

Шифратор термометрического кода в прямой двоичный код для параллельного АЦП в КМОП технологии

М.М. Пилипко, Д.В. Морозов

Санкт-Петербургский политехнический университет Петра Великого, pilipko_mm@spbstu.ru

Аннотация — Шифратор термометрического кода в прямой двоичный код является важной составной частью параллельного аналого-цифрового преобразователя (АЦП). Ошибки в термометрическом коде снижают эффективную разрядность АЦП. Из известных шифраторов термометрического кода схема на основе сумматора обеспечивает наибольшую эффективную разрядность. В работе в качестве альтернативы данному шифратору предлагается схема на основе логических элементов «исключающее ИЛИ» и мажоритарных элементов, реализованных в комплементарной металл-оксид-полупроводник (КМОП) технологии. Моделирование с помощью MATLAB подтвердило работоспособность предложенной схемы шифратора без проигрыша в эффективной разрядности АЦП по сравнению с шифратором на сумматоре. Моделирование в Cadence Virtuoso для 65-нм КМОП-технологии демонстрирует уменьшение времени задержки при использовании предложенной схемы.

Ключевые слова — параллельный аналого-цифровой преобразователь; шифратор; термометрический код; комплементарный металл-оксид-полупроводник.

I. ВВЕДЕНИЕ

Параллельные АЦП обеспечивают высокую скорость преобразования [1–4]. В N -разрядном параллельном АЦП входной сигнал с помощью компараторов сравнивается с $2^N - 1$ уровнями напряжения, и на выходах компараторов формируется термометрический код, который шифратором необходимо преобразовать в прямой двоичный код. При этом в термометрическом коде возникают ошибки, заключающиеся в том, что среди логических единиц возникает логический ноль, либо наоборот. Причинами для таких ошибок могут быть перестановка двух соседних уровней компараторов [1, 6] и разница времен задержки обработки сигналов компараторами [5, 6]. Указанные ошибки снижают эффективную разрядность параллельного АЦП [5–8].

Для реализации шифратора термометрического кода известен ряд схем [5, 9–11]. Все эти схемы одинаково преобразуют правильные входные коды, но по-разному обрабатывают коды с ошибками. Помимо устойчивости к ошибкам, важными критериями при реализации шифратора термометрического кода являются скорость преобразования, занимаемая на кристалле площадь и потребляемая мощность. Наиболее удачным решением среди известных схем является шифратор на основе сумматора [11]. Цель

данной работы – разработка схемы шифратора термометрического кода, которая будет такой же надежной, как схема на сумматоре, но обеспечит повышение быстродействия.

Во втором разделе на основе эталонного шифратора представлена классификация ошибок в термометрическом коде. В третьем разделе приведен обзор известных схем шифраторов. Четвертый раздел посвящен предлагаемой схеме шифратора на основе логических элементов «исключающее ИЛИ» и мажоритарных элементов. Пятый раздел содержит результаты сравнения шифраторов по устойчивости к ошибкам и быстродействию. В последнем разделе сделаны основные выводы.

II. КЛАССИФИКАЦИЯ ОШИБОК В ТЕРМОМЕТРИЧЕСКОМ КОДЕ

При наличии ошибок в термометрическом коде разные схемы шифраторов формируют разные выходные коды. Для сравнения шифраторов и оценки их устойчивости к ошибкам рассмотрим модель эталонного шифратора. Данная модель определяет наиболее вероятную интерпретацию для любой случайной кодовой комбинации на входе.

В N -разрядном параллельном АЦП существует 2^N правильных комбинаций термометрического кода, назовем их масками. Количество логических единиц в маске (от 0 до $2^N - 1$) определяет ее индекс и соответствующий выходной код. Используя функцию «исключающее ИЛИ», выполняем сравнение входного кода со всеми масками параллельно. Расхождения по битам между входным кодом и каждой маской суммируются. Найденная наименьшая сумма может указывать на одну или несколько наиболее близких масок. Правильному входному коду соответствует лишь одна из масок, где сумма сравнения по битам равна нулю, и выходной код равен индексу этой маски. Для входного кода с ошибками наименьшая сумма соответствует ближайшей маске, что является наиболее вероятным решением. Если эта наименьшая сумма соответствует нескольким маскам, то предлагается определять решение как округленное среднее значение индексов этих масок.

На рис. 1 представлен пример работы описанной модели эталонного шифратора для случая $N=3$. Как видно, входной код, указанный в столбце «Вход», содержит ошибку – имеет несколько переходов «логический ноль – логическая единица». В 3-

разрядном случае существует 2^3 правильных кодов – «Масок с индексами», десятичный индекс которых представляет количество логических единиц в коде и численно равен значению выходного прямого двоичного кода. С помощью функции «исключающее ИЛИ» проводим поразрядные сравнения входного кода с каждой из масок. Чем ближе входной код к маске, тем меньше логических единиц присутствует в столбце результатов сравнения. Полученные значения результатов сравнения суммируем по столбцам и определяем наименьшую сумму. В рассмотренном примере наименьшая сумма равна 1 и соответствует маскам с индексами 3 и 5. Ни одна из этих масок не является предпочтительной, поэтому вычисляем среднее значение индексов и округляем. Как видно, маска с индексом 4 не входит в число наиболее вероятных, но представляет компромисс между двумя наиболее вероятными масками, поэтому ее индекс выступает в качестве наиболее вероятного входного кода и определяет десятичное значение выходного прямого двоичного кода.

