УДК 004.383.3

**Обработка шумоподобных сигналов на СБИС 1888ВС058**

*Сергей Владимирович Ландышев*

ЗАО НТЦ «Модуль», Россия, 125190, г. Москва, 4-я ул. 8 марта, д.3. [landysh@module.ru](mailto:landysh@module.ru)

**Аннотация —** В статье рассматриваются принципы обработки шумоподобных сигналов на СБИС 1888ВC058. Описываются архитектурные особенности СБИС и программные модели аппаратных блоков. Аппаратные средства СБИС рассматриваются в контексте их применения в системах связи с реализацией пользовательских протоколов.

# Ключевые слова — NMC3, NeuroMatrix, SDR, векторно-матричная обработка, ПЦОС, СБИС, система на кристалле, шумоподобные сигналы.

**Noise-like signals processing on the 1888ВС058 ASIC**

*Sergey Landyshev*

RC Module, Russia, 125190, Moscow, 3 Eighth March 4th Street. [landysh@module.ru](mailto:m.klimenko@module.ru)

**Abstract —** The article discusses the principles of processing noise-like signals on the VLSI 1888BC058. The architectural features of VLSI and software models of the main hardware units are described. VLSI hardware is considered in the context of its application in communication systems with the implementation of user-defined data link protocols.

# Keywords: NMC3, NeuroMatrix, Software defined radio receiver, ASIC, System on a Chip, digital signal processing.

# Введение

В статье рассматривается применение специализированной схемы ЦОС в задачах обработки шумоподобных сигналов (ШПС). Такие сигналы широко используются в системах связи, навигации и радиолокации.

Формирование ШПС производится путём модуляции высокочастотного несущего колебания специальной псевдослучайной последовательностью. Являясь широкополосными сигналами, ШПС обладают высокими помехозащищённостью и скрытностью [1], вместе с тем обработка таких сигналов требует значительных вычислительных ресурсов. Для систем, работающих на высокой «чиповой» скорости, крайне затруднительна обработка только программными средствами без использования специальных аппаратных средств.

Обработка ШПС на СБИС 1888ВС058 [2] возможна различными методами в зависимости от характеристик сигнала. При проектировании микросхемы разработчиками были выделены наиболее типовые способы передачи сообщений с применением ШПС. Прежде всего это реализация канального уровня с применением кадрирования, когда передаваемые данные разбиваются на порции или кадры, которые обозначаются специальным стартовым импульсом. После обнаружения импульса приёмник захватывает кадр, декодирует сообщение и передаёт данные на обработку вышестоящим протоколам. Для такой схемы обработки используются специальные аппаратные блоки, которые гибко конфигурируются. Для реализации других протоколов, когда аппаратные блоки не могут быть сконфигурированы должным образом, предполагается применение входящих в состав СБИС двух процессорных систем NeuroMatrix© Core третьего поколения (ядро DSP NMC3). Ядро разработано ЗАО НТЦ «Модуль».

Первичная задача при реализации покадровой обработки – это обнаружение начала данных, которая сводится к поиску синхронизирующей последовательности. Момент излучения сообщения не известен заранее, поэтому необходимо обрабатывать весь поток данных непрерывно. Эта задача выполняется специальным блоком предварительной обработки и блоком обнаружения импульса.

В СБИС реализована аппаратная поддержка декодирования сообщений с применением ортогональных преобразований по типу преобразования Адамара. В этом же блоке выполнена схема выделения двух позиций с максимальными значениями после обработки (выделение символа).

Реализация протоколов верхних уровней (сетевые, транспортные, прикладные и пр.) реализуются программным образом с использованием процессорных ядер NMC3 и системы ARM, имеющей ряд интерфейсных блоков, обеспечивающих взаимодействие микросхемы с внешним миром.

