

МОДУЛИ «ОСПЧ»
БИБЛИОТЕКА УПРАВЛЕНИЯ ДЕКОДЕРОМ. ОДНОКАНАЛЬНЫЙ РЕЖИМ

Руководство программиста

Листов 39

2019

СОДЕРЖАНИЕ

1. Описание функций	6
1.1. Инициализация библиотеки	6
1.2. Деинициализация библиотеки	6
1.3. Расширенная информация об ошибках	7
1.4. Версия библиотеки декодера	7
1.5. Загрузка стандартных режимов	8
1.6. Ввод лицензионного ключа	9
1.7. Загрузка дополнительных параметров	9
1.8. Считывание параметров мезонинного модуля	12
1.9. Получение статусов и значений для стандартного режима	13
1.10. Настройка в режиме DVB-S2, DVB-S2X, DVB-S2X (Jp)	14
2. Описание констант	21
2.1. Тип мезонинного модуля (DecoderVersion)	21
2.2. Статус синхронизации или состояние ввода лицензии (SyncState)	21
2.3. Тип декодера Рида-Соломона (RsDecMode)	21
2.4. Тип декодера RS-DVB (RsDvbType)	21
2.5. Режим дескремблера (DescramblerMode)	21
2.6. Режим дескремблера V.35 (V35Type)	22
2.7. Режим демультимплексора дополнительного уровня (DmxMode)	22
2.8. Режим адаптивного корректора в режиме DVB-S2 (SwitchAdpDVBS2) ..	22
2.9. Тип фильтра в режиме DVB-S2 (SwitchFilterDVBS2)	22
2.10. Тип фильтра в режиме DVB-S2 (DVBS2ECCMode)	22
2.11. Тип фильтра в режиме DVB-S2 (DVBS2ECCRolloff)	22
2.12. Режим декодера Витерби K=7 (NskK7Mode)	23
2.13. Режим декодера Витерби K=7 C32 (NskK7S32Mode)	23
2.14. Режим декодера CCK (SskMode)	23
2.15. Режим декодера TPC (TpcMode)	23
2.16. Порядок (_ORDER)	24

2.17. Синхронизация скремблера (_SYNC)	24
2.18. Режим декодера TPC * N (TpcNMode)	24
2.19. Режим стандартного декодера TPC (TpcStandardMode)	24
2.20. Режим декодера TPC - Разное (TpcOtherMode)	24
2.21. Режим декодера TPC Datum (TpcDatumMode).....	24
2.22. Режим декодера TPC Paradise (TpcParadiseMode)	25
2.23. Режим декодера LDPC (LdpcMode)	25
2.24. Режим декодера LDPC (TccMode)	25
2.25. Режим декодера LDPC (VfLdpcMode).....	25
2.26. Скорость декодера FL LDPC (FLdpcSpeed)	26
2.27. Режим декодера FL LDPC (FLdpcMode)	26
2.28. Скорость декодера FX LDPC (FxLdpcSpeed)	26
2.29. Размер декодера FX LDPC (FxLdpcSize)	27
2.30. Кодовая скорость декодера TCC2d16s (Tcc2d16sCoderate)	27
2.31. Длина блока декодера TCC2d16s (Tcc2d16sBlocklen)	27
3. Описание типов	28
3.1. Параметры декодера (DecoderControl)	28
3.2. Параметры декодера (DecValues).....	28
3.3. Состояние декодера (DecStatus)	28
3.4. Значения температур и статусы перегрева (DecTemperature)	28
3.5. Параметры декодера RS-IESS (RslessParameters)	29
3.6. Параметры декодера RS-C (RsCParameters)	29
3.7. Параметры декодера RS-DVB (RsDvbParameters)	29
3.8. Параметры дескремблера (DescramblerParameters).....	29
3.9. Параметры демультимплектора (DmxParameters)	29
3.10. Параметры демультимплектора IBS (lbsParameters)	30
3.11. Параметры демультимплектора Edmac (EdmacParameters).....	30
3.12. Параметры DVB-S2 (DVBS2FullDec)	30
3.13. Статус режима DVB-S2 (DVBS2Status).....	30

3.14. Статистика принятых пакетов DVB-S2 (ModCodeStat)	30
3.15. Тип DVB-S2 для переключения normal/short (DVBS2M3Status)	31
3.16. Статус DVB-S2 (DvbS2EccStatus)	31
3.17. Параметры декодера Витерби K=7 (ViterbiK7Parameters)	31
3.18. Параметры декодера Витерби K=7 C32 (ViterbiK7S32Parameters)	31
3.19. Параметры декодера Витерби K=8 (ViterbiK8Parameters)	31
3.20. Параметры декодера CCK (SskParameters)	31
3.21. Параметры декодера TPC (TpcParameters)	31
3.22. Параметры декодера TPC Hamming (TpcHammingParameters)	32
3.23. Параметры декодера TPC * N (TpcNParameters)	32
3.24. Параметры декодера TPC Standard (TpcStandardParameters)	32
3.25. Параметры декодера TPC - разное (TpcOtherParameters)	32
3.26. Параметры декодера TPC - Datum (TpcDatumParameters)	33
3.27. Параметры декодера TPC - Paradise (TpcParadiseParameters)	33
3.28. Параметры декодера Синхронизатор (SynchronizerParameters)	33
3.29. Параметры декодера DVB-S2 CCM (Dvbs2CcmParameters)	33
3.30. Параметры декодера LDPC (LdpcParameters)	33
3.31. Параметры декодера TCC 2564 (Tcc2564Parameters)	33
3.32. Параметры декодера VF LDPC (VfLdpcParameters)	34
3.33. Параметры декодера FL LDPC (FlLdpcParameters)	34
3.34. Параметры декодера FX LDPC (FxLdpcParameters)	34
3.35. Параметры декодера TCC2d16s (Tcc2d16sParameters)	34
3.36. Структура для выключения декодера (DecOFFParameters)	34
4. Возвращаемые значения	35
5. Базовые операции	37
5.1. Подключение к устройству	37
5.2. Загрузка стандартных декодеров	37
5.3. Загрузка режима DVB-S2 ACM для ОСПЧ-Е, ОСПЧ-М1	38
5.4. Загрузка режима DVB-S2, DVB-S2X для ОСПЧ-Ех	38

5.5. Отключение от устройства	39
-------------------------------------	----

Библиотека Decoder.dll предназначена для загрузки мезонинных модулей «Декодер-М1», «Декодер-М2», «Декодер-М3», «Декодер-Е4» в составе модулей ОСПЧ-Ех, ОСПЧ-Е/М1.

Библиотека Decoder.dll должна находиться в той же папке, что и библиотеки ospchx.dll. Перед использованием функций библиотеки декодера устройство ОСПЧ обязательно должно быть инициализировано заранее. Перед загрузкой декодера с помощью функций библиотеки Decoder.dll необходимо остановить мастер устройства, а после снова запустить по необходимости. Подробное описание структур и интерфейс вызова функций находятся в соответствующих файлах DEC_Types.h и DEC_Defines.h.

Практически все функции библиотеки декодера имеют входной параметр unsigned char* DevNum – номер устройства в системе.

