



MaxEye Digital Audio and Video Signal Generation

AM FM RDS TMC Signal Generation Toolkit

Version 1.0.0

Remote Control C API Manual

Contents

1	Introduction	3
2	Installed File Location.....	3
3	Remote Control C APIs.....	3
3.1	MaxEye AM FM Remote TCP Open Connection.....	4
3.2	MaxEye AM FM SG Remote Set Number of Carriers.....	4
3.3	MaxEye AMFM Remote Set RDS Play Duration	5
3.4	MaxEye AM FM Remote Set Generation Mode	6
3.5	MaxEye AM FM Remote Set Hardware Settings.....	7
3.6	MaxEye AM FM Remote Set Signal Mode.....	9
3.7	MaxEye AM FM Remote Set Output Waveform Settings	10
3.8	MaxEye AM FM Remote Set AM Setting	11
3.9	MaxEye AM FM Remote Set FM RDS Settings.....	12
3.10	MaxEye_AMFM_Remote_Set_Audio_Payload_Settings	14
3.11	MaxEye AM FM Remote Set RDS Configuration	16
3.12	MaxEye AM FM Remote Set RDS Basic Tuning Settings	19
3.13	MaxEye AM FM Remote Set RDS AF Settings	21
3.14	MaxEye AM FM Remote Set RDS Message Program Item	23
3.15	MaxEye AM FM Remote Set RDS Message Radio Text.....	25
3.16	MaxEye AM FM Remote Set RDS Message Clock Time	26
3.17	MaxEye AM FM Remote Set RDS Message TMC Variant0.....	28
3.18	MaxEye AM FM Remote Set RDS Message TMC Variant1	31
3.19	MaxEye AM FM Remote Set RDS Message TMC 8A Gp.....	32
3.20	MaxEye AM FM Remote Set RDS EON PSON	34
3.21	MaxEye AMFM Remote Set RDS EON Linkage	37
3.22	MaxEye_AMFM_Remote_Set RDS EON PTYON and TA.....	38
3.23	MaxEye_AMFM_Remote Set RDS EON PINON	40
3.24	MaxEye AM FM Remote Set Carrier Configuration	41

For more information please contact info@maxeyetech.com

3.25	MaxEye AM FM Remote Set Output Waveform File Path.....	42
3.26	MaxEye AM FM Remote Set USRP Hardware Settings.....	43
3.27	MaxEye AM FM Remote Set USRP Play Waveform Settings	44
3.28	MaxEye AM FM Remote Set Play Waveform File Path.....	45
3.29	MaxEye AM FM Remote Set Play Waveform Settings	46
3.30	MaxEye AM FM Remote Set RDS Impairments	47
3.31	MaxEye_AMFM_Remote_Save_Configuration	49
3.32	MaxEye_AMFM_Remote_Load_Configuration.....	50
3.33	MaxEye_AMFM_Remote_Command.....	50
3.34	MaxEye AM FM Remote Get Error Status	51
3.35	MaxEye_AMFM_SG_Remote_Get_OutPut_Indicators	53
3.36	MaxEye AM FM Remote TCP Close Connection	54

1 Introduction

MaxEye Technologies provides generation functions in C for generating the standard complaint signals for various digital audio and video broadcasting standards. This guide explains Remote C APIs structure and how to make use of the APIs to control the AM FM RDS TMC Signal Generation toolkit Soft Front Panel (SFP) remotely, to run programming examples by using NI hardware Vector Signal Generator (NI VSG), Vector Signal Transceiver (NI VST) and Universal Software Radio Peripheral (NI USRP).

The AM FM RDS TMC signal generation is based on the standard IEC 62106:1999.

2 Installed File Location

The remote C API documentation file is located in, C:\Program Files (x86)\MaxEye\Digital Video Toolkits\AM FM RDS TMC Generation\Documentation.

(Note: - For 32-bit Operating System, C API documentation files are installed in C:\Program Files \MaxEye\Digital Video Toolkits\AM FM RDS TMC Generation\Documentation)

The remote C Examples are located in, C:\Program Files (x86)\MaxEye\Digital Video Toolkits\AM FM RDS TMC Generation\Examples\C.

(Note: - For 32-bit Operating System, C examples are installed in, C:\Program Files\MaxEye\Digital Video Toolkits\AM FM RDS TMC Generation\Examples\C)

The AM FM RDS signal generation soft front panel is located in, C:\Program Files (x86)\MaxEye\Digital Video Toolkits\AM FM RDS TMC Generation\Application.

(Note: - For 32-bit Operating System, SFP is located in C:\Program Files\MaxEye\Digital Video Toolkits\AM FM RDS TMC Generation\Application)

You can also find a shortcut to the above location from the windows start menu.

Start->All Programs->MaxEye->Digital Radio Toolkits->AM FM RDS TMC Generation

Note: - For Windows 10, Start ->MaxEye

3 Remote Control C APIs

The Remote C APIs allow user to configure and control the AM FM RDS TMC Signal Generation Soft Front Panel (SFP) remotely through TCP connection. The SFP running in the signal generation hardware acts as a TCP Client and the test program running in the remote system built using the remote C APIs acts as a TCP Server. MaxEye AM FM RDS TMC Signal Generation Toolkit provides set of C APIs to establish connection, configure parameters, initiating and stopping the signal generation and to read the output parameters. The AM FM RDS TMC Signal Generation SFP operates in two modes, remote and local. To control the SFP from remote system the SFP should be in remote mode.

For more information please contact info@maxeyetech.com

3.1 MaxEye AM FM Remote TCP Open Connection

NAME MaxEye_AMFM_Remote_TCP_Open_Connection

DESCRIPTION Opens TCP network connection between AM FM RDS TMC SFP Client and Server applications

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_TCP_Open_Connection  
(  
    char        IPAddress[],  
    uint16_t    PortNumber,  
    int32_t     TimeoutMs,  
    int32_t     ErrorCodeIn,  
    LVRefNum    *ConnectionIDOut,  
    int32_t     *ErrorCodeOut  
)
```

INPUT PARAMETERS

- IPAddress – Specifies the IP Address or network name of the remote system.
- PortNumber – Specifies the port number to establish network connection from server to client system. The default value is 7076.
- TimeoutMs – Specifies TCP Network connection timeout, in milliseconds, that the function waits to complete and return an error. The default value is 20s. A value of -1 indicates to wait indefinitely.
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.2 MaxEye AM FM SG Remote Set Number of Carriers

NAME MaxEye_AMFM_Remote_Set_Number_of_Carriers

For more information please contact info@maxeyetech.com

DESCRIPTION Configures the number of carriers to the AM FM RDS TMC Client SFP Application through TCP Network Connection. The AM FM RDS TMC Signal Generation Toolkit supports generation of multiple AM FM RDS TMC carriers.

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_Number_of_Carriers
(
    LVRefNum      *ConnectionIDIn,
    int32_t       NumberOfCarriers,
    int32_t       ErrorCodeIn,
    LVRefNum      *ConnectionIDOut,
    int32_t       *ErrorCodeOut
)
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- NumberOfCarriers – Configure the number of carriers needs to be generated.
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.3 MaxEye AMFM Remote Set RDS Play Duration

NAME MaxEye_AMFM_Remote_Set_RDS_Play_Duration

DESCRIPTION Configures the total Play Duration in seconds to the AM FM RDS Remote SFP Application through TCP Network Connection.

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_RDS_Play_Duration
(
    LVRefNum      *ConnectionIDIn,
    double        PlayDurationSeconds,

```

For more information please contact info@maxeyetech.com

```

    int32_t      ErrorCodeIn,
    LVRefNum    *ConnectionIDOut,
    int32_t      ErrorCodeOut
  )

```

INPUT PARAMETERS

- **ConnectionIDIn** – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- **PlayDurationSeconds** – Specifies the total play duration in seconds. This control decides the duration of waveform to be generated. To generate longer duration of the waveform increase the Play Duration, Seconds value.
- **ErrorCodeIn** – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- **ConnectionIDOut** – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- **ErrorCodeOut** – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- **Header** – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- **Library** – AM FM RDS TMC Generation.lib
- **DLL** – AM FM RDS TMC Generation.dll

3.4 MaxEye AM FM Remote Set Generation Mode

NAME MaxEye_AMFM_Remote_Set_Waveform_Format

DESCRIPTION Configures the Generation Mode to the AM FM RDS Remote SFP Application through TCP Network Connection.

