Serial III Audio Mixer Protocol

TABLE OF CONTENTS

1. Introduction

1.1. Purpose

Serial III is an extension to the Serial II (ESAM II) protocol. It is supported (along side ESAM II) in D/ESAM 800 racks versions 3.6 or higher. All of the communication parameters, low level protocol and SMPTE address remain unchanged from ESAM II.

The added functionality is intended to bridge the gap between ESAM II and the up coming D/ESAM Protocol. Not all features of the D/ESAM Protocol could be incorporated in this extension due to limitations in the D/ESAM 800 hardware.

1.2. Related Documents

This document is intended to be an addendum to the Serial II protocol document. It is not intended to be used alone.

1.3. Terms

Terms used in this document are generally familiar to the editing community. Other terms, however, will benefit from some clarification. Following is a list of terms and their intended interpretation:

FROM:	*Something at the start point of an edit.*
TO:	*Something at the end point of an edit.*
INPUT:	*A single monophonic input to the mixer.*
INPUT FADER:	*A single monophonic level control for an INPUT.*
MACHINE:	*A group of INPUTs, the grouping and ROUTING specified within the mixer.*
MACHINE FADER:	*Level control for a group of INPUTs specified by a particular MACHINE.*
VIRTUAL MACHINE:	*An entity established by the grouping of some INPUTs. There are 255 possible Virtual Machines, each having an ASCII name (up to nine characters), and up to 100 inputs (0-99) associated with it (0 is a special CUE TRACK). These machines are addressed by a number from 1 to 255.*
LOGICAL MACHINE:	*All parameters associated with a machine (level, routing, EQ) are stored as part of a Logical Machine. The actual inputs (and number of inputs) used for the machine depends on which Virtual Machine is currently assigned to a given Logical Machine.*
ASSIGNMENT:	*To use a Logical Machine, one must first ASSIGN it to a Virtual Machine. That is where it obtains information regarding which INPUT to access for which TRACK on the Logical Machine.*
TRACK:	*A single monophonic input associated with a particular LOGICAL MACHINE. Each LOGICAL MACHINE may have up to 10 tracks numbered 0-9.*

1.4. A Brief Introduction to the D/ESAM Audio Mixers

A primary design goal of the D/ESAM Audio Mixerwas to provide a space efficient control surface. Assignable faders and crosspoint buttons made this goal achievable. Even though you may not even have a control panel in your application, this assignability feature figures greatly in this protocol. Correct use of this protocol requires a reasonably good understanding of the D/ESAM Audio Mixer's assignable architecture.

Physical inputs are associated with a Virtual Machine in a process called Virtual Machine Definition. This is usually done from a maintenance terminal as part of installing the mixer. The inputs may appear in more than one Virtual Machine. Inputs may even be used more than once in the *same* Virtual Machine.

At some point at the beginning of each edit session, the employed Logical Machines are assigned to a Virtual Machine much like, on most edit controllers, a *green key* is assigned to a video crosspoint (e.g. 'A' is video crosspoint 5, 'R' is crosspoint 1, etc.) This establishes which set of inputs (machine) is being called 'A' and which set of inputs is called 'R'. It also establishes a sensible input order starting at 1 (e.g. 1 2 3 4 instead of 34 35 27 28). In a sense, the Virtual Machine insulates the operator from strange combinations of input numbers established by a convenient installation.

All edit-by-edit information (i.e. fader levels, track selection and routing, EQ parameters) are associated with the Logical Machine, not the Virtual Machine. This facilitates cases where the edit needs to be repeated on a different machine (different day or the machine breaks down). One merely reassigns the Logical Machine to the substitute machine and continues to edit. Since the parameters follow the Logical Machine, the only difference is which inputs are accessed.

Since adjustment parameters are associated with the Logical Machine, the edit controller can not have *random access* to all of the Virtual Machines all of the time. The desired Virtual Machine must first be assigned to a Logical Machine. After this, all desired features may be accessed by addressing that Logical Machine.

