_2}{a_2} y_{1k-1} + \frac{a_0}{a_2} y_{2k-1} + \frac{b_0}{a_2} x_k. \end{cases}$$

Эту систему уравнений можно записать в матричном виде

$$\underbrace{\begin{bmatrix} y_{1k} \\ y_{2k} \end{bmatrix}}_{Y_k} = \underbrace{\begin{bmatrix} -\frac{a_0 + a_1}{a_2} & \frac{a_0}{a_2} \\ -\frac{a_0 + a_1 + a_2}{a_2} & \frac{a_0}{a_2} \end{bmatrix}}_A \times \underbrace{\begin{bmatrix} y_{1k-1} \\ y_{2k-1} \end{bmatrix}}_{Y_{k-1}} + \underbrace{\begin{bmatrix} \frac{b_0}{a_2} & 0 \\ \frac{b_0}{a_2} & 0 \end{bmatrix}}_B \times \underbrace{\begin{bmatrix} x_k \\ 0 \end{bmatrix}}_{X_k}, \text{ т.е. } Y_k = A \cdot Y_{k-1} + B X_k.$$

Параметры многомерного фильтра Калмана являются матричными.

## 14. Проектирование цифровых фильтров

Распространенный способ проектирования заключается в том, что прототипом берется готовый аналоговый фильтр, т.е. фильтр с известными параметрами и известной передаточной функцией. Остается лишь найти его цифровой эквивалент. Один из методов нахождения цифрового эквивалента заключается в следующем. Если передаточная функция аналогового фильтра есть дробно-рациональная функция, то она раскладывается на сумму простых дробей. Каждой простой дроби соответствует известный цифровой эквивалент. Далее все эквиваленты суммируются и находится общая передаточная функция. От передаточной функции осуществляется переход к рекуррентному уравнению, реализующему фильтр.

*Пример*

Пусть задан фильтр Баттерворта второго порядка, имеющий передаточную функцию  $W_n(p) = \frac{1}{(1+p\tau_1)(1+p\tau_2)} = \frac{1}{1+p(\tau_1+\tau_2)+p^2\tau_1\tau_2}$ .

Он представляет собой последовательное соединение двух инерционных звеньев (рис.36) с постоянными времени  $\tau_1=R_1C_1$  и  $\tau_2=R_2C_2$ .

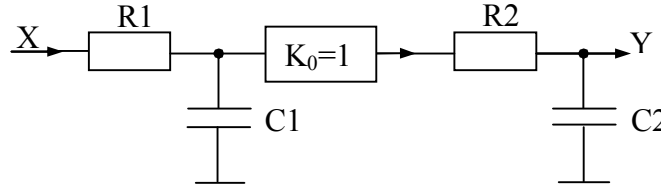


Рис.36

Для предотвращения влияния второго звена на первое используется широкополосный усилитель с коэффициентом передачи  $K_0=1$ . Квадрат модуля передаточной функции фильтра определяется зависимостью

$$A^2(\omega) = \frac{1}{1 + \omega^2(\tau_1^2 + \tau_2^2) + \omega^4\tau_1^2\tau_2^2}.$$

Качественный вид амплитудно-частотной характеристики показан на рис.37.

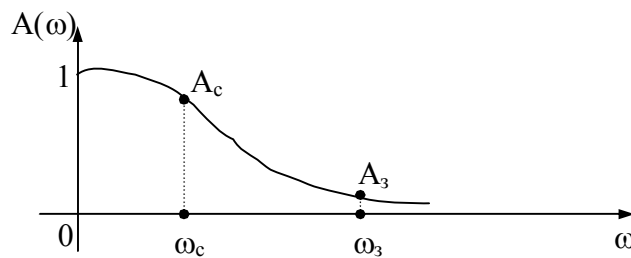


Рис.37

Здесь  $\omega_c$  – частота среза,  $\omega_3$  – частота задержания. Кроме  $\omega_c$  и  $\omega_3$  задается неравномерность частотных характеристик в полосе пропускания  $0 \leq \omega < \omega_c$ :  $1 \geq A(\omega) \geq A_c$  и в полосе задержания  $\omega_3 \leq \omega < \infty$ :  $0 \leq A(\omega) \leq A_3$ .

Установим эти параметры следующими:

$$\begin{aligned} \omega_c &= 10^3 \text{ c}^{-1}; & A_c^2 &= A^2(\omega = \omega_c) = \frac{1}{2}; \\ \omega_3 &= 10^4 \text{ c}^{-1}; & A_3^2 &= A^2(\omega = \omega_3) = \frac{1}{1000}. \end{aligned}$$

Подставляем параметры в выражение для квадрата модуля передаточной функции, решаем систему двух уравнений относительно величин  $\tau_1$ ,  $\tau_2$ , получим:

$$\tau_1 = 0,36 \cdot 10^{-3} \text{ с , или } 0,89 \cdot 10^{-3} \text{ с ;}$$

$$\tau_2 = 0,89 \cdot 10^{-3} \text{ с , или } 0,36 \cdot 10^{-3} \text{ с .}$$

Разложим функцию  $W_n(p)$  на сумму простых дробей , имеем:

$$W_n(p) = \frac{1}{\tau_2 - \tau_1} \left( \frac{1}{p - p_2} - \frac{1}{p - p_1} \right) ,$$

где  $p_2 = -\frac{1}{\tau_2}$  ,  $p_1 = -\frac{1}{\tau_1}$  . Дроби  $\frac{1}{p - p_i}$  в  $p$ -области соответствует дробь  $\frac{T_0}{z - z_i}$  в  $z$ -области , где  $z_i = e^{p_i T_0}$  . Действительно, при  $T_0 \rightarrow 0$   $z = e^{p T_0} \cong 1 + p T_0$  ,  $z_i \cong 1 - p_i T_0$  и  $\frac{T_0}{z - z_i} = \frac{1}{p - p_i}$  . Отсюда передаточная функция  $W(z)$  будет равна

$$W(z) = \frac{Y(z)}{X(z)} = \frac{b z^{-2}}{1 - a_1 z^{-1} + a_2 z^{-2}} ,$$

$$\text{где } b = \frac{T_0^2}{\tau_1 \tau_2} , \quad a_1 = 2 - T_0 \frac{\tau_1 + \tau_2}{\tau_2 \tau_1} , \quad a_2 = 1 - 2T_0 \frac{\tau_1 + \tau_2}{\tau_2 \tau_1} + \frac{T_0^2}{\tau_2 \tau_1} .$$

Получаем, что изображение выходного сигнала  $Y(z)$  ЦФ имеет вид

$$Y(z) = a_1 z^{-1} Y(z) + a_2 z^{-2} Y(z) + b z^{-2} X(z) .$$

Переходим во временную область, используя теорему сдвига, получим рекуррентную процедуру, которая описывает фильтр:

$$y_k = a_1 y_{k-1} - a_2 y_{k-2} + b x_{k-2} .$$

Найдем период дискретизации  $T_0$ . На основании неравенства Котельникова  $T_0 \leq \frac{\pi}{\omega_3}$  . Величина  $\omega_3$  известна из задания, она равна  $\omega_3 = 10^4 \text{ с}^{-1}$ . Возьмем

$$\omega_0 = \frac{2\pi}{T_0} \quad \text{на порядок больше , т.е. } \omega_0 = 10^5 \text{ с}^{-1} . \quad \text{Тогда}$$

$$T_0 = \frac{2\pi}{\omega_0} = \frac{2\pi}{10^5} = 0,63 \cdot 10^{-4} = 63 \text{ мкс} , \text{ и один цикл вычислений по рекуррентной процедуре на любом шаге } k \text{ не должен превышать } 63 \text{ мкс} .$$

## 15. Двумерная цифровая фильтрация

До сих пор рассматривали одномерную фильтрацию - обработку одномерных сигналов. На практике же существуют и используются двумерные сигналы. Например, фотоизображение является двумерным сигналом, заданным по двум координатам.

Вообще, двумерный сигнал можно преобразовать в одномерный сигнал путем разложения по строкам (как в телевидении) и далее обрабатывать одномерные сигналы. Такие задачи решаются, но иногда такого преобразования бывает недостаточно, для решения задач, например, восстановления изображений. В этом случае одномерная обработка не дает нужного эффекта, так как одновременно необходимо обрабатывать не только элементы слева и справа относительно текущего элемента, но и верхние и нижние элементы. Двумерная фильтрация здесь наиболее эффективна.