FUNCTION PROTOTYPE

```

void __cdecl MaxEye_AMFM_Remote_Set_Generation_Mode
(
    LVRefNum    *ConnectionIDIn,
    uint32_t    GenerationMode,
    int32_t      ErrorCodeIn,
    LVRefNum    *ConnectionIDOut,
    int32_t      ErrorCodeOut
)

```

For more information please contact info@maxeyetech.com

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- GenerationMode – Specifies the generation mode of the MaxEye AM FM RDS TMC Signal Generation. The default value is 2. Given below are the possible values.
 - 0 – Generate and Play Waveform (Real Time)
 - 1 – Generate and Play Waveform
 - 2 – Generate and Save Waveform
 - 3 – Play Waveform from File
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.5 MaxEye AM FM Remote Set Hardware Settings

NAME	MaxEye_AMFM_Remote_Set_Hardware_Settings
DESCRIPTION	Configures the VSG/VST hardware settings to the AM FM RDS TMC SFP Application through TCP Network Connection.
FUNCTION PROTOTYPE	<pre>void __cdecl MaxEye_AMFM_Remote_Set_Hardware_Settings (LVRefNum *ConnectionIDIn, AMFM_Hardware_Settings *HardwareSettings, int32_t ErrorCodeIn, LVRefNum *ConnectionIDOut, int32_t *ErrorCodeOut)</pre>

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- HardwareSettings – Specifies the Hardware Settings parameters

```
typedef struct
{
    LStrHandle    RFSGResource;
    double        PowerLevelDBm;
    double        ExternalAttenuationDB;
    double        ArbPreFilterGainDB;
    LStrHandle    RefClockSource;
    double        FrequencyHz;
    LStrHandle    ClockOutputTerminal;
    double        SoftwareScalingFactor;
} AMFM_Hardware_Settings
```

- RFSGResource – Specifies the Resource Name. Select the Resource Name used in NI Measurement and Automation Explorer (NI MAX) for the NI PXIe-5672/5673/5673E or NI PXIe 5644R/45R/46R or NI 5840 device.
- PowerLevelDBm – Specifies the Average Power level of the signal in dBm. The default value is -10.00dBm
- ExternalAttenuationDB – Specifies the external amplification or attenuation, if any, between the NI RF signal generator and the device under test. Positive values for this property represent amplification, and negative values for this property represent attenuation. The default value is 0.
- ArbPreFilterGainDB – Specifies the AWG Pre-filter Gain. The pre-filter gain is applied to the waveform data before any other signal processing. Reduce this value to prevent overflow in the AWG interpolation filters. Other gains on the NI-RFSG device are automatically adjusted to compensate for non-unity AWG pre-filter gain. The default value is -1 dB
- RefClockSource – Specifies the source of the Reference Clock signal. The default value is 0. Given below are the possible values
 - 0 – OnboardClock
 - 1 – RefIn
 - 2 – PXI_CLK
 - 3 – ClkIn
- FrequencyHz – Specifies the Reference Clock rate, in Hertz (Hz). The default value is 10MHz.
- ClockOutputTerminal – Specifies the terminal where the signal will be exported. The default value is 0. Given below are the possible values
 - 0 – Do not export signal
 - 1 – RefOut
 - 2 – RefOut2
 - 3 – ClkOut
 - 4 – PFI0

For more information please contact info@maxeyetech.com

- 5 – PFI1
- 6 – PFI4
- 7 – PFI5
- 8 – PXI_Trig0
- 9 – PXI_Trig1
- 10 – PXI_Trig2
- Software Scaling Factor - Specifies how much to scale the data before writing it with the niRFSG Write Arb Waveform VI. The resulting waveform must be smaller than 1.0 in complex magnitude.
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib, Labview.lib
- DLL – AM FM RDS TMC Generation.dll

3.6 MaxEye AM FM Remote Set Signal Mode

NAME MaxEye_AMFM_Remote_Set_Signal_Mode

DESCRIPTION Configures the Signal Mode Property to the AM FM RDS SFP Application through TCP Network Connection.

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_Signal_Mode
(
    LVRefNum      *ConnectionIDIn,
    uint16_t      SignalMode,
    int32_t       ErrorCodeIn,
    LVRefNum      *ConnectionIDOut,
    int32_t       *ErrorCodeOut
)
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.

For more information please contact info@maxeyetech.com

- SignalMode – Specifies the Signal Mode. The default value is 1. Given below are the possible values.
 - 0 – AM
 - 1 – FM/RDS
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.7 MaxEye AM FM Remote Set Output Waveform Settings

NAME MaxEye_AMFM_Remote_Set_Output_Waveform_Settings

DESCRIPTION Configures the AM FM RDS Output Waveform Settings to the Remote SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_Output_Waveform_Settings
(
    LVRefNum *ConnectionIDIn,
    double HeadroomDB,
    uint16_t OversamplingEnabled,
    double OutputSamplingRate,
    uint16_t SampleWidth,
    int32_t ErrorCodeIn,
    LVRefNum *ConnectionIDOut,
    int32_t *ErrorCodeOut
)
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- HeadroomDB – Specifies the Headroom value. The generator uses this value for scaling the waveform. If PAPR of the signal is higher than the Headroom value then the generator clips the

For more information please contact info@maxeyetech.com

signal. To avoid clipping, the Headroom value should be higher than the PAPR of the signal. The default value is 4 dB.

- **OverSamplingEnabled** – Specifies whether the Oversampling Property is enabled or not. If this property is set to True then the generator resamples the generated signal based on the value configured by the user for the Output Sampling Rate property. The default value is 0 (False). Given below are the possible values
 - 0 – False
 - 1 – True
- **OutputSamplingRate** – Specifies the Output Sampling Rate. The generator resamples the generated signal to a sampling rate equal to the Output Sampling Rate only if the Over Sampling Enabled property is set to True.
- **SampleWidth** – Specifies the sample width to be used to generate waveform file. The default value is 1(16 bit). MaxEye recommend using 16-bits sample width for better signal quality of the generated waveform. Given below are the possible values
 - 0 – 8 bit
 - 1 – 16 bit
- **ErrorCodeIn** – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- **ConnectionIDOut** – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- **ErrorCodeOut** – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.8 MaxEye AM FM Remote Set AM Setting

NAME MaxEye_AMFM_Remote_Set_AM_Setting

DESCRIPTION Configures the AM Settings to the Remote SFP Application through TCP Network Connection

FUNCTION PROTOTYPE `void __cdecl MaxEye_AMFM_Remote_Set_AM_Setting`
(

For more information please contact info@maxeyetech.com

```

    LVRefNum      *ConnectionIDIn,
    int32_t       CarrierIndex,
    double        AMModulationIndex,
    uint16_t      AMModulationType,
    uint16_t      AMSuppressCarrier,
    int32_t       ErrorCodeIn,
    LVRefNum      *ConnectionIDOut,
    int32_t       *ErrorCodeOut
  )

```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- CarrierIndex – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- AMModulationIndex – Specifies the modulation Index for AM. The default value is 0.3.
- AMModulationType – Specifies the AM modulation type. Select type as Double Side Band or Single Side Band or Vestigial Side Band.
- AMSuppressCarrier – Specifies whether AM suppress carrier is False or True. The default value is 0 (False). Given below are the possible values
 - 0 – False
 - 1 – True
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.9 MaxEye AM FM Remote Set FM RDS Settings

NAME MaxEye_AMFM_Remote_Set_FMRDS_Setting

For more information please contact info@maxeyetech.com

DESCRIPTION Configures the FM Settings to the Remote AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_FMRDS_Settings
(
    LVRefNum      *ConnectionIDIn,
    int32_t       CarrierIndex,
    FMSettings    *FMSettings,
    int32_t       ErrorCodeIn,
    LVRefNum      *ConnectionIDOut,
    int32_t       *ErrorCodeOut
)
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- CarrierIndex – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- FMSettings – Specifies the FM Settings parameters.

```
typedef struct
{
    uint16_t      FMMode;
    double        PilotDeviation;
    uint16_t      DataServiceEnabled;
    double        DataDeviation;
    double        FMDeviationHz;
    uint16_t      PreEmphasis;
    uint16_t      DataServiceMode;
} FMSettings
```

- FMMode – Specify the FM mode as Mono or Stereo. The default value is 1 (Stereo). Given below are the possible values
 - 0 – Mono
 - 1 – Stereo
- PilotDeviation – Configure the pilot deviation with appropriate value in %. The default value is 10%.
- DataServiceEnabled – Configure this control to True if RDS also need to be transmitted with FM. Default is true. If False, transmitted FM will not have RDS. The default value is 1 (True). Given below are the possible values
 - 0 – False
 - 1 – True
- DataDeviation – Configure Data deviation with appropriate value in %. The default value is 6%.