For a more complete description of the operating characteristics, refer to the Operators Manual for one of the GPS Digital Audio Mixers.

1.5. Control Panel Considerations

Having a control panel connected to the audio processing frame imposes some limits on system flexibility. The panel makes specific use of broader resources. This protocol is intended to exploit some features of the D/ESAM audio engine that cannot be accessed via the standard GPS D/ESAM control panels. As a consequence, some commands will operate only in the presence of a panel, some only in the absence of a panel, and some will always operate. Commands will be flagged as follows:


Must have a panel
connected to function


Must **not** have a panel
connected to function

Commands that function in both cases have no such flags.

2. The Commands

The command descriptions follow this model:

Command Name (<Read code> and <Write code>)

Description:

Description of the command.

Data:

Detailed description of the required data.

Read example:

Write example:

Applicable Effects Bank:

The effects banks which make use of the command.

Caveats:

Hazards, side effects, and nuances.

Related commands:

Cross references to related commands not in this section.

Support status:

The extent to which GPS supports the command.

2.1. Common Data Types

Data types that occur in many commands are detailed in this section rather than cluttering this document with repetition.

2.1.1. Machine Byte

Logical Machines are referred to by name. To avoid having to send the ASCII name over the serial link, an eight bit binary number between 0 and 255 has been selected for each possible Logical Machine name. The following table specifies this relationship:

Logical Machine	Binary Representation (hexadecimal)
Silent (Black) (Off)	0
A	1
B	2
...	...
Z	1A
AA	1B
AB	1C
...	...
TONE1	DF
TONE2	DE
...	...
TONE16	D0
AUX1	EF
AUX2	EE
...	...
AUX16	E0
RECORD1	FF
RECORD2	FE
...	...
RECORD16	F0

Table 2-1

Virtual Machines are referred to by number from 0 (SILENT) to 255.

All the commands that operate on machines accept a Logical Machine number.

To provide compatibility with controllers that would rather not utilize the assignability features and/or would rather not have to figure out if this mixer is assignable or not, a short-cut access method is available. A discussion of this short cut method follows.

The controller may wish to treat all mixers as if they were a series of Virtual Machines from 0 (SILENT) to 255. This approach will work for the assignable case if the controller relies on a mixer default one-to-one Logical to Virtual assignment (i.e. Logical Machine A is assigned to Virtual Machine 1, Logical Machine B is assigned to Virtual Machine 2, etc.). However, the controller will be limited to the maximum number of Logical Machines.

To support this simple approach and still provide sequential access to the AUX, RECORD and TONE machines, duplicate access will be provided to AUX, RECORD and TONE machines immediately following the number representing the last Logical Machine in the assignable mixer.

For example:

The GPS D/ESAM 800 is an assignable mixer. It has eight Logical Machines (A-H), one AUX machine, one RECORD machine and one TONE machine. AUX would then respond to EF, as well as 09. RECORD would respond to FF and 0A. TONE would respond to DF and 0B.

If the mixer did have multiple AUX, multiple RECORD, and/or multiple TONE machines, they would be grouped together sequentially at the end or the normal Logical Machines in the AUX, RECORD, TONE order.

2.1.2. Track Byte

The *track byte* specifies which of the machine's inputs the command will affect.

Every machine's inputs start at track 1 regardless of the actual input number specified as part of the Virtual Machine's definition. The track numbering continues contiguously to the last track (i.e., it is not possible to have a 4 track machine that has tracks 1,2,4 and 5.)

The last track of a particular machine depends on how many inputs were specified in that Virtual Machine's definition. The maximum for the D/ESAM 800 is 9.

A special track number, 0, is used to represent the Cue track. Associated with this Cue track are many bizarre and unusual features. Refer to the D/ESAM 800 Operator's Manual for a complete description of this track.