Двумерная фильтрация имеет аналогии с одномерной, но обладает и своей спецификой - она может производиться как в реальном времени, так и в относительном. Двумерный цифровой фильтр может иметь как рекурсивную, так и нерекурсивную структуру. Нерекурсивный фильтр реализуется в виде свертки, которая в данном случае имеет вид:

$$y[n, m] = \sum_{k=-\infty}^{+\infty} \sum_{l=-\infty}^{+\infty} x[k, l] \cdot c[n-k, m-l] ,$$

где  $k, l, m, n$  - целые числа,  $x[n, m]$  - отсчеты входного двумерного сигнала по двум координатам (цвет элемента изображения),  $y[n, m]$  - отсчеты выходного сигнала по тем же координатам,  $c[n, m]$  - импульсная реакция двумерного фильтра. Эта реакция получается, если на вход фильтра подать импульс единичной амплитуды длительностью 1 элемент по одной и по другой координатам.

Если в одномерных фильтрах время  $t$  отсчитывается в пределах от 0 до  $\infty$ , то в двумерных переменные  $k$  и  $l$  - пространственные и меняются от  $-\infty$  до  $+\infty$ . В конкретном случае количество слагаемых суммы ограничено. Поясним формулу моделью на рис.38.

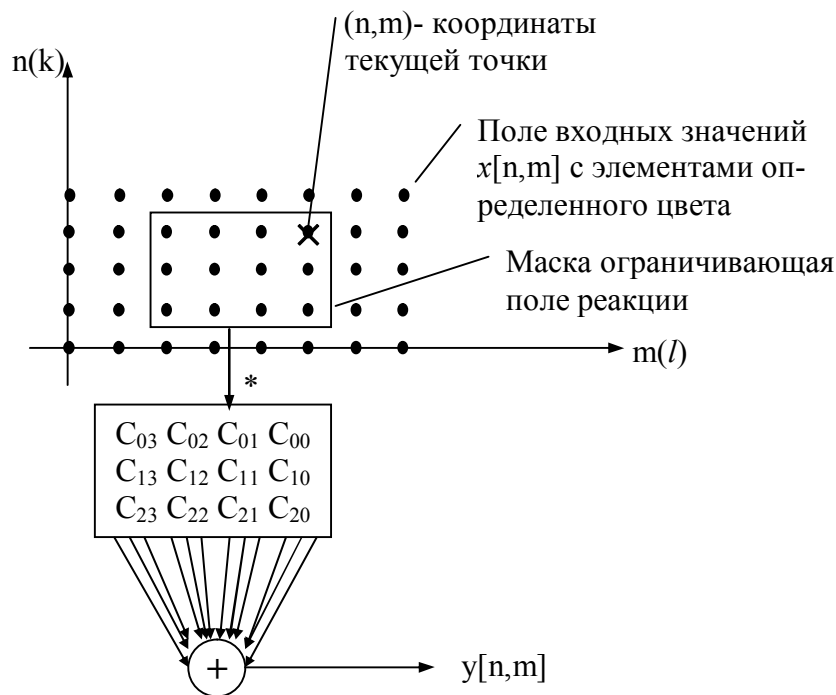


Рис.38

Для данного примера  $y[n,m] = \sum_{k=0}^2 \sum_{l=0}^3 x[n-k, m-l] \cdot c[k,l]$ .

Также как и для одномерных фильтров, для анализа двумерных фильтров используется Z- преобразование :

$$X[z_1, z_2] = Z\{x[n, m]\} = \sum_{n=-\infty}^{+\infty} \sum_{m=-\infty}^{+\infty} x[n, m] z_1^{-n} z_2^{-m}, \quad z_1 = e^{pT_{01}}, z_2 = e^{pT_{02}},$$

где  $T_{01}, T_{02}$  - периоды дискретизации по координатам  $n$  и  $m$ . Обозначим  $Y(z_1, z_2) = Z\{Y[n, m]\}$ ,  $C(z_1, z_2) = Z\{C[n, m]\}$ . Применяя Z-преобразование к свертке, получим:

$$Y(z_1, z_2) = X(z_1, z_2) \cdot C(z_1, z_2).$$

Отсюда  $C(z_1, z_2) = \frac{Y(z_1, z_2)}{X(z_1, z_2)} = W(z_1, z_2)$  - передаточная функция двумерного фильтра. Она используется для анализа и проектирования- по аналогии с одномерными фильтрами. Но анализ и проектирование происходит сложнее в силу следующих причин :

- 1) сложнее разложить  $W$  на простые дроби, так как труднее найти корни полинома знаменателя, если  $W$  - дробно-рациональная функция;

2)  $z_1, z_2$  - комплексные переменные, и  $W$  - это функция четырех действительных переменных.

Двумерный фильтр может быть реализован и как рекуррентный. Уравнение, которым он описывается, имеет вид:

$$y[n, m] = - \sum_{k=1}^{K_1} \sum_{l=1}^{L_1} a_{kl} y[n-k, m-l] + \sum_{k=0}^{K_2} \sum_{l=0}^{L_2} b_{kl} x[n-k, m-l] .$$

Структура фильтра поясняется моделью на рис.39.

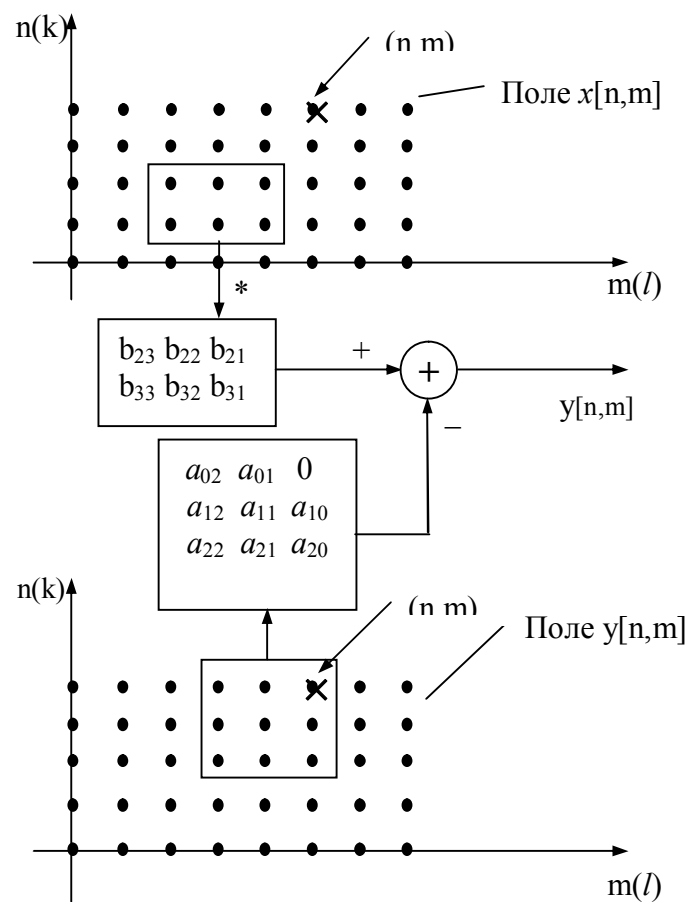


Рис.39

Фильтр производит вычисления по рекуррентной формуле

$$y[n, m] = - \sum_{k=0}^2 \sum_{l=0}^2 a_{kl} y[n-k, m-l] + \sum_{k=2}^3 \sum_{l=1}^3 b_{kl} x[n-k, m-l] .$$

## 16. Спектральный анализ сигналов

Под спектральным анализом сигналов, в общем случае, понимается обработка сигналов в частотной области. Для этого временной сигнал преобразуют в частотную область.

Большое количество задач, решаемых в частотной области, связано с оценкой формы спектра сигнала. Так подобная задача решается устройством ввода звуковой информации в ЭВМ, с помощью которого распознается человеческая речь. Распознавание речи заключается в исследовании спектральных составляющих звукового сигнала. Человеческая речь состоит из фонем - звуков. Каждая фонема имеет определенную окраску или форму спектра. Поэтому, чтобы распознать речь, нужно в каждый момент времени оценить форму ее спектра, определить какая это фонема и сопоставить ей букву.

Обработка сигналов в частотной области, в общем случае, происходит по схеме, показанной на рис.40.

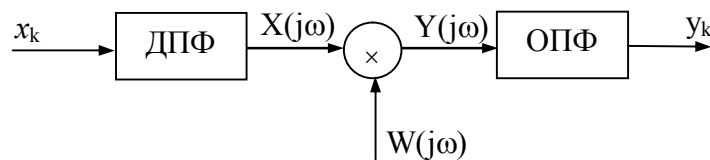


Рис.40

Здесь ДПФ – дискретное преобразование Фурье, ОПФ - обратное преобразование Фурье,  $x_k$  – входная последовательность временного сигнала,  $y_k$  – выходная последовательность,  $X(j\omega)$  – спектр входного сигнала,  $Y(j\omega)$  – спектр выходного сигнала,  $W(j\omega)$  – передаточная функция фильтра.

