

FSM System Exclusive Datenformat

Allgemeiner Aufbau:

(alle Daten hexadezimal)

F0 *00 20 0D* *<ID>* *07* *<CMD>* *<Daten> ...* ***F7***
 *1 *2 *3 *4 *5

***1 = Hersteller-ID: MIDITEMP**

***2 = Geräte-ID.** Falls mehrere FSM vorhanden sind, kann jedem FSM eine individuelle ID zugewiesen werden. Über das ID-Feld kann dann gezielt ein bestimmtes FSM angesprochen werden. Der Wert **7F** bedeutet, daß die SysEx-Nachricht von jedem FSM interpretiert wird, bei jedem anderen Wert wird nur das FSM angesprochen, dem vorher der entsprechende ID-Wert zugewiesen wurde.

Normalerweise, d.h. wenn nur ein FSM vorhanden ist, kann hier einfach immer der Datenwert **7F** eingetragen werden. Daher wird dieser Wert auch in den folgenden Beispielen verwendet.

***3 = Gerätetyp: 07 = FSM**

***4 = Kommando-ID.** Definiert die Funktion der SysEx-Nachricht:

00 = Daten/Funktion für Switch 1 ändern

01 = Daten/Funktion für Switch 2 ändern

02 = Daten für Pedal 1 ändern

03 = Daten für Pedal 2 ändern

04 = Geräte-ID des FSM ändern

***5 = Daten,** Inhalt je nach Kommando-ID

Codierung der zu sendenden MIDI Daten im SysEx Datensatz:

Da innerhalb von SysEx-Daten keine Werte $\geq 0x80$ zulässig sind, werden die MIDI Statusbytes als 0x00..0x7F codiert. So können sämtliche MIDI Event-Typen gesendet werden. Die maximale Länge eines Datensatzes beträgt 40 Bytes.

| MIDI Befehl: | Codierung: | Typ: |
|----------------------|-------------------|--|
| 8n NN VV | 0n NN VV | Note Off |
| 9n NN VV | 1n NN VV | Note On (VV=0 gilt als Note Off) |
| An NN VV | 2n NN VV | Polypressure |
| Bn NN VV | 3n NN VV | Control Change |
| Cn VV | 4n VV | Program Change |
| Dn VV | 5n VV | Channel Pressure |
| En vv VV | 6n vv VV | Pitch Bend |
| F0 data... F7 | 70 data... | SysEx (EOX= F7 wird hier nicht mit codiert!*) |
| F1..F7 | 71..77 | System Common (ggf. mit 1 od.2 Datenbytes) |
| F8..FF | 78..7F | System Realtime |

Erläuterungen:

n = MIDI Kanalnummer 0..F für Kanal 1..16

NN = Notenummer oder Controller-Nummer

VV = Datenwert (bei Noten: Velocity)

vv = Datenwert LSB

*SyEx-Daten werden ab dem codierten Statusbyte (**70**) bis zum Ende der SysEx-Meldung übernommen. Das abschließend zu sendende EOX (**F7**) wird beim Senden automatisch angehängt und darf nicht im Datensatz zur Programmierung mit codiert werden!

CMD = 00 oder 01: Funktion von Switch 1 oder 2 programmieren

Der Datenblock besteht hier aus einem „Mode“-Datenbyte, gefolgt von den bei Betätigung des Switches (Ein- bzw. Ausschalten) zu sendenden MIDI-Daten in codierter Form:

F0 00 20 0D <ID> 07 00/01 <Mode> [<MIDI Daten> ...] F7

| Mode | Bedeutung: |
|-----------|--|
| 00 | Toggle-Modus aus, vorher gespeicherte MIDI Daten für Ein-/Ausschalten werden nicht verändert. (das Feld <MIDI Daten> kann entfallen) |
| 01 | Toggle-Modus aus, die <MIDI Daten> werden für das Einschalten bzw. Drücken des Tasters übernommen. Sind keine <MIDI Daten> angegeben, werden beim Einschalten keine Daten gesendet. |
| 02 | Toggle-Modus aus, die <MIDI Daten> werden für das Ausschalten/Loslassen übernommen. Sind keine <MIDI Daten> angegeben, werden beim Ausschalten/Loslassen keine Daten gesendet. |
| 03 | Toggle-Modus aus, die <MIDI Daten> werden sowohl für das Einschalten, als auch für das Ausschalten übernommen. Der Wert VV jedes MIDI-Befehls wird beim Senden der Daten entsprechend dem aktuellen Status des Switches durch 00 („aus“) bzw. 7F („an“) ersetzt. |
| 04 | Toggle-Modus an, vorher gespeicherte MIDI Daten werden nicht verändert. (das Feld <MIDI Daten> kann entfallen) |
| 05 | Toggle-Modus an, die <MIDI Daten> werden für das Einschalten übernommen. Sind keine <MIDI Daten> angegeben, werden dann keine Daten gesendet. |
| 06 | Toggle-Modus an, die <MIDI Daten> werden für das Ausschalten übernommen. Sind keine <MIDI Daten> angegeben, werden dann keine Daten gesendet. |
| 07 | Toggle-Modus an, die <MIDI Daten> werden sowohl für das Einschalten, als auch für das Ausschalten übernommen. Der Wert VV jedes MIDI-Befehls wird beim Senden der Daten entsprechend dem aktuellen Status des Switches durch 00 („aus“) bzw. 7F („an“) ersetzt. |

Toggle-Modus:

Im Toggle-Modus verhält sich ein angeschlossener Taster wie ein Schalter, der bei jedem Drücken zwischen den Zuständen „ein“ und „aus“ hin- und herschaltet. Das Loslassen des Tasters führt hierbei zu keiner Aktion.

Bei ausgeschaltetem Toggle-Modus wird das Drücken des Tasters als „Einschalten“ und das Loslassen als „Ausschalten“ interpretiert.

Gesendete Datenwerte bei Switch1/Switch2:

Wenn mit **Mode = 01/02** oder **Mode = 05/06** getrennte Datensätze für das Ein- und Ausschalten gespeichert wurden, so werden die MIDI Daten beim jeweiligen Schaltvorgang mit beliebigen Datenwerten **VV** wie abgespeichert gesendet.

Mit **Mode = 03** oder **Mode = 07** wird beim Einschalten und beim Ausschalten der gleiche Datensatz benutzt. Allerdings werden die Werte **VV** in den gesendeten MIDI Befehlen in diesem Fall jeweils durch den aktuellen Switch-Status ersetzt.

Für einen gespeicherten Wert **VV = 00** wird beim Einschalten von Switch1/2 der Datenwert **7F**, beim Ausschalten der Wert **00** gesendet.

Wenn **VV = 7F** gespeichert wurde, wird beim Einschalten von Switch1/2 der Datenwert **00**, beim Ausschalten der Wert **7F** gesendet.

CMD = 02 oder 03: Funktion von Pedal 1 oder 2 programmieren

Der Datenblock besteht hier aus einem „Positions“-Datenbyte, gefolgt von den bei Betätigung des Switches (Ein- bzw. Ausschalten) zu sendenden MIDI-Daten in codierter Form:

F0 00 20 0D <ID> 07 02/03 <Pos> <MIDI Daten> ... F7

Für Pos = 0: Entsprechend der Codierung der zu sendenden **<MIDI Daten>** werden bei Bewegung des Pedals MIDI Befehle erzeugt. Dabei werden die Werte "VV" im Datensatz jeweils durch den Datenwert zur aktuellen Stellung des Pedals ersetzt. Für gespeicherte Werte **VV=0** wird der Wert normal, für **VV=7F** wird der Datenwert invertiert gesendet, d.h. die min/max-Stellung des Pedals wird vertauscht.