Суть применения СБИС заключается в использовании программистом предлагаемой разработчиками жёсткой логики программных моделей существующих на кристалле аппаратных блоков. Для процессорных систем NMC3 разработаны инструментальные средства программиста на базе GCC 4.8.3. Разработчику доступны компиляторы C/C++, gas, редактор связей, архиватор, эмулятор, отладчик GDB и интегрированная среда разработки на платформе eclipse. Кроме инструментальных средств существует специальная библиотека примитивов для архитектуры NeuroMatrix®, в которой содержатся оптимизированные функции для работы со скалярами, векторами, матрицами, сигналами и изображениями. Поддерживает целочисленную арифметику с 1, 2, 4, 8, 16, 32, 64-разрядными данными процессоров семейства NMC3 [3]. Для разработки ПО процессорной системы ARM предлагается использовать штатные средства — платформа DS-5 [4] или GNU инструменты.

Для программирования аппаратных блоков предлагается использование слоя аппаратных абстракций в виде программных библиотек. Важная особенность слоя — это его кроссплатформенность. Большинство аппаратных блоков для обработки ШПС «видны» как системе ARM, так и процессорным ядрам NMC3. В библиотеках предлагается общий интерфейс, а все особенности работы с конкретной системой скрыты внутри.

**СБИС** **1888ВС058**

СБИС имеет гетерогенную структуру (рисунок 1), состоящую из управляющего ядра ARM Cortex-A5 и двух DSP ядер NMC3. Процессорные ядра работают на частоте 512 МГц, общий объём банковой системы внутренней памяти 2.5 Мбайт. Коммуникационные возможности СБИС: интерфейсы Ethernet, UART, GPIO, SPI, коммуникационные порты CP для соединения нескольких СБИС, интерфейс с внешней памятью EMI.

