

ТЕХНИЧЕСКИЙ ОТЧЕТ

по исследованиям и доработкам программного обеспечения для плат СЦ4 согласно
Техническому заданию по договору №2021-11-15 от 15.11.2021

Содержание

1. Доработка кода драйвера платы уи для обеспечения стабильного обмена с платой СЦ4 и кода тестового режима для обнаружения неисправностей плат УИ	3
2. Доработка программного кода настройки аналогового коммутатора аудиокодека на ввод звука через дополнительный аналоговый путь (внутри аудиокодека) для обеспечения переключения между встроенным и внешним микрофонами на плате СЦ4.	6
3. Реализация тонкой программной настройки модуля защиты памяти ядра микроконтроллера с контролем выравнивания доступа к данным при работе с буферами сетевых пакетов для обеспечения стабильной работы сетевого стека lwIP	9
4. Реализация программной настройки областей памяти в модуле защиты памяти ядра микроконтроллера платы СЦ4 с отключением кэширования областей памяти модуля DMA Ethernet-контроллера при включенном кэше данных для обеспечения стабильного обмена пакетами по сети Ethernet.	12
5. Исследование и оценка с помощью разработанного тестового кода времени микросхемы eeprom на плате СЦ4, и доработка ПО платы СЦ4 для обеспечения стабильной записи в EEPROM.	16
6. Проведение экспериментов по подключению пультов к различному сетевому оборудованию для получения программной информации по Rtu-контроллеру платы СЦ4 с целью выявления программно-аппаратных особенностей настройки контроллера на платах СЦ4.	18
7. Доработка настройки аудиокодека (фильтры, уровни усиления) для обеспечения возможности задания требуемых параметров качества звукового сигнала.	27
8. Доработка функционала ПО «Конфигуратор».	37

1. Доработка кода драйвера платы уи для обеспечения стабильного обмена с платой СЦ4 и кода тестового режима для обнаружения неисправностей плат УИ

Тесты в составе ПО платы СЦ4 разработанные согласно техническим требованиям не предусматривают тестирование взаимодействия платы с платой УИ, что не позволяет проверить исправность канала связи между платами и аппаратных компонент платы УИ при запуске программы платы СЦ4 и в процессе технического обслуживания.

Для обеспечения обнаружения неисправностей плат УИ и стабильного обмена между платой УИ и платой СЦ4 проведены следующие доработки кода:

1. Разработан тест микросхем драйверов светодиодов и микросхем регистров кнопок УИ – подпрограммы `ui_test_board()`, `ui_test_MCP()`, `ui_test_TLC()` в драйвере платы УИ, позволяющий выявить неисправности микросхем и нарушение обмена с платой УИ в процессе начальной аппаратной настройки пульта или в процессе работы.

2. В командном процессоре программы платы СЦ4 (`CLI_CmdTestFn`) разработан код и введена команда тестового режима `TEST UI` для тестирования обмена с платой платы УИ и исправности микросхем платы УИ.

3. В коде начального самотестирования платы СЦ4 введена проверка наличия подключения платы УИ и вывод в терминал результатов проверки, а также введена возможность работы с платой СЦ4 (запись/чтение конфигурации) при отсутствии подключенной платы УИ.

4. Разработан тест светодиодов платы УИ в модуле тестового режима, по алгоритму аналогичному тесту на стенде проверки плат УИ (команда `TEST LED : [ON|OFF]`), позволяющий выявить неисправности светодиодов платы УИ в процессе технического обслуживания.

5. Разработан тест кнопок платы УИ в модуле тестового режима, по алгоритму аналогичному тесту на стенде проверки плат УИ (команда `TEST BTN : [ON|OFF]`),

позволяющий выявить неисправности кнопок платы УИ в процессе технического обслуживания.

Команда TEST UI – выполняет детектирование наличия и типа платы УИ, тестирование всех микросхем драйверов светодиодов (TLC) и регистров кнопок (MCP) с выводом результата в терминал.

Пример вывода теста платы УИ:

```
Test UI board
Device type : PDO-16
Test TLC0 ok
Test TLC1 ok
Test MCP0 ok
Test UI end
```

Возможные сообщения об неисправностях:

Device type : UI not connected – плата УИ не подсоединена к плате СЦ4.

Device type : unknown UI – неизвестный код платы УИ .

Error write TLC<i> reg <r> – ошибка доступа на запись регистра <r> микросхемы драйвера светодиодов номер <i>.

Error read TLC<i> reg <r> – ошибка доступа на чтение регистра <r> микросхемы драйвера светодиодов номер <i>.

Error read TLC<i> data <r> – ошибка контроля содержимого регистра <r> микросхемы драйвера светодиодов номер <i>.

Error write MCP<i> reg <r> – ошибка доступа на запись регистра <r> микросхемы регистра кнопок номер <i>.

Error read MCP<i> reg <r> – ошибка доступа на чтение регистра <r> микросхемы регистра кнопок номер <i>.

Error read MCP<i> reg <r> wdata=<w> rdata=<e> – ошибка содержимого регистра <r> микросхемы регистра кнопок номер <i>, записаны данные <w>, а прочитаны ошибочные данные <e>.

Команда TEST LED : ON – включает тест визуального контроля исправности светодиодов платы УИ. При этом включаются сначала все зеленые светодиоды, потом через 1 сек все красные, и далее процесс повторяется для возможности визуального контроля исправности светодиодов.

Команда TEST LED : OFF – выключает тест светодиодов платы УИ

Команда TEST BTN : ON – включает тест визуального контроля исправности кнопок платы УИ. В данном режиме тестирования при нажатии кнопки включается светодиод на плате УИ – если нажата кнопка абонента, то зеленый светодиод, соответствующий данной кнопке, если одна из 12-ти кнопок телефонного набора, то красный светодиод по номеру кода (1-12). Также при нажатии кнопки выдается сообщение о нажатии кнопки по интерфейсу RS232 в COM-терминал, если плата СЦ4 подключена по интерфейсу RS232 к компьютеру.

Команда TEST BTN : OFF – выключает тест кнопок платы УИ.

2. Доработка программного кода настройки аналогового коммутатора аудиокодека на ввод звука через дополнительный аналоговый путь (внутри аудиокодека) для обеспечения переключения между встроенным и внешним микрофонами на плате СЦ4.

Согласно схеме ГРЛМ.468351.004 ЭЗ подключение микрофонов к аудиокодеку TLV320AIC3254 выполнено следующим образом (согласно рисунку 1):

- встроенный микрофон (MIC1) – IN1_L, IN1_R.
- внешний микрофон (MIC2) – IN2_L, IN2_R.

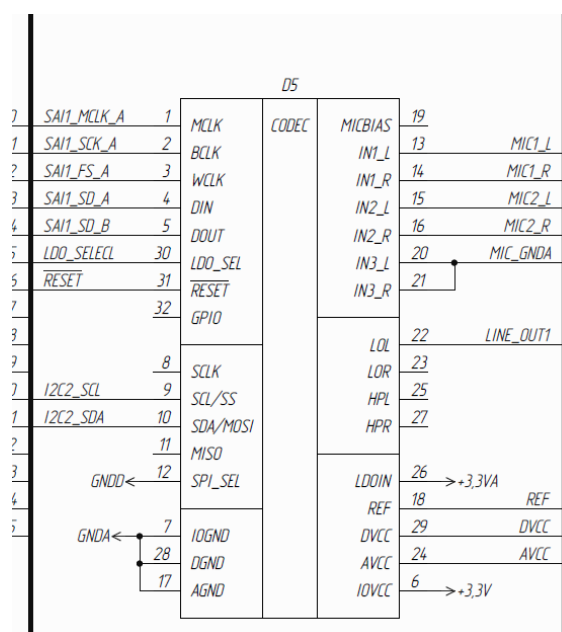


Рисунок 1 - Фрагмент схемы подключения микрофонов к аудиокодеку.

По результатам исследования вариантов подключения входных аудиосигналов по документации на программирование аналогового коммутатора аудиокодека TLV320AIC3254 было установлено, что отсутствуют варианты дифференциального подключения двух аналоговых аудиотрактов устройств с входов IN1 и IN2 на один цифровой аудиотракт.

Согласно рисунку 2 возможны следующие варианты подключения входов IN1 и IN2:

Для аудиотракта цифрового аудиотракта LeftADC:

$IN1L + CM1L$

$IN2L + IN2R$

Для аудиотракта цифрового аудиотракта RightADC:

$IN1L + IN1R$

$IN2R + CM2R$

Таким образом, один из входов должен иметь однополярное включение, что схемотехнически в данной схеме допустимо.

