

Алгоритм синтеза конструкции устройств цифровой обработки сигналов.

Репнева А.И.

Аннотация

В ходе работы показаны основные сложности, возникающие на системном этапе проектирования устройств цифровой обработки сигнала (УЦОС).

Выявлена необходимость в повышении качества проектирования, добиться которого можно путем создания алгоритма позволяющего осуществить формализацию процесса выбора оптимального КТВ.

В ходе работы сформулированы требования к множествам исходных компонент УЦОС, произведен выбор этих компонент, произведен синтез конструктивно-технологических вариантов изделия, а также выбор оптимального КТВ по комплексному показателю качества.

Ключевые слова:

цифровые устройства обработки сигналов; конструкторско-технологический вариант; системный этап проектирования; компоненты; алгоритм; синтез; декомпозиция; метод последовательных отсечений.

Введение

Рассматриваемый алгоритм синтеза конструкции устройств цифровой обработки сигналов (УЦОС) решает задачу выбора конструктивно-технологического варианта (КТВ) реализации устройства на системном этапе проектирования.

Как известно результатом системного проектирования является структурный вариант его реализации[1], т.е. известны структурные элементы, требования к ним и связи между ними. Вместе с тем один и тот же структурный вариант может иметь различные конструктивно-технологические воплощения. Эти реализации отличаются друг от друга

массогабаритными показателями, стоимостью, надежностью и т.п., а следовательно и эффективностью реализации УЦОС.

Поэтому целью данной статьи является создание алгоритма обеспечивающего реализацию структурного варианта УЦОС с максимальной эффективностью.

Решение задачи

Конструктивно - технологический вариант представляет собой конкретную реализацию устройства цифровой обработки сигнала, заданного алгоритмом функционирования на множествах исходных компонент: микропроцессоров, АЦП, ПЛИС, интерфейсные устройства, множество монтажных плат и т.д.

Конструктивно-технологический вариант можно представить в следующем виде:

$$КТВ_i, i \in \mathbf{N} \rightarrow \{m_i \subset \mathbf{M}, a_i \subset \mathbf{A}, k_i \subset \mathbf{K}, o_i \subset \mathbf{O}, \dots\}, \quad (1)$$

где \mathbf{M}_m – множество микропроцессоров;

\mathbf{A}_a – множество АЦП;

\mathbf{K}_k – множество конструктивных вариантов исполнения ПП;

\mathbf{O}_o – множество ОЗУ;

\mathbf{N} – множество КТВ.

Наличие множеств исходных компонент имеет конечный набор элементов, эти множества включают подмножества, которые удовлетворяют требованиям на реализацию алгоритма. Поскольку мощность подмножеств конечна, то получается многовариантность реализации алгоритма. В общем случае максимальное количество КТВ равно произведению мощности всех этих подмножеств, каждый КТВ будет иметь различную эффективность, под которой понимается отношение вероятности выполнения поставленной задачи к затратам[2]. Для УЦОС выполнение поставленной задачи сводится к обеспечению заданного отношения сигнал/шум, поэтому максимальная эффективность будет обеспечена при минимуме затрат. Поэтому критерием эффективности будет являться затраты.

Постановку задачи можно представить следующим образом $КТВ_o \rightarrow \min Z_i, i \in \mathbf{N}$ при выполнении условий $P \leq P_{доп.}, T \leq T_{доп.}, \lambda \leq \lambda_{доп.}, \Delta t \leq \Delta t_{доп.}$.

Данная задача относится к классу задач дискретной оптимизации[3]. Известны следующие методы решения подобных задач: метод ветвей и границ, метод последовательного анализа вариантов, метод локальной оптимизации, метод случайного поиска, метод последовательных отсечений.

Однако, основным недостатком метода ветвей и границ является экспоненциальный рост количества вычислений при увеличении размерности задачи. Более того необходимо найти допустимое решение X_k^* обладающее свойством $f(x_k^*) = f^*(h_k)$ рекордности. Рекорд характеризует приближение к оптимальному решению, для его нахождения необходимо использовать приближенные методы решения задачи [3]. В случае если не удастся найти ни одного допустимого решения, то полагаем, что $f^*(h_0) = +\infty$. Такое предположение также существенно увеличивает количество необходимых вычислений. Исходя из выше сказанного, данный метод не может быть использован для выбора оптимального КТВ.

