

ИССЛЕДОВАНИЕ ЭФФЕКТА КОНВЕЙЕРИЗАЦИИ КОМБИНАЦИОННЫХ СХЕМ

Поттосин Ю. В., Кардаш С. Н., Кириенко Н. А.

Объединённый институт проблем информатики Национальной академии наук Беларуси

Минск, Республика Беларусь

E-mail: gold@newman.bas-net.by

Рассматривается вопрос повышения быстродействия устройства без памяти, преобразующего последовательность дискретных сигналов. Поставлена задача разбиения заданной многоуровневой комбинационной схемы на заданное число каскадов, на выходах которых должны быть расположены регистры для обеспечения конвейерной обработки поступающих сигналов. Для решения этой задачи используется модель, основанная на представлении комбинационной схемы в виде ориентированного графа. В модели учитываются входная емкость, выходное сопротивление и собственные задержки элементов из библиотеки КМОП СБИС. Разработаны алгоритмы и реализующие их программы для расчета задержки комбинационной схемы.

ВВЕДЕНИЕ

Повышению производительности систем обработки информации всегда уделялось большое внимание. Одним из способов повышения производительности является применение структуры конвейерного типа, который имеет еще название «трубопровод» (перевод с английского слова «pipeline») [1]. Подобную структуру образуют несколько независимых процессоров, соединенных между собой последовательно.

При построении систем цифровой обработки сигналов в режиме реального времени широкое распространение получил систолический принцип организации вычислений [2]. Информация в систолическом процессоре распространяется по конвейеру, подобно тому, как пульсирует кровь при сокращении систолы сердца. В данном докладе излагается попытка найти способ повышения быстродействия путем конвейеризации многоуровневой комбинационной схемы, построенной на основе СБИС.

1. ИСПОЛЬЗУЕМЫЙ МЕТОД РЕШЕНИЯ

В многоуровневой схеме устройства задержка складывается из задержек элементов самой длинной цепочки. Пусть на вход комбинационной схемы поступает последовательность r наборов двоичных сигналов. Если T – время задержки схемы, то период смены сигналов не может быть меньше T . Время реакции устройства на данную последовательность в этом случае будет не меньше rT . Разобьем схему на k блоков (C_1, C_2, \dots, C_k) , и если τ_C – время задержки самого медленно действующего блока, то $T \leq k\tau_C$. На выходы каждого блока поставим элементы задержки (триггеры D), пропускающие сигналы с выходов блока по сигналу синхронизации. Этот же сигнал синхронизации определяет период смены сигналов на входе устройства, который должен быть не меньше суммы двух задержек: задержки τ_C и задержки τ_D элемента D ($\tau_{cloc} \geq \tau_C + \tau_D$). Теперь время реакции устрой-

ства на упомянутую последовательность равно $(k + r) \tau_{cloc}$.

Заданную схему требуется разбить на заданное число k блоков, чтобы обеспечить по возможности максимальное быстродействие при описанном конвейерном режиме. В качестве модели схемы используется бесконтурный орграф $G = (V, A)$ Его вершины из множества V представляют логические элементы и входные полюсы схемы, а дуги из множества A показывают направления сигналов от выходов одних элементов к входам других элементов. Каждой вершине v из V приписан вес $\tau(v)$, представляющий задержку соответствующего элемента. Вершины, соответствующие входам схемы, имеют вес, равный нулю.

Для вычисления задержки элемента сети используется модель, рассмотренная в [3], в которой задержка рассчитывается как для положительного, так и отрицательного фронта входного сигнала для каждого входного полюса элемента без учета связей. Для этого воспользуемся формулой [3, с.193]: $(rise, fall) = PROP(rise, fall) + DRIVE(rise, fall) \otimes \Sigma LIN$, где: $PROP(rise, fall)$ – собственная задержка элемента; $DRIVE(rise, fall)$ – выходное сопротивление; LIN – входная емкость стока.

В качестве задержки элемента будем использовать максимальную из рассчитанных по указанной формуле задержек всех входных полюсов для положительного и отрицательного фронтов. Параметры $PROP$, $DRIVE$, LIN могут быть получены из файла `LIBRARY power` описания библиотеки БМК системы `LeonardoSpectrum`. Структура этого файла детально рассмотрена в [3].