- **FMDeviationHz** – Configure FM deviation with appropriate value in Hz. The default value is 75 kHz.
- **PreEmphasis** – Pre-emphasis provides increased amplitude to the higher modulating frequencies prior to modulation under a well-defined pre-emphasis (high-pass filter) curve, in order to compensate the effect of noise with increasing modulation frequency. The default value is 1 (50us). Given below are the possible values
 - 0 – OFF
 - 1 – 50us
 - 2 – 75us
- **DataServiceMode** – Select mode as RDS or RBDS. Radio Data System (RDS) is a communications protocol standard for embedding small amounts of digital information in conventional FM radio broadcasts. RDS standardizes several types of information transmitted, including time, station identification and programme information. Radio Broadcast Data System (RBDS) is the official name used for the U.S. version of RDS. The default value is 1 (RDS). Given below are the possible values
 - 0 – RDS
 - 1 – RBDS
- **ErrorCodeIn** – Specifies the error code. The **ErrorCodeIn** can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- **ConnectionIDOut** – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- **ErrorCodeOut** – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.10 MaxEye_AMFM_Remote_Set_Audio_Payload_Settings

NAME MaxEye_AMFM_Remote_Set_Audio_Payload_Settings

DESCRIPTION Configures the AM FM Audio Settings to the Remote AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE `void __cdecl MaxEye_AMFM_Remote_Set_Audio_Payload_Settings`

For more information please contact info@maxeyetech.com

```

    (
        LVRefNum          *ConnectionIDIn,
        int32_t           CarrierIndex,
        Audio_Settings    *AudioSettings,
        int32_t           Left_Channel_Length,
        double            LeftChannelFrequencyHz[],
        int32_t           Right_Channel_Length,
        double            RightChannelFrequencyHz[],
        int32_t           ErrorCodeIn,
        LVRefNum          *ConnectionIDOut,
        int32_t           *ErrorCodeOut
    )
  
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- CarrierIndex – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- AudioSettings – Specifies the Audio Settings parameters

```

typedef struct
{
    uint16_t      AudioPayloadType;
    LStrHandle    AudioFilePath;
    int32_t       NoOfTonesInLeftChannel;
    int32_t       NoOfTonesInRightChannel;
    uint16_t      AudioPayloadMode;
} Audio_Settings
  
```

- AudioPayloadType – Specifies audio payload type. Configure the type as either Tone(s) or Wav File. The default value is 0 (Tone(s)). Given below are the possible values
 - 0 – Tone(s)
 - 1 – Wav File
- AudioFilePath – Specifies the audio file path. Configure this control with a valid audio file path.
- NoOfTonesInLeftChannel – Specifies the number of tones in left channel, this control corresponds to the Left Channel Tones (Hz) control. No. of Tones in Left Channel and the number of element in the Left Channel Tones (Hz) control must be same.
- AudioPayloadMode – Specifies the sync between left and right channel. The default value is 0 (L # R). Given below are the possible values
 - 0 – L # R
 - 1 – L = R
 - 2 – L = -R
 - 3 – L Only
 - 4 – R Only

For more information please contact info@maxeyetech.com

- NoOfTonesInRightChannel – Specifies the number of tones in right channel, this control corresponds to the right Channel Tones (Hz) control. No. of Tones in Left Channel and the number of element in the right Channel Tones (Hz) control must be same.
- Left_Channel_Length – Specifies the number of left channel frequencies.
- LeftChannelFrequencyHz – Specifies the frequencies in the left channel. This control is an array. Configure this control with the same number of element as the control len has been configured.
- Right_Channel_Length – Specifies the number of right channel frequencies.
- RightChannelFrequencyHz – Specifies the frequencies in the right channel. This control is an array. Configure this control with the same number of element as the control len2 has been configured.
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib, Labview.lib
- DLL – AM FM RDS TMC Generation.dll

3.11 MaxEye AM FM Remote Set RDS Configuration

NAME MaxEye_AMFM_Remote_Set_RDS_Configuration

DESCRIPTION Configures the FM RDS Global Settings to the Remote AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```

void __cdecl MaxEye_AMFM_Remote_Set_RDS_Configuration
(
    LVRefNum          *ConnectionIDIn,
    int32_t           CarrierIndex,
    RDS_Configuration *RDSConfiguration,
    int32_t           ErrorCodeIn,
    LVRefNum          *ConnectionIDOut,
    int32_t           *ErrorCodeOut
)
  
```

For more information please contact info@maxeyetech.com

INPUT PARAMETERS

- **ConnectionIDIn** – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- **CarrierIndex** – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- **RDSConfiguration** – Specifies the RDS Configuration parameters

```
typedef struct
{
    uint8_t      CountryCode;
    uint16_t     CoverageArea;
    uint8_t      ProgramReferenceNumber;
    uint16_t     ProgramTypeRDS;
    uint16_t     ProgramTypeRBDS;
    uint16_t     TrafficProgramIdentification;
    int32_t      NumberOfMessages;
} RDS_Configuration
```

- **CountryCode** – Specifies the valid and unique country code.
- **CoverageArea** – Specifies the coverage area. Select one of the coverage areas. The default value is 0 (Local). Given below are the possible values
 - 0 – Local
 - 1 – International
 - 2 – National
 - 3 – Supra-regional
 - 4 – Regional 1
 - 5 – Regional 2
 - 6 – Regional 3
 - 7 – Regional 4
 - 8 – Regional 5
 - 9 – Regional 6
 - 10 – Regional 7
 - 11 – Regional 8
 - 12 – Regional 9
 - 13 – Regional 10
 - 14 – Regional 11
 - 15 – Regional 12
- **ProgramReferenceNumber** – Specifies the valid Program Reference Number. The default value is 1.
- **ProgramTypeRDS** – Specifies the type of programme transmitted. Configure this control only if selected Data Service Mode is RDS. The default value is 1 (News). Refer the table given below for the possible values

For more information please contact info@maxeyetech.com

- ProgramTypeRBDS – Same as ProgramTypeRDS.

PTY code	RDS program type (EU)	RBDS program type (North America)
0	No programme type or undefined	No program type or undefined
1	News	News
2	Current affairs	Information
3	Information	Sports
4	Sport	Talk
5	Education	Rock
6	Drama	Classic rock
7	Culture	Adult hits
8	Science	Soft rock
9	Varied	Top 40
10	Pop music	Country
11	Rock music	Oldies
12	Easy listening	Soft
13	Light classical	Nostalgia
14	Serious classical	Jazz
15	Other music	Classical
16	Weather	Rhythm and blues
17	Finance	Soft rhythm and blues
18	Children's programmes	Language
19	Social affairs	Religious music
20	Religion	Religious talk
21	Phone-in	Personality
22	Travel	Public
23	Leisure	College
24	Jazz music	Spanish Talk
25	Country music	Spanish Music
26	National music	Hip Hop
27	Oldies music	Unassigned
28	Folk music	Unassigned
29	Documentary	Weather
30	Alarm test	Emergency test
31	Alarm	Emergency

- TrafficProgramIdentification – Specifies whether the Traffic Program Identification is ON or OFF. Traffic Program identifies if the station is capable of sending traffic announcements or not. This control works with Traffic Announcement control given in Basic Tuning and Switching section. Set both the controls Traffic Program Identification and Traffic Announcement control to ON state for sending traffic announcement. The default value is 0 (OFF). Given below are the possible values
- 0 – OFF
 - 1 – ON
- NumberOfMessages – Specifies the number of RDS messages intended to be configured. The default value is 1.
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

For more information please contact info@maxeyetech.com

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.12 MaxEye AM FM Remote Set RDS Basic Tuning Settings

NAME	MaxEye_AMFM_Remote_Set_RDS_Basic_Tuning_Setting
DESCRIPTION	Configures the RDS Message Basic Tuning Settings to the Remote AM FM RDS SFP Application through TCP Network Connection
FUNCTION PROTOTYPE	<pre> void __cdecl MaxEye_AMFM_Remote_Set_RDS_Basic_Tuning_Settings (LVRefNum *ConnectionIDIn, int32_t CarrierIndex, int32_t MessageIndex, RDS_Basic_Tuning_Settings *RDSBasicTuningSettings, int32_t ErrorCodeIn, LVRefNum *ConnectionIDOut, int32_t *ErrorCodeOut) </pre>

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- CarrierIndex – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- MessageIndex – Specifies the index value to keep a track on the current RDS message under processing.
- RDSBasicTuningSettings – Allows user to configure the following parameters

```

typedef struct
{
    uint16_t      GroupVersion;

```

For more information please contact info@maxeyetech.com

```

    uint16_t      TrafficAnnouncement;
    uint16_t      MusicSpeechSwitchCode;
    uint8_t       DecoderIdentificationCode;
    LStrHandle     ProgramServiceName;
    uint16_t      MessageType;
} RDS_Basic_Tuning_Settings

```

- **GroupVersion** – Specifies the Group Version. In Version A, the PI code is inserted in block 1 only. In Version B, the PI code is inserted in block 1 and block 3. The default value is 0 (Version A). Given below are the possible values
 - 0 – Version A
 - 1 – Version B
- **TrafficAnnouncement** – Specifies whether the Traffic Announcement is ON or OFF. Set Traffic Announcement property ON to send traffic announcement. Traffic Program Identification too must be ON. The default value is 0 (OFF). Given below are the possible values
 - 0 – OFF
 - 1 – ON
- **MusicSpeechSwitchCode** – Specifies whether music program or speech program is transmitted. The signal supports tuner with two individual volume modes one for music, the other for speech. This enables the user to configure the settings according to individual requirements. The default value is 1 (Music). Given below are the possible values
 - 0 – Speech
 - 1 – Music
- **DecoderIdentificationCode** – Specifies which possible operating modes are appropriate for use with the broadcast audio and to indicate if PTY codes are switched dynamically. The default value is 1.
- **ProgramServiceName** – Specifies the Program Service Name. The Program Service Name contains max. 8 alphanumeric characters. It can be used to inform the user about the station ID. The PS is usually displayed by every RDS enabled tuner.
- **MessageType** – Specifies the type of RDS Message. The default value is 0 (Basic Tuning and Switching Information). Given below are the possible values
 - 0 – Basic Tuning and Switching Information
 - 1 – Programme Item Number and Slow Labelling Codes
 - 2 – Radio Text
 - 3 – Clock-time and date
 - 4 – Traffic Message Channel
 - 5 – EnhancedOtherNetworks
- **ErrorCodeIn** – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- **ConnectionIDOut** – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- **ErrorCodeOut** – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- **Header** – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- **Library** – AM FM RDS TMC Generation.lib, Labview.lib
- **DLL** – AM FM RDS TMC Generation.dll

3.13 MaxEye AM FM Remote Set RDS AF Settings

NAME	MaxEye_AMFM_Remote_Set_RDS_AF_Setting
DESCRIPTION	Configures the RDS AF Settings to the Remote AM FM RDS SFP Application through TCP Network Connection
FUNCTION PROTOTYPE	<pre> void __cdecl MaxEye_AMFM_Remote_Set_RDS_AF_Settings (LVRefNum *ConnectionIDIn, int32_t CarrierIndex, int32_t MessageIndex, RDS_AF_Settings *RDSAFSettings, int32_t Alternate_Frequencies_Length, double AlternateFrequencies[], int32_t NumberOfAFsInListLength, uint8_t NumberOfAFsInList[], int32_t Tuning_Frequency_Length, double TuningFrequency[], int32_t AF_List_Length, double AFList[], int32_t AF_Type_Length, int32_t AFType[], int32_t ErrorCodeIn, LVRefNum *ConnectionIDOut, int32_t *ErrorCodeOut) </pre>

INPUT PARAMETERS

- **ConnectionIDIn** – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.