Метод локальной оптимизации и метод случайного поиска[3,4], основанные на определении окрестности искомого решения в силу построения локального оптимума многократное количество раз, обладают схожими с методом ветвей и границ недостатками. При этом попытка снижения размерности задачи может привести к невозможности нахождения оптимального решения, что не допустимо при решении задачи выбора оптимального КТВ.

Метод последовательного анализа вариантов[3] основан на пошаговом поиске решений и отсеке неоптимальных решений в процессе конструирования.

К недостаткам метода можно отнести получение допустимых частичных решений с последующим тестированием, что не может быть использовано при выборе оптимального варианта КТВ.

При решении задачи дискретной оптимизации методом последовательных отсеков производится рассмотрение всех допустимых вариантов, однако анализ ограничений и отсечение оставшегося множества решений производится последовательно на каждом шаге, таким образом, в результате отсеков создается множество решений небольшой мощности, среди которых достоверно находится оптимальный вариант.

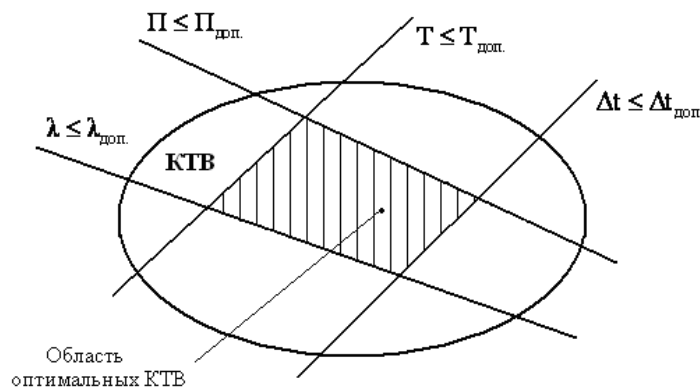


Рисунок 1 – Отсеченная область КТВ

Дальнейший выбор оптимального варианта можно осуществить с помощью метода последовательного перебора, так как мощность отсеченного подмножества небольшая.

Сходимость метода последовательных отсечений вариантов рисунок 1:

$$\begin{aligned}
 & \text{КТВ}_i, i \in \mathbf{N}, \\
 & \text{КТВ}_1 \subset \text{КТВ}_i (\lambda < \lambda_{\text{доп.}}), \\
 & \text{КТВ}_2 \subset \text{КТВ}_1 (\Pi < \Pi_{\text{доп.}}, \lambda < \lambda_{\text{доп.}}), \\
 & \text{КТВ}_3 \subset \text{КТВ}_2 (T < T_{\text{доп.}}, \Pi < \Pi_{\text{доп.}}, \lambda < \lambda_{\text{доп.}}), \\
 & \text{КТВ}_4 \subset \text{КТВ}_3 (\Delta t < \Delta t_{\text{доп.}}, T < T_{\text{доп.}}, \Pi < \Pi_{\text{доп.}}, \lambda < \lambda_{\text{доп.}}).
 \end{aligned} \tag{2}$$

После первого шага отсечения по ограничению $\lambda \leq \lambda_{\text{доп.}}$ из множества $\text{КТВ}_i, i \in \mathbf{N}$ получаем подмножество КТВ_1 , содержащее в себе варианты удовлетворяющие ограничению по интенсивности отказов. После второго шага отсечения по ограничению $\Pi \leq \Pi_{\text{доп.}}$ из множества КТВ_1 получаем подмножество КТВ_2 , варианты которого обеспечивают допустимые потери и удовлетворяют ограничению по интенсивности отказов. После третьего шага отсечения по ограничению $T \leq T_{\text{доп.}}$ из множества КТВ_2 получаем подмножество КТВ_3 , варианты которого обеспечивают допустимое время выполнения программы, допустимые потери и удовлетворяют ограничению по интенсивности отказов. После четвертого шага отсечения по ограничению $\Delta t \leq \Delta t_{\text{доп.}}$ из множества КТВ_3 получаем подмножество КТВ_4 , варианты которого обеспечивают нормальный тепловой режим и удовлетворяю всем ограничениям рассмотренным выше.

