

НЕРАВНОПОЛОСНЫЕ БАНКИ ФИЛЬТРОВ ДЛЯ СЛУХОВЫХ АППАРАТОВ: АНАЛИЗ АЛГОРИТМОВ, АВТОМАТИЗАЦИЯ ПРОЕКТИРОВАНИЯ

М.И. Вашкевич, А.А. Петровский
Белорусский государственный университет информатики
и радиоэлектроники, Минск.

Рассматриваются вопросы отображения алгоритмов неравнополосных косинусно-модулированных банков фильтров на основе фазового преобразования на процессоры для слуховых аппаратов. Делается акцент на автоматизацию данного процесса. Анализируются требования, предъявляемые к аппаратной реализации банка фильтров для слуховых аппаратов. На основе результатов анализа формируются основные стратегии по отображению алгоритмов неравнополосных банков фильтров на процессоры для слуховых аппаратов.

Введение

Основной задачей, решаемой в слуховых аппаратах, являются частотный анализ акустического сигнала и усиление/ослабление отдельных составляющих спектра. При этом зачастую средством декомпозиции сигнала служит банк цифровых фильтров, который реализован в виде отдельного процессорного ядра [1]. Сложность проектирования подобных устройств связана с требованиями, накладываемыми на банк фильтров:

- задержка банка фильтров не должна превышать 8–9 мс [2];
- входной сигнал поступает с частотой дискретизации 16–20 кГц;
- потребляемая мощность аппаратной реализации банка фильтров должна составлять 1 мВт на 1 В напряжения питания [3], т. е. потребляемый ток равен 1 мА. Данное требование вытекает из следующих рассуждений: слуховой аппарат должен обеспечивать как минимум 16-часовую бесперебойную работу в течение 7 дней (112 часов). Если при этом использовать батарею на 1,3 В (150 мА/ч), то максимальный ток должен составлять 1,33 мА [1];
- тактовая частота сопроцессора, реализующего банк фильтров, в практических приложениях не должна превышать 1–2,5 МГц [1, 3];
- для согласованности со слуховой системой человека банк фильтров должен обладать свойством неравнополосности, т. е. должен аппроксимировать критические частотные полосы;

Для того чтобы конечное устройство удовлетворяло заданным требованиям, необходимо на этапе проектирования иметь возможность изменять регулируемые параметры и наблюдать изменения выходных характеристик. Это приводит к необходимости создания среды проектирования сопроцессора банка фильтров для слуховых аппаратов. В последующих разделах описываются математическая модель предлагаемого банка фильтров, его структурная схема, алгоритмы эффективной реализации всех блоков банка фильтров. Излагаются идеи по автоматизации процесса проектирования банка фильтров.

1. Математическая модель банка фильтров

В качестве банка фильтров для слуховых аппаратов предлагается использовать неравнополосный косинусно-модулированный банк фильтров (КМБФ). Данный банк

фильтров образуется из своего равнополосного аналога путем применения фазового преобразования [4], т. е. замены всех элементов задержки на фазовые звенья $z^{-1} \rightarrow A(z)$. Математически M -канальный банк фильтров задается передаточными функциями фильтров анализа и синтеза:

$$\begin{aligned} H_k(z) &= a_k b_k H\left(A^{-1}(z)W_{2M}^{(k+0,5)}\right) + \bar{a}_k \bar{b}_k H\left(A^{-1}(z)W_{2M}^{-(k+0,5)}\right); \\ F_k(z) &= \bar{a}_k b_k H\left(A^{-1}(z)W_{2M}^{(k+0,5)}\right) + a_k \bar{b}_k H\left(A^{-1}(z)W_{2M}^{-(k+0,5)}\right); \end{aligned} \quad (1)$$

где $a_k = e^{j(-1)^k \pi/4}$, $b_k = W_{2M}^{\frac{N-1}{2}(k+0,5)}$, $W_M = e^{j2\pi/M}$, а верхняя черта означает комплексно-сопряженное число. Через $H(z)$ в (1) обозначен фильтр-прототип нижних частот с конечной импульсной характеристикой (КИХ) порядка N и частотой среза $\omega_c = \pi/2M$. Данный фильтр рассчитывается методами, разработанным для равнополосных КМБФ [5, 6]. $A(z)$ в (1) – это фазовое звено, в данной работе ограничимся рассмотрением фазового звена первого порядка, которое определяется как

$$A(z) = (\alpha + z^{-1})/(1 + \alpha z^{-1}), \quad |\alpha| \leq 1, \quad A(e^{j\omega}) = e^{j\phi(\omega)},$$

где

$$\phi(\omega) = -\omega + 2\text{arctg}(\alpha \sin \omega / (\alpha \cos \omega - 1)). \quad (2)$$

Замена $z^{-1} \rightarrow A(z)$ приводит к деформации оси частот ($\omega \rightarrow \phi(\omega)$) и получению неравнополосного банка фильтров. Этот процесс показан на рис. 1.

