datavideo

SE600 RS-422 Remote Control Protocol

VER: 0.1a

Release date: 2013/07/30

1. Physical layer

1.1 Control I/O format: RS-422

1.2 SE600 remote RS-422 pin definition: DSUB-9F: (Pin2=TX_P, Pin7=TX_N),
(Pin3=RX_P, Pin8=RX_N),
(Pin1, 5, 9=GND),

1.3 Communication baud rate: **115200** BPS

1.4 Data format: **8 bits** serial, **LSB** first, **1 start** bit, **1 stop** bit, **none parity.**

1.5 Please wait more 30 uSEC between two bytes when send the control command.

2. Data link layer

2.1 Frame format

1st	2nd	3rd	4th	5th	6th	7th	,,,	Last-2	Last-1	Last
Header	**ID**	**Length**	Data0	Data1	Data2	Data3	,,,	**Chksum_L**	**Chksum_H**	**End**

1) Header

The Code consisting of one byte is for frame synchronization.

The frame header send from the master machine is termed the command header.

The command header byte is fixed as fallow.

1st: F0h (base 16)

The frame header send from the slave machine is termed the *return header*.

The *return header* byte is fixed as fallow.

1st: FCh (base 16)

2) ID number

The equipment ID number is composed of 8 bits

The ID of SE600 main board: **3Ch** (base 16)

3) Length

The length is the sum of bytes from the header to the end.

It is composed of 8 bits and the length must less than **128**.

Note: **06h (base 16) < Length < = 80h(base 16)**

4) Data

Data block used by application layer.

Refer to Section 3 ~ .

5) Checksum

The 8 bits checksum is obtain from header to the last data,

and then convert to two numeric ASCII code.

Checksum=header+ID+legth+data0+data1+…+last_data

chksum_L=30h + (low nibble of checksum)

chksum_H=30h + (high nibble of checksum shift to right 4bits)

6) End

The end byte is fixed to **FFh** (base 16).

3. Application layer

The application layer designates the command structure and contents.

3.1 Command data format

4th	5th	6th	7th	8th	9th	10th	11th	…
Command Group	**Control Mode**	Operated #1	Operated #2	Operated #3	Operated #4	Operated #5	Operated #6	…

1) The command group

05h = SE600 main board control. => The follow command is for main board.

08h = SE600 Extention card control. => The follow command is for extension card.

2) The Control Mode

(for main board group only, in the other command group please set to **01h**)

00h = **ASK** status mode.

01h = **NORMAL** key mode.

02h = **T-BAR** mode.

3) The operated refer to the section 4~.

3.2 Return data format

4th	5th	6th	7th	8th	9th	10th	11th	…
Command Group	**Control Mode**	parameter #1	parameter #2	parameter #3	parameter #4	parameter #5	parameter #6	…

1) **The Command Group**

05h = SE600 main board control.

07h = SE600 main board message. => The follow parameters are ASCII strings for controller to display.

08h = SE600 Extention card control.

2) **The Control Mode**

Same as the 3.1 command data format's control mode.

3) The parameters refer to the follow description.

4. Operated

4.1 The command group = 05h (SE600 Main board control)

4.1.1 The control mode = 00h (ASK status mode): Ask the SE600 main set's status.

1st	2nd	3rd	4th	5th	6th	7th	8th
F0h Header	3Ch ID	08h length	***05h*** group	***00h*** ASK mode	39h cksm_L	33h cksm_H	FFh end

4.1.2 **The return data stream from SE600 (ASK status mode)**

1st	2nd	3rd	4th	5th	6th	7th	8th	9th
FCh	3Ch	13h length	***05h***	***00h***	Parameter #1	Parameter #2	Parameter #3	Parameter #4

10th ~ 15th	16th	17th	18th	19th
Parameter #5~10	Parameter #11	cksm_L	cksm_H	FFh

4.1.2.1 ***parameter-1*** **= busy flags**

bit0=busy in transition = '1'
bit1=busy in T-bar= '1'
bit2=busy in effect= '1'
bit3=NTSC ('0') or PAL ('1')
bit4=main source be frozen = '1'
bit5=sub source be frozen = '1'
bit6=fade to black function is active ='1'
bit7=0

