

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
институт
Вычислительная техника
кафедра

УТВЕРЖДАЮ

Заведующий кафедрой

_____ О. В. Непомнящий

подпись инициалы, фамилия

« _____ » _____ 2020 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.01 – «Информатика и вычислительная техника»

код – наименование направления

Программируемый декодер СОК-ПСС

тема

Руководитель	_____	<u>доцент, канд. тех. наук</u>	<u>А. И. Постников</u>
	подпись, дата	должность, ученая степень	инициалы, фамилия
Выпускник	_____		<u>Д. А. Калинин</u>
	подпись, дата		инициалы, фамилия
Нормоконтролер	_____		<u>А. И. Постников</u>
	подпись, дата		инициалы, фамилия

Красноярск 2020

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
институт
Вычислительная техника
кафедра

УТВЕРЖДАЮ

Заведующий кафедрой

_____ О. В. Непомнящий

подпись инициалы, фамилия

«_____» _____ 2020 г.

ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
в форме бакалаврской работы

Студенту Калинину Денису Александровичу

Группа КИ16-08Б Направление (специальность) 09.03.01 «Информатика и вычислительная техника»

Тема выпускной квалификационной работы «Программируемый декодер СОК-ПСС»

Утверждена приказом по университету № 6623с от 26.05.2020

Руководитель ВКР А. И. Постников, доцент, канд. тех. наук
Исходные данные для ВКР: задание на ВКР, система остаточных классов, программируемые логические интегральные схемы.

Перечень разделов ВКР: анализ задания на ВКР и предметной области, разработка программируемого декодера СОК-ПСС, проверка работоспособности.

Перечень графического материала: Презентация, выполненная с помощью Microsoft Office PowerPoint.

Руководитель ВКР

подпись

А. И. Постников

инициалы и фамилия

Задание принял к исполнению

_____ Д. А. Калинин

подпись, инициалы и фамилия студента

« ____ » _____ 20__ г.

РЕФЕРАТ

Выпускная квалификационная работа по теме «Программируемый декодер СОК-ПСС» содержит 45 страниц текстового документа, 6 таблиц, 29 иллюстраций, 10 формул, 14 использованных источников.

СИСТЕМА ОСТАТОЧНЫХ КЛАССОВ, ДЕКОДИРОВАНИЕ, ПЕРЕДАЧА ИНФОРМАЦИИ КАДРАМИ, ПРОГРАММИРУЕМЫЕ ЛОГИЧЕСКИЕ ИНТЕГРАЛЬНЫЕ СХЕМЫ, МОДЕЛИРОВАНИЕ, VERILOG.

Цель работы: разработка компонента аппаратуры системы передачи и обработки информации (программируемого декодера), осуществляющего преобразование кадра информации, представленной в СОК, в кадр информации, представленной в двоичной ПСС.

Задачи:

- изучение принципов и особенностей СОК;
- выбор элементной базы для создания декодера;
- выбор и обоснование размера и формата кадра информации, обрабатываемой декодером;
- выбор и обоснование формата служебных и информационных слов кадра;
- выбор и обоснование набора модулей СОК, с которыми будет работать разрабатываемый декодер;
- разработка программируемого декодера СОК-ПСС;
- проверка работоспособности декодера.

Актуальность работы обусловлена применением СОК в различных областях с целью ускорения вычислений и повышения надежности передачи данных, а ее новизна заключается в возможности разработанной структуры устройства осуществлять преобразование сжатой информации.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Анализ задания на ВКР и предметной области.....	4
1.1 Цель и задачи выпускной квалификационной работы.....	4
1.2 Система остаточных классов	4
1.3 Принципы передачи информации кадрами.....	8
1.4 Анализ существующих решений.....	9
1.5 Выводы по первому разделу	9
2 Разработка программируемого декодера СОК-ПСС	10
2.1 Выбор элементной базы и средств разработки.....	10
2.2 Выбор структуры кадра.....	11
2.3 Выбор набора модулей СОК.....	13
2.4 Структура программируемого декодера СОК-ПСС	15
2.5 Выводы по второму разделу	32
3 Проверка работоспособности	32
3.1 Функциональное моделирование программирования	32
3.2 Функциональное моделирование декодирования	36
3.3 Временное моделирование	39
3.4 Выводы по третьему разделу.....	41
ЗАКЛЮЧЕНИЕ	42
СПИСОК СОКРАЩЕНИЙ	43
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	44

ВВЕДЕНИЕ

Применение в ЭВМ позиционной двоичной системы счисления оправдано простотой аппаратной реализации функциональных узлов, реализующих операции над двоичными операндами в вычислительной системе.

Однако особенности позиционных систем счисления в определенных случаях затрудняют решение поставленных задач. Примером такой задачи может быть распараллеливание арифметических операций [1]. Ввиду последовательного распространения переноса в позиционных системах счисления решением в данном случае может являться изменение структуры сумматора путем введения в схему дополнительных логических элементов для организации параллельного переноса [2]. Подобный метод не всегда может быть целесообразен.

Другим решением является использование системы счисления со специальными характеристиками переноса, примером которой является непозиционная система остаточных классов [1, 3]. При этом в системах передачи и обработки информации требуется наличие модулей преобразования информации из одного представления в другое. Целью данной выпускной квалификационной работы является разработка аппаратного модуля, реализующего функцию преобразования информации, представленной в системе остаточных классов, в позиционное двоичное представление.

1 Анализ задания на ВКР и предметной области

1.1 Цель и задачи выпускной квалификационной работы

Целью работы является разработка компонента аппаратуры системы передачи и обработки информации (программируемого декодера), осуществляющего преобразование кадра информации, представленной в СОК, в кадр информации, представленной в двоичной позиционной системе счисления.

Задачами, которые необходимо решить в процессе проектирования, являются:

- изучение принципов и особенностей СОК;
- выбор элементной базы для создания декодера;
- выбор и обоснование размера и формата кадра информации, обрабатываемой декодером;
- выбор и обоснование формата служебных и информационных слов кадра;
- выбор и обоснование набора модулей СОК, с которыми будет работать разрабатываемый декодер;
- разработка программируемого декодера СОК-ПСС;
- проверка работоспособности декодера.

1.2 Система остаточных классов

Система остаточных классов представляет собой непозиционную систему счисления. Пусть задан набор целых положительных чисел $p_1, p_2 \dots p_k$, которые в дальнейшем будем называть модулями (основаниями) СОК. Тогда любое положительное число A может быть представлено в виде:

$$A = \alpha_1 B_1 + \alpha_2 B_2 + \dots + \alpha_k B_k, \quad (1)$$

где α_i – наименьший положительный остаток от деления числа A на модуль p_i ;

V_i – базис модуля p_i (целое положительное число).

Число A изображается набором остатков по каждому из модулей, т. е. $A = (\alpha_1, \alpha_2, \dots, \alpha_k)$. При этом, если модули $p_1, p_2 \dots p_k$ являются взаимно-простыми целыми числами, такое изображение целого положительного числа A является единственным [3]. Таким образом, образование остатков α_i производится независимо друг от друга.

Диапазон представления чисел в системе остаточных классов определяется произведением всех оснований системы, т. е.

$$P = p_1 p_2 \dots p_k, \quad (2)$$

где P – число возможных значений в системе с основаниями $p_1, p_2 \dots p_k$ ($0 \leq A \leq P - 1$).

Операции сложения, вычитания и умножения двух чисел $A = (\alpha_1, \alpha_2, \dots, \alpha_k)$ и $B = (\beta_1, \beta_2, \dots, \beta_k)$ определяются следующими формулами

$$A + B = (\alpha_1 + \beta_1, \alpha_2 + \beta_2, \dots, \alpha_k + \beta_k), \quad (3)$$

$$A - B = (\alpha_1 - \beta_1, \alpha_2 - \beta_2, \dots, \alpha_k - \beta_k), \quad (4)$$

$$AB = (\alpha_1 \beta_1, \alpha_2 \beta_2, \dots, \alpha_k \beta_k). \quad (5)$$

При этом необходимо учитывать, что величина $\alpha_i + \beta_i$ ($\alpha_i - \beta_i, \alpha_i \beta_i$) берется по соответствующему модулю p_i , т. е. $0 \leq \alpha_i + \beta_i$ ($\alpha_i - \beta_i, \alpha_i \beta_i$) $\leq p_i - 1$.

Если $\beta_i > \alpha_i$, то вычитание остатков выполняется таким образом, что $p_i + \alpha_i - \beta_i \geq 0$.

Операция деления в системе остаточных классов в общем виде

записывается аналогично формулам (3-5) и может выполняться только в случае деления без остатка [3].

Одно из положительных качеств СОК заключается в сравнительной простоте выполнения арифметических операций, так как основаниями, как правило, являются небольшие числа. Ввиду этого арифметические действия можно описывать в виде таблиц, которые записываются в память. В этом случае выполнение операций сводится к выборке результатов по заданным остаткам операндов [3].

К положительным качествам СОК также можно отнести отсутствие связей между разрядами числа, что позволяет существенно повысить скорость работы вычислительных устройств [3].

Система остаточных классов обладает свойством избыточности, что проявляется в возможности расширения или сокращения набора оснований, не искажающей при этом исходное число [3].

Однако представление чисел в СОК исключает возможность их сравнения. Также затруднена работа с отрицательными и дробными числами, вследствие чего сфера практического использования непозиционной системы остаточных классов ограничена. Несмотря на это, она находит свое применение в таких областях, как цифровая обработка сигналов [1, 4], высокопроизводительные вычисления [5], высоконадежные системы передачи и обработки информации [5, 6].

В основу работы разрабатываемого декодера положен алгоритм преобразования СОК-ПСС, блок-схема которого приведена на рисунке 1.

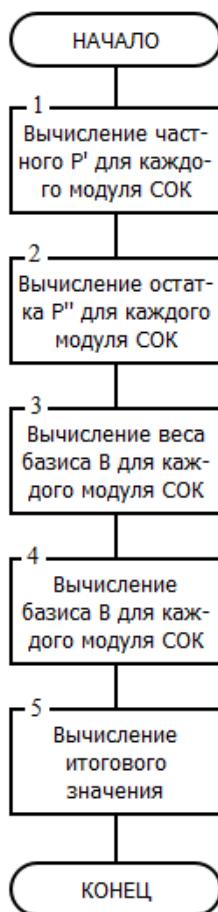


Рисунок 1 – Блок-схема алгоритма перевода числа из СОК в ПСС

В блоке 1 происходит вычисление частного P'_i для каждого модуля СОК p_i по формуле

$$P'_i = P / p_i, \quad (6)$$

где P – диапазон представления чисел в системе остаточных классов, определяемый по формуле (2).

В блоке 2 проводится вычисление остатка P'' для каждого модуля СОК p_i по формуле

$$P''_i = P'_i \bmod p_i. \quad (7)$$

В блоке 3 осуществляется нахождение веса φ_i базиса B_i для каждого модуля СОК p_i из равенства

$$\varphi_i = (kp_i + 1) / P''_i, \quad (8)$$

где φ_i – целое положительное число;

k – целое неотрицательное число.

В блоке 4 происходит вычисление базиса B_i для каждого модуля СОК p_i по формуле

$$B_i = \varphi_i P'_i. \quad (9)$$

В блоке 5 итоговое значение переводимого числа в ПСС вычисляется по формуле

$$A_{\text{ПСС}} = \alpha_1 B_1 + \alpha_2 B_2 + \dots + \alpha_k B_k. \quad (10)$$

Подробное описание формул 6-10 и их математическое обоснование можно найти в [3].