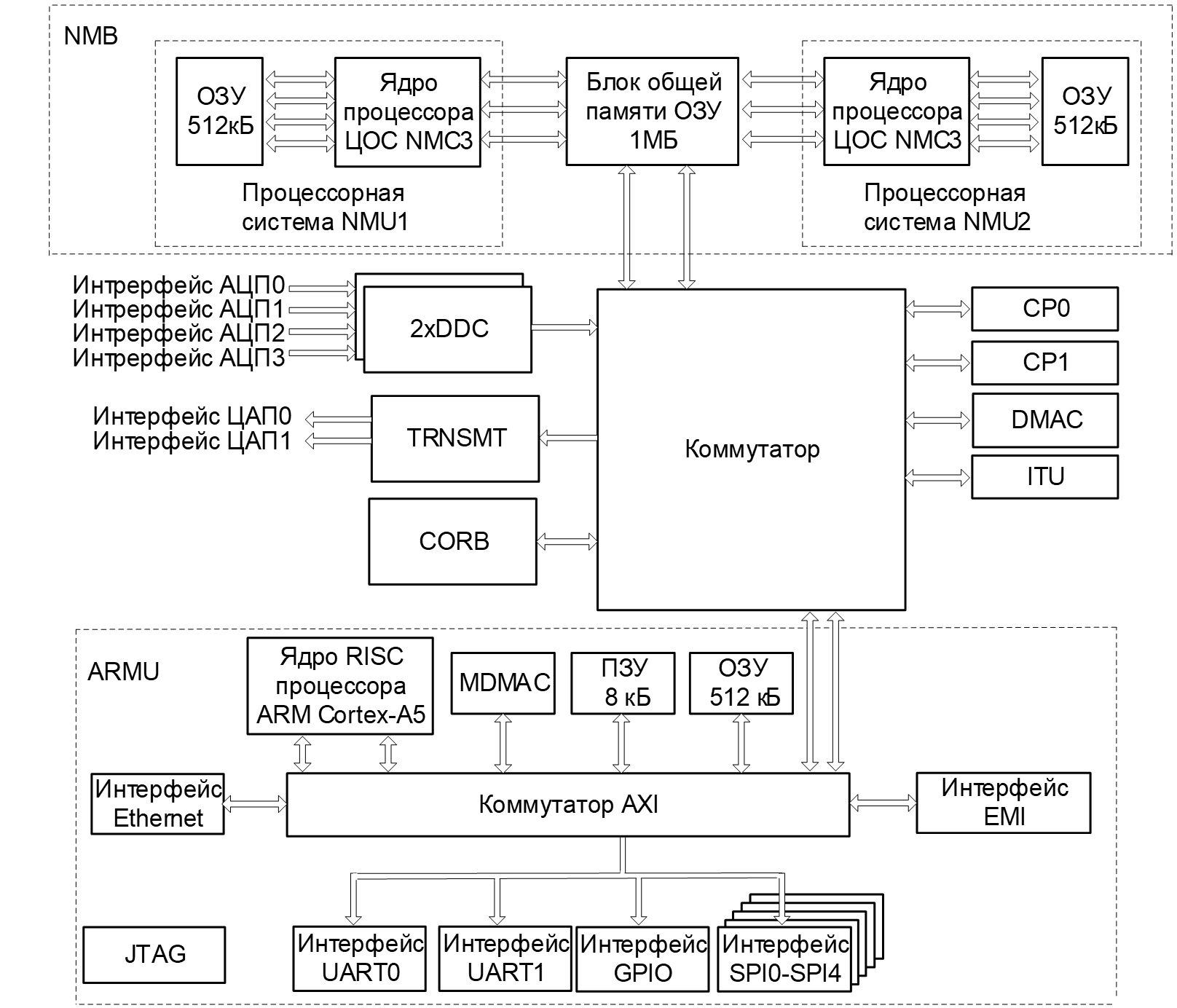


Рисунок 1 — Обобщенная структурная схема СБИС 1888ВС058

Связная специфика микросхемы обеспечивается тремя блоками:

* 2xDDC – двухканальный блок предварительной обработки сигналов с интерфейсами подключения четырёх АЦП (до 128 МГц, 14 бит).
* TRNSMT – блок формирования сигнала с интерфейсами подключения двух ЦАП (до 512 МГц, 16 бит).
* CORB – блок декодера символов сообщения выполняющий корреляционную обработку.

На рисунке 2 показана структура одного канала блока предобработки. В канале реализуется конвейер обработки оцифрованных комплексных отсчётов, приходящих из блока интерфейсов с АЦП. Количество АЦП – четыре преобразователя – обусловлены двумя каналами предобработки – на каждый канал по два АЦП для формирования входного квадратурного сигнала. Отсчёты обрабатываемого сигнала последовательно модифицируются в блоках цифрового гетеродина (смесителя), КИХ-фильтра, понижение темпа поступления отсчётов (децимации), БИХ-фильтра и обнаружителя, который в свою очередь состоит из согласованного фильтра, вычислителя порогов и блока сравнения корреляционного пика с вычисленным порогом.