Описание значений, возвращаемых функциями находится в файле DEC_Errors.h

При успешном выполнении функция возвращает «0», в противном случае – код ошибки. Подробную информацию об ошибке можно получить при помощи вызова функции [getDecErrorDescription](#). Перед началом работы необходимо также проинициализировать декодер, по завершении работы - деинициализировать.

После каждой загрузки всех параметров декодера необходимо вызывать функцию ResetPLL и CircuitReset. В режиме DVB-S2 необходимо вызывать функцию [FullDVBS2CircuitReset](#).

Во все функции кроме [getDecErrorDescription](#) передается параметр DevNum – номер устройства в системе.

Для обозначения устройств ОСПЧ-Е1, ОСПЧ-Е2, ОСПЧ-Е3, ОСПЧ-Е4 в документе используется сокращение ОСПЧ-Ех.

Для управления устройствами ОСПЧ помимо функций библиотеки декодера, требуются функции библиотек ospchx.dll: DeviceSupportedCount, GetFullDeviceName, CreateDeviceInstance, rSynthLoad, GetDeviceSlotID, GetLConvertorType, MasterStop, MasterGetBuffers, FPGALoad, ADCLoad, FirstGainControl, SetCarrier, ResetPLL, CircuitReset, ReleaseDeviceInstance, и другие. Информация по данным функциям доступна в документе «Модули ОСПЧ. Библиотека функций. Руководство программиста».

1. ОПИСАНИЕ ФУНКЦИЙ

1.1. Инициализация библиотеки

Для работы с устройством с помощью библиотеки декодера необходимо сначала произвести ее инициализацию. Без инициализации доступна только функция считывания версии API [getDecoderApiVersion](#).

Для инициализации библиотеки необходимо вызвать функцию создания объектов работы с устройством [createDecInstance](#), после чего выполнить подключение к устройству и инициализацию декодера путем вызова функции [initDecoder](#). Инициализацию необходимо производить для каждого используемого устройства.

```
int createDecInstance (  
    const unsigned char* DevNum  
);
```

Описание	Инициализация объектов модуля декодирования
----------	---

```
int initDecoder (  
    const unsigned char* DevNum  
);
```

Описание	Подключение к устройству, инициализация декодера
----------	--

1.2. Деинициализация библиотеки

По окончании работы с устройством необходимо выполнить отключение от него и деинициализацию библиотеки. Для этого предназначена функция [releaseDecoder](#).

```
int releaseDecoder (  
    const unsigned char* DevNum  
);
```

Описание	Деинициализация декодера
----------	--------------------------

1.3. Расширенная информация об ошибках

В процессе работы с библиотекой декодера возможно возникновение ошибок. Для получения подробной информации о возникшей ошибке, а также место ее возникновения в библиотеке предназначена функция [getDecErrorDescription](#).

```
int getDecErrorDescription (
    const unsigned char* DevNum,
    char* description,
    char* location,
    char* dbgInfo,
    int maxStrLen
);
```

Описание	Возвращает место возникновения и описание последней ошибки (например, путь к файлу при ошибке E_OPEN_FILE_ERROR), а также отладочную информацию для включения в лог. При вызове параметры обнуляются, т.е. при повторном вызове функция вернет E_NO_ERROR, выходные параметры заполнены не будут
Входные параметры	int maxStrLen – максимальная длина строки описания ошибки, места возникновения, или отладочной информации
Выходные параметры	char* description – дополнительная информация об ошибке (например, для ошибок доступа к файлу возвращает путь к файлу) char* location – место возникновения ошибки в библиотеке decoder.dll (требуется для отладки) char* dbgInfo – отладочная информация
Возвращаемое значение	Код последней ошибки

1.4. Версия библиотеки декодера

Для идентификации версии API декодера предназначена функция [getDecoderApiVersion](#). Данная функция может быть использована без предварительной инициализации библиотеки и загрузки режима.

```
char* getDecoderApiVersion ();
```


Описание	Считывание версии библиотеки декодера
Возвращаемое значение	Версия библиотеки декодера в строковом формате (char*)

1.5. Загрузка стандартных режимов

Загрузка стандартных режимов включает в себя загрузку ПЛИС мезонинного модуля, а также ввод параметров путем записи в регистры ПЛИС. Для загрузки ПЛИС предусмотрена функция `loadDecFpga`. Ввод параметров осуществляется функцией `loadDecReg`. Выбор файла, загружаемого в ПЛИС, а также загрузка ОЗУ мезонина (если требуется) выполняется автоматически на основании переданных параметров. Считывание различных информационных данных, а также ввод параметров допустимы только после успешного выполнения функции `loadDecFpga`.

```
int loadDecFpga (
    const unsigned char* DevNum,
    DecoderControl* ctrl,
    const char* firmwarePath
);
```

Описание	Загрузка ПЛИС декодера в соответствии с режимом
Входные параметры	<i>DecoderControl</i> * ctrl – структура параметров декодера char* firmwarePath – путь к директории Firmware

```
int loadDecReg (
    const unsigned char* DevNum,
    DecoderControl* ctrl,
    const char* firmwarePath
);
```

Описание	Загрузка параметров выбранного декодера в ПЛИС
Входные параметры	<i>DecoderControl</i> * ctrl – структура параметров декодера char* firmwarePath – путь к директории Firmware

1.6. Ввод лицензионного ключа

Для полноценного функционирования некоторые режимы декодера требуют установки лицензий. Ввод лицензионного ключа выполняется функцией `installDecLicense`. Вызов данной функции необходимо произвести сразу после загрузки ПЛИС декодера. О необходимости ввода лицензии для выбранного режима можно судить по параметру `DecStatus::license`, который считывается с помощью функции `readDecStatus`.

```
int installDecLicense (
    const unsigned char* DevNum,
    unsigned long long * key
);
```

Описание	Ввод лицензионного ключа декодера
Входные параметры	<code>unsigned long long * key</code> – лицензионный ключ декодера

1.7. Загрузка дополнительных параметров

После загрузки декодера может потребоваться установка дополнительных параметров или включение дополнительных модулей обработки сигнала. Для включения инверсии спектра или изменения порядка каналов предусмотрены функции `changeSpectrInversion` и `changeChannelOrder`. Управление инвертором и дифференциальным декодером осуществляется функциями `setInvertor` и `setDiffDecoder`. Если требуется мультипликативный дескремблер или демультимплексор, используются функции `setDescrambler` и `setDmx`. Управление декодером Рида-Соломона осуществляется с помощью функций `setDecRsOff`, `setDecRsless`, `setDecRsC`, `setDecRsDvb`.

```
int changeSpectrInversion (
    const unsigned char* DevNum,
    const bool value
);
```

Описание	Управление инверсией спектра
Входные параметры	<code>bool value</code> – инверсия спектра

<pre>int changeChannelOrder (const unsigned char* DevNum, const bool value);</pre>	
Описание	Управление порядком каналов
Входные параметры	bool value – порядок каналов