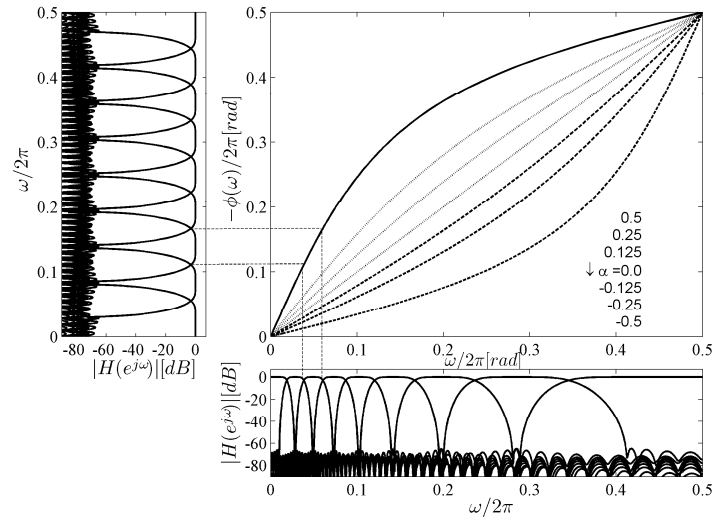


Рис. 1. Неравнополосный банк фильтров на основе деформации оси частот

Таким образом, ширина полос в банке фильтров регулируется параметром α в выражении (2).

2. Полифазная структура банка фильтров

В [7] для эффективной реализации КМБФ было предложено использовать полифазное представление фильтра-прототипа

$$H(z) = \sum_{r=0}^{N-1} h[n]z^{-n} = \sum_{q=0}^{2M-1} \sum_{r=0}^{m-1} h[q + 2rM]z^{-(q+2rM)}. \quad (3)$$

Длина фильтра прототипа $N = 2mM$, где m – произвольное целое число. Применяя выражение (3) и выполняя преобразования аналогичные [7], можно записать следующие выражения для эффективной реализации КМБФ на основе деформации оси частот:

$$H_k(z) = \sum_{q=0}^{2M-1} c_{k,q} \sum_{r=0}^{m-1} (-1)^r h[q + 2rM] (A(z))^{(q+2rM)}, \quad (4)$$

где $c_{k,l} = 2 \cos\left(\frac{(2k+1)\pi}{2M}\left(l - \frac{N-1}{2}\right) + (-1)^k \frac{\pi}{4}\right)$, введем также обозначение $g[q + 2rM] = (-1)^r h[q + 2rM]$. На основании (4) строится полифазная структура неравнополосного КМБФ анализа, структура банка синтеза получается аналогичным образом (рис. 2).