4.1.2.2 ***parameter-2*** **= speed & effect level**

bit0~3= Speed No.
bit4~6= Effect level
bit7=0;

4.1.2.3 ***parameter-3*** **= current transition**

Bit6~bit0 = 0=cut, 1= fade (dissolve)
2=wipe_left_to_right, 3=wipe_right_to_left,
4=wipe_bottom_to_top, 5=wipe_top_to_bottom,
6=wipe_center_to_top/bottom, 7=wipe_top/bottom_to_center
8=wipe_center_to_left/right, 9=wipe_left/right_to_center
10=wipe_left/top_to_right/bottom, 11=wipe_right/bottom_to_left_top
12=wipe_left/bottom_to_right/top, 13=wipe_right/top_to_left/bottom
14=wipe_center_to_4coner, 15=wipe_4corner_to_center
bit7=0;

4.1.2.4 ***parameter-4*= current effect**

bit0~4=Current special effect

0 = no special,

3= Mosaic, 4= Paint, 6= black & white.

bit7=0

4.1.2.5 ***parameter-5*= current MAIN-SOURCE**

1~8=> Main source=CH1~CH8

9 => Main source=background color

10=> Main source =PIP

11=> Main source=Freeze

bit7=0

4.1.2.6 ***parameter-6*= current SUB-SOURCE**

1~8=> Sub source=CH1~CH8

9=> Sub source=background color

10=> Sub source =PIP

11=> Sub source=Freeze

bit7=0

4.1.2.7 ***parameter-7*= PIP flag**

bit0=PIP enable ('1'=enable)

bit1=PIP_1 enable ('1'=enable)

bit2=PIP_2 enable

bit4=Lumakey enable

bit5=Lumakey preview is acting

bit7=0

4.1.2.8 ***parameter-8* = some special effect's status**

bit0 = POS_CNTL on ('1'=ON)

bit2 = LOGO_2 on

bit3 = LOGO_1 on

bit4 = BDR on. (border on)

bit5 = GPI_IN on

bit6 = GPI_OUT on

bit7 = 0

4.1.2.9 ***parameter-9* = BG color & GPI_OUT action status**

bit0~3 =Background color

0=Black, 1=Blue, 2=Red, 3=Magenta, 4=Green, 5=Cyan,

6=Yellow, 7=White, 0Eh=Lines, 0Fh=Color bar

bit4=GPI_OUT is acting

bit7=0

4.1.2.10 ***parameter-10* = border style & border color**

bit0~3=border color

0=Black, 1=Blue, 2=Red, 3=Magenta, 4=Green, 5=Cyan,
6=Yellow, 7=White, 0Eh=Lines, 0Fh=Color bar

bit4~6=border style

0=border off, 1=narrow border, 1=middle border, 2=wide border

bit7=0

4.1.2.11 ***parameter-11 =*** **PIP-BACKGROUND source**

0~7=> PIP-BG =CH1~CH8

8 => PIP-BG =background color

4.1.2.12 ***parameter-12 = AUX-1 source***

Bit0~bit3= AUX-1 source

0~7=> AUX-1 source=CH1~CH8

8 => AUX-1 source=background color

14=> AUX-1 source =Main source

15=> AUX-1 source =Sub source

Bit4~bit6= PIP-1 source

0~7=> PIP-1 source=CH1~CH8

bit7=0

4.1.2.13 ***parameter-13 = AUX-2 source***

Bit0~bit3= AUX-2 source

0~7=> AUX-2 source=CH1~CH8

8 => AUX-2 source=background color

14=> AUX-2 source =Main source

15=> AUX-2 source =Sub source

Bit4~bit6= PIP-2 source

0~7=> PIP-2 source=CH1~CH8

bit7=0

4.1.2.14 ***Parameter-14 = Setting flag***

bit0=setting on (‘1’=ON)

bit1=PIP setting on

bit2=LOGO setting on

bit3=Lumakey setting on

bit4=FREEZE LOCK on

bit7=0

4.2 The control mode = 01h (NORMAL key mode)

Control the SE600 main board's function.