1.3 Принципы передачи информации кадрами

Передача данных по каналу связи может осуществляться как отдельными байтами (битами), так и массивом байтов, называемым кадром. Под кадром понимается структура данных, представляющая собой непрерывную последовательность полей [7].

При передаче информации кадрами с целью исключения ее искажения необходимо поддерживать синхронизацию по кадрам, для достижения которой приемник должен распознавать признаки начала и окончания кадра. Такими признаками являются байты синхронизации (флаги) [8].

Кадры информации, помимо полей данных и флагов, могут содержать дополнительные вспомогательные поля, определяемые протоколом передачи.

1.4 Анализ существующих решений

Рассмотренные ниже решения преобразования информации из системы остаточных классов в двоичную позиционную систему счислений подробно описаны в [5, 9].

Характерной особенностью структуры устройства преобразования чисел из системы остаточных классов в двоичный код, которая рассмотрена в [9], является последовательное вычисление числа по его остаткам с помощью двоичного накапливающего сумматора. Такое решение, с одной стороны, позволяет сократить аппаратные затраты, но, с другой стороны, приводит к увеличению времени преобразования числа.

В [9] также рассмотрена структура устройства, осуществляющего преобразование с применением таблицы соответствия. Такой подход позволяет свести процесс вычисления числа по его остаткам к выборке значений из памяти, однако увеличение количества модулей СОК потребует большего объема памяти.

Преобразователь кодов из СОК в ПСС, являющийся функциональным узлом модулярно-логарифмического процессора [5], функционирует на основе вычисления интервальной характеристики числа, которая вычисляется в мультиоперандном сумматоре. Целью применения мультиоперандного сумматора является ускорение вычислений за счет существенного увеличения аппаратных затрат.

1.5 Выводы по первому разделу

Система остаточных классов как непозиционная система счисления имеет свои достоинства и недостатки. С одной стороны, применение СОК при

обработке информации ускоряет вычисления. Свойство избыточности позволяет говорить о повышении надежности передачи данных, представленных в СОК. С другой стороны, данная система счисления ограничивает использование дробных и отрицательных чисел, что в совокупности с ограниченным диапазоном представления чисел делает ее узкоспециализированным инструментом в области передачи и обработки информации, по сравнению с традиционными позиционными системами счисления.

Тем не менее анализ предметной области показывает, что исследования и разработки во многих перспективных сферах ведутся с принятием во внимание такого узкоспециализированного инструмента, как система остаточных классов, поэтому разработка программируемого декодера, осуществляющего преобразование СОК-ПСС, является актуальной.

При анализе существующих аппаратных решений не выявлено устройств и структур, обеспечивающих обработку кадров сжатых данных, поэтому наличие у разрабатываемого декодера такой функции позволяет говорить о новизне работы.

2 Разработка программируемого декодера СОК-ПСС

2.1 Выбор элементной базы и средств разработки

Выбор специализированной микросхемы (ASIC) в качестве элементной базы позволит максимально эффективно осуществлять декодирование, но изменение структуры кадра или разрядности обрабатываемых слов, может потребовать повторной разработки в полном объеме и, как следствие, много времени и трудозатрат.

С другой стороны, выбор микропроцессора или микроконтроллера даст возможность без значительных затрат модифицировать устройство. Однако в

этом случае будет снижена скорость преобразования информации ввиду программного управления.

Наилучшим решением, с точки зрения скорости выполнения преобразования и возможности внесения изменений, является выбор программируемой логической интегральной схемы в качестве элементной базы разрабатываемого декодера СОК-ПСС. Все дальнейшее проектирование устройства осуществляется с учетом выбора ПЛИС в качестве элементной базы.

Одним из мировых лидеров в области микросхем программируемой логики и средств разработки для них является корпорация Intel. Целевой элементной базой выбрана ПЛИС EP4CE115 семейства Cyclone IV E, достоинством которой является сочетание низкой стоимости и достаточного количества аппаратных ресурсов для разработки и дальнейшей модификации разрабатываемого устройства [10].

Используемой в разработке системой автоматизированного проектирования является Quartus Prime. Для проверки работоспособности используется система HDL-моделирования ModelSim.

2.2 Выбор структуры кадра

Разрабатываемый программируемый декодер СОК-ПСС предназначен для преобразования кадра СОК в кадр ПСС. Структура, представленная на рисунке 2, одинакова как для входных, так и для декодированных кадров.



Рисунок 2 – Структура кадра

Информационное слово (рисунок 3), количество которых в кадре может быть произвольным, содержит выровненные по левому краю (по старшему разряду) остатки от деления на модули СОК, записанные с использованием минимального количества разрядов. Это позволяет экономить память входного буфера.

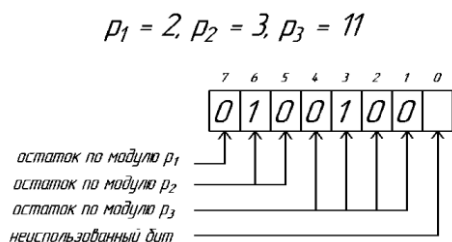


Рисунок 3 – Пример информационного слова

Служебное слово (рисунок 4) служит для идентификации границ кадра и, в отличие от информационного слова, содержит в разрядах хотя бы одного остатка от деления сам модуль СОК.

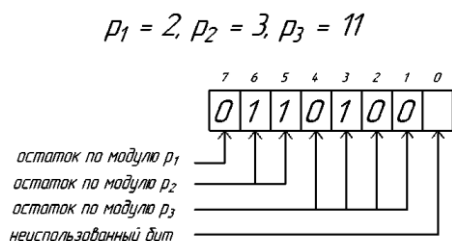


Рисунок 4 – Пример служебного слова

Роль служебных слов заключается в логическом разделении потока данных, поступающих на вход декодера, поэтому служебные слова кадров могут быть как одинаковыми, так и различными.

Разрядность информационной шины разрабатываемого декодера СОК-ПСС, по которой поступают обрабатываемые кадры, ограничена 8 битами.

Выбор такой разрядности обусловлен ориентацией на применение устройства в системах передачи и обработки информации, где для создания числовых данных используются языки высокого уровня, имеющие типы данных заданной разрядности (пример: C, C++) или микроконтроллеры, имеющие регистры и порты заданной разрядности (пример: микроконтроллеры Atmel AVR).

Необходимо учитывать, что данное ограничение не является принципиальным и разрядность информационной шины может быть изменена в зависимости от условий применения декодера, так как его структура является реконфигурируемой из-за выбора элементной базы.

2.3 Выбор набора модулей СОК

Набор модулей СОК представляет собой одновременно используемые при преобразовании целые положительные взаимно-простые числа. Посредством изменения этого набора может осуществляться программирование разрабатываемого декодера.

Критериями выбора набора модулей являются:

- разрядность информационной шины декодера для обрабатываемых кадров;
- диапазон представления чисел в СОК;
- скорость преобразования СОК-ПСС.

Разрядность информационной шины декодера для обрабатываемых кадров определяет максимальное количество разрядов для записи числа в виде остатков.

Диапазон представления чисел в СОК определяется произведением модулей и ограничивается разрядностью информационной шины декодера.

Скорость преобразования СОК-ПСС находится в обратной зависимости с количеством использованных разрядов слова для записи остатков по модулям СОК.

Примеры наборов модулей рассмотрены в таблице 1.

Таблица 1 – Примеры наборов модулей СОК

Набор модулей СОК	Минимальная разрядность слова	Диапазон представления чисел	Количество периодов синхросигнала для преобразования слова
2, 3, 5	6	0 – 29	6
5, 6	6	0 – 29	6
30	5	0 – 29	5
11, 15	8	0 – 164	8
165	8	0 – 164	8
2, 3, 5, 7	9	0 – 209	9
2, 15, 7	8	0 – 209	8
14, 15	8	0 – 209	8

На основании анализа таблицы 1 можно сделать следующие выводы:

- существуют разные по количеству чисел наборы модулей СОК, обладающие одинаковыми характеристиками;
- сокращение чисел в наборе модулей СОК не всегда приводит к увеличению скорости преобразования.

Выбор того или иного набора модулей СОК также зависит от условий задачи, решаемой системой, в состав которого входит разрабатываемый декодер СОК-ПСС, поэтому:

- диапазон декодируемых чисел должен соответствовать требуемому множеству значений;
- количество модулей СОК в наборе должно обеспечивать максимальную скорость декодирования при условии оптимального ускорения вычислительной задачи за счет представления чисел в СОК, так как, чем больше небольших модулей использовано в наборе модулей СОК, тем быстрее могут выполняться задачи в системе обработки информации за счет распараллеливания вычислений разрядов результата.

С учетом представленных критериев память набора модулей СОК разрабатываемого декодера ограничена тремя ячейками памяти. При этом при преобразовании можно использовать и меньшее число модулей СОК.

2.4 Структура программируемого декодера СОК-ПСС

Структурная схема программируемого декодера СОК-ПСС представлена на рисунке 5.

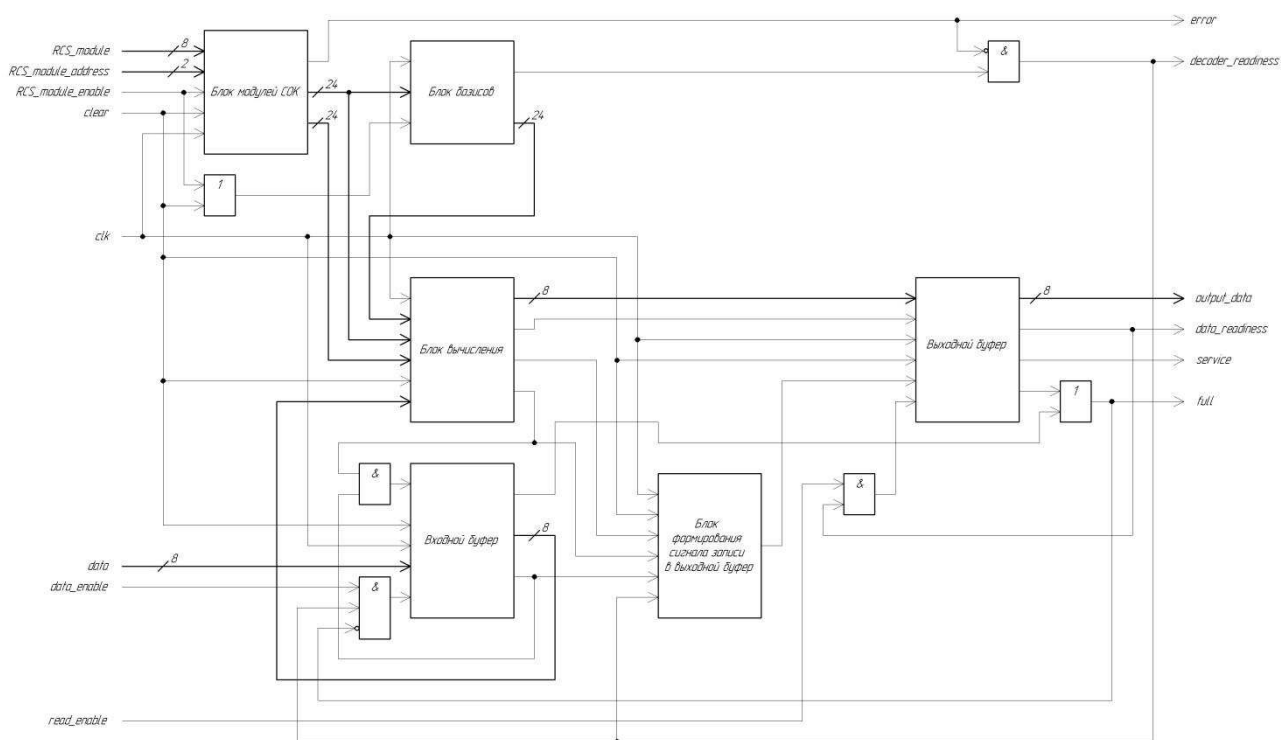


Рисунок 5 – Структурная схема программируемого декодера СОК-ПСС

Описание входов и выходов декодера приведено в таблице 2.