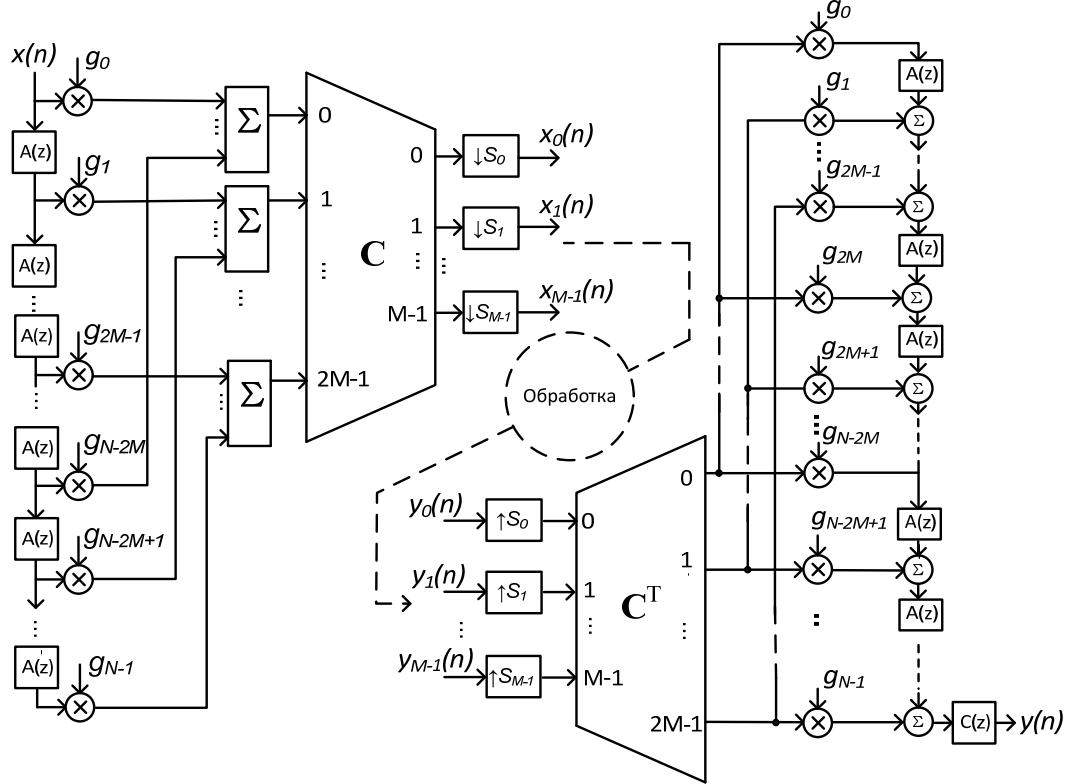


Рис. 2. Полифазная структура неравнополосного банка фильтров на основе деформации оси частот

На рис.2 введены структурные элементы, требующие пояснения. Блок косинусной модуляции (КМ) обозначен через $\mathbf{C} = [c_{k,l}]_{0 \leq k < M, 0 \leq l < 2M}$, в нем выполняется умножение выходов полифазных компонент на матрицу косинусной модуляции \mathbf{C} . Коэффициенты децимации/интерполяции обозначены через S_0, S_1, \dots, S_{M-1} . Алгоритм выбора этих параметров описан в работе [8].

2. Вычислительная сложность

Используя структурную схему на рис. 2, можно оценить вычислительную сложность банка фильтров в терминах количества операций сложения и умножения. Для простоты рассмотрим вычислительную сложность только банка фильтров анализа:

$$MUL_A = MUL_{AP} + MUL_{FIR} + MUL_{MOD};$$

$$ADD_A = ADD_{AP} + ADD_{FIR} + ADD_{MOD},$$

где MUL_A и ADD_A – это количество операции умножения и сложения соответственно. MUL_{AP} , MUL_{FIR} и MUL_{MOD} – число умножений необходимых для реализации цепочки фазовых звеньев, полифазной фильтрации и блока КМ соответственно. Аналогично ADD_{AP} , ADD_{FIR} и ADD_{MOD} это количество операций сложения, необходимых для организации вычислений в этих блоках.

MUL_{FIR} и ADD_{FIR} зависят от порядка фильтра-прототипа (N), так $MUL_{\text{FIR}} = N$, а $ADD_{\text{FIR}} = N - 2M$. Составляющие MUL_{AP} и ADD_{AP} зависят как от длины фильтра-прототипа, так и от схемы реализации фазового звена. Если фазовое звено реализуется согласно уравнению:

$$y[n] = x[n - 1] + \alpha(x[n] - y[n - 1]), \quad (5)$$

то реализация цепочки фазовых звеньев потребует $MUL_{\text{AP}} = N - 1$, $ADD_{\text{AP}} = 2(N - 1)$.

В блоке косинусной модуляции выполняется умножение вектора выходов полифазных компонентов $2M \times 1$ на матрицу косинусной модуляции \mathbf{C} размера $M \times 2M$. Для прямой реализации данной операции потребуется $MUL_{\text{MOD}} = 2M^2$, $ADD_{\text{MOD}} = M^2 - M$.

В дальнейшем будет показано, как вычислительная сложность банка фильтров может быть уменьшена.