4.2.1 The control data stream

1st	2nd	3rd	4th	5th	6th	7th	8th
F0h	3Ch ID	0fh	05h group	01h Normal key mode	***Key Command code***	***00h***	***00h***

9th	10th	11th	12th	13th	14th	15th
00h	00h	00h	00h	cksm_L	cksm_H	FFh

4.2.2 The return data stream from SE600

The SE600 return the SE600's status same as 4.1.2 normally.

(Note: key command code from 1 to 109)

4.2.3 The Key Command Code

1(01h) = KEY_SUB_1, 2(02h) = KEY_SUB_2
3(03h) = KEY_SUB_3, 4(04h) = KEY_SUB_4
5(05h) = KEY_SUB_5, 6(06h) = KEY_SUB_6
7(07h) = KEY_SUB_7, 8(08h) = KEY_SUB_8
9(09h) = KEY_SUB_BK, 10(0Ah) = KEY_SUB_PIP
11(0Bh) = KEY_SUB_FZ, 12(0Ch) = KEY_SUB_PVW

13(0Dh) = KEY_MAIN_1, 14(0Eh) = KEY_MAIN_2
15(0Fh) = KEY_MAIN_3, 16(10h) = KEY_MAIN_4
17(11h) = KEY_MAIN_5, 18(12h) = KEY_MAIN_6
19(13h) = KEY_MAIN_7, 20(14h) = KEY_MAIN_8
21(15h) = KEY_MAIN_BK, 22(16h) = KEY_MAIN_PIP
23(17h) = KEY_MAIN_FZ, 24(18h) = KEY_MAIN_FTB

31(1Fh) = KEY_SPEED_1, 32(20h) = KEY_SPEED_2
33(21h) = KEY_SPEED_3, 34(22h) = KEY_SPEED_4
35(23h) = KEY_SPEED_5,

37(25h) = KEY_GPI_TRIGGER, 38(26h) = KEY_CUT
39(27h) = KEY_TAKE,

40(28h) = KEY_BDR, 41(29h) = KEY_BDR_COLOR

42(2Ah) = KEY_BG,

44(2Ch) = KEY_B/W, **45(2Dh) = KEY_LOGO_SET**

46(2Eh) = KEY_LUMAKEY_SET,

49(31h) = KEY_PIP_SET

50(32h) = KEY_MOSAIC, **51(33h) = KEY_PAINT**

54(36h) = KEY_C_TO_4CNR, **55(37h) = KEY_4CNR_TO_C (C=center, CNR=corner)**

56(38h) = KEY_FADE, **57(39h) = KEY_SPD (SPD=speed)**

58(3Ah) = KEY_L/T_TO_R/B, **59(3Bh) = KEY_R/B_TO_L/T (L=left, R=right, T=top, B=bottom)**

