ENTTEC Pty/Ltd
ACN 089 252 791
ABN 94 388 357 479
PO Box 282
Kew, Vic, 3101
Australia

Tel +61 39819 2433
Fax +6139819 2733
Web:www.enttec.com

DMX USB Pro Widget API Specification 1.44

Purpose

This document specifies the interface requirements for PC based application programs to use the DMX USB Pro Widget to send or receive DMX512 packets.

PC Setup

Install the VCOM FT245BM device driver on the PC. Since the driver is a virtual COM port, the baudrate setting used to open this COM port is a dummy value, and does not control the USB communication speed.

Application Message Format

The PC based application program communicates with the Widget via the FTDI driver. The table below specifies the general format of the messages between the application program and the FTDI driver.

Size In Bytes	Description
1	Start of message delimiter, hex 7E.
1	Label to identify type of message. See below for value.
1	Data length LSB. Valid range for data length is 0 to 600.
1	Data length MSB.
data_length	Data bytes.
1	End of message delimiter, hex E7.

Firmware Varieties

The Widget firmware version identifies the set of application messages that are supported by the firmware. The Widget will ignore any unsupported application messages. The following table shows the available firmware varieties.

ENTTEC Pty/Ltd
ACN 089 252 791
ABN 94 388 357 479
PO Box 282
Kew, Vic, 3101
Australia

Tel +61 39819 2433
Fax +6139819 2733
Web:www.enttec.com

Firmware Version MSB	Description
1	Normal DMX firmware. Supports all messages except Send RDM (label=7), Send RDM Discovery Request(label=11) and receive RDM .
2	RDM firmware.This enables the Widget to act as an RDM Controller or RDM responder. Supports all messages except Receive DMX On Change (label=8) and Change Of State Receive (label=9).
3	RDM Sniffer firmware. This is for use with the Enttec RDM packet monitoring application.

Widget LED

The Widget firmware will blink the LED at 8Hz rate for one cycle, each time a DMX or RDM packet is received, and will blink the LED at 2Hz rate for one cycle, each time a DMX or RDM packet is transmitted.

Application Messages

1. Reprogram Firmware Request (Label = 1, no data)

This message requests the Widget firmware to run the Widget bootstrap to enable reprogramming of the Widget firmware.

2. Program Flash Page Request (Label=2)

This message programs one Flash page of the Widget firmware. The Flash pages must be programmed in order from first to last Flash page, with the contents of the firmware binary file.

Size In Bytes	Description
64	One page of firmware binary file.

3. Program Flash Page Reply (Label=2)

The Widget sends this message to the PC on completion of the Program Flash Page request.

ENTTEC Pty/Ltd
ACN 089 252 791
ABN 94 388 357 479
PO Box 282
Kew, Vic, 3101
Australia

Tel +61 39819 2433
Fax +6139819 2733
Web:www.enttec.com

Size In Bytes	Description
4	Success character array, set to 'TRUE' if firmware page was programmed successfully, set to 'FALS' if firmware page programming failed.

4. Get Widget Parameters Request (Label=3)

This message requests the Widget configuration.

Size In Bytes	Description
1	LSB of user configuration size in bytes. Valid range for user configuration size is 0 to 508.
1	MSB of user configuration size in bytes.

5. Get Widget Parameters Reply (Label=3)

The Widget sends this message to the PC in response to the Get Widget Parameters request.

Size In Bytes	Description
1	Firmware version LSB. Valid range is 0 to 255.
1	Firmware version MSB. Valid range is 0 to 255.
1	DMX output break time in 10.67 microsecond units. Valid range is 9 to 127.
1	DMX output Mark After Break time in 10.67 microsecond units. Valid range is 1 to 127.
1	DMX output rate in packets per second. Valid range is 1 to 40.
user_configuration_size	User defined configuration data. See Set Widget Parameters request.

ENTTEC Pty/Ltd
ACN 089 252 791
ABN 94 388 357 479
PO Box 282
Kew, Vic, 3101
Australia

Tel +61 39819 2433
Fax +6139819 2733
Web:www.enttec.com

6. Set Widget Parameters Request (Label=4)

This message sets the Widget configuration. The Widget configuration is preserved when the Widget loses power.

Size In Bytes	**Description**
1	LSB of user configuration size in bytes. Valid range for user configuration size is 0 to 508.
1	MSB of user configuration size in bytes.
1	DMX output break time in 10.67 microsecond units. Valid range is 9 to 127.
1	DMX output Mark After Break time in 10.67 microsecond units. Valid range is 1 to 127.
1	DMX output rate in packets per second. Valid range is 1 to 40, or 0 for fastest rate possible (this will make the most difference when the output universe size is smallest).
user_configuration_size	User defined configuration data.

ENTTEC Pty/Ltd
ACN 089 252 791
ABN 94 388 357 479
PO Box 282
Kew, Vic, 3101
Australia

Tel +61 39819 2433
Fax +6139819 2733
Web:www.enttec.com

7. Received DMX Packet (Label=5)

The Widget sends this message to the PC unsolicited, whenever the Widget receives a DMX or RDM packet from the DMX port, and the Receive DMX on Change mode is 'Send always'.

Size In Bytes	Description
1	DMX receive status. When this is 0, the DMX data in this message is valid. When this is nonzero, the DMX data in this message is corrupted. Bit 0: 0=No error,1=Widget receive queue overflowed. Bit 1: 0=No error,1=Widget receive overrun occurred.
1 to 513	Received DMX data beginning with the start code. The size of the received DMX data can be determined from the overall message size.

8. Output Only Send DMX Packet Request (Label=6)

This message requests the Widget to periodically send a DMX packet out of the Widget DMX port at the configured DMX output rate. This message causes the widget to leave the DMX port direction as output after each DMX packet is sent, so no DMX packets will be received as a result of this request.