<pre>int setDiffDecoder (const unsigned char* DevNum, bool decRsMode, bool enable);</pre>	
Описание	Управление дифференциальным декодером
Входные параметры	bool decRsMode – выбор декодера для которого будет настроен дифференциальный декодер (false – основной декодер, true – декодер Рида-Соломона) bool enable – включение/выключение дифференциального декодера

<pre>int setInvertor (const unsigned char* DevNum, bool decRsMode, bool enable);</pre>	
Описание	Управление инвертором
Входные параметры	bool decRsMode – выбор декодера для которого будет настроен инвертор (false – основной декодер, true – декодер Рида-Соломона) bool enable – включение/выключение инвертора

<pre>int setDescrambler (const unsigned char* DevNum, bool decRsMode, bool enable, DescramblerParameters* control);</pre>	
Описание	Управление дескремблером

Входные параметры	bool decRsMode – выбор декодера для которого будет настроен дескремблер (false – основной декодер, true – декодер Рида-Соломона) bool enable – включение/выключение дескремблера <i>DescramblerParameters*</i> control – параметры дескремблера
-------------------	---

<pre>int setDmx (const unsigned char* DevNum, bool enable, DmxParameters* control);</pre>	
Описание	Установка параметров доп. ступени демультиплексора
Входные параметры	bool enable – включение/выключение демультиплексора <i>DmxParameters*</i> control – параметры демультиплексора

<pre>int setDecRsOff (const unsigned char* DevNum);</pre>	
Описание	Выключение декодера Рида-Соломона

<pre>int setDecRsless (const unsigned char* DevNum, RslessParameters* control);</pre>	
Описание	Включение декодера Рида-Соломона IESS и загрузка параметров
Входные параметры	<i>RslessParameters*</i> control – параметры декодера RS-IESS

<pre>int setDecRsC (const unsigned char* DevNum, RsCParameters* control);</pre>	
Описание	Включение декодера Рида-Соломона C и загрузка параметров

Входные параметры	<i>RsCParameters*</i> control – параметры декодера RS-C
-------------------	---

<pre>int setDecRsDvb (const unsigned char* DevNum, RsDvbParameters* control);</pre>	
Описание	Включение декодера Рида-Соломона DVB и загрузка параметров
Входные параметры	<i>RsDvbParameters*</i> control – параметры декодера RS-DVB

1.8. Считывание параметров мезонинного модуля

После инициализации библиотеки доступны функции считывания версии мезонинного модуля и кода DNA. Данные функции не требуют предварительной загрузки режима ([loadDecFpga](#)).

<pre>int getDecoderVersionOnBoard (const unsigned char* DevNum, DecoderVersion *DecVersion);</pre>	
Описание	Считывание версии мезонина декодера
Выходные параметры	<i>DecoderVersion*</i> DecVersion – версия мезонинного модуля

<pre>int getDnaCode (unsigned char* DevNum, char* code, int maxStrLen);</pre>	
Описание	Считывание DNA-кода
Входные параметры	int maxStrLen – максимальная длина строки DNA-кода
Выходные параметры	char* code – буфер, в который будет записан DNA-код в строковом формате

1.9. Получение статусов и значений для стандартного режима

После загрузки и настройки становится доступным считывание параметров и статусов декодера. Для получения температуры, а также флагов перегрева мезонина предназначена функция `readDecTemperature`. Считывание статуса синхронизации и лицензии осуществляется функцией `readDecStatus`. С помощью данной функции определяется не только корректность, но и необходимость ввода лицензионного ключа. Считывание вероятности битовой ошибки на выходе декодера осуществляется функцией `readBer`. Для расчета ОСШ могут понадобиться параметры `cod1` и `cod2`, которые можно получить функцией `getDecValues`.

```
int readDecTemperature (
    const unsigned char* DevNum,
    DecTemperature* values
);
```

Описание	Считывание температуры и статуса перегрева
Выходные параметры	<i>DecTemperature* values</i> — значения температур и статусы перегрева

```
int readDecStatus (
    const unsigned char* DevNum,
    DecStatus* decState
);
```

Описание	Считывание статуса синхронизации и лицензии декодера
Выходные параметры	<i>DecStatus* decState</i> – значения синхронизации и необходимости/корректности ввода лицензии

```
int readBer (
    const unsigned char* DevNum,
    double* value
);
```

Описание	Считывание вероятности битовой ошибки на выходе декодера (BER)
----------	--

Выходные параметры	<code>double* value</code> – значение BER
--------------------	---

<pre>int getDecValues (const unsigned char* DevNum, DecValues* values);</pre>	
Описание	Считывание значений <code>cod1</code> и <code>cod2</code> в соответствие с используемым режимом
Выходные параметры	<i>DecValues* values</i> – параметры <code>cod1</code> и <code>cod2</code>

1.10. Настройка в режиме DVB-S2, DVB-S2X, DVB-S2X (Jp)

Загрузка устройства в режимах DVB-S2, DVB-S2X, DVB-S2X (Jp) отличается от загрузки стандартных декодеров. Для загрузки данных режимов, а также считывания значений и параметров предусмотрены функции из данного раздела. Использование функций режимов DVB-S2, DVB-S2X, DVB-S2X (Jp) описано в разделах 5.3, 5.4, а также в примерах из дистрибутива библиотеки.

<pre>int setDvbS2E4Mode (unsigned char* DevNum, SIGNAL_TYPE signalType, bool changeHeaderModulationAlgorithm);</pre>	
Описание	Установка режима DVB-S2 / S2X / S2X (Jp)
Входные параметры	<code>SIGNAL_TYPE signalType</code> – вид модуляции <code>bool changeHeaderModulationAlgorithm</code> – режим с изменением закона модуляции заголовка после 26 символа, в соответствии со спецификацией DVB-S2X

<pre>int loadDvbS2XCustomTable (unsigned char* DevNum, unsigned int* values, int count);</pre>	
--	--

Описание	Загрузка таблицы модкодов для DVB-S2X (Jp)
Входные параметры	unsigned int* values – указатель на массив значений модкодов int count – количество элементов массива модкодов

```
int FullDVBS2FilterLoad (
    unsigned char* DevNum,
    char* FiltersDir,
    const double Clk
);
```

Описание	Загрузка фильтра в режимах DVB-S2
Входные параметры	char* FiltersDir – путь к каталогу фильтров const double Clk – тактовая частота (Гц)

```
int FullDVBS2Load (
    unsigned char* DevNum,
    char* firmwarePath,
    const double Clk
);
```

Описание	Загрузка демодулятора DVB-S2
Входные параметры	char* firmwarePath – путь к директории Firmware const double Clk – тактовая частота (Гц)

```
int loadDecRam (
    unsigned char* DevNum,
    char* fpgaFile,
    char* ramFile,
    int ramNumber
);
```

Описание	Загрузка ОЗУ декодера
Входные параметры	char* fpgaFile – путь к файлу ext_ram.bit (для устройств с мезонинным модулем «Декодер-M1») char* ramFile – путь к файлу ОЗУ int ramNumber – номер ОЗУ декодера (0 или 1)

```
int FullDVBS2Decoder (
```