For more information please contact info@maxeyetech.com

- CarrierIndex – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- MessageIndex – Specifies the index value to keep a track on the current RDS message under processing.
- RDSAFSettings – Specifies the RDS AF Settings parameters

```
typedef struct
{
    uint16_t      AlternativeFrequencyMethod;
    int32_t      NumberOfAlternateFrequencies;
    int32_t      NumberOfAFLists;
} RDS_AF_Settings
```

- AlternateFrequencyMethod – Specifies the Alternate Frequency Method. Select between Method A or Method B. Method A is used for lists up to 25 in number. Method B is used for larger lists. The default value is 0 (Method A). Given below are the possible values
 - 0 – Method A
 - 1 – Method B
- NumberOfAlternateFrequencies – Specifies the total number of alternative frequencies.
- NumberOfAFLists – Specifies the total number of AF Lists.
- Alternate_Frequencies_Length – Specifies the size of AlternateFrequencies.
- AlternateFrequencies – Specifies the alternative frequencies
- NumberofAFsInListLength – Specifies the size of NumberOfAFsInList.
- NumberOfAFsInList – Specifies the number of alternate frequencies in each list. Each row corresponds to each list.
- Tuning_Frequency_Length – Specifies the size of TuningFrequency
- TuningFrequency – Specify the tuning frequency. The alternate frequency list is specified for this tuning frequency.
- AF_List_Length – Specifies the size of AFList.
- AFList – Specify the alternate frequencies for each AF list. Each row corresponds to each AF list. Enter the alternate frequencies column wise for each list
- AF_Type_Length – Specifies the size of AFType.
- AFType – Specify the type of each alternate frequency entered in the AF List. Each row corresponds to each AF list. Configure the type of alternate frequencies column wise for each list. The default value is 0 (Same Block and Programme). Given below are the possible values
 - 0 – Same Block and Programme
 - 1 – Different Regions or Programme

- **ErrorCodeIn** – Specifies the error code. The **ErrorCodeIn** can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- **ConnectionIDOut** – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- **ErrorCodeOut** – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.14 MaxEye AM FM Remote Set RDS Message Program Item

NAME	MaxEye_AMFM_Remote_Set_RDS_Message_Program_Item
DESCRIPTION	Configures the RDS Message Program Type Settings to the Remote AM FM RDS SFP Application through TCP Network Connection
FUNCTION PROTOTYPE	<pre> void __cdecl MaxEye_AMFM_Remote_Set_RDS_Message_Program_Item (LVRefNum *ConnectionIDIn, int32_t CarrierIndex, int32_t MessageIndex, Program_Item *RDSMessageProgramItem, int32_t ErrorCodeIn, LVRefNum *ConnectionIDOut, int32_t *ErrorCodeOut) </pre>

INPUT PARAMETERS

- **ConnectionIDIn** – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- **CarrierIndex** – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- **MessageIndex** – Specifies the index value to keep a track on the current RDS message under processing.
- **RDSMessageProgramItem** – Allows user to configure the following parameters

For more information please contact info@maxeyetech.com


```
typedef struct
{
    uint16_t    LinkageActuator;
    uint8_t     ExtendedCountryCode;
    uint8_t     Day;
    uint8_t     Hour;
    uint8_t     Minute;
    uint16_t    LanguageCode;
    uint16_t    GroupVersion;
    uint16_t    MessageType;
} Program_Item
```

- LinkageActuator – Specifies whether the Linkage Actuator is ON or OFF. Linkage actuator allows linking this station to another station that offers the same program. The default value is 0 (OFF). Given below are the possible values
 - 0 – OFF
 - 1 – ON
- ExtendedCountryCode – Specifies the unique and valid extended country code for the selected country. This control is used only when the Extended Paging Protocol (EPP) is used.
- Day – Specifies the day of the broadcast.
- Hour – Specifies the hour of the broadcast.
- Minute – Specifies the minute of the broadcast.
- LanguageCode – Specifies the language of the target audience.
- GroupVersion – Specifies the Group Version. Select the desired Group Version. In Version A, the PI code is inserted in block 1 only. In Version B, the PI code is inserted in block 1 and block 3. The default value is 0 (Version A). Given below are the possible values
 - 0 – Version A
 - 1 – Version B
- MessageType – Specifies the type of RDS Message. The default value is 0 (Basic Tuning and Switching Information). Given below are the possible values
 - 0 – Basic Tuning and Switching Information
 - 1 – Programme Item Number and Slow Labelling Codes
 - 2 – Radio Text
 - 3 – Clock-time and date
 - 4 – Traffic Message Channel
 - 5 – EnhancedOtherNetworks
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.

- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.15 MaxEye AM FM Remote Set RDS Message Radio Text

NAME MaxEye_AMFM_Remote_Set_RDS_Message_Radio_Text

DESCRIPTION Configures the RDS Text Settings to the Remote AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_RDS_Message_Radio_Text
(
    LVRefNum          *ConnectionIDIn,
    int32_t           CarrierIndex,
    int32_t           MessageIndex,
    RDS_Radio_Text   *Radio_Text,
    int32_t           ErrorCodeIn,
    LVRefNum          *ConnectionIDOut,
    int32_t           *ErrorCodeOut
)
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- CarrierIndex – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- MessageIndex – Specifies the index value to keep a track on the current RDS message under processing.
- Radio_Text – Specifies Radio Text parameters

```
typedef struct
{
    uint16_t   TypeABFlag;
    LStrHandle RadioText;
    uint16_t   GroupVersion;
}
```

For more information please contact info@maxeyetech.com

```
    uint16_t      MessageType;  
} RDS_Radio_Text
```

- TypeABFlag – Specifies if a screen clear is requested or not. If the receiver detects a change in the Type A/B Flag, then the whole Radio Text display shall be cleared and the newly received Radio Text message segments shall be written into the display.
- RadioText – Specifies the Radio Text to be transmitted.
- GroupVersion – Specifies the Group Version. In Version A, the PI code is inserted in block 1 only. In Version B, the PI code is inserted in block 1 and block 3. The default value is 0 (Version A). Given below are the possible values
 - 0 – Version A
 - 1 – Version B
- MessageType – Specifies the type of RDS Message. The default value is 0 (Basic Tuning and Switching Information). Given below are the possible values
 - 0 – Basic Tuning and Switching Information
 - 1 – Programme Item Number and Slow Labelling Codes
 - 2 – Radio Text
 - 3 – Clock-time and date
 - 4 – Traffic Message Channel
 - 5 – EnhancedOtherNetworks
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib, Labview.lib
- DLL – AM FM RDS TMC Generation.dll

3.16 MaxEye AM FM Remote Set RDS Message Clock Time

NAME MaxEye_AMFM_Remote_Set_RDS_Message_Clock_Time

For more information please contact info@maxeyetech.com

DESCRIPTION Configures the FM RDS Message Clock Time Settings to the Remote AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_RDS_Message_Clock_Time
(
    LVRefNum          *ConnectionIDIn,
    int32_t           CarrierIndex,
    int32_t           MessageIndex,
    RDS_Clock_Time   *RDSMessageClockTime,
    int32_t           ErrorCodeIn,
    LVRefNum          *ConnectionIDOut,
    int32_t           *ErrorCodeOut
)
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- CarrierIndex – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- MessageIndex – Specifies the index value to keep a track on the current RDS message under processing.
- RDSMessageClockTime – Specifies RDS Message Clock Time parameters

```
typedef struct
{
    uint16_t      MessageType;
    uint16_t      GroupVersion;
    int32_t       Year;
    int32_t       Month;
    int32_t       Date;
    uint16_t      UTCHour;
    uint16_t      UTCMinute;
    int16_t       LocalTimeOffset;
} RDS_Clock_Time
```

- Year – Specifies the year of broadcast.
- Month – Specifies the month of broadcast.
- Date – Specifies the date of broadcast.
- UTCHour – Specifies the hour of broadcast in UTC time.
- UTCMinute – Specifies the Minute of broadcast in UTC time.
- LocalTimeOffset – Specifies the Local Time Offset, if any.
- GroupVersion – Specifies the Group Version. In Version A, the PI code is inserted in block 1 only. In Version B, the PI code is inserted in block 1 and block 3. The default value is 0 (Version A). Given below are the possible values
 - 0 – Version A
 - 1 – Version B