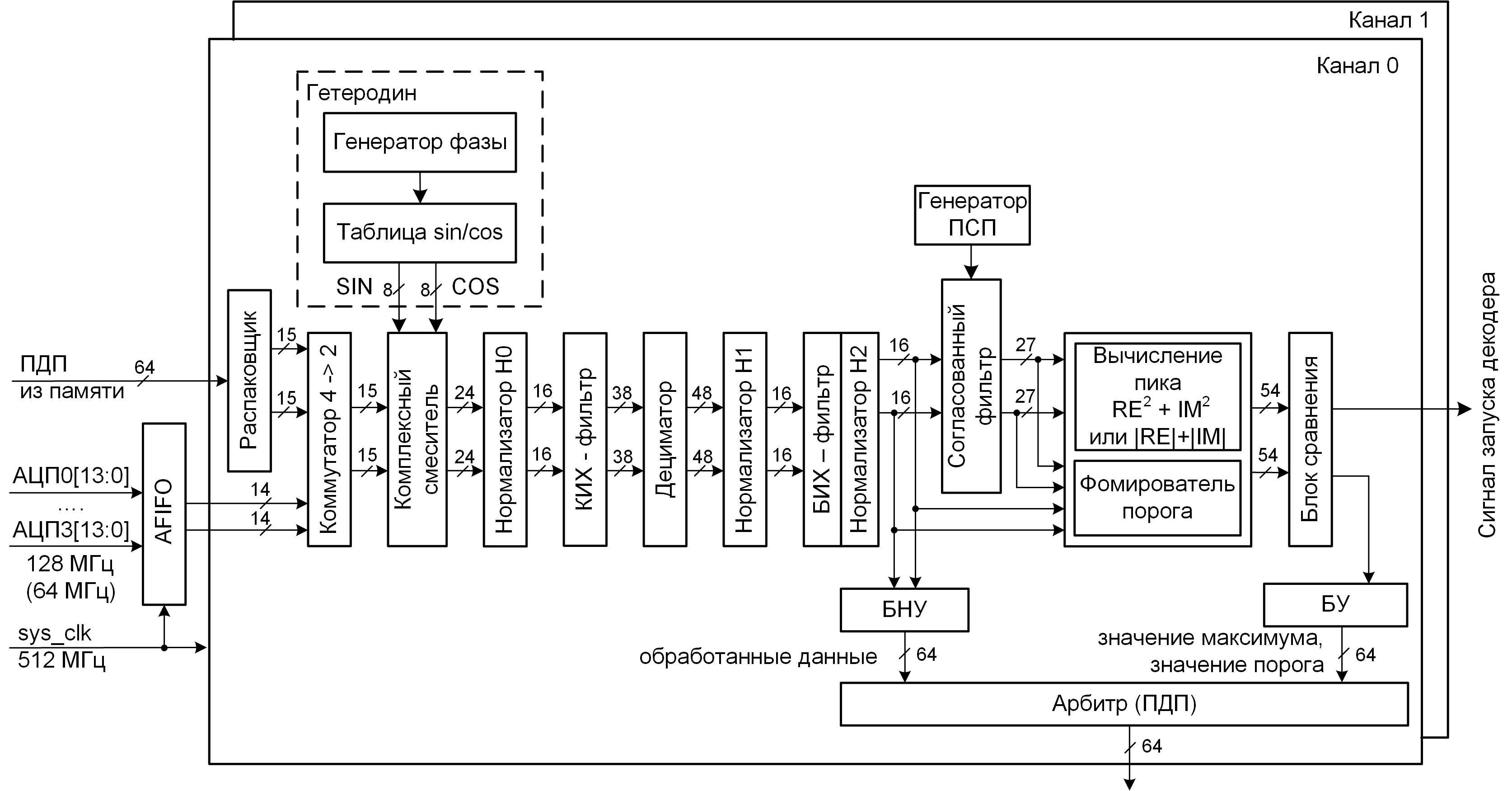


Рисунок 2 — Структура канала DDC

Все операции в конвейере производятся над комплексными целочисленными операндами в разрядной сетке, которая обеспечивает вычисления без потерь. После обработки в блоках, оперирующими числами в формате с фиксированной точкой (блоки смесителя и фильтров), предусмотрены блоки нормализации. Запись данных в специальные циклические буферы происходит в двух точках конвейера – после нормализации БИХ-фильтра (БНУ – Блок Накопления и Упаковки) и после обнаружения корреляционного пика (БУ – Блок Упаковки). В первом буфере содержится предварительно обработанный сигнал, и он заполняется синхронно по заданной временной метке кадрами фиксированного размера. Во втором буфере содержится структурированная информация об обнаруженном пике, и он заполняется по событию от обнаружителя.