Таблица 2 – Входы и выходы программируемого декодера СОК-ПСС

	Обозначение	Описание
Входы	RCS_module	8-разрядная шина данных для записи модулей СОК в память
	RCS_module_address	2-разрядная шина адреса, по которому модуль СОК с шины RCS_module записывается в память
	RCS_module_enable	Сигнал разрешения записи модуля СОК с шины RCS_module по адресу RCS_module_address
	clear	Сброс декодера, при котором обнуляются адреса чтения и записи во входном и выходном буферах, а также очищается память модулей СОК
	clk	Синхросигнал
	data	8-разрядная шина данных для записи обрабатываемых слов во входной буфер
	data_enable	Сигнал разрешения записи обрабатываемого слова во входной буфер
	read_enable	Сигнал разрешения чтения следующего доступного значения из выходного буфера
Выходы	error	Сигнал об ошибке в наборе модулей СОК
	decoder_readiness	Сигнал о готовности декодера принимать и обрабатывать слова; формируется после вычисления базисов для модулей СОК, записанных в память, при отсутствии ошибки в наборе оснований
	output_data	Выходная 8-разрядная шина данных
	data_readiness	Сигнал о наличии в выходном буфере непрочитанных данных
	service	Сигнал идентификации служебного слова; если сигнал активен, на шине output_data находится служебное слово, иначе на шине output_data находится декодированное информационное слово
	full	Сигнал о переполнении входного и/или выходного буфера

Модули СОК по шине RCS_module записываются в память блока модулей СОК по адресу, установленному на шине RCS_module_address, при наличии разрешающего сигнала RCS_module_enable. В блоке модулей СОК формируется сигнал ошибки набора модулей error.

С выходной шины блока модулей СОК модули поступают в блок базисов, где осуществляется вычисление коэффициентов, необходимых для декодирования. По окончании вычисления декодер считается запрограммированным, и при отсутствии ошибки в наборе модулей СОК формируется выходной сигнал decoder_readiness.

Слова декодируемых кадров поступают по шине data и записываются во входной буфер по нарастающему фронту синхроимпульса при наличии разрешающего сигнала data_enable, готовности декодера и наличии свободного места во входном буфере.

Из входного буфера слова в порядке записи поступают на вход блока вычислений, в котором на основании заданного набора модулей СОК осуществляются вычисления чисел по их остаткам, а также происходит идентификация служебных слов. По завершению цикла преобразования СОК-ПСС и формирования разрешающего запись сигнала в соответствующем блоке декодированное слово записывается в выходной буфер данных, откуда оно может быть прочитано по шине output_data при наличии сигнала read_enable.

Синтезированное из описания на языке Verilog RTL-представление разработанного программируемого декодера СОК-ПСС представлено на рисунке 6.

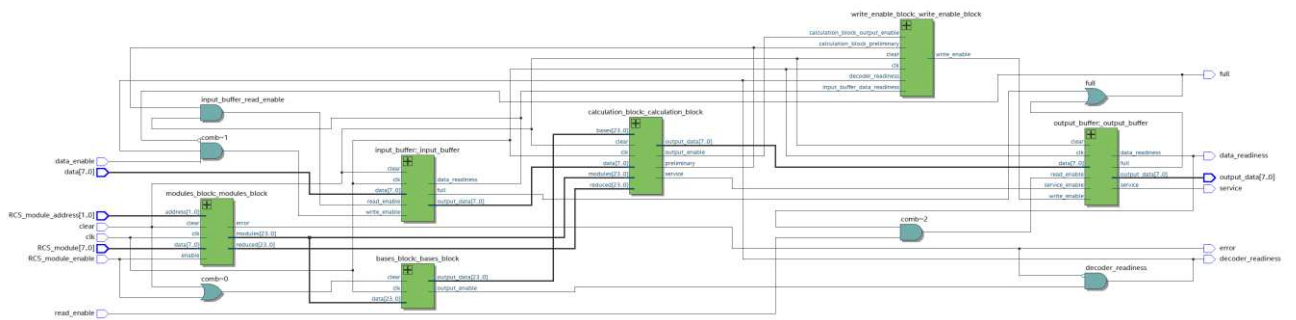


Рисунок 6 – RTL-представление программируемого декодера СОК-ПСС

Программируемый декодер СОК-ПСС имеет следующие функциональные блоки:

- блок модулей СОК (modules_block);
- блок базисов (bases_block);
- входной буфер данных (input_buffer);
- выходной буфер данных (output_buffer);
- блок вычисления (calculation_block);
- блок формирования сигнала записи в выходной буфер (write_enable_block).

Структурная схема блока модулей СОК представлена на рисунке 7.

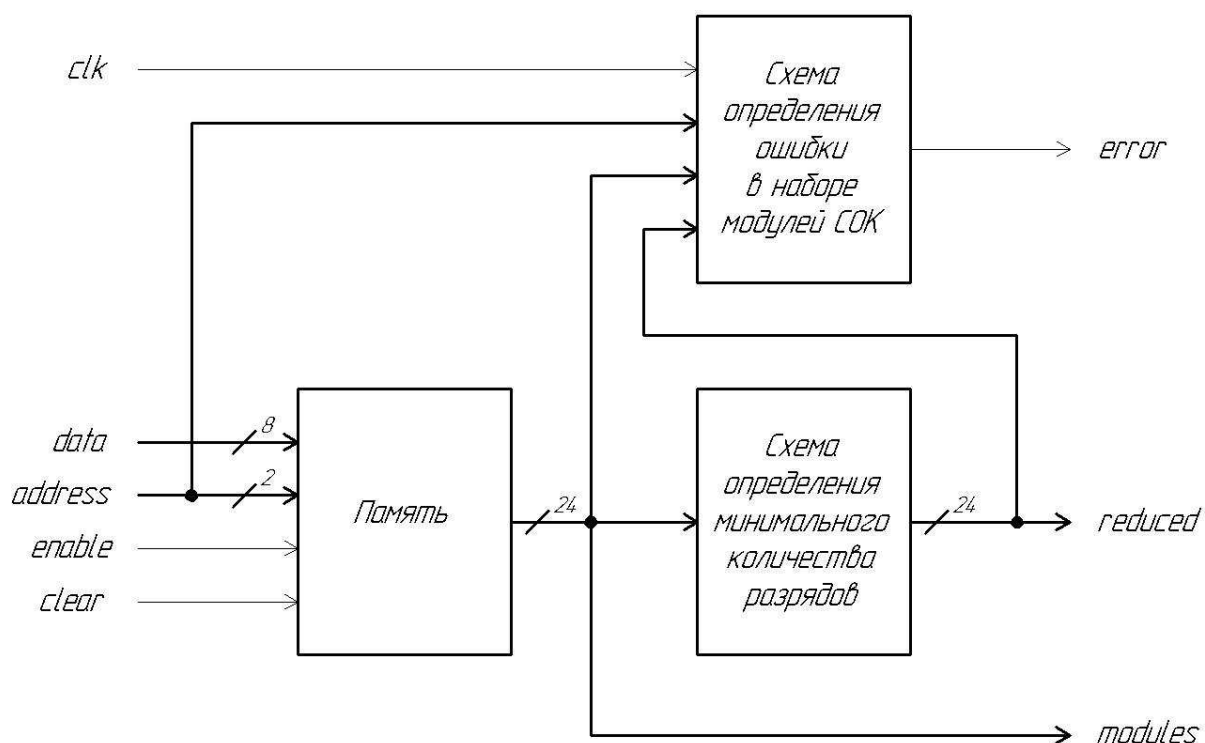


Рисунок 7 – Структурная схема блока модулей СОК

Блок модулей СОК предназначен для хранения набора оснований СОК, в соответствии с которым декодер осуществляет преобразование, а также определения ошибки в наборе модулей и минимального количества разрядов, необходимого для записи числа в виде остатков от деления.

По нарастающему фронту сигнала разрешения enable число с шины data по адресу с шины address записывается в память блока. На выходе памяти из всех трех 8-разрядных чисел, формируется общая 24-разрядная шина modules, содержащая набор модулей СОК. Данные с нее поступают в схемы определения ошибки и минимального количества разрядов, а также в другие блоки декодера.

Синхросигнал clk предназначен для тактирования части схемы определения ошибки.

Выходная 24-разрядная шина данных reduced схемы определения минимального количества разрядов содержит для каждого модуля в наборе

минимальное количество разрядов, с помощью которых можно записать остаток от деления на данный модуль в двоичной позиционной системе счисления. Например, для числа 7 минимальное количество разрядов равно 3, а для числа 2 – 1.

В схеме определения ошибки формируется сигнал о наличии ($error = 1$) или отсутствии ($error = 0$) ошибки в наборе модулей СОК. Набор оснований считается ошибочным, если выполняется хотя бы одно из следующих условий:

- все основания равны нулю;
- среди оснований есть единицы;
- среди не равных нулю оснований есть числа, которые не являются взаимно-простыми;
- сумма минимального количества разрядов для записи остатков от деления на ненулевые основания превышает установленную разрядность слова.

Сигнал ошибки $error$ формируется также при попытке записи модуля по несуществующему адресу.

Синтезированное RTL-представление блока модулей СОК представлено на рисунке 8.

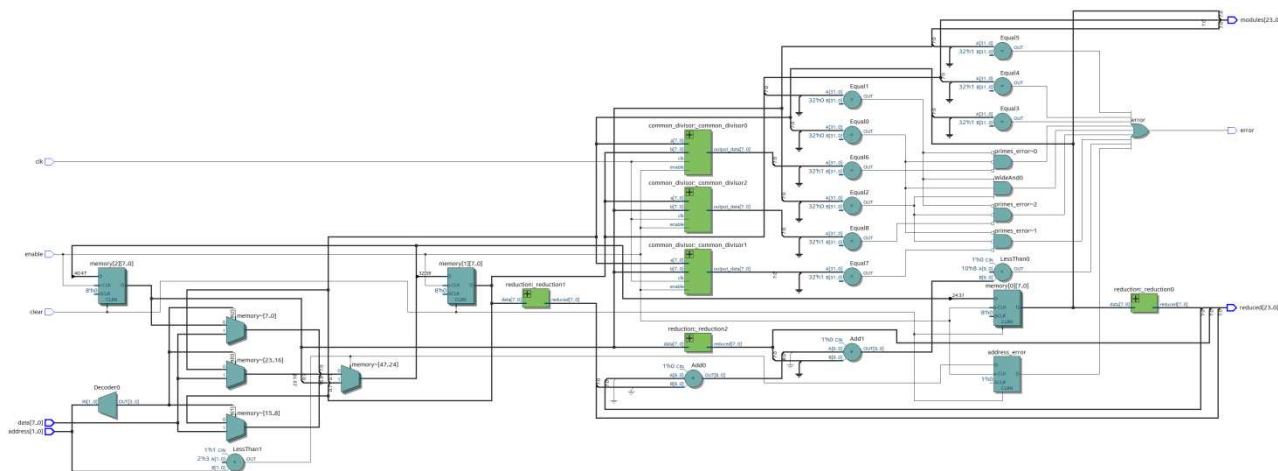


Рисунок 8 – RTL-представление блока модулей СОК

Схема определения минимального количества разрядов включает для каждого модуля СОК отдельную комбинационную схему, реализованную на языке описания аппаратуры Verilog в виде отдельного модуля (reduction). RTL-представление модуля reduction приведено на рисунке 9.