For more information please contact info@maxeyetech.com

- **MessageType** – Specifies the type of RDS Message. The default value is 0 (Basic Tuning and Switching Information). Given below are the possible values
 - 0 – Basic Tuning and Switching Information
 - 1 – Programme Item Number and Slow Labelling Codes
 - 2 – Radio Text
 - 3 – Clock-time and date
 - 4 – Traffic Message Channel
 - 5 – EnhancedOtherNetworks
- **ErrorCodeIn** – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- **ConnectionIDOut** – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- **ErrorCodeOut** – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.17 MaxEye AM FM Remote Set RDS Message TMC Variant0

NAME MaxEye_AMFM_Remote_Set_RDS_Message_TMC_Variant0

DESCRIPTION Configures the RDS Message TMC Variant0 Settings to the Remote AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```

void __cdecl MaxEye_AMFM_Remote_Set_RDS_Message_TMC_Variant0
(
    LVRefNum          *ConnectionIDIn,
    int32_t           CarrierIndex,
    int32_t           MessageIndex,
    RDS_TMC_Variant0 *RDSMessageTMCVariant0,
    int32_t           ErrorCodeIn,
    LVRefNum          *ConnectionIDOut,
    int32_t           *ErrorCodeOut
)
  
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- CarrierIndex – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- MessageIndex – Specifies the index value to keep a track on the current RDS message under processing.
- RDSMessageTMCVariant0 – Specifies RDS Message Variant 0 parameters

```
typedef struct
{
    uint16_t      MessageType;
    uint16_t      GroupVersion;
    uint8_t       LocationTableNumber;
    uint16_t      National;
    uint16_t      Regional;
    uint16_t      Urban;
    uint16_t      AlternativeFrequencyIndicator;
    uint16_t      International;
    uint16_t      TransmissionMode;
} RDS_TMC_Variant0
```

- GroupVersion – Specifies the Group Version. In Version A, the PI code is inserted in block 1 only. In Version B, the PI code is inserted in block 1 and block 3. The default value is 0 (Version A). Given below are the possible values
 - 0 – Version A
 - 1 – Version B
- MessageType – Specifies the type of RDS Message. The default value is 0 (Basic Tuning and Switching Information). Given below are the possible values
 - 0 – Basic Tuning and Switching Information
 - 1 – Programme Item Number and Slow Labelling Codes
 - 2 – Radio Text
 - 3 – Clock-time and date
 - 4 – Traffic Message Channel
 - 5 – EnhancedOtherNetworks
- LocationTableNumber – Specifies the location table number.
- International – Specifies whether International is ON or OFF. The default value is 0 (OFF). Always set this control OFF. Given below are the possible values
 - 0 – OFF
 - 1 – ON
- National – Specifies whether National is ON or OFF. The default value is 0 (OFF). Always set this control OFF. Given below are the possible values

For more information please contact info@maxeyetech.com

- 0 – OFF
- 1 - ON
- Regional – Specifies whether Regional is ON or OFF. The default value is 0 (OFF). Always set this control OFF. Given below are the possible values
 - 0 – OFF
 - 1 - ON
- Urban – Specifies whether Urban is ON or OFF. The default value is 0 (OFF). Always set this control OFF. Given below are the possible values
 - 0 – OFF
 - 1 - ON
- AlternativeFrequencyIndicator – Specifies if all frequencies of a programme network fulfil the conditions for using Tuning Information or not. If set to 1, indicates that all alternative frequencies broadcast in the 0A-group for the audio program can be used to switch to the next station. In all other cases the AFI bit is set to 0. The default value is 0 (OFF). Always set this control OFF. Given below are the possible values
 - 0 – OFF
 - 1 - ON
- TransmissionMode – Specifies the Transmission Mode. Choose the Basic Mode as Enhanced mode is not currently supported by the toolkit. The default value is 0 (Basic). Given below are the possible values
 - 0 – Basic
 - 1 – Enhanced
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.18 MaxEye AM FM Remote Set RDS Message TMC Variant1

NAME MaxEye_AMFM_Remote_Set_RDS_Message_TMC_Variant1

DESCRIPTION Configures the RDS Message TMC Variant1 Settings to the Remote AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_RDS_Message_TMC_Variant1
(
    LVRefNum *ConnectionIDIn,
    int32_t CarrierIndex,
    int32_t MessageIndex,
    RDS_TMC_Variant1 *RDSMessageTMCVariant1,
    int32_t ErrorCodeIn,
    LVRefNum *ConnectionIDOut,
    int32_t *ErrorCodeOut
)
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- CarrierIndex – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- MessageIndex – Specifies the index value to keep a track on the current RDS message under processing.
- RDSMessageTMCVariant1 – Specifies RDS Message Variant 1 parameters

```
typedef struct
{
    uint8_t      ServiceIdentifier;
    uint16_t     Inter8AGap;
    uint16_t     DelayTimeTd;
    uint16_t     ActivityTimeTa;
    uint16_t     WindowTimeTw;
} RDS_TMC_Variant1
```

- ServiceIdentifier – The Service Identifier identifies the data service provider in its country.
- Inter8AGap – Specifies the Inter 8A Gap. The default value is 3 (11 Groups). Always choose 11 Groups. Given below are the possible values
 - 0 – 3 Groups
 - 1 – 5 Groups
 - 2 – 8 Groups
 - 3 – 11 Groups
- DelayTimeTd – Specifies the Dealy Time. The default value is 1. Always choose 1 sec. Given below are the default values
 - 0 – 0 sec

For more information please contact info@maxeyetech.com

- 1 – 1 sec
- 2 – 2 sec
- 3 – 3 sec
- ActivityTimeTa – Specifies the Activity Time. The default value is 0. Always choose 1 sec. Given below are the default values
 - 0 – 1 sec
 - 1 – 2 sec
 - 2 – 4 sec
 - 3 – 8 sec
- WindowTimeTw – Specifies the Window Time. The default value is 1. Always choose 1 sec. Given below are the default values
 - 0 – 0 sec
 - 1 – 1 sec
 - 2 – 2 sec
 - 3 – 4 sec
 - 4 – 8 sec
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.19 MaxEye AM FM Remote Set RDS Message TMC 8A Gp

NAME MaxEye_AMFM_Remote_Set_RDS_Message_TMC_8A_Gp

DESCRIPTION Configures the RDS Message 8A Settings to the Remote AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE `void __cdecl MaxEye_AMFM_Remote_Set_RDS_Message_TMC_8A_Gp`
`(`

For more information please contact info@maxeyetech.com

```

LVRefNum          *ConnectionIDIn,
int32_t           CarrierIndex,
int32_t           MessageIndex,
RDS_TMC_Variant_8A *RDSMessageTMC8A,
int32_t           ErrorCodeIn,
LVRefNum          *ConnectionIDOut,
int32_t           *ErrorCodeOut

```

```
)
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- CarrierIndex – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- MessageIndex – Specifies the index value to keep a track on the current RDS message under processing.
- RDSMessageTMC8A – Specifies RDS Message TMC 8A parameters

```

typedef struct
{
    uint64_t LocationCode;
    uint64_t EventCode;
    uint16_t Direction;
    uint16_t DiversionAdvice;
    uint16_t DurationCode;
    uint16_t ExtentCode;
} RDS_TMC_Variant_8A

```

- LocationCode – Specifies the location where the event occurred. The default value is 0.
- EventCode – Specifies the Even Code. Configure this control with standard event codes to define the event occurred. The default value is 0.
- Direction – Specifies the direction as positive or negative from the location where the event has happened. The default value is 0 (Positive). Given below are the possible values
 - 0 – Positive
 - 1 – Negative
- DiversionAdvice – Specifies whether Diversion Advice is ON or OFF. Set Diversion Advice to ON, to receive diversion advices the navigation system. The default value is 0 (OFF). Given below are the possible values
 - 0 – OFF
 - 1 – ON
- DurationCode – Specifies the Duration Code. Configure this control to simulate the information about the timeframe of the traffic event or any event. The default value is 0. Possible values are form 0 to 7.

- ExtentCode – Specifies the extension of the current event, it is measured in terms of nearby Location Table entries. The default value is 0. Possible values are form 0 to 7.
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.20 MaxEye AM FM Remote Set RDS EON PSON

NAME	MaxEye_AMFM_Remote_Set_RDS_EON_PSON
DESCRIPTION	Configures the RDS EON PSON Settings to the Remote AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
(
  LVRefNum          *ConnectionIDIn,
  int32_t           CarrierIndex,
  int32_t           MessageIndex,
  RDS_Enhanced_Other_Networks_PSON *Enhanced_Other_Network_PSON,
  int32_t           LengthofAF,
  double            AlternateFrequenciesON[],
  int32_t           ErrorCodeIn,
  LVRefNum          *ConnectionIDOut,
  int32_t           *ErrorCodeOut
);
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- CarrierIndex – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.

For more information please contact info@maxeyetech.com

- MessageIndex – Specifies the index value to keep a track on the current RDS message under processing.
- RDSMessage EON PSON – Specifies RDS EON PSON Message parameters