ДПФ для сигнала  $x$  выглядит следующим образом (раздел 9):

$$X(j\omega) = \sum_{k=0}^{\infty} x_k e^{-j\omega k T_0}.$$

Предполагается, что сигнал  $x(t)$  длится бесконечно и имеет неограниченное количество временных отсчетов. Вследствие этого  $X(j\omega)$  является непрерывной функцией от аргумента  $\omega$ . Практически же количество временных отсчетов сигнала ограничено числом  $N$ , т.е. его длительность равна  $NT_0$  (рис.41).

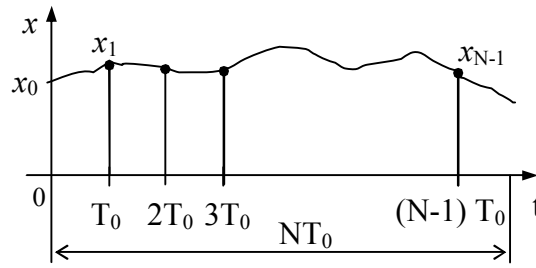


Рис.41

Тогда ДПФ запишется в виде  $X(j\omega) = \sum_{k=0}^{N-1} x_k e^{-j\omega k T_0}$ . При этом частота  $\omega$  не может быть любой. Она равна  $\omega = n\Omega$ , где  $n=0,1,2,\dots,N-1$ ;  $\Omega = 2\pi/NT_0$  - основная частота. Учитывая эти замечания, имеем  $X(jn\Omega) = \sum_{k=0}^{N-1} x_k e^{-jn\Omega k T_0} = X_n$ .  $X_n$  - в общем случае комплексная величина. Спектр содержит частоты, кратные основной частоте. Это указывает на то, что спектр из непрерывного (рис.19) превращается в линейчатый (рис.42).

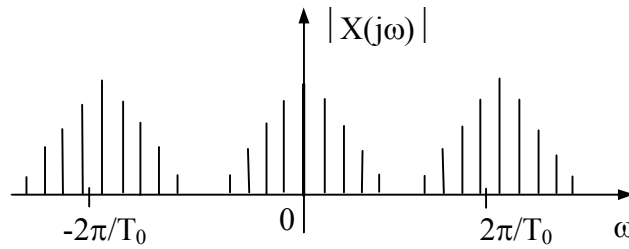


Рис.42

Покажем, что спектр  $X_n$  имеет период  $N$  по переменной  $n$ . Для любого целого  $m$  имеем

$$X_{n+mN} = \sum_{k=0}^{N-1} x_k e^{-j(n+mN)\Omega k T_0} = \sum_{k=0}^{N-1} x_k e^{-jn\Omega k T_0} e^{-jmN\Omega k T_0} = \sum_{k=0}^{N-1} x_k e^{-jn\Omega k T_0} e^{-j2\pi m} = X_n.$$

Таким образом, число спектральных коэффициентов точно такое же, как число временных отсчетов сигнала и равно  $N$ .

ОПФ для сигнала с неограниченной длиной выполняется с помощью выражения (раздел 9)

$$x_k = \frac{T_0}{2\pi} \int_{-\frac{\pi}{T_0}}^{\frac{\pi}{T_0}} X(j\omega) e^{j\omega k T_0} d\omega.$$



Из этого выражения получается ОПФ сигнала, содержащего N отсчетов (рис.41), если положить  $d\omega=\Omega$  и учесть, что спектр такого сигнала линейчатый и периодический

$$x_k = \frac{1}{N} \sum_{n=-\frac{N-1}{2}}^{\frac{N-1}{2}} X_n e^{-jn\Omega k T_0} .$$

Подобное ОПФ называется рядом Фурье, его можно представить также в следующем виде

$$x_k = \frac{1}{N} \sum_{k=0}^{N-1} X_n e^{-jn\Omega k T_0} .$$

Рассмотрим детальнее ДПФ. Так как справедливо равенство  $e^{-jn\Omega k T_0} = \cos n\Omega k T_0 - j \sin n\Omega k T_0$ , то спектральные коэффициенты  $X_n$  разлагаются на два слагаемых

$$X_n = \underbrace{\sum_{k=0}^{N-1} x_k \cos n\Omega k T_0}_{X_{cn}} - j \underbrace{\sum_{k=0}^{N-1} x_k \sin n\Omega k T_0}_{X_{sn}} = X_{cn} - jX_{sn} ,$$

содержащих косинусную  $X_{cn}$  и синусную  $X_{sn}$  составляющие. Иной записью для  $X_n$  является форма  $X_n = A_n e^{j\varphi_n}$ , где  $A_n = \sqrt{X_{cn}^2 + X_{sn}^2}$  - с точностью до коэффициента  $\frac{2}{N}$  представляет собой амплитуду спектральной составляющей,  $\varphi_n = \arctg\left(\frac{X_{sn}}{X_{cn}}\right)$  - фаза спектральной составляющей. Спектральный анализ для спектральной составляющей с номером n заключается в нахождении пары чисел  $X_{cn}$ ,  $X_{sn}$ , либо пары чисел  $A_n$ ,  $\varphi_n$ .

На рис.43 показана структурная схема анализатора спектра, на которой более подробно раскрыты синусные каналы формирования спектральных составляющих. Обработку сигнала можно производить в темпе его поступления.

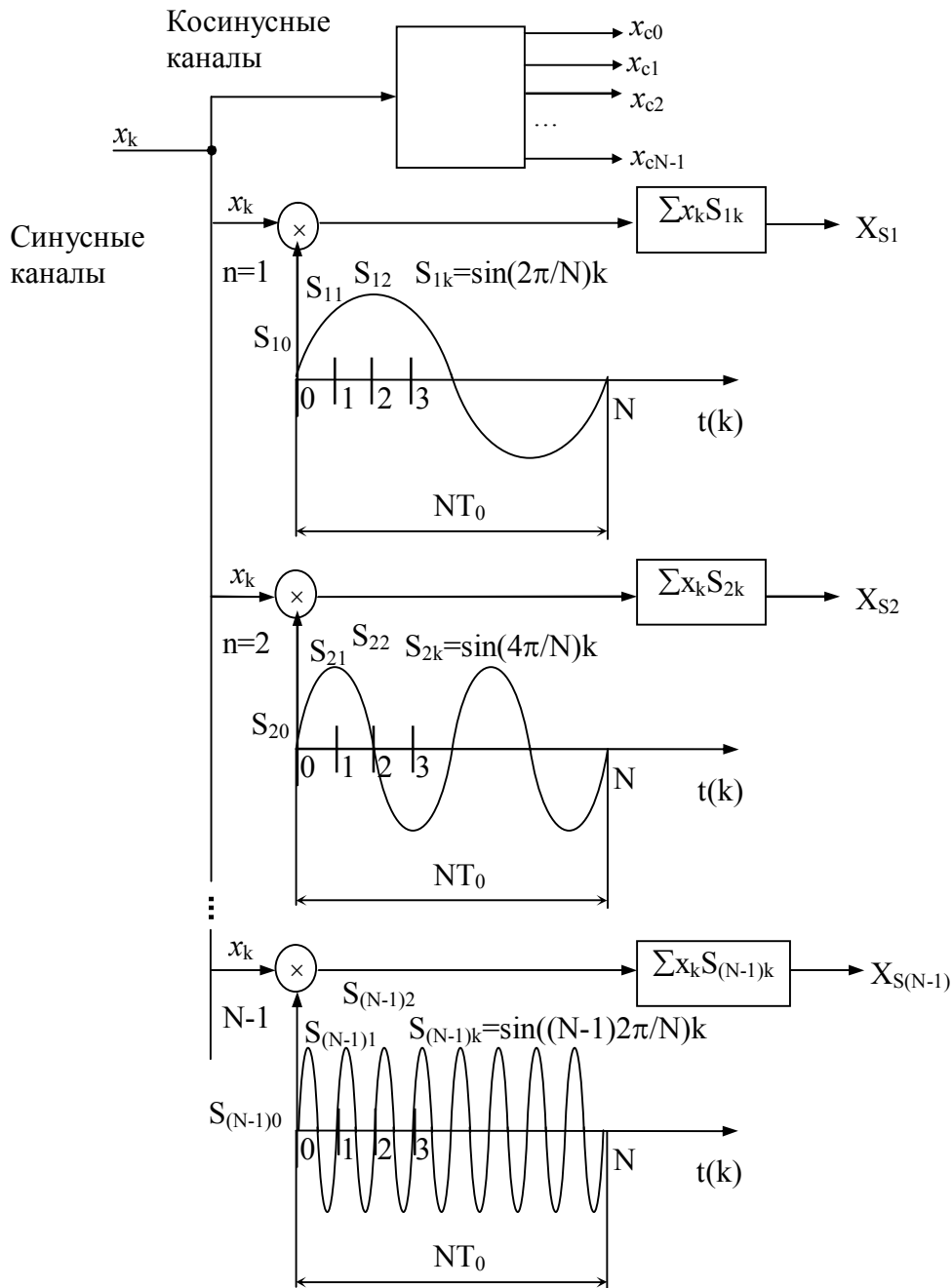


Рис.43

## 17. Быстрое преобразование Фурье

Практической реализации ДПФ, при которой рассчитывается  $N$  спектральных коэффициентов и обычно  $N \gg 1$ , препятствует необходимость выполнения большого количества операций ( $N^2$  - умножений и  $N^2$  - сложений над комплексными числами). Алгоритмы, позволяющие уменьшить количество операций, называются быстрым преобразованием Фурье (БПФ).