<pre> unsigned char* DevNum, SIGNAL_TYPE modulation, const DVBS2FullDec* Control); </pre>	
Описание	Загрузка декодера DVB-S2
Входные параметры	SIGNAL_TYPE modulation – вид модуляции DVBS2FullDec* Control – параметры режима DVB-S2

<pre> int FullDVBS2Clock (unsigned char* DevNum, const double Clk); </pre>	
Описание	Установка тактовой частоты в режиме DVB-S2
Входные параметры	double Clk– тактовая частота (Гц)

<pre> int FullDVBS2SpectrInv (unsigned char* DevNum, bool *SVType); </pre>	
Описание	Настройка инверсии спектра в режиме DVB-S2
Входные параметры	bool *SVType – инверсия спектра

<pre> int FullDVBS2ADPCorr (unsigned char* DevNum, const SwitchAdpDVBS2 *ADPType); </pre>	
Описание	Настройка адаптивного корректора в режиме DVB-S2
Входные параметры	SwitchAdpDVBS2*ADPType – режим адаптивного корректора

<pre> int FullDVBS2RollOff (unsigned char* DevNum, </pre>	
--	--

const SwitchFilterDVBS2 *RFType);	
Описание	Настройка фильтра в режиме DVB-S2
Входные параметры	SwitchFilterDVBS2 *RFType – тип фильтра

int checkDvbS2E4Load (unsigned char* DevNum, unsigned int* bits);	
Описание	Проверка загрузки режима DVB-S2/DVB-S2X (готовности интерфейса) для ОСПЧ-Е4
Выходные параметры	unsigned int* bits – зарезервировано

int FullDVBS2Status (unsigned char* DevNum, DVBS2Status* DVBStatus);	
Описание	Считывание статуса режима DVB-S2
Выходные параметры	DVBS2Status* DVBStatus – статус режима

char* FullDVBS2GetModcodeString (unsigned char* DevNum);	
Описание	Считывание модкода режима DVB-S2
Возвращаемое значение	Модкод в строковом формате (char*)

bool DVBS2E1GetPacketSyncStatus (unsigned char* DevNum);	
Описание	Считывание статуса синхронизации пакетов режима DVB-S2

Возвращаемое значение	Статус синхронизации (bool)
-----------------------	-----------------------------

```
int DVBS2E1GetModCodeStatistics (
    unsigned char* DevNum,
    ModCodeStat* statistic,
    int DVBS2Mode
);
```

Описание	Считывание статистики принятых пакетов режима DVB-S2
Входные параметры	int DVBS2Mode – режим DVB-S2 (0- DVB-S2, 1- DVB-S2X)
Выходные параметры	<i>ModCodeStat*</i> statistic – статистика пакетов

```
int GetDVBS2M3Status (
    unsigned char* DevNum,
    DVBS2M3Status* dvbStatus
);
```

Описание	Считывание типа DVB-S2 на мезонинных модулях «Декодер-М3» для переключения normal/short
Выходные параметры	<i>DVBS2M3Status*</i> dvbStatus – тип и флаг готовности режима DVB-S2

```
double GetDVBS2EM1GC2Value (
    unsigned char* DevNum
);
```

Описание	Считывание значения APY2 для DVB-S2 на платах ОСПЧ-Е/М1
Возвращаемое значение	Значение APY2 (double)

```
double GetDVBS2EM1BERValue (
    unsigned char* DevNum,
    bool eccMode
);
```

Описание	Считывание вероятности битовой ошибки на выходе декодера для DVB-S2 на платах ОСПЧ-Е/М1
Входные параметры	bool eccMode – режим DVB-S2 (0-CCM, 1-ACM)
Возвращаемое значение	Значение BER (double)

<pre>int GetDVBS2ECCStatus (unsigned char* DevNum, DvbS2EccStatus* dvbStatus);</pre>	
Описание	Считывание статуса DVB-S2 для устройств ОСПЧ-Е/М1, а также устройств с мезонином «Декодер-М2»
Выходные параметры	<i>DvbS2EccStatus*</i> dvbStatus – статус DVB-S2

<pre>int getDVBS2CorrectedPacketsCount (unsigned char* DevNum, double* value);</pre>	
Описание	Считывание количества исправленных пакетов в режиме DVB-S2
Выходные параметры	double* value – количество исправленных пакетов

<pre>int FullDVBS2VectDiag (unsigned char* DevNum, unsigned short* plQ, unsigned int* cbWritten, unsigned int timeoutMs);</pre>	
Описание	Считывание данных для построения векторной диаграммы в режиме DVB-S2
Входные параметры	unsigned int* cbWritten – размер массива значений plQ unsigned int timeoutMs – время на выполнение операции (таймаут ожидания данных)

Выходные параметры	<code>unsigned short* plQ</code> – указатель на массив, в который будут записаны значения IQ для построения векторной диаграммы <code>unsigned int* cbWritten</code> – количество записанных значений IQ
--------------------	---

```
int FullDVBS2CircuitReset (  
    unsigned char* DevNum  
);
```

Описание	Сброс различных схем
----------	----------------------

2. ОПИСАНИЕ КОНСТАНТ

2.1. Тип мезонинного модуля (DecoderVersion)

DecoderVersion		
DECODER_M1	0	«Декодер-M1»
DECODER_M2	1	«Декодер-M2»
DECODER_M3	2	«Декодер-M3»
DECODER_E4	3	«Декодер-E4»
DECODER_UNDEFINED	99	вид мезонинного модуля не определен

2.2. Статус синхронизации или состояние ввода лицензии (SyncState)

SyncState		
Для статуса синхронизации:		
SYNCSTATE_NO	0	Синхронизации нет
SYNCSTATE_YES	1	Синхронизация есть
SYNCSTATE_BAD	2	Синхронизация есть, но много ошибок в сигнале
SYNCSTATE_NA	3	Статус синхронизации недоступен или не определен для данного режима (возможно, режим не загружен)
Для статуса ввода лицензии:		
SYNCSTATE_NO	0	Лицензионный ключ требуется, не введен
SYNCSTATE_YES	1	Лицензионный ключ требуется, введен корректно
SYNCSTATE_BAD	2	Лицензионный ключ требуется, не принят
SYNCSTATE_NA	3	Ключ не требуется

2.3. Тип декодера Рида-Соломона (RsDecMode)

RsDecMode		
DRS00_IESS	0	Декодер RS-IESS
DRS01_C	1	Декодер RS-C
DRS02_DVB	2	Декодер RS-DVB
DRS03_OFF	3	Декодер выключен
DRS99_UNKNOWN	99	Состояние декодера не определено

2.4. Тип декодера RS-DVB (RsDvbType)

Тип декодера RS-DVB		
RS_DVB_204_188	0	
RS_DVB_160_146	1	

2.5. Режим дескремблера (DescramblerMode)

DescramblerMode		
V35	0	V.35

C320	1	3.20C
NONSTANDART	2	Нестандартный

2.6. Режим дескремблера V.35 (V35Type)

V35Type		
IESS	0	IESS
USUAL	1	Обычный

2.7. Режим демультиплексора дополнительного уровня (DmxMode)

