

[Forum](#) » [DMXControl Projects e.V.](#) » [Hardwareprojekte / Selbstbau](#) »

Serielles Protokoll zum eurolite USB-DMX512-PRO interface

Wolfgang W.

Mittwoch, 6. Februar 2013, 21:01

Serielles Protokoll zum eurolite USB-DMX512-PRO interface

Hallo zusammen,

ich habe mir das oben genannte Interface gekauft. Nun würde ich gerne mit einer eigenen Anwendung das Interface benützen.

Leider finde ich im Internet keine Doku, wie man das Interface anspricht.

Hat jemand dies schon mal gemacht und/oder hat eine Anleitung, in welcher Form die Anweisung an das Interface gesendet werden.

Vielen dank für Eure Hilfe.

Wolfgang 😊

Markus M.

Donnerstag, 7. Februar 2013, 13:00

Das Interface nutzt einen Befehl des Enttec Pro Interfaces.

Bin gerade nur per Handy online, aber wenn du auf der Seite von Enttec schaust, dann wirst du da die Informationen finden.

Allerdings ist das Protokoll grottenschlecht implementiert.

Ich glaub ich muss das Universe 60 mal pro Sekunde senden, damit ca 20 Hz hinten rauskommen.

Wird nichts mehr gesendet, sendet auch das Interface nichts mehr.

Wolfgang W.

Donnerstag, 7. Februar 2013, 16:07

Hallo Markus,

ich hab mal bei Enttec nachgeschaut. Was ich gefunden habe ist ein TCP/IP Protokol.

Ich möchte mein Problem etwas genauer beschreiben. Ich habe eine LED-Matrix 8x8 (80x80 cm)gebaut (sollen einmal vier werden). Diese lässt sich über DMXControl bereits ansteuern. Leider gibt es (aus meiner Sicht) keine passenden Steuerelemente um Muster (wie Buchstaben, Smily, etc) einfach zu erfassen. Weiter geplant sind Laufschriften und Lauflichter.

Daher würde ich ein eigenes Programm erstellen (Visual C, Visual Basic), welches auf die COMx zugreift und die entsprechenden Channels an das

Eurolite sendet. Wie gesagt finde ich keine Dokumentation wie ich es dem Eurolite USB-DMX512- Pro sage.

Freue mich auf eine Nachricht von Dir.

Viele Grüße

Wolfgang

fisl

Donnerstag, 7. Februar 2013, 16:24

Schicks zurück und kauf was gescheites...

Wenn du dir das wirklich antun willst:

Enttec DMX USB Pro Widget API Specification 1.44

Das Teil kann nur: 8. Output Only Send DMX Packet Request (Label=6), aber statt 25-513 Bytes, müssen immer 513 Bytes gesendet werden.

Dennis

Wolfgang W.

Samstag, 9. Februar 2013, 07:02

Halle Dennis,

wie das immer so ist, wenn man mit etwas beginnt ist man nicht der "Schlauste".

Daher freue ich mich, dass er Leute wie Dich gibt, die schon Experten sind. Und nach einer gewissen Zeit Kann ich dies auch an die nächsten weitergeben. Daher meine Frage:

"Was ist Deiner Meinung nach etwas Gescheites?"

Und hast Du irgendwo einen Link oder Codeschnipsel wo das Interface mit Visual Basic angesprochen wird?

Vielen Dank für Deine Hilfe.

Wolfgang

fisl

Samstag, 9. Februar 2013, 12:56

Zitat von »Wolfgang W.«

"Was ist Deiner Meinung nach etwas Gescheites?"

Da steh ich zum Glück nicht mit meiner Meinung alleine. fx5.de oder Selbstbau digital-enlightenment.de. Nachzulesen in unzähligen Threads hier

im Forum.

Zitat von »Wolfgang W.«

Und hast Du irgendwo einen Link oder Codeschnipsel wo das Interface mit Visual Basic angesprochen wird?

Beispielcode habe ich nicht. Aber wie du der Spezifikation entnehmen kannst, musst du nur die 4 Byte vom Header (Application Message Format), dann die Daten inkl. Startbyte und zum Abschluss noch 0xe7 schicken. Laut Markus 60 mal pro Sekunde. Zum initialisieren eines seriellen Ports wird wohl Beispielcode geben.

Medium

Montag, 14. April 2014, 16:18

Gerät bleibt komplett stumm. Was tun?

Ahoi!

Ich versuche seit einiger Zeit nun schon dieses Gerätchen zum Reden zu bewegen, aber leider komplett ohne auch nur einen Ansatz von Erfolg.

Den Treiber habe ich installiert, und der virtuelle COM Port ist verfügbar und auch ansprechbar. Dann habe ich mir die Doku (DMX USB Pro Widget API Specification 1.44) wie mancherorts empfohlen zur Brust genommen, und zunächst ganz naiv versucht dem Teil ein paar DMX Daten zu schicken (Label 6). Da daraufhin nichts an meinem Dimmerpack passierte, habe ich mal versucht die Config des Interfaces zu lesen. Laut Doku soll dabei ein Label 3 geschickt werden, sowie die Länge der UserConfig (was auch immer diese enthält, das ist nirgends dokumentiert). Diese habe ich daher auch bei 0 belassen.