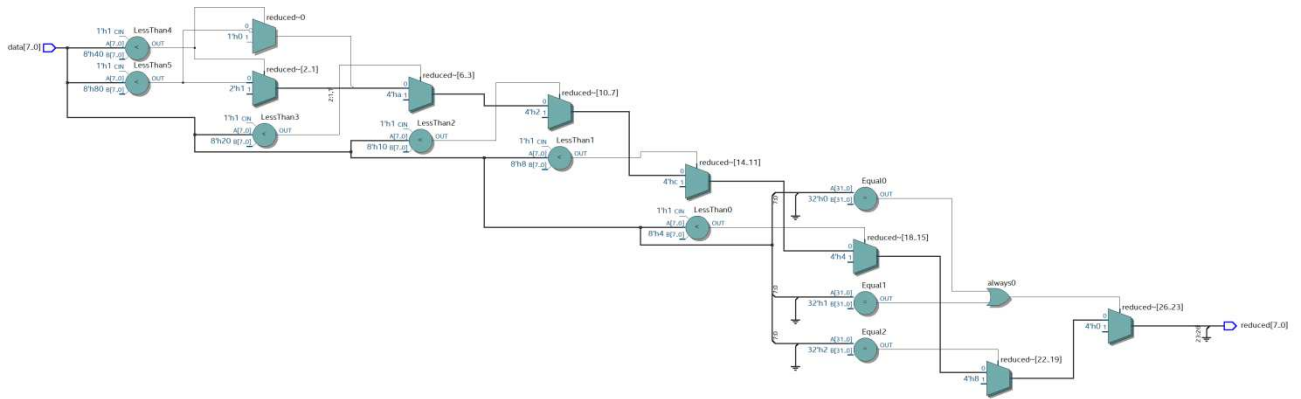


Рисунок 9 – RTL-представление модуля reduction

Схема определения ошибки в наборе модулей СОК включает три идентичные схемы для определения наибольшего общего делителя двух чисел, по которому происходит определение взаимно-простых чисел. Схема определения наибольшего общего делителя реализована на языке описания аппаратуры Verilog в виде отдельного модуля (common_divisor). RTL-представление модуля common_divisor приведено на рисунке 10.

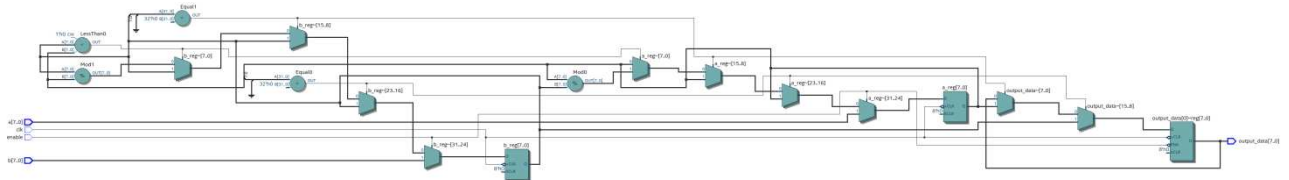


Рисунок 10 – RTL-представление модуля common_divisor

Структурная схема блока базисов представлена на рисунке 11.

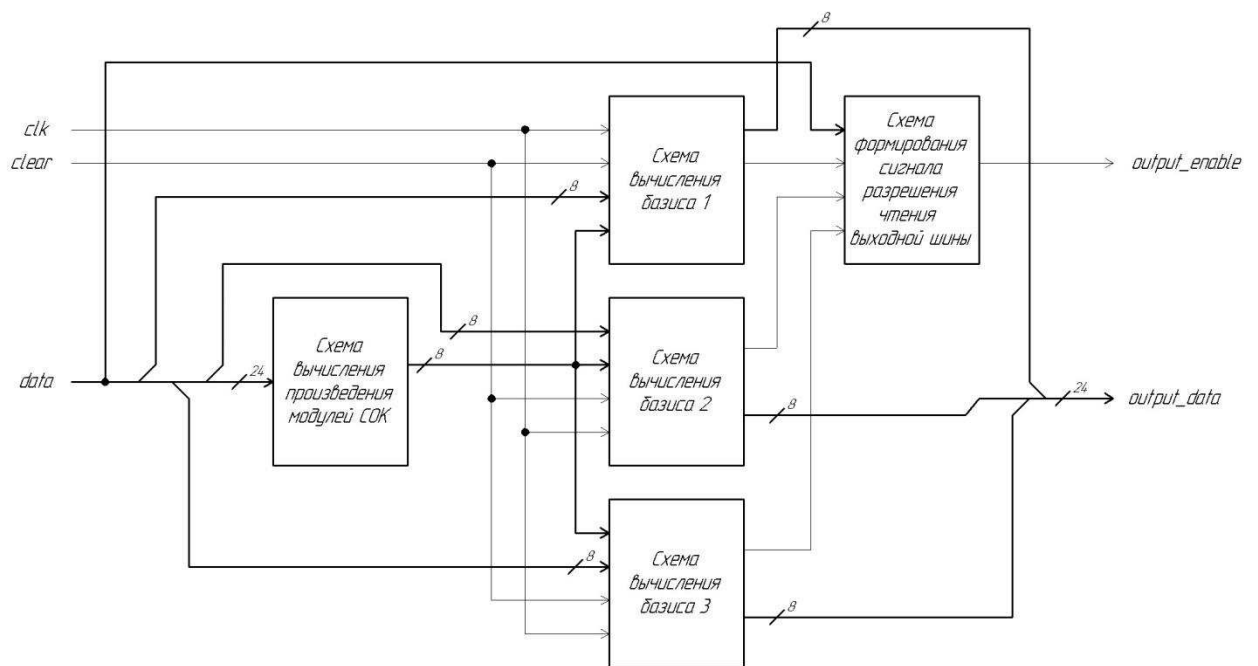


Рисунок 11 – Структурная схема блока базисов

В блоке базисов осуществляется вычисление базиса для каждого модуля СОК в соответствии с алгоритмом перевода числа из СОК в ПСС. После вычисления базисов и при отсутствии ошибки в наборе модулей декодер становится запрограммированным (сигнал декодера `decoder_readiness`).

Базисы модулей СОК, поступающих на входную 24-разрядную шину `data`, вычисляются параллельно друг с другом и объединяются в выходную 24-разрядную шину данных блока `output_data`, которая может быть прочитана другими блоками декодера при наличии разрешающего сигнала `output_enable`.

Схемы вычисления базисов идентичны друг другу. Каждая из них состоит из арифметического блока и счетчика для подбора веса базиса в соответствии с алгоритмом перевода числа из СОК в ПСС. Сигналы `clear` и `clk` предназначены для сброса и тактирования счетчиков соответственно.

Синтезированное RTL-представление блока базисов представлено на рисунке 12.

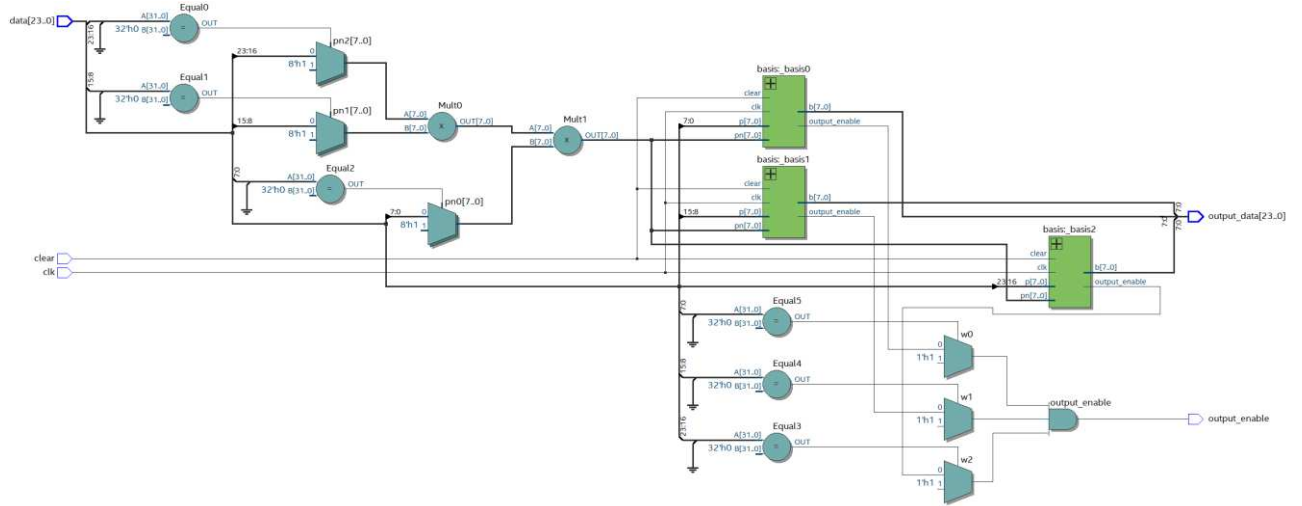


Рисунок 12 – RTL-представление блока базисов

Схема вычисления базиса реализована на языке описания аппаратуры Verilog в виде отдельного модуля (basis). RTL-представление модуля basis приведено на рисунке 13.

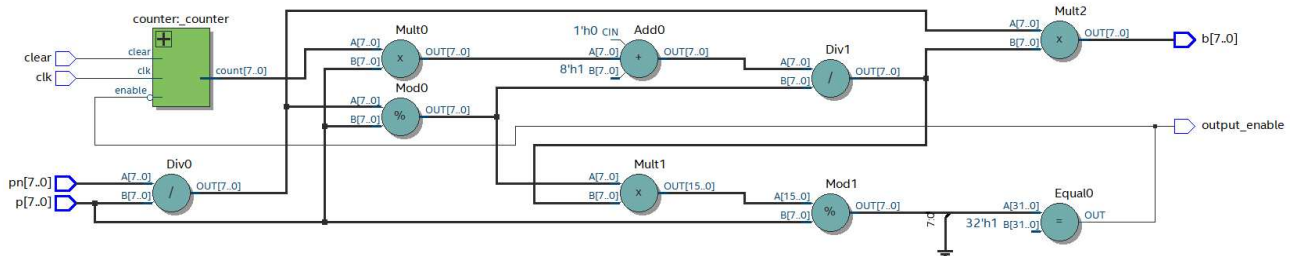


Рисунок 13 – RTL-представление модуля basis

Счетчик на языке описания аппаратуры Verilog реализован как отдельный модуль (counter). RTL-представление модуля счетчика (counter) представлено на рисунке 14.