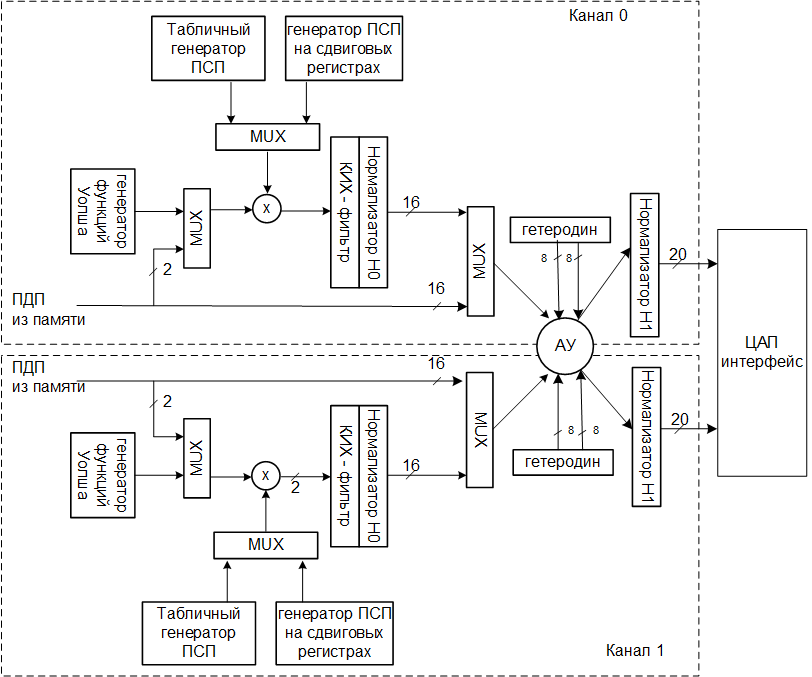


Рисунок 3 — Структурная схема блока формирования сигналов

Задача блока формирования сигналов (рисунок 3) – синтез данных, поступающих в интерфейс ЦАП. Устройством поддерживается работа в одном из трёх режимов:

* Режим действительных входных данных и комплексных выходных. В этом режиме используется один комплексный канал передачи с действительными исходными данными. Комплексный сигнал формируется после обработки в блоке гетеродина.
* Режим двух действительных каналов. В этом режиме каналы работают независимо друг от друга.
* Режим одного комплексного канала. В этом режиме каналы работают как один комплексный канал – один канал формирует синфазную составляющую, другой – квадратурную.

**Построение устройств связи с применением СБИС** **1888ВС058**

Аппаратный состав микросхемы позволяет реализовать полный стек сетевых протоколов. Физический уровень обеспечивается интерфейсами АЦП и ЦАП. Канальный уровень реализуется с применением блоков предварительной обработки, декодера, блока формирования сигналов и процессорных систем NMC3. Остальные вышестоящие уровни реализуются на процессорных системах ARM и NMC3.

Наиболее полная аппаратная поддержка обеспечивается на канальном уровне взаимодействия по следующему принципу: передаваемые сообщения являются последовательностями символов двух типов – стартового символа (СС) и информационного символа (ИС). Сначала передаётся один или несколько стартовых символов, затем передаются информационные символы (см. рисунок 4). Между символами могут быть детерминированные паузы.

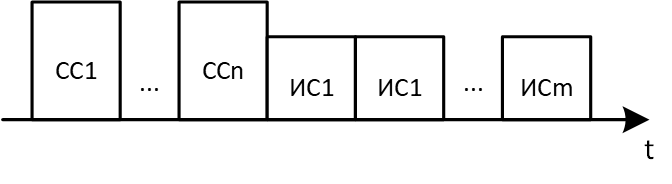


Рисунок 4 — Структура сообщения

Символы состоят из битовой последовательности, по которой формируется фазово-манипулированный сигнал (BPSK модуляция). Стартовый символ формируется из последовательности с высокими автокорреляционными свойствами. Такими свойствами обладают, например, М-последовательности и коды Голда [1], которые формируются с применением регистров сдвига с обратной связью (LFSR). В блоке предварительной обработки и блоке формирования сигналов поддерживается формирование последовательностей – сумм от выходов трёх независимых LFSR. Также можно использовать произвольные последовательности, задаваемые в виде таблицы. Длина последовательности может составлять 64, 128, 256, 512 или 1024 битов.

Информационные символы формируются из строки матрицы, состоящей из функций Уолша (W). Для передачи информации, размером *m* бит выбирается последовательность-строчка (Wm) из матрицы размером 2mx2m. Строки матрицы упорядочены по Адамару, номер строки – это и есть передаваемая информация или символ. Например, при передаче значений в диапазоне 0 … 255 используется матрица 256x256. Биты ИС также могут быть сложены по модулю два с генерируемой блоками последовательностью, которая образует шумовую гамму информационного символа.

