Style Files - Introduction and Details

Version 2.0   19 June 2014

1 Preface

Modern Yamaha[1] keyboards provide sophisticated accompaniment functions. They have built in "accompaniment styles" for a number of different musical genres. But many keyboards are not limited to these built in styles. They provide the capability to use additional styles loaded into the keyboard, or even to create new styles. Additional styles may be purchased, downloaded from the internet, created from various sources or created from scratch.

There is a lot of information available on the internet regarding these styles. But this information is widely spread and difficult to find, especially for beginners.

This document tries to summarize all this information to provide an easy entry point for beginners as well as a reference for advanced style creators or software programmers. The document focuses on the technical details of styles. It does not cover the musical aspects for creating styles (see chapter 7.3 for links to style creation info).

As said, most of the information is already available on the Internet, due to the hard work and generosity of a lot of people. Please refer to chapter 6 for details about the contributing people. The authors of this document explicitly want to avoid the impression that they may claim credit for other peoples work.

The authors of this document are not affiliated with Yamaha in any way, and Yamaha has not specifically approved the inclusion any of the information therein. The information presented has assembled from material posted by others on the internet, or discovered through private experimentation; no representation is made regarding its accuracy or completeness.

2 Introduction

This document is about styles for Yamaha keyboards. Keyboards from other manufacturers may have similar style functionality, but the details are very different. So everything in this document is only related to Yamaha keyboards. Some features of styles are not available on all keyboards, especially not on older models.

There are two categories of styles:
- Built in styles
- Loadable styles

Smaller/older keyboards may have only built in styles and do not support loadable styles. Modern keyboards often support both categories.

A loadable style is a file, exactly like used on Microsoft Windows[2] computers. How this style file is loaded into the keyboard depends on the keyboard model. It may be loaded using a standard floppy disk, an USB stick, a Smart Media card, a CD or a connection to a computer. This implies that the style file may easily be transferred to/from a computer.[3]

Style files can be modified by some keyboard models as well as with programs running on a computer. There are a lot of free programs available that allow creating and modifying styles in any manner. (See chapter 7.1 for more information.)

The main subjects of this document are the details about these style files.

---

[1] Yamaha is a trademark of Yamaha Corporation.
[2] Windows is a trademark of Microsoft Corporation.
[3] Periodically users report problems trying to access, view, rename or save style files.
To eliminate these problems, in Windows Explorer/Tools/Folder Options/View Tab, confirm the following settings:
'Hide extensions for known file types' is unchecked
'Display the content of system folders' is checked
'Show hidden files and folders' is checked.

3 Styles – What they are and how they work in general

A style is a special form of a type 0 midi file followed by several information sections. To function, it must be loaded into the PSR. This process reads the file and establishes some of the instrument settings based upon commands in the midi and information sections. When the accompaniment is started (via synch start, the Start button or an external midi command) the portions of the midi section are played in response to the state of the front panel style control buttons.

Internally, a style starts by specifying the tempo, the time signature and the copyright followed by several sections that are defined by marker events.

The first two sections, SFF1 (or SFF2) and SInt, occupying the first measure of the midi part, include a Midi On plus midi commands to setup the default instruments and the amount of DSP (only DSP1 as a system effect is available for styles) used for each track.

Each of the other markers (Intro A, Main B, etc) defines musical patterns that are triggered by the keying chords. Intros play only once when triggered and then turn control over to the next section selected by the panel buttons. Main sections (A, B, C, and D) repeat until the style is stopped or an Ending or an Intro is selected. Ending sections play once and the style is stopped. Fill Ins are triggered manually, or play automatically (if Auto Fill is On) when a new main section is selected.

When a style is playing in the instrument, the SFF and SInt sections are executed when a style section is changed. This resets the voices and other channel parameters to their initial values. Because of this, if its is desired to change the voice or other settings for a single section, new settings can be inserted in only this section and the style will revert to the default whenever another section is selected.

Fill Ins are limited to one measure in length; other sections can be any length up to 255 measures, but are typically 2-8 measures.

4 Style File Format

This chapter describes how the various data is stored in style files. This includes the structures of the data, their sizes, their order, their coding, etc. This information is especially useful for programmers.

A description about the contents and detailed meaning of the data (i.e. everything that may be added or modified with one of the special style editors) is described in chapter 5.
This information is useful for programmers as well as for all people trying to create or modify styles.

4.1 Conventions

Throughout the document numbers are written in different kinds, depending on what is appropriate in the context:

- Decimal   Decimal numbers are written without any prefix or suffix,
                    e.g. 256.
- Hexadecimal   Hexadecimal numbers are written with the suffix "H",
                    e.g. 1FH, 25H.
- Binary   Binary numbers are written with the suffix "B",
                    e.g. 00001110B

See the appendix A for a decimal-hexadecimal translation table.

4.2   General

Style files may have different file extensions. Currently the following are known:

- ".sty"   Standard
- ".bcs"   Basic
- ".prs"   Professional
- ".sst"   Session
- ".pcs"   PianoCombo
- ".pst"   Pianist
- ".fps"   Free Play Style

All these style file types have the same format, which is described in this chapter. Older keyboard may require that the style file has the extension ".sty". In this case, just renaming the style file to the extension ".sty" will often work.

Currently not much is known about the effect of these style file extensions except that the style file type is displayed together with the style name on some instruments.

4.3   General structure of a style file

A simple style file is just a standard midi file. This midi file has to be a midi type 0 file. It contains the musical sequences of the style as well as some control information. The midi data has to follow a number of rules. These rules are described in detail in chapter 4.5.

Modern style files are extended by adding special data sections at the end of this midi file.

An extended style file consists of one or more different sections of the following types:
- MIDI section   (mandatory)
- CASM section   (optional)
- OTS (One Touch Setting) section   (optional)
- MDB (Music Finder) section   (optional)
- MH section   (optional) (very rarely used)

The midi section is the only mandatory section. It contains the musical sequences of the style. An optional CASM section contains extended information for the keyboard how to interpret and control playing of the style section. While its inclusion is optional, very likely the style's author used it to convey important information and the style will not reproduce properly if removed. The OTS (One Touch Setting) section contains information for the four settings selectable from the keyboard. These can be used to easily setup the keyboard before using the style. The MDB (Music Finder) section contains information for what songs this particular style is appropriate. This information is automatically added to the Music Finder function, if the keyboard supports it. In very rare cases there is a MH section at the end of the style file. Nothing is known about the purpose of this section. For details see the next chapters.

Only one section of each type may be present in a style file.

Note that older/smaller keyboard models may not support styles with OTS or MDB sections. There may also be a limit on the size of the style file that can be loaded into the keyboard. There are programs available that can be used in this case to remove unwanted sections or parts from the style file. For more details see chapter 7.1.

The common order of the sections in the file is at follows:
1. Midi section
2. CASM section
3. OTS (One Touch Setting) section
4. MDB (Music Finder) section
5. MH section

Other orders may also work, but for compatibility reasons it is recommended to use the above order.

Programs that work with style files should not depend on the order and existence of optional sections when reading style files. When writing style files, the programs should use the common order of the sections to avoid possible problems with the various keyboard models.

Note: Many programs designed to read and/or edit standard midi files (e.g. sequencers, editors, players) will remove the optional sections and the files generated by them will not function properly in the instrument. This can be avoided by using programs that specifically designed to work with style files.

4.4   General structure of sections

The overall structure of a style file is:

Section 1 (mandatory Midi section)

Section 2 (optional)

Section 3 (optional)

Section 4 (optional)
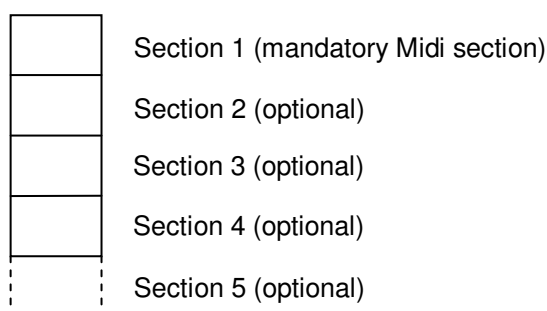
Section 5 (optional)

Note: As the optional section 5 (MH section) is very rarely used, and especially not present in current style files, it is no more shown in the following diagrams.

The sections are structured in such a way that the beginning and end of a section can be found without having to know all the internal details of the concerning structure. This allows a program to find the beginning of a specific section without even to know anything about the other section types.

As said above, section 1 is always a standard midi file structure of a midi type 0 file. The general structure of this section is a little bit different than the structure of sections 2...4, which share the same common structure.

Structure of section 1 (midi section):
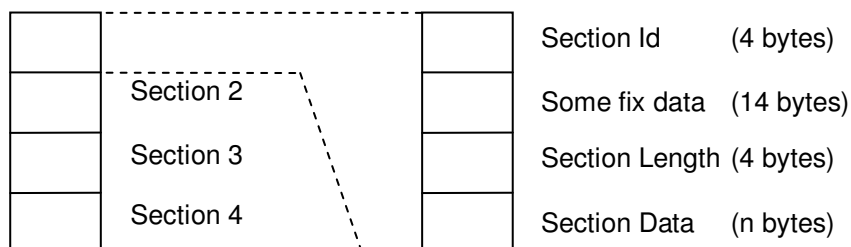
Section 2

Section 3

Section 4

Section Id   (4 bytes)

Some fix data   (14 bytes)

Section Length (4 bytes)

Section Data   (n bytes)

Common structure for sections 2…4:

Section 1

Section 3

Section 4

Section Id   (4 bytes)

Section Length (4 bytes)

Section Data   (n bytes)
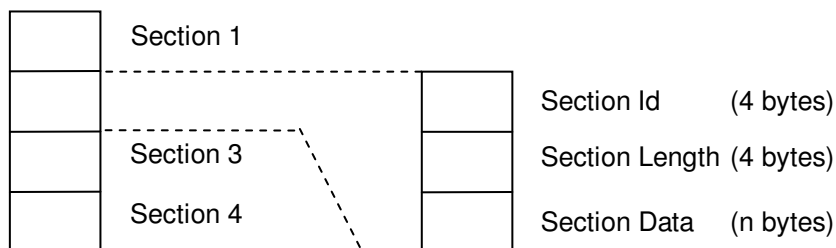
Details of general structure for section 1 (midi section):

Byte Index[4]		Description
0..3	byte[0] = 'M' (4DH) byte[1] = 'T' (54H) byte[2] = 'h' (68H) byte[3] = 'd' (64H)	This 4-character sequence identifies this section as a midi file, which in a style consists of a midi header followed by a track header and track data. Note: The characters are case sensitive.
4..7	Nr of data bytes = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the nr of header data bytes following. The length of the midi header is always 6 bytes. This means, the first byte which is counted here of the track header is byte[14].
8..13	Data bytes	Midi header data.
14..17	byte[14] = 'M' (4DH) byte[15] = 'T' (54H) byte[16] = 'r' (72H) byte[17] = 'k' (6BH)	This 4-character sequence identifies the midi track. Note: The characters are case sensitive.
18..21	Nr of data bytes = 256*256*256*byte[18] + 256*256*byte[19] + 256*byte[20] + byte[21]	Number of bytes in the midi track. The first byte that is counted here is byte[22].
22..n	Midi data bytes (Number as given above)	Midi track data. More details are described in chapter 4.5.

Table 1

Details of common structure for section 2...4:

Byte Index		Description
0..3	byte[0] = 'X' byte[1] = 'X' byte[2] = 'X' byte[3] = 'X'	This 4-character sequence identifies the type of the section. (See the individual section chapters for more information.) Note: The characters are case sensitive.
4..7	Nr of data bytes = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the nr of data bytes following. This means, the first byte that is counted here is byte[8].
8..n	Data bytes	Section dependent data.

---

[4] The byte index always starts from the beginning of the section, structure or substructure which is currently discussed.

	(Number as given above)	More details are described in chapter 4.5.2, 4.7 and 4.8.

Table 2

More details are described in the individual section chapters below.

4.5 Midi Data section

4.5.1 General

The midi section is midi type 0, which means that there is one midi track.
In the first measure there is a marker event which informs about the version of the style file format. Currently there are two different marker values:

- SFF1
- SFF2   New format introduced with the Tyros 3 keyboard (Sept. 2008).
             Also named "SFF GE".
             The only difference is the new "Cbt2" sctructure described in chapter 4.6.3.2

SFF1 format files that are loaded into instruments that support SFF2 are automatically converted to SFF2.

4.5.2 Midi Command Format

4.5.2.1 General

The data following the Midi and Track headers are midi events. Unlike the header data, the fields are not organized in a fixed format, but are records of various lengths whose general format is:
                    Execution Time - Command Byte – Data

They are generally organized as follows.

Function	Description	Byte Length
Execution Time	Number of ticks since last event.	Variable length, <=4
Command ID	Identifies the type of the command 00H .. 7FH running mode, command id not present 80H .. EFH midi events F0H sysex FFH meta data	1
Data	3 types: midi events, sysex events, meta events	Varies by command type

Table 3

There is also an abbreviated command format, called running mode where the Command ID is omitted, i.e.: Time Data. In this case the last Command ID is used. This mode is identified by a value <128 in the Command ID location.

The use of variable length formatting and running mode was included in the specification to reduce the size (and hence transmission time) of midi files.

4.5.2.2 Details

4.5.2.2.1 Time

There are one to four time bytes that are at the beginning of each midi event. Time is measured in "delta time" which is defined as the number of ticks (the resolution of which is defined in the header) before the midi event is to be executed. I.e., a delta time of 0 = immediately; a delta time of 960 when the resolution is 1920 ticks per quarter note is after a 1/8 note rest. Delta time is a variable length format using 7 of the 8 available bits; the maximum time value of any time byte is 127 (7FH). The first or 8th bit is used to identify the last of the delta time bytes; the least significant byte is indicated by a leading bit =0, all other bytes have a leading bit=1.

Total delta time= 128^3 (byte4) + 128^2(byte3) + 128*(byte2) + byte1

4.5.2.2.2 Midi Events (Command ID 127-239)

Midi event send commands to one of the 16 possible midi channels. The event command consists of a leading 4 bits that identifies the command and a trailing 4 bits that identifies the midi channel. In the table below, X=midi channel (0-15, 0H-FH). Available commands are:

Command ID (Hex)	Data	Description	Byte Length
8X	nn vv	Note On. nn=note number (0-127); vv= velocity (0-127)	3
9X	nn vv	Note Off; see above	3
AX	kk vv	Key Press	3
BX	cc uu	Control Change; cc=controller number*, uu = data value	3
CX	pp	Program Change; pp= program number (0-127)	2
EX	v1 v2	Pitch Wheel Change; v1= bottom value, v2=top value	3

* allowed values listed in "Meaning, Functionality and Requirements of Midi Data used in Styles" later in this document.

Table 4

4.5.2.2.3 Sysex Events

Sysex Events which are used to provide instrument control such a master pitch, DSP settings, etc. They do not specify a channel. The total event length is equal to the sum of the command ID byte, data length byte(s) in variable length format and the data length.

Byte	Function	Description
1	Command ID	always 240 (F7H)
1+	Length, not including ID and length byte(s)	Variable length format
Defined by length	Sysex Data	Last byte always 247 (F7H)

Table 5

4.5.2.2.4 Meta Events

Meta events convey general information such as copyright, lyrics, tempo, time & key signature. They do not specify a channel. The total event length is equal to the sum of the command ID byte, data length byte(s) in variable length format and the data length.

Command ID	Meta ID	Length	Data Description

255(FFH)	0(00H)	2	Sequence number
"	1(01H)	Length of text	Text data
"	2(02H)	Length of text	Copyright text
"	3(03H)	Length of text	Track name text
"	4(04H)	Length of text	Track Instrument name text
"	5(05H)	Length of text	Lyric text
"	6(06H)	Length of text	Marker text
"	7(07H)	Length of text	Cue point text
"	32(20H)	1	Midi Channel Prefix
"	33(21H)	1	Midi Port
"	47(2FH)	1	End of Track
"	81(51H)	3	Tempo in microseconds /quarternote
"	84(54H)	5	SMPTE Offset (hr + min + sec + frame + frame)
"	88(58H)	4	Time signature=numerator + denominator (2= quarter, 3= eighth) + Ticks in metronome click + number of 32nd notes to the quarter note
"	89(59H)	2	Key signature= sharps/flats (- value= number of flats, 0= key of C, + value = number of sharps) + major/minor(0/1)
"	127 (7FH)	Length of data	Yamaha sequencer specific info.

Table 6

4.6   CASM section

The information in the CASM section is necessary if the midi section does not follow the rules for "simple" style files, which do not necessarily need a CASM section (see chapter 5.2.1 for the rules). The CASM section gives instructions to the instrument on how to deal with the midi data.