Во второй версии платы СЦ4 было рекомендовано подсоединить внешний микрофон к входу IN3, а внутренний к IN2 микросхемы аудиокодека.

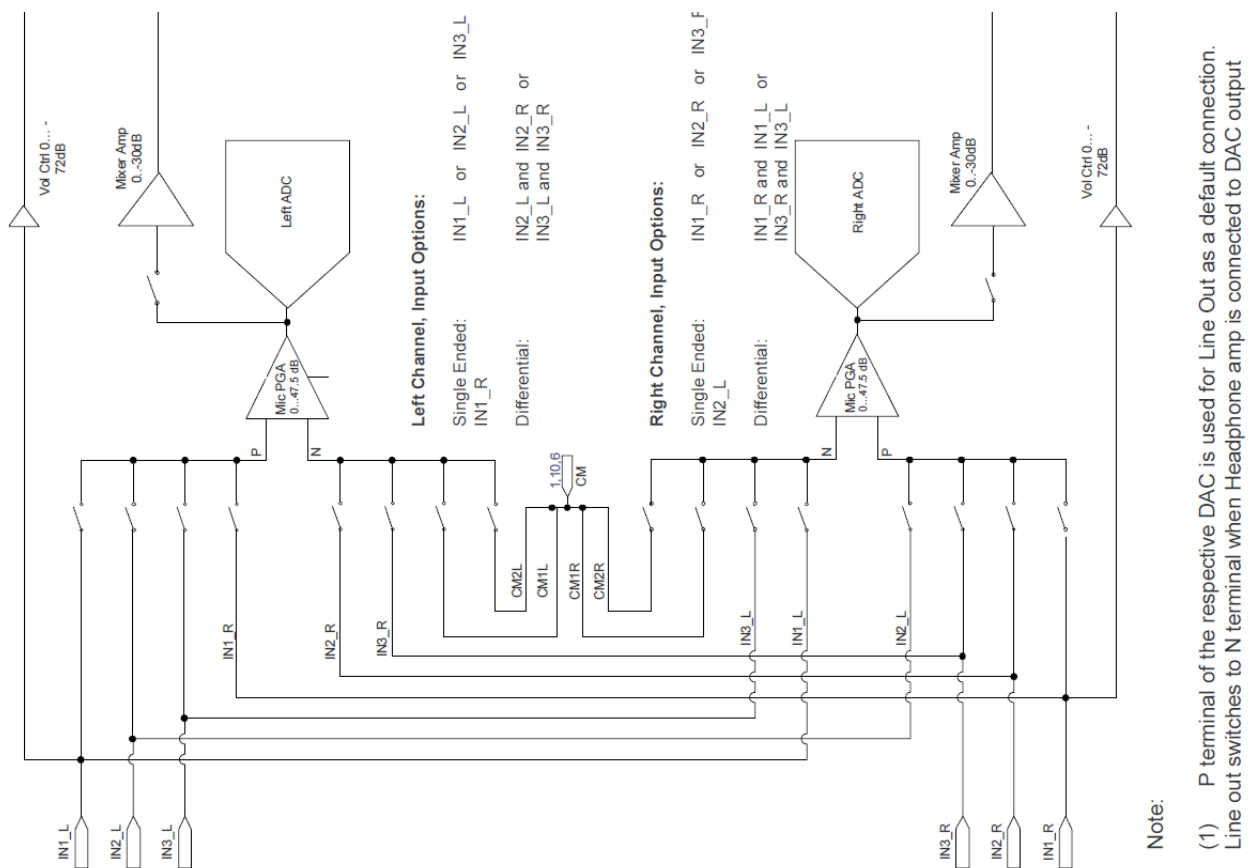


Рисунок 2 - Фрагмент схемы аналоговой коммутации сигналов аудиокодека

TLV320AIC3254 – коммутация входных аудиосигналов.

С учетом того, что для вывода звука в плате СЦ4 используется вывод LOL, который использует цифровой тракт LeftDAC, соответствующий тракту LeftADC, для обеспечения корректного обмена по аудиошине SAI между аудиокодеком и

микроконтроллером, в качестве входного аудиотракта необходимо использовать LeftADC.

Были сформированы настройки аналогового коммутатора аудиокодека на ввод звука через дополнительный аналоговый путь (CM1L) для первой версии платы.

Для подключения внутреннего микрофона необходимо установить регистры аудиокодека:

```
TLV320AIC3254_REG_LMICPGA_PMUX = 0x40
```

```
TLV320AIC3254_REG_LMICPGA_NMUX = 0x40
```

Для подключения внешнего микрофона значения регистров:

```
TLV320AIC3254_REG_LMICPGA_PMUX = 0x10
```

```
TLV320AIC3254_REG_LMICPGA_NMUX = 0x10
```

Для обеспечения настройки аналогового коммутатора аудиокодека на ввод звука с микрофонов и автоматического переключения между встроенным и внешним микрофонами на плате СЦ4 при подключении/отключении внешнего микрофона был доработан код драйвера аудиокодека (структура с настройками TLV320_ADCSetup_Seq, функция инициализации аудиокодека `aic_init`, функция переключения входных аудиоустройств `aic_setInDev`).

3. Реализация тонкой программной настройки модуля защиты памяти ядра микроконтроллера с контролем выравнивания доступа к данным при работе с буферами сетевых пакетов для обеспечения стабильной работы сетевого стека lwIP

Стек lwIP использует для формирования передаваемых по сети пакетов специальную область памяти `lwip_pool`, в которой формируются данные отправляемых пакеты и далее считываются из этой памяти контроллером DMA в составе Ethernet контроллера. Так как контроллер DMA работает только с выровненным на границу слова (32-бит) доступом к памяти, необходимо обеспечить генерацию кода программы при доступе к данной памяти только с использованием выравнивания на границу слова.

В существующем коде предоставленном от ООО «Группа промышленных технологий» в качестве основы для разработки ПО платы СЦ4, вышеуказанные требования обеспечивались размещением памяти в домене памяти D2 и отключением работы кэш памяти данных микроконтроллера, выровненный доступ не использовался.

В результате:

1. Значительно снижалась производительность Ethernet-контроллера.
2. При выполнении участка кода с командами чтения/записи с невыровненным доступом происходил аппаратный сбой (HardFault) микроконтроллера, что приводило к прекращению работы программы платы СЦ4.

В результате изучения документации, дополнительной информации на сайте производителя микроконтроллера STM32H753, для обеспечения стабильной работы сетевого стека lwIP, были выполнены следующие доработки программного кода:

1. Был доработан файл распределения памяти программы платы СЦ4 для размещения области памяти `lwip_pool` в памяти микроконтроллера, к которой имеет доступ контроллер DMA.

RAM_D2 (xrw) : ORIGIN = 0x30000000, LENGTH = 288K

...

```
.lwip_pool_space (NOLOAD) : {  
    . = ABSOLUTE(0x30040000);  
    *(.lwip_pool)  
    . = ABSOLUTE(0x30047FFF);  
}>RAM_D2
```

2. Реализован код настройки модуля защиты памяти ядра для данной области памяти. Область памяти настраивается как некэшируемая память, разделяемая между устройствами без буферизации:

```
MPU_InitStruct.Enable = MPU_REGION_ENABLE;  
MPU_InitStruct.Number = MPU_REGION_NUMBER1;  
MPU_InitStruct.TypeExtField = MPU_TEX_LEVEL0;  
MPU_InitStruct.AccessPermission = MPU_REGION_FULL_ACCESS;  
MPU_InitStruct.DisableExec = MPU_INSTRUCTION_ACCESS_ENABLE;  
MPU_InitStruct.IsShareable = MPU_ACCESS_SHAREABLE;  
MPU_InitStruct.IsCacheable = MPU_ACCESS_NOT_CACHEABLE;  
MPU_InitStruct.IsBufferable = MPU_ACCESS_NOT_BUFFERABLE;  
MPU_InitStruct.BaseAddress = 0x30040000;  
MPU_InitStruct.Size = MPU_REGION_SIZE_32KB;
```

3. Доработаны настройки компиляции стека lwIP компилятором GCC - применен ключ «-mno-unaligned-access», который обеспечивает генерацию кода и адресов переменных только с учетом выровненного на границу слова микроконтроллера доступом к памяти.

В результате выполненных доработок, тестирование передачи данных по сети через стек lwIP при выполнении программы платы СЦ4 показало стабильную работу сетевого стека lwIP.