В итоге получаем множество КТВ_4 , содержащее в себе оптимальные варианты, которое является подмножеством исходных вариантов $\text{КТВ}_i, i \in \mathbf{N}$ и по мощности меньше исходных вариантов, поэтому метод последовательных отсечений позволяет определить окрестность решения поставленной задачи. И дальнейшее решение задачи можно выполнить с помощью метода полного перебора.

Алгоритм решения поставленной задачи представлен на рисунке 2. Он описывает все этапы разработки КТВ устройства ЦОС, начиная от задания требований к устройству до выбора оптимального варианта КТВ.

Рассмотрим подробнее представленный алгоритм:

1. Ввод алгоритма.

Ввод алгоритма включает:

- описание модели алгоритма;

- раскраска модели алгоритма;
- верификация модели алгоритма.

Рассмотрим каждую из этих задач подробнее. Как показано в [1] модель алгоритма функционирования ЦУОС представляется иерархической сетью Петри. Описание этой сети представляется в виде таблиц матриц переходов с входными и выходными позициями[1], а также таблицей базовой системы команд микропроцессоров, выбор которой представлен в [1]. Ввод осуществляется вручную .

Задача раскраски модели алгоритма – это идентифицировать взаимно однозначное соответствие между переходами элементарной сети Петри модели алгоритма и типами выполняемых операций[5], которые представлены в таблице 1. Для этого необходимо выбрать вершину, которая является начальной для данного перехода и выбрать из списка базовой системы команд, необходимое нам действие. После присвоения каждому переходу операции, происходит окрашивание перехода в цвет, соответствующей операции.

Таблица – 1

Операции алгоритма в соответствии с цветами сети Петри.

Базовая операция	Цвет
Умножение	Желтый
Сложение	Синий
Чтение	Красный
Запись	Черный
АЦП	Розовый
Вычисление модуля	Темно-синий
БПФ	Зеленый
Сложение с накоплением	Голубой
Деление	Темно-голубой
Сравнение с накоплением	Темно-розовый

Полученная таким образом описание модели алгоритма является адекватной исходному алгоритму[1]. После представления сети, стоит задача верификации, целью которой является проверка работоспособности модели, т.е. проверка начальной маркировки сети, проверка заикливания сети, зависание сети и т.д. Порядок проведения верификации модели представлен в [1]. Также с помощью верификации мы получаем частоты повторения всех вершин, которые входят в сеть.