2.1.3. Bus Byte

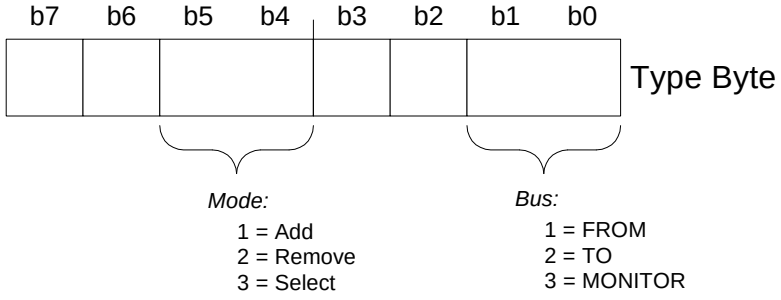
The *bus byte* specifies which of the program outputs the command will affect.

A set bit cause action on that bus. Eight buses may be accessed individually or in any combination unless otherwise noted in the command's data description.

b7	b6	b5	b4	b3	b2	b1	b0
Bus 1	bus 2	Bus 3	Bus 4	Bus 5	Bus 6	Bus 7	Bus 8

2.1.4. Type Byte

The detail for the type byte used by many commands is provided here:



The Bus bits determine which of the three bus groups the command is directed to: The TO bus (Preset), the FROM bus (Program), or the MONITOR bus (Preview). The Mode bits select one of three actions to take:

Add: Add new sources to existing sources.

Remove: Take away sources from existing sources.

Select: Remove all sources and then add only this new source.

2.2. Setup Commands

2.2.1. Virtual Machine Definition (13)

Description:

Report the current definition of a Virtual Machine. A Virtual Machine definition consists of:

- An address (which Virtual Machine number from 0 - 255),
- A name (ASCII characters followed by a NULL),
- And a series of input numbers.

Data:

The first byte is the address. The address must be between 1 and 255.

The next sequence of up to ten bytes represents the Name. The Name is limited to the characters A-Z, a-z, 0-9, and '_' (underline). The Name must start with a letter. This ASCII string must be terminated by a NULL (0). The null is used to determine the end of the name and the start of the input number series.

The input numbers must be between 1 and 56. The first input number defines the Cue track. If there is no Cue track for this machine, the first input number must be zero (0). The series of input numbers following Cue must not number more than 9.

Read example:

Interrogate Virtual Machine twelve:

Length	Bank	Command	Address
03	01	13	0C

Response assuming Virtual Machine twelve is called 'VTR1', has no Cue track, and has four inputs: 33, 34, 35, 36:

STX	Length	Bank	Command	Address	Name	Cue	Inputs
02	0D	01	93	0C	56,54,52,31,00	00	21,22,23,24

The response if the machine is undefined:

STX	Length	Bank	Command	Address	Name	Cue
02	05	01	93	0C	00	00

Write example:

(Not supported in the version)

Applicable Effects Bank:

1 (Mixer).

Caveats:

None.

Related commands:

None.

Support status:

Read command supported.

2.2.2. Logical Machine Assignment (95)

Description:

Sets the Virtual Machine currently assigned to the specified Logical Machine.

Data:

The Write command has two bytes of data. The first byte represents the Logical Machine, according to Table 2.1. The second byte represents a Virtual Machine.

Write example:

Assign Logical Machine RECORD1 to Virtual Machine 3:

Length	Bank	Command	Logical Machine	Virtual; Machine
04	01	95	FF	03

Applicable Effects Bank:

1 (Mixer)

Caveats:

None.

Related commands:

None.

Support status:

Write command supported.

2.2.3. Machine Configuration (CB)

Description:

This command sets the FROM, TO or MONITOR source configuration for the specified machine. Configuration here refers to the input track to output bus mapping (or bus assignment as it is sometimes termed).

Data:

The write command has at least three bytes of data. The first in both cases is a machine byte data type as described in2.1.1 Machine Byte. The second is the type byte data type as described in2.1.4 Type Byte. The Add, Remove, Select part of the type byte is ignored.