4. Реализация программной настройки областей памяти в модуле защиты памяти ядра микроконтроллера платы СЦ4 с отключением кэширования областей памяти модуля DMA Ethernet-контроллера при включенном кэше данных для обеспечения стабильного обмена пакетами по сети Ethernet.

В программе платы СЦ4 необходимо обеспечить стабильный обмен пакетами по сети Ethernet на аппаратном уровне Ethernet-контроллера.

В результате изучения документации, дополнительной информации на сайте производителя микроконтроллера STM32H753, для обеспечения стабильной работы стабильного обмена пакетами по сети Ethernet через Ethernet-контроллер микроконтроллера необходимо, чтобы выполнялись следующие условия:

1. Все данные передаваемые между Ethernet-контроллером не должны располагаться в DTCM RAM, так как DMA контроллер не имеет к ней доступа. Стандартные проекты генерируемые при помощи среды STM32CubeMX используют DTCM RAM, поэтому необходимо дорабатывать файл распределения памяти для размещения данных в областях памяти доступных DMA контроллеру

2. Дескрипторы буферов приема (RX) и передачи (TX) DMA контроллера, кроме того что должны располагаться в памяти доступной DMA контроллеру, также должны быть настроены как память разделяемая между устройствами с выровненным доступом.

3. Буфер приема пакетов контроллера Ethernet должен иметь отделенный регион памяти от дескрипторов буферов приема и передачи должен быть настроен как память, разделяемая между устройствами

В существующем коде предоставленном от ООО «Группа промышленных технологий» в качестве основы для разработки ПО платы СЦ4, вышеуказанные требования обеспечивались только отключением работы кэш памяти данных микроконтроллера.

В результате:

1. Значительно снижалась производительность всего микроконтроллера.

2. При приеме пакетов наблюдалось частичное нарушение информации в принимаемых пакетах из-за коллизий операций чтения микроконтроллера и записи Ethernet-контроллера при доступе к приемному буферу Ethernet-контроллера.

3. Наблюдалась передача в сеть некорректных пакетов из-за нарушения информации в дескрипторах буферов DMA-контроллера при операциях обновления информации в них со стороны микроконтроллера и Ethernet-контроллера.

Для обеспечения стабильного обмена пакетами по сети Ethernet через Ethernet контроллер микроконтроллера, были выполнены следующие доработки программного кода:

1. Был доработан файл распределения памяти программы платы СЦ4 для размещения областей памяти дескрипторов DMA и приемного буфера Ethernet-контроллера во внутренней памяти микроконтроллера, к которой имеет доступ контроллер DMA.

```
RAM_D2 (xrw) : ORIGIN = 0x30000000, LENGTH = 288K
```

```
...
```

```
.eth_dma (NOLOAD) : {  
    . = ABSOLUTE(0x30000000);  
    *(.RxDecripSection)  
  
    . = ABSOLUTE(0x30000060);  
    *(.TxDecripSection)  
  
    . = ABSOLUTE(0x30000200);  
    *(.RxArraySection)  
} >RAM_D2
```

2. Реализован код настройки модуля защиты памяти ядра для дескрипторов буферов приема и передачи. Область памяти настраивается как некэшируемая память, разделяемая между устройствами без буферизации:

```
MPU_InitStruct.Enable = MPU_REGION_ENABLE;  
MPU_InitStruct.Number = MPU_REGION_NUMBER0;  
MPU_InitStruct.BaseAddress = 0x30000000;  
MPU_InitStruct.Size = MPU_REGION_SIZE_512B;  
MPU_InitStruct.SubRegionDisable = 0x0;  
MPU_InitStruct.TypeExtField = MPU_TEX_LEVEL0;  
MPU_InitStruct.AccessPermission = MPU_REGION_FULL_ACCESS;  
MPU_InitStruct.DisableExec = MPU_INSTRUCTION_ACCESS_ENABLE;  
MPU_InitStruct.IsShareable = MPU_ACCESS_SHAREABLE;  
MPU_InitStruct.IsCacheable = MPU_ACCESS_NOT_CACHEABLE;  
MPU_InitStruct.IsBufferable = MPU_ACCESS_NOT_BUFFERABLE;
```

3. Реализован код настройки модуля защиты памяти ядра для буферов приема Ethernet-контроллера. Область памяти настраивается как кэшируемая память, разделяемая между устройствами без буферизации:

```
MPU_InitStruct.Enable = MPU_REGION_ENABLE;  
MPU_InitStruct.Number = MPU_REGION_NUMBER2;  
MPU_InitStruct.BaseAddress = 0x30000000;  
MPU_InitStruct.Size = MPU_REGION_SIZE_16KB;  
MPU_InitStruct.SubRegionDisable = 0x0;  
MPU_InitStruct.TypeExtField = MPU_TEX_LEVEL1;  
MPU_InitStruct.AccessPermission = MPU_REGION_FULL_ACCESS;  
MPU_InitStruct.DisableExec = MPU_INSTRUCTION_ACCESS_ENABLE;  
MPU_InitStruct.IsShareable = MPU_ACCESS_SHAREABLE;  
MPU_InitStruct.IsCacheable = MPU_ACCESS_CACHEABLE;
```

```
MPU_InitStruct.IsBufferable = MPU_ACCESS_NOT_BUFFERABLE;
```

В результате выполненных доработок, тестирование передачи и приема данных по сети через Ethernet-контроллер при выполнении программы платы СЦ4 показало стабильный обмен сетевыми пакетами.

5. Исследование и оценка с помощью разработанного тестового кода времени микросхемы eeprom на плате СЦ4, и доработка ПО платы СЦ4 для обеспечения стабильной записи в EEPROM.

Согласно документации на микросхему памяти EEPROM M24M01 время записи составляет 5 ms (Рисунок 3, параметр t_w). Согласно документации по алгоритму работы с микросхемой в этот период доступ и обмен с микросхемой невозможен. Для проверки корректного выполнения процедуры записи информации микросхемой, согласно документации, необходимо выполнить опрос и обмен с микросхемой через данный интервал времени.

t_{DHDL}	t_{BUF}	Time between Stop condition and next Start condition	1300	-	ns
$t_{WLDL}^{(6)(1)}$	$t_{SU:WC}$	\overline{WC} set up time (before the Start condition)	0	-	μs
$t_{DHWL}^{(7)(1)}$	$t_{HD:WC}$	\overline{WC} hold time (after the Stop condition)	1	-	μs
t_w	t_{WR}	Internal Write cycle duration	-	5	ms
$t_{NS}^{(1)}$	-	Pulse width ignored (input filter on SCL and SDA) - single glitch	-	50 ⁽⁸⁾	ns

Рисунок 3 – Фрагмент характеристик микросхемы памяти EEPROM M24M01.

Результаты исследования работы микросхемы показали, что параметр t_w не соответствует реальному времени обмена с микросхемой при записи разных объемов данных с учетом разбиения EEPROM на страницы.

Для исследования и оценки времени обмена с микросхемой eeprom в модуле тестового режима был разработана функция `eeprom_test_writedelay`.

В командном процессоре программы платы СЦ4 (`CLI_CmdTestFn`) разработан код и введен режим команды тестирования EEPROM режима TEST EEPROM : TIME для тестирования времени записи микросхемы памяти EEPROM.

Пример вывода команды «TEST EEPROM : TIME»:

```
EEPROM byte write delay: 5
EEPROM block write delay: 28
EEPROM full write delay: 14332
```


Возможные сообщения об неисправностях:

Error EEPROM: write byte – ошибка доступа на запись отдельного байта в микросхему EEPROM.

Error EEPROM: write block – ошибка доступа на запись блока (страницы) данных в микросхему EEPROM.

На основе анализа работы микросхемы на ряде плат СЦ4 при помощи разработанного кода, при записи конфигурационных данных были получены следующие результаты:

Действие	Объем данных, байт	Время записи, мс
Запись одного байта в странице EEPROM	1	5
Запись одной страницы EEPROM	256	28
Запись всего объем EEPROM	256*512	14334

Таким образом, установлено, что:

1. Параметр t_w – это время записи одного байта, но не страницы.
2. Время записи страницы не согласуется с временем t_w , но согласуется с общим объемом EEPROM – $28*512 \sim 14336$.
3. Для реализации алгоритма в программе нужно использовать не параметр t_w , согласно документации, а установленное время записи страницы.