This includes:
- Assigning the sixteen possible midi channels to 8 accompaniment channels which are available to a style in the instrument (9 = sub rhythm, 10 = rhythm, 11 = bass, 12 = chord 1, 13 = chord 2, 14 = pad, 15 = phrase 1, 16 = phrase 2). More than one midi channel may be assigned to an accompaniment channel.
- Allowing the PSR to edit the source channel in StyleCreator. This setting is overridden by the instrument if the style has > 1 midi source channel assigned to an accompaniment channel. In this case the source channels are not editable in the instrument and external software must be employed.
- Muting/enabling specific notes or chords to trigger the accompaniment. In practice, chord choices are often used in Main sections and while Intros and Endings occasionally use both (e.g. ModernPicking.prs).
- The key that is used in the midi channel. Styles often use different keys for the midi data. Styles without a CASM must be in the key of CMaj7.
- How the chords and notes are transposed as chords are changed and how notes held through chord changes are reproduced.
- The range of notes generated by the style.

See chapter 5.2 for a more detailed description of the usage of this data.

4.6.1   General

There is only one CASM section in a style file.

The CASM section allows defining separate instructions for each style part (e.g. Intro A, Main B) of each source midi channel.
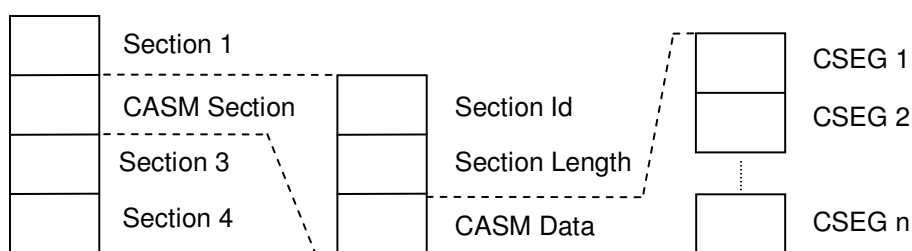
First level of details about the structure of the CASM section:

Byte Index		Description
0..3	byte[0] = 'C' (43H) byte[1] = 'A' (41H) byte[2] = 'S' (53H) byte[3] = 'M' (4DH)	This 4-character sequence identifies this section as a CASM section. Note: The characters are case sensitive.
4..7	Nr of data bytes = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the nr of data bytes following. This means, the first byte that is counted here is byte[8].
8..n	Data bytes (Number as given above)	CASM data. More details are described in chapter 4.6.2.

Table 7

4.6.2 CASM data

The CASM data consists of one or more CSEG structures.

A CSEG structure contains the data related to a style part in the midi section (e.g. Intro A, Main B). One CSEG structure may be associated to more than one style part, which means that these style parts share the same data. For current instruments there may be a maximum number of 16 CSEG structures (for 3 Intros, 3 Endings, 4 Variation Fill Ins, 4 Mains and 1 Break Fill In).

The number of CSEG structures depends on the number of style parts in the midi section and whether there are CSEG structures covering multiple style parts. The actual number of CSEG structures has to be derived from the size of the CASM data. This means, as long as the total size of found structures is less than the size of the CASM data, there will be additional CSEG structures.

CASM Section

CSEG 1
CSEG 2

Section Id
Section Length
CASM Data

CSEG n

Section Id
Section Length
CSEG Data

Details about the CSEG structure:

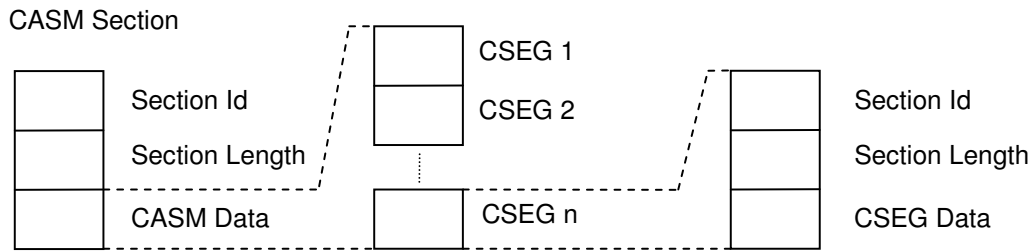| Byte Index		Description
0..3	byte[0] = 'C' (43H) byte[1] = 'S' (53H) byte[2] = 'E' (45H) byte[3] = 'G' (47H)	This 4-character sequence identifies this section as a CSEG structure. Note: The characters are case sensitive.
4..7	Nr of data bytes = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the nr of data bytes following. This means, the first byte that is counted here is byte[8].
8..n	Data bytes (Number as given above)	CSEG data. More details are described in chapter 4.6.3

Table 8

4.6.3 CSEG Data

The CSEG structure consists of one Sdec structure, one or more Ctab structures and one or more optional Cntt structures.

The number of Ctab and Cntt structures has to be derived from the size of the CSEG data. This means, as long as the total size of found structures is less than the size of the CSEG data there will be additional Ctab or Cntt structures. Alternatively, examining the data for repeated use of the IDs used in the CASM section (CSEG, Sdec, Ctab and Cntt) will also access any additional CASM data.

CASM Data

CSEG 1
CSEG 2

CSEG n

Section Id
Section Length
CSEG Data

Sdec
Ctab 1

Ctab n

Cntt 1 (optional)

Cntt m (optional)

4.6.3.1 Sdec structure

There is one Sdec structure at the beginning of the CSEG data.

The Sdec structure defines for which style part or parts (e.g. Intro A, Main B) the following data in the Ctab and Cntt structures belongs to. If there is more than one style part listed, this means that these style parts share the same data.
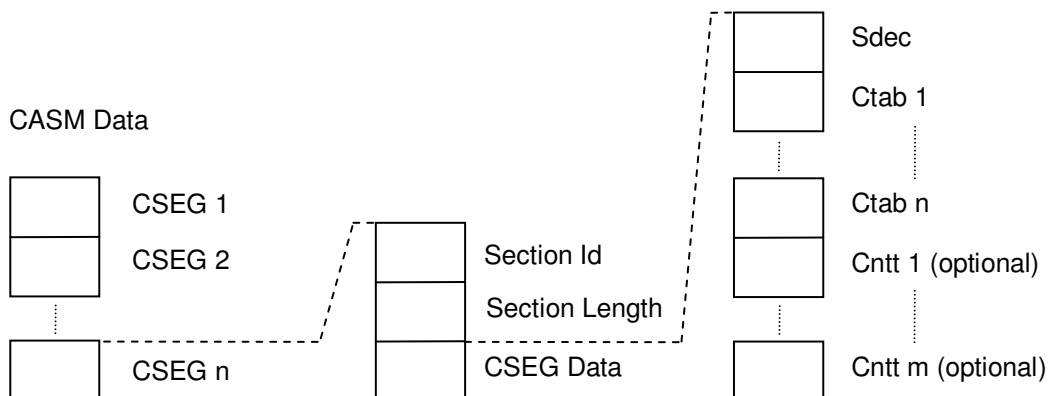
Details about the Sdec structure:

Byte Index		Description
0..3	byte[0] = 'S' (53H) byte[1] = 'd' (64H) byte[2] = 'e' (65H) byte[3] = 'c' (63H)	This 4-character sequence identifies this structure as an Sdec structure. Note: The characters are case sensitive.
4..7	Nr of data bytes = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the nr of data bytes following. This means, the first byte that is counted here is byte[8].
8..n	Data bytes (Number as given above) e.g. byte[8] = 'M' byte[9] = 'a' byte[10] = 'i' byte[11] = 'n' byte[12] = ' ' byte[13] = 'A' byte[14] = ',' byte[15] = 'I' byte[16] = 'n' byte[17] = 't' byte[18] = 'r' byte[19] = 'o' byte[20] = ' ' byte[21] = 'B'	Sdec data. Contains one or more names of style parts for which this CSEG data has to be used. Multiple names are separated with commas. There is no comma after the last style part name string. The strings are case sensitive. Valid strings are: Intro A Intro B Intro C Intro D (only supported by PSR-2000) Main A Main B Main C Main D Fill In AA Fill In BB Fill In CC Fill In DD Fill In BA (for the "Break" section) Ending A Ending B Ending C Ending D (only supported by PSR-2000)

Table 9

4.6.3.2 Ctab (Ctb2) structure

Immediately after the Sdec structure there are one or more Ctab structures. The number of Ctab structures depends upon the number of midi channels used in style parts covered by the SDEC section. There is one Ctab structure for each midi source channel used in the midi section for the related style sections.

The number of Ctab structures has to be derived from the size of the CSEG data.

Since the introduction of the Tyros 3 keyboard there are two types of Ctab structures, the old "Ctab" and the new "Ctb2". The new "Ctb2" structure can not be processed by the keyboards prior to the Tyros 3. When loading such a style file, the keyboard reports an error.

The two versions of the structures can be distinguished by the first four bytes, which represent the characters "Ctab" or "Ctb2". The first part of the Ctab data part is the same for both structures, the second part is different.

Details about the Ctab structure:

Byte Index		Description
0..3	byte[0] = 'C' (43H) byte[1] = 't' (74H) byte[2] = 'a' (61H) byte[3] = 'b' (62H)	This 4-character sequence identifies this structure as a Ctab structure. Note: The characters are case sensitive.
4..7	Nr of data bytes = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the nr of data bytes following. This means, the first byte that is counted here is byte[m+8].
8..n	Data bytes (Number as given above)	Ctab data. More details are described in chapter 4.6.3.3.

Table 10

Details about the Ctb2 structure:

Byte Index		Description
0..3	byte[0] = 'C' (43H) byte[1] = 't' (74H) byte[2] = 'b' (62H) byte[3] = '2' (32H)	This 4-character sequence identifies this structure as a Ctb2 structure. Note: The characters are case sensitive.
4..7	Nr of data bytes = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the nr of data bytes following. This means, the first byte that is counted here is byte[m+8].
8..n	Data bytes (Number as given above)	Ctb2 data. More details are described in chapter 4.6.3.3.

Table 11

4.6.3.3   Ctab / Ctb2 data (first part)

The format of the first part of the Ctab data and the Ctb2 data is the same for both structures.

Byte Index		Description
0	Source channel Values 00H .. 0FH	Midi source channel The values 00H .. 0FH represent the midi channels 1..16. For details about the meaning see chapter 5.2.2.
1…8	Name	Name.

	Character 1 = byte[1] : Character 8 = byte[8]	The name can be any string up to 8 characters long. Unused characters at the end of the name are filled with spaces (20H). Does not contain any "end of string" termination (like '\0').

9	Destination channel
	Values 08H .. 0FH
		The accompaniment midi channel to which the source channel should be mapped.
		
		The values 08H .. 0FH represent the midi channels 9..16.
		Valid destination channels are 9..16:
		  9 = sub rhythm
		  10 = rhythm
		  11 = bass
		  12 = chord 1
		  13 = chord 2
		  14 = pad
		  15 = phrase 1
		  16 = phrase 2
		
		For details about the meaning see chapter 5.2.2.

10	Editable Values 00H or 01H	Defines if the data in the source channel is editable. 00H = Channel data is editable 01H = Channel data is NOT editable The instruments inhibit editing of non rhythm parts (destination > 10) and force the user to delete all events in the Style Creator. The setting only has meaning to Style Creator, and then not always there. If a destination channel has more than one source channel, then the instrument will not allow editing, regardless of this setting.
11	Note Mute (Part 1) Values 00H .. 0FH This byte has to be interpreted as an array of bits. Each bit represents one note. (Bit 7 = highest bit (MSB)) Bit 7 = unused, always 0 Bit 6 = unused, always 0 Bit 5 = unused, always 0 Bit 4 = unused, always 0 Bit 3 = note B Bit 2 = note Bb Bit 1 = note A Bit 0 = note G#	Bit = 1 -> note will be played Bit = 0 -> note will NOT be played Example: 0EH (= 00001110B) means: • Notes A, Bb, B will be played. • If note G# is pressed the accompaniment is muted. For details about the meaning of these values see chapter 5.2.4.
12	Note Mute (Part 2) Values 00H .. FFH This byte has to be interpreted as an array of bits. Each bit represents one note. (Bit 7 = highest bit (MSB))	Bit = 1 -> note will be played Bit = 0 -> note will NOT be played

	Bit 7 = note G Bit 6 = note F# Bit 5 = note F Bit 4 = note E Bit 3 = note Eb Bit 2 = note D Bit 1 = note C# Bit 0 = note C	
13	Chord Mute (Part 1) Values 00H .. 0FH This byte has to be interpreted as an array of bits. Each bit represents one chord type. (Bit 7 = highest bit (MSB)) Bit 7 = unused, always 0 Bit 6 = unused, always 0 Bit 5 = unused, always 0 Bit 4 = unused, always 0 Bit 3 = ? (maybe unused?) Bit 2 = autostart enable Bit 1 = 1+2+5 Bit 0 = sus4	Bit = 1 -> when this chord is played the accompaniment of this source channel is played. Bit = 0 -> when this chord is played the accompaniment of this source channel is muted. Example: 02H (= 00000010B) means: When playing a sus4 chord the accompaniment is muted. Bit 2 and Bit 3 are only used for drum and percussion channels. If Bit 2 = 1 then Auto Start is enabled and the channel will play accompaniment before the first chord is pressed. This allows the drums to play from the beginning and instruments to come in later. For details about the meaning of these values see chapter 5.2.3.
14	Chord Mute (Part 2) Values 00H .. FFH This byte has to be interpreted as an array of bits. Each bit represents one chord type. (Bit 7 = highest bit (MSB)) Bit 7 = 1+5 Bit 6 = 1+8 Bit 5 = 7aug Bit 4 = Maj7aug Bit 3 = 7(#9) Bit 2 = 7(b13) Bit 1 = 7(b9) Bit 0 = 7(13)	Bit = 1 -> when this chord is played the accompaniment of this source channel is played. Bit = 0 -> when this chord is played the accompaniment of this source channel is muted.
15	Chord Mute (Part 3) Values 00H .. FFH This byte has to be interpreted as an array of bits. Each bit represents one chord type. (Bit 7 = highest bit (MSB)) Bit 7 = 7#11 Bit 6 = 7(9)	Bit = 1 -> when this chord is played the accompaniment is played. Bit = 0 -> when this chord is played the accompaniment is muted.

	Bit 5 = 7b5 Bit 4 = 7sus4 Bit 3 = 7th Bit 2 = dim7 Bit 1 = dim Bit 0 = minMaj7(9)	
16	Chord Mute (Part 4) Values 00H .. FFH This byte has to be interpreted as an array of bits. Each bit represents one chord type. (Bit 7 = highest bit (MSB)) Bit 7 = minMaj7 Bit 6 = min7(11) Bit 5 = min7(9) Bit 4 = min(9) Bit 3 = m7b5 Bit 2 = min7 Bit 1 = min6 Bit 0 = min	Bit = 1 -> when this chord is played the accompaniment is played. Bit = 0 -> when this chord is played the accompaniment is muted.
17	Chord Mute (Part 5) Values 00H .. FFH This byte has to be interpreted as an array of bits. Each bit represents one chord type. (Bit 7 = highest bit (MSB)) Bit 7 = aug Bit 6 = Maj6(9) Bit 5 = Maj7(9) Bit 4 = Maj(9) Bit 3 = Maj7#11 Bit 2 = Maj7 Bit 1 = Maj6 Bit 0 = Maj	Bit = 1 -> when this chord is played the accompaniment is played. Bit = 0 -> when this chord is played the accompaniment is muted.
18	Source Chord Values 00H .. 0BH	Determines the original key of the source channel together with the following byte (i.e. the key used when recording the source channel). On the instruments the default, CMaj7 (the source root is "C" and the source chord type is "Maj7"), is automatically selected whenever the preset data is deleted prior to recording a new style, regardless of the source root and chord included in the preset data. 00H = C 01H = C# 02H = D 03H = Eb 04H = E 05H = F 06H = F#

V2.0

		07H = G 08H = G# 09H = A 0AH = Bb 0BH = B	
19	Source Chord Type Values 00H .. 21H	00H = Maj 01H = Maj6 02H = Maj7 03H = Maj7#11 04H = Maj(9) 05H = Maj7(9) 06H = Maj6(9) 07H = aug 08H = min 09H = min6 0AH = min7 0BH = min7b5 0CH = min(9) 0DH = min7(9) 0EH = min7(11) 0FH = minMaj7	10H = minMaj7(9) 11H = dim 12H = dim7 13H = 7th 14H = 7sus4 15H = 7b5 16H = 7(9) 17H = 7#11 18H = 7(13) 19H = 7(b9) 1AH = 7(b13) 1BH = 7(#9) 1CH = Maj7aug 1DH = 7aug 1EH = 1+8 1FH = 1+5 20H = sus4 21H = 1+2+5 22H = cancel (stop all instruments)
20..n		Ctab data (second part) More details are described in chapter 4.6.3.4 or Ctb2 data (second part) More details are described in chapter 4.6.3.5	

Table 12

4.6.3.4   Ctab data (second part)

This is the format of the second part of a Ctab data structure.

Byte Index		Description
20	Note Transposition Rule (NTR) Values 00H .. 01H	Specifies the transposition rule to be used by the transposition table. 00H = Root Transposition 01H = Root Fixed For details about the meaning of these values see chapter 5.2.5.
21	Note Transposition Table (NTT) Values 00H .. 05H	Specifies the note transposition table to be used for source pattern transposition. 00H = Bypass 01H = Melody 02H = Chord 03H = Bass 04H = Melodic Minor 05H = Harmonic Minor For details about the meaning of these values see chapter 5.2.5.2.