The rest of the data provided in the write command will comply to the bus byte data type discussed in section2.1.3 Bus Byte. There is always one of these data types representing the configuration for the Cue track. This is required even if the machine

lacks a Cue track. In most cases, this may be set to zero. If Cue track switching is important to you, read up on its use in the Operators Manual[1].

The bus bytes following the Cue track bus byte set the configuration for input tracks in sequence from track one to 9.

Write example:

Set the AUX1 configuration for track one to feed the first four output buses. Assume it is a two track machine with annoying audio on track 2. Set the same configuration on the MONITOR bus:

Length	Bank	Command	Machine	Type	Cue buses	Input buses
07	01	CB	EF	01	00	F0,00
Length	**Bank**	**Command**	**Machine**	**Type**	**Cue buses**	**Input buses**
07	01	CB	EF	03	00	F0,00

Applicable Effects Bank:

1 (Mixer).

Caveats:

If you do not write to all tracks of the machine, the omitted tracks remain unchanged.

Related commands:

None.

Support status:

Write command supported, except that TO configuration (type byte = 2) is not supported. TO configuration follows the FROM configuration.

2.2.4. Special Modes (C3)

Description:

This command provides the ability to either set or clear bits in the Special Modes bit map. The bit map is as follows:

7	6	5	4	3	2	1	0
Set/$\overline{\text{Clear}}$			X fade Split	Fx Out	Mon Pst	Mon Pgm	Fdr Split

If the SET/CLEAR bit is logical 1, any bit set in the bit map will turn on the respective mode. If the SET/CLEAR bit is logical 0, any bit set in the bit map will turn off the respective mode. The modes are described below.

[1] Refer to section 2-63 of the GPS D/ESAM 800 Operator's Manual for further information.

Fdr Split

When this mode is off, the TO (Preset) portion of any fader control command is ignored. The FROM (Program) value is used for the level of audio selected on either or both of the FROM or TO buses.

When this mode is on, TO and FROM levels are independent for all fader values; TO controlling the level of the corresponding selections on the TO bus and FROM controlling the level of the corresponding selections on the FROM bus. When Fdr Split is first turned on, all TO faders are initialized to equal that of their respective FROM fader values.

Mon Pgm/Mon Pst

These two controls correspond to the PGM FROM and PST TO tally buttons on the D/ESAM control panel. They are multi-function buttons providing access to many D/ESAM features not covered here. They are included in this command because they have an effect on monitoring.

In the case where MIX OUT is off, and FADER SPLIT is on, the level of audio selections on the MONITOR (Preview) bus reflect the setting of:

- The FROM faders if Mon Pgm is asserted,
- The TO faders if Mon Pst is asserted,
- The average of the two settings (mid point of the transition) if both are asserted.

In the case where MIX OUT is on, these bits control whether the audio comes from the FROM bus or the TO bus.

If a control panel is connected to the rack, one of these two bits must be asserted. In the absence of a control panel, clearing both bits puts the mixer in the Eight Bus mode. The MONITOR functions in the protocol are disabled.

Fx Out

Asserting this bit shuts off all effects processing.

Xfade Split

Not supported in this version.

Data:

The Write command has only the special modes bit map as data.

Write example:

Turn On Fader Split mode:

Length	Bank	Command	Special Modes Bits
03	01	C3	81

Set the MON mode to PST. Two commands are necessary for this because one command must make sure MON PGM is off:

Length	Bank	Command	Special Modes Bits
03	01	C3	84

Length	Bank	Command	Special Modes Bits
03	01	C3	02

Applicable Effects Bank:

1 (Mixer).

Caveats:

To ensure that some bits are set and other bits are cleared requires **two** commands.

Putting the mixer in the eight channel mode with the panel connected may have undesirable and unexpected results.

In this version only, the fader split lamp on the control panel will not indicate the proper fader split mode if fader split mode is changed using this command.

Related commands:

2.4.3 Fader Track (19 and 99).

2.6 Transition Commands

Crossfade Position (C4).