Начальное состояние								
Вход	Маски с индексами							
	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1	1
1	0	0	0	0	0	1	1	1
0	0	0	0	0	1	1	1	1
1	0	0	0	1	1	1	1	1
1	0	0	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1
Сравнение входного кода со всеми масками с помощью «исключающее ИЛИ»								
	0	0	0	0	0	0	0	1
	0	0	0	0	0	0	1	1
	1	1	1	1	1	0	0	0
	0	0	0	0	1	1	1	1
	1	1	1	0	0	0	0	0
	1	1	0	0	0	0	0	0
	1	0	0	0	0	0	0	0
Суммы по столбцам результатов «исключающее ИЛИ»								
	4	3	2	1	2	1	2	3
Определение наименьших сумм								
				1		1		
Индексы масок для наименьших сумм								
				3		5		
Наиболее вероятный код								
Усреднение индексов				4				

Рис. 1. Модель эталонного шифратора для случая $N=3$

На основе модели эталонного шифратора введем классификацию ошибок. Наименьшая сумма различий между ошибочным входным кодом и правильным кодом может составлять 1, 2 и т.д. Эта сумма определяет, называется ли ошибка «одиночной», «двойной» и т.д. Для одиночной ошибки можно найти «расстояние» от одинокой логической единицы до группы логических единиц термометрического кода или от одинокого логического нуля до группы логических нулей. Некоторые авторы называют это

расстояние «порядком» [6, 7], а в данной работе назовем это «типом» ошибки. Для одиночной ошибки типа 2 и выше целесообразно различать ошибки «а» и «б» в зависимости от того, что ошибочно – логическая единица или логический ноль, соответственно. При этом количество ошибочных кодов составляет 2^N-2 для ошибок типа 1, 2^N-3 для типа 2а и 2б и т.д. Общее количество кодов с одиночными ошибками составляет $(2^N-2)^2$. Наиболее вероятные ошибки возникают вблизи правильной точки перехода термометрического кода «логический ноль – логическая единица». Кодов с двойными ошибками всего существует $(2^N-1)(2^N-2)^2/2$, но только ошибки вида «1010» заслуживают внимания, другие двойные ошибки возникают существенно реже.

В табл. 1 для 3-разрядного АЦП показан пример правильного термометрического кода и ошибочных кодов – одиночные ошибки типов 1, 2а, 2б и наиболее частая двойная ошибка. Кроме этого приведены примеры произвольных входных кодов, имеющих случайные одиночную или двойную ошибки. (Ошибки выделены жирным шрифтом.)

Таблица 1

Примеры ошибочных кодов для 3-разрядного АЦП

Разряды термометрического кода	Правильный код	Одиночные ошибки				Двойные ошибки	
		Тип 1	Тип 2а	Тип 2б	Случайные	Наиболее частые	Случайные
t7	0	0	0	0	0	0	0
t6	0	0	0	0	1	0	1
t5	0	1	1	1	0	1	0
t4	1	0	0	1	0	0	0
t3	1	1	0	0	0	1	1
t2	1	1	1	1	0	0	0
t1	1	1	1	1	0	1	0

Одиночная ошибка типа 1 наиболее вероятна. При этом логический ноль между логическими единицами обычно считается неправильным [7, 8] и заменяется логической единицей, как в приоритетном шифраторе. Однако в статье [5] предлагается подход, который подразумевает использование мажоритарной функции MAJ для каждого входного цифрового значения D_i и двух соседних цифровых значений D_{i-1} и D_{i+1} термометрического кода перед преобразованием в прямой двоичный код:

$$MAJ(D_i) = D_i \times D_{i-1} + D_{i-1} \times D_{i+1} + D_i \times D_{i+1}.$$

Для одиночной ошибки типа 1 такой подход меняет местами логическую единицу и логический ноль, что приводит к правильному коду, показанному во втором столбце табл. 1. В случае ошибки типа 2 (и выше) одинокая логическая единица (ноль) заменяется логическим нулем (единицей).

Аппаратная реализация эталонного шифратора сложна для используемых на практике значений разрядности АЦП (например, $N = 8$). Кроме того, его корректирующие свойства являются избыточными, поскольку наиболее вероятны только определенные типы ошибок [12]. Существующие схемы шифраторов представляют попытки достичь малого времени задержки наряду с приемлемым преобразованием кодов с ошибками. Отметим, что большинство известных схем создают разную нагрузку на выходах разных компараторов. Следовательно, шифратор должен уметь обрабатывать ошибки, в том числе вызванные им самим. При этом практический интерес представляют только одиночные и двойные ошибки.

III. ИЗВЕСТНЫЕ СХЕМЫ ШИФРАТОРОВ

Известны четыре основных типа шифраторов [5, 9–11], которые реализуются на основе: постоянного запоминающего устройства (ПЗУ); дерева логических элементов (ДЛЭ); мультиплексора; сумматора. В шифраторах на ПЗУ и ДЛЭ сначала определяется разряд перехода входного термометрического кода между логическими нулями и логическими единицами, при этом термометрический код преобразуется в унитарный код. Далее для получения каждого разряда выходного прямого двоичного кода к определенным разрядам унитарного кода применяется логическая функция «ИЛИ». При наличии ошибки сразу несколько разрядов унитарного кода принимают уровень логической единицы, что нарушает принцип работы массива ПЗУ или ДЛЭ. Оба эти типа шифраторов работают как приоритетные схемы, поэтому в наихудшем случае ошибка в выходном коде может достигать полной шкалы АЦП.