		Note: The NTT values used for this byte differ from NTT values used in other structures.
22	High Key Values 00H .. 0BH	Specifies the upper root limit. Chords with a root higher than the specified limit will be played in the octave immediately below the high-key limit. This setting is effective only when the NTR (Note Transposition Rule) (above) is set to "Root Trans". 00H = C 01H = C# 02H = D 03H = Eb 04H = E 05H = F 06H = F# 07H = G 08H = G# 09H = A 0AH = Bb 0BH = B For details about the meaning see chapter 5.2.7.
23	Note Low Limit Values 00H .. 7FH	The values 00H .. 7FH represent the midi note numbers 0 .. 127. "Note Low Limit" and "Note High Limit" specify the low and high note limits for all notes in the specified part. Notes outside this range are transposed to the nearest octave within the range. For details about the meaning see chapter 5.2.6.
24	Note High Limit Values 00H .. 7FH	The values 00H .. 7FH represent the midi note numbers 0 .. 127. For details about the meaning see chapter 5.2.6.
25	Retrigger Rule (RTR) Values 00H .. 05H	Specifies how notes held through chord changes will be handled. 00H = Stop 01H = Pitch shift 02H = Pitch shift to root 03H = Retrigger 04H = Retrigger to root 05H = Note generator For details about the meaning of these values see chapter 5.2.8.
26..n	Special features	One or more data bytes. If the value of byte 26 is 00H (= no special feature), then there are no following bytes. If the value of byte 26 is 01H (=extra break drum voice), then there are 4 following bytes (bytes 27...30). For more details see chapter 5.2.9.

V2.0

Table 13

4.6.3.5 Ctb2 data (second part)

This is the format of the second part of a Ctb2 data structure.
The full range of midi notes can be split up to a maximum of 3 sections (byte 20 and 21), for low, middle and high notes. For each range there is a separate set of data (NTR, NTT,..RTR).

There are also some still unknown bytes.

Byte Index		Description
20	Lowest note of middle notes.	Specifies the lowest midi note value which is part of the "middle note section" (see bytes 28...33). All notes below this note belong to the "low notes section". If the value of this byte is 0, then the data in the "low notes section" is not used.
21	Highest note of middle notes.	Specifies the highest midi note value which is part of the "middle note section" (see bytes 28...33). All notes above this note belong to the "high notes section". If the value of this byte is 7FH, then the data in the "high notes section" is not used.
22..27	Ctb2 sub-structure for low notes	Specifies the ctb2 data for low notes. For more details see chapter 4.6.3.6
28..33	Ctb2 sub-structure for middle notes	Specifies the ctb2 data for middle notes. For more details see chapter 4.6.3.6
34..39	Ctb2 sub-structure for high notes	Specifies the ctb2 data for high notes. For more details see chapter 4.6.3.6
40..46	Unknown bytes	The meaning of these 7 bytes is unknown. For what is currently known see chapter 4.6.3.7

Table 14

4.6.3.6 Ctb2 data sub-structure

This is the format of the Ctb2 data sub-structure.

Byte Index		Description
0	Note Transposition Rule (NTR) Values 00H .. 02H	Specifies the transposition rule to be used by the transposition table. 00H = Root Transposition 01H = Root Fixed 02H = Guitar For details about the meaning of these values see chapter 5.2.5.
1	Note Transposition Table (NTT) Values 00H .. 0AH Bit 7 indicates Bass on/off (Bit 7 = highest bit (MSB))	Specifies the note transposition table to be used for source pattern transposition. If NTR = "Root Transposition" or "Root Fixed": 00H / 80H = Bypass 01H / 81H = Melody 02H / 82H = Chord

		03H / 83H = Melodic minor 04H / 84H = Melodic minor 5th Var. 05H / 85H = Harmonic minor 06H / 86H = Harmonic minor 5th Var. 07H / 87H = Natural minor 08H / 88H = Natural minor 5th Var. 09H / 89H = Dorian 0AH / 8AH = Dorian 5th Var. If Bass is off values 00H – 0AH are used, else values 80H – 8AH. If NTR = "Guitar" 00H / 80H = All-Purpose 01H / 81H = Stroke 02H / 82H = Arpeggio If Bass is off values 00H - 02H are used, else values 80H - 82H. The part (channel) for which Bass is set to On recognize on-bass chords allowed in the Fingered-on-Bass fingering mode, regardless of the NTT setting. For details about the meaning of these values see chapter 5.2.5.2.
2	High Key Values 00H .. 0BH	Specifies the upper root limit. Chords with a root higher than the specified limit will be played in the octave immediately below the high-key limit. This setting is effective only when the NTR (Note Transposition Rule) (above) is set to "Root Trans". 00H = C 01H = C# 02H = D 03H = Eb 04H = E 05H = F 06H = F# 07H = G 08H = G# 09H = A 0AH = Bb 0BH = B For details about the meaning see chapter 5.2.7.
3	Note Low Limit Values 00H .. 7FH	The values 00H .. 7FH represent the midi note numbers 0 .. 127. "Note Low Limit" and "Note High Limit" specify the low and high note limits for all notes in the specified part. Notes outside this range are transposed to the nearest octave within the range. For details about the meaning see chapter

4	Note High Limit Values 00H .. 7FH	The values 00H .. 7FH represent the midi note numbers 0 .. 127. For details about the meaning see chapter 5.2.6.
5	Retrigger Rule (RTR) Values 00H .. 05H	Specifies how notes held through chord changes will be handled. 00H = Stop 01H = Pitch shift 02H = Pitch shift to root 03H = Retrigger 04H = Retrigger to root 05H = Note generator For details about the meaning of these values see chapter 5.2.8.

Table 15

4.6.3.7 Ctb2 unknown data bytes

The meaning of these 7 bytes at the end of the Ctb2 data is still unknown.
Some observasions are listed about the usage in existing styles.
It seems that there are some bytes that are mainly used for drum channels. But most drum channels do NOT use these bytes, so someone can assume that they are only for extended effects.

Most styles, especially the Tyros3 styles, use the following values for these 7 bytes, for drum and non-drum channels:

> 00H 00H 00H 00H 80H 00H 00H

So this setting may be used as a default until further information is available.

Byte Index			Description
40	Values: In most cases In rare cases In very rare cases	 00H 80H 62H..7EH 83H 8AH	If this value is 80H there may be an extra break voice (like a Crash Cymbal in drum channels) for non-drum channels, when playing the 3- or 4-finger break. The extra break drum voice will sound at time 0 within the break measure. STILL UNSURE. THIS HAS TO BE VERIFIED.
41	Values: In most cases In some cases	 00H 01H	If the value is 00H the channel may be a drum channel or a non-drum channel. If the value is 01H, then the channel is always a drum channel. In this case also bytes 43, 44 and 45 have significant different values and byte 40 is always 00H.. This case seems to be the enhanced case for drum channels.
42	Values: Always	 00H	
43	Values: In most cases In some cases	 00H 18H	If the value is 18H, then the channel is always a drum channel. In this case also byte 41 is 01H and bytes 44 and 45 have significant different values.
44	Values:		If the value is not 7FH, 80H or 00H, then the

		channel is always a drum channel. In this case also byte 41 is 01H and bytes 43 and 45 have significant different values. Byte 44 has always a higher value as byte 43, except both are 00H.
	In most cases 7FH In some cases 80H In rare cases 00H In rare cases 31H...54H (only if value of byte 41 is 01H)	
45	Values: In most cases 00H In rare cases 22H...5AH (only if value of byte 41 is 01H)	If the value is not 00H, then the channel is always a drum channel. In this case also byte 41 is 01H and bytes 43 and 44 have significant different values.
46	Values: Always 00H	

Table 16

4.6.3.8   Cntt structure

Immediately after the last Ctab structure there may be one or more optional Cntt structures. Cntt structures are not used if the style file contains Ctb2 structures.

The number of Cntt structures has to be derived from the size of the CSEG data. Alternatively, the following data may be examining by looking for repeating "Cntt" section identifiers.

It seems that for (newer) instruments supporting the Cntt structure, the data in the Cntt structure overrides the data of the corresponding NTT.

The presence of Cntt data in the CASM section of a style file is incompatible with the Mixer in some (older) models, e.g. PSR 740. The Mixer (PSR 740 manual page 90) just don't work.

Details about the structure of the Cntt structure:

Byte Index		Description
0..3	byte[0] = 'C' (43H) byte[1] = 'n' (6EH) byte[2] = 't' (74H) byte[3] = 't' (74H)	This 4-character sequence identifies this section as a Cntt structure. Note: The characters are case sensitive.
4…7	Nr of data bytes = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the nr of data bytes following. This means, the first byte which is counted here is byte[8]. As the Cntt is a two byte record, this value is always = 2.
8	Source Channel Values 00H .. 0FH	Midi source channel. The values 00H .. 0FH represent the midi channels 1..16.
9	Note Transposition Table Bit 7 indicates "Bass on/off" Bits 6 .. 0 defines the table type (Bit 7 = highest bit (MSB))	00H / 80H = Bypass 01H / 81H = Melody 02H / 82H = Chord 03H / 83H = Melodic minor 04H / 84H = Melodic minor 5th Var. 05H / 85H = Harmonic minor 06H / 86H = Harmonic minor 5th Var.

		07H / 87H = Natural minor 08H / 88H = Natural minor 5th Var. 09H / 89H = Dorian 0AH / 8AH = Dorian 5th Var. If Bass is off values 00H - 0AH are used, else values 80H - 8AH. The part (channel) for which Bass is set to On recognize on-bass chords allowed in the Fingered-on-Bass fingering mode, regardless of the NTT setting. For details about the meaning of these values see chapter 5.2.5.2.

Table 17

4.7 OTS (One Touch Setting) section

The OTS is used to establish keyboard settings that can be saved and recalled from a style, and is generally used to set up the right/left voices. OTS data includes similar, but fewer, settings than a registration.

4.7.1 General

The OTS settings are stored in the OTS section of a style file in form of OTS tracks. Each OTS setting corresponds to one OTS track.

4.7.2 General structure

The OTS section consists of a section id, section length, and the OTS data. The OTS data itself consists of one or more OTS tracks. The number of OTS tracks follows from the section length and the length of the individual OTS tracks. Each OTS track also contains the information about its size.

Note: The file format allows that there may be an OTS section with no OTS tracks; the section length is 0 in this case. As it makes no sense to have an empty OTS section, programs creating/modifying styles should not create such empty OTS sections. In this case no OTS section should be created in the style file.
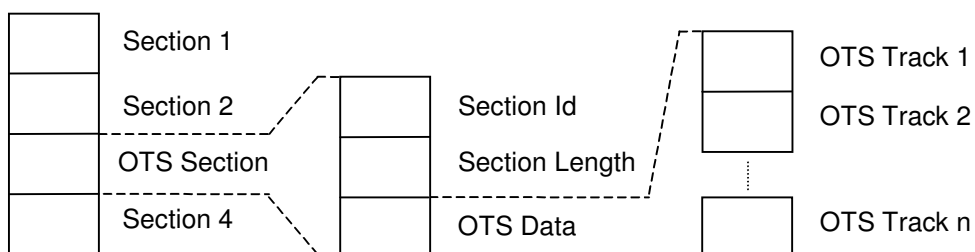
Details about the structure of the OTS section:
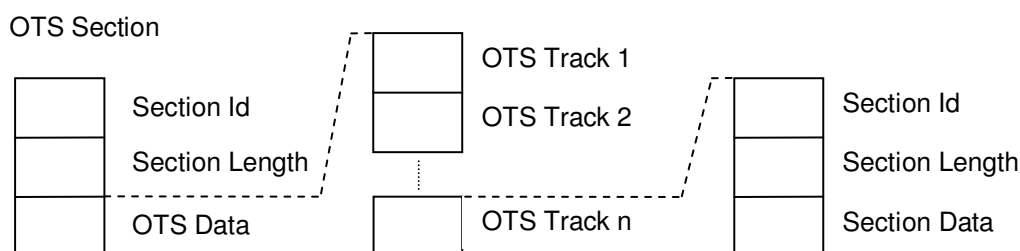
Byte		Description

Index		
0..3	byte[0] = 'O' (4FH) byte[1] = 'T' (54H) byte[2] = 'S' (53H) byte[3] = 'c' (63H)	This 4-character sequence identifies this section as an OTS section. Note: The characters are case sensitive.
4…7	Nr of data bytes = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the nr of data bytes following. This means, the first byte which is counted here is byte[8].
8..n	Data bytes (Number as given above)	OTS data. More details are described in chapter 4.7.3

Table 18

4.7.3 OTS data

The OTS data consists of one or more OTS tracks. The number of OTS tracks follows from the section length and the length of the individual OTS tracks. Therefore the number of OTS tracks is variable. Currently a maximum of 4 is used, but the file format supports any number of OTS tracks.

An OTS track is a standard midi track.

OTS Section

	OTS Track 1		Section Id
Section Id	OTS Track 2		Section Length
Section Length			Section Data
OTS Data	OTS Track n		

So the following data structure may exist multiple times inside the OTS data area shown above. This can be derived from the size of the OTS data area and the size of the individual OTS midi tracks.

Byte Index		Description
0..3	byte[0] = 'M' (4DH) byte[1] = 'T' (54H) byte[2] = 'r' (72H) byte[3] = 'k' (6BH)	This 4-character sequence identifies the following data as an OTS track, which is in standard midi track format. Note: The characters are case sensitive.
4..7	Nr of data bytes = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the nr of data bytes following. This means, the first byte which is counted here is byte[8].
8..x	Data bytes (Number as given above)	OTS (midi) track data. These data bytes form a standard midi track.

**Table 19**

4.8 MDB (Music Finder) section

The music finder function supports the musician in setting up the keyboard for playing a certain song.

When activated on the keyboard, the music finder function shows a list of song titles (also called a list of song records). For each song title there are additional fields for style, tempo and time signature. If the musician selects a song, the keyboard will load the associated style and adjusts the tempo.

There are some more fields (genre, keyword1 and keyword2), which are used by the music finder search function.

4.8.1 General

The assignment between the song title and style, tempo, time signature, is done in the style file via so-called MDB records, which are stored in the MDB section of the style file. One MDB record defines one song title and the associated information. There may be any number of MDB records in a style file.
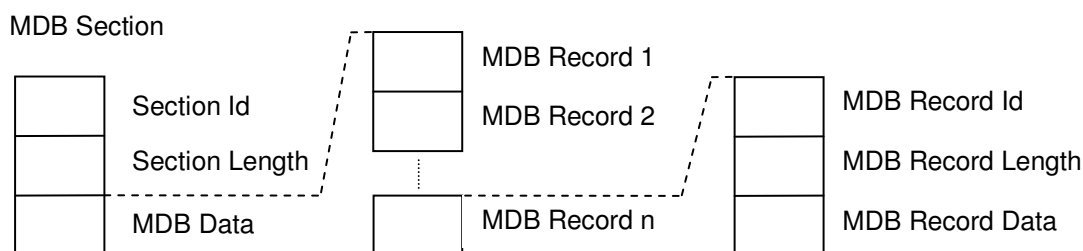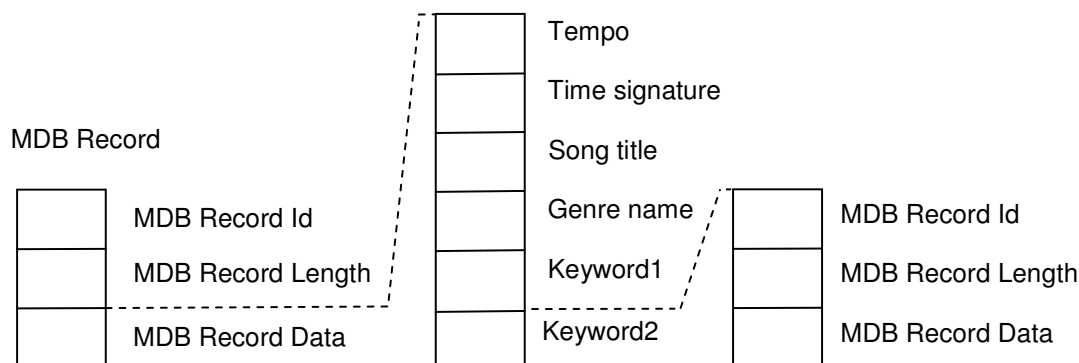

4.8.2 MDB section

Byte Index		Description
0..3	byte[0] = 'F' byte[1] = 'N' byte[2] = 'R' byte[3] = 'c'	This 4-character sequence identifies this section as a MDB section. Note: The characters are case sensitive.
4..7	Nr of data bytes = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the length of the MDB section. The first byte counted is byte[8].
8..n	Data bytes (Number as given above)	MDB data area. This area contains one or more MDB records described in chapter 4.8.3.