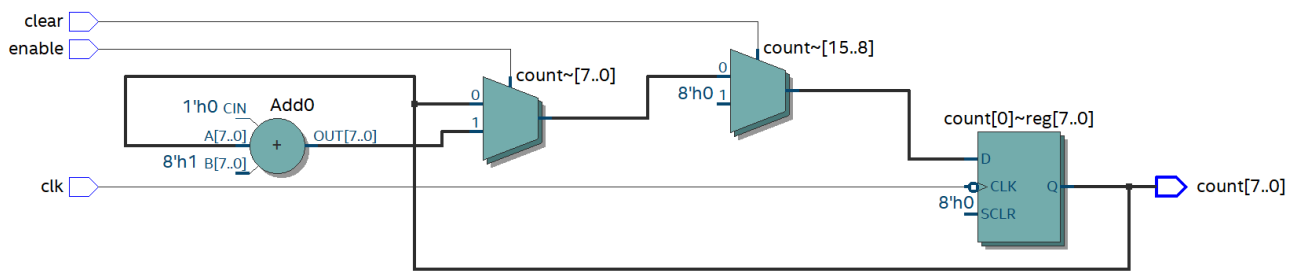


Рисунок 14 – RTL-представление модуля counter

Структурная схема входного буфера данных представлена на рисунке 15.

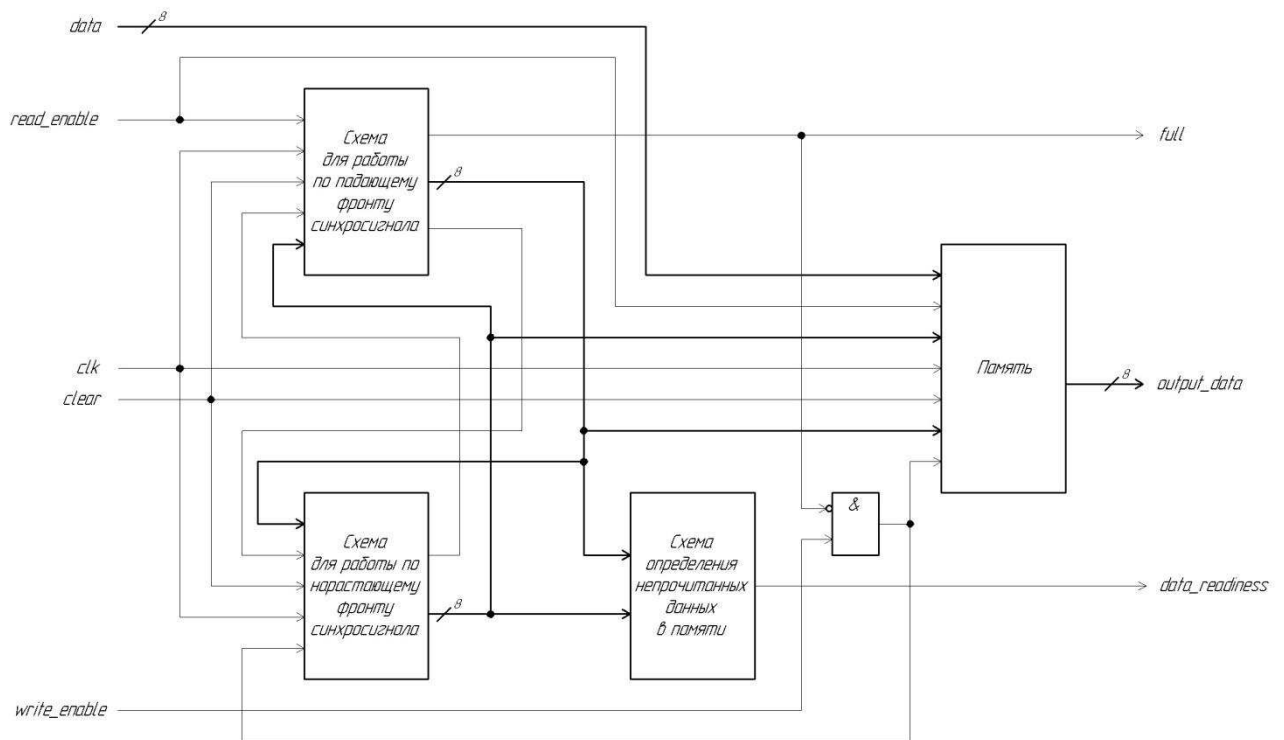


Рисунок 15 – Структурная схема входного буфера данных

Входной буфер данных предназначен для хранения кадров СОК. Буфер представляет собой модуль ОЗУ с отдельными сигналами разрешения чтения и записи, а также отдельными шинами адреса. Проблема чтения во время записи решается за счет осуществления записи по нарастающему фронту синхросигнала, а чтения – по падающему.

Во входном буфере есть схемы, работающие по фронтам синхросигнала. В схеме, работающей по нарастающему фронту синхросигнала, формируются:

- адрес для чтения;
- вспомогательный сигнал для адреса записи.

В схеме, работающей по падающему фронту синхросигнала, формируются:

- адрес для записи;
- вспомогательный сигнал для адреса чтения;
- сигнал переполнения памяти буфера данных.

Для хранения адресов чтения и записи предусмотрены отдельные 10-разрядные регистры, которые позволяют адресовать 1024 ячейки памяти входного буфера. После осуществления чтения/записи значение соответствующего регистра увеличивается. Когда адреса чтения и записи становятся одинаковыми, то есть все значения буфера прочитаны, адреса обнуляются. Также адреса обнуляются при активном сигнале сброса clear.

Выходами буфера являются сигналы о наличии непрочитанных (необработанных) данных (data_readiness) и переполнении (full), а также шина данных (output_data).

RTL-представление входного буфера данных представлено на рисунке 16.

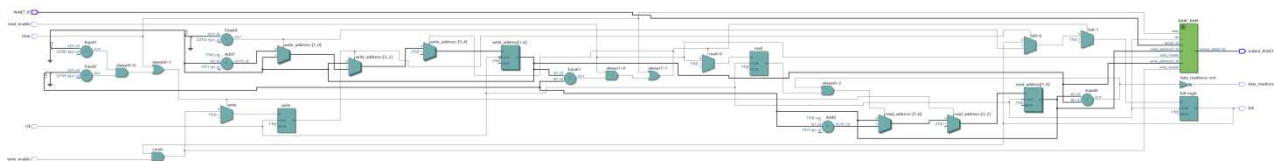


Рисунок 16 – RTL-представление входного буфера данных

Память входного буфера реализована на языке описания аппаратуры Verilog в виде отдельного модуля (RAM). RTL-представление модуля RAM приведено на рисунке 17.

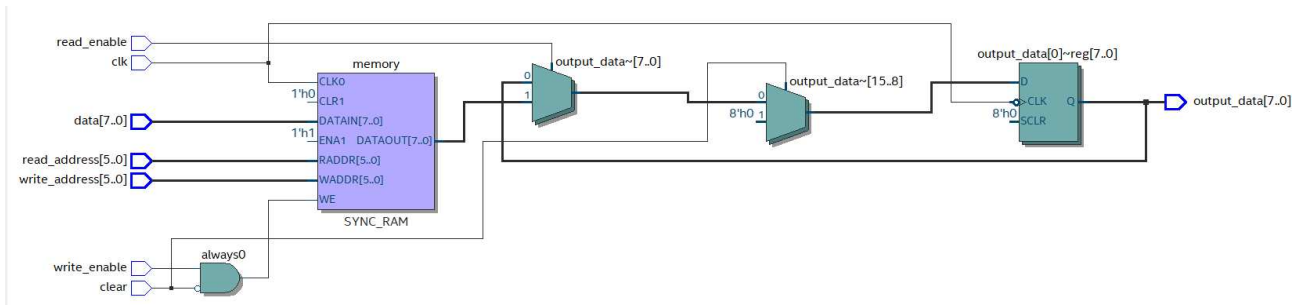


Рисунок 17 – RTL-представление модуля RAM

Синтезированное RTL-представление выходного буфера данных представлено на рисунке 18.

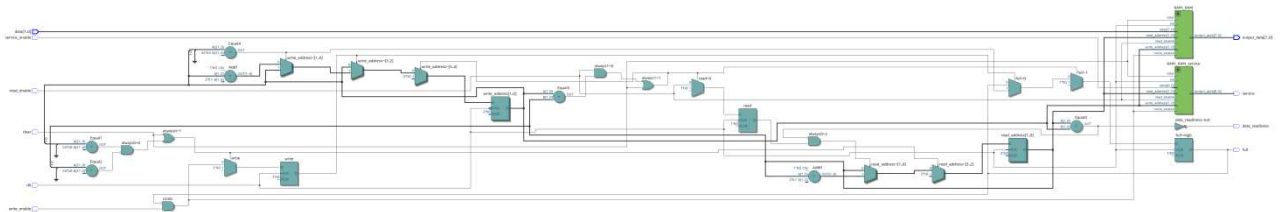


Рисунок 18 – RTL-представление выходного буфера данных

Выходной буфер данных предназначен для хранения декодированных кадров ПСС. Структура и принцип работы аналогичны входному буферу данных, за исключением того, что в выходном буфере имеются дополнительный модуль памяти (модуль RAM, рисунок 17) для хранения массива флагов, идентифицирующих служебные слова, а также дополнительный выход service, сигнализирующий о наличии на выходной шине данных служебного слова.

При записи декодированного слова в память буфера, на его вход service_enable из блока вычислений поступает сигнал-идентификатор служебного слова. Если service_enable = 1, то по тому же адресу, по которому происходит запись декодированного слова, в дополнительный модуль памяти записывается одноразрядное значение 1. Иначе по этому адресу записывается

одноразрядное значение 0. При чтении слов из выходного буфера на выходе service устанавливается значение из дополнительного модуля памяти, соответствующее по адресу декодированному слову на выходной шине output_data.

Структурная схема блока вычисления представлена на рисунке 19.

