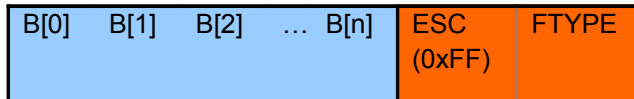


# CodeSkin RA-Link v0.2

## Introduction

CodeSkin has defined a proprietary protocol for transmitting data over a serial communication link (such as RS-232). This protocol is called “RA-Link” and corresponds to the Data Link layer of the OSI model. It provides a simple way of packing n bytes of binary data into a frame, as shown below:



An escape character (ESC = 0xFF) is used by the protocol to signal the end of a frame. The character following ESC identifies the type of the frame (FTYPE).

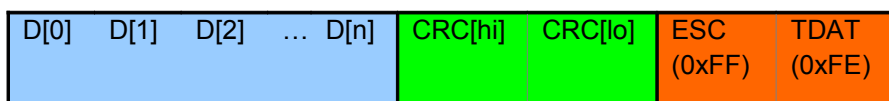
<i><b>FTYPE</b></i>	<i><b>Name</b></i>	<i><b>Description</b></i>	<i><b>Mnemonic</b></i>
0x00 – 0xF0	Reserved	Deprecated types	--
0xF1	Error-Frame	Used to signal error conditions as well as the reception of invalid frames	TERR
0xF2	Length-Frame	Frame defining length of subsequent data frame	TLEN
0xF3 – 0xFB	Reserved	Reserved for future use	--
0xFC – 0xFD	Reserved	Deprecated types	--
0xFE	Data-Frame	Frame containing data	TDAT

The bytes leading up to the escape sequence (B[0] – B[n]) contain the frame data. Values between 0x00 and 0xFF are allowed. However, due to the special meaning of 0xFF (ESC), double-escaping is necessary for packing the value 0xFF. In other words, a datum of B[k]=0xFF has to be transmitted as two subsequent 0xFFs.

The maximal allowable frame length, including all ESC characters and FTYPE, is 128 bytes.

## Data Frame

Application data (or commands) are transmitted by means of data-frames. They are of variable length, depending on the size of the data to be transmitted.



A data-frame consists of three groups:

1. n-bytes of binary data – if a byte equals 0xFF, then it is encoded as {0xFF,0xFF}
2. 16 bit CRC (cyclic redundancy check)
3. End-of-frame escape sequence {0xFF, 0xFE}

The CRC uses the CCITT polynomial ( $x^{16}+x^{12}+x^5+1$ ), with an initial value of 0xFFFF, and is calculated over the binary data (D[0]-D[n]), including eventual ESC characters. If a CRC byte equals 0xFF then it has to be escaped. The data-frame escape sequence ({0xFF, 0xFE) is not included in the CRC calculation.

The maximal allowable data length (n), not including any ESC characters, CRC or TDAT, is 60 bytes.

### Example 1:

Data to be transmitted: {0x01, 0x02, 0x03, 0x04, 0x05}

Encoded frame: {0x01, 0x02, 0x03, 0x04, 0x05, 0x93, 0x04, 0xFF, 0xFE}

### Example 2 (demonstrates escaping of 0xFF):

Data to be transmitted: {0xFF, 0xFE, 0xFD, 0xFC, 0xFB}

Encoded frame: {0xFF, 0xFE, 0xFE, 0xFD, 0xFC, 0xFB, 0xDF, 0x6F, 0xFF, 0xFE}

## Length-Frame

The length-frame indicates the data length of a subsequent data-frame. It has a fixed length of 3 bytes.

DLEN	ESC (0xFF)	TLEN (0xF2)
------	---------------	----------------

Its purpose is to allow the communication link to implement timeouts and conduct additional error checking. While its use is recommended, it is not mandatory. Therefore, a receiver shall accept a data-frame even without a preceding length-frame. However, if a length-frame has been received, then the receiver shall verify the length of the following data-frame and issue an error-frame if there is a mismatch.

The value of DLEN corresponds to the data length (n), not including any ESC characters, CRC or TDAT. Only values between 1 and 60 are legal for DLEN.

## Error-Frame

Error-frames are used to signal reception problems and provide a means for data stream synchronization. They have a fixed length of 3 bytes.

ERROR	ESC (0xFF)	TERR (0xF1)
-------	---------------	----------------

The following error codes are defined:

<b><i>ERROR</i></b>	<b><i>Description</i></b>
0x00	No error (can be used for synchronization)
0x01	CRC error
0x02	Invalid length
0x03	Value out of range
0x04	Invalid frame type
0x05	Rx overrun
0x06 – 0xFE	Reserved for future use
0xFF	Illegal