Table 20

4.8.3 MDB record

Byte Index		Description
0..3	byte[0] = 'F' byte[1] = 'N' byte[2] = 'R' byte[3] = 'P'	This 4-character sequence identifies the following data as a MDB record, which contains the data for one song.
4..7	Nr of data bytes = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the length (in number of bytes) of the MDB record data. The first byte counted is byte[8].
8..n	Data bytes (Number as given above)	MDB record data. More details are described in chapter 4.8.4.

Table 21

4.8.4 MDB record data

Byte Index		Description

0..2	tempo =
	    256*256*byte[0]
	+ 256*byte[1]
	+ byte[2]
		Tempo
		Unit: Microseconds per quarter note.
		Note: To calculate the tempo in BPM:
		  BPM = 60,000,000 / tempo_in_microseconds

3..4	byte[3] = beats per measure byte[4] = note that gets one beat	Time signature (e.g. for a waltz: byte[3] = 3 byte[4] = 4)
5..n	Data bytes (The number is variable)	MDB song title data record. More details are described in chapter 4.8.5.
n1..n2	Data bytes (The number is variable)	MDB genre data record. More details are described in chapter 4.8.6.
n3..n4	Data bytes (The number is variable)	MDB keyword1 data record More details are described in chapter 4.8.7
n5..n6	Data bytes (The number is variable)	MDB keyword2 data record More details are described in chapter 4.8.8.

Table 22

4.8.5 MDB song title data record

Byte Index		Description
0..3	byte[0] = 'M' byte[1] = 'n' byte[2] = 'a' byte[3] = 'm'	This 4-character sequence identifies the following data as the title of the song.
4..7	titlelength = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the length (in number of bytes) of the following character sequence. The first byte counted is byte[8].
8..n	byte[8] = 'x' byte[9] = 'x' byte[10] = 'x' : : byte[titlelength+7]	The title of the song.

Table 23

4.8.6 MDB genre title data record

Byte Index		Description
0..3	byte[0] = 'G' byte[1] = 'n' byte[2] = 'a' byte[3] = 'm'	This 4-character sequence identifies the following data as the name of the genre.
4..7	genrelength = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the length (in number of bytes) of the following character sequence. The first byte counted is byte[8].
8..n	byte[8] = 'x' byte[9] = 'x' byte[10] = 'x' : : byte[genrelength+7]	The genre of the song

Table 24

4.8.7 MDB keyword1 record

Byte Index		Description
0..3	byte[0] = 'K' byte[1] = 'w' byte[2] = 'd' byte[3] = '1'	This 4-character sequence identifies the following data as the keyword1.
4..7	keyword1length = 256*256*256*byte[4] + 256*256*byte[5]	Indicates the length (in number of bytes) of the following character sequence. The first byte counted is byte[8].

	+ 256*byte[6] + byte[7]	
8..n	byte[8] = 'x' byte[9] = 'x' byte[10] = 'x' : : byte[keyword1length+7]	The keyword1

Table 25

4.8.8 MDB keyword2 record

Byte Index		Description
0..3	byte[0] = 'K' byte[1] = 'w' byte[2] = 'd' byte[3] = '2'	This 4-character sequence identifies the following data as the keyword2.
4..7	keyword2length = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the length (in number of bytes) of the following character sequence. The first byte counted is byte[8].
8..n	byte[8] = 'x' byte[9] = 'x' byte[10] = 'x' : : byte[keyword2length+7]	The keyword2

Table 26

4.9 MH Section

This section can be found in a very small number of style files. Typically the section is at the end of the style file. Nothing is known about the purpose of this section.
(Maybe the section is related to PSR-8000 keyboards?)

Details of general structure for the MH section:

Byte Index		Description
0..3	byte[0] = 'M' (4DH) byte[1] = 'H' (48H) byte[2] = 'h' (68H) byte[3] = 'd' (64H)	This 4-character sequence identifies this section as a MH section. Note: The characters are case sensitive.
4..7	Nr of data bytes = 256*256*256*byte[4] + 256*256*byte[5] + 256*byte[6] + byte[7]	Indicates the nr of header data bytes following. This means, the first byte which is counted here is byte[8]. The length of the MHhd header is always 8 bytes.
8..15	Data bytes	MHhd header data.
16..19	byte[16] = 'M' (4DH) byte[17] = 'H' (48H) byte[18] = 't' (74H)	This 4-character sequence identifies the MHtr track.

|	byte[19] = 'r' (72H)	Note: The characters are case sensitive.
20..23	Nr of data bytes = 256*256*256*byte[20] + 256*256*byte[21] + 256*byte[22] + byte[23]	Number of bytes in the MHtr track. This means, the first byte that is counted here is byte[24].
24..n	Data bytes (Number as given above)	MHtr track data. Nothing is known about the purpose of this data.

Table 27

5 Style File Data

This chapter describes the details about the meaning, functionality and requirements about the data stored in the style file sections described in chapter 4.

5.1 *Midi section*

For the following description it is helpful to have a basic knowledge of midi and standard midi files. For some links to midi tutorials and midi specifications see chapter 7.2.

5.1.1 Meaning, Functionality and Requirements of Midi Data used in Styles

The midi section of a style consists of some initial file related data, then two initializing markers SFF1 or SFF2 and SInt used to initialize the PSR/Tyros, set up instrument voices, and the markers used to delineate the midi patterns by the selected sections (e.g. Main A, Ending B).
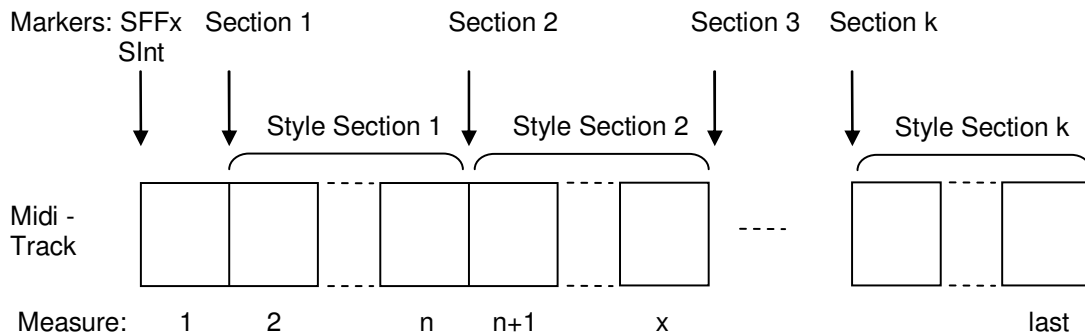
Generally a style should include at least Main A, Intro A, Ending A and Fill AA sections. The instrument will operate with less, but other users will miss these basic sections. However, all styles should have a Main A. Some instruments will not load them otherwise. Also, if a style does not have all fill sections for all contained main parts, then the Ending A may repeat (e.g. if you are playing Main B and you call Ending A, but there is no Fill BB).

Markers are case sensitive and correct spelling, including spaces, is critical.

Valid markers are:

Intro A	Main A	Fill In AA	Ending A
Intro B	Main B	Fill In BB	Ending B
Intro C	Main C	Fill In CC	Ending C

	Main D	Fill In DD	
		Fill In BA	
		Fill In AB (only for PSR 8000)	

Table 28

Note: "Fill In BA" represents the "Break" section.

Not all midi events are processed by style files in any section of the midi data. See the table below:

Event	Setup Sections (SFFx, SInt)	Other Sections
Note Off	—	OK
Note On	—	OK
Program Change	OK	OK
Pitch Bend	OK	OK
Control#0 (Bank Select MSB)	OK	OK
Control#1 (Modulation)	OK	OK
Control#6 (Data Entry MSB)	OK	—
Control#7 (Master Volume)	OK	OK
Control#10 (Panpot)	OK	OK
Control#11 (Expression)	OK	OK
Control#32 (Bank Select LSB)	OK	OK
Control#38 (Data Entry LSB)	OK	—
Control#71 (Harmonic Content)	OK	OK
Control#72 (Release Time)	OK	—
Control#73 (Attack Time)	OK	—
Control#74 (Brightness)	OK	OK
Control#84 (Portamento Control)	—	OK
Control#91 (Reverb Send Level)	OK	OK
Control#93 (Chorus Send Level)	OK	OK
Control#94 (Variation Send Level)	OK	OK
Control#98 (NRPN LSB)	OK	—
Control#99 (NRPN MSB)	OK	—
Control#100 (RPN LSB)	OK	—
Control#101 (RPN MSB)	OK	—

Table 29

5.1.1.1 Measure 1

The following midi data has to be completed in the first measure of the midi data. Usually all events are on measure 1, beat 1, tick 0 (1:01:000). It is important that they are located in the file in the sequence as mentioned below.

Initial data: The first commands after the midi track header are usually time signature, tempo and copyright (optional). Time Signature is used to determine the metronome behavior and perhaps the score display; its value does not affect the play back of the note events. This is determined by the time values associated with the note on-off events. The tempo sets the default tempo of the instrument.

SFF1 or SFF2: This marker must come before the SInt marker. It is followed by the StyleName, which is a Meta Event identified by ID=3 (see Table 6). The length of meta text events (except copyright) usually is limited in practice to a size which fits in a PSR display field. In factory styles, StyleName is generally followed by sysex events that define the style (see Table 30). The importance of these sysex is not understood.

Sysex Event	Description
F0 43 76 1A 10 00 01 01 01 00 01 00 00 F7	XGWorks Style code

F0 43 73 39 11 00 46 00 F7	Clavinova function
F0 43 73 01 51 05 00 01 08 00 00 00 00 00 00 00 00 00 F7	Clavinova function
F0 43 73 01 51 05 00 02 08 00 00 00 00 00 00 00 00 00 F7	Clavinova function

Table 30

SInt: The SInt marker must be after the above data and is generally followed by Midi On, Controller and Program Change Midi Events necessary to initialize the midi channels and sysex to set up the DSP:

Command	Description
F0 7E 7F 09 01 F7	Midi On sysex
BX, 7, volume data (0-127)	Control Change Volume
BX, 91, level data (0-127)	Control Change Reverb Send Level
BX, 93, level data (0-127)	Control Change Chorus Send Level
BX, 0, MSB value (0-127)	Control Change Bank Select MSB
BX, 32, LSB value (0-127)	Control Change Bank Select LSB
CX, program change number (0-127)	Program Change; Note: For XG voices to be properly recognized, the program change must be preceded by MSB & LSB Bank Select Events.
F0 43 10 4C 02 01 00 dd dd F7	Reverb Type (dd from Effects List)
F0 43 10 4C 02 01 20 dd dd F7	Chorus Type (dd from Effects List)
F0 43 10 4C 02 01 40 dd dd F7	Variation Type (dd from Effects List)
F0 43 10 4C 02 01 5A tt F7	Variation Connection Type; for styles, tt = 01 (system)
F0 43 10 4C 03 00 00 tt tt F7	DSP 2 Effect Type (tt = type)
F0 43 10 4C 03 01 00 tt tt F7	DSP 3 Effect Type (tt = type)
F0 43 10 4C 03 00 pp xx F7	DSP 2 Parameter (pp parameter, xx = value)
F0 43 10 4C 03 01 pp xx F7	DSP 3 Parameter (pp parameter, xx = value)
F0 43 10 4C 08 08 07 03 F7	Channel 9 assigned to drums 1
F0 43 10 4C 08 09 07 02 F7	Channel 10 assigned to drums 2
F0 43 10 4C 08 ch 72 xx F7	EQ Bass Gain (ch = channel, xx = value)
F0 43 10 4C 08 ch 73 xx F7	EQ Treble Gain (ch = channel, xx = value)
F0 43 10 4C 08 ch 76 xx F7	EQ Bass Frequency (ch = channel, xx = value)
F0 43 10 4C 08 ch 77 xx F7	EQ Treble Frequency (ch = channel, xx = value)
F0 43 10 4C 30 nn dd xx F7	CH10 Drum Edit Note (nn = note, dd = from MIDI Parameter Change table DRUM SETUP, xx = value)

Table 31

Any following sections use these definitions until they are overwritten. E.g.; if channel 1 has a program change in SInt and Main C, then channel 1 will use the SInt definitions until Main C is played. Thereafter channel 1 will use the new definition. For this reason, if any changes are made in a section following SInt, then all sections should have commands which establish these parameters.

Newer instruments (e.g. Tyros 4 and 5) behave different than stated above. The SInt section is executed every time a style section is selected. The effect of this is that the default is used in all sections that do not have data at the beginning of the section.

5.1.1.2   Measure 2 and following measures

Measures 2 and following contain the musical patterns of the style.

Intros, Mains, Endings, Fill Ins, Break: These can be in any order, and should begin in measure 2. Primarily midi events include note-on and note off, controller (e.g. expression),

pitch wheel and program change events. Fill Ins and Break are limited to a single measure; other sections can be any length but are generally >= 4 measures.

5.1.2 Midi Channel Usage

The accompaniment system supports the following midi channels for accompaniment input:

Acc. Channel	Acc. Part	Usage
9	Sub-Rhythm	Secondary percussion instruments.
10	Rhythm	Main percussion instruments.
11	Bass	Main bass instrument.
12	Chord 1	Often used for rhythm guitar.
13	Chord 2	Often used for piano.
14	Pad	Often used for violins or similar.
15	Phrase 1	Often used for brass instruments.
16	Phrase 2	Often used for brass instruments.

Table 32

The midi section may either follow these channel assignment, or a CASM section has to be added to the style. A CASM section allows mapping any midi channel to the required accompaniment channels (see chapter 5.2).

5.1.3 Key and allowed Notes

During playback the accompaniment system transposes the musical sections of the style according to the chord currently played by the musician. For this the instrument needs to know the original key of the musical section. If nothing is stated the instrument assumes the key CMaj7. If the key of the musical section is not CMaj7, a CASM section has to be added to the style. A CASM section allows using any key (see chapter 5.2.2). If the contents of a midi channel are not based on CMaj7 this information has to be specified in the elements "Source Chord" and "Source Chord Type" of Table 12.

There are restrictions about the notes which are allowed to be used in the style. Using other notes may lead to a wrong transposition.

The allowed notes are:

Acc. Channel	Acc. Part	Allowed notes (based on CMaj7)
9	Sub-Rhythm	All
10	Rhythm	All
11	Bass	C, D, E, G, A, B
12	Chord 1	C, E, G, B
13	Chord 2	C, E, G, B
14	Pad	C, E, G, B
15	Phrase 1	C, D, E, G, A, B
16	Phrase 2	C, D, E, G, A, B

Table 33

For Intro and Ending sections autonomously playing all chord changes, which means that for these sections no transposition will be performed (selected in the corresponding NTT, see chapter 5.2.5.2), all notes are allowed also in acc. channels 11 .. 16 of these sections.

5.1.4 Voices

Styles may use any factory voices, except SA and organ flute voices. User voices made by editing preset voices cannot be used. In the Tyros 2, Custom Voices are allowed. (Note: SA voices can be assigned using PC programs; it is not known if this has any negative consequences.)

Voices are identified by two values. The bank and the program (= instrument) number. The bank number defines a group of instruments; the program number identifies the instrument inside this group. The bank number is usually given as a two byte value MSB (Most Significant Byte) and LSB (Least Significant Byte).

Currently the following voices are known:

Voice Type	Voice Bank (MSB)	Voice Bank (LSB)	Program Nr.
GM	0	0	0 .. 127
XG	0	0 .. 101	0 .. 127
Panel	0	102 .. 127	0 .. 127
Mega, Super Articulation, Ensemble	8, 9, 109	0 .. 127	0 .. 127
Organ	10		
Expansion Memory	63		
Sound Effects	64	0	0 .. 127
New	104		
GM2 Drums	120	0	0 .. 127
GM2	121	0 .. 127	0 .. 127
Drum and Percussion, Special Effects	126 .. 127	0	0 .. 127

Table 34

When using these voices it has to be taken into account that not all voices are available on all keyboard models. If a style uses a voice that is not supported by the keyboard, and the MSB Bank Select value is zero, the instrument will automatically substitute the voice. Therefore the style will work, but it will sound (maybe only slightly) different. The result of this substitution may sound acceptable (e.g. when substituting XG and GM voices) or unusable (e.g. when substituting Mega voices with GM voices).

There are software programs available, which can substitute voices in styles in a more sophisticated way, to adapt them for various keyboard models. See chapter 7.1.

A similar problem may arise when the style uses a drum kit, which is not available on the keyboard model. Then wrong percussion instruments may be used.

Currently no software program is known, which supports total (MidiPlayer does Latin Kits) remapping of drum kits, so this has to be done manually. MixMaster has a Drum Edit View which will edit the voice and the drum note used by the pattern.

5.1.5 Special Effects

5.1.5.1 Half Bar Fill Ins

The Fill In section of a style has to be one measure in length. If a style should have a Fill In section that is only a half measure in length, this is not possible in the common way, but by a trick. For the Fill In measure the tempo is doubled (therefore the measure plays in ½ of the time of an original measure) and all note event durations in this measure are doubled (so that the notes and rests have again the original duration).

This modification is supported by the programs "StyleAdjust" and "Style Half Bar Fill Creator" (see chapter 7.1.3).