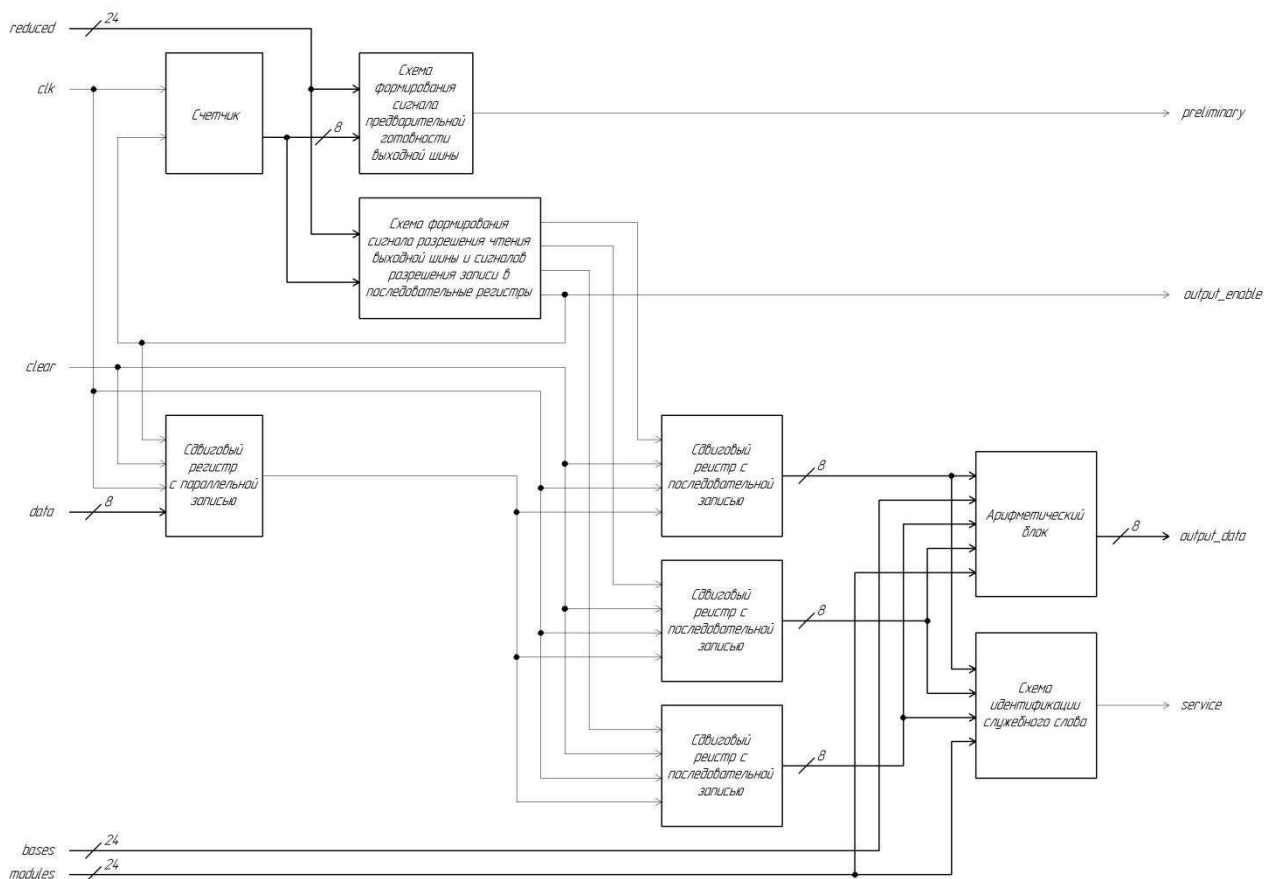


Рисунок 19 – Структурная схема блока вычисления

Блок вычисления предназначен для вычисления числа по его остаткам, а также идентификации служебного слова.

Число, представленное в виде остатков, записанных с помощью минимального количества разрядов, поступает с 8-разрядной шины данных data в сдвиговый регистр с параллельной записью. Запись в этот регистр осуществляется по отрицательному фронту синхросигнала, если окончено преобразование предыдущего числа. После записи числа по отрицательному

фронту синхросигнала происходит сдвиг хранимого числа в сторону старших разрядов и выдвинутый разряд записывается по положительному фронту в один из регистров с последовательной записью, если запись разрешена. В зависимости от значения счетчика запись может быть разрешена только в один из сдвиговых регистров с последовательной записью.

После окончания сдвига в регистрах с последовательной записью хранятся остатки от деления, записанные с помощью одинакового количества разрядов. Их содержимое поступает на входы схемы идентификации служебного слова и арифметического блока, где число вычисляется по остаткам и базисам с шины bases с учетом набора модулей с шины modules.

Сигнал предварительной готовности выходной шины preliminary формируется на предпоследнем такте цикла преобразования для начала чтения следующего значения из входного буфера, чтобы к окончанию преобразования в блоке вычисления на шине data присутствовало новое слово.

Длина цикла преобразования зависит от количества разрядов для записи остатков, которые для каждого модуля в наборе заданы на шине reduced.

RTL-представление блока вычисления данных представлено на рисунке 20.

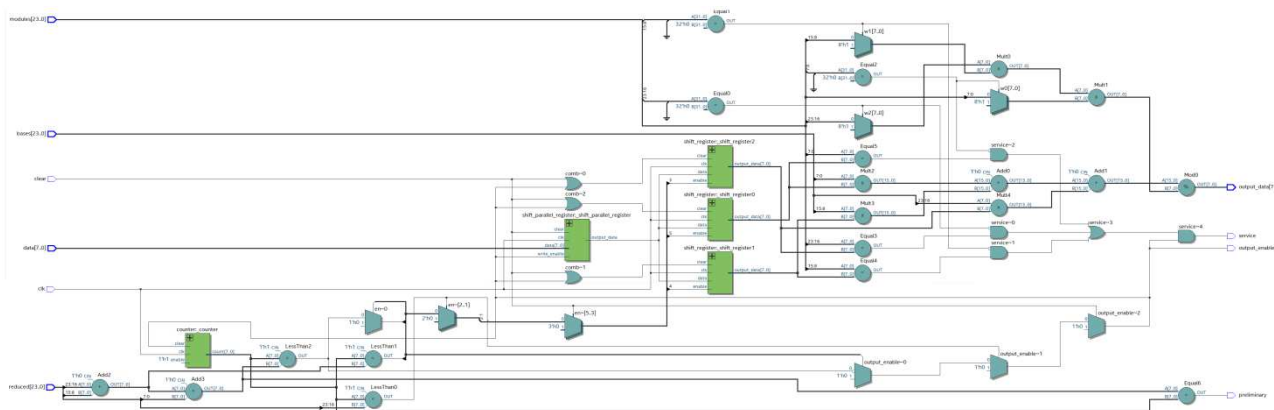


Рисунок 20 – RTL-представление блока вычисления

Сдвиговые регистры с параллельной и последовательной записью реализованы на языке описания аппаратуры Verilog в виде отдельных модулей (shift_parallel_register и shift_register). RTL-представления модулей shift_parallel_register и shift_register приведены на рисунках 21, 22 соответственно.

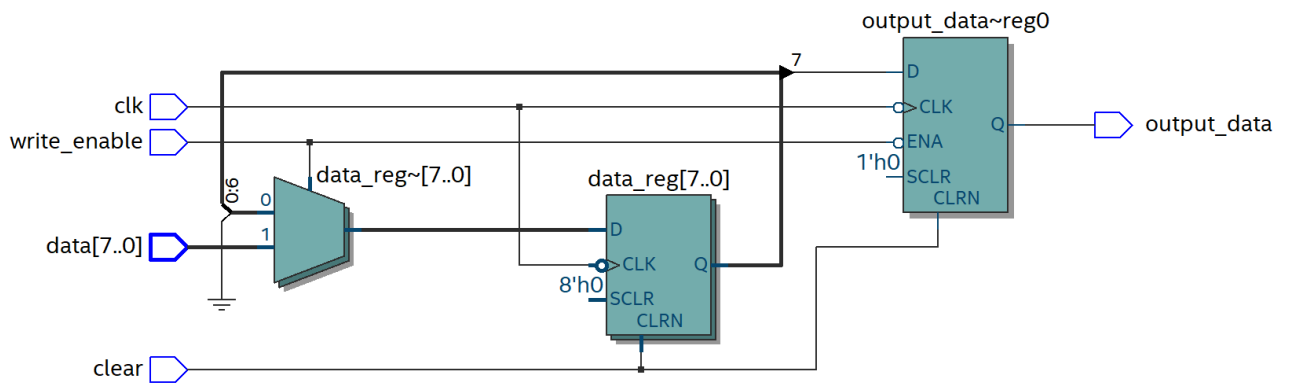


Рисунок 21 – RTL-представление модуля shift_parallel_register

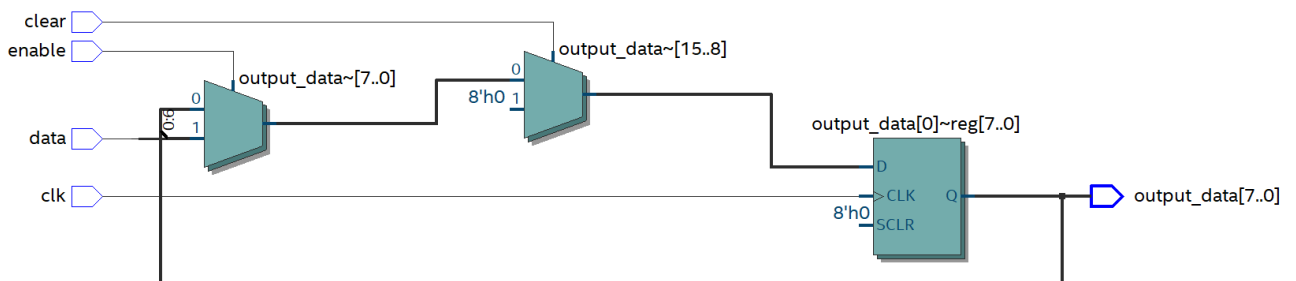


Рисунок 22 – RTL-представление модуля shift_register

Структурная схема блока формирования сигнала записи в выходной буфер представлена на рисунке 23.

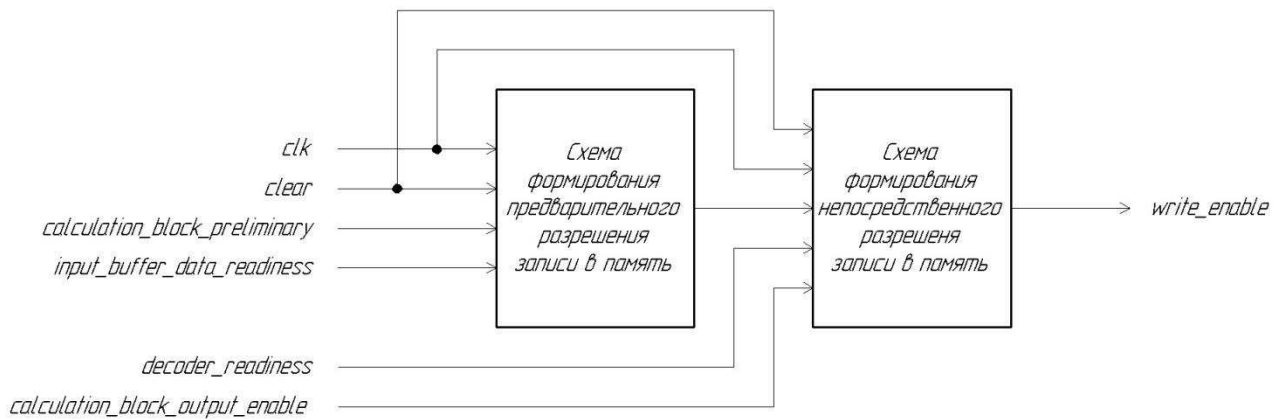


Рисунок 23 – Структурная схема блока формирования сигнала записи в выходной буфер

Блок формирования сигнала записи в выходной буфер состоит из двух ступеней.

Первая ступень (схема формирования предварительного разрешения записи в память) предназначена для проверки условия начала нового цикла преобразования в блоке вычисления, которое заключается в чтении входного буфера при окончании текущего цикла преобразования (декодирования) в блоке вычисления. Если будет начат новый цикл, то есть на входах *calculation_block_preliminary* (сигнал предварительного окончания цикла) и *input_buffer_data_readiness* (наличие во входном буфере непрочитанных (необработанных) данных) установлен высокий уровень сигнала, то данное событие фиксируется в памяти первой ступени.