3. Быстрый алгоритм реализации косинусной модуляции

В работе [7] был предложен способ уменьшения вычислительной сложности реализации блока КМ путем факторизации матрицы \mathbf{C} :

$$\mathbf{C} = \begin{cases} \sqrt{M}(-1)^{m_1} \mathbf{DCT4}_M[(\mathbf{I}_M - \mathbf{J}_M) \quad -(\mathbf{I}_M + \mathbf{J}_M)] \\ \text{для четных } m \ (m = 2m_1) \\ \sqrt{M}(-1)^{m_1} \mathbf{DCT4}_M[(\mathbf{I}_M + \mathbf{J}_M) \quad (\mathbf{I}_M - \mathbf{J}_M)] \\ \text{для нечетных } m \ (m = 2m_1 + 1), \end{cases} \quad (6)$$

где $\mathbf{DCT4}_M$ – матрица дискретного косинусного преобразования типа 4 (ДКП-4),

$$\mathbf{DCT4}_M = \left[\sqrt{\left(\frac{2}{M}\right)} \cos\left(\frac{\pi(2k+1)(2l+1)}{4M}\right) \right]_{0 \leq k, l < M}, \quad \mathbf{I}_M = \begin{bmatrix} 1 & & \\ & \ddots & \\ & & 1 \end{bmatrix}, \quad \mathbf{J}_M = \begin{bmatrix} & & 1 \\ & \ddots & \\ 1 & & \end{bmatrix}. \quad (7)$$

Применение формулы (6) позволяет уменьшить вычислительную сложность блока КМ почти в два раза: $MUL_{\text{MOD}} = M^2$, $ADD_{\text{MOD}} = M^2 + 2M$.

В данной работе предлагается иная схема факторизации матрицы \mathbf{C} . Для матриц суммирования из (6) введем следующие обозначения:

$$\mathbf{S}_M^e = [(\mathbf{I}_M - \mathbf{J}_M) \quad -(\mathbf{I}_M + \mathbf{J}_M)]; \quad \mathbf{S}_M^o = [(\mathbf{I}_M + \mathbf{J}_M) \quad (\mathbf{I}_M - \mathbf{J}_M)]. \quad (8)$$

Было замечено, что (8) можно записать следующим образом

$$\mathbf{S}_M^e = \begin{cases} \begin{bmatrix} \mathbf{I}_{M/2} & \mathbf{I}_{M/2} \\ -\mathbf{J}_{M/2} & \mathbf{J}_{M/2} \end{bmatrix} \times \begin{bmatrix} \mathbf{I}_{M/2} & -\mathbf{J}_{M/2} & \mathbf{0} \\ \mathbf{0} & -\mathbf{I}_{M/2} & -\mathbf{J}_{M/2} \end{bmatrix}, & \text{для четных } M; \\ \begin{bmatrix} \mathbf{I}_{\frac{M-1}{2}} & \mathbf{I}_{\frac{M-1}{2}} \\ -\mathbf{J}_{\frac{M-1}{2}} & \mathbf{J}_{\frac{M-1}{2}} \end{bmatrix} \times \begin{bmatrix} \mathbf{I}_{\frac{M-1}{2}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{J}_{\frac{M-1}{2}} & \mathbf{0} \\ \mathbf{0} & -\mathbf{I}_{\frac{M-1}{2}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -\mathbf{J}_{\frac{M-1}{2}} \end{bmatrix}, & \text{для нечетных } M; \end{cases} \quad (9)$$

$$\mathbf{S}_M^o = \begin{cases} \begin{bmatrix} \mathbf{I}_{M/2} & \mathbf{I}_{M/2} \\ -\mathbf{J}_{M/2} & \mathbf{J}_{M/2} \end{bmatrix} \times \begin{bmatrix} \mathbf{I}_{M/2} & \mathbf{J}_{M/2} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{M/2} & -\mathbf{J}_{M/2} \end{bmatrix}, & \text{для четных } M; \\ \begin{bmatrix} \mathbf{I}_{\frac{M-1}{2}} & \mathbf{I}_{\frac{M-1}{2}} \\ \mathbf{J}_{\frac{M-1}{2}} & -\mathbf{J}_{\frac{M-1}{2}} \end{bmatrix} \times \begin{bmatrix} \mathbf{I}_{\frac{M-1}{2}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_{\frac{M-1}{2}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{J}_{\frac{M-1}{2}} \\ \mathbf{0} & \mathbf{I}_{\frac{M-1}{2}} & \mathbf{0} \end{bmatrix}, & \text{для нечетных } M. \end{cases} \quad (10)$$

Использование выражений (9) и (10) позволяет уменьшить количество сложений в блоке КМ до $ADD_{MOD} = M^2 + M$.