Рассмотрим алгоритм Тьюки - Кули. Он заключается в том, что временная последовательность делится на 2 части, с количеством отсчетов  $N/2$  в каждой. Для этих половин применяют преобразование Фурье и находят спектраль-

ные коэффициенты. Количество операций умножения или сложения оказывается равным  $2 \cdot (N/2)^2 = N^2/2$ . Если каждую из половин временных последовательностей еще раз поделить пополам и вычислить ДПФ, то количество операций будет  $4 \cdot (N/4)^2 = N^2/4$ , и т.д. Так можно дойти до такого состояния, когда в каждой части временной выборки останутся по 2 отсчета. Это соответствует количеству шагов  $\nu = \log_2 N$ . Суммарное количество операций сложения или умножения будет равно  $\nu \cdot N$ .

Алгоритм Тьюки - Кули дает выигрыш в количестве операций в 100 раз при  $N=1000$ .

Рассмотрим математическую сторону алгоритма Тьюки - Кули. Исходная последовательность  $x_k$  разбивается на четные  $x_{2k}$  и нечетные  $x_{2k+1}$  составляющие,  $k = 0, 1, 2, \dots, \frac{N}{2} - 1$ . Вычислим коэффициенты спектра  $X_n$ , при  $n = 0, 1, 2, \dots, N - 1$

$$X_n = \sum_{k=0}^{N-1} x_k e^{-jkn\Omega T_0} = \sum_{k=0}^{\frac{N}{2}-1} x_{2k} e^{-j2kn\Omega T_0} + \sum_{k=0}^{\frac{N}{2}-1} x_{2k+1} e^{-j(2k+1)n\Omega T_0} = X_n^{10} + e^{-jn\Omega T_0} X_n^{11},$$

где  $n = 0, 1, 2, \dots, \frac{N}{2} - 1$ ,  $X_n^{10} = \sum_{k=0}^{\frac{N}{2}-1} x_{2k} e^{-j2kn\Omega T_0}$ ,  $X_n^{11} = \sum_{k=0}^{\frac{N}{2}-1} x_{2k+1} e^{-j2kn\Omega T_0}$ .

Верхние индексы коэффициентов  $X_n$ : 1) указывают номер шага до окончания процедуры вычислений (в данном случае равен 1); 2) указывают на номер группы (в данном случае 0 или 1). Для вычисления  $X_n$  при  $n = \frac{N}{2}, \dots, N - 1$ , необходимо учесть, что  $X_n^{10}, X_n^{11}$  имеют период  $N/2$ , т.е. нужно доопределить  $X_n$  для  $n = N/2, \dots, N - 1$ :

$$\begin{aligned} X_n &= X_{n+\frac{N}{2}}^{10} + e^{-j(n+\frac{N}{2})\Omega T_0} \cdot X_{n+\frac{N}{2}}^{11} = X_n^{10} + e^{-j(n+\frac{N}{2})\Omega T_0} \cdot X_n^{11} = X_n^{10} + e^{-j\frac{N}{2}\Omega T_0} \cdot e^{-jn\Omega T_0} \cdot X_n^{11} = \\ &= X_n^{10} - e^{-jn\Omega T_0} \cdot X_n^{11}. \end{aligned}$$

Следовательно, получаем

$$\left. \begin{aligned} X_n &= X_n^{10} + e^{-jn\Omega T_0} \cdot X_n^{11} \\ X_{n+\frac{N}{2}} &= X_n^{10} - e^{-jn\Omega T_0} \cdot X_n^{11} \end{aligned} \right\} n = 0, 1, \dots, \frac{N}{2} - 1.$$

Запись последних формул упрощается, если обозначить  $e^{j\Omega T_0} = z$ , имеем

$$\left. \begin{aligned} X_n &= X_n^{10} + z^{-n} X_n^{11} \\ X_{n+\frac{N}{2}} &= X_n^{10} - z^{-n} X_n^{11} \end{aligned} \right\}, n = 0, 1, \dots, \frac{N}{2} - 1.$$

Коэффициенты  $X_n^{10}$ ,  $X_n^{11}$  могут быть вычислены через коэффициенты  $X_n^{20}$ ,  $X_n^{21}$  и  $X_n^{22}$ ,  $X_n^{23}$  следующим образом

$$\left. \begin{aligned} X_n^{10} &= X_n^{20} + z^{-2n} X_n^{21} \\ X_{n+\frac{N}{4}}^{10} &= X_n^{20} - z^{-2n} X_n^{21} \end{aligned} \right\}, n = 0, 1, \dots, \frac{N}{4} - 1;$$

$$\left. \begin{aligned} X_n^{11} &= X_n^{22} + z^{-2n} X_n^{23} \\ X_{n+\frac{N}{4}}^{11} &= X_n^{22} - z^{-2n} X_n^{23} \end{aligned} \right\}, n = 0, 1, \dots, \frac{N}{4} - 1.$$

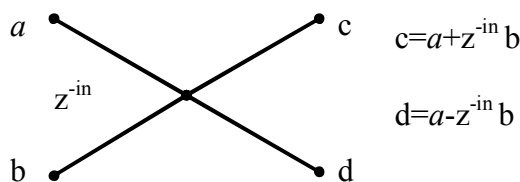
На шаге  $v$  процедура обработки предполагает вычисления  $2^{v-1}$  пар коэффициентов. Так, для произвольной пары  $l$  она будет такой

$$\left. \begin{aligned} X_n^{ve} &= X_0^{v+1q} + z^{-vn} X_0^{v+1n} \\ X_{n+1}^{ve} &= X_0^{v+1q} - z^{-vn} X_0^{v+1n} \end{aligned} \right\}, n = 0.$$

Поскольку  $X_0^{v+1q} = x^q$ ,  $X_0^{v+1n} = x^n$ , где  $x^q$ ,  $x^n$  - четное и нечетное значения временной последовательности  $x_k$ , то последняя пара преобразований раскрывается в виде

$$\left. \begin{aligned} X_0^{ve} &= x^q + x^n \\ X_1^{ve} &= x^q - x^n \end{aligned} \right\}.$$

Перед началом процедуры вычислений известными бывают лишь значения  $x_k$ . Поэтому числовые вычисления  $X_n$  производят в порядке, обратном рассмотренному. Для наглядности вычислений используют граф “бабочка” (рис.44), выходные значения которого  $c$ ,  $d$  вычисляются из входных  $a$ ,  $b$ .



*Пример*

Рис.44

Быстрое преобразование Фурье сигнала  $x$  при  $N=8$ . Реализация БПФ поясняется на рис.45.