Вторая ступень формирует сигнал разрешения записи, непосредственно поступающий в выходной буфер данных. Этот сигнал формируется при следующих условиях:

- декодер корректно запрограммирован (*decoder readiness* = 1);
- декодирование в блоке вычисления полностью завершено (*calculation_block_output_enable*);
- первой ступенью блока предварительно зафиксировано начало нового цикла преобразования.

Входные линии синхросигнала (clk) и сброса (clear) предназначены соответственно для тактирования и обнуления элементов памяти ступеней блока.

RTL-представление блока формирования сигнала записи в выходной буфер представлено на рисунке 24.

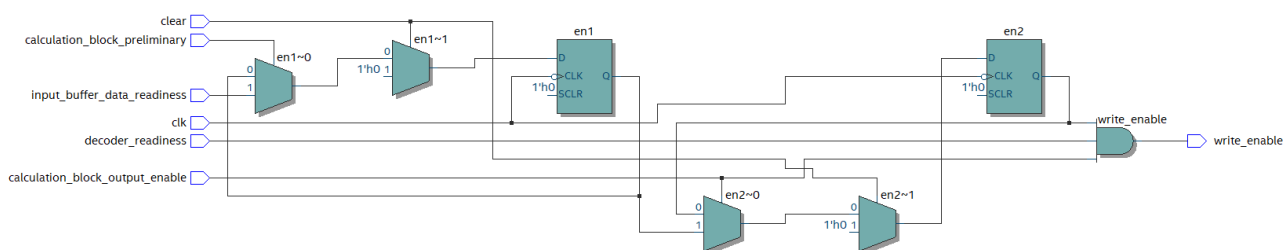


Рисунок 24 – RTL-представление блока формирования сигнала записи в выходной буфер

Результаты полного синтеза разработанного проекта в САПР Quartus Prime приведены на рисунке 25.

Flow Status	Successful - Fri Jun 19 18:36:36 2020
Quartus Prime Version	18.1.0 Build 625 09/12/2018 SJ Lite Edition
Revision Name	Project
Top-level Entity Name	decoder
Family	Cyclone IV E
Device	EP4CE115F29C7
Timing Models	Final
Total logic elements	2,671 / 114,480 (2 %)
Total registers	307
Total pins	36 / 529 (7 %)
Total virtual pins	0
Total memory bits	17,408 / 3,981,312 (< 1 %)
Embedded Multiplier 9-bit elements	14 / 532 (3 %)
Total PLLs	0 / 4 (0 %)

Рисунок 25 – Результаты полного синтеза проекта в САПР Quartus Prime

2.5 Выводы по второму разделу

Приведенная в данном разделе структура программируемого декодера СОК-ПСС позволяет осуществлять целевую функцию устройства: преобразовывать кадры сжатой информации, представленной в СОК, в кадры ПСС.

Все функциональные блоки устройства описаны на языке Verilog только с использованием синтезируемых конструкций данного языка, что позволяет говорить о полной реализуемости проекта декодера на выбранной элементной базе.

На основании результатов полного синтеза разработанного ПЛИС-проекта можно сделать вывод о достаточно высокой эффективности его реализации, с точки зрения аппаратных затрат, так как в рамках ресурсов целевой ПЛИС использовано всего лишь 2 % логических элементов и 7 % контактов.

3 Проверка работоспособности

Проверка работоспособности программируемого декодера СОК-ПСС заключалась в проведении с помощью системы HDL-моделирования ModelSim функционального и временного моделирования, для автоматизации которого разработан специализированный модуль (тестбенч), генерирующий входные воздействия в соответствии с описанием на языке Verilog.

3.1 Функциональное моделирование программирования

Характер изменений сигналов на входах декодера при функциональном моделировании программирования представлен в таблице 3.

Таблица 3 – Изменения сигналов на входах декодера при функциональном моделировании программирования

Входы декодера	Характер изменения сигналов
clear	одинокый прямоугольный импульс
clk	меандр с частотой 33,3 МГц
RCS_module	случайное изменение
RCS_module_address	циклическая последовательность значений 00 ₂ , 01 ₂ , 10 ₂
RCS_module_enable	циклическая последовательность прямоугольных импульсов
data, data_enable, read_enable	постоянный низкий уровень сигнала

Фрагмент временной диаграммы, полученной при функциональном моделировании программирования, приведен на рисунке 26.

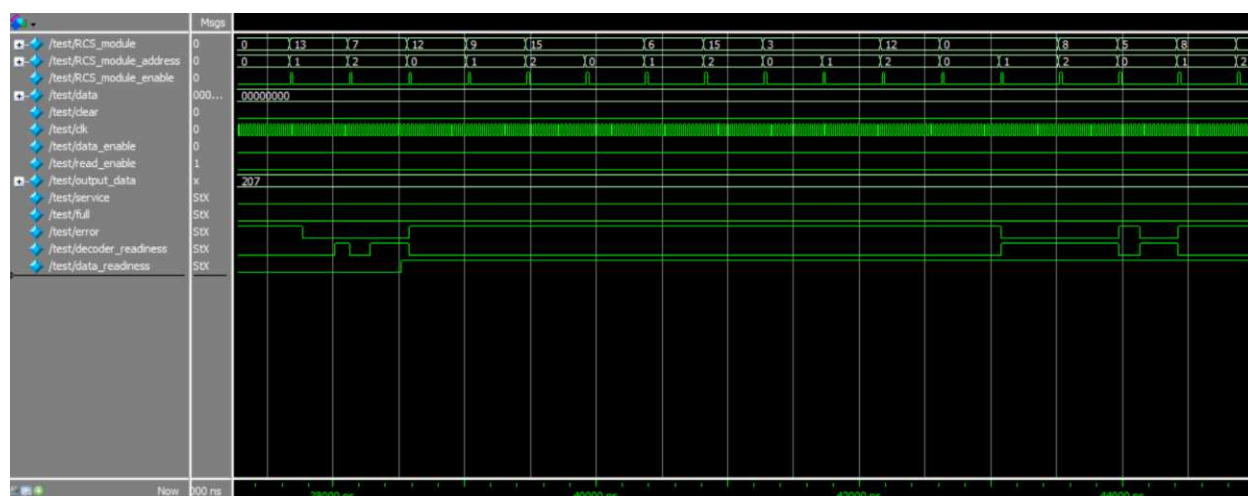


Рисунок 26 – Фрагмент временной диаграммы, полученной при функциональном моделировании программирования

Результаты функционального моделирования программирования представлены в таблице 4.

Таблица 4 – Результаты функционального моделирования программирования

Номер набора модулей СОК	Набор модулей СОК	Набор модулей СОК является корректным (да/нет)	Значения сигналов на выходах декодера		Программирование произведено успешно (да/нет)
			decoder_readiness	error	
1	4, 2, 15	Нет	0	1	Нет
2	4, 2, 6	Нет	0	1	Нет
3	15, 2, 6	Нет	0	1	Нет
4	15, 2, 12	Нет	0	1	Нет
5	0, 2, 12	Нет	0	1	Нет
6	0, 13, 12	Да	1	0	Да
7	0, 13, 7	Да	1	0	Да
8	12, 13, 7	Нет	0	1	Нет
9	12, 9, 7	Нет	0	1	Нет
10	12, 9, 15	Нет	0	1	Нет
11	15, 9, 15	Нет	0	1	Нет
12	15, 6, 15	Нет	0	1	Нет
13	3, 6, 15	Нет	0	1	Нет
14	3, 3 15	Нет	0	1	Нет
15	3, 3, 12	Нет	0	1	Нет
16	0, 3, 12	Нет	0	1	Нет
17	0, 0, 12	Да	1	0	Да
18	0, 0, 8	Да	1	0	Да
19	5, 0, 8	Да	1	0	Да
20	5, 8, 8	Нет	0	1	Нет
21	5, 8, 15	Нет	0	1	Нет
22	9, 8, 15	Нет	0	1	Нет
23	9, 6, 15	Нет	0	1	Нет
24	9, 6, 14	Нет	0	1	Нет
25	7, 6, 14	Нет	0	1	Нет
26	7, 7, 14	Нет	0	1	Нет

Окончание таблицы 4

Номер набора модулей СОК	Набор модулей СОК	Набор модулей СОК является корректным (да/нет)	Значения сигналов на выходах декодера		Программирование произведено успешно (да/нет)
			decoder_readiness	error	
27	7, 7, 6	Нет	0	1	Нет
28	5, 7, 6	Нет	0	1	Нет
29	5, 7, 13	Нет	0	1	Нет
30	12, 7, 13	Нет	0	1	Нет
31	12, 4, 13	Нет	0	1	Нет
32	12, 4, 8	Нет	0	1	Нет
33	6, 4, 8	Нет	0	1	Нет
34	6, 11, 8	Нет	0	1	Нет
35	6, 11, 11	Нет	0	1	Нет
36	9, 11, 11	Нет	0	1	Нет
37	9, 6, 11	Нет	0	1	Нет
38	9, 6, 6	Нет	0	1	Нет
39	9, 1, 6	Нет	0	1	Нет
40	9, 1, 3	Нет	0	1	Нет
41	4, 1, 3	Нет	0	1	Нет
42	4, 2, 3	Нет	0	1	Нет
43	4, 2, 4	Нет	0	1	Нет
44	6, 2, 4	Нет	0	1	Нет
45	6, 3, 4	Нет	0	1	Нет
46	6, 3, 11	Нет	0	1	Нет
47	3, 3, 11	Нет	0	1	Нет
48	3, 2, 11	Да	1	0	Да
49	3, 2, 0	Да	1	0	Да
50	4, 2, 0	Нет	0	1	Нет
51	4, 11, 0	Да	1	0	Да

3.2 Функциональное моделирование декодирования

Характер изменений сигналов на входах декодера при функциональном моделировании декодирования представлен в таблице 5.

Таблица 5 – Изменения сигналов на входах декодера при функциональном моделировании декодирования

Входы декодера	Характер изменения сигналов
clear	одинокый прямоугольный импульс
clk	меандр с частотой 33,3 МГц
RCS_module	последовательность значений $7_{10}, 2_{10}, 15_{10}$
RCS_module_address	последовательность значений $00_2, 01_2, 10_2$
RCS_module_enable	последовательность прямоугольных импульсов
data	случайное изменение
data_enable	циклическая последовательность прямоугольных импульсов
read_enable	постоянный высокий уровень сигнала

Фрагмент временной диаграммы, полученной при функциональном моделировании декодирования, приведен на рисунке 27.