Дальнейшее уменьшение вычислительной сложности блока КМ связано с использованием быстрого алгоритма для вычисления ДКП-4. В работе [9] предлагается метод вывода формул для быстрых дискретных косинусных и синусных преобразований в рамках алгебраической теории обработки сигналов. Общая формула для быстрого алгоритма ДКП-4 формата n имеет вид:

$$\mathbf{DCT4}_n = \mathbf{K}_m^n \left(\mathbf{I}_k \otimes_i \mathbf{DCT4}_m \left(\frac{i+1/2}{k} \right) \right) (\overline{\mathbf{DST3}}_k \otimes \mathbf{I}_m) \overline{\mathbf{B}}_{k,m}^{C4}, \quad (11)$$

где $n = km$; \otimes – кронекеровское произведение ($A \otimes B = [a_{k,l}B]_{k,l}$ для $A = [a_{k,l}]$), \oplus – прямая сумма ($A \oplus B = \begin{bmatrix} A & \\ & B \end{bmatrix}$); $\mathbf{DCT4}_m(r)$ – смещенное ДКП-4; $\overline{\mathbf{DST3}}_k$ – масштабированное дискретное синусное преобразование типа III,

$$\overline{\mathbf{B}}_{k,m}^{C4} = \begin{bmatrix} \mathbf{I}_m & -\mathbf{J}_m & & \\ & \ddots & \ddots & \\ & & \mathbf{I}_m & -\mathbf{J}_m \\ & & & \mathbf{I}_m \end{bmatrix}; \quad \mathbf{K}_m^n = (\mathbf{I}_k \oplus \mathbf{J}_k \oplus \mathbf{I}_k \oplus \mathbf{J}_k \oplus \dots) \mathbf{L}_m^n. \quad (12)$$

\mathbf{L}_m^n является матрицей перестановки, которая определяется как

$$\mathbf{L}_m^n: \quad i \mapsto im \bmod n - 1, \quad \text{для } 0 \leq i < n - 1, \\ n - 1 \mapsto n - 1,$$

что означает: матрица \mathbf{L}_m^n имеет в строке i значение 1 в позиции $(im \bmod n - 1)$ и 0 – во всех остальных. Подробно вывод формулы (11) описан в [9].

Важность выражения (11) в том, что оно позволяет получить несколько различных быстрых алгоритмов одного и того же преобразования. Фактически каждому быстрому алгоритму соответствует своя факторизация матрицы преобразования. Среди полученных факторизаций можно найти такую, для реализации которой требуется наименьшее число умножений. Важно отметить, что поиск быстрого алгоритма за счет формализации (11) может быть полностью автоматизирован. В качестве примера приведем две факторизации матрицы восьмиточечного ДКП-4, которые были получены способом формального синтеза.

$$\mathbf{DCT4}_8 = \mathbf{K}_4^8 \text{diag}(\mathbf{K}_2^4 \begin{bmatrix} \mathbf{DCT4}_2(1/8) & \\ & \mathbf{DCT4}_2(7/8) \end{bmatrix} (\overline{\mathbf{DST3}}_2(1/4) \otimes \mathbf{I}_2) \overline{\mathbf{B}}_{2,2}^{C4}, \quad (13) \\ \mathbf{K}_2^4 \begin{bmatrix} \mathbf{DCT4}_2(3/8) & \\ & \mathbf{DCT4}_2(5/8) \end{bmatrix} (\overline{\mathbf{DST3}}_2(3/4) \otimes \mathbf{I}_2) \overline{\mathbf{B}}_{2,2}^{C4}) \times (\overline{\mathbf{DST3}}_2(1/4) \otimes \mathbf{I}_4) \overline{\mathbf{B}}_{2,4}^{C4}$$