Support status:

Write command Supported.

2.3. Input Selecting Commands

2.3.1. Machine (96)

Description:

This command SELECTs, ADDs, or REMOVEs a machine on either the FROM, TO or MONITOR buses. The machine's configuration[2] is provided by the mixer. Write has no effect if source control is disabled.

Data:

The write command has two bytes of data. The first is the Type Byte data type. The second byte is the Machine Byte data type.

Write example:

Put the RECORD1 machine on the MONITOR bus, removing anything that was there previously:

[2] *Configuration* refers to the routing of input tracks to output buses.

Length	Bank	Command	Type	Machine
04	01	96	33	FF

Select the A and D machines on the TO bus removing what was there previously:

Length	Bank	Command	Type	Machine
04	01	96	32	01

Length	Bank	Command	Type	Machine
04	01	96	12	04

Applicable Effects Bank:

1 (Mixer).

Caveats:

None.

Related commands:

2.2.2 Logical Machine Assignment (95)

Support status:

Write only support at this time. Eight Logical Machines (A - H), in addition to AUX1, TONE1, and RECORD1, are supported on the D/ESAM 800. Virtual Machine short-cut (discussed in section 2.1.1 Machine Byte) is also supported for up to machine 0b (11 decimal). (AUX = EF & 09, R = FF & 0A, and TONE = DF & 0B.)

2.4. Input Level Commands

2.4.1. Level representation

The following fader commands share the same level data format. There are three shapes of control. They are:

- **Position** 8 bits Fader position from 0 - 255
- **Log** 14 bits Adjustment in Decibels
- **Linear** 24 bits *The* audio multiplier

Detail of each shape follows:

Position

The position shape represents linear fader position. It mimics the taper produced by the control panel software. The software taper in GPS mixers provides fine resolution at the top travel of the fader changing to coarse resolution near the bottom.

The strong point of this kind of control is its simplicity. The drawback of using this control is that the resulting gain for a given position value may vary from one design to another.

The following table identifies the approximate gain associated with some possible fader position values on the GPS D/ESAM 800 mixer.

Fader Markings (dB)	Fader Position Value	Actual Gain (dB)
+6	255	+5.9
+3	234	+3.0
0	208	0.0
-3	182	-3.0
-6	156	-5.9
-12	129	-11.9
-18	102	-18.0
-24	75	-24.0
-30	49	-29.9
-50	23	-49.5
	1	-94.3
-∞	0	-∞

Log

The Log shape of fader data has a linear relationship with gain in Decibels (dB). The basic relationship is:

$$L = 160G + 15{,}424 \qquad \text{(absolute mode)}$$

$$\Delta L = 160G \qquad \text{(relative mode)}$$

where G is the gain in dB and L is the log form of the fader data. The one exception to this formula is the case where L = 0. In this case, the gain of the mixer is reduced to minus infinity (No input audio passes to the output). To cover this special limit condition, the formula might better be written as:

for $L > 0$, $G = (L - 15{,}424)/160$

for $L = 0$, $G = -\infty$

The following table of Log value versus resulting gain was computed using the above formula (absolute mode):

Log Value (decimal)	Log Value (Hexadecimal)	Resulting Gain
16,383	3FFF	+6dB
15,424	3C40	0dB
14,464	3880	-6dB
11,584	2D40	-24dB
1	0001	-96dB
0	0000	-∞

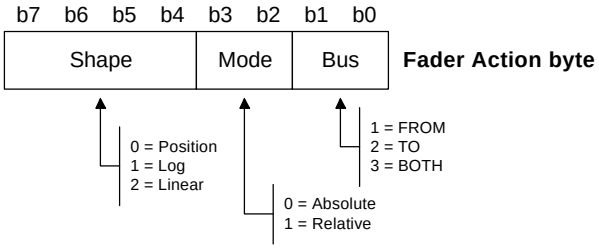
The data is transmitted in two bytes with most significant byte first. If a number greater than 3FFF is sent, the value will be saturated to 3FFF.