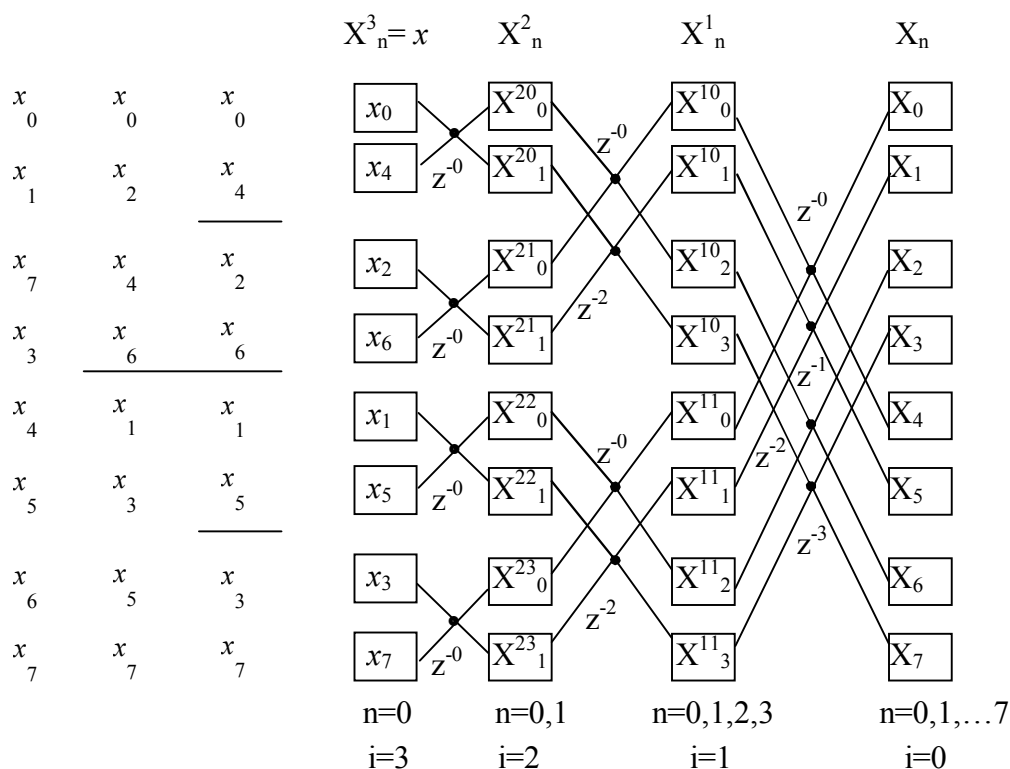


Рис.45

3 столбца в левой части (рис.45) поясняют процесс сортировки последовательности сигнала  $x_k$  на четные и нечетные значения. Сортировку можно производить и в реальном времени. Тогда окончательное место отсчета  $x_k$  определяется путем зеркального отражения индекса  $k$ , записанного в двоичном коде. После сортировки происходят вычисления с применением процедуры “бабочка”. Окончательные значения коэффициентов Фурье образуются через 3 шага последовательности вычислений.

БПФ можно реализовать, когда получены все временные отсчеты сигнала. Поэтому коэффициенты Фурье формируются с задержкой.

## 18. МОДЕЛИРОВАНИЕ ДЕТЕРМИНИРОВАННЫХ СИГНАЛОВ

Моделирование детерминированных сигналов осуществляется по их функциональным описаниям. Если временная функция сигнала  $y(t)$ ,  $t \geq 0$ , то цифровое устройство формирования вырабатывает дискретный сигнал  $y[kT_0]$ ,  $k=0,1,2,\dots$ ,  $T_0$  – период дискретизации. Период дискретизации  $T_0$  должен выбираться в пределах  $t_B < T_0 < 1/2 F_B$ , где  $t_B$  – быстродействие устройства формирования,  $1/2 F_B$  – значение, при котором не происходит потери информации, опре-

деляемое неравенством Котельникова,  $F_B$  – наибольшее значение частоты в спектре сигнала.

При формировании детерминированных сигналов типа степенных, тригонометрических и экспоненциальных функций могут использоваться стандартные подпрограммы формирования таких функций. Тогда моделирование подобных сигналов заключается в обращении к подпрограммам в цикле через моменты времени  $T_0$  и вывод функции  $y[kT_0]$  на выход формирователя.

При сложных детерминированных сигналах описание функций  $y(t)$  может задаваться таблицей.

Эффективно использование рекуррентных алгоритмов, обеспечивающих экономные вычисления, которые целесообразно применять в специализированных процессорах.

Рассмотрим несколько примеров использования рекуррентных алгоритмов.

### Пример 1

Сформировать линейный сигнал  $y(t) = at + b$ . Дискретный сигнал представим как

$$y[kT_0] = akT_0 + b = a[(k-1)+1]T_0 + b = y[(k-1)T_0] + aT_0, \quad y[0] = b.$$

Схема формирования цифрового сигнала показана на рис.46.

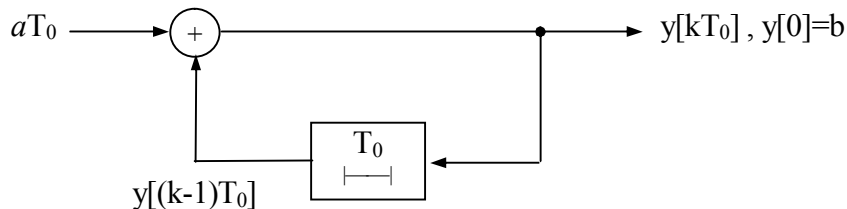


Рис.46

### Пример 2

Сформировать экспоненциальный сигнал  $y(t) = e^{-at}$ . Имеем

$$y[kT_0] = e^{-akT_0} = e^{-a[(k-1)+1]T_0} = e^{-a(k-1)T_0} \cdot e^{-aT_0} = cy[(k-1)T_0], \quad y[0] = 1, \quad c = e^{-aT_0}.$$

Схема формирования подобного дискретного сигнала приведена на рис.47.

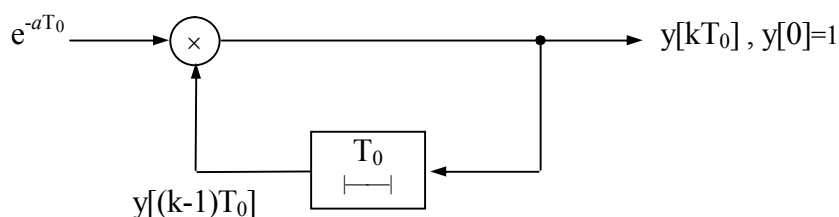


Рис.47

**Пример 3**

Сформировать синусоидальный  $\sin \omega t$  сигнал.

Сложим левые и правые части тригонометрических соотношений

$$y[kT_0] = \sin[\omega kT_0] = \sin[\omega(k-1)T_0 + \omega T_0] = \sin[\omega(k-1)T_0] \cos \omega T_0 + \cos[\omega(k-1)T_0] \sin \omega T_0 ,$$

$$y[(k-2)T_0] = \sin[\omega(k-2)T_0] = \sin[\omega(k-1)T_0 - \omega T_0] = \sin[\omega(k-1)T_0] \cos \omega T_0 - \cos[\omega(k-1)T_0] \sin \omega T_0 ,$$

получим следующий рекуррентный алгоритм формирования

$$y[kT_0] = ay[(k-1)T_0] - y[(k-2)T_0] , \quad y[0] = 0 , \quad y[-T_0] = -\sin \omega T_0$$

где  $a = 2\cos \omega T_0$ .

Реализация алгоритма показана на рис.48.

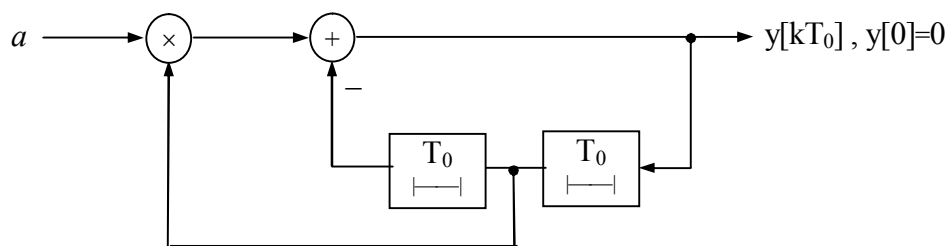


Рис.48

Рекуррентный алгоритм формирования произвольного детерминированного сигнала можно найти не только рассмотренным в примерах методом, но и используя Z-преобразование временной функции сигнала.

Полным описанием случайного сигнала (процесса) является многомерный закон распределения вероятностей значений сигнала, в моменты времени кратные периоду дискретизации  $T_0$ . В общем виде задача формирования сигнала по заданной произвольной функции распределения вероятностей весьма трудно решаемая. К счастью существуют процессы, достаточно распространенные в природе и технике, которые допускают достаточно простое описание и соответственно их формирование. К таким процессам принадлежат гауссовские или нормальные процессы. Известно, что многомерный закон распределения, описывающий гауссовский процесс  $y(t)$  определяется всего двумя характеристиками: математическим ожиданием  $m$  и корреляционной функцией  $R$ , т.е.  $y=y(m,R,t)$ .