и

$$\mathbf{DCT4}_8 = \mathbf{K}_2^8 \begin{bmatrix} \mathbf{DCT4}_2(1/8) & & & \\ & \mathbf{DCT4}_2(3/8) & & \\ & & \mathbf{DCT4}_2(5/8) & \\ & & & \mathbf{DCT4}_2(7/8) \end{bmatrix} \quad (14) \\ ((\mathbf{K}_2^4 \begin{bmatrix} \overline{\mathbf{DST3}}_2(1/4) & \\ & \overline{\mathbf{DST3}}_2(3/4) \end{bmatrix} (\overline{\mathbf{DST3}}_2 \otimes \mathbf{I}_2) \overline{\mathbf{B}}_{2,2}^{S3}) \otimes \mathbf{I}_2) \overline{\mathbf{B}}_{4,2}^{C4}.$$

Факторизация (13) получена при сочетании параметров $k = 4, m = 2$, а (14) – при $k = 2, m = 4$. Если учесть, что

$$\overline{\mathbf{DST3}}_2(r) = \mathbf{F}_2 \text{diag}(1, 2 \cos \frac{r\pi}{2}), \quad \mathbf{DCT4}_2(r) = \text{diag} \left(\cos \frac{r\pi}{4}, \sin \frac{r\pi}{4} \right) \mathbf{F}_2 \begin{bmatrix} 1 & \\ 0 & 2 \cos \frac{r\pi}{2} \end{bmatrix}, \quad \mathbf{F}_2 = \begin{bmatrix} 1 & \\ 1 & -1 \end{bmatrix},$$

то можно убедиться, что в обоих случаях требуется 20 операций умножения. Однако для произвольного n могут получаться факторизации разной сложности.

В случае, когда n является степенью двойки, число умножений для быстрого алгоритма ДКП-4 равно $1/2 \log_2(n) + n$, а сложений – $3/2 \log_2(n)$ [9]. Таким образом вычислительная сложность блока КМ равна:

$$MUL_{MOD} \approx 1/2 \log_2(M) + M; \quad ADD_{MOD} \approx 3/2 \log_2 M + 2M. \quad (15)$$

4. Реализация цепочки фазовых звеньев

Учитывая (5), каскадное соединение цепочки фазовых звеньев (ФЗ) имеет следующий вид (рис. 3).

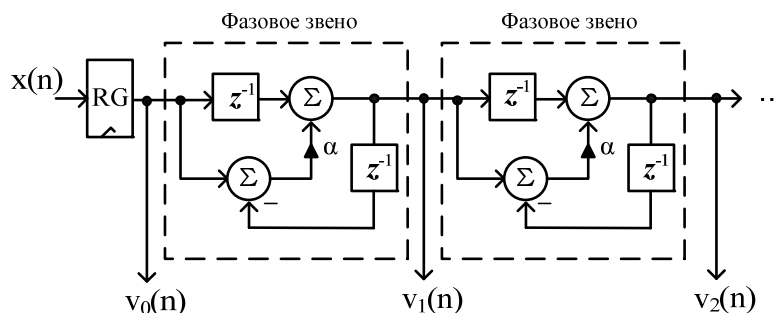
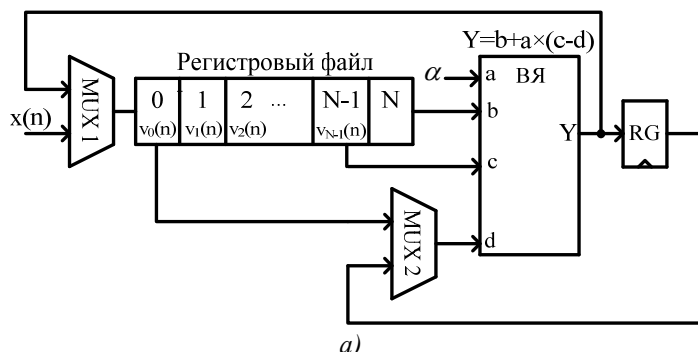


Рис. 3. Блок-схема цепочки фазовых звеньев

Данная блок-схема показывает, что цепочка ФЗ обладает регулярной структурой, что позволяет выделить одно вычислительное ядро (ВЯ) и организовать вычисление путем правильной подачи данных на вход ВЯ (рис. 4, а). С другой стороны, цепочку ФЗ можно реализовать в виде вычислительной процедуры на универсальном процессоре (рис. 4, б).



```

1: R1 = x[n];
2: R2 = v[0];
3: v[0] = R1;
4: for i = 1, 2, ..., N - 1 do
5:   R3 = R2;
6:   R2 = v[i];
7:   R1 = (R2 - R1) * alpha + R3;
8:   v[i] = R1
9: end for

```

а) аппаратная реализация; б) программная реализация

Рис. 4. Цепочка фазовых звеньев: а) аппаратная реализация; б) программная реализация

Из рис. 4, а видно, что мультиплексоры MUX1 и MUX2 обеспечивают правильную подачу входных отсчетов в ВЯ. Наиболее ресурсоемким элементом является регистровый файл, в котором хранится память всех секций, составляющих цепочку ФЗ. Длина регистрового файла равна $N + 1$, где N – это порядок фильтра-прототипа.