По полученным результатам была разработана функция `eeprom_waitwrite_polling`, обеспечивающая стабильный обмен с микросхемой при записи данных и доработана функция `eeprom_write_block` для записи блока данных в EEPROM.

6. Проведение экспериментов по подключению пультов к различному сетевому оборудованию для получения программной информации по Phy-контроллеру платы СЦ4 с целью выявления программно-аппаратных особенностей настройки контроллера на платах СЦ4.

Для обеспечения стабильности подключения пультов ПДО системы GIT-Comm IPS к различному сетевому оборудованию, исключения задержек передачи речевого сигнала и, как следствие, ухудшения разборчивости речи необходимо провести исследования с применением различного сетевого оборудования. Также требуется проверить работоспособность пультов в случае использования разных подсетей сети Ethernet.

Для тестирования пультов ПДО системы GIT-Comm IPS применялось следующее аппаратное обеспечение:

1. Персональный компьютер на базе процессора AMD FX-8370, операционная система Windows 7.
2. Персональный компьютер на базе процессора Intel Core i3-8100, операционная система Windows 7.
3. Программируемый маршрутизатор D-Link SmartPro Switch DGS-1510-20, 2 шт.
4. Роутер Keenetic Giga II.
5. Роутер Keenetic Start KN-1111-01RU.
6. Роутер Keenetic Giga KN-1011.
7. Роутер TP-LINK TL-WR1043ND v1.
8. Коммутатор Eltex MES2324P.
9. Неуправляемый коммутатор TP-Link TL-SF1008P.
10. Витая пара стандарта UTP категории 5.

Исследовалась стабильность установки сетевого подключения интерфейса Ethernet пультов с сетевым оборудованием. Для этого осуществлялось подключение пультов с использованием кабелей длиной от 0,5 до 10 метров. В ходе проведенных экспериментов установлено, что на перечисленном выше оборудовании установка

соединения происходит стабильно вне зависимости от длины кабеля и используемого устройства.

Далее проводилось исследование стабильности работы установленного соединения. Для этого пульт, подключенный к оборудованию, находился постоянно подключенным к сети, при этом производилась постоянная проверка наличия сетевого подключения с помощью команды PING с персонального компьютера и сервисных команд TEST PHY : STATUS и TEST PHY : PING пульта. Эксперимент проводился 30 минут для каждого вида оборудования. В ходе эксперимента не было выявлено случаев самопроизвольного разрыва связи или сбоев передачи данных.

Проводился эксперимент по взаимодействию с пультами ПДО в пределах существующей локальной сети. Локальная сеть имеет следующие параметры:

Количество оконечных устройств в сети: ~50.

Диаметр сети: ~20 метров.

Количество промежуточных устройств между пультами: минимум 2.

В таких условиях проводилась проверка работоспособности пультов в стандартных сценариях использования: установка связи, голосовое общение в различных режимах связи, конфигурирование и обновление прошивки по локальной сети. В ходе проведенных экспериментов не было выявлено сбоев или помех в работе пультов или сетевого оборудования.

При передаче данных ПО пультов ПДО формирует следующие массивы данных:

1. Пакеты командного протокола: максимальный размер информационной части пакета 1400 байт, заголовок UDP – 8 байт, заголовок IP – 20 байт.

2. Пакеты голосовых сообщений: размер информационной части пакета 160 байт, заголовок RTP – 12 байт, заголовок UDP – 8 байт, заголовок IP – 20 байт.

Пакеты командного протокола передаются асинхронно, поэтому не создают помех в работе локальной сети Ethernet. Для обеспечения работоспособности системы присутствует синхронный обмен между пультами пакетам типа IPSPING-IPSPONG, которые обеспечивают передачу информации о возможности установки

связи с пультами. Данные пакеты имеют размер информационной части 48 байт, общий размер пакета, включая заголовки UDP и IP, составляет 76 байт. Пакет IPSPING отправляется каждые 0,5 с для каждого имеющегося абонента, максимальное количество для одного пульта – 48. Информационный поток для одного пульта при этом составляет

$$(76 \text{ байт} * 8 / 0,5 \text{ сек}) * 48 = 58368 \text{ бит/сек или } 58,4 \text{ кбит/сек}$$

Время ответа на пакет IPSPING должно составлять не более 500 мс, поэтому латентность локальной сети не должна превышать данного параметра.

Пакет RTP имеет размер информационной части 160 байт, общий размер пакета, включая заголовки RTP, UDP и IP, составляет 200 байт. Звук с микрофона считывается с разрядностью 8 бит и частотой дискретизации 8000 Гц. За 1 секунду передается $8 * 8000$ бит или 8000 байт, т.е. $8000 \text{ байт} / 160 \text{ байт} = 50 \text{ пакетов/сек}$. Передача пакетов RTP осуществляется с периодом $1/50 = 20 \text{ мс}$, соответственно после установки связи информационный поток для одного пульта составляет

$$200 \text{ байт} * 8 * 50 \text{ пакетов/сек} = 80000 \text{ бит/сек или } 80 \text{ кбит/сек}$$

Соответственно для дуплексной связи информационный поток составляет $80 \text{ кбит/сек} * 2 = 160 \text{ кбит/сек}$.

При передаче RTP-пакетов должна быть обеспечена латентность локальной сети не более 20 мс для исключения ухудшения разборчивости принимаемого речевого сигнала.

Общая требуемая пропускная способность локальной сети при обеспечении дуплексной связи между двумя пультами и отсутствии других абонентов:

$$58,4 \text{ кбит/сек} * 2 + 80 \text{ кбит/сек} * 2 = 276,8 \text{ кбит/сек}.$$

В ходе проведенных исследований были составлены следующие рекомендации по настройке сетевого оборудования для исключения прерывания произносимых фраз из-за сбоев передачи сетевых потоков:

Пропускная способность локальной сети Ethernet при обеспечении дуплексной связи между двумя пультами и отсутствии других абонентов: не менее 276,8 кбит/сек.

Латентность локальной сети Ethernet: не менее 20 мс.

Вышеуказанные параметры нужно учитывать при выборе и настройке сетевого оборудования и проектировании локальной сети.

Далее проводилось исследование работы пультов ПДО системы GIT-Comm IPS в двух разных подсетях.

В ходе испытаний использовалось следующее оборудование:

- а) коммутатор PoE – MES2324P “Eltex”;
- б) роутер, например Keenetic Start KN-1111-01RU;
- в) пульт №1 – ПДО-16;
- г) пульт №2 – ПДО-32;
- д) пульт №3 – ПДО-16Н.
- е) витая пара стандарта UTP категории 5 длиной до 10 метров.

Ниже приведен пример настройки оборудования для данного эксперимента.

Для проведения испытаний необходимо определить два диапазона используемых подсетей. В примере настройки будут использоваться подсети 192.168.66.0 (подсеть №1 – основная) и 192.168.2.0 (подсеть №2 – дополнительная). Необходимо также определить статический адрес из подсети №1, который будет являться внешним для подсети №2. В примере настройки таким адресом является 192.168.66.254. В подсети №2 нужно определить адрес шлюза – в примере настройки – 192.168.2.1.

Далее необходимо настроить пульта с помощью ПО «Конфигуратор» следующим образом:

1. Пульт №1 и пульт №2: настроить сетевой адрес из подсети №1, маску подсети задать согласно параметрам сети, шлюз – выделенный статический адрес (в данном примере – 192.168.66.254)

2. Пульт №3: настроить сетевой адрес из подсети №2, маску подсети – согласно параметрам сети, шлюз – шлюз подсети №2 (в данном примере – 192.168.2.1).

3. Настроить коммутационный план проверяемых пультов следующим образом: на кнопку №1 пульта №1 назначить вызов пульта №2 в режиме дуплекса с автоподтверждением, на кнопку №2 пульта №1 назначить вызов пульта №3 в

режиме дуплекса с автоподтверждением. Пульты №2 и №3 настроить аналогичным образом на пульт №1 и друг на друга.

Настройка пультов должна осуществляться с применением подключения по RS232.

Соединить пульты в соответствии со схемой, приведенной на рисунке 4.