,

,

где *i* – номер бита последовательности, *j* – номер LFSR, *m* – передаваемая информация.

Кодовое разделение абонентов осуществляется выбором уникальных последовательностей для формирования СС и шумовой гаммы для ИС.

В каналах блока предварительной обработки осуществляется демодуляция и фильтрация всех отсчётов входного сигнала. В режиме приёма сообщений в блоке также осуществляется обнаружение стартовых символов.

Обнаружение начинается с корреляционной обработки отсчётов в согласованном фильтре. Фильтр работает на максимальной частоте отсчётов в четыре раза меньшей частоты дискретизации АЦП, то есть максимальная частота обработки отсчётов в фильтре составит 32 МГц при использовании АЦП 128 МГц. Согласованный фильтр 1024-го порядка спроектирован, как один фильтр 64-го порядка с аккумулятором частичных сумм, которые сохраняются в памяти. Такой фильтр задерживает выдачу результата на 16 тактов и позволяет гибко конфигурироваться. В каждом канале можно одновременно задействовать несколько фильтров. Количество и порядок фильтров – взаимосвязанные параметры, они могут принимать одно из следующих значений: 1 фильтр 1024 порядка, 2 фильтра 512 порядка, 4 фильтра 256 порядка, 8 фильтров 128 порядка и 16 фильтров 64 порядка. Это означает возможность одновременного приёма сигнала от нескольких абонентов, когда база стартового импульса менее 1024.

После корреляционной обработки происходит выделение корреляционного пика с применением одного из двух порогов.

В первом случае пороговое значение вычисляется по формуле:

, где

* *thresholdn* – оцениваемая масштабированная мощность шума (порог).
* *k* – масштабирующий коэффициент.
* *R* – размер окна, в котором оценивается мощность шума.
* *I, Q* – синфазная и квадратурная части сигнала после корреляционной обработки.

Условием наличия сигнала в отсчёте с номером *n* является превышение порога: .

– мощность сигнала на выходе согласованного фильтра, где

* *P* – порядок согласованного фильтра.
* *b* – импульсная характеристика согласованного фильтра.
* *I, Q* – синфазная и квадратурная части исходного сигнала.

Во втором случае (см. рисунок 5) порог определяется выражением:

, где

* ,
* , *S* – выход согласованного фильтра.
* Среднее значение в скользящем окне размером *W*, определяется выражением: .