60(3Ch) = KEY_L/B_TO_R/T, **61(3Dh) = KEY_R/T_TO_L/B**

62(3Eh) = KEY_C_TO_T/B, **63(3Fh) = KEY_T/B_TO_C**

64(40h) = KEY_C_TO_L/R, **65(41h) = KEY_L/R_TO_C,**

66(42h) = KEY_L_TO_R, **67(43h) = KEY_R_TO_L**

68(44h) = KEY_B_TO_T, **69(45h) = KEY_T_TO_B**

70(46h) = KEY_LUMAKEY_PV, **71(47h) = KEY_LUMAKEY_ON**

72(48h) = KEY_PIP1_EN, **73(49h) = KEY_PIP2_EN**

74(4Ah) = KEY_LOGO1_EN, **75(4Bh) = KEY_LOGO2_EN**

78(4Eh) = KEY_LEFT_ARROW, **79(4Fh) = KEY_RIGHT_ARROW**

80(50h) = KEY_UP, **81(51h) = KEY_DOWN**

82(52h) = KEY_SETTING,

85(55h) = KEY_ENTER

90(5Ah) = KEY_AUX1_SOURCE, **91(5Bh) = KEY_AUX2_SOURCE**

92(5Ch) = SET_PIP1_SOURCE, **93(5Dh) = SET_PIP2_SOURCE**

94(5Eh) = SET_PIP_BG, **95(5Fh) = ASK_PIP_SOURCE**

109(6Dh) = KEY_FREEZE_LOCK

4.2.4 Example

Example#1: **Switch main source to CH5**

*F0h,3Ch,0Fh,05h,**01h,11h**,00h,00h,00h,00h,00h,00h,32h,35h,FFh*

KEY_MAIN_5

Example#2: **Do auto take**

*F0h,3Ch,0Fh,05h,**01h,27h**,00h,00h,00h,00h,00h,00h,38h,36h,FFh*

KEY_TAKE

Example#3: **Switch main source to CH1**

*F0h,3Ch,0Fh,05h,**01h,0Dh**,00h,00h,00h,00h,00h,00h,3Eh,34h,FFh*

Example#4: **Switch main source to CH2**

*F0h,3Ch,0Fh,05h,**01h,0Eh**,00h,00h,00h,00h,00h,00h,3Fh,34h,FFh*

Example#5: **Switch main source to CH3**

*F0h,3Ch,0Fh,05h,**01h,0Fh**,00h,00h,00h,00h,00h,00h,30h,35h,FFh*

Example#6: **Switch main source to CH4**

*F0h,3Ch,0Fh,05h,**01h,10h**,00h,00h,00h,00h,00h,00h,31h,35h,FFh*

Example#7: **Switch main source to CH5**

*F0h,3Ch,0Fh,05h,**01h,11h**,00h,00h,00h,00h,00h,00h,32h,35h,FFh*

Example#8: **Switch main source to CH6**

*F0h,3Ch,0Fh,05h,**01h,12h**,00h,00h,00h,00h,00h,00h,33h,35h,FFh*

Example#9: **Switch main source to CH7**

*F0h,3Ch,0Fh,05h,**01h,13h**,00h,00h,00h,00h,00h,00h,34h,35h,FFh*

Example#10: **Switch main source to CH8**

*F0h,3Ch,0Fh,05h,**01h,14h**,00h,00h,00h,00h,00h,00h,35h,35h,FFh*

Example#11: **Switch main source to main-PIP**

*F0h,3Ch,0Fh,05h,**01h,16h**,00h,00h,00h,00h,00h,00h,37h,35h,FFh*

Example#12: **Switch sub source to CH1**

*F0h,3Ch,0Fh,05h,**01h,01h**,00h,00h,00h,00h,00h,00h,32h,34h,FFh*

Example#13: **Switch sub source to CH2**

*F0h,3Ch,0Fh,05h,**01h,02h**,00h,00h,00h,00h,00h,00h,33h,34h,FFh*

Example#14: **Switch sub source to CH3**

*F0h,3Ch,0Fh,05h,**01h,03h**,00h,00h,00h,00h,00h,00h,34h,34h,FFh*

Example#15: **Switch sub source to CH4**

*F0h,3Ch,0Fh,05h,**01h,04h**,00h,00h,00h,00h,00h,00h,35h,34h,FFh*

Example#16: **Switch sub source to CH5**

*F0h,3Ch,0Fh,05h,**01h,05h**,00h,00h,00h,00h,00h,00h,36h,34h,FFh*

Example#17: **Switch sub source to CH6**