Шифратор на основе мультиплексора представляет набор деревьев, представленный на рис. 2, а. Дерево для разряда выходного прямого двоичного кода b_i использует в качестве сигналов данных разряды входного термометрического кода t_j , индексы которых нацело делятся на 2^i . Управляющие сигналы мультиплексоров в дереве подаются разрядами термометрического кода, индексы которых делятся на степени двойки от 2^{i+1} до 2^{N-1} . Таким образом, важность разряда входного термометрического кода зависит от делимости его индекса на степень двойки. Разряд с индексом 2^{N-1} напрямую определяет старший разряд выходного прямого двоичного кода. Если этот разряд термометрического кода окажется неверным, то ошибка в выходном прямом двоичном коде может достигать половины полной шкалы АЦП.

Для исправления ошибок в шифраторах используют специальные схемы, например, [8] для одиночных ошибок типа 1 и [7] для типа 2а. При этом в работе [7] для ошибки типа 2а (см. табл. 1) отдельно стоящая логическая единица считается правильной, а два логических нуля между логическими единицами – ошибочными. Такое решение представляется спорным.

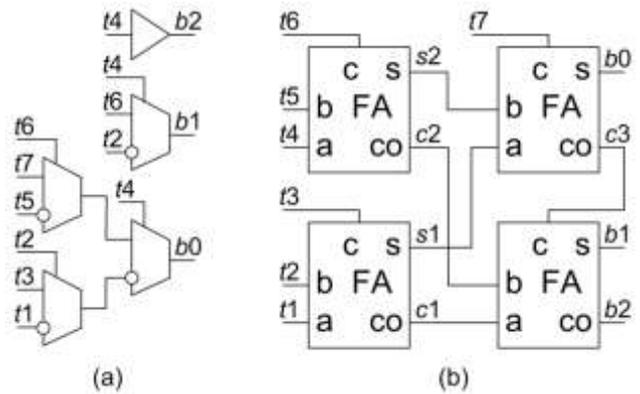


Рис. 2. Примеры трехразрядных шифраторов на основе (а) мультиплексора, (б) сумматора

Шифратор на основе сумматора [1–4] для трехразрядного случая показан на рис. 2, б. В схеме входной термометрический код t_j преобразуется в выходной прямой двоичный код b_i ; деревом полных сумматоров (FA). Обычно используется дерево Уоллеса [11]. Схема подсчитывает количество логических единиц в термометрическом коде, не отдавая приоритет ни одному из разрядов. Эта схема работает медленнее, чем другие решения, но является наиболее надежной. Если входной термометрический код содержит k ошибок, то выходной прямой двоичный код будет отличаться не более чем на k от идеальной интерпретации исправления ошибочного входного термометрического кода. Известен целый ряд реализаций одноразрядного полного сумматора. Предложенный в [13] вариант является наиболее быстродействующим благодаря разделению формирования функций суммы и переноса. Сумма формируется как последовательное соединение двух двухвходовых логических элементов «исключающее ИЛИ» (рис. 3, а) [14], а перенос формируется трехвходовым мажоритарным элементом (рис. 3, б). Обе схемы на рис. 3 имеют на выходе инвертор, который обеспечивает нагрузочную способность. Учитывая изложенное, схему рис. 2, б можно реализовать, как показано на рис. 4, где мажоритарный элемент представлен блоком с обозначением «М».

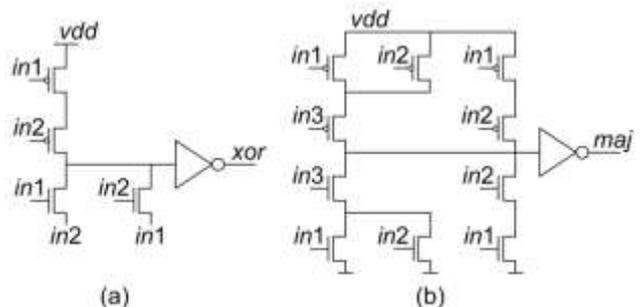


Рис. 3. Схемы (а) логического элемента «исключающее ИЛИ» [14], (б) мажоритарного элемента

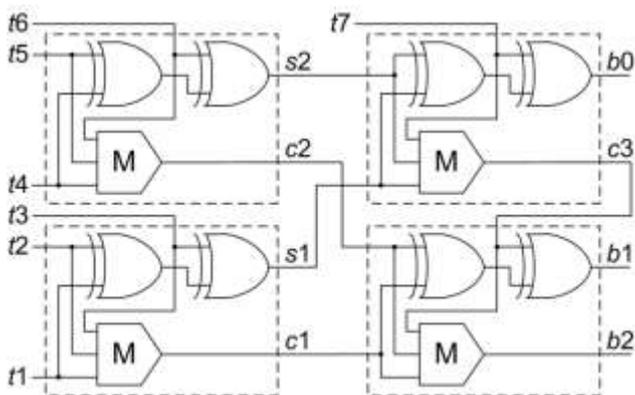


Рис. 4. Трехразрядный шифратор на основе сумматора

IV. ПРЕДЛАГАЕМАЯ СХЕМА ШИФРАТОРА

Предлагаемая схема шифратора строится на основе мажоритарных элементов и элементов «исключающее ИЛИ». При этом целью является уменьшение времени задержки схемы шифратора при устойчивости к одиночным и двойным ошибкам.