5. Реализация полифазных компонент фильтра-прототипа

Входом полифазных компонент служат выходные сигналы цепочки ФЗ (рис. 2). Общая блок-схема формирования выхода i -го полифазного компонента показана на рис. 5.

Практическая реализация полифазных компонент может быть выполнена с использованием МАС-структур, на основе распределенной арифметики [10], либо про-

граммно на универсальном процессоре. Последние две возможности показаны на рис. 6.

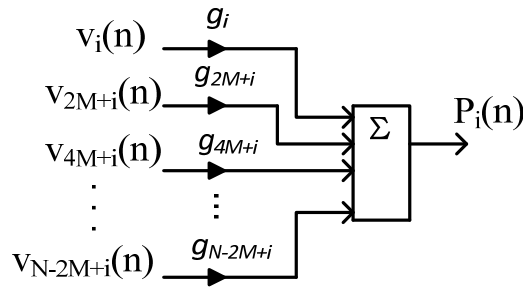
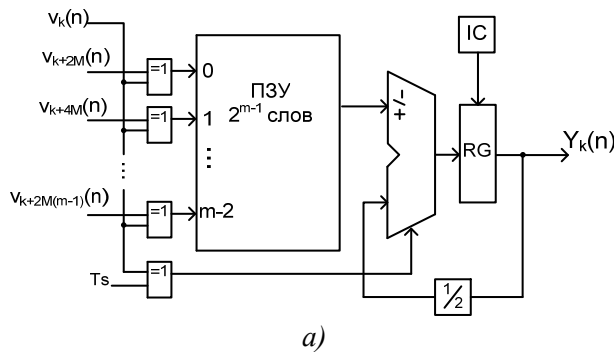


Рис. 5. Схема вычисления i -го полифазного компонента



```

1: for k = 0, 1, ..., 2M - 1 do
2:   R1 = 0;
3:   for i = 0, 1, ..., (N/2M) - 1 do
4:     R2 = v[k + 2Mi];
5:     R3 = g[k + 2Mi];
6:     R1 = R2 × R3 + R1;
7:   end for
8:   P[k] = R1;
9: end for

```

Рис. 6. Полифазный компонент: а) аппаратная реализация; б) программная реализация

5. Среда проектирования сопроцессора банка фильтров

Ранее упоминалось, что на этапе разработки сопроцессора для слуховых аппаратов желательно иметь среду проектирования банка фильтров. Для этого необходимо объединить в комплекс все программы по расчету банка фильтров и генерированию VHDL-описания сопроцессора (рис. 7).