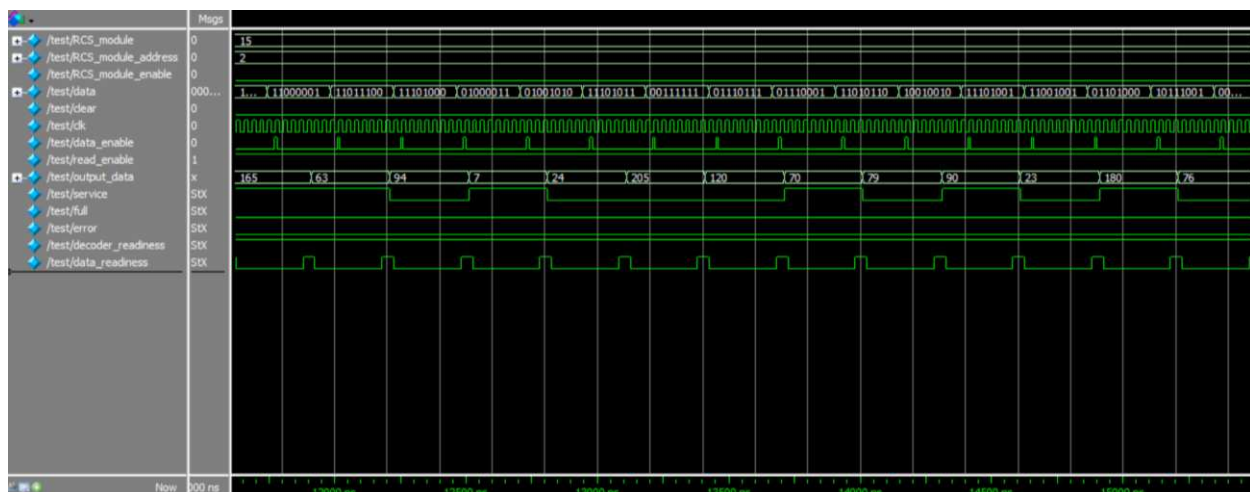


Рисунок 27 – Фрагмент временной диаграммы, полученной при функциональном моделировании декодирования

Результаты функционального моделирования декодирования представлены в таблице 6.

Таблица 6 – Результаты функционального моделирования декодирования

Номер слова	Набор модулей СОК	Декодируемое слово	Значения на выходах декодера		Декодирование или идентификация служебного слова произведены успешно (да/нет)
			output_data	service	
1	7, 2, 15	01110100 ₂	199 ₁₀	0	Да
2		11001101 ₂	118 ₁₀	0	Да
3		00011101 ₂	133 ₁₀	0	Да
4		00110110 ₂	141 ₁₀	0	Да
5		01100011 ₂	108 ₁₀	0	Да
6		10110110 ₂	201 ₁₀	0	Да
7		01001011 ₂	86 ₁₀	0	Да
8		11111110 ₂	119 ₁₀	1	Да

Продолжение таблицы 6

Номер слова	Набор модулей СОК	Декодируемое слово	Значения на выходах декодера		Декодирование или идентификация служебного слова произведены успешно (да/нет)
			output_data	service	
9	7, 2, 15	01011110 ₂	149 ₁₀	0	Да
10		00010001 ₂	91 ₁₀	0	Да
11		01000001 ₂	16 ₁₀	0	Да
12		01111000 ₂	143 ₁₀	0	Да
13		10100010 ₂	152 ₁₀	0	Да
14		10001001 ₂	144 ₁₀	0	Да
15		11001000 ₂	188 ₁₀	0	Да
16		00101010 ₂	190 ₁₀	0	Да
17		01100000 ₂	150 ₁₀	0	Да
18		00111001 ₂	99 ₁₀	0	Да
19		01101111 ₂	150 ₁₀	1	Да
20		01011111 ₂	135 ₁₀	1	Да
21		00110111 ₂	127 ₁₀	0	Да
22		10110100 ₂	19 ₁₀	0	Да
23		11000111 ₂	202 ₁₀	0	Да
24		11011001 ₂	69 ₁₀	0	Да
25		00010010 ₂	77 ₁₀	0	Да
26		10010001 ₂	151 ₁₀	0	Да
27		11011111 ₂	195 ₁₀	1	Да
28		10000001 ₂	46 ₁₀	0	Да
29		01001100 ₂	72 ₁₀	0	Да
30		11010010 ₂	167 ₁₀	0	Да
31		01010111 ₂	37 ₁₀	0	Да
32		01100110 ₂	66 ₁₀	0	Да
33		00010101 ₂	35 ₁₀	0	Да

Окончание таблицы 6

Номер слова	Набор модулей СОК	Декодируемое слово	Значения на выходах декодера		Декодирование или идентификация служебного слова произведены успешно (да/нет)
			output_data	service	
34	7, 2, 15	00111111 ₂	15 ₁₀	1	Да
35		10011111 ₂	165 ₁₀	1	Да
36		11110011 ₂	63 ₁₀	1	Да
37		01100100 ₂	94 ₁₀	0	Да
38		11110111 ₂	7 ₁₀	1	Да
39		01101001 ₂	24 ₁₀	0	Да
40		01011010 ₂	205 ₁₀	0	Да
41		00100000 ₂	120 ₁₀	0	Да
42		11101010 ₂	70 ₁₀	1	Да
43		01010100 ₂	79 ₁₀	0	Да
44		11001111 ₂	90 ₁₀	1	Да
45		01011000 ₂	23 ₁₀	0	Да
46		10101111 ₂	180 ₁₀	1	Да
47		11000001 ₂	76 ₁₀	0	Да
48		11011100 ₂	27 ₁₀	0	Да
49		11101000 ₂	98 ₁₀	1	Да
50		01000011 ₂	198 ₁₀	0	Да
51		01001010 ₂	100 ₁₀	0	Да

3.3 Временное моделирование

Характер изменения сигналов на входах декодера при проведении временного моделирования программирования и декодирования аналогичен приведенному в таблицах 3, 5.

3.4 Выводы по третьему разделу

На основании результатов проведенного моделирования можно сделать вывод о том, что с точки зрения программирования и декодирования, разработанный декодер СОК-ПСС работает корректно.

Объективность результатов проведенной проверки работоспособности обусловлена:

- учетом реальных временных задержек при распространении сигналов в целевой ПЛИС;
- случайным изменением значений на входных шинах данных декодера.

ЗАКЛЮЧЕНИЕ

В ходе выполнения данной выпускной квалификационной работы разработан и протестирован ПЛИС-проект компонента аппаратуры системы передачи и обработки информации, осуществляющий преобразование кадра информации, представленной в непозиционной системе остаточных классов, в кадр позиционной двоичной системы счисления.

В данном проекте предусмотрено программирование декодера посредством задания набора модулей СОК, в соответствии с которым осуществляется преобразование кадров. Для хранения входных и декодированных данных имеются соответствующие входной и выходной буферы.

В процессе проектирования разработаны структурные схемы декодера и его основных функциональных блоков. Кроме этого, определены принципы выбора оснований СОК для устройства и структура обрабатываемых кадров. Также изучены особенности непозиционной системы остаточных классов и представлен алгоритм перевода чисел из СОК в ПСС.

Работа выполнена в соответствии с нормативными документами [11-14].

СПИСОК СОКРАЩЕНИЙ

ПЛИС – программируемая логическая интегральная схема

ПСС – позиционная система счисления

САПР – система автоматизированного проектирования

СОК – система остаточных классов

HDL – hardware description language (язык описания аппаратуры)

RTL – register transfer level (уровень регистровых передач, функциональный уровень)

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Горденко, Д. В., Горденко, Н. В., Павлюк, Д. Н., Петлина, Е. М., Юрданов, Д. В. К вопросу применения системы остаточных классов в современных устройствах цифровой обработки сигналов / Д. В. Юрданов, Д. В. Горденко, Д. Н. Павлюк, Е. М. Петлина, Н. В. Горденко // *Фундаментальные исследования : научный журнал.* – 2016. – № 2. – С. 318–322.

2. Постников, А. И. Схемотехника ЭВМ : учеб. пособие / А. И. Постников, В. И. Иванов, О. В. Непомнящий. – Красноярск : Сиб. федер. ун-т, 2018. – С. 110–122.

3. Савельев, А. Я. Выполнение арифметических операций в непозиционных системах счисления / А. Я. Савельев // *Арифметические и логические основы цифровых автоматов : учебник / А. Я. Савельев.* – Москва, 1980. – Гл. 7. – С. 132–143.

4. Галанина, Н. А., Дмитриев, Д. Д. Разработка конфигурационного файла для реализации дискретного преобразования Фурье в системе остаточных классов на ПЛИС / Н. А. Галанина, Д. Д. Дмитриев // *Вестник Чувашского университета.* – 2011. – № 3. С. 119–124.

5. Осинин, И.П. Модулярно-логарифмический процессор / И. П. Осинин // *Известия ЮФУ. Технические науки.* – 2016. – № 11. – С. 13–26.

6. Калмыков, М. И., Бабенко, Л. К., Калмыков, И. А., Ефременков, И. Д., Мирошников, Д. А. Использование избыточных модулярных кодов при разработке отказоустойчивых запросно-ответных систем распознавания спутника / М. И. Калмыков, Л. К. Бабенко, И. А. Калмыков, И. Д. Ефременков, Д. А. Мирошников // *Фундаментальные исследования.* – 2017. – № 12. – С.292–296.

7. ГОСТ Р ИСО/МЭК 19762–1–2011 Информационные технологии. Технологии автоматической идентификации и сбора данных (АИСД). Гармонизированный словарь. Часть 1. Общие термины в области АИСД. – Введ. 01.05.2012. – Москва : Стандартинформ, 2012. – 36 с.

8. Курс лекций по дисциплине «Элементы теории передачи информации» [Электронный ресурс]. – Режим доступа <http://www.supervideoman.narod.ru/index.htm>.

9. Исмаилов, Ш.-М. А. Алгоритмы и структуры устройств преобразования чисел из системы остаточных классов в двоичный код с вычисляемыми и табличными слагаемыми / Ш.-М. А. Исмаилов // Мониторинг. Наука и технологии. – 2009. – № 1. – С. 89–98.

10. Официальный сайт корпорации Intel [Электронный ресурс]. – Режим доступа: <https://www.intel.ru>.

11. СТО 4.2–07–2014 Система менеджмента качества. Общие требования к построению, изложению и оформлению документов учебной деятельности. – Красноярск : СФУ, 2014. – 60 с.

12. ГОСТ 2.701–2008 Единая система конструкторской документации. Схемы. Виды и типы. Общие требования к выполнению. – Взамен ГОСТ 2.701–84 ; введ. 01.07.2009. – Москва : Стандартинформ, 2009. – 13 с.

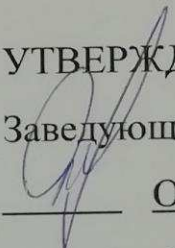
13. ГОСТ 2.702–2011 Единая система конструкторской документации. Правила выполнения электрических схем. – Взамен ГОСТ 2.702–75 ; введ. 01.01.2012. – Москва : Стандартинформ, 2011. – 22 с.

14. ГОСТ 2.721–74 Единая система конструкторской документации. Обозначения условные графические в схемах. Обозначения общего применения. – Введен 01.07.1975. – Москва : Стандартинформ, 2008. – 33 с.

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
институт
Вычислительная техника
кафедра

УТВЕРЖДАЮ

Заведующий кафедрой


О. В. Непомнящий

подпись инициалы, фамилия

«_____» _____ 2020 г.

БАКАЛАВРСКАЯ РАБОТА

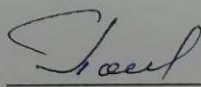
09.03.01 – «Информатика и вычислительная техника»

код – наименование направления

Программируемый декодер СОК-ПСС

тема

Руководитель



подпись, дата


доцент, канд. тех. наук

должность, ученая степень

А. И. Постников

инициалы, фамилия

Выпускник

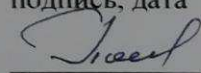
 25.06.20

подпись, дата

Д. А. Калинин

инициалы, фамилия

Нормоконтролер



подпись, дата

А. И. Постников

инициалы, фамилия

Красноярск 2020