Одноразрядный сумматор сам по себе представляет двухразрядный шифратор, который, считая количество логических единиц в трех входных сигналах, формирует выходной двухразрядный код. Однако его работу можно рассмотреть в другом ключе. Младший выходной разряд определяется при помощи дерева из двух логических элементов «исключающее ИЛИ», которое обрабатывает все входные сигналы (все индексы делятся на 2^0). Для определения старшего выходного разряда схема использует мажоритарный элемент для трех входных сигналов – сигнала с индексом, кратным 2^1 , и двух соседних. Учет делимости индексов на степени двойки напоминает шифратор на основе мультиплексора.

Обобщая описанный выше подход, в N -разрядном случае (рис. 5) для получения младшего разряда b_0 выходного прямого двоичного кода дерево логических элементов «исключающее ИЛИ» формируется с использованием всех $2^N - 1$ разрядов входного термометрического кода, поскольку все индексы делятся на 2^0 . Разряд b_1 выходного прямого двоичного кода может быть получен с помощью дерева логических элементов «исключающее ИЛИ» для разрядов входного термометрического кода, индексы которых делятся на 2^1 . Однако эти разряды могут содержать ошибки. Поэтому сначала используем мажоритарный элемент для каждого разряда входного термометрического кода, индекс которого делится на 2^1 , и двух соседних. Для этого потребуется массив мажоритарных элементов, который формирует $2^{N-1} - 1$ скорректированных сигналов, что соответствует количеству входов в $(N-1)$ -разрядном шифраторе. После исправления ошибок в разрядах входного термометрического кода разряд b_1 выходного прямого двоичного кода формируется с использованием дерева логических элементов «исключающее ИЛИ». Далее, для каждого исправленного разряда входного термометрического кода, индекс которого делится на

2^2 , и двух соседних разрядов используется массив мажоритарных элементов, который формирует $2^{N-2} - 1$ сигналов, как в $(N-2)$ -разрядном шифраторе. Процедура повторяется итеративно до тех пор, пока не будут определены все N разрядов выходного прямого двоичного кода. Как видно из схемы на рис. 5, чем больший вес имеет разряд выходного прямого двоичного кода, тем больше этапов исправления ошибок применяется для его определения.

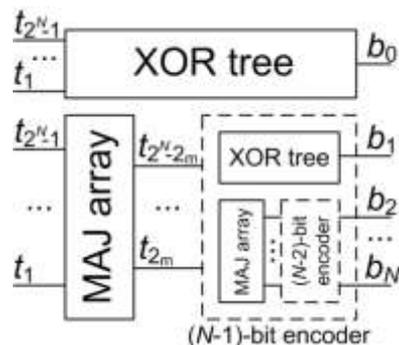


Рис. 5. Предлагаемая структура шифратора

Пример предлагаемой трехразрядной схемы шифратора показан на рис. 6. Дерево логических элементов «исключающее ИЛИ» использует все разряды входного термометрического кода t_1-t_7 для формирования младшего разряда b_0 выходного прямого двоичного кода. В то же время каждый разряд входного термометрического кода с индексами, кратными 2^1 (т.е. t_2, t_4, t_6), и соседние с ними разряды подаются на мажоритарные элементы для коррекции ошибок. Полученные сигналы (t_{2m}, t_{4m}, t_{6m}) используются в двухразрядном шифраторе, где дерево из двух элементов «исключающее ИЛИ» определяет разряд b_1 выходного прямого двоичного кода, а разряд b_2 – выходной сигнал мажоритарного элемента для t_{4m} (индекс делится на 2^2) и соседних t_{2m} и t_{6m} .

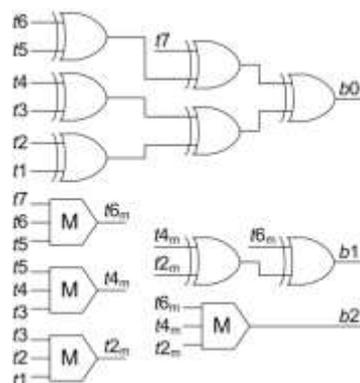


Рис. 6. Пример трехразрядной предлагаемой схемы шифратора

В схеме на рис. 6 используется такое же количество мажоритарных элементов и элементов «исключающее ИЛИ», что и в схеме на рис. 4, в то время как самый длинный путь прохождения сигнала от входа до выхода короче, поскольку здесь используются

отдельные деревья (как в шифраторе на основе мультиплексора) вместо одного дерева сумматоров.