DmxMode		
DMX_IBS	0	IBS
DMX_RESERVED	1	зарезервировано
DMX_EDMAC	2	EDMAC

2.8. Режим адаптивного корректора в режиме DVB-S2 (SwitchAdpDVBS2)

SwitchAdpDVBS2		
S2_OFF	0	Выключен
S2_HEADER	1	Заголовок
S2_DATA	2	Данные

2.9. Тип фильтра в режиме DVB-S2 (SwitchFilterDVBS2)

SwitchFilterDVBS2		
S2_NYQ_0_2	0	Нyq-0.2
S2_NYQ_0_25	1	Нyq-0.25
S2_NYQ_0_35	2	Нyq-0.35

2.10. Тип фильтра в режиме DVB-S2 (DVBS2ECCMode)

DVBS2ECCMode		
ECC_MODE_GENERIC_PACKETIZED	0	
ECC_MODE_GENERIC_CONTINUOUS	1	
ECC_MODE_RESERVED	2	
ECC_MODE_TRANSPORT_STREAM	3	

2.11. Тип фильтра в режиме DVB-S2 (DVBS2ECCRolloff)

DVBS2ECCRolloff		
-----------------	--	--

ECC_ROLLOFF_035	0
ECC_ROLLOFF_025	1
ECC_ROLLOFF_020	2
ECC_ROLLOFF_RESERVED	3

2.12. Режим декодера Витерби K=7 (NskK7Mode)

NskK7Mode		
NSK_K7_0_12	0	1/2
NSK_K7_1_34	1	3/4
NSK_K7_2_78	2	7/8
NSK_K7_3_23_DVB	3	2/3 DVB
NSK_K7_4_34_DVB	4	3/4 DVB
NSK_K7_5_56_DVB	5	5/6 DVB
NSK_K7_6_78_DVB	6	7/8 DVB
NSK_K7_7_23_TCM_IESS	7	2/3 TCM IESS
NSK_K7_8_23_TCM_DSNG	8	2/3 TCM DSNG
NSK_K7_9_56_TCM_DSNG	9	5/6 TCM DSNG

2.13. Режим декодера Витерби K=7 C32 (NskK7S32Mode)

NskK7S32Mode		
NSK_K7S32_0_12	0	1/2
NSK_K7S32_1_34	1	3/4
NSK_K7S32_2_78	2	7/8

2.14. Режим декодера CCK (SskMode)

SskMode		
SSK_0_12_K36	0	1/2 K=36
SSK_1_34_K63	1	3/4 K=63
SSK_2_78_K89	2	7/8 K=89
SSK_3_12_K41	3	1/2 K=41
SSK_4_34_K57	4	3/4 K=57

2.15. Режим декодера TPC (TpcMode)

TpcMode		
TPC_MODE_HAMMING	0	TPC Hamming
TPC_MODE_N	1	TPC * N
TPC_MODE_STANDARD	2	Стандартный
TPC_MODE_DATUM	3	TPC Datum
TPC_MODE_PARADISE	4	TPC Paradise
TPC_MODE_OTHER	5	Другие режимы TPC
Режимы TPC Datum, TPC Paradise доступны только для мезонинных модулей «Декодер-М3»		

2.16. Порядок (_ORDER)

_ORDER		
BEFORE	0	Перед
AFTER	1	После

2.17. Синхронизация скремблера (_SYNC)

_SYNC		
FRAME	0	По кадру
SUPERFRAME	1	По суперкадру

2.18. Режим декодера TPC * N (TpcNMode)

TpcNMode		
TPC_N_128_120_128_120	0	(128, 120) x (128, 120)
TPC_N_64_57_64_57	1	64, 57) x (64, 57)
TPC_N_32_26_32_26_4_3	2	(32, 26) x (32, 26) x (4, 3)
TPC_N_32_26_32_21	3	(32, 26) x (32, 21)

2.19. Режим стандартного декодера TPC (TpcStandardMode)

TpcStandardMode		
TPC_STD_5_16	0	5/16
TPC_STD_21_44	1	21/44
TPC_STD_3_4	2	3/4 (ранее назывался (64,57)x(46,39))
TPC_STD_7_84	3	3/4 (ранее назывался (128,120)x(128,120))
TPC_STD_095	4	0.95
Режимы 5/16, 21/44, 0.95 доступны только для мезонинных модулей «Декодер-М3»		

2.20. Режим декодера TPC - Разное (TpcOtherMode)

TpcOtherMode		
TPC_OTHER_64_57_32_26_x8_ADD	0	(64, 57) x (32, 26) x 8

2.21. Режим декодера TPC Datum (TpcDatumMode)

TpcDatumMode		
TPC_DATUM_ADVANCED_4K_1_2	0	
TPC_DATUM_ADVANCED_4K_3_4	1	
TPC_DATUM_ADVANCED_4K_7_8	2	
TPC_DATUM_ADVANCED_4K_0950	3	
TPC_DATUM_ADVANCED_16K_0453	4	
TPC_DATUM_ADVANCED_16K_1_2	5	

TPC_DATUM_ADVANCED_16K_3_4	6
TPC_DATUM_ADVANCED_16K_7_8	7
TPC_DATUM_ADVANCED_16K_0922	8
TPC_DATUM_M5_LEGACY_3_4	9
TPC_DATUM_M5_LEGACY_7_8	10
TPC_DATUM_M5_SHORT_3_4	11
TPC_DATUM_M5_SHORT_7_8	12
TPC_DATUM_M5_FULL_3_48	13

2.22. Режим декодера TPC Paradise (TpcParadiseMode)

TpcParadiseMode	
TPC_PARADISE_0477	0
TPC_PARADISE_0493	1
TPC_PARADISE_0666	2
TPC_PARADISE_0789	3
TPC_PARADISE_0875	4
TPC_PARADISE_093	5

2.23. Режим декодера LDPC (LdpcMode)

LdpcMode		
LDPC_0_12	0	1/2
LDPC_1_23	1	2/3
LDPC_2_34	2	3/4

2.24. Режим декодера LDPC (TccMode)

TccMode		
TCC_0_QPSK_12	0	1/2 ФМ4
TCC_1_QPSK_23	1	2/3 ФМ4
TCC_2_QPSK_34	2	3/4 ФМ4
TCC_3_QPSK_56	3	5/6 ФМ4
TCC_4_QPSK_78	4	7/8 ФМ4
TCC_5_PSK8_23	5	2/3 ФМ8
TCC_6_PSK8_34	6	3/4 ФМ8
TCC_7_PSK8_56	7	5/6 ФМ8
TCC_8_PSK8_89	8	8/9 ФМ8

2.25. Режим декодера LDPC (VfLdpcMode)

VfLdpcMode	
VFLDPC_BPSK_0488	0
VFLDPC_QPSK_0533	1
VFLDPC_QPSK_0631	2
VFLDPC_QPSK_0706	3

VFLDPC_QPSK_0803	4
VFLDPC_QAM8_0576	5
VFLDPC_QAM8_0642	6
VFLDPC_QAM8_0711	7
VFLDPC_QAM8_0780	8
VFLDPC_QAM16_0644	9
VFLDPC_QAM16_0731	10
VFLDPC_QAM16_0780	11
VFLDPC_QAM16_0829	12
VFLDPC_QAM16_0853	13
VFLDPC_BPSK_0493	14
VFLDPC_QPSK_0493	15
VFLDPC_QPSK_0654	16
VFLDPC_QPSK_0734	17