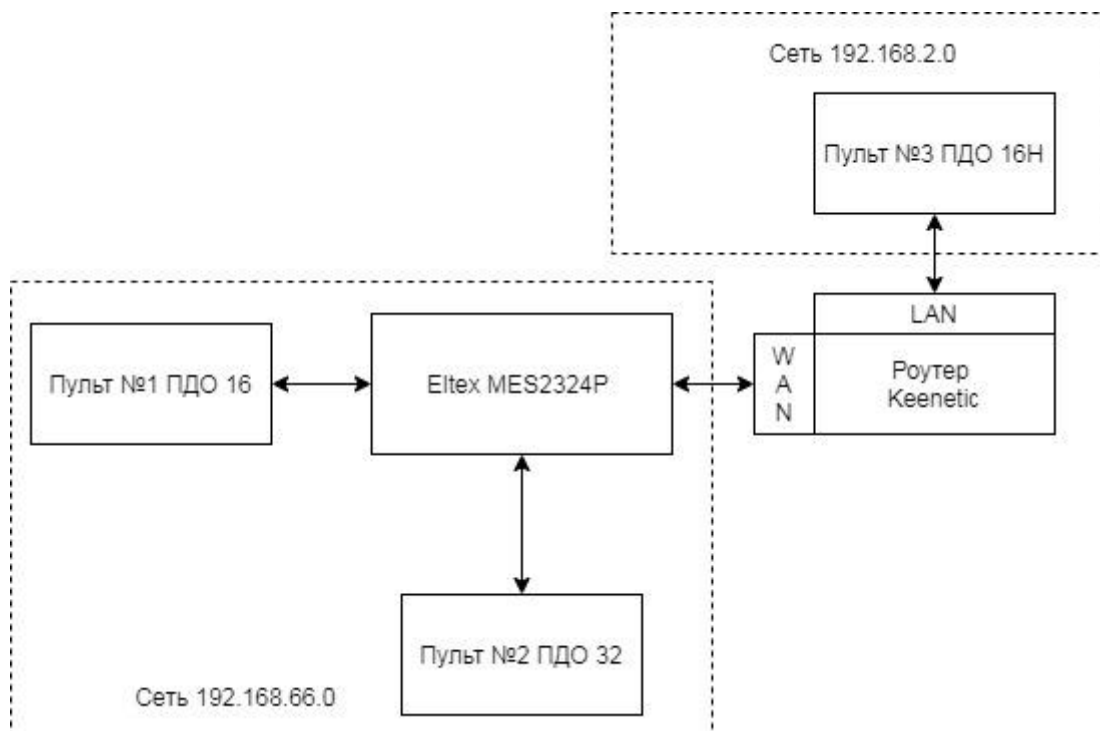


Рисунок 4 – Схема соединения пультов ПДО для проведения проверки

Далее необходимо настроить роутер для объединения двух подсетей. Настройка будет показана на основе интерфейса KeenOS роутеров семейства Keenetic. Настройка роутеров других производителей осуществляется согласно их документации.

Пример настройки роутера:

1. Необходимо настроить роутер на работу с подсетью 192.168.2.0, для этого перейти в раздел «Мои сети и WiFi → Домашняя сеть».

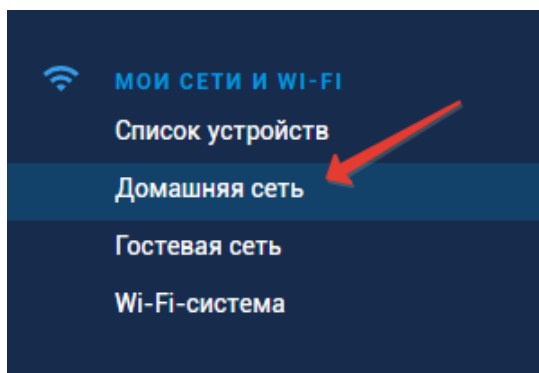


Рисунок 5 – Раздел «Мои сети и WiFi → Домашняя сеть»

2. В разделе «параметры IP» задать настройки как на рисунке 6 и нажать кнопку «Сохранить».

Параметры IP

Укажите параметры IP сегмента «Домашняя сеть».

IP-адрес	<input type="text" value="192.168.2.1"/>
Маска подсети	<input type="text" value="255.255.255.0"/> ▼
DHCP-сервер	<input checked="" type="radio"/> Включен <input type="radio"/> Выключен <input type="radio"/> Relay <input style="float: right;" type="button" value="?"/>

Скрыть настройки DHCP

Начальный адрес пула	<input type="text" value="192.168.2.33"/>
Размер пула адресов	<input type="text" value="120"/>
Время аренды	<input type="text" value="25200"/> секунд
Адрес шлюза	<input type="text"/>
Сервер DNS 1	<input type="text"/>
Сервер DNS 2	<input type="text"/>

Использовать NAT ☒ Для подключения устройств сегмента к интернету

Рисунок 6 – Пример настройки домашней подсети

3. Для настройки внешнего сетевого интерфейса WAN необходимо перейти в раздел «Интернет → Проводной».

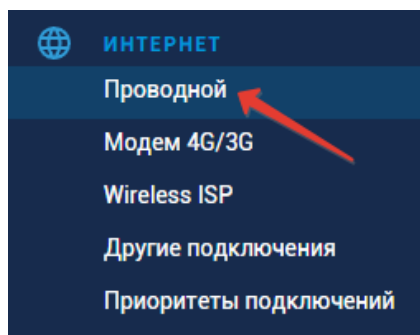


Рисунок 7 – Раздел «Интернет → Проводной»

4. В разделе «Параметры IP и DNS» необходимо выставить параметр «Настройка IP» в положение «Ручной» и задать настройки внешнего IP адреса из подсети №1. Пример настройки приведен на рисунке 8. Для завершения настройки необходимо нажать кнопку «Сохранить».

Параметры IP и DNS

Настройка IP	Ручная
IP-адрес	192.168.66.254
Маска подсети	255.255.255.0
Основной шлюз	
DNS 1	

Рисунок 8 – Пример настройки внешнего сетевого интерфейса WAN

5. Далее необходимо настроить маршрутизацию между двумя подсетями, для чего перейти в раздел «Сетевые правила → Маршрутизация».

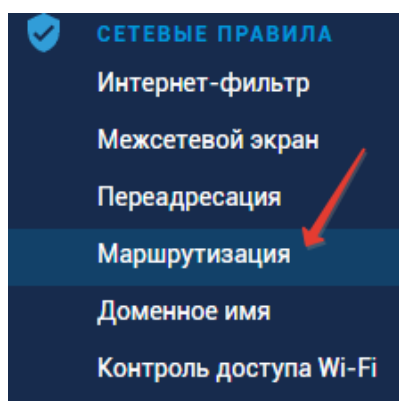


Рисунок 9 – Раздел «Сетевые правила → Маршрутизация»

6. В разделе «Сетевые правила → Маршрутизация» необходимо в подразделе «Пользовательские маршруты» нажать кнопку «Добавить маршрут» и перейти в меню добавления нового маршрута.

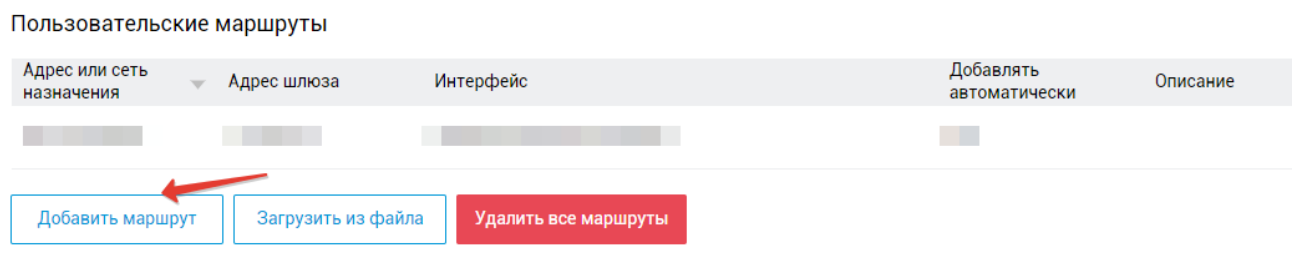


Рисунок 10 – Раздел «Интернет → Проводной»

7. Необходимо установить пункт «Тип маршрута» в положение «Маршрут до сети» и далее настроить маршрутизацию. В поле «Адрес сети назначения» указывается подсеть №1, в поле «Маска подсети» – заданная маска подсети №1, в поле «Адрес шлюза» – адрес внешнего сетевого интерфейса. Остальные параметры выставить согласно рисунку 11. Для подтверждения настроек необходимо нажать кнопку «Сохранить».

Параметры статического маршрута

Выберите тип маршрута и укажите адрес назначения. Введите адрес шлюза или выберите интерфейс, через который будет направляться соответствующий трафик.