V. СРАВНЕНИЕ ШИФРАТОРОВ

С целью оценки влияния ошибок, шифраторы промоделированы в MATLAB. Кодом имитировалась логическая функция и структура для каждого шифратора. Для тестирования применялись наборы входных термометрических кодов с одиночными и двойными ошибками, показанными в табл. 1. При этом выходной прямой двоичный код каждого шифратора сравнивался с выходным кодом эталонного шифратора. В табл. 2 приведены результаты для различных пятиразрядных шифраторов. Показаны среднее и максимальное отклонения полученного выходного прямого двоичного кода от идеальной интерпретации ошибочного входного термометрического кода, выраженные в единицах младшего значащего разряда (МЗР). Как видно из табл. 2, в большинстве шифраторов одиночные и двойные ошибки могут вызвать значительное отклонение выходного кода от идеального. В частности, для шифраторов на основе ПЗУ и ДЛЭ отклонение может достигать 31 (полная шкала). Схема коррекции ошибок из [8] по-разному справляется с ошибками типов 2a и 2b, которые равновероятны (поскольку ошибка находится на равном расстоянии от правильного перехода «логический ноль – логическая единица», см. табл. 1). Схемы [15] и [12] ограничивают значение ошибки в наиболее вероятных случаях (ошибка до 5, а не 14-17), но максимум ошибки все же велик. Схема на основе мультиплексора адекватно обрабатывает наиболее частые ошибки, но дает сбой при случайных ошибках. Две последние строки показывают, что по устойчивости к ошибкам предложенная в работе схема в большинстве случаев не хуже, чем схема на основе сумматора. В случае наиболее частой двойной ошибки входной термометрический код будет иметь вид, например, $t7..t1 = 0010101$, т.е. имеет три пары «логический ноль – логическая единица». Эталонный шифратор из раздела II находит три наиболее вероятные маски (0000001, 0000111, 0011111, т.е. 1, 3, 5), и в результате выбирается средняя, равная 3. Шифратор на основе сумматора считает логические единицы, что дает тот же результат, равный 3. Предлагаемая схема шифратора (рис. 6) применяет логическую функцию «исключающее ИЛИ» ко всем входным сигналам, при этом $b0 = 1$. Мажоритарные элементы среди 001, 101, 101 дают $t6m = 0$, $t4m = 1$ и $t2m = 1$. Дерево логических элементов «исключающее ИЛИ», использующее эти данные, формирует $b1 = 0$, а мажоритарный элемент выдает $b2 = 1$. Таким образом, выходной прямой двоичный код равен 5, что является одной из наиболее вероятных масок, но не средней из них. Отклонение составляет 2, что можно считать приемлемым. Как видно, для других двойных ошибок предлагаемая схема шифратора имеет преимущество, которое объясняется следующим. Четное количество ошибок не влияет на значение разряда $b0$ выходного прямого двоичного кода, в то время как данные для получения

следующего разряда являются результатом коррекции мажоритарными элементами, которых нет в схеме шифратора на основе сумматора. Для двойной ошибки в столбце «Случайные» табл. 1 шифратор на основе сумматора просто считает логические единицы и выдает код 2, в то время как предложенная схема шифратора подавляет обе ошибки и выдает правильный код 0.

Шифратор на основе сумматора и предложенная схема шифратора в диапазоне разрядностей N от 3 до 8 были синтезированы в Cadence Genus с использованием библиотеки стандартных ячеек 65-нм КМОП-технологии и протестированы в Cadence Virtuoso на схемном уровне. На рис. 7 выходной сигнал в виде преобразованного кода для $N=3$ показывает все возможные переходы между кодовыми комбинациями. Всего таких переходов $2^N \times (2^N - 1)$. Выходной код меняется каждую наносекунду. Ось времени можно разделить на интервалы длительностью 1 нс, которые затем накладываются друг на друга, как на глазковой диаграмме. При этом хорошо видны все переходы, а самый последний из них определяет наибольшую задержку в шифраторе (рис. 8). Схема на основе сумматора имеет задержку 138 пс, в то время как предложенная схема имеет задержку 108 пс. Описанный подход был применен при моделировании шифраторов с разрядностью от 4 до 8. В табл. 3 представлены количество транзисторов и максимальная задержка для шифратора на основе сумматора и предложенной схемы шифратора. Несмотря на увеличение занимаемой на кристалле площади на 10-12%, повышение быстродействия на 21-45% делает целесообразным использование предлагаемой схемы шифратора.

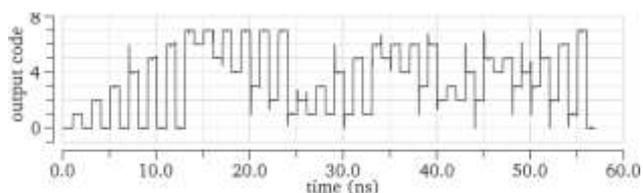


Рис. 7. Трехразрядный выходной сигнал со всеми переходами

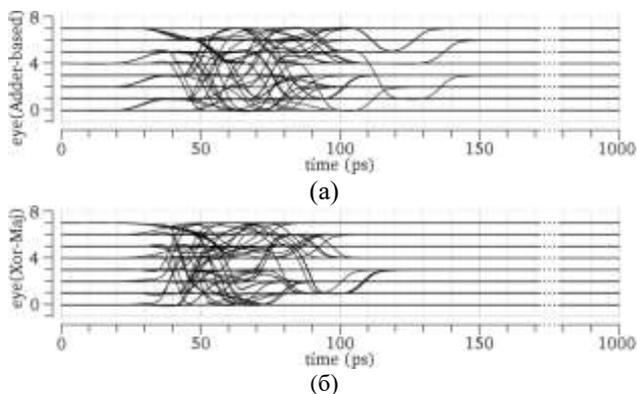


Рис. 8. Время задержки переключений для (а) шифратора на основе сумматора, (б) предложенной схемы шифратора

Таблица 2

Среднее / максимальное отклонение выходного прямого двоичного кода от выходного кода эталонного шифратора, МЗР

Шифратор на основе	Одиночные ошибки				Двойные ошибки	
	Тип 1	Тип 2a	Тип 2b	Случайные	Наиболее частые	Случайные
ПЗУ и ДЛЭ	3.27 / 15	6.03 / 17	3.03 / 14	8.57 / 31	4.85 / 14	12.50 / 31
ПЗУ и ДЛЭ со схемой коррекции ошибок [8]	1 / 1	6.03 / 17	0 / 0	7.24 / 31	4.85 / 14	10.92 / 31
ПЗУ с кодом Грея [15]	0.97 / 2	2 / 5	1.69 / 5	6.43 / 31	2.14 / 5	8.07 / 31
ПЗУ с инверсным кодом [12]	0.97 / 2	1.72 / 5	1.41 / 3	6.43 / 31	1.93 / 5	7.91 / 31
модифицированного ПЗУ [16]	1 / 1	1.48 / 4	0.97 / 3	1.24 / 16	2 / 2	2.25 / 24
мультиплексора	1 / 1	0.72 / 3	0.72 / 3	0.54 / 16	2 / 2	1.11 / 24
мультиплексора со коррекции ошибок [8]	1 / 1	2.28 / 3	0 / 0	0.63 / 17	2 / 2	1.32 / 25
сумматора	0 / 0	1 / 1	1 / 1	0.97 / 1	0 / 0	1.39 / 2
предложенной схемы	0 / 0	1 / 1	1 / 1	0.97 / 1	2 / 2	0.26 / 2