5.1.5.2 Multiple Time Signatures

In general styles can only have one time signature. But, by the same trick used for the "Half Bar Fill Ins", this can also be achieved by increasing/decreasing the tempo and decreasing/increasing the note event durations in the opposite direction.

This modification is supported by the program "Style Tempo Editor" (see chapter 7.1.3).

5.1.6 Restrictions for older keyboard models

This chapter list restrictions for older keyboard models, which are not mentioned before.

Midi file resolution: Is limited to max. 480 ticks per quarter note for e.g. PSR7000 and PSR8000 (?).

5.2 CASM section

This chapter will provide information which is useful when creating or modifying CASM data using one of the CASM editors or style creation programs.

The CASM section gives instructions to the instrument on how to deal with the midi data.
It provides additional possibilities to create more diversified styles.

The CASM section allows defining separate instructions for each style part (e.g. Intro A, Main B) of each source midi channel. To reduce the complexity, some CASM editor programs may only support defining separate instructions for each source midi channel, treating all style parts the same. However, these sections can be easily expanded as required.

5.2.1 Styles without a CASM Section

Style files do not necessarily need a CASM section, as long as the midi section of the style follows these rules:

- Only midi channels 9 .. 16 are used.
- Usage of midi channels is according to Table 32.
- Midi channels have to be based on key CMaj7.
- There is only one channel per part (i.e. no separate channel for major and minor chord types)

5.2.2 Midi channel usage and assignment

The accompaniment system supports 8 midi channels (accompaniment channels) as listed in Table 32.

If no CASM section is present in the style file, then the accompaniment system assumes that the midi part of the style file only uses these midi channels, with the assignment according to Table 32.

A CASM section instead allows using up to 16 midi channels in the midi part of the style file. These "source channels" are then assigned to the 8 "accompaniment channels". In the instrument, any source channel can be assigned to any accompaniment channel, with the restriction that drum voices can only be assigned to the Rhythm channel (10); the SubRhythm channel can accept both. Multiple source channels can be assigned to a single accompaniment channel. In PC programs, any instrument can be assigned to any channel.

In general, any voice non-drum channel can be assigned to any of the accompaniment channels 11 .. 16, as it seems that they work quite equally. E.g. a source channel containing

chord data may be assigned to accompaniment channel 11, which is identified as the Bass part on the instrument. Since the instrument display always calls channel 11 "Bass", it would be confusing if the channel does not contain the bass pattern.

The assignment of the channels is done using the structure described in Table 12.

When assigning the channels the restrictions regarding the allowed notes have to be followed according to Table 33.

This provides e.g. the following possibilities:
- One accompaniment channel can play more than one instrument depending upon the chord type, if source tracks redefine the voice.
- One accompaniment channel can play different instruments in style sections by redefining the voice within the style section.

5.2.3 Chord Mute

Normally the accompaniment is played by the instrument continuously until it is stopped or an ending part has been finished. The "Chord Mute" allows the style to define chord types, which when played, temporarily mute the accompaniment contributed by that track. Use chord type "cancel" to mute all instruments.

This can be specified for each style part of each source midi channel.

Valid chord types are:

Maj	min6	dim7	7(#9)
Maj6	min7	7th	Maj7aug
Maj7	min7b5	7sus4	7aug
Maj7#11	min(9)	7b5	1+8
Maj(9)	min7(9)	7(9)	1+5
Maj7(9)	min7(11)	7#11	sus4
Maj6(9)	minMaj7	7(13)	1+2+5
Aug	minMaj7(9)	7(b9)	cancel
Min	dim	7(b13)	

Table 35

This provides e.g. the following possibilities:
- Different patterns play in response to chord types (Maj, min, …).
- The whole non rhythm accompaniment can be muted by keying a cancel chord (equal to three consecutive keys in Fingered Mode).
- The whole non rhythm accompaniment excluding the bass can be muted by keying a cancel chord (equal to four consecutive keys in Fingered Mode).
- Dedicated source channels can be muted to disable a voice at certain times during playback by keying a dedicated chord.
- Dedicated source channels can be unmuted to enable a voice at certain times during playback by keying a dedicated chord.

5.2.4 Note Mute

The "Note Mute" allows the style to mute (or enable) a track based upon the Root Chord.
E.g. when C is disabled for a bass track, then keying chords C, Cm etc. mutes the bass track.

5.2.5 Note Transposition

Depending on the chord currently played by the musician, the accompaniment system of the instrument has to calculate which notes should sound, based on the notes given in the corresponding midi track.

There are two parameters which control this calculation, the "Note Transposition Rule" and the "NoteTransposition Table". Each source channel can have a different combination of these parameters.

5.2.5.1   Note Transposition Rule (NTR)

This specifies what notes of the possible chord notes will be played when a chord is transposed to a new key. There are two modes available:

- Root Transposition:
  When transposed the pitch relationship between notes is maintained, i.e. the same inversion of the chord is used. For example, the notes C3, E3, and G3 in the key of C will become F3, A3, and C4 when transposed to F.

  This setting is used for parts that contain melodic lines and the bass part.

- Root Fixed:
  The note is kept as close as possible to the previous note range, i.e. a different inversion of the chord may be used. For example, the notes C3, E3, and G3 in the key of C will become C3, F3, and A3 when transposed to F.

  This setting is use for chordal parts, e.g. for rhythm guitar parts.

- Guitar: (only available in SFF2 styles)
  The notes are transposed to match the chords as played on a guitar, i.e. a different inversion of the chord may be used.

  This setting is only used for guitar parts.

If NTR is "Guitar" the following apply. In contrast to other NTRs there is no harmonic relation between source and target notes. Each source note is mapped to one of the guitar strings. The pitch or harmonic function will be irrelevant. The mapping of source notes to guitar strings is as follows:

B	-> 1st string (high E)
A	-> 2nd string (B)
G	-> 3rd string (G)
F	-> 4th string (D)
E	-> 5th string (A)
D	-> 6th string (low E)
C#	-> a quint above/below
C	-> root note

That means you can control exactly which of the six strings should sound at what time. C and C# will be mapped to the root of an on-bass chord, if parameter BASS is set to on. It is recommended not to use C, C#, D and E at the same time.

If source notes will be moved by an octave this does not mean that the chord will sound an octave lower/higher. Rather you can control which chord position on the fretboard will be used:

C2 - B2	-> 1st position
C3 - B3	-> 2nd position
C4 - B4	-> 3rd position

C5 - B5   -> 4th position

5.2.5.2 Note Transposition Table (NTT)

The note transposition table specifies the method to be used for source pattern transposition, i.e. how the source cord type (e.g. Maj7) is transposed to the destination chord type (e.g. min6).

If NTR is not "Guitar" the following apply:

- Bypass: No transposition.
  No transposition. Playback is independent of the specific chord type used during playback.
  This has to be used for drum channels (as these notes should never be transposed) and for other special effects sounds. Used also for Intros and Endings if they already contain chord progressions.

- Melody:
  Should be used for melodic channels that are assigned to accompaniment channels "Phrase 1" and "Phrase 2".

- Chord:
  Should be used for chord oriented channels that are assigned to accompaniment channels "Chord 1" and "Chord 2" when they contain piano or guitar-like chordal parts.

- Bass:
  Should be used for bass channels that are assigned to accompaniment channel "Bass".
  In newer instruments this is replaced by NTT Melody with the option "Bass On" selected.

- Melodic Minor:
  Should be used for melodic channels that are assigned to accompaniment channels "Phrase 1" and "Phrase 2" for style parts where only major/minor chords are played, e.g. Intros and Endings.
  Lowers the third scale degree by a semitone when the played chord changes from a major to a minor chord, or raises the minor third scale degree a semitone when changing from a minor to a major chord. Other notes are not changed.

- Harmonic Minor:
  Should be used for chord oriented channels that are assigned to accompaniment channels "Chord 1" and "Chord 2" for style parts where only major/minor chords are played, e.g. Intros and Endings.
  Lowers the third and sixth scale degrees by a semitone when changing from a major to a minor chord, or raises the minor third and flatted sixth scale degrees a semitone when changing from a minor to a major chord. Other notes are not changed.

- Harmonic minor 5th Var:
  (only available if the "Cntt" structure in the style file is supported.)
  Mainly used in "Session Styles".

- Natural minor:
  (only available if the "Cntt" structure in the style file is supported.)
  Mainly used in "Session Styles".

- Natural minor 5th Var:
  (only available if the "Cntt" structure in the style file is supported.)
  Mainly used in "Session Styles".

- Dorian minor:
  (only available if the "Cntt" structure in the style file is supported.)

Mainly used in "Session Styles".

- Dorian minor 5th Var:
  (only available if the "Cntt" structure in the style file is supported.)
  Mainly used in "Session Styles".

If NTR is "Guitar" the following apply:

- All-Purpose:
  Should be used if the accompaniment is a mixture of "Stroke" and "Arpeggio".
  .
- Stroke
  Should be used for chord oriented channels.

- Arpeggio
  Should be used for finger picking oriented channels

5.2.5.3 Typical settings for note transposition parameters

Below are some typical settings for NTR and NTT and their use cases.

NTR	NTT	Usage
Root Fixed	Bypass	Drum channels.
Root Trans	Bypass	Intros and Endings already containing chord progressions.
Root Fixed	Melody	Monophonic channels.
Root Trans	Melody	Melodic channels that are assigned to accompaniment channels "Phrase 1" and "Phrase 2".
Root Fixed	Chord	Chord oriented channels that are assigned to accompaniment channels "Chord 1" and "Chord 2" when they contain piano or guitar-like chordal parts.
Root Trans	Chord	
Root Trans	Bass or Melody + Bass On	Bass channels that are assigned to accompaniment channel "Bass".
Root Trans	Melodic Minor	Melodic channels that are assigned to accompaniment channels "Phrase 1" and "Phrase 2" for style parts where only major/minor chords are played, e.g. Intros and Endings.
Root Trans	Harmonic Minor	Chord oriented channels that are assigned to accompaniment channels "Chord 1" and "Chord 2" for style parts where only major/minor chords are played, e.g. Intros and Endings.

Table 36

5.2.5.4 Recommendations when using NTR ROOT FIXED with NTT CHORD

Reinhold Pöhnl gives in his book "Styles & Patterns" (page 81, for a full reference see chapter 8) some recommendations for using the correct notes in the source pattern.
The rest of this chapter is a translation from the German original.

For the source pattern always the "most important" notes of the chord should be used!
In case of SOURCE ROOT C and SOURCE CHORRD Maj7 these are: E, G and B.

What are the "most important" notes of a chord? The quotes indicate that there is something special. In contrast to classical music theory the accompaniment system appraises the notes of the source pattern this way:

In all three note chords the "most important" notes are the three notes of the chord. This sounds trivial - and is trivial!
But then it continues different:
In all four note chords the "most important" notes are the (three) notes without the root note! And here accompaniment system appraises differently than classical music theory. Normally the fifth is omitted first, but the accompaniment system first leaves out the root note.

The three "most important" notes of a five note chord are the (three) notes without the root note and the fifth.

Here are some source chords and their "most important" notes.
The remaining chord notes are in brackets.

SOURCE C   : C, E, G
SOURCE C7   : E, G, Bb, (C)
SOURCE Cmaj7   : E, G, B, (C)
SOURCE Cm11   : F, Bb, Eb, (C), (G)
SOURCE C6_9   : E, A, D, (C), (G)
SOURCE Cm   : C, Eb, G
SOURCE Cm7   : Eb, G, Bb, (C)

For NTR ROOT FIXED with NTT CHORD the following applies for all source patterns with at least three note chords:

The "most important" notes of the chord in the source pattern will be the "most important" notes of the destination chord. The source notes are mapped to different destination notes (without double notes!), all "most important" notes of the chord are played.

For five note chords, like Min11, additionally the following applies:
The root note and fifth of the source chord will always be mapped to the root note and fifth of the destination chord, never to any other note like third or seventh.

5.2.6 Note Limits

The values 00H .. 7FH represent the midi note numbers 0 .. 127.

"Note Low Limit" and "Note High Limit" specify the low and high note limits for all notes played in the specified part. If a transposed note is outside this range, then the note is transposed to the nearest octave within the range. The range must be at least one octave.

This can be used to ensure that only notes are played that are in the range of the respective instrument.

Example: When LOW = C3 and HIGH = D4
Root Motion: C C# D#
Notes Produced: E3-G3-C4 / F3-G#3-C#4 / D#3-G3-A#3

5.2.7 High Key

Specifies the upper root limit. Chords with a root higher than the specified limit will be played in the octave immediately below the high-key limit. This setting is effective only when the NTR (Note Transposition Rule) is set to "Root Trans".

This is used to keep bass lines to a bass range.

Example: When HIGH KEY = F.
Root Motion: C C# D F F#
Notes Produced: C3-E3-G3 / C#3-F3-G#3 / D3-F#3-A3 / F3-A3-C4 / F#2-A#2-C#3

5.2.8 Retrigger Rule (RTR)

Specifies how notes behave through chord changes.

- Stop:
  The note is stopped.
  (Rarely used.)

- Pitch shift:
  The pitch of the note will bend without attack to match the type of the new chord.
  (Common for most tracks.)

- Pitch shift to root:
  The pitch of the note will bend without attack to match the root of the new chord.
  (Common for bass track.)

- Retrigger:
  The note is retriggered with attack at a new pitch matching the new chord type.
  (Only for special use.)

- Retrigger to root:
  The note is retriggered with attack at a new pitch matching the new chord root.
  (Only for special use.)

- Note generator:
  This setting will only be available if programmed in the original style. A designated note is produced with designated pitch, length, and velocity matching the new chord.

5.2.9 Special Features

At the end of the Ctab structure special features can be defined for a midi source channel. Currently there is only one special feature ("extra break drum voice") used by a very small number of styles.

An extra break drum voice (e.g. a Crash Cymbal) can be added when playing the 3- or 4-finger break. The extra break drum voice will sound at time 0 within the break measure.

For this purpose there must be created a MIDI channel in the MIDI part of the style file with only the Drum Set definition (Program Change, MSB and LSB); and NO notes. This (almost empty) channel must be redirected to a Keyboard Drum Channel (Rhythm Sub or Rhythm Main). Furthermore the extra drum voice and its volume can be defined.

For this MIDI channel a "normal" Ctab structure with the following exceptions must be created:

Byte Index	Description	Value
9	Destination channel	08H = Sub Rhythm or 09H = Rhythm

11	Note Mute (Part 1)	0FH
12	Note Mute (Part 2)	FFH
13	Chord Mute (Part 1)	04H
14	Chord Mute (Part 2)	00H
15	Chord Mute (Part 3)	00H
16	Chord Mute (Part 4)	00H
17	Chord Mute (Part 5)	00H
18	Source Chord	00H
19	Source Chord Type	00H
20	Note Transposition Rule (NTR)	01H
21	Note Transposition Table	00H
22	High Key	00H
23	Note Low Limit	00H
24	Note High Limit	7FH
25	Retrigger Rule (RTR)	03H
26	Special feature id	01H = "extra break drum voice"
27	Meaning unknown	00H
28	Meaning unknown	18H
29	Instrument	Typical instruments are: 31H = Crash Cymbal 1 32H = Crash Cymbal 2
30	Volume	00H .. 7FH (0 .. 127)

Table 37

6 Credits

The authors of this document wish to express their gratitude to all the members of the PSR community who have shared their knowledge and experience on style making and style files, and especially to some of the early pioneers in style file programming: Jørgen Sørensen, Dan Phalen, Heiko Plate and Evgeny Osenenko.

7 Links

7.1 Software

This chapter provides links to free software. The list may not be complete.

7.1.1 General Style Software

Midi and Style Player (by Jørgen Sørensen)
(Note: A standard midi player will not produce an acceptable result for styles.)
http://www.jososoft.dk/yamaha/software.htm

MidiPlayer (by Michael P. Bedesem)
Supports playing midi, style, voice, multipad, wav and mp3 files on a PC. Views all internals of a style file in detail..
(Note: A standard midi player will not produce an acceptable result for styles.)
http://psrtutorial.com/MB/midiplayer.html

StyleDump (by Michael P. Bedesem)
Views all internals of a style file in detail. (Unlike MidiPlayer, StylePlayer does not convert or modify the file in any way.)
http://psrtutorial.com/MB/styledump.html

StylePlayer (by Michael P. Bedesem)
Supports playing style files on a PC. Similar to MidiPlayer but more focused on style playing.
(Note: A standard midi player will not produce an acceptable result for styles.)
http://psrtutorial.com/MB/styleplayer.html

7.1.2 Style Adaptation Software

This section lists software which may be used if a style should only be adapted to a different keyboard model.

MidiPlayer (by Michael P. Bedesem)
Supports various functions to convert styles for the different PSR instruments.
http://psrtutorial.com/MB/midiplayer.html

Mix Master (by Michael P. Bedesem)
Supports various functions to convert styles for the different PSR instruments.
http://psrtutorial.com/MB/mixMaster.html

Style Old Format Converter (by Jørgen Sørensen)
Converts styles for use on older/smaller keyboards.
http://www.jososoft.dk/yamaha/software.htm

Style Revoicer (by Jørgen Sørensen)
Change voices and parameters, transpose etc. in style files.
http://www.jososoft.dk/yamaha/software.htm

7.1.3 Style Modification Software

This section lists software which may be used to modify a style in various ways.