```
typedef struct {  
    Enum13 MessageType;  
    uint8_t EONCountryCode;  
    Enum25 FrequencyMethod;  
    Enum15 EONCoverageArea;  
    uint8_t EONReferenceNumber;  
    uint16_t EONPICodeValue;  
    LStrHandle EONProgramServiceName;  
    Enum12 GroupVersion;  
    Enum26 EONInformation;  
} RDS_Enhanced_Other_Networks_PSON;
```

- MessageType – Specifies the type of RDS Message. The default value is 0 (Basic Tuning and Switching Information). Given below are the possible values
 - 0 – Basic Tuning and Switching Information
 - 1 – Programme Item Number and Slow Labelling Codes
 - 2 – Radio Text
 - 3 – Clock-time and date
 - 4 – Traffic Message Channel
 - 5 – EnhancedOtherNetworks
- EONCountryCode – Specifies the valid and unique country code.
- FrequencyMethod – Select one of the Frequency methods. The default value is 0 (AlternateFrequency). Given below are the possible values
 - 0 – AlternateFrequency
 - 1 – MappedFrequency
- EONCoverageArea – Specifies the coverage area. Select one of the coverage areas. The default value is 0 (Local). Given below are the possible values
 - 0 – Local
 - 1 – International
 - 2 – National
 - 3 – Supra-regional
 - 4 – Regional 1
 - 5 – Regional 2
 - 6 – Regional 3
 - 7 – Regional 4
 - 8 – Regional 5
 - 9 – Regional 6
 - 10 – Regional 7
 - 11 – Regional 8

For more information please contact info@maxeyetech.com

- 12 – Regional 9
- 13 – Regional 10
- 14 – Regional 11
- 15 – Regional 12
- EONReferenceNumber – Specifies the valid Program Reference Number. The default value is 1.
- EONProgramServiceName – Specifies the Program Service Name. The Program Service Name contains max. 8 alphanumeric characters. It can be used to inform the user about the station ID. The PS is usually displayed by every RDS enabled tuner.
- GroupVersion – Specifies the Group Version. In Version A, the PI code is inserted in block 1 only. In Version B, the PI code is inserted in block 1 and block 3. The default value is 0 (Version A). Given below are the possible values
 - 0 – Version A
 - 1 – Version B
- EONInformation – Specifies one of the Information Type. The default value is 0 (PTYON And TA ON)
 - 0 – PSON
 - 1 – LinkageInformation
 - 2 – PTYON And TA ON
 - 3 – PIN ON
- LengthofAF – Specifies the Size of Alternate Frequency.
- AlternateFrequenciesON[] – Specifies the Alternate Frequencies.
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.21 MaxEye AMFM Remote Set RDS EON Linkage

NAME MaxEye_AMFM_Remote_Set_RDS_EON_Linkage

DESCRIPTION Configures the RDS EON Linkage Settings to the Remote AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
(
    LVRefNum          *ConnectionIDIn,
    int32_t           CarrierIndex,
    int32_t           MessageIndex,
    RDS_Enhanced_Other_Networks_Linkage *Enhanced_Other_Network_Linkage,
    int32_t           LengthofMappedFreq,
    double            MappedFrequencies[],
    int32_t           MappedFrequency_Row,
    int32_t           ErrorCodeIn,
    LVRefNum          *ConnectionIDOut,
    int32_t           *ErrorCodeOut
);
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- CarrierIndex – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- MessageIndex – Specifies the index value to keep a track on the current RDS message under processing.
- RDSMessage EON Linkage – Specifies RDS EON Linkage Message parameters

```
typedef struct {
    uint16_t LinkageSetNumberLSN;
    Enum9 LinkageActuatorLA;
    Enum9 InternationalLinkageSetILS;
    Enum9 ExtendedGenericEG;
} RDS_Enhanced_Other_Networks_Linkage;
```

- LinkageSetNumberLSN – Specifies the Linkage Set Number LSN for other programs. The LSN is a 12-bit number.
- LinkageActuatorLA – Specifies the Linkage Set actuator for Enables/disables in the programs. The default value is 0 (off). Given below are the possible values
 - 0 – Off
 - 1 – On
- InternationalLinkageSetILS – Specifies the International Linkage Set for Enables/disables in the programs. The default value is 0 (off). Given below are the possible values
 - 0 – Off

For more information please contact info@maxeyetech.com

- 1 – On
- ExtendedGenericEG – Specifies the Extended Generic for Enables/disables in the programs. The default value is 0 (off). Given below are the possible values
- 0 – Off
- 1 – On
- LengthofMappedFreq – Specifies the Length of Mapped Frequency.
- MappedFrequencies[] – Specifies the Mapped Frequencies.
- MappedFrequency_Row – Specifies the size of Mapped Frequency.
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.22 MaxEye_AMFM_Remote_Set RDS EON PTYON and TA

NAME MaxEye_AMFM_Remote_Set_RDS_EON_PTYON_and_TA

DESCRIPTION Configures the RDS EON PTYON and TA Settings to the Remote AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
(
  LVRefNum          *ConnectionIDIn,
  int32_t           CarrierIndex,
  int32_t           MessageIndex,
  RDS_Enhanced_Other_Networks_PTYON_and_TA
  int32_t           *Enchanced_Other_Network_PTYON_TA,
  LVRefNum          ErrorCodeIn,
  int32_t           *ConnectionIDOut,
  int32_t           *ErrorCodeOut
);
```

For more information please contact info@maxeyetech.com

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- CarrierIndex – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- MessageIndex – Specifies the index value to keep a track on the current RDS message under processing.
- RDS Message PTYON and TA – Specifies RDS EON PTYON and TA Message parameters

```
typedef struct {  
    Enum9 EONTPIdentification;  
    Enum9 EONTrafficAnnouncement;  
    Enum17 EONProgramType;  
} RDS_Enhanced_Other_Networks_PTYON_and_TA;
```

- EONTPIdentification – Specifies the Traffic Program Identification as ON or OFF for other programs. Traffic Program identifies if the station is capable of sending traffic announcements or not. This control works with Traffic Announcement control given in EON Message. The default value is 0 (off). Given below are the possible values
 - 0 – Off
 - 1 – OnNote: - Set both the controls Traffic Program Identification and Traffic Announcement control to ON state for sending traffic announcement.
- EONTrafficAnnouncement – Specifies whether the Traffic Announcement is ON or OFF. Set Traffic Announcement property ON to send traffic announcement. Traffic Program Identification too must be ON. The default value is 0 (OFF). Given below are the possible values
 - 0 – OFF
 - 1 – ON
- EONProgramType – Specifies the type of programme transmitted. Configure this control only if selected Data Service Mode is RDS. The default value is 1 (News). Refer the table given in the MaxEye AM FM Remote Set RDS Configuration API for the possible values.
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.

For more information please contact info@maxeyetech.com

- **ErrorCodeOut** – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- **Header** – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- **Library** – AM FM RDS TMC Generation.lib
- **DLL** – AM FM RDS TMC Generation.dll

3.23 MaxEye_AMFM_Remote Set RDS EON PINON

NAME MaxEye_AMFM_Remote_Set_RDS_EON_PINON

DESCRIPTION Configures the RDS EON PINON Settings to the Remote AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```

    (
        LVRefNum          *ConnectionIDIn,
        int32_t           CarrierIndex,
        int32_t           MessageIndex,
        RDS_Enhanced_Other_Networks_PINON *Enchanced_Other_Network_PINON,
        int32_t           ErrorCodeIn,
        LVRefNum          *ConnectionIDOut,
        int32_t           *ErrorCodeOut
    );

```

INPUT PARAMETERS

- **ConnectionIDIn** – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- **CarrierIndex** – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- **MessageIndex** – Specifies the index value to keep a track on the current RDS message under processing.
- **RDSMessage PINON** – Specifies RDS EON PINON Message parameters

```

typedef struct {
    int8_t DayON;
    uint8_t HourON;
    uint8_t MinuteON;
    uint8_t ProgramItemNumberCode;
    Enum16 ProgTypeRDSOIN;
} RDS_Enhanced_Other_Networks_PINON;

```

- **DayON** – Specify the Day of the broadcast for other programs.
- **HourON** – Specifies the Specify the Hour of the broadcast for other programs.
- **MinuteON** – Specify the Minute of the broadcast for other programs.

For more information please contact info@maxeyetech.com

- **ErrorCodeIn** – Specifies the error code. The **ErrorCodeIn** can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- **ConnectionIDOut** – Returns the TCP connection reference. **Connection ID Out** is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- **ErrorCodeOut** – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- **Header** – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- **Library** – AM FM RDS TMC Generation.lib
- **DLL** – AM FM RDS TMC Generation.dll

3.24 MaxEye AM FM Remote Set Carrier Configuration

NAME MaxEye_AMFM_Remote_Set_Carrier_Configuration

DESCRIPTION Configures the Carrier Frequency and Signal Bandwidth for each carrier based on carrier index value to the AM FM RDS Remote SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_Carrier_Configuration
(
    LVRefNum      *ConnectionIDIn,
    int32_t       CarrierIndex,
    double         CarrierFrequency,
    double         RelativePowerDB,
    int32_t       ErrorCodeIn,
    LVRefNum      *ConnectionIDOut,
    int32_t       *ErrorCodeOut
)
```

INPUT PARAMETERS

- **ConnectionIDIn** – Specifies the TCP connection reference. **Connection ID In** is a network connection reference that uniquely identifies the TCP connection.
- **CarrierIndex** – Specifies the index value of the selected carrier. The default value of the **Carrier Index** is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- **CarrierFrequency** – Specifies the **Carrier Frequency** for the selected carrier in Hz.
- **Relative Power** – Specifies the **Relative Power** for the selected Carrier in dB.