Таблица 3

Сравнение характеристик шифраторов

Шифратор на основе	Разрядность					
	3	4	5	6	7	8
Количество транзисторов						
сумматора	126	308	728	1596	3360	6916
предложенной схемы	128	340	812	1782	3748	7746
различие в %	1,6	10,4	11,5	11,6	11,5	12,0
Время задержки, пс						
сумматора	138	204	292	355	434	531
предложенной схемы	108	134	181	212	245	289
различие в %	21,7	34,3	38,0	40,3	43,5	45,6

Поскольку структура шифратора (рис. 5) требует всего два типа цифровых ячеек, их можно разработать самостоятельно. На рис. 9 показаны топологии шифраторов с использованием специально разработанных ячеек «исключающее ИЛИ» и мажоритарного элемента в 65-нм КМОП технологии. В этом случае размер топологии составляет 11,7 мкм × 26,7 мкм для обеих схем.

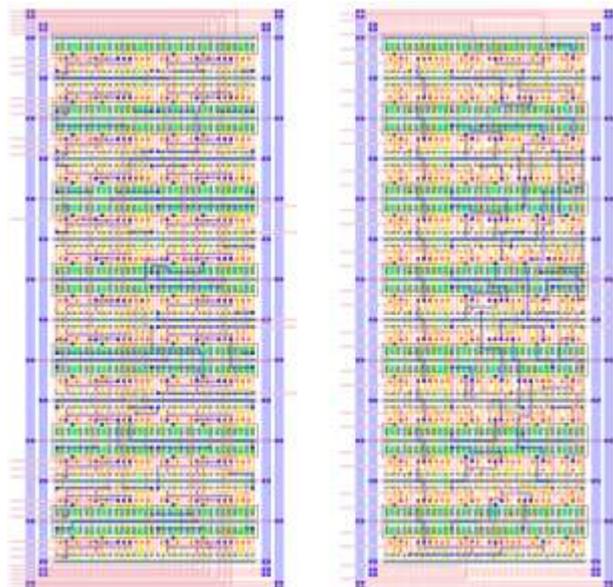


Рис. 9. Топология для шифратора на основе сумматора (слева) и предложенной схемы шифратора (справа) в 5-битном случае

Чтобы удовлетворить требования к скорости преобразования, шифратор часто реализуют в виде нескольких каскадов, разделенных триггерами [3]. Схема с повышенным быстродействием требует меньшего количества каскадов, что означает меньшее количество тактов между выборкой сигнала на входе и получением соответствующего выходного прямого двоичного кода.

VI. ЗАКЛЮЧЕНИЕ

Параллельные АЦП применяются в системах сбора данных и связанных системах. Шифратор термометрического кода в прямой двоичный код является неотъемлемой частью параллельных АЦП. Разработчики чаще всего применяют шифратор на основе сумматора из-за его устойчивости к ошибкам в термометрическом коде. В работе предложена схема шифратора на основе мажоритарных элементов и элементов «исключающее ИЛИ», которая при том же количестве элементов имеет более короткие пути прохождения сигнала от входа до выхода по сравнению со схемой на основе сумматора. Схемы шифраторов были синтезированы и протестированы в 65-нм КМОП-технологии с использованием системы автоматизированного проектирования компании Cadence Design Systems. При разрядности от 4 до 8 предложенная схема имеет на 10–12% большее количество транзисторов, но обладает меньшим на 34–45% временем задержки распространения сигналов. Результаты моделирования показали, что предложенная схема шифратора так же устойчива к одиночным и двойным ошибкам во входном термометрическом коде, как и схема на основе сумматора.