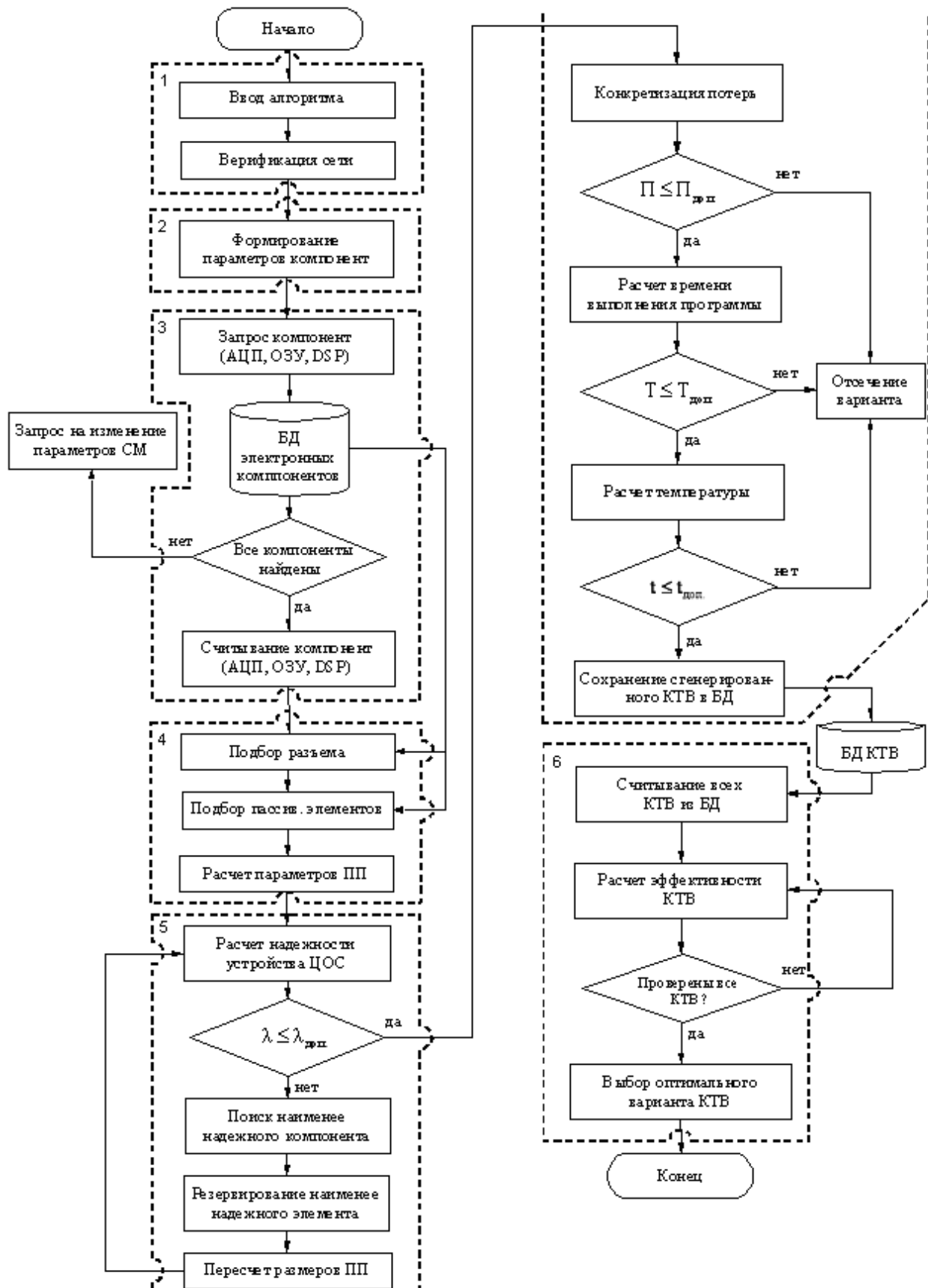


Рисунок 2 – Алгоритм синтеза оптимального КТВ устройства ЦОС

2. Формирование требований к компонентам.

Целевой задачей является отображение алгоритма на множество компонент, т.е. получение КТВ. Таким образом, решение этой задачи сводится к определению основных параметров компонент исходя из параметров системной модели. Поэтому для решения этой задачи необходимо конкретизировать системную модель, в данной работе предлагается остановиться на уже рассмотренном случае, т.е. на идеализированном основном уравнении радиолокации.

В данном разделе формируются основные параметры компонент: АЦП – l , F_d , θ ; ОЗУ – V , l , t_0 ; DSP – l , T_B . Вышеперечисленные параметры определяются из системной модели при условии обеспечения допустимых потерь $\Pi_{\text{доп}}$. Методика расчета этих компонентов рассмотрена в [6]. Исходя из полученных данных, формируется запрос к базе данных.

3. Выбор компонент.

По рассчитанным параметрам формируется запрос к базе данных по нужному компоненту. Происходит формирование списков компонент АЦЦ, DSP, ОЗУ и др.

Исходные компоненты размещаются либо во внутренней базе данных, которая формируется в процессе эксплуатации нашей системы проектирования и пополняется, либо на внешних базах данных фирм производителей. Поэтому выбор сводится к формированию запроса к базе данных по каждому компоненту. Основой запроса являются параметры, рассчитанные на предыдущем шаге. Запрос к внешним источникам осуществляется следующим образом:

Программный код "<http://www.analog.com/ru/index.html>" программный код;

Программный код "<http://www.sec-online.ru/production/components.html>" программный код;

Программный код "<http://www.transistor.ru/catalog/components>" программный код.

4. Синтез КТВ.

Результатом выбора компонент является подмножества $\{M_O, A_O, O_O\}$ из data sheet. Задача синтеза КТВ из полученных подмножеств, сформировать и выбрать КТВ который удовлетворяет ограничениям, а также определить по параметрам компонентов площадь печатной платы, число слоев платы и расчет печатной платы [6]. Методика расчета этих компонентов рассмотрена в [6].

5. Проверка ограничений.