Linear

Not supported in this version.

2.4.2. Fader Action Byte

A Fader Action byte is part of each of the fader commands. It controls whether the shape is Position, Log, or Linear. It also controls whether the command affects FROM, TO, or both buses. Further, the value may be applied in an absolute or relative mode. The Fader Action byte is detailed below.



In the relative mode of operation, the Shape values are treated as signed numbers that adjust the current setting by that amount. If positive, the value will be adjusted upward; if negative, downward. The mixer is responsible for limiting the result of the change.

2.4.3. Fader Track (19 and 99)

Description:

This command adjusts a particular track's level on a particular machine. If the selected mode is absolute, the track is set equal to the new value. If the mode selected is relative, the offset is applied to the track.

Data:

The first byte is the machine represented as a machine byte data type described in section 2.1.1 Machine Byte.

The second byte is a track byte data type described in section 2.1.2 Track Byte.

The third byte is the Fader Action byte described in section 2.4.2 Fader Action Byte.

The level is represented by one byte of data in position mode or two bytes of data in log mode.

Write example:

Set fader A2 to unity:

Length	Bank	Command	Machine	Track	Action	Log Value
07	01	99	01	02	11	3C,40

Increase the level on track F4 by 6dB:

Length	Bank	Command	Machine	Track	Action	Log Value
07	01	99	06	04	15	03,C0

Applicable Effects Bank:

1 (Mixer).

Caveats:

None.

Related commands:

2.2.4 Special Modes (C3).

Support status:

Write only command supported at this time. Position and Log shapes supported. The TO bus fader commands are ignored if the special mode FDR SPLIT is off (see Special Modes (C3)). The TO level follows the FROM level until FDR SPLIT is turned on.

2.4.4. Track Phase Invert (9C)

Description:

This command switches in a 180º phase invert of the specified track.

Data:

The Write command has two bytes of data to specify a particular machine and track. The first byte is the machine represented as a machine byte data type described in section 2.1.1 Machine Byte. The second byte is a track byte data type described in section 2.1.2 Track Byte.

Write example:

Activate phase invert for track H2:

Length	Bank	Command	Machine	Track
04	01	9C	08	02

Applicable Effects Bank:

1 (Mixer).

Caveats:

None.

Related commands:

None.

Support status:

Write command supported.

2.4.5. Track Phase Normal (9D)

Description:

This command turns off the 180º phase invert for the specified track.

Data:

The Write command has two bytes of data to specify a particular machine and track. The first byte is the machine represented as a machine byte data type described in section 2.1.1 Machine Byte. The second byte is a track byte data type described in section 2.1.2 Track Byte.

Write example:

Deactivate phase invert for track H2:

Length	Bank	Command	Machine	Track
04	01	9D	08	02

Applicable Effects Bank:

1 (Mixer).

Caveats:

None.

Related commands:

None.

Support status:

Write command supported.

2.5. Monitor Control Commands

2.5.1. Monitor Matrix (CA) 

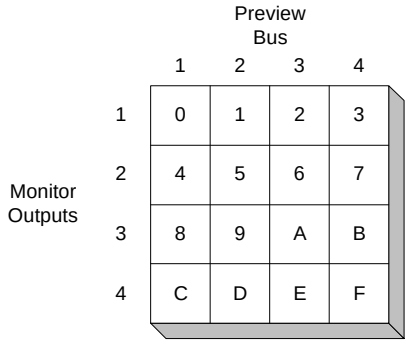
Description:

The monitor matrix is a sixteen element array of controls between the Preview bus and the monitor outputs. The D/ESAM 800 control panel makes use of this small 4X4 mixer to perform functions such as monitor level control, Balance, mono and 2 CH mode. Use this command in the absence of a control panel to access the 4X4 monitor mixer directly.

Data:

The write command has three bytes of data.

The first is the element number. Element numbers range from 0 to 15 as indicated below.