Далее будем рассматривать гауссовские сигналы стационарного типа, у которых характеристики  $m$  и  $R$  не зависят от текущего времени  $t$ , а корреляционная функция зависит лишь от разности двух моментов времени  $\tau$ . Математическое ожидание по физическому смыслу есть среднее значение или постоянная составляющая сигнала, формирование которой весьма простая задача. Поэтому не теряя общности, в дальнейшем будем полагать, что требуется сформировать гауссовский сигнал с нулевым математическим ожиданием и заданной корреляционной функцией.

Корреляционная функция определяет степень связи двух отсчетов сигнала, разделенных интервалом  $\tau$ . Чем меньше  $\tau$ , тем сильнее связь и больше корреляционная функция  $R$ . Примеры двух типовых корреляционных функций приведены на рис.49.

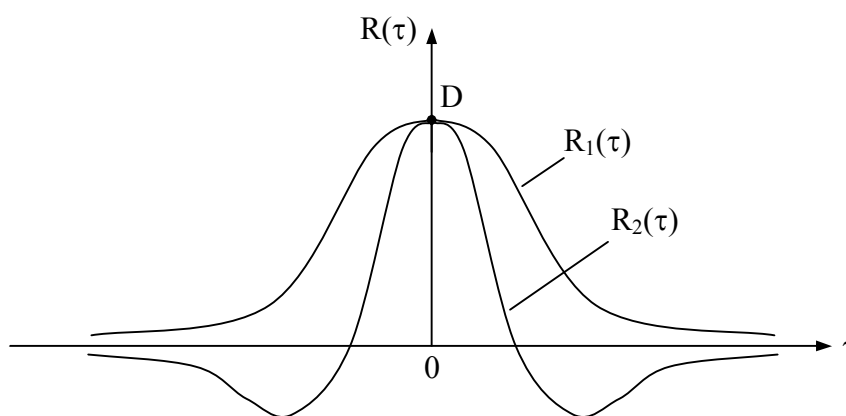


Рис.49



Корреляционная функция стационарного сигнала обладает свойством четности:  $R(-\tau)=R(\tau)$ , изменяется в пределах  $0 \leq R \leq D$ , где  $D=R(0)$  – дисперсия сигнала, и стремится к нулю при  $\tau \rightarrow \pm \infty$ . Корреляционная функция однозначно определяет энергетический спектр сигнала  $S(\omega)$  с помощью преобразования

$$S(\omega) = \int_{-\infty}^{\infty} R(\tau) e^{j\omega\tau} d\tau .$$

Энергетический спектр характеризует распределение мощности сигнала по частоте. Он является действительной функцией частоты  $\omega$  и равен квадрату модуля амплитудно-частотной характеристики сигнала. Фазовые соотношения для случайных процессов не имеют смысла. Обратное преобразование имеет следующий вид

$$R(\tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} S(\omega) e^{-j\omega\tau} d\omega .$$

Так как  $R$  и  $S$  связаны однозначно, то исходной характеристикой задания сигнала может являться и спектр  $S(\omega)$ .

Схема формирования случайного гауссовского сигнала, реализующего метод формирующего фильтра показана на рис.50.

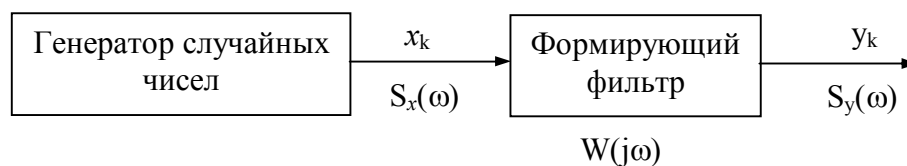


Рис.50

Генератор случайных чисел вырабатывает на каждом шаге  $k$  независимые друг от друга числа  $x_k$ , распределенные по нормальному закону. Формирующий фильтр связывает эти числа между собой для получения необходимой формы энергетического спектра  $S_y(\omega)$  выходного сигнала  $y_k$ .

Гауссовские числа  $x_k$  получают из равномерных  $u_i$ , распределенных в интервале  $(0,1)$ , с помощью алгоритма

$$u_i = u_{i-1} \cdot \lambda \pmod{d} , \quad u_0 ,$$

где  $u_0$  – начальное значение,  $\lambda$  – постоянный коэффициент,  $\pmod{d}$  – взятие числа по модулю  $d$ . При десятичном исчислении  $d=10^n$ , где  $n$  – разрядность генератора(процессора), а при двоичном  $d=2^4$ .

При умножении двух  $n$ -разрядных чисел образуется число длиной  $2n$  разрядов. с помощью операции  $\text{mod } d$  отсекается старшая  $n$ -разрядная часть слова и сохраняется младшая. Тем самым обеспечивается независимость чисел друг от друга. Коэффициент  $\lambda$  влияет на равномерность генерирования чисел. Рекомендуется выбирать  $\lambda$  для десятичной системы счисления из ряда  $7^3, 7^{13}, 7^{17}$ , а для двоичной –  $2^{18+3}$  или  $5^{2k+1}$ , где  $k$ -любое целое число. В качестве  $u_0$  можно взять произвольное число из интервала  $(0,1)$ .

Числа  $u_i$  имеют математическое ожидание величиной  $0,5$  и дисперсию величиной  $1/12$ .

При формировании независимых нормально распределенных чисел  $x_k$  с нулевым математическим ожиданием и дисперсией  $D_x=1$  можно использовать следующий простой алгоритм

$$x_k = \sum_{i=1}^{12} (u_i - 0,5) .$$

Для передаточной функции формирующего фильтра  $W(j\omega)$  спектр выходного сигнала  $S_y(\omega)$  (рис.50) определяется соотношением

$$S_y(\omega) = S_x(\omega) |W(j\omega)|^2 .$$

Энергетический спектр  $S_x(\omega)$  чисел  $x_k$  является равномерным по частоте  $\omega$  (рис.51), поскольку корреляционная функция значений  $x$  имеет узкий дельта-образный вид (значения  $x_k$  коррелированы только сами с собой). Отсюда следует, что квадрат модуля передаточной функции должен повторять форму желаемого спектра  $S_y(\omega)$ .

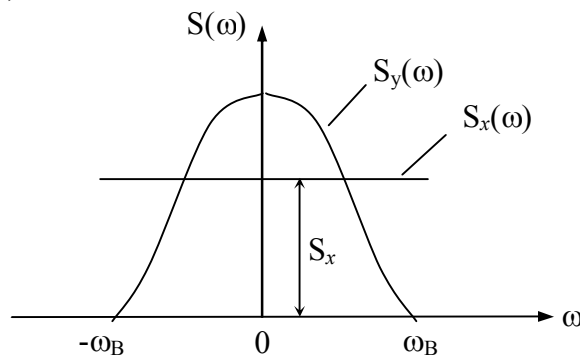


Рис.51

Величина  $S_x$  спектра входного сигнала формирующего фильтра находится из равенства

$$D_x = R_x(0) = \frac{1}{2\pi} \int_{-\omega_B}^{\omega_B} S_x d\omega = \frac{S_x \omega_B}{\pi} = 1 ,$$

откуда  $S_x = \frac{\pi}{\omega_B}$ . Следовательно, передаточная функция фильтра определяется уравнением

$$|W(j\omega)|^2 = \frac{\omega_B}{\pi} S_y(\omega) .$$

Так как  $|W|^2 = W \cdot W^*$ , где  $W^*$  – комплексно-сопряженная функция, то полученному уравнению соответствует множество передаточных функций с одинаковыми амплитудами, но разными фазовыми частотными характеристиками. Выберем фильтр с нулевой фазовой частотной характеристикой, для него  $W(j\omega) = W(\omega) = W^*(\omega)$ . Тогда имеем

$$W(\omega) = \sqrt{\frac{\omega_B}{\pi} S_y(\omega)} .$$

Используя обратное преобразование Фурье для аналогового сигнала, найдем импульсную реакцию  $g(t)$  формирующего фильтра:

$$g(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} W(j\omega) e^{j\omega t} d\omega = \frac{1}{2\pi} \int_{-\omega_B}^{\omega_B} \sqrt{\frac{\omega_B}{\pi} S_y(\omega)} e^{j\omega t} d\omega = \frac{1}{\pi} \int_0^{\omega_B} \sqrt{\frac{\omega_B}{\pi} S_y(\omega)} \cos \omega t d\omega .$$