*F0h,3Ch,0Fh,05h,**01h,06h**,00h,00h,00h,00h,00h,00h,37h,34h,FFh*

Example#18: **Switch sub source to CH7**

*F0h,3Ch,0Fh,05h,**01h,07h**,00h,00h,00h,00h,00h,00h,38h,34h,FFh*

Example#19: **Switch sub source to CH8**

*F0h,3Ch,0Fh,05h,**01h,08h**,00h,00h,00h,00h,00h,00h,39h,34h,FFh*

Example#20: **Switch sub source to sub-BK**

*F0h,3Ch,0Fh,05h,**01h,09h**,00h,00h,00h,00h,00h,00h,3ah,34h,FFh*

Example#21: **Switch sub source to sub-PIP**

*F0h,3Ch,0Fh,05h,**01h,0Ah**,00h,00h,00h,00h,00h,00h,3bh,34h,FFh*

4.2.5 ***T-bar command***

1st	2nd	3rd	4th	5th	6th	7th	8th
F0h	**3Ch ID**	**0fh**	**05h group**	**02h T-BAR mode**	***T-BAR enable***	***T-bar value-L***	***T-bar value-H***

9th	10th	11th	12th	13th	14th	15th
00h	00h	00h	00h	cksm_L	cksm_H	FFh

T-bar value-L'b0~b4= the low 4 bits of T-bar's value

T-bar value-H'b0~b4= the high 4 bits of T-bar's value

(T-bar value => from 00h to ffh)

Example#1: **T-Bar enable and moving**

*F0h,3Ch,0Fh,05h,**02h,01h,0Lh,0Hh**,00h,00h,00h,00h,3xh,3yh,FFh*

Example#2: **T-Bar disable**

*F0h,3Ch,0Fh,05h,**02h,00h,0Lh,0Hh**,00h,00h,00h,00h,3xh,3yh,FFh*

4.2.6 AUX & PIP source command

****

Command code => **90(5Ah) = SET AUX1 SOURCE**

Parameter #1 = > 30h~37h=CH1~CH8, 38h=BG

****

Command code => **91(5Bh) = SET AUX2 SOURCE**

Parameter #1 = > 30h~37h=CH1~CH8, 38h=BG

****

Command code => **92(5Ch) = SET PIP1 SOURCE**

Parameter #1 = > 30h~37h=CH1~CH8, 38h=BG

--- Answer ---

Parameter #1 = > PIP1 source: 30h~38h

Parameter #2 = > PIP2 source: 30h~38h

Parameter #3 = > PIP_BG source: 30h~37h

****

Command code => **93(5Dh) = SET PIP2 SOURCE**

Parameter #1 = > 30h~37h=CH1~CH8, 38h=BG

--- Answer ---

Parameter #1 = > PIP1 source: 30h~38h

Parameter #2 = > PIP2 source: 30h~38h

Parameter #3 = > PIP_BG source: 30h~37h

****

Command code => **94(5Eh) = SET PIP BACKGROUND SOURCE**

Parameter #1 = > 30h~37h=CH1~CH8,

--- Answer ---

Parameter #1 = > PIP1 source: 30h~38h

Parameter #2 = > PIP2 source: 30h~38h

Parameter #3 = > PIP_BG source: 30h~37h

Example#1: **Switch PIP1 source to CH3**

*F0h,3Ch,0Ah,05h,**01h,5Ch**,32h,3Ah,3Ch,FFh*

5Ch=*SET_PIP1_SOURCE*

32h=CH-3

Answer from SE600: *FCh,3Ch,0Ch,05h,01h,5Fh,32h,31h,32h,3Eh,33h,FFh*

Example#2: **Switch PIP1 source to CH1 :** *F0h,3Ch,0Ah,05h,**01h,5Ch**,30h,38h,3Ch,FFh*

Example#3: **Switch PIP1 source to CH2 :** *F0h,3Ch,0Ah,05h,**01h,5Ch**,31h,39h,3Ch,FFh*

Example#4: **Switch PIP1 source to CH3 :** *F0h,3Ch,0Ah,05h,**01h,5Ch**,32h,3Ah,3Ch,FFh*

Example#5: **Switch PIP1 source to CH4 :** *F0h,3Ch,0Ah,05h,**01h,5Ch**,33h,3Bh,3Ch,FFh*