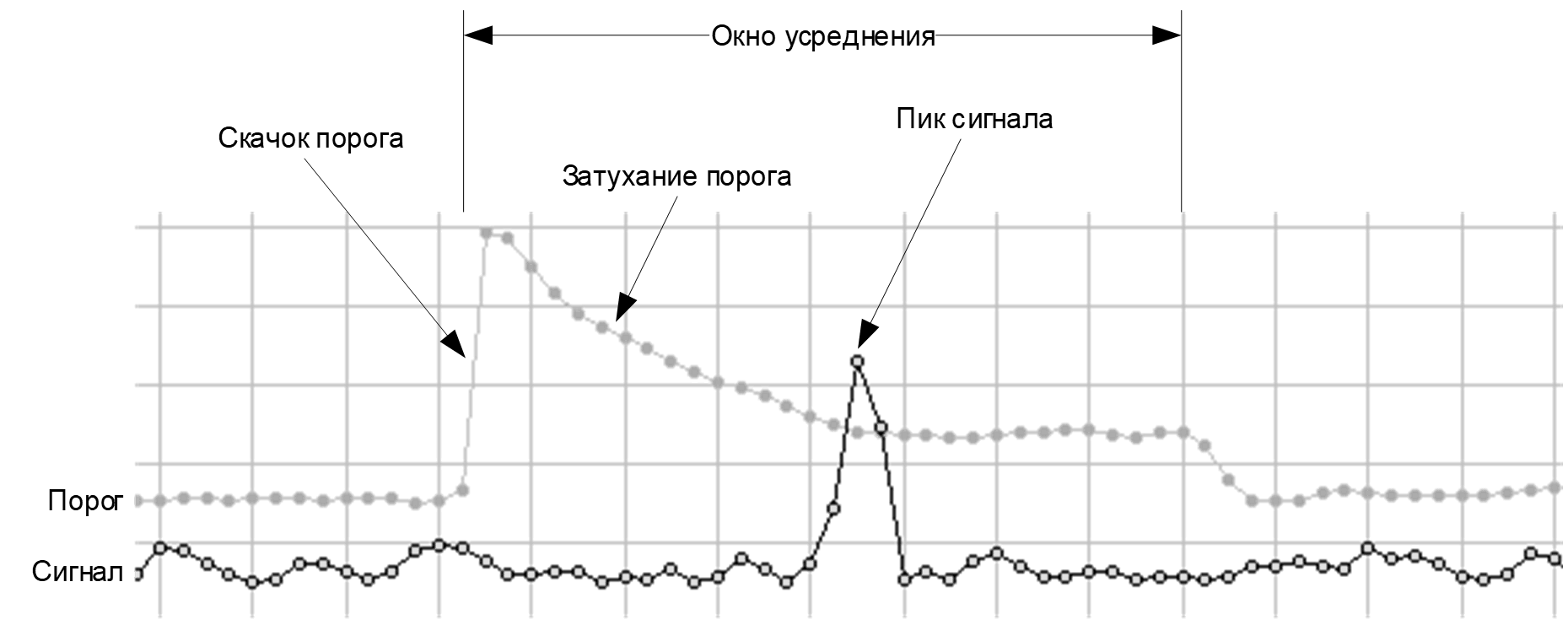


Рисунок 5 — Вариант порога с затуханием

После порогового обнаружения сигнала обнаружителем формируется структурированный отчёт о выделенном импульсе и инициируется прерывание процессорных ядер.

Во время работы обнаружителя отсчёты возможных информационных символов сохраняются в циклическом буфере, откуда они выбираются для обработки после детектирования стартового символа. Декодирование информационных символов сводится к последовательному формированию задания для блока CORB и последующей обработке сформированного блоком структурированного отчёта: для каждого из информационных символов сначала снимается шумовая последовательность, затем производится умножение отсчётов блока на матрицу Адамара, со строками-функциями Уолшу, упорядоченными по Адамару. Позиция максимального по уровню отсчёта в произведении является искомым словом.

, , где

* *W* – искомое декодированное слово.
* *M`n* – декодированный сигнал.
* *L* – размер информационного импульса.
* *H* – матрица Адамара.
* *G* – шумовая последовательность.
* *I, Q* – синфазная и квадратурная части исходного сигнала.

Передача сообщения выполняется конфигурированием блока формирования сигналов TRNSMT для каждого передаваемого символа.

Управление аппаратными блоками DDC, CORB и TRNSMT осуществляется как со стороны ядер NMC3, так и со стороны процессорной системы ARM. Выбор того или иного способа управления зависит от выбранных сетевых протоколов и типа конечного связного устройства. Типовое построение устройства связи предполагает управление блоками одним из ядер NMC3. Такой способ позволит реализовать пользовательские протоколы, в которых предусматривается обработка, не поддерживаемая в блоках СБИС. В этих случаях предварительную обработку можно выполнить на DSP NMC3, оперативно дообрабатывая неполностью обработанные отсчёты из циклического буфера БНУ (см. рисунок 2). В этом случае целесообразно использовать программный пакет NMPP, предоставляющий набор глубоко оптимизированных функций для обработки векторов и матриц данных с учётом архитектурных особенностей DSP NMC3.

При управлении блоками со стороны ядер NMC3 на системе ARM выполняются прикладные и коммуникационные задачи, а также может быть реализован процессор приложений, работающий под управлением операционной системы.

