

AN013: Senden von DMX mit AVR

Inhalt

- Einleitung
- Nutzungsbedingungen
- DMX 512
- Beschreibung des Codes

Einleitung

In dieser Application Note wird die Erzeugung eines DMX-Signals mit Hilfe von AVR-Mikrocontrollern beschrieben. Die state machine wurde in C für den DMX-Transceiver von Henne's Sites geschrieben, sollte sich aber auch auf andere Schaltungen portieren lassen.

Nutzungsbedingungen

Die state machines können gemäß der gnu general public license (GPL) genutzt werden.

Falls eine GPL-konforme Nutzung nicht möglich ist, wenden Sie sich bitte an den Urheber.

DMX 512

DMX 512 ist ein unidirektionales differentielles serielles Protokoll mit einer Übertragungsrate von 250kbaud. Ein Byte besteht aus einem Startbit [4], acht Datenbits [14, 15] und zwei Stoppbits [7, 8]. Es gibt kein Paritätsbit (8n2). Zwischen den Bytes können inter byte gaps [9] eingeschoben werden, um auch langsameren Empfängern einen stabilen Empfang zu ermöglichen.

Es sind ein Master und bis zu 32 Slaves in einem Bus vorgesehen. Die Zahl der Slaves kann durch Einsatz von Splittern oder Boostern erhöht werden. Ein Universum kann bis zu 512 Kanäle haben. Die Übertragung wird durch einen Break [1] von mind. 88µs Länge gefolgt von einem Mark-after-Break [2] von mind. 8µs Länge eingeleitet. Anschließend wird das Startbyte [14] übertragen, das bei DMX den Wert Null hat.

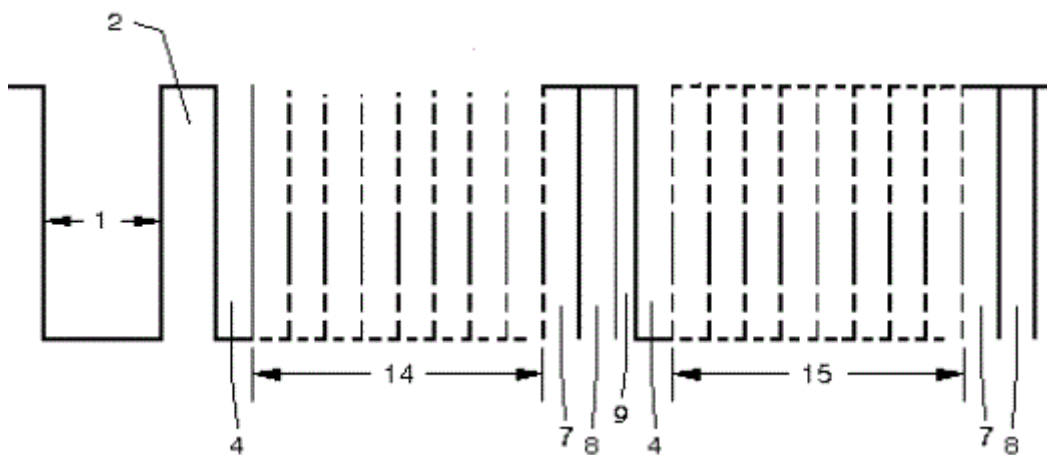


Abbildung 1: Aufbau des DMX-Protokolls

Beschreibung des Codes

Der Code besteht neben der Initialisierung aus einer interrupt service routine:

In der „USART Transmission Complete“-ISR wird zunächst geprüft, ob ein Break gesendet werden soll. Falls ja, wird mit einer sehr viel kleineren Baudrate ein Nullbyte gesendet, das die Empfänger als einen Break mit über 100µs und einen Mark After Break von über 8µs interpretieren.

Um nach dem Break das Startbyte senden zu können, wird die Baudrate wieder auf 250kBit/s gesetzt und erneut ein Nullbyte gesendet. Abschließend wird der Kanalzähler zurückgesetzt.

Nach Senden des Startbytes werden die einzelnen Kanäle übertragen. Ist das Universum komplett gesendet, wird wieder mit einem Break begonnen.

Eine Besonderheit ist die variable Interbyte-Gap: Je größer sie ist, desto geringer ist die DMX-Refreshrate und desto länger blockiert die ISR den AVR. Allerdings benötigen manche Empfänger diese Zeit (der Transceiver nicht!), um das gerade empfangende Byte zu verarbeiten. Die Gap sollte deshalb so klein wie nötig gewählt werden. Zudem sollten mindestens 50 Kanäle gesendet werden, um die Empfänger nicht mit einer zu hohen Refreshrate zu überfordern.