Style Creator (in the PSR/Tyros)

CasmEdit (by Evgeny Osenenko)
Manipulate style files and create and modify CASM sections.
http://www.mnppsaturn.ru/osenenko/Main_eng.htm

CASM Editor
Displays and edits the CASM section.
http://www.jososoft.dk/yamaha/software.htm

MDB Editor (by Jørgen Sørensen)
Add, edit or delete records in the MDB section in style files.
http://www.jososoft.dk/yamaha/software.htm

MidiPlayer (by Michael P. Bedesem)
Edit channel transpose, OTS and individual (vs measure) voice, volume, pan, tempo, and effect events. Converts styles (voices, volumes, OTS) from one instrument to another (including MegaVoice to standard voice) .
 http://psrtutorial.com/MB/midiplayer.html

Mix Master (by Michael P. Bedesem)
Supports editing of patterns (note/velocity/length/time), channel assignments, voices, effects, controllers, user sysex, notes, drum kits, drum kit & voice assignment, drum note characteristics (vol, pan, effect send, pitch, LPF, EG Rate), quantize, fade In/Out, Retardando

Casm viewing but no editing functions. Does not destroy the Casm or other style sections when file is saved.
http://psrtutorial.com/MB/mixMaster.html

OTS Editor (by Jørgen Sørensen)
Edit all parameters in OTS sections.
http://www.jososoft.dk/yamaha/software.htm

PsrStyleDatabase (by Peter Wierzba)
Can also add, edit or delete records in the MDB section in style files.
http://www.wierzba.homepage.t-online.de/

StyleAdjust (by Michael P. Bedesem)
This program permits unlimited experimentation with the Tempo and note duration of a style
http://psrtutorial.com/MB/bedesem.html (available by email request to mpb@sover.net)

StyleEdit (by Michael P. Bedesem)
Edit internal stylename, tempo, copyright info and main section order for a PSR style.
http://psrtutorial.com/MB/bedesem.html (available by email request to mpb@sover.net)

StyleFix (by Michael P. Bedesem)
Convert internal names from numbers to names, replace bass voices for older instruments, exchange CD for AB sections, add a Break, remove OTS/MF Database sections.
http://psrtutorial.com/MB/bedesem.html (available by email request to mpb@sover.net)

Style Half Bar Fill Creator (by Jørgen Sørensen)
Create fill in's and breaks of half the normal bar length, e.g. a 2/4 fill in a 4/4 style file.
http://www.jososoft.dk/yamaha/software.htm

StyleMaker (by Michael P. Bedesem)
This program supports editing an existing style file in a sequencer, or creating a PSR style from a midi song file, or midi/style templates (including Band In The Box styles).
http://psrtutorial.com/MB/bedesem.html (No longer supported; available by email request to mpb@sover.net)

Style ReMixer (by Jørgen Sørensen)
Remix style parts in styles.
http://www.jososoft.dk/yamaha/software.htm

Style Revoicer (by Jørgen Sørensen)
Change voices and parameters, transpose etc. in style files.
http://www.jososoft.dk/yamaha/software.htm

Style Split and Splice (by Jørgen Sørensen)
Split and splice style files for editing the midi part of the style file with sequencer software.
http://www.jososoft.dk/yamaha/software.htm

Style Tempo Editor (by Jørgen Sørensen)
Set individual tempo in style parts.
http://www.jososoft.dk/yamaha/software.htm

Style Time and Tempo in Name (by Jørgen Sørensen)
Add time signature and tempo values in the style file name.
http://www.jososoft.dk/yamaha/software.htm

Style Time Editor (by Jørgen Sørensen)
Set individual time signatures in style parts.
http://www.jososoft.dk/yamaha/software.htm

Visual Styler (by MojoFlux)
Copy style parts from more style files to one single style file.
http://www.crestonhall.com/music/vs.php

7.1.4 Style Making Software

This section lists software which may be used to create styles from scratch or from midi files.

Style Creator (in the PSR/Tyros)

Midi2style (by Jørgen Sørensen)
For creating style files from midi files.
http://www.jososoft.dk/yamaha/software.htm

StyleMaker (by Michael P. Bedesem)
http://psrtutorial.com/MB/bedesem.html

One Man Band
http://www.1manband.nl/omb.htm

StyleMagic YA
http://www.midisoft.pl/en

Style Works XT
http://www.emc-musicsoftware.com

7.2 Midi Specification and Tutorials

This chapter provides links to documents explaining the MIDI system.

http://home.roadrunner.com/~jgglatt/tech/midispec.htm

http://home.roadrunner.com/~jgglatt/tech/midifile.htm

http://www.ibiblio.org/emusic-l/info-docs-FAQs/MIDI-doc/index.html

http://www.jososoft.dk/yamaha/articles.htm

http://www.midi.org/about-midi/smf/rp017.shtml

7.3 Style Creation Tutorials

This chapter provides links to information covering also the musical aspects of style making.

Style Creation Course (by Jørgen Sørensen)
http://www.jososoft.dk/yamaha/articles.htm

How to make style on a keyboard (Simon Williams)
http://psrtutorial.com/lessons/faq/PSRFAQ.htm (see topic B. 18.)

Style Creation Workshop (by Heidrun Dolde) (only available in German)
http://heidruns-musikerseiten.de/tyros/workshop.html

8 References

1. Yamaha Manual Library
   http://www.yamaha.co.jp/manual/english/index.php

2. Pöhnl, Reinhold: *"Styles & Patterns"*. PPVMEDIEN GmbH, Bergkirchen 2003,
   ISBN: 978-3-932275 (only available in German)

3. Sørensen, Jørgen: Various articles about styles
   http://www.jososoft.dk/yamaha/articles.htm

4. Michael P. Bedesem: Frequently Asked Questions for PSRs, CVPs & Tyros
   http://psrtutorial.com/lessons/faq/PSRFAQ.htm

9 Disclaimer

Copyright Status: The information in this document is not copyrighted by the authors, and is provided only for the personal educational use of the readers.

This document contains some information that has been obtained from Internet sites, product manuals, and public/private communication between individuals and where such use has not always been specifically authorized. The authors of this document are making such material available in our efforts to advance the understanding of how midi, style and related files structures are defined, and how they can be applied by readers in creating files for their own personal use. The authors believe this constitutes a 'fair use' of any such copyrighted material as provided for in section 107 of the US Copyright Law. If readers wish to use material from this document for purposes that go beyond 'fair use', they must obtain permission from any applicable copyright owner.

In accordance with Title 17 U.S.C. Section 107, the material in this paper is distributed without profit to those who have expressed a prior interest in receiving the included information for research and educational purposes.

Liability: The authors do not warrant or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, product, or process described in this document. Various content has been obtained from information published on the internet and may be subject to copyright by original authors. Use of any such copyrighted material is subject to the terms and conditions of use established by copyright holder.

Endorsement: Do not endorse or recommend any software products to the exclusion of any others. The authors have identified software products that they are familiar with and acknowledge that there may be many others available. Interested parties are encouraged to suggest others for further editions of this document.

Links: This document provides links to Internet sites for the convenience of the readers. The authors are not responsible for the availability or content of these sites. It is the responsibility of the user to examine the copyright and licensing restrictions on any such sites and to secure all necessary permissions.

Trademarks: Yamaha is a registered trademark of Yamaha Corp. Windows is a registered trademark of Microsoft Corporation. All other trademarks are acknowledged.

Affiliation: Please note that the authors are not associated in any way with Yamaha.

10 Assistance by Readers

Assembling and maintaining information on styles, their creation, editing and use is a challenge in the absence of formal documentation. This especially so when new capabilities and settings are regularly introduced with new files and instruments.

The authors urge readers to provide feedback on errors in this document, discoveries that they make, or other information they may happen upon while creating or editing these important files.

All contributions are welcomed and will be acknowledged in subsequent updates with the contributor's permission.

11 Open Issues and Questions

Following is a list of various questions. If anybody can contribute any answers or hints, please contact the authors to help making this document more complete.

1. Are there differences concerning the contents of the style file types (.sty, .bcs, .prs,…) ? None have been noticed so far.

2. How to work with chord and scale tone rules: While we are generally familiar with the chord and scale rules, we really do not understand them and how they interact with the CASM settings. There are some styles where a D in a CMaj7 pattern works just fine and others where it does not. And what to do when you really want an F in the pattern?

3. CASM Settings: How do the various transposition settings change the reproduction?

4. Session Styles: How to make them?

5. Is the "Note Mute" in the Ctab practically used? What possibilities are provided by the "Note Mute" functionality?

6. Meaning (and naming) of Bit 2 and Bit 3 in "Chord Mute (Part 1)" in the Ctab.

7. Full meaning of Cntt structures and how they compete with NTT.
    (Maybe the Cntt section has been introduced for compatibility reasons? I. e. older PSRs may be programmed in such a manner that they did not allow to increase the set of NTT values in the Ctab section? If so, then one would expect that on the newer instruments the Cntt value would always override the value in the Ctab section, which seems to be the case.)

8. How many Cntt structures are commonly used compared to the number of Ctab structures?

9. Some more details about the note transposition table described in 5.2.5.2. would be interesting.

10. What NTT settings are practically used?

11. Intros can be setup so that they follow the chord changes of the left hand, and so that they autonomously change the chords. How is this programmed in the style? With NTT = Bypass?

12. If Intro and Ending sections autonomously play all chord changes, then are there any notes allowed in these parts? Not only the restricted set (transposed to the current played chord) listed in Table 33?

13. How to prepare a style so that it can be used in XGWorks?

A. Decimal-Hexadecimal Conversion Table

Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex
0	00	32	20	64	40	96	60	128	80	160	A0	192	C0	224	E0
1	01	33	21	65	41	97	61	129	81	161	A1	193	C1	225	E1
2	02	34	22	66	42	98	62	130	82	162	A2	194	C2	226	E2
3	03	35	23	67	43	99	63	131	83	163	A3	195	C3	227	E3
4	04	36	24	68	44	100	64	132	84	164	A4	196	C4	228	E4
5	05	37	25	69	45	101	65	133	85	165	A5	197	C5	229	E5
6	06	38	26	70	46	102	66	134	86	166	A6	198	C6	230	E6
7	07	39	27	71	47	103	67	135	87	167	A7	199	C7	231	E7
8	08	40	28	72	48	104	68	136	88	168	A8	200	C8	232	E8
9	09	41	29	73	49	105	69	137	89	169	A9	201	C9	233	E9
10	0A	42	2A	74	4A	106	6A	138	8A	170	AA	202	CA	234	EA
11	0B	43	2B	75	4B	107	6B	139	8B	171	AB	203	CB	235	EB
12	0C	44	2C	76	4C	108	6C	140	8C	172	AC	204	CC	236	EC
13	0D	45	2D	77	4D	109	6D	141	8D	173	AD	205	CD	237	ED
14	0E	46	2E	78	4E	110	6E	142	8E	174	AE	206	CE	238	EE
15	0F	47	2F	79	4F	111	6F	143	8F	175	AF	207	CF	239	EF
16	10	48	30	80	50	112	70	144	90	176	B0	208	D0	240	F0
17	11	49	31	81	51	113	71	145	91	177	B1	209	D1	241	F1
18	12	50	32	82	52	114	72	146	92	178	B2	210	D2	242	F2
19	13	51	33	83	53	115	73	147	93	179	B3	211	D3	243	F3
20	14	52	34	84	54	116	74	148	94	180	B4	212	D4	244	F4
21	15	53	35	85	55	117	75	149	95	181	B5	213	D5	245	F5
22	16	54	36	86	56	118	76	150	96	182	B6	214	D6	246	F6
23	17	55	37	87	57	119	77	151	97	183	B7	215	D7	247	F7
24	18	56	38	88	58	120	78	152	98	184	B8	216	D8	248	F8
25	19	57	39	89	59	121	79	153	99	185	B9	217	D9	249	F9
26	1A	58	3A	90	5A	122	7A	154	9A	186	BA	218	DA	250	FA
27	1B	59	3B	91	5B	123	7B	155	9B	187	BB	219	DB	251	FB
28	1C	60	3C	92	5C	124	7C	156	9C	188	BC	220	DC	252	FC
29	1D	61	3D	93	5D	125	7D	157	9D	189	BD	221	DD	253	FD
30	1E	62	3E	94	5E	126	7E	158	9E	190	BE	222	DE	254	FE
31	1F	63	3F	95	5F	127	7F	159	9F	191	BF	223	DF	255	FF

B. Icon List for Voices and Styles

The Icons are determined by the .SXYZ or .TXYZ in the file name.

Icons sorted by voice / style type

S001	S001NLGrandPno1.bmp	S060	S06060s_Organ.bmp	S098	S098VintageMute.bmp
S002	S002NLGrandPno2.bmp	S043	S043Rotor_Organ.bmp	S120	S120MutedGuitar.bmp
S003	S003Live_Grand.bmp	S049	S049Jazz_Organ1.bmp	S082	S082Aloha_Gtr.bmp
S004	S004Grand_Piano.bmp	S051	S051RotaryDrive.bmp	S099	S099SlideGuitar.bmp
S005	S005BrightPiano.bmp	S058	S058Jazz_Organ2.bmp	S109	S109HawaiianGtr.bmp
S012	S012Oct_Piano_1.bmp	S061	S061Jazz_Organ3.bmp	S089	S089PedalSteel.bmp
S013	S013Oct_Piano_2.bmp	S388	S388Jazz_Draw.bmp	S118	S118Mandolin.bmp
S399	S411PianoStr.bmp	S389	S389BluesOrgan.bmp	S123	S123NL_Wood_Bass.bmp
S936	S950_Piano_Choir.bmp	S390	S390SixteenOne.bmp	S126	S126UprightBass.bmp
S017	S017Midi_Grand.bmp	S391	S391SixteenTwo.bmp	S136	S136AcoBass.bmp
S010	S010Honky_Tonk.bmp	S392	S392SixteenFour.bmp	S138	S138Bass_Cymbal.bmp
S006	S006NLHarpsi8.bmp	S393	S393Even_Bars.bmp	S124	S124NL_ElecBass.bmp
S007	S007NLHc8_4.bmp	S394	S394Pop_Organ.bmp	S125	S125Finger_Bass.bmp
S008	S008Harpsichord.bmp	S395	S395RockingOrg.bmp	S127	S127Pick_Bass.bmp
S009	S009GrandHarpsi.bmp	S396	S396Percussive.bmp	S129	S129Slap_Bass.bmp
S937	S951_Harpsi_Str.bmp	S397	S397GospelOrg.bmp	S135	S135Funk_Bass.bmp
S950	S953_CVP209.bmp	S398	S398Pad_Organ.bmp	S408	S554VeloSlap.bmp
S011	S011Rock_Piano.bmp	S068	S068Reed_Organ.bmp	S128	S128Jaco_Bass.bmp
S016	S016NL_CP80.bmp	S069	S069Musette.bmp	S137	S137Fretless.bmp
S018	S018CP_80.bmp	S070	S070Tutti_Accrd.bmp	S144	S144Click_Bass.bmp
S014	S014NL_EP1.bmp	S071	S071Small_Accrd.bmp	S130	S130Analog_Bass.bmp
S023	S023Hyper_Tines.bmp	S072	S072Accordion.bmp	S131	S131DX_FunkBass.bmp
S024	S024Cool_EP.bmp	S074	S074Steirisch.bmp	S132	S132DrySynBass.bmp
S026	S026New_Tines.bmp	S076	S076Soft_Accrd.bmp	S133	S133Touch_Bass.bmp
S028	S028DX_Modern.bmp	S073	S073Tango_Accrd.bmp	S134	S134Hi_Q_Bass.bmp
S030	S030Modern_EP.bmp	S075	S075Bandoneon.bmp	S139	S139Fusion_Bass.bmp
S032	S032Super_DX.bmp	S077	S077Modern_Harp.bmp	S140	S140Rave_Bass.bmp
S301	S301DX_Pad.bmp	S078	S078Blues_Harp.bmp	S141	S141Dance_Bass.bmp
S015	S015NL_EP2.bmp	S079	S079Harmonica.bmp	S142	S142Synth_Bass.bmp
S019	S019Galaxy_EP.bmp	S080	S080NLFolkGtr.bmp	S143	S143Snap_Bass.bmp
S020	S020Stage_EP.bmp	S096	S096Folk_Guitar.bmp	S145	S145Live_Strs.bmp
S021	S021Polaris_EP.bmp	S106	S106CampfireGtr.bmp	S146	S146Live_Algro.bmp
S022	S022Jazz_Chorus.bmp	S103	S103Spanish_Gtr.bmp	S147	S147Live_Orch.bmp
S025	S025Phase_EP.bmp	S081	S081Live_Nylon.bmp	S156	S156Strings.bmp
S027	S027Funk_EP.bmp	S093	S093Live_Class.bmp	S157	S157OrchStrings.bmp
S029	S029Vintage_EP.bmp	S107	S107SmoothNylon.bmp	S163	S163SlowStrings.bmp
S031	S031Tremolo_EP.bmp	S113	S113Classic_Gtr.bmp	S165	S165MarcatoStrs.bmp
S035	S035Suitcase_EP.bmp	S407	S507Ukulele.bmp	S148	S148SymphonStr.bmp
S036	S036Venus_EP.bmp	S086	S08612StrGuitar.bmp	S149	S149ChamberStrs.bmp
S033	S033NL_Clavi.bmp	S084	S084Cool_JGtr.bmp	S150	S150OberStrings.bmp
S034	S034Clavi.bmp	S088	S088Vintage_Amp.bmp	S158	S158StrQuartet.bmp
S037	S037Wah_Clavi.bmp	S094	S094Cool_JSolo.bmp	S159	S159ConcertoStr.bmp
S038	S038NLPipeOrgnP.bmp	S104	S104Octave_Gtr.bmp	S160	S160Analog_Strs.bmp
S039	S039NLPipeOrgF1.bmp	S121	S121Jazz_Guitar.bmp	S162	S162Bow_Strings.bmp
S040	S040NLPipeOrgF2_.bmp	S087	S087SolidGuitar.bmp	S166	S166Syn_Strings.bmp
S041	S041NLPipeOrgnT.bmp	S091	S091Funk_Guitar.bmp	S152	S152Orch_Brass.bmp
S062	S062ChapelOrgn1.bmp	S092	S09260s_Clean.bmp	S153	S153Orch_Flute.bmp
S063	S063ChapelOrgn2.bmp	S095	S095VintageOpen.bmp	S154	S154Orch_FlBr.bmp
S064	S064ChapelOrgn3.bmp	S097	S097Solid_Chord.bmp	S155	S155Orch_Oboe.bmp
S065	S065TheatreOrg1.bmp	S100	S100Lead_Guitar.bmp	S151	S151Solo_Violin.bmp
S066	S066TheatreOrg2.bmp	S102	S102VintageTrem.bmp	S161	S161Soft_Violin.bmp
S067	S067Pipe_Organ.bmp	S108	S108Tremolo_Gtr.bmp	S168	S168Viola.bmp
S042	S042Cool_Organ.bmp	S111	S111BrightClean.bmp	S173	E_S173Fiddle.bmp
S044	S044Rock_Organ1.bmp	S112	S112Wah_Guitar.bmp	S169	S169Cello.bmp
S047	S047Cool_Jazz.bmp	S119	S119CleanGuitar.bmp	S170	S170Contrabass.bmp
S054	S054DrawbarOrg.bmp	S116	S116Elec12Str.bmp	S164	S164TremoloStrs.bmp
S055	S055Click_Organ.bmp	S090	S090Crunch_Gtr.bmp	S167	S167PizzStrings.bmp
S056	S056Stadium_Org.bmp	S110	S110Heavy_Stack.bmp	S178	S178OrchHit.bmp
S045	S045Dance_Organ.bmp	S117	S117FeedbackGtr.bmp	S410	S617OrchHit.bmp
S046	S046Gospel_Org.bmp	S083	S083Carlos_Gtr.bmp	S171	S171Harp.bmp
S048	S048Purple_Org.bmp	S101	S101Chorus_Gtr.bmp	S179	S179Sweet_Trump.bmp
S050	S050Rock_Organ2.bmp	S105	S105Deep_Chorus.bmp	S183	S183SoftTrumpet.bmp
S052	S052Full_Rocker.bmp	S115	S115Distortion.bmp	S186	S186SoloTrumpet.bmp
S053	S053ElecOrgan.bmp	S122	S122Overdrive.bmp	S181	S181SweetMuteTp.bmp
S057	S057Mellow_Draw.bmp	S114	S114DX_JazzGtr.bmp	S184	S184JazzTrumpet.bmp
S059	S059Bright_Draw.bmp	S085	S085Cool_EGtr.bmp	S185	S185Muted_Trump.bmp