2.26. Скорость декодера FL LDPC (FlLdpcSpeed)

FlLdpcSpeed	
FLLDPC_S_0499	0
FLLDPC_S_0532	1
FLLDPC_S_0639	2
FLLDPC_S_0710	3
FLLDPC_S_0726	4
FLLDPC_S_0778	5
FLLDPC_S_0798	6
FLLDPC_S_0828	7
FLLDPC_S_0851	8

2.27. Режим декодера FL LDPC (FlLdpcMode)

FlLdpcMode	
FLLDPC_M_LowLatency	0
FLLDPC_M_Balanced	1
FLLDPC_M_LowBER	2

2.28. Скорость декодера FX LDPC (FxLdpcSpeed)

FxLdpcSpeed	
FXLDPC_SPEED_1_2	0
FXLDPC_SPEED_2_3	1
FXLDPC_SPEED_3_4	2
FXLDPC_SPEED_14_17	3
FXLDPC_SPEED_7_8	4
FXLDPC_SPEED_10_11	5
FXLDPC_SPEED_16_17	6

2.29. Размер декодера FX LDPC (FxDpcSize)

FxDpcSize	
FXLDPC_SIZE_256	0
FXLDPC_SIZE_512	1
FXLDPC_SIZE_1K	2
FXLDPC_SIZE_2K	3
FXLDPC_SIZE_4K	4
FXLDPC_SIZE_8K	5
FXLDPC_SIZE_16K	6

2.30. Кодовая скорость декодера TCC2d16s (Tcc2d16sCoderate)

Tcc2d16sCoderate		
TCC2D16S_CR_12	0	1/2
TCC2D16S_CR_23	1	2/3
TCC2D16S_CR_45	2	4/5
TCC2D16S_CR_67	3	6/7

2.31. Длина блока декодера TCC2d16s (Tcc2d16sBlocklen)

Tcc2d16sBlocklen	
TCC2D16S_S_680	0
TCC2D16S_S_1752	1

3. ОПИСАНИЕ ТИПОВ

3.1. Параметры декодера (DecoderControl)

DecoderControl		
SIGNAL_TYPE	modulationType	вид модуляции
double	clockFrequency	тактовая частота (Гц)
void*	param	указатель на структуру параметров декодера (<i>ViterbiK7Parameters, ViterbiK7S32Parameters, ViterbiK8Parameters, SskParameters, TpcParameters, SynchronizerParameters, Dvbs2CcmParameters, LdpcParameters, Tcc2564Parameters, VfLdpcParameters, FLdpcParameters, FxLdpcParameters, DecOFFParameters</i>)

3.2. Параметры декодера (DecValues)

DecValues		
double	cod1	
double	cod2	

3.3. Состояние декодера (DecStatus)

DecStatus		
SyncState	dec	статус синхронизации декодера
SyncState	drs1	статус синхронизация декодера Рида-Соломона
SyncState	drs2	статус синхронизация демультиплексителя Рида-Соломона
SyncState	dmxFrame	статус синхронизация демультиплексора по кадру
SyncState	dmxSuperframe	статус синхронизация демультиплексора по супер кадру
SyncState	license	статус состояние лицензии

3.4. Значения температур и статусы перегрева (DecTemperature)

DecTemperature		
char	Fpga	температура ПЛИС декодера
char	ECC	температура ECC
char	ANA	температура АНА
bool	FpgaDecOverheat	перегрев ПЛИС декодера
bool	ECCOverheat	перегрев ECC
bool	ANAOverheat	перегрев АНА
Параметры ECC, АНА, ECCOverheat, ANAOverheat актуальны только для мезонинных модулей «Декодер-М1»		

3.5. Параметры декодера RS-IESS (RsIessParameters)

RsIessParameters		
<i>RsDecMode</i>	decoderRsMode	тип декодера Рида-Соломона (= DRS00_IESS, только чтение)
unsigned short	BlockLen	
unsigned short	InfCount	
unsigned short	StrShift	
unsigned short	StrCount	
bool	Scrambler	

3.6. Параметры декодера RS-C (RsCParameters)

RsCParameters		
<i>RsDecMode</i>	decoderRsMode	тип декодера Рида-Соломона (=DRS01_C, только чтение)
unsigned short	BlockLen	размер блока
unsigned short	Syncs	сдвиг строк
unsigned short	StrCount	количество строк

3.7. Параметры декодера RS-DVB (RsDvbParameters)

RsDvbParameters		
<i>RsDecMode</i>	decoderRsMode	тип декодера Рида-Соломона (=DRS02_DVB, только чтение)
<i>RsDvbType</i>	Type	тип декодера RS-DVB
bool	Scrambler	включение аддитивного скремблера

3.8. Параметры дескремблера (DescramblerParameters)

DescramblerParameters		
<i>DescramblerMode</i>	mode	режим дескремблера
<i>V35Type</i>	v35Mode	тип дескремблера v.35 (если выбран дескремблер v.35, т.е. DescramblerMode mode = V35)
unsigned int	Polynome	полином для нестандартного дескремблера (если выбран нестандартный дескремблер, т.е. DescramblerMode mode = NONSTANDART)

3.9. Параметры демультимплексора (DmxParameters)

DmxParameters		
<i>DmxMode</i>	Type	тип демультимплексора
<i>IbsParameters</i>	ibsOptions	параметры демультимплексора IBS (если выбран демультимплексор IBS, т.е. DmxMode Type = DMX_IBS)

<i>EdmacParameters</i>	edmacOptions	параметры демultipлексора EDMAC (если выбран демultipлексор IBS, т.е. DmxMode Type = DMX_EDMAC)
bool	Inversion	Инверсия

3.10. Параметры демultipлексора IBS (IbsParameters)

IbsParameters		
unsigned short	Period	период
unsigned short	Frame	кадр
bool	Descrambler	дескремблер

3.11. Параметры демultipлексора Edmac (EdmacParameters)

EdmacParameters		
unsigned short	Period	период
bool	Descrambler	дескремблер

3.12. Параметры DVB-S2 (DVBS2FullDec)

DVBS2FullDec		
bool	BBHeader	BB header
bool	MultiChannel	Multi channel
bool	SyncDelete	удаление синхронизации
bool	ModCode	включение функции выравнивания пакетов
unsigned short	PVSize	выравнивание пакетов в байтах
bool	DFL	обработка поля DFL
bool	UPL	обработка поля UPL
Параметры DFL, UPL актуальны только для мезонинных модулей «Декодер-M2», «Декодер-M3»		

3.13. Статус режима DVB-S2 (DVBS2Status)

DVBS2Status		
unsigned char	DEMSync	синхронизация демодулятора
unsigned char	DECSync	синхронизация декодера
unsigned char	DECSyncErr	отсутствие ошибок на выходе декодера
double	DiffCarrier	отклонение от установленной несущей частоты
double	Clock	измеренное значение тактовой частоты