Mein Delphi-Code zum senden der Config-Anfrage sieht so aus:



Quellcode

```
1 procedure TForm1.Button1Click(Sender: TObject);
2 var
3     FWidgetConfigRequest: array[0..6] of Byte;
4 begin
5     FWidgetConfigRequest[0] := $7E; // Start
6     FWidgetConfigRequest[1] := $03; // Label=3: Config Request Message
7     FWidgetConfigRequest[2] := $02; // Data Length LSB
8     FWidgetConfigRequest[3] := $00; // Data Length MSB
9     FWidgetConfigRequest[4] := $00; // Data: UserConfig Length LSB
10    FWidgetConfigRequest[5] := $00; // Data: UserConfig Length MSB
11    FWidgetConfigRequest[6] := $E7; // Stop
12    TAPDComPort1.PutBlock(FWidgetConfigRequest, Length(FWidgetConfigRequest));
13 end;
```

Für die COM Kommunikation setze ich die TurboPower Async Pro Komponenten ein, die wir auch schon zigfach problemlos für die Steuerung von Industriemaschinen im Einsatz haben. Der DataAvail-Trigger, der immer

dann feuert wenn am COM Port neue Daten anstehen, bleibt aber leider nach Senden obiger Message ruhig - das Interface sendet nichts zurück.

Da ich die DMX Funktionen innerhalb eines bestehenden Programmes von uns benötige, sind Freestyler oder DMXControl leider auch keine Option (zumal mir DMXControl mit "Das Programm funktioniert nicht mehr" abschmiert sobald ich das Häkchen beim Enttec-Output-Plugin setze). Da hier aber, wenn ich nicht irre, auch der Autor besagtem Plugins schreibt, hoffe ich, dass mir bzgl. des Protokolls ein wenig auf's Pferd geholfen werden könnte. Ihr habt es ja irgendwie geschafft, sonst wäre es ja nicht im Programm 😊

Ich wäre für jegliche Tipps seeeeeehhrrrr dankbar. Es gibt einfach nichts grausigeres als Hardware die komplett ohne auch nur den Ansatz einer Doku verkauft wird. (Leider soll es genau das Teil sein, es wird von meinem Kunden beige stellt.)

Vielen Dank im Voraus und viele Grüße!

Markus M.

Montag, 14. April 2014, 19:09

Also ich kann nur von dem Interface abraten.
Das Interface unterstützt nur Label 6 des Enttec Protokolls.
Außerdem muss die Message ca 40 mal pro Sekunde gesendet werden, damit das Interface eine DMXAusgabe erzeugt.

Medium

Montag, 14. April 2014, 21:59

Ah!

Ja, das wurde mir beim Lesen hier auch irgendwann deutlich. Also dass es nicht sehr empfehlenswert ist. Leider steht von solchen Details nichts in den Bewertungen bei Thomann, und soooo tief recherchiert man bei einem 100€ Artikel dann auch im Vorfeld wieder nicht. Da ist die Arbeitszeit zu teuer für. Zumal ich als Entwickler im Industriebereich selten mit der Materie zu tun habe, und nichtmals weiss auf was man bei welchen Vorhaben wo und wie genau beachten sollte. Ich hatte nichtmals die Fragestellung vorher. (Das ist nachher bestimmt diese ominöse "Erfahrung" von der immer alle reden 😊)

Das Teil mit Messages vollballern ist kein Problem. Wenn es das so will, kann es das gerne haben. Komisches Verhalten dennoch. Der Hinweis, dass nur Label 6 damit geht hat mir allerdings jetzt sicherlich noch 1-2 Tage Kopferbrechen erspart, danke!!

Dann versuche ich das morgen nochmals. Nur zur Sicherheit: Angenommen ich will 12 Kanäle ab Kanal 1 setzen, so wäre die Message (Hex):
7E 06 0D 00 00 Ch1 Ch2 Ch3 Ch4 Ch6 Ch6 Ch7 Ch8 Ch9 Ch10 Ch11 Ch12
E7

Wobei die 00 vor den Kanaldaten das DMX-Start Signal ist.
Ch[n] sind die Byte-Werte für den jeweiligen Kanal. (Also nicht in Hex, sondern Platzhalter hier.) Habe ich das so richtig verstanden?

Medium

Dienstag, 15. April 2014, 11:35

Gelöst!

Es funktioniert!

Dank der Infos von hier und einem sehr freundlichen Mitarbeiter von Eurolite kann ich nun meine Dimmerpacks mit dem Adapter steuern. Die gemachten Erkenntnisse möchte ich natürlich nicht für mich behalten:

1) Wie Markus M. schrieb, wird höchst wahrscheinlich wirklich nur Label 6 (DMX Out) vom Protokoll unterstützt

2) Das Enttec Protokoll sieht vor, auch weniger als 512 Kanäle zu übertragen, so lange man die Längenangaben anpasst. Dies versteht der Adapter von Eurolite NICHT. Hier müssen immer alle 512 Kanäle geschickt werden, mitsamt korrekter Längenangabe.

3) Ich habe problemlos Ergebnisse erzielt, auch wenn ich nur jede volle Sekunde eine Message abgesetzt habe. Für eine flüssige Abfolge habe ich das Sendeintervall auf 30ms gestellt, was ebenfalls gut funktionierte. (Der COM-Port sollte dafür allerdings mit 115200 Baud geöffnet werden.)

Die einzige von dem Gerät verstandene Message ist daher in Hex: 7E 06 01 02 00 [512 * Channel Data] E7

Vielen Dank nochmals (auch an alle Poster der vergangenen Jahre zu dem Thema) und viele Grüße!