Example#6: **Switch PIP1 source to CH5 :** *F0h,3Ch,0Ah,05h,**01h,5Ch**,34h,3Ch,3Ch,FFh*

Example#7: **Switch PIP1 source to CH6 :** *F0h,3Ch,0Ah,05h,**01h,5Ch**,35h,3Dh,3Ch,FFh*

Example#8: **Switch PIP1 source to CH7 :** *F0h,3Ch,0Ah,05h,**01h,5Ch**,36h,3Eh,3Ch,FFh*

Example#9: **Switch PIP1 source to CH8 :** *F0h,3Ch,0Ah,05h,**01h,5Ch**,37h,3Fh,3Ch,FFh*

Example#10: **Switch PIP background to CH1 :** *F0h,3Ch,0Ah,05h,**01h,5Eh**,30h,3Ah,3Ch,FFh*

Example#11: **Switch PIP background to CH2 :** *F0h,3Ch,0Ah,05h,**01h,5Eh**,31h,3Bh,3Ch,FFh*

Example#12: **Switch PIP background to CH3 :** *F0h,3Ch,0Ah,05h,**01h,5Eh**,32h,3Ch,3Ch,FFh*

Example#13: **Switch PIP background to CH4 :** *F0h,3Ch,0Ah,05h,**01h,5Eh**,33h,3Dh,3Ch,FFh*

Example#14: **Switch PIP background to CH5 :** *F0h,3Ch,0Ah,05h,**01h,5Eh**,34h,3Eh,3Ch,FFh*

Example#15: **Switch PIP background to CH6 :** *F0h,3Ch,0Ah,05h,**01h,5Eh**,35h,3Fh,3Ch,FFh*

Example#16: **Switch PIP background to CH7 :** *F0h,3Ch,0Ah,05h,**01h,5Eh**,36h,30h,3Dh,FFh*

Example#17: **Switch PIP background to CH8 :** *F0h,3Ch,0Ah,05h,**01h,5Eh**,37h,31h,3Dh,FFh*

Example#18: **Switch PIP2 source to CH1 :** *F0h,3Ch,0Ah,05h,**01h,5Dh**,30h,39h,3Ch,FFh*

Example#19: **Switch PIP2 source to CH2 :** *F0h,3Ch,0Ah,05h,**01h,5Dh**,31h,3Ah,3Ch,FFh*

Example#20: **Switch PIP2 source to CH3 :** *F0h,3Ch,0Ah,05h,**01h,5Dh**,32h,3Bh,3Ch,FFh*

Example#21: **Switch PIP2 source to CH4 :** *F0h,3Ch,0Ah,05h,**01h,5Dh**,33h,3Ch,3Ch,FFh*

Example#22: **Switch PIP2 source to CH5 :** *F0h,3Ch,0Ah,05h,**01h,5Dh**,34h,3Dh,3Ch,FFh*

Example#23: **Switch PIP2 source to CH6 :** *F0h,3Ch,0Ah,05h,**01h,5Dh**,35h,3Eh,3Ch,FFh*

Example#24: **Switch PIP2 source to CH7 :** *F0h,3Ch,0Ah,05h,**01h,5Dh**,36h,3Fh,3Ch,FFh*

Example#25: **Switch PIP2 source to CH8 :** *F0h,3Ch,0Ah,05h,**01h,5Dh**,37h,30h,3Dh,FFh*

4.3 The Main Board Extension Command Code

* Control the SE600 main board's function.