Тип маршрута	Маршрут до сети
Описание	
Адрес сети назначения	192.168.66.0
Маска подсети	255.255.255.0
Адрес шлюза	192.168.66.254
Интерфейс	Любое
Добавлять автоматически	<input checked="" type="checkbox"/> ?

Рисунок 11 – Настройка маршрута между двумя подсетями.

8. Проверить с устройства, подключенного к роутеру и находящемся в подсети №2 доступность подсети №1, используя команду PING и существующий IP-адрес подсети №1. При отсутствии ответа проверить корректность проведенных настроек.

Вышеуказанные испытания проводились согласно документу «Методика проверки работы пультов серии ПДО изделия GIT-Comm IPS в разных подсетях локальной сети Ethernet».

В ходе проведения данного эксперимента согласно прилагаемой методике было показано, что разработанное ПО плат СЦ4 полностью обеспечивает взаимодействие пультов ПДО системы GIT-Comm IPS при подключении к разным подсетям локальной сети.

Таким образом, в результате выполнения работ были определены рекомендации по настройке сетевого оборудования для исключения прерывания произносимых фраз из-за сбоев передачи сетевых потоков, проведен ряд экспериментов по подключению пультов к различному сетевому оборудованию, которые показали, что программное обеспечение пультов ПДО системы GIT-Comm IPS обеспечивает стабильное сетевое подключение по интерфейсу Ethernet и дополнительной настройки контроллера Ethernet на платах СЦ4 не требуется. Также было проведено экспериментальное исследование взаимодействия пультов ПДО системы GIT-Comm IPS при подключении к разным подсетям локальной сети, которое показало, что разработанное ПО обеспечивает бесперебойную связь между пультами, находящимися в разных подсетях локальной сети Ethernet.

7. Доработка настройки аудиокодека (фильтры, уровни усиления) для обеспечения возможности задания требуемых параметров качества звукового сигнала.

Необходимо обеспечить следующие параметров качества звукового сигнала:

- отсутствие эффекта звуковой петли в режиме дуплекс;
- при устранении эффекта звуковой петли исключить обрывание речевых фраз при уровнях громкости и чувствительности по умолчанию (при включении пультов ПДО).

- уверенный прием речевого сигнала с расстояния до 50 см от микрофона;

Вышеуказанные параметры качества проверяются согласно документу «Методика проверки качества передачи речевого сигнала пультами серии ПДО изделия GIT-Comm IPS», представленному в Приложении 1 к Дополнительному соглашению № 1 к Договору № 2021-11-15 от 15 ноября 2021 г.

Согласно техническим требованиям ТЗ этапа 1 разработки ПО платы СЦ4 параметры уровни усиления входа и выхода были заданы следующим образом:

- "- регулировка громкости должна обеспечивать регулировку в диапазоне от - 21 дБ до + 3 дБ с шагом регулировки 3 дБ относительно уровня 0 дБ громкости;

- регулировка чувствительности должна обеспечивать регулировку в диапазоне от -9 дБ до + 6 дБ с шагом регулировки 1,5 дБ относительно уровня 0 дБ чувствительности"

Таким образом, задаются следующие значения параметров:

Параметр	Значение
Уровень громкости	
диапазон шагов	от -7 до +1
число уровней	9
шаг регулировки	3 дБ
диапазон усиления ЦАП аудиокодека	от -21 дБ до +3 дБ
Уровень чувствительности	
диапазон шагов	от -6 до +4
число уровней	10
шаг регулировки	1,5 дБ
диапазон усиления АЦП аудиокодека	от -9 дБ до +6 дБ

В ходе промежуточных испытаний на этапе 1 по предложению специалистов ООО «ГИТ» были расширены диапазоны регулировки и установлены следующие параметры уровней:

Параметр	Значение
Уровень громкости	
диапазон шагов	от -7 до +4
число уровней	12
шаг регулировки	3 дБ
диапазон усиления ЦАП аудиокодека	от -21 дБ до +12 дБ
Уровень чувствительности	
диапазон шагов	от -8 до +8
число уровней	13
шаг регулировки	1,5 дБ
диапазон усиления АЦП аудиокодека	от -12 дБ до +12 дБ

Дальнейшие испытания показали, что такие диапазоны могут вызывать перегрузки в аналоговой части платы по входу и выходу.

Специалистами ООО «ГИТ» было предложено сократить размер шага и количество диапазонов:

Параметр	Значение
Уровень громкости	
диапазон шагов	от -6 до +2
число уровней	9
шаг регулировки	1,5 дБ
диапазон усиления ЦАП аудиокодека	от -9 дБ до +3 дБ
Уровень чувствительности	
диапазон шагов	от -6 до +3
число уровней	10
шаг регулировки	1 дБ
диапазон усиления АЦП аудиокодека	от -6 дБ до +3 дБ

ООО «ЭТС» были проведены экспериментальные исследования пультов на каждом уровне громкости/чувствительности установлено:

1. Данный размеры шага (1,5 и 1) как громкости, так и чувствительности слишком мал, слабо различим на слух.

2. Диапазон регулировки громкости не позволяет ослабить звук на громкоговорителе для приемлемого уровня при тихой обстановке.

3. Диапазон регулировки чувствительности, не позволяет регулировать чувствительность с уровнями хорошо различимые на слух. Наиболее проявляется при увеличении чувствительности.

Для устранения вышеуказанных проблем, были сокращены диапазоны регулировки при сохраненном шаге:

Параметр	Значение
Уровень громкости	
диапазон шагов	от -6 до +2
число уровней	9
шаг регулировки	3 дБ
диапазон усиления ЦАП аудиокодека	от -18 дБ до +6 дБ
Уровень чувствительности	
диапазон шагов	от -6 до +6
число уровней	13
шаг регулировки	1,5 дБ
диапазон усиления АЦП аудиокодека	от -9 дБ до +9 дБ

Данные параметры позволяют:

1. Обеспечить изменение громкости осязаемое на слух.
2. Исключить перегрузку выхода.
3. Ослабить выходной звук на громкоговорителе для приемлемого уровня при тихой обстановке.
4. Исключить перегрузку входа (эффект "дребезжания") при слишком большом уровне чувствительности микрофона.
5. Исключить малые значения чувствительности, при которых для различимой передачи надо произносить речь на расстоянии не более 1-5 см от микрофона, и которые не используются на практике.

При необходимости дальнейшего изменения уровней, можно внести изменения в исходном коде модуля driver_AIC (файл driver_AIC.h):

Параметр	Описание	Текущее значение
VOL_OUT_MIN	Нижняя граница уровня усиления ЦАП аудиокодека (громкость)	-36
VOL_OUT_MAX	Верхняя граница уровня усиления ЦАП аудиокодека (громкость)	12

Параметр	Описание	Текущее значение
VOL_MIC_MIN	Нижняя граница уровня усиления АЦП аудиокодека (чувствительность)	-18
VOL_MIC_MAX	Верхняя граница уровня усиления АЦП аудиокодека (чувствительность)	18
VOL_OUT_STEP	Шаг изменения уровня усиления ЦАП аудиокодека (громкости)	6
VOL_MIC_STEP	Шаг изменения уровня усиления АЦП аудиокодека (чувствительности)	3

Примечание: каждый параметр задается в единицах дБ умноженных на два, так как шаг изменения уровней ЦАП/АЦП в аудиокодеке равен 0,5 дБ.

Также в ходе исследования уровней настроек усиления микрофонного тракта в аудиокодеке было установлено, что для обеспечения уверенного приема с расстояния 50см от источника звука до микрофона, необходимо увеличить предусиление сигнала в аудиокодеке на входе АЦП до 22...23,5 дБ. Для этого в коде инициализации аудиокодека регистр `TLV320AIC3254_REG_LMICPGA_VOL` инициализируется значением `0x2F`, что соответствует 23,5дБ.

Для подавления эффекта звуковой петли в режиме дуплекс был разработан алгоритм и программный код модуля `SELC_sound_process`, реализующий фильтрацию входных и выходных сигналов с учетом текущего уровня сигнала, обеспечивающий приглушение сигнала входного или выходного сигнала на одном из пультов при встречном разговоре.

Схема модуля подавления эффекта звуковой петли в режиме связи дуплекс приведена на рисунке 12.