Отсечение вариантов КТВ по требованиям по потерям, по времени выполнения программы, по анализу перегрева $\Pi \leq \Pi_{\text{доп}}$, $T \leq T_{\text{доп}}$, $\Delta t \leq \Delta t_{\text{доп}}$, а также по интенсивности

отказов $\lambda \leq \lambda_{\text{доп}}$. Однако если при первых двух требования вариант отсекается, то при не выполнении условия по интенсивности отказов, КТВ подлежит коррекции и пересчету.

При расчете надежности выделяются структурные элементы, независимые по потоку отказов, которые соединяются в общую схему с позиции надежности. Отказ любого из компонентов устройства ЦОС ведет к отказу всего изделия.

Расчетная модель такой системы последовательное соединение, показанная на рисунке 3, основным показателем является вероятность безотказной работы $P(t)$.

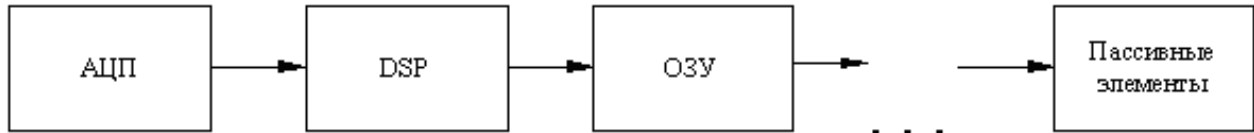


Рисунок 3 – Последовательная модель надежности

Методика расчета надежности сводится к:

1. Классификации компонент на группы: БИС, диоды, резисторы, конденсаторы и т.д.
2. Определение $\lambda_{\text{экс}}$ каждой из групп.
3. Определение $\lambda_{\text{эс}}$.
4. Определение вероятности безотказной работы.

Далее в соответствии с методикой расчета представленной в [7] определяем интенсивность отказов. Для повышения надежности устройства ЦОС используем покомпонентное резервирование, принцип которого в резервировании наименее надежных модулей. Если $P < P_{\text{зад}}$, то определяем самый слабый элемент и вводим необходимое резервирование. Если $P \geq P_{\text{зад}}$, то рассчитываем эффективность.

6. Синтез и оптимизация КТВ.

Процесс выбора оптимального КТВ в общем случае сводится к считыванию и рассмотрению всех КТВ из базы данных, после чего производится расчет эффективности, каждого из них. По результатам сравнения эффективности различных КТВ, производится выбор оптимального по комплексному показателю качества.

Выводы

1. Выполнена постановка задачи выбора КТВ на системном этапе проектирования, позволяющая определить наиболее эффективный КТВ реализации устройства при известном структурном варианте.
2. Проведен анализ и выбор метода решения оптимизационной задачи, позволяющей за конечное число шагов найти оптимальный КТВ.

3. Предложен алгоритм синтеза КТВ, позволяющий на системном этапе проектирования проводить анализ эффективности различных КТВ реализации заданной структуры УЦОС.

Библиографический список

1. Репнева А.И., Ушкар М.Н. «Моделирование алгоритмов функционирования цифровой устройств с использованием сетей Петри», Электронный журнал «Труды МАИ», Выпуск № 49, 2011г.

2. Бусленко Н.П., Калашников В.В., Коваленко И.Н. «Лекции по теории сложных систем», Москва, «Советское радио» 1973г.

3. Ковалев М.М. «Дискретная оптимизация (целочисленное программирование)», Минск, изд-во БГУ им. В.И.Ленина, 1977г.

4. Чернышова Г.Д., Каширина И.Л. «Дискретная оптимизация», Воронеж, 2003.

5. Джеймс Питерсон. «Теория сетей Петри и моделирование систем», Москва, Мир 1984г. – 264с.

6. Ушкар М.Н. «Микропроцессорные устройства в радиоэлектронной аппаратуре», Москва, «Радио и связь», 1988г.

7. Епишина Ю.Г., Б.А. Смиренина «Справочник по надежности», Москва, Мир 1969г.

Сведения об авторах

Репнева Анастасия Игоревна, аспирант Московского авиационного института (национального исследовательского университета).
МАИ, Волоколамское шоссе, д.4, Москва, А-80, ГСП-3, 125993
тел.: 89150968795; e-mail: fil_Nastena@mail.ru