**Библиотека слоя аппаратных абстракций**

Для программного управления аппаратными блоками СБИС разрабатывается библиотека с интерфейсом С++. Сборка компонентов библиотеки осуществляется в двух вариантах – для процессорных ядер NMC и для системы ARM.

Программная модель аппаратных блоков состоит из множества регистров-конфигураторов, расположенных в выделенном адресном пространстве (порты ввода-вывода). Работа с этими регистрами выполняется в режиме битовых операций – установка или снятие одного или группы битов, при этом не осуществляется проверка корректности и совместимости настроек. Для управления такими устройствами с применением высокоуровневых языков программирования разрабатываются библиотеки слоя аппаратных абстракций (HAL — Hardware Abstraction Layer). Использование HAL минимизирует возможные ошибки и позволяет разработчику абстрагироваться от аппаратных особенностей и сконцентрироваться на решении целевой задачи.

В контексте объектно-ориентированного программирования устройство описывается как класс. Конфигурируемые регистры описываются как структуры с битовыми полями и являются полями класса-устройства. В методах модификации регистров, где это необходимо, осуществляется проверка корректности устанавливаемых значений. Работа с устройствами в слое абстракций сводится к инициализации, конфигурировании и обработки асинхронных событий (прерывания), которая выполняется в пользовательских функциях обратного вызова.

Для СБИС 1888ВС058 поддерживается работа со следующими компонентами со стороны процессорных систем ARM или NMC3:

|  |  |  |
| --- | --- | --- |
| *Аппаратный блок* | *ARM* | *NMC3* |
| Контроллер прерываний | + | + |
| Блок формирования временных шкал и синхронизации с внешними событиями (ITU) | + | + |
| Система синхронизации и сброса (CRG) | + | – |
| Контроллер интерфейса с внешней памятью (EMI) | + | – |
| Контроллер прямого доступа к памяти | + | + |
| Блок сдвоенных интервальных таймеров | + | – |
| Блок сторожевого таймера | + | – |
| Контроллер интерфейса Ethernet | + | – |
| Блоки контроллеров синхронного последовательного порта (SPI) | + | – |
| Блоки контроллеров асинхронного последовательного порта (UART) | + | – |
| Контроллеры портов ввода/вывода общего назначения (GPIO) | + | – |
| Системный контроллер | + | – |
| Блок упаковки/распаковки векторов данных | – | + |
| Блок формирования временных шкал и синхронизации с внешними событиями (ITU) | + | + |
| Блок каналов предварительной обработки (DDC) | + | + |
| Блок декодеров сообщения (CORB) | + | + |
| Блок формирования данных для передачи в ЦАП (TRNSMT) | + | + |
| Байтовый коммуникационный порт (COM) | + | + |

Версия библиотеки классов С++ для системы ARM не предназначена для использования в операционной системе не реального времени (OS Linux).

**Заключение**

Использование СБИС 1888ВС058 для построения систем связи возможно с применением встроенных аппаратных каналов предварительной обработки сигнала, блока декодеров и блока формирования сигналов для передачи. В СБИС аппаратурно поддерживается канальный уровень связи с использованием посимвольной передачи сообщений c кодовым разделением абонентов. При использовании других протоколов предполагается обработка отсчётов на двух ядрах DSP NMC3. Для разработки ПО используется кроссплатформенные средства слоя аппаратных абстракций.

**Список литературы**

1. Варакин Л.Е. Системы связи с шумоподобными сигналами. Москва: «Радио и связь». Редакция литературы по радиотехнике, 1985.
2. ЗАО НТЦ «Модуль». Микросхема интегральная 1888ВC058. Руководство по эксплуатации.
3. "NMPP. Документация", ЗАО НТЦ «Модуль» [Электронный ресурс]. URL: http://rc-module.github.io/nmpp/modules.html.
4. "Arm DS-5 Development Studio documentation", Arm Ltd. [Электронный ресурс]. URL: https://developer.arm.com/tools-and-software/embedded/legacy-tools/ds-5-development-studio/docs.