For more information please contact info@maxeyetech.com

- **ErrorCodeIn** – Specifies the error code. The **ErrorCodeIn** can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- **ErrorCodeOut** – Returns the error code, which describes the error information. Use this value to refer to error code in subsequent function calls.
- **ConnectionIDOut** – Connection ID Out is a network connection refnum that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.25 MaxEye AM FM Remote Set Output Waveform File Path

NAME MaxEye_AMFM_Remote_Set_Output_Waveform_File_Path

DESCRIPTION Configures the path to save the generated waveform to the AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_Output_Waveform_File_Path
(
    LVRefNum    *ConnectionIDIn,
    char        WavfeormFilePath[],
    int32_t     ErrorCodeIn,
    LVRefNum    *ConnectionIDOut,
    int32_t     *ErrorCodeOut
)
```

INPUT PARAMETERS

- **ConnectionIDIn** – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- **WavfeormFilePath** – Specifies the path of the file to save the generated waveform.
- **ErrorCodeIn** – Specifies the error code. The **ErrorCodeIn** can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- **ConnectionIDOut** – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.

For more information please contact info@maxeyetech.com

- **ErrorCodeOut** – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib, Labview.lib
- DLL – AM FM RDS TMC Generation.dll

3.26 MaxEye AM FM Remote Set USRP Hardware Settings

NAME MaxEye_AMFM_Remote_Set_USRP_Hardware_Settings

DESCRIPTION Configures USRP hardware settings to the AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_USRP_Hardware_Settings
(
    LVRefNum    *ConnectionIDIn,
    char        USRPIPAddress[],
    double      USRPCarrierFrequencyHz,
    double      GainDB,
    char        ActiveAntenna[],
    int32_t     ErrorCodeIn,
    LVRefNum    *ConnectionIDOut,
    int32_t     *ErrorCodeOut
)
```

INPUT PARAMETERS

- **ConnectionIDIn** – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- **USRPIPAddress** – Specifies the IP address of the NI USRP.
- **USRPCarrierFrequencyHz** – Specifies the Center Frequency of the AM FM RDS signal in Hz.
- **GainDB** – Specifies the aggregate gain, in dB, to be applied to the RF signal.
- **ActiveAntenna** – Specifies the antenna port to be used for this channel.
- **ErrorCodeIn** – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

For more information please contact info@maxeyetech.com

- **ConnectionIDOut** – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- **ErrorCodeOut** – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.27 MaxEye AM FM Remote Set USRP Play Waveform Settings

NAME MaxEye_AMFM_Remote_Set_USRP_Play_Waveform_Settings

DESCRIPTION Configures USRP Play Waveform File Settings to the AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_USRP_Play_Waveform_Settings
(
    LVRefNum      *ConnectionIDIn,
    int32_t       WriteBlockSizeInSamples,
    uint16_t      SampleWidth,
    char          WaveformFilePathDialogIfEmpty[],
    int32_t       ErrorCodeIn,
    LVRefNum      *ConnectionIDOut,
    int32_t       *ErrorCodeOut
)
```

INPUT PARAMETERS

- **ConnectionIDIn** – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- **WriteBlockSizeInSamples** – Specifies the size of the block in samples. The waveform is written in the hardware as blocks.
- **SampleWidth** – Specifies the sample width value. Use the same sample width value used for saving the waveform in the file.
- **WavformFilePathDialogIfEmpty** – Specifies the path of the file to be played in the AM FM RDS TMC Signal generation.
- **ErrorCodeIn** – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

For more information please contact info@maxeyetech.com

- **ConnectionIDOut** – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- **ErrorCodeOut** – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.28 MaxEye AM FM Remote Set Play Waveform File Path

NAME MaxEye_AMFM_Remote_Set_Play_Waveform_File_Path

DESCRIPTION Configures Play Waveform from File Path to the AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_Play_Waveform_File_Path  
(  
    LVRefNum    *ConnectionIDIn,  
    char        WaveformFilePath[],  
    int32_t     ErrorCodeIn,  
    LVRefNum    *ConnectionIDOut,  
    int32_t     *ErrorCodeOut  
)
```

INPUT PARAMETERS

- **ConnectionIDIn** – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- **WaveformFilePath** – Specifies the waveform file path to play the waveform.
- **ErrorCodeIn** – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- **ConnectionIDOut** – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- **ErrorCodeOut** – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

For more information please contact info@maxeyetech.com

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.29 MaxEye AM FM Remote Set Play Waveform Settings

NAME MaxEye_AMFM_Remote_Set_Play_Waveform_Settings

DESCRIPTION Configures Play Waveform from File Settings to the AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_Play_Waveform_Settings
(
    LVRefNum          *ConnectionIDIn,
    Play_Waveform_Settings *PlayWaveformSettings,
    int32_t           ErrorCodeIn,
    LVRefNum          *ConnectionIDOut,
    int32_t           *ErrorCodeOut
)
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- PlayWaveformSettings – Specifies Play Waveform Settings parameters

```
typedef struct
{
    double          CenterFrequencyHz;
    int32_t         WriteBlockSizeInSamples;
    int32_t         StreamingWaveformSizeInSamples;
    uint16_t        SampleWidth;
} Play_Waveform_Settings
```

- CenterFrequencyHz – Specifies the center frequency of the AM FM RDS signal in Hz.
- WriteBlockSizeInSamples – Specifies the size of the block in samples. The waveform is written in the hardware as blocks.
- StreamingWaveformSizeSamples – Specifies the total memory allocated in the hardware for streaming the waveform in samples.
- SampleWidth – Specifies the sample width value. Use the same sample width value used for saving the waveform in the file.
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

For more information please contact info@maxeyetech.com

- **ConnectionIDOut** – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- **ErrorCodeOut** – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- **Header** – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- **Library** – AM FM RDS TMC Generation.lib
- **DLL** – AM FM RDS TMC Generation.dll

3.30 MaxEye AM FM Remote Set RDS Impairments

NAME MaxEye_AMFM_Remote_Set__RDS_Impairments

DESCRIPTION Configures the Impairment properties for each carrier to the AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Set_RDS_Impairments
(
    LVRefNum          *ConnectionIDIn,
    int32_t           CarrierIndex,
    AMFM_Impairments *Impairments,
    int32_t           ErrorCodeIn,
    LVRefNum          *ConnectionIDOut,
    int32_t           *ErrorCodeOut
)
```

INPUT PARAMETERS

- **ConnectionIDIn** – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- **CarrierIndex** – Specifies the index value of the selected carrier. The default value of the Carrier Index is 0 which corresponds to the first carrier. For generating multi carrier signal, configure the parameters for each carrier index.
- **Impairments** – Specifies the Impairment Configuration parameters

```
typedef struct
{
    uint16_t      ImpairmentsEnabled;
    uint16_t      AWGEnabled;
    double        CarrierToNoiseRatioDB;
    double        FrequencyOffsetHz;
    double        ClockOffsetPPM;
    AMFM_IQ_Impairments IQImpairments;
}AMFM_Impairments
```

- **IQImpairments** – Specifies IQ Impairment Configuration parameters

For more information please contact info@maxeyetech.com