3.14. Статистика принятых пакетов DVB-S2 (ModCodeStat)

ModCodeStat		
unsigned int	Count	размер массива
char*	StatArray	указатель на массив для записи статистики

3.15. Тип DVB-S2 для переключения normal/short (DVBS2M3Status)

DVBS2M3Status		
bool	ModeReady	актуальность параметра Mode
int	Mode	режим DVB-S2

3.16. Статус DVB-S2 (DvbS2EccStatus)

DvbS2EccStatus		
bool	npd	
bool	issy	
bool	isCCMMode	
bool	isSISMode	
DVBS2ECCMode	mode	
DVBS2ECCRollOff	rollOff	
int	isi	

3.17. Параметры декодера Витерби K=7 (ViterbiK7Parameters)

ViterbiK7Parameters		
NskK7Mode	mode	режим декодера витерби

3.18. Параметры декодера Витерби K=7 C32 (ViterbiK7S32Parameters)

ViterbiK7S32Parameters		
NskK7S32Mode	mode	режим декодера Витерби K=7 C32
bool	dmxEn	включение демультимплексора
unsigned int	frameLength	длина кадра
char*	pspFile	файл ПСП
unsigned int	synchronicSign	синхропризнак

3.19. Параметры декодера Витерби K=8 (ViterbiK8Parameters)

ViterbiK8Parameters		

3.20. Параметры декодера ССК (SskParameters)

SskParameters		
SskMode	mode	режим декодера ССК
unsigned char	puncturing	выкалывание

3.21. Параметры декодера ТРС (TpcParameters)

TpcParameters		
---------------	--	--

<i>TpcMode</i>	mode	режим декодера TPC
void*	params	параметры выбранного режима TPC (<i>TpcHammingParameters</i> , <i>TpcNParameters</i> , <i>TpcStandardParameters</i> , <i>TpcOtherParameters</i> , <i>TpcDatumParameters</i> , <i>TpcParadiseParameters</i>)

3.22. Параметры декодера TPC Hamming (*TpcHammingParameters*)

<i>TpcHammingParameters</i>		
unsigned char	nX, kX, nY, kY, nZ, kZ	ПУ код по x, y, z
unsigned short	Reduce1	сокращение 1й оси
unsigned short	Reduce2	сокращение 1й плоскости
bool	Dopos	дополнительная четность
unsigned short	NumBlocks	количество блоков
unsigned short	UWLength	длина уникального слова
unsigned int	UW1, UW2, UW3	уникальные слова
_ORDER	UWDel	удаление уникального слова
bool	Dintrlv	деперемежитель
unsigned short	FindGain	порог синхронизации при поиске
unsigned short	CheckGain	порог синхронизации при проверке
unsigned int	ScrPoly	аддитивный скремблер полином
unsigned int	ScrSet	аддитивный скремблер установка
_ORDER	Scrsw	включение аддитивного скремблера
_SYNC	Scrsync	синхронизация скремблера
unsigned short	DelLastBits	удаление последних бит

3.23. Параметры декодера TPC * N (*TpcNParameters*)

<i>TpcNParameters</i>		
<i>TpcNMode</i>	mode	режим декодера TPC * N
unsigned short	numBlocks	количество блоков

3.24. Параметры декодера TPC Standard (*TpcStandardParameters*)

<i>TpcStandardParameters</i>		
<i>TpcStandardMode</i>	mode	режим TPC Standard
bool	descramblerEn	включение аддитивного дескремблера

3.25. Параметры декодера TPC - разное (*TpcOtherParameters*)

<i>TpcOtherParameters</i>		
<i>TpcOtherMode</i>	mode	режим декодера TPC - разное

3.26. Параметры декодера TPC - Datum (TpcDatumParameters)

TpcDatumParameters		
<i>TpcDatumMode</i>	mode	режим декодера TPC Datum
bool	descramblerEn	включение аддитивного дескремблера

3.27. Параметры декодера TPC - Paradise (TpcParadiseParameters)

TpcParadiseParameters		
<i>TpcParadiseMode</i>	mode	режим декодера TPC Paradise
bool	descramblerEn	включение аддитивного дескремблера

3.28. Параметры декодера Синхронизатор (SynchronizerParameters)

SynchronizerParameters		
unsigned int	blockLen	количество блоков
unsigned short	uwLength	длина уникального слова
unsigned int	UW1, UW2, UW3	уникальные слова
unsigned int	infBlockLen	длина информационного блока
bool	dintrlv	деперемежитель
unsigned int	rowCount	количество строк
unsigned int	colCount	количество столбцов
unsigned char	rdOrder	порядок чтения столбцов (DIRECT – Прямой, BACKWARD – Обратный)
unsigned int	scrPoly	аддитивный скремблер полином
unsigned int	scrSet	аддитивный скремблер установка

3.29. Параметры декодера DVB-S2 CCM (Dvbs2CcmParameters)

Dvbs2CcmParameters		
bool	bbHeader	
bool	multichannel	
bool	syncDelete	

3.30. Параметры декодера LDPC (LdpcParameters)

LdpcParameters		
<i>LdpcMode</i>	mode	режим декодера LDPC

3.31. Параметры декодера TCC 2564 (Tcc2564Parameters)

Tcc2564Parameters		
<i>TccMode</i>	mode	режим декодера TCC 2564
unsigned int	threshold	порог
bool	interleaver	перемежитель

3.32. Параметры декодера VF LDPC (VfLdpcParameters)

VfLdpcParameters		
<i>VfLdpcMode</i>	mode	режим декодера VF LDPC

3.33. Параметры декодера FL LDPC (FlLdpcParameters)

FlLdpcParameters		
<i>FlLdpcSpeed</i>	speed	скорость
<i>FlLdpcMode</i>	mode	режим
bool	scrambler	включение скремблера

3.34. Параметры декодера FX LDPC (FxLdpcParameters)

FxLdpcParameters		
<i>FxLdpcSpeed</i>	speed	скорость
<i>FxLdpcSize</i>	size	размер
bool	scrambler	включение скремблера

3.35. Параметры декодера TCC2d16s (Tcc2d16sParameters)

Tcc2d16sParameters		
<i>Tcc2d16sCoderate</i>	coderate	кодированная скорость
<i>Tcc2d16sBlocklen</i>	blockLen	размер блока
unsigned char	blockCount	количество блоков

3.36. Структура для выключения декодера (DecOFFParameters)