Für Pos > 0: Wie bei Pos=0, werden die **<MIDI Daten>** entsprechend der Codierung gesendet. Allerdings wird durch den Pos-Wert zusätzlich ein Byte bestimmt, dessen Wert dann durch die Pedalstellung ersetzt wird. In der Regel ist diese Möglichkeit nur in Verbindung mit dem Senden von System-Exclusive Befehlen nützlich, um zu bestimmen, welches Byte innerhalb der SysEx-Daten den Datenwert **VV** darstellt. Die Positionsangabe wird ab dem ersten Byte der **<MIDI Daten>** gezählt.

Auch bei Pos >0 gilt: Für gespeicherte Werte **VV=0** wird der Wert normal, für **VV=7F** wird der Datenwert invertiert gesendet, d.h. die min/max-Stellung des Pedals wird vertauscht.

CMD = 04: Geräte ID des FSM ändern

F0 00 20 0D <ID> 07 04 <neue ID> F7

Beispiele:

FSM Werkseinstellung für Taster an Anschluß (Switch 1): FX-Mute f. Yamaha 01V/96

Dazu werden auf MIDI Kanal 2 Control Change Befehle für die Controllernummern 72 bis 75 erzeugt, um die vier Effekt-Pfade per Tastendruck gleichzeitig ein- und ausschalten zu können.

F0 00 20 0D 7F 07 00 07 31 48 00 31 49 00 31 4A 00 31 4B 00 F7
*1 *2 *3 *4 *5 *6 *4 *5 *6 *4 *5 *6 *4 *5 *6

- *1= ID-Feld (MIDITEMP, Geräte-ID und Geräte-Typ FSM)
- *2= CMD = 00: Daten für Switch 1
- *3= Mode = 07: Toggle-Modus (Umschaltung ein<->aus bei jedem Tastendruck), der folgende Datensatz wird zum Ein- und Ausschalten verwendet.
- *4= 31 codiertes Statusbyte „Control Change“ auf MIDI-Kanal 2 (→ B1)
- *5= Controllernummer 48..4B (hexadezimal) = Nummer 72..75 (dezimal)
- *6= Datenwert (VV) = 0, Schalterstellung ein<->aus wird nicht vertauscht

FSM Werkseinstellung für Taster am Anschluß (Switch 2): sendet MIDI Start/Stop

Dazu werden zwei SysEx-Befehle an das FSM gesendet:

```

F0 00 20 0D 7F 07 01 05 7A F7
      *1          *2 *3 *4

F0 00 20 0D 7F 07 01 06 7C F7
      *1          *2 *5 *6

```

- *1= ID-Feld (MIDITEMP, Geräte-ID und Geräte-Typ FSM)
- *2= CMD = 01: Daten für Switch 2
- *3= Mode = 05: Toggle-Modus, folgende Daten werden beim Einschalten gesendet.
- *4= 7A = codiertes Statusbyte „MIDI Start“ (FA)
- *5= Mode = 06: Toggle-Modus, folgende Daten werden beim Ausschalten gesendet.
- *6= 7C = codiertes Statusbyte „MIDI Stop“ (FC)

Werkseinstellungen für Pedal 1/2: Expression Control (Control Chg. #11) auf MIDI Kanal 1 bzw. 2:

```

F0 00 20 0D 7F 07 02/03 00 30/31 0B 00 F7
      *1          *2 *3 *4 *5 *6

```

- *1= ID-Feld (MIDITEMP, Geräte-ID und Geräte-Typ FSM)
- *2= CMD = 02 bzw. 03: Daten für Pedal 1 bzw. 2
- *3= Pos = 00: kein Byte muß als VV markiert werden, da nur CC-Daten zu senden sind.
- *4= 30 bzw. 31 = codiertes Statusbyte für Control Change auf MIDI Kanal 1 bzw. 2
- *5= 0B = Controllernummer 11(dezimal): „Expression Control“
- *6= 00 = Controllerwert nicht invertiert.

Änderungen vorbehalten. Keine Haftung für Irrtümer und Druckfehler.
 © MIDITEMP GmbH Stand_01_2010

Weitere Beispiele bzw. Downloads finden Sie auf unserer Homepage, www.Miditemp.de