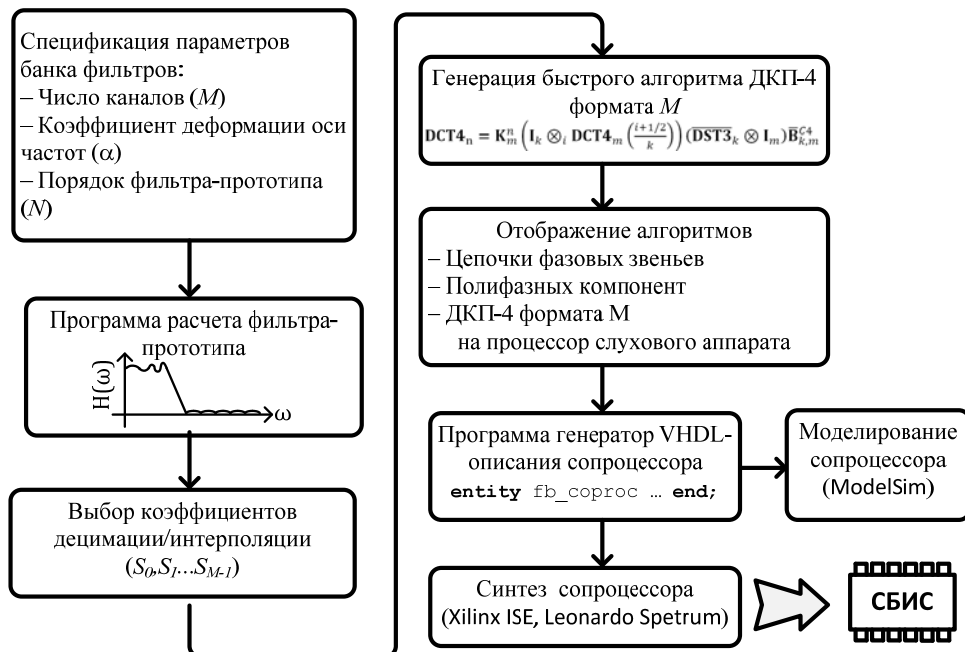


Рис. 7. Этапы проектирования сопроцессора банка фильтров

После того, как выбран способ реализации блоков банка фильтров процесс проектирования сопроцессора банка фильтров может быть полностью автоматизирован. Этапы проектирования показаны на рис.7.

Заключение

При рассмотрении вопросов отображения алгоритмов неравнополосных банков фильтров на процессоры для слуховых аппаратов акцент делался на поиск эффективных алгоритмов и их реализаций на спецпроцессоре для слуховых аппаратов. В частности приведен способ получения быстрого алгоритма ДКП-4, рассмотрены аппаратный и программный подход к реализации цепочки ФЗ и блока полифазных компонент.

Список литературы

1. WOLA filter bank design requirements in hearing aids / M. Rosa-Zurera, [et al.] // Pattern recognition and information processing: proc. of the 10-th intern. conf. – Minsk, Belarus, 2009. – P. 215–218.
2. Bauml, R.W. Uniform polyphase filter banks for use in hearing aids: design and constraints / R.W. Bauml, W. Sorgel // European signal proc. conf. proceedings of 16-th inter. conf., Switzerland, Lausanne, August, 2008. – Lausanne, 2008.
3. An ultra low-power programmable DSP system for hearing aids and other audio application / T. Schneider, [et al.] // Inter. conf. on Signal Processing Application & Technology (ICSPAT): proc. of inter. conf., USA, Orlando, 1999. – Orlando, 1999 – P. 1–5.
4. Parfieniuk, M. Tunable non-uniform filter bank mixing cosine modulation with perceptual frequency warping by allpass transformation / M. Parfieniuk, A. Petrovsky // Automatic Control and Computer Sciences. – 2004. – Vol. 38, № 4. – P. 44–52.
5. Lin, Y.P. A Kaiser window approach for the design of prototype filters of cosine modulated filterbanks / Y.P. Lin, P.P. Vaidyanathan // IEEE Signal Processing Letters. – 1998. – Vol. 5, № 6. – P. 132–134.
6. Furtado, M.B. Numerically efficient optimal design of cosine-modulated filter banks with peak-constrained least-squares behavior / M.B. Furtado, P.S. Diniz, S.L. Netto // IEEE transactions on circuits and systems. – 2006. – Vol. 52, № 3, – P. 597–608.
7. Koilpillai, R.D. Cosine-modulated FIR filter banks satisfying perfect reconstruction / R. D. Koilpillai, P.P. Vaidyanathan // IEEE Trans. On Signal Processing. – 1992. – Vol. 4, № 4. – P. 770–783.
8. Parfieniuk, M. Simple rule of selection of subsampling ratios for warped filter banks / M. Parfieniuk, A.A. Petrovsky // Изв. Белорус. инженерной академии. – 2003. – № 1. – С. 130–134.
9. Püschel, M. Algebraic Signal Processing Theory: Cooley-Tukey Type Algorithms for DCTs and DSTs / M. Püschel and J.M.F. Moura // IEEE Transactions on Signal Processing. – 2008. – Vol. 56, № 4. – P. 1502–1521.
10. Вашкевич, М.И. Применение распределенной арифметики при аппаратной реализации неравнополосных банков фильтров / М.И. Вашкевич, А.А. Петровский // Информационные системы и технологии : материалы 5 Междунар. конф.-форума, Минск, 16-17 нояб. 2009 г. в 2 ч. / БГУИР. – Минск, 2009. – Ч. 2. – С. 198–201.