S187	S187Air_Trumpet.bmp	S260	S260Bagpipe.bmp	S340	S340Under_Heim.bmp
S182	S182SweetFlugel.bmp	S257	S257Whistle.bmp	S342	S342Hi_Bias.bmp
S188	S188Flugel_Horn.bmp	S264	S264Pro_Heaven.bmp	S343	S343Vinylead.bmp
S180	S180Sweet_Tromb.bmp	S265	S265Sunbeam.bmp	S331	S331Synchronize.bmp
S189	S189Trombone.bmp	S276	S276Bell_Heaven.bmp	S345	S345Clockwork.bmp
S191	S191Solo_Tromb.bmp	S288	S288Vox_Humana.bmp	S341	S341Rhythmatic.bmp
S192	S192Soft_Tromb.bmp	S261	S261LiveGospel.bmp	S302	S302Symbiont.bmp
S216	S216SmoothTromb.bmp	S262	S262Live_Humm.bmp	S303	S303Stargate.bmp
S193	S193MellowTromb.bmp	S263	S263Hah_Choir.bmp	S304	S304Area_51.bmp
S194	S194French_Horn.bmp	S283	S283Choir.bmp	S305	S305Dark_Moon.bmp
S190	S190BaritonHorn.bmp	S285	S285Vocal_Ensbl.bmp	S308	S308Solaris.bmp
S195	S195Bariton_Hit.bmp	S290	S290Uuh_Choir.bmp	S333	S333Stardust.bmp
S197	S197Tuba.bmp	S270	S270Live_Doo.bmp	S309	S309Time_Travel.bmp
S196	S196Alp_Bass.bmp	S271	S271Live_Bah.bmp	S310	S310Millenium.bmp
S198	S198Live_OctBr.bmp	S272	S272Live_Dao.bmp	S293	S293Atmosphere.bmp
S199	S199Live_Brass.bmp	S273	S273Live_Mmh.bmp	S339	S339Sun_Bell.bmp
S200	S200Live_HyBrs.bmp	S274	S274Gothic_Vox.bmp	S346	S346NL_Vibe.bmp
S201	S201BrasSection.bmp	S275	S275Huh_Choir.bmp	S351	S351Jazz_Vibes.bmp
S204	S204Pop_Brass.bmp	S268	S268Live_Vocal.bmp	S350	S350Vibraphone.bmp
S211	S211Step_Brass.bmp	S269	S269Bah_Choir.bmp	S347	S347NL_Marimba.bmp
S212	S212BrightBrass.bmp	S278	S278DooBa_Scats.bmp	S352	S352Marimba.bmp
S213	S213Soft_Brass.bmp	S279	S279Daa_Choir.bmp	S353	S353Xylophone.bmp
S214	S214Full_Horns.bmp	S280	S280Doo_Choir.bmp	S356	S356Glocken.bmp
S205	S205Sforzando.bmp	S281	S281Dooom_Choir.bmp	S348	S348NL_Celesta.bmp
S209	S209Jump_Brass.bmp	S282	S282Live_Dooom.bmp	S355	S355Celesta.bmp
S210	S210Big_Brass.bmp	S266	S266SweetHeaven.bmp	S349	S349NL_Stee_Drum.bmp
S217	S217High_Brass.bmp	S267	S267DreamHeaven.bmp	S354	S354Steel_Drums.bmp
S221	S221Brass_Hit.bmp	S277	S277Pan_Heaven.bmp	S357	S357Music_Box.bmp
S220	S220MellowHorns.bmp	S300	S300Fantasia.bmp	S358	S358TubularBell.bmp
S218	S218Ober_Brass.bmp	S417	S753ChorBell.bmp	S359	S359Kalimba.bmp
S222	S222Analog_Brs.bmp	S418	S757XmasBell.bmp	S360	S360Dulcimer.bmp
S226	S226Soft_Analog.bmp	S419	S758VibeBell.bmp	S361	S361Timpani.bmp
S227	S227FunkyAnalog.bmp	S420	S760AirBells.bmp	S362	S362LiveStdKit.bmp
S228	S228TechnoBrass.bmp	S421	S761BellHarp.bmp	S368	S368StdKit1.bmp
S229	S229Synth_Brass.bmp	S284	S284Air_Choir.bmp	S369	S369StdKit2.bmp
S219	S219Trumpet_Ens.bmp	S289	S289Voices.bmp	S384	S384StyleLvStd.bmp
S224	S224TrbSection.bmp	S294	S294Xenon_Pad.bmp	S376	S376Jazz_Kit.bmp
S206	S206MoonLight.bmp	S295	S295Skydiver.bmp	S364	S364LiveBrush.bmp
S208	S208Saxy_Mood.bmp	S298	S298Equinox.bmp	S367	S367LiveBrsh_P.bmp
S215	S215Brass_Combo.bmp	S286	S286Insomnia.bmp	S377	S377Brush_Kit.bmp
S225	S225Small_Brass.bmp	S296	S296Far_East.bmp	S365	S365LiveStd_P.bmp
S202	S202BigBandBrs.bmp	S312	S312Dunes.bmp	S386	S386StyLvStd_P.bmp
S203	S203MellowBrass.bmp	S287	S287Cyber_Pad.bmp	S387	S387StyLvFunk_P.bmp
S207	S207MillerNight.bmp	S292	S292Neo_WarmPad.bmp	S366	S366LiveFunk_P.bmp
S223	S223BallroomBrs.bmp	S306	S306Ionosphere.bmp	S363	S363LiveFunkKt.bmp
S230	S230Sweet_Tenor.bmp	S291	S291Wave_2001.bmp	S370	S370Hit_Kit.bmp
S242	S242Tenor_Sax.bmp	S307	S307Golden_Age.bmp	S385	S385StyleLvFunk.bmp
S234	S234Growl_Sax.bmp	S297	S297Template.bmp	S371	S371Room_Kit.bmp
S231	S231Sweet_Alto.bmp	S311	S311Transform.bmp	S372	S372Rock_Kit.bmp
S241	S241Alto_Sax.bmp	S299	S299Glass_Pad.bmp	S373	S373Electro_Kit.bmp
S235	S235BreathTenor.bmp	S318	S318Square_Lead.bmp	S374	S374Analog_Kit.bmp
S236	S236BreathyAlto.bmp	S319	S319SawLead.bmp	S375	S375Dance_Kit.bmp
S243	S243BaritoneSax.bmp	S344	S344Skyline.bmp	S378	S378SymphonyKit.bmp
S244	S244Rock_Bari.bmp	S411	S680SineLead.bmp	S379	S379Arabic_Kit.bmp
S232	S232Sweet_Sprno.bmp	S313	S313Oxygen.bmp	S380	S380LiveCuban.bmp
S237	S237Soprano_Sax.bmp	S314	S314Matrix.bmp	S381	S381LivePopLtn.bmp
S239	S239Sax_Section.bmp	S315	S315Wire_Lead.bmp	S382	S382SFX_Kit1.bmp
S240	S240WoodwindEns.bmp	S316	S316Hip_Lead.bmp	S383	S383SFX_Kit2.bmp
S233	S233Sweet_Clari.bmp	S317	S317Hop_Lead.bmp	S400	S436DXKotoEP.bmp
S238	S238MelClarinet.bmp	S320	S320Fire_Wire.bmp	S401	S457Balafon2.bmp
S248	S248Clarinet.bmp	S321	S321Analogon.bmp	S402	E_S458Log_Drum.bmp
S245	S245Oboe.bmp	S322	S322Funky_Lead.bmp	S403	S461ChrchBel.bmp
S246	S246EnglishHorn.bmp	S323	S323Paraglide.bmp	S404	S462Carillon.bmp
S247	S247Bassoon.bmp	S324	S324Robolead.bmp	S406	S466Santur.bmp
S249	S249Sweet_Flute.bmp	S325	S325Fargo.bmp	S405	S465Cimbalom.bmp
S251	S251ClassFlute.bmp	S326	S326Portatone.bmp	S409	S584YangChin.bmp
S253	S253Flute.bmp	S327	S327Blaster.bmp	S172	S172Hackbrett.bmp
S336	S336Synth_Flute.bmp	S328	S328Big_Lead.bmp	S174	S174Banjo.bmp
S254	S254Piccolo.bmp	S329	S329Warp.bmp	S175	S175Sitar.bmp
S250	S250Sweet_Pan.bmp	S330	S330Adrenaline.bmp	S176	S176Koto.bmp
S252	S252Pan_Flute.bmp	S332	S332Tiny_Lead.bmp	S177	S177Shamisen.bmp
S255	S255EthnicFlute.bmp	S334	S334Aero_Lead.bmp	S424	S797Tambra.bmp
S256	S256Shakuhachi.bmp	S335	S335Mini_Lead.bmp	S425	S798Tamboura.bmp
S258	S258Recorder.bmp	S337	S337Sub_Aqua.bmp	S426	S801Rabab.bmp
S259	S259Ocarina.bmp	S338	S338Impact.bmp	S427	S802Gopichnt.bmp

S428 S803Oud.bmp	S497 S877Coaster.bmp	S572 va065XN.bmp
S429 S806T_Koto.bmp	S498 S878SbMarine.bmp	S573 va066XN.bmp
S430 S807Kanoon.bmp	S499 S879Laughing.bmp	S574 va067XN.bmp
S431 S811Shanai.bmp	S500 S880Scream.bmp	S575 va068XN.bmp
S432 S812Shanai2.bmp	S501 S881Punch.bmp	S576 va069XN.bmp
S433 S813Pungi.bmp	S502 S882Heart.bmp	S577 va070XN.bmp
S434 S814Hichriki.bmp	S503 S883FootStep.bmp	S578 va071XN.bmp
S435 S815TnklBell.bmp	S504 S884MchinGun.bmp	S579 va072XN.bmp
S436 E_S816Bonang.bmp	S505 S885LaserGun.bmp	S580 va073XN.bmp
S437 S817Gender.bmp	S506 S886Xplosion.bmp	S581 va074XN.bmp
S438 S818Gamelan_Gong.bmp	S507 S887FireWork.bmp	S582 va075XN.bmp
S439 S819St_Gamelan.bmp	S508 va001XN.bmp	S583 va076XN.bmp
S422 S762Gamelmba.bmp	S509 va002XN.bmp	S584 va077XN.bmp
S440 S820Rama_Cym.bmp	S510 va003XN.bmp	S585 va078XN.bmp
S441 S821AsianBel.bmp	S511 va004XN.bmp	S586 va079XN.bmp
S445 S825ThaiBell.bmp	S512 va005XN.bmp	S587 va080XN.bmp
S442 S822Agogo.bmp	S513 va006XN.bmp	S588 va081XN.bmp
S444 S824GlasPerc.bmp	S514 va007XN.bmp	S589 va082XN.bmp
S446 S826WoodBlok.bmp	S515 va008XN.bmp	S590 va083XN.bmp
S447 S827Castanet.bmp	S516 va009XN.bmp	S591 va084XN.bmp
S448 S828TaikoDrm.bmp	S517 va010XN.bmp	S592 va085XN.bmp
S449 S829GrCassa.bmp	S518 va011XN.bmp	S593 va086XN.bmp
S450 S830MelodTom.bmp	S519 va012XN.bmp	S594 va087XN.bmp
S451 S831Mel_Tom2.bmp	S520 va013XN.bmp	S595 va088XN.bmp
S452 S832Real_Tom.bmp	S521 va014XN.bmp	S596 va089XN.bmp
S453 S833Rock_Tom.bmp	S522 va015XN.bmp	S597 va090XN.bmp
S455 S835Ana_Tom.bmp	S523 va016XN.bmp	S598 va091XN.bmp
S454 S834SynDrum.bmp	S524 va017XN.bmp	S599 va092XN.bmp
S456 S836ElecPerc.bmp	S525 va018XN.bmp	S600 va093XN.bmp
S457 S837RevCymbl.bmp	S526 va019XN.bmp	S601 va094XN.bmp
S458 S838FretNoiz.bmp	S527 va020XN.bmp	S602 va095XN.bmp
S466 S846CuttngNz.bmp	S528 va021XN.bmp	S603 va096XN.bmp
S467 S847CttngNz2.bmp	S529 va022XN.bmp	S604 va097XN.bmp
S468 S848Str_Slap.bmp	S530 va023XN.bmp	S605 va098XN.bmp
S459 S839BrthNoiz.bmp	S531 va024XN.bmp	S606 va099XN.bmp
S469 S849FlKClik.bmp	S532 va025XN.bmp	S607 va100XN.bmp
S412 S741AfrcnWnd.bmp	S533 va026XN.bmp	S608 va101XN.bmp
S413 S742Caribean.bmp	S534 va027XN.bmp	S609 va102XN.bmp
S414 S744Prologue.bmp	S535 va028XN.bmp	S610 va103XN.bmp
S415 S745Ancestrl.bmp	S536 va029XN.bmp	S611 va104XN.bmp
S416 S748Popcorn.bmp	S537 va030XN.bmp	S612 va105XN.bmp
S423 S780Night.bmp	S538 va031XN.bmp	S613 va106XN.bmp
S460 S840Seashore.bmp	S539 va032XN.bmp	S614 va107XN.bmp
S461 S841Tweet.bmp	S540 va033XN.bmp	S615 va108XN.bmp
S462 S842Telphone.bmp	S541 va034XN.bmp	S616 va109XN.bmp
S463 S843Helicptr.bmp	S542 va035XN.bmp	S617 va110XN.bmp
S464 S844Applause.bmp	S543 va036XN.bmp	S618 va111XN.bmp
S465 S845Gunshot.bmp	S544 va037XN.bmp	S619 va112XN.bmp
S470 S850Rain.bmp	S545 va038XN.bmp	S620 va113XN.bmp
S471 S851Thunder.bmp	S546 va039XN.bmp	S621 va114XN.bmp
S472 S852Wind.bmp	S547 va040XN.bmp	S622 va115XN.bmp
S473 S853Stream.bmp	S548 va041XN.bmp	S623 va116XN.bmp
S474 S854Bubble.bmp	S549 va042XN.bmp	S624 va117XN.bmp
S475 S855Feed.bmp	S550 va043XN.bmp	S625 va118XN.bmp
S476 S856Dog.bmp	S551 va044XN.bmp	S626 va119XN.bmp
S477 S857Horse.bmp	S552 va045XN.bmp	S627 va120XN.bmp
S478 S858Bird_2.bmp	S553 va046XN.bmp	S628 va121XN.bmp
S479 S859Ghost.bmp	S554 va047XN.bmp	S629 va122XN.bmp
S480 E_S860Maou.bmp	S555 va048XN.bmp	S630 va123XN.bmp
S481 S861TelDial.bmp	S556 va049XN.bmp	S631 va124XN.bmp
S487 S867Telphon2.bmp	S557 va050XN.bmp	S632 va125XN.bmp
S482 S862DoorSqek.bmp	S558 va051XN.bmp	S633 va126XN.bmp
S483 S863Door_Slam.bmp	S559 va052XN.bmp	S634 va127XN.bmp
S484 S864Scratch.bmp	S560 va053XN.bmp	S635 va128XN.bmp
S485 S865Scratch_2.bmp	S561 va054XN.bmp	S636 va129XN.bmp
S486 S866WindChm.bmp	S562 va055XN.bmp	S637 va130XN.bmp
S488 S868CarEngin.bmp	S563 va056XN.bmp	S638 va131XN.bmp
S489 S869Car_Stop.bmp	S564 va057XN.bmp	S639 va132XN.bmp
S490 S870Car_Pass.bmp	S565 va058XN.bmp	S640 va133XN.bmp
S491 S871CarCrash.bmp	S566 va059XN.bmp	S641 va134XN.bmp
S492 S872Siren.bmp	S567 va060XN.bmp	S642 va135XN.bmp
S493 S873Train.bmp	S568 va061XN.bmp	S643 va136XN.bmp
S494 S874Jetplane.bmp	S569 va062XN.bmp	S644 va137XN.bmp
S495 S875Starship.bmp	S570 va063XN.bmp	S645 va138XN.bmp
S496 S876Burst.bmp	S571 va064XN.bmp	S646 va139XN.bmp