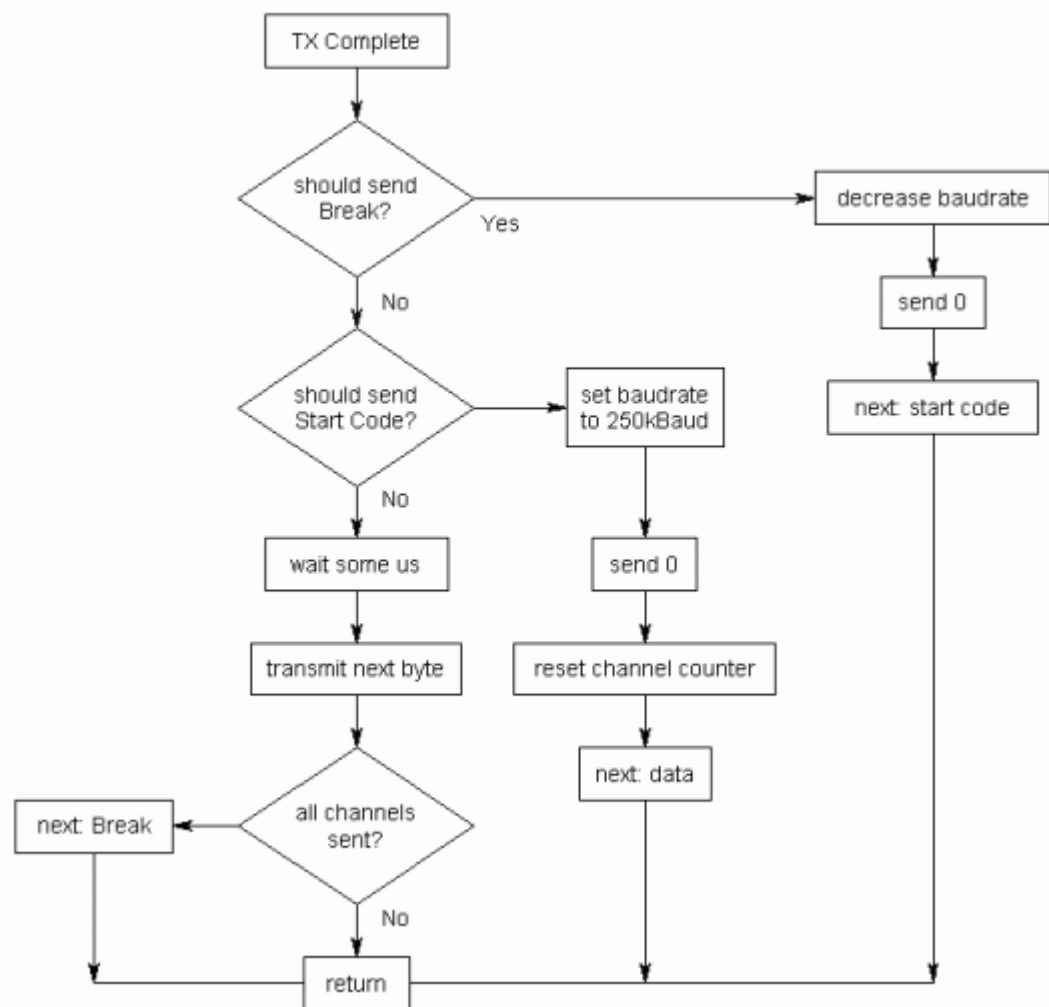


Abbildung 2: Flussdiagramm der state machine

Der Code wurde geschrieben im ATMEL Studio 7.

In der Headerdatei müssen folgende Variablen und Konstanten definiert sein:

```
#define F_CPU (8000000UL)           //oscillator freq.(typical 8MHz or 16MHz)
```

```
volatile uint8_t DmxField[50];     //array of DMX vals
```

F_CPU ist die Quarzfrequenz in Hz.

DmxField ist das Sendearray mit den zu sendenden Kanälen.

Mit „init_dmx“ wird der USART für das Senden von DMX-Daten initialisiert und das Sendearray auf 0 gesetzt.

© Hendrik Hölscher

all rights reserved

Das ungenehmigte Kopieren von Inhalten sowie Mirroring dieser AN ist untersagt.

Die Autoren übernehmen keine Haftung oder Gewährleistung.