The periodic DMX packet output will stop and the Widget DMX port direction will change to input when the Widget receives any request message other than the Output Only Send DMX Packet request, or the Get Widget Parameters request.

Size In Bytes	Description
25 to 513	DMX data to send, beginning with the start code. The overall message size specifies the size of the DMX data to send, and also sets the universe size (the number of DMX channels which are output).

ENTTEC Pty/Ltd
ACN 089 252 791
ABN 94 388 357 479
PO Box 282
Kew, Vic, 3101
Australia

Tel +61 39819 2433
Fax +6139819 2733
Web:www.enttec.com

9. Send RDM Packet Request (Label=7)

This message requests the Widget to send an RDM packet out of the Widget DMX port, and then change the DMX port direction to input, so that RDM or DMX packets can be received.

Size In Bytes	Description
1 to 513	RDM data to send, beginning with the start code. The overall message size specifies the size of the RDM data to send.

10.Receive DMX on Change (label = 8)

This message requests the Widget send a DMX packet to the PC only when the DMX values change on the input port.

By default the widget will always send, if you want to send on change it must be enabled by sending this message.

This message also reinitializes the DMX receive processing, so that if change of state reception is selected, the initial received DMX data is cleared to all zeros.

Size In Bytes	Description
1	0: Send always 1: Send on data change only

ENTTEC Pty/Ltd
ACN 089 252 791
ABN 94 388 357 479
PO Box 282
Kew, Vic, 3101
Australia

Tel +61 39819 2433
Fax +6139819 2733
Web:www.enttec.com

11.Received DMX Change Of State Packet (Label=9)

The Widget sends one or more instances of this message to the PC unsolicited, whenever the Widget receives a changed DMX packet from the DMX port, and the Receive DMX on Change mode is 'Send on data change only'.

Size In Bytes	Description
1	Start changed byte number.
5	Changed bit array, where array bit 0 is bit 0 of first byte and array bit 39 is bit 7 of last byte.
1 to 40	Changed DMX data byte array. One byte is present for each set bit in the Changed bit array.

The user program can decode the message into a 513 byte received DMX data array, beginning with the start code. The algorithm to do this is shown below:

```
On startup, zero out the 513 byte received_dmx_array
For each Change Of State packet received
   changed_byte_index = 0
   For bit_array_index = 0 to 39
      If changed_bit_array[bit_array_index] is 1 then
         received_dmx_array[start_changed_byte_number * 8 + bit_array_index] =
            changed_dmx_data_array[changed_byte_index]
         Increment changed_byte_index
      Endif
   Endfor
Endfor
```

12.Get Widget Serial Number Request (Label = 10, no data)

This message requests the Widget serial number, which should be the same as that printed on the Widget case.

13.Get Widget Serial Number Reply (Label = 10)

The Widget sends this message to the PC in response to the Get Widget Serial Number request.

ENTTEC Pty/Ltd
ACN 089 252 791
ABN 94 388 357 479
PO Box 282
Kew, Vic, 3101
Australia

Tel +61 39819 2433
Fax +6139819 2733
Web:www.enttec.com

Size In Bytes	Description
4	BCD serial number, with LSB stored at lowest address. On old Widgets, the serial number was not programmed, and the value would be hex 0FFFFFFFF.

14. Send RDM Discovery Request (Label=11)

This message requests the Widget to send an RDM Discovery Request packet out of the Widget DMX port, and then receive an RDM Discovery Response (see Received DMX Packet).

Size In Bytes	Description
38	DISC_UNIQUE_BRANCH RDM request packet to send.

RDM Implementation Details

Enttec has an RDM protocol PC library that works with the Widget, for creating RDM capable PC applications.

The RDM protocol is a half duplex request/response protocol. To comply with the RDM protocol, the RDM Responder must send only a single response message in reply to each request message from the RDM Controller.

The Widget UID consists of the Enttec manufacturer ID concatenated with the Widget Serial Number, as follows:

UID[0]=hex 45, UID[1]=hex 4E, UID[2]=SN[3], UID[3]=SN[2],
UID[4]=SN[1], UID[5]=SN[0].

There is a special message with its own label for sending the RDM Discovery request, to enable the Widget to receive the discovery response, which has no break unlike all other types of RDM packet. The Send RDM request should be used to send any RDM packet other than the RDM discovery request.

The behaviour of the RDM Responder Widget is specified in the following table.

ENTTEC Pty/Ltd
ACN 089 252 791
ABN 94 388 357 479
PO Box 282
Kew, Vic, 3101
Australia

Tel +61 39819 2433
Fax +6139819 2733
Web:www.enttec.com

Request Received By RDM Responder Widget	**Reply Sent By RDM Responder Widget**	**Request Message Passed To PC**
CC=DISCOVERY_COMMAND PID=DISC_UNIQUE_BRANCH	Discovery response when Widget UID is inside range of UIDs in request message and RDM discovery has not been muted.	No
CC=DISCOVERY_COMMAND PID=DISC_MUTE	DISC_MUTE response when request message was not broadcast.	No
CC=DISCOVERY_COMMAND PID=DISC_UN_MUTE	DISC_UN_MUTE response when request message was not broadcast.	No
CC=GET_COMMAND or CC=SET_COMMAND	ACK_TIMER response with time of 0.1 seconds.	Yes. PC should respond with Queued message.
CC=GET_COMMAND PID=QUEUED_MESSAGE	Queued message (see ACK_TIMER).	No
Any unrecognised message or message with bad RDM checksum.	None	Yes

The RDM Responder Widget has space to store a single queued RDM message only. The ACK_TIMER and queued message mechanism ensure that the timing requirements of the RDM specification can be met, even though the PC application is slow to respond due to USB latency or scheduling delays for example.