S647 va140XN.bmp	S722 ST003Disco1.bmp	S764 ST049Note8L.bmp
S648 va141XN.bmp	S723 ST003Disco2.bmp	S765 ST050Note8S.bmp
S649 va142XN.bmp	S724 ST004Club.bmp	S914 ST052S_.bmp
S650 va143XN.bmp	S725 ST005PlayPiano.bmp	S766 st053Note82L_1.bmp
S651 va144XN.bmp	S726 ST006PlayEGuitar.bmp	S767 ST053Note82L_2.bmp
S652 va145XN.bmp	S727 ST007PlayAGuitar.bmp	S768 ST054Note82S.bmp
S653 va146XN.bmp	S728 ST008PlayBanjo.bmp	S769 ST059Note16L.bmp
S654 va147XN.bmp	S729 ST009PlayAccord.bmp	S770 ST060Note16S.bmp
S655 va148XN.bmp	S730 ST010PlayBass.bmp	S912 ta063SongCreator.bmp
S656 va149XN.bmp	S731 ST011PrayTrumpet.bmp	S793 TA009Score1.bmp
S657 va150XN.bmp	S732 ST012PlaySax.bmp	S794 ta010Score2.bmp
S658 va151XN.bmp	S733 ST013PianoTrio.bmp	S932 ta060ScoreDemo.bmp
S659 va152XN.bmp	S734 ST014Metronome.bmp	S817 ta033SongSetting.bmp
S660 va153XN.bmp	S735 ST015Grass1.bmp	S931 ta061LylicDemo.bmp
S661 va154XN.bmp	S736 ST017Grass2.bmp	S792 TA008Keyboard.bmp
S662 va155XN.bmp	S737 ST018Grass3.bmp	S947 ta055MezzoForte.bmp
S663 va156XN.bmp	S738 st019Grass4.bmp	S948 ta056Forte.bmp
S664 va157XN.bmp	S739 ST019Grass5.bmp	S949 ta057Fortissimo.bmp
S665 va158XN.bmp	S740 ST021Bottole1.bmp	S837 taD02PanelStyle.bmp
S666 va159XN.bmp	S741 ST022Bottole2.bmp	S929 taD03StyleCreator.bmp
S667 va160XN.bmp	S742 ST023RecPlayer1.bmp	S928 ta062SoundCreator.bmp
S668 SN001Mozart.bmp	S743 ST024RecPlayer2.bmp	S910 taD08Mpad.bmp
S669 SN002Bach.bmp	S744 ST025Record.bmp	S911 taD07MpadCreator.bmp
S670 SN003Beethoven.bmp	S901 KD2SoundBoard1Icon.bmp	S941 ta051_MIDI.bmp
S671 SN004Chopin.bmp	S902 KD3SoundBoard2Icon.bmp	S832 ta045LogoGM.bmp
S672 SN005Schubert.bmp	S859 KD1LightIcon.bmp	S829 ta042LogoXG.bmp
S673 SN006Mendelssohn.bmp	S850 K01SkyIcon.bmp	S833 ta046LogoSFF.bmp
S674 SN007Schumann.bmp	S851 K02FlowerGardenIcon.bmp	S831 ta044LogoXF.bmp
S675 SN008Rubinstein.bmp	S852 K03EarthIcon.bmp	S830 ta043LogoVH.bmp
S676 SN009Liszt.bmp	S853 K04RedRoseIcon.bmp	S945 ta052CueTimeLogo.bmp
S677 SN010Haydn.bmp	S854 K05JapanIcon.bmp	S834 ta047LogoDOC.bmp
S678 sn012Debussy.bmp	S900 K11Japan2Icon.bmp	S835 ta048PanelMF.bmp
S679 sn013Dvorak.bmp	S855 K06USAIcon.bmp	S836 ta049PanelOTS.bmp
S680 sn014Albeniz.bmp	S856 K07GermanyIcon.bmp	S913 ta052_Effect_s.bmp
S681 sn015Tchaikovsky.bmp	S899 K10Germany2Icon.bmp	S946 ta054Onsa.bmp
S682 sn016Joplin.bmp	S858 K09GBRIcon.bmp	S944 ta053FollowLightLogo.bmp
S683 sn017Weber.bmp	S857 K08FranceIcon.bmp	S807 TA022RotarySp1.bmp
S684 i01Sunset.bmp	S745 ST026FlagUSA.bmp	S808 ta022RotarySp2.bmp
S685 i02Grass.bmp	S746 ST027FlagJapan.bmp	S791 TA007CVPPanel.bmp
S686 i03Rose_Yellow.bmp	S747 ST028FlagGBR.bmp	S823 ta038Utility.bmp
S687 i04Rose_Red.bmp	S942 st082SFlagIreLand.bmp	S816 TA032Tune.bmp
S688 i05Tulip.bmp	S943 st083SFlagScotLand.bmp	S819 TA035Contoroller1.bmp
S689 i06Swllowtail.bmp	S748 ST029FlagGerman.bmp	S825 taD04MIDI.bmp
S690 i07Butterfly.bmp	S749 ST030FlagItalia.bmp	S826 TA040Video_Out1.bmp
S691 i08Leaf_Green.bmp	S750 ST031FlagFrance.bmp	S827 ta040Video_Out2.bmp
S692 i09Maple_Green.bmp	S751 ST032FlagSpain.bmp	S784 TA001FloppyDisk.bmp
S693 i10Leaf_Red.bmp	S752 ST033FlagBrazil.bmp	S785 TA002CD_ROM.bmp
S694 i11Apple.bmp	S753 ST034FlagArgen.bmp	S786 TA003CD.bmp
S695 i12Cherry.bmp	S754 ST035MapNAmerica.bmp	S787 taD01Folder.bmp
S696 i13Banana.bmp	S755 ST036MapSAmerica.bmp	S788 ta004Folder2.bmp
S697 i14Orange.bmp	S756 ST037MapJapan.bmp	S795 TA011Conducter.bmp
S698 i15Snowfield.bmp	S757 ST038MapGBR.bmp	S818 ta034StyleSetting.bmp
S699 i16Sunflower.bmp	S758 ST039MarGerman.bmp	S822 taD06RegistContent.bmp
S700 i17Saturn.bmp	S759 ST040MapItalia.bmp	S917 taD05Regist.bmp
S701 i18Beer.bmp	S760 ST041MapFrance.bmp	S796 TA012LSI1.bmp
S702 i19Woods.bmp	S761 ST042MapSpain.bmp	S797 ta012LSI2.bmp
S703 i20SnowMountain.bmp	S789 TA005MapWorld.bmp	S805 TA018Setup1.bmp
S704 i21CherryBlossom.bmp	S790 TA006Earth.bmp	S806 ta018Setup2.bmp
S705 i22Beach.bmp	S828 TA041TalkSetting.bmp	S811 TA025BackUp1.bmp
S706 i23CoconutTree.bmp	S815 ta027Mic.bmp	S812 ta025BackUp2.bmp
S707 i24XmaTree.bmp	S771 ta053_VH_Duet.bmp	S813 TA026Preset1.bmp
S708 i25Cat.bmp	S772 ta054_VH_Trio.bmp	S814 ta026Preset2.bmp
S709 i26Dog.bmp	S773 ta055_VH_Male.bmp	S798 TA013Wrench.bmp
S710 i27Coconut_BlueSky.bmp	S774 ta056_VH_Female.bmp	S799 TA014Driver1.bmp
S711 i28Coconut_Sunset.bmp	S775 ta057_VH_Chodal1.bmp	S800 TA015Driver2.bmp
S712 i29Pleiades.bmp	S776 ta058_VH_Chodal2.bmp	S903 KD4Wavelet1Icon.bmp
S713 i30Penguin.bmp	S777 ta059_VH_Chodal3.bmp	S904 KD5Wavelet2Icon.bmp
S714 i031TreeOnGrass.bmp	S778 st076_16Beat_1.bmp	S905 KD6Wavelet3Icon.bmp
S715 i032lowerGraden.bmp	S779 st077_16Beat_2.bmp	S906 KD7Wavelet4Icon.bmp
S716 i33FallingSun.bmp	S780 st078_16Beat_3.bmp	S907 KD8Wavelet5Icon.bmp
S717 i34DeadLeaf.bmp	S781 st079_8Beat_1.bmp	S908 KD9Wavelet6Icon.bmp
S718 i35Lighting.bmp	S782 st080_8Beat_2.bmp	S909 KD10Wavelet7Icon.bmp
S719 i36Candle.bmp	S783 st081_8Beat_3.bmp	S887 WP_28Icon.bmp
S720 ST001SocialDanc1.bmp	S762 ST043Note4L.bmp	S888 WP_29Icon.bmp
S721 ST002SocialDanc2.bmp	S763 ST044Note4S.bmp	S889 WP_30Icon.bmp

V2.0

S890 WP_31Icon.bmp
S896 WP_37XIcon.bmp
S860 WP_01Icon.bmp
S861 WP_02Icon.bmp
S862 WP_03Icon.bmp
S863 WP_04Icon.bmp
S864 WP_05Icon.bmp
S865 WP_06Icon.bmp
S866 WP_07Icon.bmp
S867 WP_08Icon.bmp
S868 WP_09Icon.bmp
S869 WP_10Icon.bmp
S870 WP_11Icon.bmp
S871 WP_12Icon.bmp
S872 WP_13Icon.bmp
S873 WP_14Icon.bmp
S874 WP_15Icon.bmp
S875 WP_16Icon.bmp
S876 WP_17Icon.bmp
S877 WP_18Icon.bmp
S878 WP_19Icon.bmp
S880 WP_21Icon.bmp
S881 WP_22Icon.bmp
S882 WP_23Icon.bmp
S883 WP_24Icon.bmp
S884 WP_25Icon.bmp
S885 WP_26Icon.bmp
S886 WP_27Icon.bmp
S892 WP_33Icon.bmp
S893 WP_34Icon.bmp
S894 WP_35Icon.bmp
S838 bkc_01Icon.bmp
S839 bkc_02Icon.bmp
S840 bkc_03Icon.bmp
S841 bkc_04Icon.bmp
S842 bkc_05Icon.bmp
S843 bkc_06Icon.bmp
S844 bkc_07Icon.bmp
S845 bkc_08Icon.bmp
S846 bkc_09Icon.bmp
S847 bkc_10Icon.bmp
S848 bkc_11Icon.bmp
S849 bkc_12Icon.bmp
S990 S990_SteelMega.bmp
S991 S991_HiStringMega_2.bmp
S992 S992_CleanGtMega.bmp
S993 S993_OverdriveMega.bmp
S994 S994_DistortionMega.bmp
S995 S995_FingerBassMega1.bmp
S996 S996_PickBassMega.bmp
S997 S997_FretlessMega.bmp
S998 S998_AcoBassMega.bmp
S999 S955_CustomNormal.bmp
S000 S956_CustomDrum.bmp
T001 Security_VGA_Lock.bmp
T002 Security_VGA_Edit.bmp
T003 Sxxx_12StrGuitarMega.bmp
T004 vgaC_OS_Org_FaRa.bmp
T005 vgaC_OS_Ed_FaRa.bmp
T006 CCVGAS.bmp
T007 HPVGAS.bmp
T008 SLVGAS.bmp
T010 i39Wave_CV_Normal.bmp
T009 i40Wave_CV_Drum.bmp
T069 S955_LcutCustomNormal.bmp
T070 S956_LcutCustomDrum.bmp
T071 i39_LcutWave_CV_Normal.bmp
T072 i40_LcutWave_CV_Drum.bmp
T013 i42Audio.bmp
T014 i41WaveAiff.bmp
T015 i38LibraryInfo.bmp
T016 i37TextFile.bmp
T068 SX100_FRET.bmp
T011 WP_45Icon.bmp
T012 WP_46Icon.bmp

T065 WP_47Icon.bmp
T066 WP_48Icon.bmp
T017 Sx01SA_Harpsi.bmp
T018 Sx02SA_Organ.bmp
T019 Sx03SA_Rocker.bmp
T020 Sx04SA_Strs1.bmp
T021 Sx05SA_Strs2.bmp
T022 Sx06SA_Strs3.bmp
T023 Sx07SA_Strs4.bmp
T024 Sx08SA_Strs5.bmp
T025 Sx09SA_Strs6.bmp
T026 Sx10SA_Brass1.bmp
T027 Sx11SA_Brass2.bmp
T028 Sx12SA_Brass3.bmp
T029 Sx13SA_Brass4.bmp
T030 Sx14SA_Brass5.bmp
T031 Sx15SA_Brass6.bmp
T032 Sx16SA_Brass7.bmp
T033 Sx17SA_Tp1.bmp
T034 Sx18SA_Tp2.bmp
T035 Sx19SA_Tp3.bmp
T036 Sx20SA_Tp4.bmp
T037 Sx21SA_Tp5.bmp
T038 Sx22SA_Tp6.bmp
T039 Sx23SA_Tp7.bmp
T040 Sx24SA_Tenor1.bmp
T041 Sx25SA_Tenor2.bmp
T042 Sx26SA_Tenor3.bmp
T043 Sx27SA_Nylon1.bmp
T044 Sx28SA_Spanish.bmp
T045 Sx29SA_Nylon2.bmp
T046 Sx30SA_Steel1.bmp
T047 Sx31SA_Steel2.bmp
T048 Sx32SA_Clean1.bmp
T049 Sx33SA_Clean2.bmp
T050 Sx34SA_Clean3.bmp
T051 Sx35SA_Clean4.bmp
T052 Sx36SA_Clean5.bmp
T053 Sx37SA_Dist1.bmp
T054 Sx38SA_Dist2.bmp
T055 Sx39SA_Dist3.bmp
T056 Sx40SA_Pedal.bmp
T067 SX41SA_SYNTH.bmp
T057 Sxx1Mega_Nylon.bmp
T058 Sxx2Mega_Clean.bmp
T059 Sxx3Mega_Clean2.bmp
T060 Sxx4Mega_Strs.bmp
T061 Sxx5Mega_Strs2.bmp
T062 Sxx6Mega_Brass.bmp
T063 Sxx7Mega_Trmp.bmp
T064 Sxx8Mega_Tenor.bmp

V2.0