Далее переходим к дискретному времени  $t = kT_0$  и определяем импульсную реакцию цифрового фильтра  $c_k = g_k T_0$ . Наконец применяя формулу “свертка” (раздел 7), получаем алгоритм вычисления случайного гауссовского сигнала

$$y_k = \sum_{i=k-m}^{k+m} x_{k-i} c_i = \sum_{i=-m}^m x_i c_{k-i} .$$

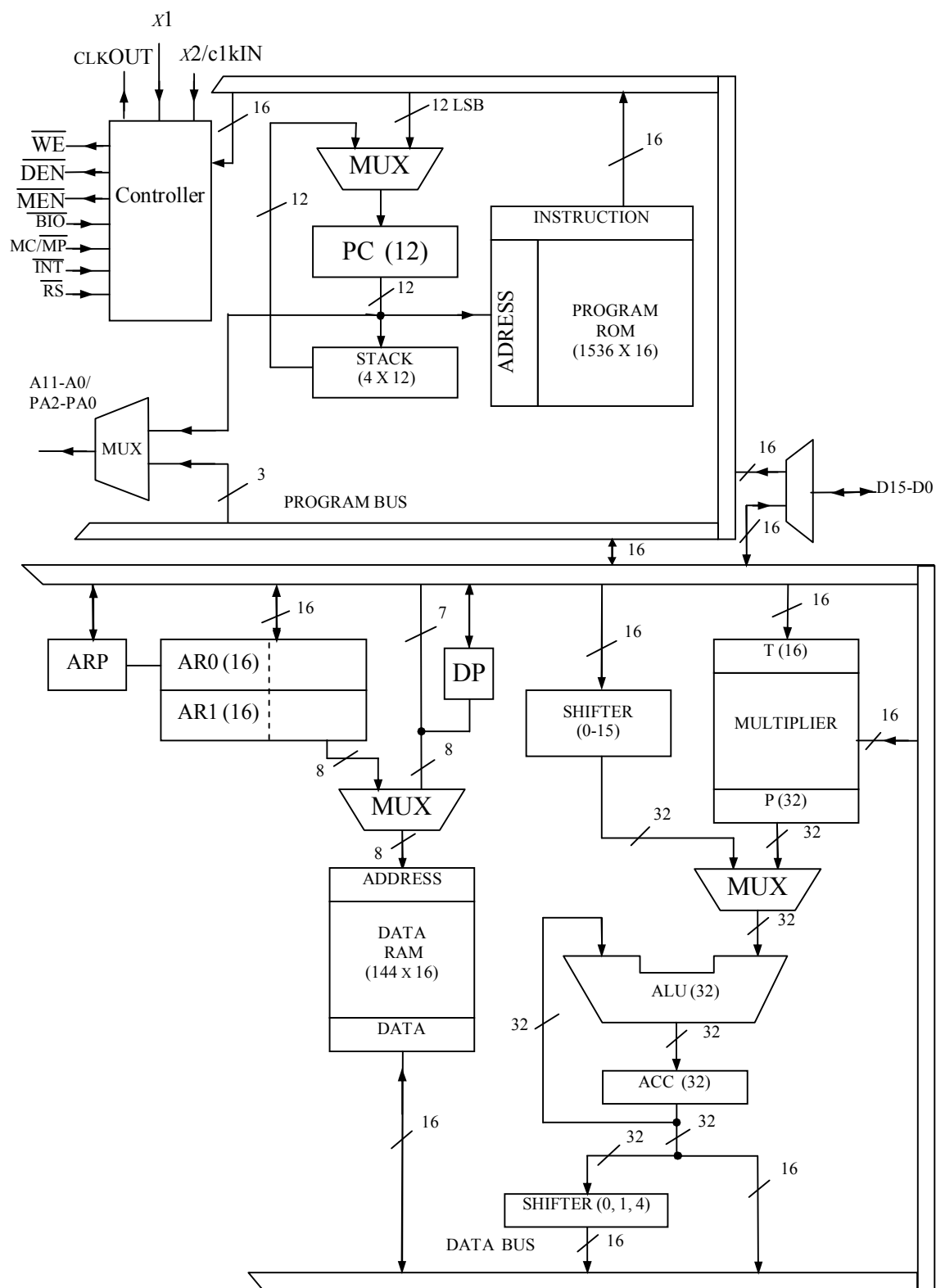
Заметим, что  $c_{-i} = c_i \neq 0$ , если  $i \leq m$ , что является следствием четности корреляционной функции формируемого сигнала.

Изложенный в пособии материал содержит вводные данные о современных системах и методах обработки сигнальной информации. Эти системы и методы постоянно развиваются и совершенствуются. Так в последнее время для обработки сигналов все шире начало использоваться эффективное вейвлет-преобразование. Более глубокая информация о средствах и способах обработки сигналов в выпускаемой технической литературе, из которой заинтересованный читатель, освоив данное пособие, может самостоятельно подчерпнуть новые сведения.

## ЛИТЕРАТУРА

1. Цифровые процессоры обработки сигналов: Справочник / А. Г. Остапенко, С. И. Лавлинский, А. Б. Сушков и др.; Под ред. А. Г. Остапенко. - М.: Радио и связь, 1994.- 256с.
2. Цифровая обработка сигналов : Учебное пособие для вузов / Л. М. Гольденберг, Б. Д. Матюшкин, М. Н. Поляк. - М.: Радио и связь, 1990.- 256с.
3. Вишняков В. А. Цифровое моделирование сигналов: Учебное пособие / РАТИ.- Рыбинск, 1993.- 88с.
4. Рабинер Л., Гоулд Б. Теория и применение цифровой обработки сигналов. - М.: Мир, 1978.- 848с.
5. Руководство пользователя по сигнальным микропроцессорам семейства ADSP-2100 / Пер. с англ. О. В. Луневой ; Под ред. А. Д. Викторова. – Санкт-Петербург СПб ГЭТУ, 1997. –520с.
6. Быков В. В. Цифровое моделирование в статической радиотехнике. –М.: Советское радио, 1971. –328с.
7. Марр Д. Зрение. Информационный подход к изучению, представлению и обработке зрительных образов. –М.: Радио и связь, 1987. –546с.
8. Корн Г., Корн Т. Справочник по математике для научных работников и инженеров. –М. : Наука, 1968. –720с.
9. Маслюк Л. Л. Дайджест вейвлет-анализа. Компьютерра. –1998.– № 8.