```
typedef struct
{
    double IDCOffset;
    double QDCOffset;
    double IQGainImbalanceDB;
    double QuadratureSkewDeg;
}AMFM_IQ_Impairments
```

- IDCOffset – Specifies the In-phase DC offset value. The toolkit adds the DC offset to the in-phase signal component (I) of the complex waveform as a percentage of the root mean square magnitude of the unaltered I signal. The default value is 0.
- QDCOffset – Specifies the Quadrature DC offset value. The toolkit adds the DC offset to the quadrature-phase signal component (Q) of the complex waveform as a percentage of the root mean square magnitude of the unaltered Q signal. The default value is 0.
- IQGainImbalanceDB – Specifies the ratio, in dB, of the mean amplitude of the in-phase (I) signal to the mean amplitude of the quadrature-phase (Q) signal. The default value is 0.
- QuadratureSkewDeg – Specifies the deviation in angle from 90 degrees between the in-phase (I) and quadrature-phase (Q) signals. The default value for the Quadrature Skew is 0.
- ImpairmentsEnabled – Specifies whether the impairment addition is enabled or not. If this property is set to True then the toolkit adds the impairments to the generated signal as per the user configuration for the supported impairments. The default value is 0 (False). Given below are the possible values
 - 0 – False
 - 1 – True
- CarrierToNoiseRatio – Specifies the Carrier to Noise ratio of the generated signal. The default value is 0 dB.
- FrequencyOffsetHz – Specifies the frequency offset in Hz. The toolkit applies frequency offset to the created waveform based on the value configured in this property. The applied frequency offset is relative to the signal generator's carrier frequency. The default value is 0.
- ClockOffsetPPM – Specifies the clock offset in parts per million (ppm). The toolkit applies the clock offset to the generated waveform based on this value. The applied clock offset is relative to the clock frequency of the signal generator. The default value is 0.
- AWGNEnabled – Specifies if the AWGN noise addition is enabled or not. If this property is set to True then the toolkit adds Additive White Gaussian Noise (AWGN) to the created waveform based on the value configured in the Carrier to Noise Ratio property. The default value is 0 (False). Given below are the possible values
 - 0 – False
 - 1 – True
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- **ConnectionIDOut** – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- **ErrorCodeOut** – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- **Header** – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- **Library** – AM FM RDS TMC Generation.lib
- **DLL** – AM FM RDS TMC Generation.dll

3.31 MaxEye_AMFM_Remote_Save_Configuration

NAME	MaxEye_AMFM_Remote_Save_Configuration
DESCRIPTION	Configures the file path to save the configurations in file.
FUNCTION PROTOTYPE	<pre>void __cdecl MaxEye_AMFM_Remote_Save_Configuration (LVRefNum *ConnectionIDIn, char SaveConfigurationFilePath[], int32_t ErrorCodeIn, LVRefNum *ConnectionIDOut, int32_t *ErrorCodeOut)</pre>

INPUT PARAMETERS

- **ConnectionIDIn** – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- **SaveConfigurationFilePath[]** – Specifies the file path to save the configurations in file.
- **ErrorCodeIn** – Specifies the error code. The ErrorCodeIn can accept error information from previously called C function. Use this information to decide if any functionality should be bypassed in the event of errors from other C APIs.

OUTPUT PARAMETERS

- **ConnectionIDOut** – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- **ErrorCodeOut** – Returns the error code, passes error or warning information out of a Function to be used by other C APIs.

DEPENDENCIES

For more information please contact info@maxeyetech.com

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.32 MaxEye_AMFM_Remote_Load_Configuration

NAME MaxEye_AMFM_Remote_Load_Configuration

DESCRIPTION Configures the file path to load the saved configurations from file.

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Load_Configuration
(
    LVRefNum      *ConnectionIDIn,
    char          LoadConfigurationFilePath[],
    int32_t       ErrorCodeIn,
    LVRefNum      *ConnectionIDOut,
    int32_t       *ErrorCodeOut
)
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- LoadConfigurationFilePath[] – Specifies the file path to load the saved configurations from file.
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C function. Use this information to decide if any functionality should be bypassed in the event of errors from other C APIs.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of a Function to be used by other C APIs.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.33 MaxEye_AMFM_Remote_Command

NAME MaxEye_AMFM_Remote_Command

For more information please contact info@maxeyetech.com

DESCRIPTION Configures the Generation Mode Type to the AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_Command
(
    LVRefNum      *ConnectionIDIn,
    AMFM_Command Command,
    int32_t       ErrorCodeIn,
    LVRefNum      *ConnectionIDOut,
    int32_t       *ErrorCodeOut
)
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- Command – Specifies the generator status using the below values The default value is 0.
 - 0 – Start Generation
 - 1 – Stop Generation
 - 2 – Save Configuration
 - 3 – Load Configuration
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.
- ErrorCodeOut – Returns the error code, passes error or warning information out of an API to be used by other C API functions.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.34 MaxEye AM FM Remote Get Error Status

NAME MaxEye_AMFM_Remote_Get_Error_Status

DESCRIPTION Receives Error Message from the Remote SFP Application through TCP Network Connection

FUNCTION PROTOTYPE `void __cdecl MaxEye_AMFM_Remote_Get_Error_Status`
 For more information please contact info@maxeyetech.com

```

    (
        LVRefNum          *ConnectionIDIn,
        int32_t           ErrorCodeIn,
        LVRefNum          *ConnectionIDOut,
        AMFM_Get_Generation_Parameters *GetGenerationParameters,
        AMFM_OutputIndicator *OutputIndicator
        int32_t           LengthofErrorStatus,
        char              ErrorStatus[],
        int32_t           *ErrorCodeOut
    )
  
```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- ConnectionIDOut – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent function calls.

```

typedef struct {
    int32_t GenerationMode;
    double GeneratingFrames;
    double MaximumNumberOfFrames;
    LVBoolean GenerationStatus;
    int32_t ErrorOrWarning;
    LVBoolean GenerationCompleted;
} AMFM_Get_Generation_Parameters;
  
```

- GenerationMode – Returns the Generation Mode.
- GenerationFrames – Returns the current frame number being generated to the user.
- MaximumNumberofFrames – Returns the Maximum Number of Frames configured.
- GeneratingStatus – Returns the current status of the generation.
- ErrorOrWarning – Returns the Current Error Status is Warning or Error.
- GenerationCompleted – Returns the status of Current frame is equal to the Maximum Number of Frames Configured.
- LengthofErrorStatus – Specifies the length of ErrorStatus.

```

typedef struct {
    double CenterFrequencyHz;
    double WaveformSamplingRateHz;
    double PAPR;
  
```

For more information please contact info@maxeyetech.com

```

    double PlayDurationSeconds;
} AMFM_OutputIndicator;

```

- CentreFrequencyHz – Indicates the center frequency of the multiple carrier waveform. The same frequency must be used when using Play Waveform from File.
- WaveformSamplingRateHz – Indicates the sampling frequency of the generated IQ baseband waveform. Same sampling rate/IQ rate must be used when using Play Waveform from File.
- PAPR – Indicates Peak to Average Power Ratio, which is calculated by dividing the peak power by the Root Mean Square (RMS) value of the waveform. This value is used to set the Headroom (dB) value.
- ErrorStatus – Returns the description of the error occurred.
- ErrorCodeOut – Returns the code of the error occurred, to the user.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.35 MaxEye_AMFM_SG_Remote_Get_OutPut_Indicators

NAME MaxEye_AMFM_SG_Remote_Get_OutPut_Indicators

DESCRIPTION Returns the Output indicators from the AM FM RDS SFP Application through TCP Network Connection

FUNCTION PROTOTYPE `void __cdecl MaxEye_AMFM_SG_Remote_Get_OutPut_Indicators`

```

(
    LVRefNum          *ConnectionIDIn,
    int32_t           ErrorCodeIn
    LVRefNum          *ConnectionIDOut,
    Int32_t           *GetOutPut
    AMFM_OutputIndicator *OutputIndicator,
    int32_t           *ErrorCodeOut
)

```

INPUT PARAMETERS

- ConnectionIDIn – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.
- ErrorCodeIn – Specifies the error code. The ErrorCodeIn can accept error information from previously called C API. Use this information to decide if any functionality should be bypassed in the event of errors from other C APIs.

OUTPUT PARAMETERS

For more information please contact info@maxeyetech.com

- **ConnectionIDOut** – Returns the TCP connection reference. Connection ID Out is a network connection reference that uniquely identifies the TCP connection. Use this value to refer to this connection in subsequent API calls.
- **OutputIndicator** – Returns the values of Center Frequency, Output Sampling Rate, Play duration, PAPR and Generated frame from the AM FM RDS TMC SFP Client.

```
typedef struct
{
  double CenterFrequencyHz;
  double OutputSamplingRateHz;
  double PAPR;
  double GeneratedFrames;
} AMFM_OutputIndicator;
```

- **CenterFrequencyHz** – Returns the Center Frequency Value of Generated waveform in Hz
- **OutputSamplingRateHz** – Returns the Sampling Rate of Generated Waveform in Hz
- **PAPR** – Returns the PAPR value of generated waveform in Seconds.
- **GeneratedFrames** – Returns the No of Frames generated.
- **GetOutput-** Check Whether the Waveform Output Indicators received the values from SFP.
- **ErrorCodeOut** – Returns the error code, passes error or warning information out of a Function to be used by other C APIs.

DEPENDENCIES

- Header – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- Library – AM FM RDS TMC Generation.lib
- DLL – AM FM RDS TMC Generation.dll

3.36 MaxEye AM FM Remote TCP Close Connection

NAME MaxEye_AMFM_Remote_TCP_Close_Connection

DESCRIPTION Closes TCP network connection between AM FM RDS SFP Client and Server applications

FUNCTION PROTOTYPE

```
void __cdecl MaxEye_AMFM_Remote_TCP_Close_Connection
(
    LVRefNum *ConnectionIDIn,
    int32_t   ErrorCodeIn
    int32_t   *ErrorCodeOut
)
```

INPUT PARAMETERS

- **ConnectionIDIn** – Specifies the TCP connection reference. Connection ID In is a network connection reference that uniquely identifies the TCP connection.

- **ErrorCodeIn** – Specifies the error code. The **ErrorCodeIn** can accept error information from previously called C API function. Use this information to decide if any functionality should be bypassed in the event of errors from other C API functions.

OUTPUT PARAMETERS

- **ErrorCodeOut** – Returns the error code, passes error or warning information out of a Function to be used by other C APIs.

DEPENDENCIES

- **Header** – AM FM RDS TMC Generation.h, extcode.h, fundtype.h, platdefines.h
- **Library** – AM FM RDS TMC Generation.lib
- **DLL** – AM FM RDS TMC Generation.dll