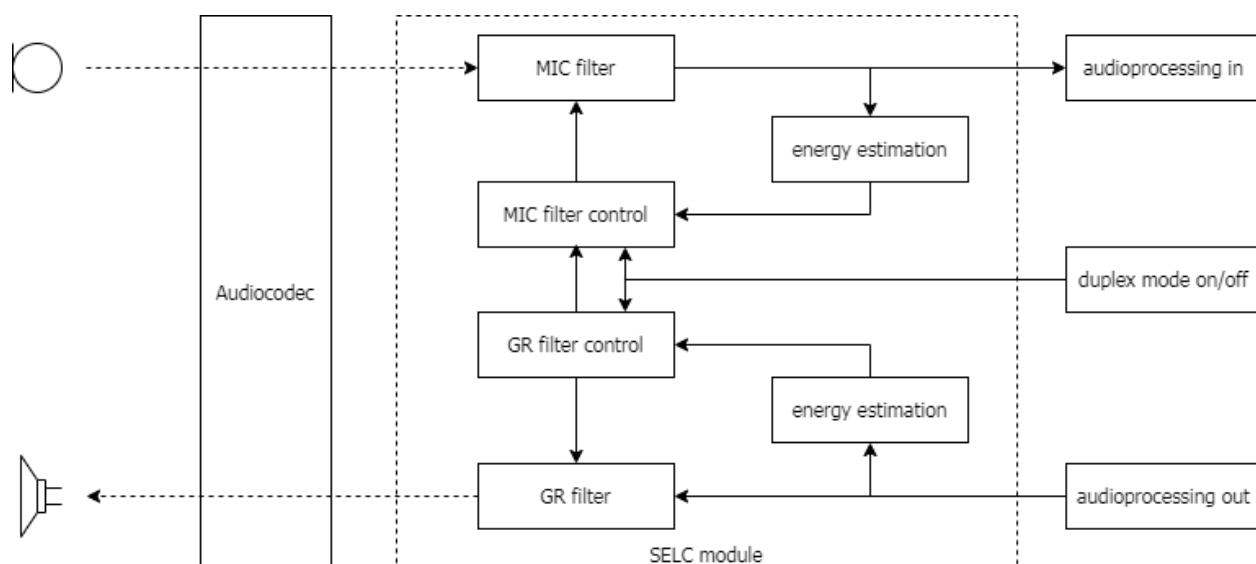


Рисунок 12 – Структурная схема программного модуля подавления эффекта звуковой петли

Блоки «energy estimation» оценивают текущий уровень входного и выходного сигнала и ослабляют соответствующий сигнал, при этом приоритет имеет выходной сигнал, то есть при наличии выходного сигнала выше порогового значения, ослабляется входной сигнал, а входной сигнал не ослабляется только при уровне выходного сигнала ниже порогового.

В алгоритме подавления эффекта звуковой петли используются следующие параметры:

ес_enable – ECEN, флаг включения алгоритма подавления эффекта звуковой петли.

ес_GR_threshold - GRT, уровень детектора сигнала на громкоговоритель, минимальное значение $GRT_{min} = 0$, максимальное значение $GRT_{max} = SAI_DMA_BUFFER_SIZE * MAX_INT16^2 / 256 = 160 * 2^{15} * 2 / 256 = 671088640$, рекомендуемый уровень для платы СЦ4: 80000-1000000.

ес_MIC_threshold - MCT, уровень детектора сигнала с микрофона, определяет как минимальное значение $MCT_{min} = 0$, максимальное значение $MCT_{max} = SAI_DMA_BUFFER_SIZE * MAX_INT16^2 / 256 = 160 * 2^{15} * 2 / 256 = 671088640$, рекомендуемый уровень для платы СЦ4: 300000-1000000.

ес_fade_level - уровень ослабления сигналов для фильтров, диапазон FDL = 1..12, коэффициент ослабления по амплитуде определяется как $1/(2^{FDL})$.

ес_MIC_disable_time - время включение громкоговорителя при детектировании звукового сигнала, диапазон MDT = 1..100, время задается как MDT*20мс.

ес_GR_enable_time - время включение громкоговорителя при детектировании звукового сигнала, диапазон GRET = 1..100, время задается как GRET*20мс.

ес_GR_disable_time - - время включение громкоговорителя при детектировании звукового сигнала, диапазон GRDT = 1..100, время задается как GRDT*20мс.

При необходимости исследования и задания параметров модуля можно использовать специальные команды COM-терминала тестового режима.

Команда TEST GET ЕС позволяет получить информацию о текущих настройках алгоритма и уровнях.

Пример вывода команды:

```
Audioloop disabler param:
ELC enabled      : 1
GR Threshold     : 500000
MIC Threshold    : 500000
Fade level       : 4
MIC dis time     : 10
GR en time       : 1
GR dis time      : 10

MAX MIC level    : 10000
MAX GR level     : 0
```

MAX MIC level – текущая максимальная оценка энергии сигнала с микрофона с момента прошлого исполнения команды. Может использоваться для оценки фонового уровня сигнала на микрофоне и выбора требуемого уровня детектора сигнала с микрофона.

MAX GR level – текущая максимальная оценка энергии сигнала на громкоговоритель с момента прошлого исполнения команды. Может использоваться

для оценки уровня входного сигнала принимаемого по сети и выбора требуемого уровня детектора сигнала на громкоговоритель.

Команда TEST SET EC позволяет задать настройки алгоритма подавления эффекта звуковой петли.

Формат команды:

TEST SET EC : <ECEN> : <GRT> : <MCT> : <FDL> : <MDT> : <GRET> :
<GRDT>

При выполнении команды производится вывод в терминал задаваемых параметров, также как в команде TEST GET EC.

Пример команды:

TEST SET EC : 1 : 500000 : 400000 : 4 : 10 : 1 : 10

Так как регулировка чувствительности аппаратно выполняется на входе до применения к входному сигналу алгоритма подавления звуковой петли, для исключения обрывания речевых фраз при заданном уровне чувствительности, необходимо выполнять автоматическое задание параметров порогового уровня детектора сигнала с микрофона.

Для оценки требуемого порогового уровня проведены экспериментальные исследования параметров принимаемого сигнала на расстоянии 50 см с разными уровнями чувствительности. Оценка проводилась при измерении фонового шума, а также источника тестовых сигналов с контролем значения MAX MIC level через СОМ-терминал по методике «Методика проверки качества передачи речевого сигнала пультами серии ПДО изделия GIT-Comm IPS», представленному в Приложении 1 к Дополнительному соглашению № 1 к Договору № 2021-11-15 от 15 ноября 2021 г.

Оценка энергии входного сигнала в цифровых единицах выполняется по формуле:

$$E_{est} = \frac{\sum_{i=0}^{SAI_BUF-N} sin^2(i)}{256},$$

где $sin(i)$ – отсчеты входного сигнала в формате int16,

SAI_BUF_N – количество отчетов в буфере DMA интерфейса SAI аудиокодека, равное 160, соответствующее отрезку сигнала 20мс.

При этом получаемое значение E_{est} пропорционально энергии сигнала на входе с коэффициентом 160/256.

В ходе экспериментальных исследований, было установлено, что для обеспечения работы алгоритма без обрывания речевых фраз при источнике звука на расстоянии до 50 см при настройках чувствительности по умолчанию при 0 дБ пороговое значение L_{thmic0} должно соответствовать E_{est} при $E_{est} = 400000$.

Пороговые уровни детектора для остальных уровней чувствительности микрофона могут быть рассчитаны по формуле:

$$L_{thmicL} = L_{thmic0} \cdot 10^{\frac{L \cdot L_{st}}{10}},$$

где L – требуемый уровень,

L_{st} – шаг изменения уровня в дБ,

L_{thmic0} – пороговое значение для уровня 0 дБ.

Были рассчитаны уровни порога детектора с учетом текущего шага регулировки чувствительности ($L_{st} = 1,5$ дБ) для требуемых уровней:

Уровень	Усиление, дБ	Порог
0	0	400000
-1	-1,5	283200
-2	-3	200505
-3	-4,5	141957
-4	-6	100506
-5	-7,5	71158
-6	-9	50380
+1	1,5	35669
+2	3	25253
+3	4,5	564971
+4	6	797982
+5	7,5	1127094
+6	9	1591940

Рассчитанные значения используются в программе платы СЦ4 в модуле SELC_sound_process из массива es_MIC_threshold_param для автоматического

задания параметров порогового уровня детектора сигнала с микрофона `es_MIC_threshold`.

Также были проведены эксперименты по использованию АРУ (AGC) аудиокодека. В программе платы СЦ4 реализован код настройки АРУ на значения по умолчанию для платы СЦ4. При необходимости исследования возможности использования АРУ возможно включение и задание настроек через COM-терминал.