ЛИТЕРАТУРА

- [1] A. Varzaghani et al., A 10.3-GS/s, 6-Bit Flash ADC for 10G Ethernet Applications // *IEEE Journal of Solid-State Circuits*, vol. 48, no. 12, pp. 3038-3048, Dec. 2013, doi: 10.1109/JSSC.2013.2279419.
- [2] J. Pernillo and M. P. Flynn, A 1.5-GS/s Flash ADC With 57.7-dB SFDR and 6.4-Bit ENOB in 90 nm Digital CMOS // *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 58, no. 12, pp. 837-841, Dec. 2011, doi: 10.1109/TCSII.2011.2168020.
- [3] T. Sundström and A. Alvandpour, A 6-bit 2.5-GS/s flash ADC using comparator redundancy for low power in 90 nm CMOS // *Analog Integrated Circuits and Signal Processing*, vol. 64, no. 3, pp. 215-222, Sept. 2010, doi: 10.1007/s10470-009-9391-x.
- [4] P. Mroszczyk, J. Goodacre and V. F. Pavlidis, Energy Efficient Flash ADC With PVT Variability Compensation Through Advanced Body Biasing // *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 66, no. 11, pp. 1775-1779, Nov. 2019, doi: 10.1109/TCSII.2019.2891580.
- [5] C. W. Mangelsdorf, A 400-MHz input flash converter with error correction // *IEEE Journal of Solid-State Circuits*, vol. 25, no. 1, pp. 184-191, Feb. 1990, doi: 10.1109/4.50302.
- [6] K. Uyttenhove, M. S. J. Steyaert, A 1.8-V 6-bit 1.3-GHz flash ADC in 0.25- μ m CMOS // *IEEE Journal of Solid-State Circuits*, vol. 38, no. 7, pp. 1115-1122, July 2003, doi: 10.1109/JSSC.2003.813244.
- [7] N. Agrawal and R. Paily, An improved ROM architecture for bubble error suppression in high speed flash ADCs // 2008 Annual IEEE Student Paper Conference, Aalborg, 2008, pp. 1-5, doi: 10.1109/AISPC.2008.4460547.
- [8] Bui Van Hieu, Seunghyun Beak, Seunghwan Choi, Jongkook Seon and Taikyeong Ted. Jeong, Thermometer-to-binary encoder with bubble error correction (BEC) circuit for Flash Analog-to-Digital Converter (FADC) // *International Conference on Communications and Electronics 2010, Nha Trang, 2010*, pp. 102-106, doi: 10.1109/ICCE.2010.5670690.
- [9] Daegy Lee, Jincheol Yoo, Kyusun Choi and J. Ghaznavi, Fat tree encoder design for ultra-high speed flash A/D converters // *The 2002 45th Midwest Symposium on Circuits and Systems, 2002 (MWSCAS-2002)*, Tulsa, OK, USA, 2002, pp. II-87 – II-90, doi: 10.1109/MWSCAS.2002.1186804.
- [10] J. Lee, B. C. Michael, H. Park and B. Park, A 7b 1GS/s 60mW folding ADC in 65nm CMOS // 2010 International SoC Design Conference, Seoul, 2010, pp. 338-341, doi: 10.1109/SOCDC.2010.5682901.
- [11] F. Kaess, R. Kanan, B. Hochet and M. Declercq, New encoding scheme for high-speed flash ADC's // 1997 IEEE International Symposium on Circuits and Systems (ISCAS), Hong Kong, 1997, pp. 5-8 vol.1, doi: 10.1109/ISCAS.1997.608492.
- [12] S. Padoan, A. Boni, C. Morandi and F. Venturi, A novel coding scheme for the ROM of parallel ADCs, featuring reduced conversion noise in the case of single bubbles in the thermometer code // 1998 IEEE International Conference on Electronics, Circuits and Systems. Surfing the Waves of Science and Technology (Cat. No.98EX196), Lisboa, Portugal, 1998, pp. 271-274 vol.2, doi: 10.1109/ICECS.1998.814878.
- [13] D. V. Morozov and M. M. Pilipko, A circuit implementation of a single-bit CMOS adder // *Russian Microelectronics*, vol. 42, no. 2, pp. 113-118, Feb. 2013, doi: 10.1134/S106373971302008X.
- [14] Jyh-Ming Wang, Sung-Chuan Fang and Wu-Shiung Feng, New efficient designs for XOR and XNOR functions on the transistor level // *IEEE Journal of Solid-State Circuits*, vol. 29, no. 7, pp. 780-786, July 1994, doi: 10.1109/4.303715.
- [15] C. Chan, Y. Zhu, S. Sin, U. Seng-Pan, R. P. Martins and F. Maloberti, A 7.8-mW 5-b 5-GS/s Dual-Edges-Triggered Time-Based Flash ADC // *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, no. 8, pp. 1966-1976, Aug. 2017, doi: 10.1109/TCSI.2017.2682268.
- [16] Yao-Jen Chuang, Hsin-Hung Ou and Bin-Da Liu, A novel bubble tolerant thermometer-to-binary encoder for flash A/D converter // 2005 IEEE VLSI-TSA International Symposium on VLSI Design, Automation and Test, 2005 (VLSI-TSA-DAT), Hsinchu, Taiwan, 2005, pp. 315-318, doi: 10.1109/VDAT.2005.1500084.

A CMOS Thermometer-to-Binary Encoder for a Flash ADC

M.M. Pilipko and D.V. Morozov

Peter the Great St.Petersburg Polytechnic University pilipko_mm@spbstu.ru

Abstract — Flash analog-to-digital converter (ADC) is known for its high conversion speed [1–4]. An important part of a flash ADC is the thermometer-to-binary encoder. Errors occur in the thermometric code, which means that a logical zero appears among logical ones, or vice versa. The reasons for such errors can be the displacement of two adjacent comparator levels [1, 6] or the difference in the delay times of signal processing by the comparators [5, 6]. Errors reduce the effective number of bits of a flash ADC [5–8]. Some circuits are known to implement a thermometer-to-binary encoder [5, 9–11]. An idea of a reference encoder is proposed to compare errors of real circuits. Only certain types of errors are most probable [12]. The adder-based encoder has the best resistance to errors. The adder proposed in [13] is the fastest due to the separate formation of the sum and carry functions. The sum is formed as a series connection of two two-input XOR logical elements [14], and the carry is formed by a three-input majority element. A new encoder is presented as an alternative. It uses the same logic cells but has a different structure. MATLAB simulation compares the new structure with known encoders and error correction circuits [1-16]. The presented encoder structure is as effective in handling errors in the input code as the adder-based one. A 65-nm CMOS technology was applied to synthesize and simulate circuits using Cadence software. In comparison with the adder-based circuit, the presented encoder has a 34-45% lower delay, while it needs 10-12% more transistors for bit widths up to 8.

