

Soundfile auf das Soundmodul JQ6500 übertragen

Frank

JQ6500 mit wav-Dateien laden - 2MB Limit beachten - JQ6500 muß erkannt werden, wenn nicht → https://www.youtube.com/watch?v=9h_RWsfRnY4

1. JQ6500 wird erkannt als
2. Es enthält:
3. MusicDownload.exe starten
4. Zweite Tab auswählen und Button drücken:
5. Gewünschte wav-Dateien auswählen
6. Die Dateien werden in der Liste angezeigt:

Hinweis: Es empfiehlt sich diese Liste zu dokumentieren, damit man die Reihenfolge der Dateien weiß!

7. Im ersten Tab den Button auswählen: 8. Wenn alles funktioniert hat steht da:

Martin

https://github.com/Hardi-St/MobaLedLib_Docu/blob/master/Erfahrungsberichte/Bericht%20DOMAPI_2.pdf

Mit etwas Pech muss man das JQ6500 zuerst einmal initialisieren, da es sonst nicht vom PC erkannt wird. Siehe auch MobaLedLib-Thread und diverse Internet-Seiten (Video des Italieners und VMWare). JQ6500 via USB an PC anschließen. Wenn man Glück hat, wird das Modul als neues CD-Laufwerk erkannt und kann mit dem Explorer angezeigt werden ①. Nach Start der MusicDownload.exe erscheint ein spartanisches chinesisches Menü ②. Nach Drücken der rechten „????“ können die Sound-Files vom PC ausgewählt werden ③. Das Modul hat normalerweise 2 MB Speicher. Überschreitet man diesen mit den ausgewählten Dateien, kann der Upload nicht durchgeführt werden. Es erscheint in chinesische Fehlermeldung. Zurück auf dem linken Reiter „????“ den Knopf in der Mitte drücken, um den Upload-Vorgang zu starten. Es erscheint unten ein grüner Balken. In der Mitte links steht sogar das Wort „flash“ ④. Wenn der Upload fertig ist, ändert sich der Text und der grüne Balken ist komplett gefüllt ⑤. Dann sollten alle Files auf dem JQ6500 gespeichert sein. Dann die App schließen und das USB-Kabel entfernen. Welche Dateien aufgespielt wurden, lässt sich hinterher nicht mehr feststellen. Daher am besten die Dinger durchnummerieren (z.B. mit permanent Stift auf dem USB-Anschluss) und die Namen der Sounds irgendwo dokumentieren.

2

Für einen ersten Test den Arduino-sketch „24. DCC und Sound“ aufspielen. Dazu JQ6500-Modul auswählen (#define-Befehl am Anfang des Programmes auskommentieren). Achtung: Das Programm funktioniert in der Originalversion nur, wenn die RGB-LED (Heartbeat) auf der Hauptplatine durch einen R1 = 100 Ohm Widerstand ersetzt wird. Ich habe das Programm angepasst, da auf der Hauptplatine LED0 eine RGB_Heartbeat LED eingesteckt ist (LED0). Der Sound liegt an LED1, siehe die geänderten Soundanweisungen im Sketch („Sound_JQ6500_Seq1(1, 0) “ etc.). Die 5 Kanäle + weitere Funktionen können über den seriellen Arduino-Monitor angesteuert und ggf. kalibriert werden. Das Arduino-Programm sendet PWM-Impulse an das JQ6500 und simuliert damit den Spannungsteiler am ADKEY-Eingang. Zur Aktivierung des Test-Modus gibt man in der „Senden“ Zeile das „seriellen Monitors“ das Paragraph „§“ Zeichen ein. Mit „L1“ die erste LED wählen. Da wir bei 0 mit der Zählung starten (LED0 = Heartbeat-RGB-LED), ist die LED1 dann das Soundmodul, das direkt an der Hauptplatine angeschlossen ist! Über „P“ + Zahl kann das Tastverhältnis des PWM-Signal eingestellt werden. „P29“ sollte den ersten Sound abrufen. Vordefiniert sind folgende Schwellwerte und ihre Zuordnung zu den JQ6500-Funktionen:

```
// ADKey1 400ms 470 Ohm
```

```
#define SOUND_JQ6500_ADKEY1 29 1 #define SOUND_JQ6500_ADKEY2 40 2 #define  
SOUND_JQ6500_ADKEY3 50 3 #define SOUND_JQ6500_ADKEY4 60 4 #define SOUND_JQ6500_ADKEY5  
80 5 #define SOUND_JQ6500_ADKEY6 162 Prev/Vol- #define SOUND_JQ6500_ADKEY7 231 Next/Vol+
```

Diese Tabelle steckt in der Datei MobaLedLib.h. Die Standardwerte passen für mein Modul. Man kann die Soundtest auch direkt im Program Generator über den LED-Farbtest durchführen. Auch lassen sich die Schwellwerte im Excel definieren (Expertenmodus). Eine Anpassung ist nur bei exotischen WS2811 Modulen notwendig! Folgende Kommandos im seriellen Monitor sind verwendbar, um die LEDs anzusteuern: L10 ⇒ LED Nr L+ ⇒ Next LED L- ⇒ Previous LED ENTER ⇒ Repeat last input (Useful with +, -, T or impulse) X ⇒ Exit test mode h45 ⇒ Hue = 45 v100 ⇒ Value = 100 H20 ⇒ Hue = 20 H+4 ⇒ Hue+=4 H- ⇒ Hue- H+ ⇒ Hue++ R30 ⇒ Red LED value G+ ⇒ Increment Green value B-3 ⇒ Decrement green value by 3 P50 ⇒ Impulse with a value of 50 (Sound module tests) i100 ⇒ Impulse length [2ms] i100 = 200 ms M2 ⇒ Impulse mask 1 = Red, 2 = Green, 4 = Blue, 7 = white A ⇒ All LEDs on (255) C ⇒ All LEDs off e3 ⇒ Enable input 3 d3 ⇒ Disable input 3 t3 ⇒ Toggle input 3 Das Testprogramm eignet sich auch gut für die Einstellung von LED-Farben. ===== 3 ===== Wenn man ein Soundfile endlos nacheinander abspielen möchte, muss man 3 Zeilen im Excel eingeben. Die erste Zeile aktiviert das Soundfile erstmalig bzw. einmalig beim Senden des DCC-Befehls (hier Adresse „308“). Macht man das nicht, würde der Sound erst nach Ablauf der Mindestzeit starten. Die zweite Zeile aktiviert die Random-Funktion. Diese setzt hier alle 20,3 Sekunden eine Variable „Sound-Loop“ für 1 Sekunde. Die beiden Wertepaare Min/Max sind identisch. Die Variable wiederum triggert dann in der 3. Zeile das erneute Abspielen des Soundfiles. Zwischen 2 Sound-Durchgängen ist eine kurze Pause für ein paar ms, was im Anlagenbetrieb nicht auffallen dürfte.

From:
<https://wiki.mobaledlib.de/> - MobaLedLib Wiki

Permanent link:
https://wiki.mobaledlib.de/anleitungen/anwendungen/verwendung_jq6500?rev=1610301592

Last update: 2021/01/10 18:59