Сформируем последовательность каскадов L_1, L_2, \dots, L_m , представляющую собой упорядоченное разбиение множества вершин V орграфа G с таким свойством, что если вершина v принадлежит полуокрестности исхода $N^+(u)$ вершины u , то эти вершины находятся в разных каскадах

и каскад, содержащий вершину u , предшествует в этой последовательности каскаду с вершиной v (не обязательно непосредственно). Если длины путей от входов схемы к ее выходам различны, то такое разбиение не является единственным. Следует выбрать такой вариант разбиения на каскады, чтобы сумма весов всех каскадов была по возможности минимальной. Под весом каскада понимаем максимум весов вершин, принадлежащих данному каскаду. Можно выделить два типа вершин орграфа G . К одному типу отнесем вершины, которые лежат на самых длинных путях в орграфе G . Они строго распределяются по каскадам и не могут менять свое положение. Их назовем неподвижными. Положение в каскадах других вершин, которые назовем подвижными, можно менять в определенных пределах, скажем, от каскада L_1 до каскада L_r , $l \leq r$. Эти пределы устанавливаются с помощью алгоритма, подобного алгоритму топологической сортировки.

Для окончательного распределения вершин по каскадам так, чтобы сумма весов каскадов была по возможности минимальной, предлагается следующий способ. Удалив из орграфа G неподвижные вершины вместе с инцидентными им ребрами, получим орграф H , в каждой компоненте которого выделим вершину с максимальным весом. Эту вершину поместим в один из допустимых для нее каскадов с максимальным весом. Границы положения вершин при этом изменятся, и некоторые вершины из подвижных перейдут в неподвижные. Дальнейшее распределение по каскадам можно вести для каждой компоненты орграфа H описанным выше способом.

Все пути в орграфе приводятся к единой длине путем добавления новых вершин с нулевым весом. Каждому из каскадов соответствует множество значений веса, приписанных вершинам, принадлежащих данному каскаду. Максимальное значение веса в этом каскаде представляет собой задержку прохождения сигнала в этом каскаде. Заданную комбинационную схему надо разбить на заданное число блоков с минимизацией задержки в самом медленно действующем блоке. Каждый блок представляет собой упорядоченное множество каскадов.

II. РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЯ

Предложенный подход реализован в системе синтеза конвейерных логических схем AgCon, разработанной в лаборатории логического проектирования Объединенного института проблем информатики НАН Беларуси [5]. Система позволяет выполнять преобразование комбинационной логической схемы, заданной структурным

VHDL-описанием, в конвейерную структуру. В таблице 1 представлено исследование задержек и энергопотребления ряда схем в зависимости от числа блоков конвейера. Входные последовательности во всех случаях состояли из десяти случайных наборов. Схемы для экспериментов выбирались из множества тестовых примеров [6]. В таблице 1 используются следующие обозначения: k – число блоков схемы, d – задержка схемы, A – средний потребляемый схемой ток в миллиамперах (мА).

Таблица 1 – Исследование задержек и энергопотребления схем

Имя схемы	k	d, нс	A, мА
b2	-	23,98	2,5
	3	7,78	29,28
	6	4,5	54,34
	10	2,64	76,74
	-	21,5	1,16
	5	5,43	29,56
chkn	10	2,55	43,47
	-	14,99	1,12
	3	5,0	15,83
dist	6	2,55	29,20
	-	14,7	0,48
	4	4,97	9,89
m2	9	2,35	21,71

Конвейеризация позволяет значительно повысить быстродействие схемы (до 10 раз), при этом энергопотребление схемы возрастает значительно быстрее (до 40 раз). Варьируя числом блоков разбиения, проектировщик может выбрать оптимальный по быстродействию и энергопотреблению вариант схемного решения.

III. СПИСОК ЛИТЕРАТУРЫ

1. Каган Б.М., Каневский М.М. Цифровые вычислительные машины и системы. М.: Энергия, 1973.
2. Кухарев Г.А., Шмерко В.П., Зайцева Е.Н. Алгоритмы и систолические процессоры для обработки многозначных данных. Минск: Наука і техника, 1990.
3. Поттосин, Ю.В. Повышение быстродействия комбинационной схемы путем конвейеризации / Ю.В. Поттосин, С.Н. Кардаш // Информатика – Минск: Объединенный институт проблем информатики НАН Беларуси, 2013. № 1. – С. 120–128.
4. Бибило, П.Н. VHDL. Эффективное использование при проектировании цифровых схем / П.Н. Бибило, Н.А. Авдеев. – М.: СОЛОН-ПРЕСС, 2006. – 344 с.
5. Бибило, П.Н. Архитектура системы синтеза конвейерных логических схем / П.Н. Бибило, Н.А. Кириенко, В.И. Романов // Информационные технологии и системы 2012 (ИТС 2012): материалы международной научной конференции, БГУИР, Минск. – Минск: БГУИР, 2012. – С. 184 – 185.
6. Berkeley PLA test set [Electronic source]. – Mode of access: <http://www1.cs.columbia.edu/cs4861/sis/espresso-examples/ex/>. – Date of access: 10.10.2011.