Keywords — Analog-digital conversion, thermometer code, code conversion, logic gates, error correction, CMOS technology.

REFERENCES

- [1] A. Varzaghani et al., A 10.3-GS/s, 6-Bit Flash ADC for 10G Ethernet Applications // IEEE Journal of Solid-State Circuits, vol. 48, no. 12, pp. 3038-3048, Dec. 2013, doi: 10.1109/JSSC.2013.2279419.
- [2] J. Pernillo and M. P. Flynn, A 1.5-GS/s Flash ADC With 57.7-dB SFDR and 6.4-Bit ENOB in 90 nm Digital CMOS // IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 58, no. 12, pp. 837-841, Dec. 2011, doi: 10.1109/TCSII.2011.2168020.
- [3] T. Sundström and A. Alvandpour, A 6-bit 2.5-GS/s flash ADC using comparator redundancy for low power in 90 nm CMOS // Analog Integrated Circuits and Signal Processing, vol. 64, no. 3, pp. 215–222, Sept. 2010, doi: 10.1007/s10470-009-9391-x.
- [4] P. Mroszczyk, J. Goodacre and V. F. Pavlidis, Energy Efficient Flash ADC With PVT Variability Compensation Through Advanced Body Biasing // IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 66, no. 11, pp. 1775-1779, Nov. 2019, doi: 10.1109/TCSII.2019.2891580.
- [5] C. W. Mangelsdorf, A 400-MHz input flash converter with error correction // IEEE Journal of Solid-State Circuits, vol. 25, no. 1, pp. 184-191, Feb. 1990, doi: 10.1109/4.50302.
- [6] K. Uyttenhove, M. S. J. Steyaert, A 1.8-V 6-bit 1.3-GHz flash ADC in 0.25- μ m CMOS // IEEE Journal of Solid-State Circuits, vol. 38, no. 7, pp. 1115-1122, July 2003, doi: 10.1109/JSSC.2003.813244.
- [7] N. Agrawal and R. Paily, An improved ROM architecture for bubble error suppression in high speed flash ADCs // 2008 Annual IEEE Student Paper Conference, Aalborg, 2008, pp. 1-5, doi: 10.1109/AISPC.2008.4460547.
- [8] Bui Van Hieu, Seunghyun Beak, Seunghwan Choi, Jongkook Seon and Taikyeong Ted. Jeong, Thermometer-to-binary encoder with bubble error correction (BEC) circuit for Flash Analog-to-Digital Converter (FADC) // International Conference on Communications and Electronics 2010, Nha Trang, 2010, pp. 102-106, doi: 10.1109/ICCE.2010.5670690.
- [9] Daegyu Lee, Jincheol Yoo, Kyusun Choi and J. Ghaznavi, Fat tree encoder design for ultra-high speed flash A/D converters // The 2002 45th Midwest Symposium on Circuits and Systems, 2002 (MWSCAS-2002), Tulsa, OK, USA, 2002, pp. II-87 – II-90, doi: 10.1109/MWSCAS.2002.1186804.
- [10] J. Lee, B. C. Michael, H. Park and B. Park, A 7b 1GS/s 60mW folding ADC in 65nm CMOS // 2010 International SoC Design Conference, Seoul, 2010, pp. 338-341, doi: 10.1109/SOCDC.2010.5682901.
- [11] F. Kaess, R. Kanan, B. Hochet and M. Declercq, New encoding scheme for high-speed flash ADC's // 1997 IEEE International Symposium on Circuits and Systems (ISCAS), Hong Kong, 1997, pp. 5-8 vol.1, doi: 10.1109/ISCAS.1997.608492.
- [12] S. Padoan, A. Boni, C. Morandi and F. Venturi, A novel coding scheme for the ROM of parallel ADCs, featuring reduced conversion noise in the case of single bubbles in the thermometer code // 1998 IEEE International Conference on Electronics, Circuits and Systems. Surfing the Waves of Science and Technology (Cat. No.98EX196), Lisboa, Portugal, 1998, pp. 271-274 vol.2, doi: 10.1109/ICECS.1998.814878.
- [13] D. V. Morozov and M. M. Pilipko, A circuit implementation of a single-bit CMOS adder // Russian Microelectronics, vol. 42, no. 2, pp. 113-118, Feb. 2013, doi: 10.1134/S106373971302008X.
- [14] Jyh-Ming Wang, Sung-Chuan Fang and Wu-Shiung Feng, New efficient designs for XOR and XNOR functions on the transistor level // IEEE Journal of Solid-State Circuits, vol. 29, no. 7, pp. 780-786, July 1994, doi: 10.1109/4.303715.
- [15] C. Chan, Y. Zhu, S. Sin, U. Seng-Pan, R. P. Martins and F. Maloberti, A 7.8-mW 5-b 5-GS/s Dual-Edges-Triggered Time-Based Flash ADC // IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 64, no. 8, pp. 1966-1976, Aug. 2017, doi: 10.1109/TCSI.2017.2682268.
- [16] Yao-Jen Chuang, Hsin-Hung Ou and Bin-Da Liu, A novel bubble tolerant thermometer-to-binary encoder for flash A/D converter // 2005 IEEE VLSI-TSA International Symposium on VLSI Design, Automation and Test, 2005 (VLSI-TSA-DAT), Hsinchu, Taiwan, 2005, pp. 315-318, doi: 10.1109/VDAT.2005.1500084.