* The control mode = 01h (NORMAL key mode)

4.3.1 The control data stream

1st	2nd	3rd	4th	5th	6th	7th	8th
F0h	3Ch ID	xxh length	05h group	01h Normal key	***Extension Command <u>code</u>***	***Parameter #1***	***Parameter #2***

9th,,,	Last-2	Last-1	Last
Parameter #,,,	cksm_L	cksm_H	FFh

4.3.2 The Main Board Extension Command Code

117 = EXT_ASK_MAINBLARD_VERSION_CMD

118 = EXT_GET_CURRENT_USER_CMD

119 = EXT_RECALL_USER_PARAMETER_CMD

120 = EXT_STORE_USER_PARAMETER_CMD

121 = EXT_RESET_ALL_CMD

;;;

141 = EXT_TRANSITION_SPEED_CMD,

4.4.1 The Extension Main Bard Command for Memory

****

Command code => **117(75h) = EXT_ASK_MAINBOARD_VERSION_CMD**

Parameter #1 = > NC

--- Answer ---

Parameter #1~5 = > C1 version ASCII code

Parameter #6~10 = > F1 version ASCII code

Parameter #11~15 = > F2 version ASCII code

****

Command code => **118(76h) = EXT_GET_CURRENT_USER_CMD**

Parameter #1 = > NC

--- Answer ---

Parameter #1 = > Current User No.: 31h~33h (3 users)

****

Command code => **119(77h) = EXT_RECALL_USER_PARAMETER_CMD**

Parameter #1 = > User No.: 31h~33h (3 users),

--- Answer ---

Parameter #1 = > User No.: 31h~33h (3 users)

****

Command code => **120(78h) = EXT_STORE_USER_PARAMETER_CMD**

Parameter #1 = > User No.: 31h~33h (3 users)

--- Answer ---

Parameter #1 = > User No.: 31h~33h (3 users)

****

Command code => **121(79h) = EXT_RESET_ALL_CMD**

Parameter #1 = > NC

--- Answer ---

Parameter #1 = > Current User No.: 31h~33h

4.4.2 The Extension Main Bard Command For Speed key Setting

****

Command code => **141(8Dh) = EXT_TRANSITION_SPEED_CMD**

Parameter #1 = > Get/Set: 30h=get, 31h=set (or 00h=get/01h=set)

Parameter #2 = > 30h=reset to default, 31h=+1 frame, 32h= -1 frame, 33h=set by parameter#4 (or 00h,01h,02h,03h)

Parameter #3 = > Speed key No.: 31h~35h(or 01h~05h): Speed#1~Speed#5

When (para#1=31h and papa#2=33h)

Parameter #4 = > Transition duration: 1~90(5Ah) frames

--- Answer ---

Parameter #1 = > Get/Set: 30h=get, 31h=set (or 00h=get/01h=set)

Parameter #2 = > 30h=reset to default, 31h=+1 frame, 32h= -1 frame, 33h=set by parameter#4 (or 00h,01h,02h,03h)

Parameter #3 = > Speed key No.: 31h~35h(or 01h~05h): Speed#1~Speed#5

Parameter #4 = > Transition duration: 1~90(5Ah) frames

Example: Set the speed#2 to 10 frames.

The control data stream: *F0h,3C**h,0Dh**,05h,01h**,8Dh,01h**,03h,02h,0Ah,**3ch,3dh,FFh*

F0h, ID(3Ch) , Length(0Dh), 05h, 01h, Command code(8Dh),para#1(01h),para#2(03h), para#3(02h) , para#4(0Ah), checksum_L(3ch), checksum_M(3dh), FFh

Checksum=F0h+3Ch+0Dh+05h+01h+8Dh+01h+03h+02h+0Ah=dch

The Answer from SE600 = *FCh,3C**h,0Dh**,05h,01h**,8Dh,01h,03h,02h,0Ah,**38h,3Eh,FFh*

---- Or

The control data stream: *F0h,3C**h,0Dh**,05h,01h**,8Dh,31h**,33h,32h,0Ah,**3ch,36h,FFh*

F0h, ID(3Ch) , Length(0Dh), 05h, 01h, Command code(8Dh),para#1(31h),para#2(33h), para#3(32h) , para#4(0Ah),checksum_L(3ch), checksum_M(36h), FFh

Checksum=F0h+3Ch+0Dh+05h+01h+8Dh+31h+33h+32h+0Ah=6ch

The Answer from SE600 = *FCh,3C**h,0Dh**,05h,01h**,8Dh,31h,33h,32h,0Ah,**38h,37h,FFh*