Для включения, чтения и задания параметров АРУ используется команда терминала отладочного режима `TEST SET AIC` и `TEST GET AIC`.

Значения настройки АРУ подобранные на основе экспериментальных исследований:

Идентификатор регистра в коде программы	Номер регистра	Значение регистра	Параметры	Значение параметра
TLV320AIC3254_ REG_LADC_AGC_CR1	0x56	0x11	AGC_TargetLevel AGC_GainHysteresis	-8.0dBFS 1.0dB
TLV320AIC3254_ REG_LADC_AGC_CR2	0x57	0x1E	AGC_Hysteresis AGC_Noise_Threshold	1.0dB 0x1E = 30 = -60 dB
TLV320AIC3254_ REG_LADC_AGC_CR3	0x58	0x2F	AGC_MAX_PGAgain	+23.5dB
TLV320AIC3254_ REG_LADC_AGC_CR4	0x59	0x08	AGC_Attack_Time	3*32 clk = 96 /8000 = 12ms
TLV320AIC3254_ REG_LADC_AGC_CR5	0x5A	0x00	AGC_Decay_Time	1*512 clk = 512 /8000 = 64ms
TLV320AIC3254_ REG_LADC_AGC_CR6	0x5B	0x09	AGC_Noise_Debounce	1024 clk =1024 /8000 = 128ms
TLV320AIC3254_ REG_LADC_AGC_CR7	0x5C	0x06	AGC_Signal_Debounce	64 clk = 64 /8000 = 8 ms

Примечание: все регистры находятся на странице 0 регистрового файла аудиокодека.

Для включения AGC необходимо установить старший бит в регистре TLV320AIC3254_REG_LADC_AGC_CR1. При этом необходимо учитывать, что остальные биты в данном регистре задают параметры AGC согласно документации на аудиокодек (AGC_TargetLevel, AGC_GainHysteresis).

Для включения используется команда (при AGC_TargetLevel = -8.0dBFS, AGC_GainHysteresis = 1.0dB):

TEST SET AIC : 56 : 91

Для выключения AGC используется команда:

TEST SET AIC : 56 : 11

Экспериментальные исследования показали, что модель примененного в пультах микрофона имеет высокую чувствительность к шумам и слабым посторонним звукам. Полученный порог детектора микрофона, обеспечивающий устранение эффекта звуковой петли, а также параметр AGC_Noise_Threshold APY аудиокодека необходимо устанавливать достаточно высоким, что не позволяет передавать тихую речь. Рекомендуется применить в пультах микрофоны с повышенной шумозащитой, что позволит, кроме ослабления шума и посторонних звуков, снизить порог детектора и обеспечить более широкий диапазон уровней передачи речевого сигнала, а также полноценно использовать APY аудиокодека для реализации автоматического усиления.

Таким образом, в результате выполнения работ были определены уровни усиления с исключением перегрузок в аналоговой части платы по входу и выходу и обеспечивающие прием сигнала на микрофон с расстояния до 50 см, разработан программный код для устранения эффекта звуковой петли в режиме дуплекс и рассчитаны параметры алгоритма для исключения обрывания речевых фраз, что в совокупности обеспечивает требуемые параметры качества звукового сигнала.

8. Доработка функционала ПО «Конфигуратор».

1. Доработка функционала ПО «Конфигуратор» для добавления типов поддерживаемых пультов в конфигураторе в соответствии с типами приборов по ТЗ;

Согласно ТЗ был актуализирован список поддерживаемых устройств для настройки типа пультов в ПО «Конфигуратор». Новый список приведен ниже:

- Пульт ПДО-16
- Пульт ПДО-16Н
- Пульт ПДО-32
- Пульт ПДО-32Н
- Пульт ПДО-48
- Устройство УПВ-2
- Устройство УПВ-4
- Устройство УПВ-6
- Устройство УПВ-2Н
- Устройство УПВ-4Н
- Устройство УПВ-А
- Модуль СЛ1
- Модуль СЛ2
- Усилитель УМ-300-2

Изменение списка поддерживаемых пультов привело к коррекции ранее разработанных модулей в части проверки подключенного устройства:

- ConfigureDialog;
- FmWareDialog.

Были изменены следующие файлы:

- main\classes\base_types\station_types.h;
- dialogs\fmwaredlg\fmwaredlg.cpp;
- dialogs\configdlg\configuredlg.ui;
- main\classes\base_model\items\station_item\model_keys\key.h;
- main\classes\base_model\items\station_item\station_item.cpp.

2. Доработка функционала назначения функций на нажатие кнопок пультов

Согласно ТЗ был изменен список функций кнопок пультов при настройке коммуникационного плана в ПО «Конфигуратор». Новый список приведен ниже:

- Нет (None)
- Симплекс (Simplex PA)
- Дуплекс (Duplex)
- Дуплекс Авто (Duplex Auto)
- Группа (Group PA)
- Циркуляр (Circular)
- Конференция (Conference)
- Оповещение Вкл/Выкл (Alarm on/off)
- МК Вкл/Выкл (Mic on/off)
- ГР Вкл/Выкл (Speaker on/off)
- Индикатор линии (Line indicator)
- Контакт Вкл/Выкл (Contact on/off)
- Контакт Вкл/Выкл Симплекс (Contact on/off Simplex PA)
- Контакт Вкл/Выкл Дуплекс Авто (Contact on/off Duplex Auto)
- Трансляция Вкл/Выкл (Line on/off Simplex PA)
- Трансляция Вкл/Выкл (Line on/off Duplex Auto)

Были добавлены следующие функциональные возможности:

- При добавлении кнопки анализируется текущий тип пульта, и происходит отказ в добавлении кнопки, если ее порядковый номер превышает максимально допустимое число кнопок настраиваемого пульта;
- При изменении идентификатора кнопки анализируется текущий тип пульта и происходит отказ в изменении идентификатора, если его номер превышает максимально допустимый размер;
- Добавлена возможность сортировки списка параметров кнопок;
- Добавление кнопки сделано в виде вставки на нужную позицию с последующей сортировкой по полю идентификатора;

- При изменении типа пульта сделан анализ на максимально допустимое количество поддерживаемых кнопок при новом типе пульта и, если это количество меньше существующих, то при смене типа пульта происходит удаление имеющихся кнопок с идентификаторами, превышающими данное количество;

- Добавлен класс unit-тестирования редактора кнопок с тестированием добавления, удаления, сериализации/десериализации параметров списка кнопок.

Были изменены следующие файлы:

- main\classes\base_model\items\station_item\station_item.cpp;
- main\classes\base_model\items\station_item\station_item.h;
- main\classes\base_model\items\station_item\model_keys\model_keys.cpp;
- main\classes\base_model\items\station_item\model_keys\model_keys.h;
- main\classes\base_model\items\station_item\test_station_item.h;
- main\classes\base_model\items\station_item\test_station_item.cpp.

Были добавлены следующие функциональные возможности:

- Добавлена возможность добавления нового пульта путем вставки на выбранную позицию с анализом порядкового номера пульта;

- Добавлена возможность сортировки отображаемого списка введенных пультов по всем полям структуры;

- При изменении типа пульта сделан анализ на максимально допустимое количество поддерживаемых кнопок при новом типе пульта и, если это количество меньше существующих, , то при смене типа пульта происходит удаление имеющихся кнопок с идентификаторами, превышающими данное количество;

- Переписана система сериализации информации о настройках пульта в части сортировки структур по cid-у пульта;

- Добавлен класс – unit-тестирования редактора списка пультов с тестированием добавления, удаления, сериализации/ десериализации параметров списка пультов.

Были изменены следующие файлы:

- main\classes\base_model\items\stations_item\stations_item.cpp;
- main\classes\base_model\items\stations_item\stations_item.h;

- main\classes\base_model\items\stations_item\model_stations\model_stations.cpp;
- main\classes\base_model\items\stations_item\model_stations\model_stations.h;
- main\classes\base_model\items\stations_item\test_stations_item.h;
- main\classes\base_model\items\stations_item\test_stations_item.cpp.

Добавлена проверка типа подключенного пульта: если принято значение 0 (неизвестный тип устройства), то выводится диалог с предложением продолжить процесс конфигурирования. Изменены файлы:

- dialogs\configdlg\configuredlg.cpp.

Осуществлен перевод доработанных модулей и интерфейсов на русский язык. Проведено документирование заголовочных файлов доработанных модулей. Доработаны следующие документы:

- Описание функций и алгоритмов работы ПО «Конфигуратор»
- Руководство оператора ПО «Конфигуратор»