## ФУНКЦИОНАЛЬНЫЙ БЛОК ЦСП TMS32010



## Система команд ЦСП TMS32010

Мнемоника	Алгоритм	Число слов	Число циклов	Описание команды
ABS	$ACC \leftarrow  ACC $	1	1	Тестирование абсолютного значения аккумулятора
ADD A,C	$ACC \leftarrow M(A) * 2^C \rightarrow ACC$	1	1	Сложение содержимого ячейки A памяти данных с учетом сдвига C с содержимым аккумулятора
ADDDH A	$ACC \leftarrow M(A) * 2^{16} \rightarrow ACC$	1	1	Сложение содержимого ячейки A памяти данных со старшими разрядами аккумулятора
ADDS A	$ACC \leftarrow M(A) \rightarrow ACC$	1	1	Сложение младших разрядов аккумулятора с содержанием указанной ячейки памяти данных с подавлением расширения знака. Данные обрабатываются как 16-разрядные положительные числа (а не как дополнительный код целого в ADD)
AND A	$ACC \leftarrow M(A) * ACC \rightarrow ACC$	1	1	Операция И содержимого младших разрядов аккумулятора с содержанием ячейки A; операция И старших разрядов с нулевыми значениями.
APAC	$ACC \leftarrow P \rightarrow ACC$	1	1	Сложение содержимого P-регистра и аккумулятора
B W	PC	2	2	Безусловный переход в ячейку памяти программ W, загрузка адреса W в счетчик программ
BANZ W	W $\rightarrow$ PC, если AR $\neq$ 0, AR - 1 $\rightarrow$ AR	2	2	Переход на ячейку памяти W, если содержимое вспомогательного регистра не равно нулю, а содержимое AR декрементируется
BGEZ W	W $\rightarrow$ PC, если ACC $\geq$ 0	2	2	Переход на ячейку памяти W, если содержимое аккумулятора больше или равно нулю
BGZ W	W $\rightarrow$ PC, если ACC $>$ 0	2	2	Переход на ячейку памяти W, если содержимое аккумулятора больше нуля. В противном случае счетчик команд инкрементируется
BIOZ W	W $\rightarrow$ PC, если BIO=0	2	2	Переход на ячейку памяти W, если уровень сигнала на входе BIO равно нулю. В противном случае счетчик команд инкрементируется
BLEZ W	W $\rightarrow$ PC, если ACC $\leq$ 0	2	2	Переход на ячейку памяти W, если содержимое аккумулятора меньше или равно нулю
BLZ W	W $\rightarrow$ PC, если ACC $<$ 0	2	2	То же, если содержимое аккумулятора меньше нуля
BNZ W	W $\rightarrow$ PC, если ACC $\neq$ 0	2	2	То же, если содержимое аккумулятора не равно нулю
BV W	W $\rightarrow$ PC, если OV=1	2	2	То же, если содержимое OV равно 1
BZ W	W $\rightarrow$ PC, если ACC=0	2	2	То же, если содержимое аккумулятора равно нулю
CAL A	PC+1 $\rightarrow$ TOS ACC(11...0) $\rightarrow$ PC	1	2	Вызов подпрограммы по косвенному адресу, содержащемуся в аккумуляторе. Текущее значение счетчика декрементируется и помещается на вершину стека
CALL W	PC+1 $\rightarrow$ CTEK W $\rightarrow$ PC	2	2	Вызов подпрограммы по прямому адресу
DINT	1 $\rightarrow$ INTM	1	1	Запрещение прерываний: запись единицы в разряд управления прерыванием
DMOW A	M(A) $\rightarrow$ M(A+1)	1	1	Перемещение данных в памяти данных: содержимое ячейки A копируется в соседнюю ячейку, адрес которой на единицу больше
EINT	0 $\rightarrow$ INTM	1	1	Разрешение прерываний: запись нуля в разряд управления прерыванием
IN A,PA	PA $\rightarrow$ M(A)	1	1	Ввод данных через порт с адресом PA в ячейку A памяти данных
LAC A,C	$M(A) * 2^C \rightarrow ACC$	1	1	Загрузка аккумулятора содержанием ячейки A со сдвигом C
LACK A	K(7..0) $\rightarrow$ ACC	1	1	Загрузка аккумулятора восьмиразрядной константой K
LAR AR,A	M(A) $\rightarrow$ AR	1	1	Загрузка вспомогательного регистра AR содержанием ячейки A памяти данных
LARK AR,K	K(7..0) $\rightarrow$ AR	1	1	Загрузка вспомогательного регистра AR восьмиразрядной константой K
LARP Y	Y $\rightarrow$ ARP	1	1	Загрузка указателя вспомогательного регистра однобайтовой константой Y
LDP A	M(A) $\rightarrow$ DP	1	1	Загрузка указателя страницы памяти данных младшим разрядом ячейки A памяти данных
LDPK D	D $\rightarrow$ DP	1	1	Загрузка указателя страницы памяти данных однокбитной константой D
LST A	M(A) $\rightarrow$ F	1	1	Загрузка регистра состояния содержимым ячейки A памяти данных
LT A	M(A) $\rightarrow$ T	1	1	Загрузка регистра T содержимым указанной ячейки памяти данных
LTA A	M(A) $\rightarrow$ T, ACC+P $\rightarrow$ ACC	1	1	Загрузка T-регистра содержимым A; сложение содержимого регистра произведения P с аккумулятором, загрузка результата в аккумулятор
LTD A	M(A) $\rightarrow$ T, ACC+P $\rightarrow$ ACC, M(A) $\rightarrow$ M(A+1)	1	1	Загрузка T-регистра содержимым A; сложение содержимого регистра произведения P с аккумулятором, загрузка результата в аккумулятор; перемещение содержимого ячейки памяти данных в следующую по старшинству ячейку
MAR /*+/-,Y	AR+1/AR-1 $\rightarrow$ AR, Y $\rightarrow$ ARP	1	1	Модификация содержимого вспомогательного регистра: либо остается без изменений, либо инкрементируется, либо декрементируется. Указатель вспомогательного регистра загружается следующим значением ARP или остается неизменным
MPY A	T*M(A) $\rightarrow$ P	1	1	Умножение содержимого T-регистра на содержимое ячейки A памяти данных
MPYK Q	T*O(12..0) $\rightarrow$ P	1	1	Непосредственное умножение содержимого T-регистра на 13-разрядную константу. Результат в P-регистре
NOP	PC+1 $\rightarrow$ PC	1	1	Нет операции

OR A	$0 + ACCH \rightarrow ACCH,$ $M(A) + ACCL \rightarrow ACCL$	1	1	Операция логического ИЛИ старших разрядов аккумулятора с нулем, а младших с содержимым ячейки A памяти данных
OUT A,PA	$M(A) \rightarrow PA$	1	1	Вывод данных через порт
PAC	$P \rightarrow ACC$	1	1	Загрузка аккумулятора содержимым P-регистра
POP	$TOS \rightarrow ACC$	1	2	Извлечение содержимого вершины стека в аккумулятор
PUSH	$ACC(11..0) \rightarrow TOS$	1	2	Загрузка содержимого младших разрядов аккумулятора в стек
RET	$TOS \rightarrow PC$	1	2	Возврат из подпрограммы
ROVM	$0 \rightarrow OVM$	1	1	Сброс в «0» разряда управления режимом управления режимом переполнения (запрещение прерывания)
SACH A,X	$ACCH * 2^X \rightarrow M(A),$ $X=0,1,4$	1	1	Запоминание старших разрядов аккумулятора со сдвигом
SACL A	$ACCL \rightarrow M(A)$	1	1	Запоминание младших разрядов аккумулятора
SAR AR,A	$AR \rightarrow M(A)$	1	1	Запоминание содержимого вспомогательного регистра в указанной ячейке памяти
SOVM	$1 \rightarrow OVM$	1	1	Установление в «1» разряда управления режимом переполнения
SPAC	$ACC - P \rightarrow ACC$	1	1	Вычитание содержимого P-регистра из аккумулятора
SSTA	$F \rightarrow M(A)$	1	1	Запоминание битов регистра состояния F в указанной ячейке памяти
SUB A,C	$ACC - M(A) * 2^C \rightarrow ACC$	1	1	Вычитание из аккумулятора содержимого ячейки A памяти данных со сдвигом
SUBC A	$Y = ACC - M(A) * 2^{15}$ Если $y \geq 0$ , то $y * 2 + 1 \rightarrow ACC$ , Если $y < 0$ , то $y * 2 \rightarrow ACC$	1	1	Операция условного вычитания
SUBH A	$ACC(31..15) -$ $M(A) * 2^{16} \rightarrow ACC$	1	1	Вычитание из старших разрядов аккумулятора содержимого ячейки A памяти данных
SUBS A	$ACCL - M(A) \rightarrow ACC$	1	1	Вычитание из младших разрядов аккумулятора содержимого ячейки A с подавлением расширения знака
TBLR A	$PЗУ \rightarrow M(A)$	1	3	Считывание таблицы: слово из любой ячейки памяти программ (встроенной, внешней) передается в указанную ячейку памяти данных
TBLW F	$M(A) \rightarrow ПЗУ$	1	3	Запись в таблицу: слово из указанной ячейки памяти данных передается в ячейку внешней памяти
XOR A	$ACCH + 0 \rightarrow ACCH,$ $ACC + M(A) \rightarrow ACC$	1	1	Исключающее ИЛИ младших разрядов аккумулятора
ZAC	$0 \rightarrow ACC$	1	1	Установка в нуль аккумулятора
ZALH A	$M(A) * 2^{16} \rightarrow ACCH,$ $0 \rightarrow ACC$	1	1	Установка в нуль аккумулятора и загрузка старших разрядов аккумулятора
ZALS A	$M(A) \rightarrow ACCL,$ $0 \rightarrow ACCH$	1	1	Установка в нуль аккумулятора и загрузка младших разрядов с подавлением знакорасширения



## СОДЕРЖАНИЕ

Введение.....	3
1. Сигналы и процессоры их обработки.....	3
2. Цифровой сигнальный процессор TMS 32010.....	5
3. Проектирование систем на базе процессоров обработки сигналов.....	8
4. Программирование сигнальных процессоров.....	12
5. Дискретизация сигналов.....	14
6. Цифровые фильтры.....	15
7. Нерекурсивные фильтры.....	16
8. Рекурсивные фильтры.....	20
9. Описание фильтров в частотной области. Дискретное преобразование Фурье. Z –преобразование.....	22
10. Вычисление реакций фильтров (Решение рекуррентных уравнений).....	25
11. Передаточная функция фильтра.....	27
12. Устойчивость цифровых фильтров.....	30
13. Фильтр Калмана.....	32
14. Проектирование цифровых фильтров.....	34
15. Двумерная цифровая фильтрация.....	37
16. Спектральный анализ сигналов.....	40
17. Быстрое преобразование Фурье.....	43
18. Моделирование детерминированных сигналов.....	46
19. Формирование случайных сигналов.....	49
20. Заключение.....	53
21. Литература.....	54
22. Приложение №1. Функциональный блок ЦСП TMS 32010 .....	55
23. Приложение №2. Система команд ЦСП TMS 32010 .....	56