DecOFFParameters		

4. ВОЗВРАЩАЕМЫЕ ЗНАЧЕНИЯ

E_NO_ERROR	0	Выполнено успешно
E_INSTANCE_NOT_INIT	-101	Объекты, необходимые для работы не созданы, не проведена инициализация. Может возникнуть если была вызвана функция настройки/установки параметров перед createDecInstance и initDecoder.
E_LICENSE_NOT_ACCEPTED	-111	Лицензионный ключ не принят
E_MODE_NOT_SUPPORTED	-112	На данной аппаратной конфигурации режим не поддерживается или еще не реализован
E_INCORRECT_PARAMETERS	-113	Неправильные параметры, или параметры вне диапазона допустимых значений
E_FPGA_DEC_NOT_LOADED	-114	ПЛИС декодера не загружена, сначала нужно выполнить loadDecFpga. Может возникнуть при установке параметров декодера без предварительного вызова функции загрузки ПЛИС данного режима.
E_INCOMPATIBLE_MODES	-115	Переданные параметры не соответствуют ранее загруженному режиму. Возникает если в функцию loadDecReg переданы параметры режима, не соответствующие ранее загруженному с помощью функции loadDecFpga.
E_NO_DATA	-116	Нет данных (или данные не готовы)
E_OPEN_FILE_ERROR	-117	Не удалось открыть файл
E_BAD_DATA_FILE	-118	Неправильный файл для загрузки
E_DVB_INTERFACE_LOAD_ERROR	-119	Интерфейс DVB-S2 / DVB-S2X не загрузился
E_INTERNAL_ERROR	-120	Внутренняя ошибка
E_UNHANDLED_EXCEPTION	-121	Внутренняя ошибка
E_DEVICE_ALREADY_CONNECTED	-131	Устройство уже подключено, попытка подключения к занятому устройству
E_DEVICES_NOT_FOUND	-132	Не найдено поддерживаемых устройств ОСПЧ
E_DEVICE_TYPE_NOT_SUPPORTED	-133	Устройство данного типа не поддерживается
E_NO_OSPCHLIB_DEFINED	-141	Для данного типа устройства и среды запуска библиотека ОСПЧ не определена
E_OSPCHLIB_BAD_INTERFACE	-142	Некоторые функции библиотеки ОСПЧ не были загружены
E_OSPCHLIB_ERROR	-143	Необработанная ошибка библиотеки ospchx.dll
E_OSPCHLIB_EXCEPTION	-144	Необработанное исключение при вызове функции библиотеки ospchx.dll
E_IO_ERROR	-151	Ошибка при выполнении _devoutp/_devinp
E_IO_EXCEPTION	-152	Исключение при выполнении _devoutp/_devinp

E_FPGA_LOAD_ERROR	-161	Ошибка загрузки ПЛИС (fpgaLoad вернула не 0)
E_ECC_NOT_READY	-162	ECC не готова
E_DECODER_NOT_SUPPORTED	-163	Тип мезонинного модуля не поддерживается или неправильно определен
E_ECC_LUT_VERIFICATION_ERROR	-164	Ошибка верификации загружаемого LUT-файла
E_SN_READ_ERROR	-165	Ошибка считывания серийного номера
E_DNA_READ_ERROR	-166	Ошибка считывания кода DNA
E_UNHANDLED_DEVICE_ERROR	-180	Прочие ошибки устройства

5. БАЗОВЫЕ ОПЕРАЦИИ

5.1. Подключение к устройству

Для инициализации библиотеки и подключения к устройству нужно выполнить следующие действия:

- 1) Получить количество поддерживаемых устройств в системе, их должно быть более 0: `DeviceSupportedCount`
- 2) Определить соответствие типа устройства (определяется по имени) и номера устройства в системе: `GetFullDeviceName`
- 3) Создать интерфейс взаимодействия с устройством: `CreateDeviceInstance`
- 4) Для устройств ОСПЧ-Ех выполнить загрузку синтезатора ТЧ, указать тип опорного генератора: `rSynthLoad`
- 5) Инициализировать объекты библиотеки `decoder.dll`: `createDeclInstance`, `initDecoder`
- 6) Считать параметры устройства, необходимые для дальнейшей работы: `GetDeviceSlotID`, `getDecoderVersionOnBoard`, `GetLConvertorType`

5.2. Загрузка стандартных декодеров

Последовательность загрузки стандартных декодеров:

- 1) Ввести настройки необходимого режима, используя соответствующую структуру (`ViterbiK7Parameters`, `ViterbiK7S32Parameters`, `ViterbiK8Parameters`, `SskParameters`, `TpcParameters`, `SynchronizerParameters`, `Dvbs2CcmParameters`, `LdpcParameters`, `Tcc2564Parameters`, `VfLdpcParameters`, `FlLdpcParameters`, `FxLdpcParameters`, `DecOFFParameters`)
- 2) Установить параметры структуры `DecoderControl`, параметр `param` должен содержать ссылку на структуру, настроенную в предыдущем шаге.
- 3) Загрузить ПЛИС декодера: `loadDecFpga`
- 4) Ввести лицензионный ключ (если выбранный режим предусматривает лицензии): `installDecLicense`
- 5) Загрузить параметры декодера: `loadDecReg`
- 6) Установить инверсию спектра и порядок каналов: `changeSpectrInversion`, `changeChannelOrder`

- 7) Включить дифференциальный декодер (если данный вид обработки сигнала требуется): [setDiffDecoder](#)
- 8) Установить параметры мультипликативного дескремблера (если данный вид обработки сигнала требуется): [setDescrambler](#)
- 9) Включить инвертор (если данный вид обработки сигнала требуется): [setInvertor](#)
- 10) Установить параметры декодера Рида-Соломона (если данный вид обработки сигнала требуется): [setDecRsless](#), [setDecRsC](#), [setDecRsDvb](#)
- 11) Установить параметры демультимплексора (если данный вид обработки сигнала требуется): [setDmx](#)
- 12) Выполнить сброс схем: [CircuitReset](#)

5.3. Загрузка режима DVB-S2 ACM для ОСПЧ-Е, ОСПЧ-М1

- 1) Остановить DMA-мастер: [MasterStop](#)
- 2) Получить буферы памяти, установить их размер: [MasterGetBuffers](#)
- 3) Загрузить ПЛИС демодулятора в соответствие с аппаратной конфигурацией и режимом: [FPGALoad](#)
- 4) Загрузить ПЛИС декодера конфигурацией `dvb_ecc.bit`: [FPGALoad](#)
- 5) Загрузить таблицы LUT: [FullDVBS2Load](#)
- 6) Установить параметры режима: [FullDVBS2Decoder](#)
- 7) Настроить фильтр: [FullDVBS2RollOff](#)
- 8) Настроить адаптивный корректор: [FullDVBS2ADPCorr](#)
- 9) Настроить инверсию спектра: [FullDVBS2SpectrInv](#)
- 10) Загрузить АЦП: [ADCLoad](#)
- 11) Настроить АРУ1: [FirstGainControl](#)
- 12) Установить несущую частоту: [SetCarrier](#)
- 13) Установить тактовую частоту: [FullDVBS2Clock](#)
- 14) Загрузить фильтры: [FullDVBS2FilterLoad](#)
- 15) Выполнить сброс схем: [FullDVBS2CircuitReset](#), [CircuitReset](#)

5.4. Загрузка режима DVB-S2, DVB-S2X для ОСПЧ-Ех

5.5. Отключение от устройства

- 1) Высвободить объекты библиотеки decoder.dll: [releaseDecoder](#)
- 2) Удалить интерфейс взаимодействия с устройством:
`ReleaseDeviceInstance`