The second and third data bytes are the mix level represented as a log value. This log data has a linear relationship with gain in Decibels (dB). The basic relationship is:

L = 160G + 16,384

where G is the gain in dB and L is the log data. The one exception to this formula is the case where L = 0. In this case, the gain is reduced to minus infinity (No audio passes to the output). To cover this special limit condition, the formula might better be written as:

for L > 0, G = (L - 16,384)/160

for L = 0, G = -∞

The following table of Log value versus resulting gain was computed using the above formula:

Log Value (decimal)	Log Value (Hexadecimal)	Resulting Gain
16,384	4000	0dB
15,424	3C40	-6B
12,544	3100	-24dB
1	0001	-102dB
0	0000	-∞

The data is transmitted in two bytes; most significant byte first. If a number greater than 4000 hex is sent, the value will be saturated to 4000 hex.

Write example:

Set the monitor matrix to a unity diagonal rendering it transparent. This is what you have to do when operating the mixer in eight channel mode and you need to effectively disable the monitor mixer function:

Length	Bank	Command	Element	Value
05	01	CA	00	40,00

Length	Bank	Command	Element	Value
05	01	CA	01	00,00

Length	Bank	Command	Element	Value
05	01	CA	02	00,00

Length	Bank	Command	Element	Value
05	01	CA	03	00,00

Length	Bank	Command	Element	Value
05	01	CA	04	00,00

Length	Bank	Command	Element	Value
05	01	CA	05	40,00

Length	Bank	Command	Element	Value
05	01	CA	06	00,00

Length	Bank	Command	Element	Value
05	01	CA	07	00,00

Length	Bank	Command	Element	Value
05	01	CA	08	00,00

Length	Bank	Command	Element	Value
05	01	CA	09	00,00

Length	Bank	Command	Element	Value
05	01	CA	0A	40,00

Length	Bank	Command	Element	Value
05	01	CA	0B	00,00

Length	Bank	Command	Element	Value
05	01	CA	0C	00,00

Length	Bank	Command	Element	Value
05	01	CA	0D	00,00

Length	Bank	Command	Element	Value
05	01	CA	0E	00,00

Length	Bank	Command	Element	Value
05	01	CA	0F	40,00

Applicable Effects Bank:

1 (Mixer).

Caveats:

If used with a panel, can have unexpected results.

Related commands:

None.

Support status:

Write command Supported.

2.6. Transition Commands

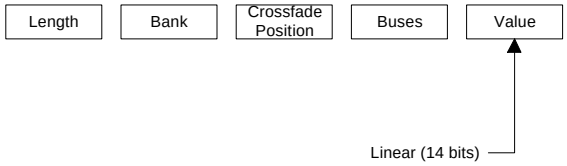
2.6.1. Crossfade Position (C4)

Description:

The Crossfade Position command is provided to accommodate control situations where the transition is not automatic but is controlled manually. For example, the transition may be controlled by a 'Tee Bar' on a master control switcher or the "Jog Knob" on a post production edit controller.

Data:

The form of the Crossfade Position command is as follows:



The value for program level is taken to be the complement of the preset level specified in this command. A value of 0x0000 hex represents full program. A value of 0x3ff0 represents full preset. The special code 0x3fff will cause a program/preset flip-flop to occur on the specified bus(es).

Write example:

Put the TO sources fully on the program outputs (buses 1, 2, 3 and 4):

Length	Bank	Command	Buses	Value
05	01	C4	F0	3F,F0

Applicable Effects Bank:

1 (Mixer)

Caveats:

Whenever the Crossfade Position is off the limit (not equal 0), auto transitions for those buses are disabled. Also, moves of a Crossfade have no effect if an auto transition is running on that bus. However, the mixer will take the position as soon as the auto transition completes.

Also, when used with D/ESAM 800 panels connected and Crossfade channels assigned, the channel status window will not indicate that the Crossfade is off the limit.

Related commands:

None.

Support status:

Write command supported.