

## CMX RS-232 control guide

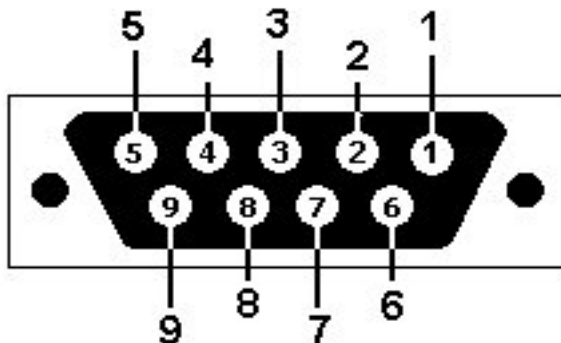
Revision 3.0  
3/12/12

### Overview

The RS-232 port on the CMX series switches is a flexible interface which can easily control all aspects of the device operation. Many system integrators prefer RS-232 control over IR because it allows the controlling device to monitor the status of the CMX switch. The CMX series switches also feature a feedthrough RS-232 port which can allow multiple devices to be controlled from a single RS-232 source.

### Hardware specifics

The CMX switch uses a DB 9 female connector (shown below) for connection to the controller.



**DB9: View looking into female connector**

Only 3 pins on the connector are needed:

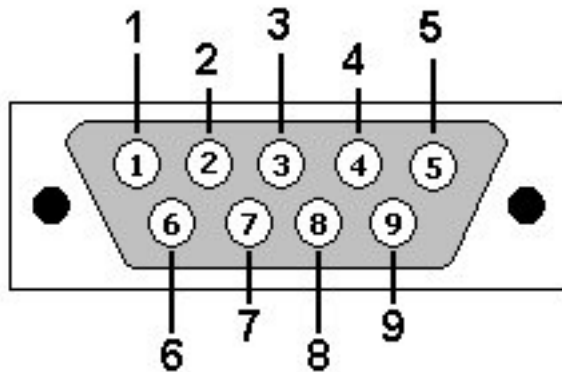
Pin 2: TX (from CMX to controller)

Pin 3: RX (from controller to CMX)

Pin 5: Ground

The cable used should be a standard straight through RS-232 cable with a DB9 male plug to DB9 or 25 female plug (also known as a serial modem cable).

The CMX switch uses a DB 9 male connector (shown below) for the expansion port.



### **DB9: View looking into male connector**

Only 3 pins on the connector are needed:

Pin 2: RX (from downstream to CMX)

Pin 3: TX (from CMX to downstream)

Pin 5: Ground

The cable used should be a standard straight through RS-232 cable with a DB9 female to DB9 or 25 male plug (also known as a serial modem cable).

## Protocol settings

Baud rate : 9600  
Data bits: 8  
Stop bits: 1  
Parity: None  
Flow control: None

## Commands

CMX will echo back all characters sent to it. This is the easiest way to verify if your cable connection is correct. The CMX does not add <lf> after any <cr> received, so if you are using windows Hyper-terminal you should change the default settings to allow line feed on carriage return.

All commands are terminated by <cr> (carriage return, ascii code 0xD). **NOTE: all references to <cr> in this document mean the single ascii character NOT the four characters “<cr>”.** Commands can be forwarded to the expansion port by adding a prefix “/F” to each command (see section below).

### **Selecting input “00” will disable the selected output.**

The following are the valid command sets for all CMX devices. Different models can be differentiated by the controller by the status readback.

Command set for config controls:

CF##<cr> : Config control

1<sup>st</sup> # selects which control you will change

2<sup>nd</sup> # F control off, T turns control on

Control 1: Clear ZeroKey IR codes & config bits (T runs clear, auto sets back to F)

Control 2: LED off

Control 3: Power off

Control 4: Advanced scheme enable (no effect on CRM1616x)

Control 5: Debug control 1 (leave as “F”)

Control 6: Debug control 2 (leave as “F”)

STAT<cr> : Request device status

STATL<cr> : Request device status of audio DSP settings

STATAUDIO<cr>: Formatted status readback of all audio settings

Faab<cr>: Flash modes control

aa is a number 01-08 selects the flash slot to control

b is either S or R (S saves current config, R recalls saved config)

Zaahhhh<cr>: Zone restriction control

aa is a number 01-16 selects the zone

hhhh is a 4 digit hexadecimal number corresponding to the ZONE MASK

Each bit of the MASK allows you to restrict an input

0 = input restricted

1 = input allowed

The LSB of the word is input 0, the MSB is input 16

AV Command set (valid in basic mode): (Note CMX1616V/A do not support this mode)

Vabc<cr> : All output control

a is a number 1-8 selects the desired output to control

b is a number 0-8 selects the input to switch the output to

c is either "D" or "O": D selects digital coax, O selects digital toslink

AV Command set (advance mode):

Note that CMX1616V only supports V,C,O controls

CMX1616A only supports A,P,C,O controls

Vaabb<cr> : Component video output control

aa is a number 01-16 selects the desired output to control

bb is a number 00-16 selects the input to switch the output to

Aaabb<cr> : Audio output control

aa is a number 01-16 selects the desired output to control

bb is a number 00-16 selects the input to switch the output to

Paabb<cr> : Composite video output control

aa is a number 01-16 selects the desired output to control

bb is a number 00-16 selects the input to switch the output to

Cabc<cr> : Digital audio coax output control

a is a number 1-8 selects the desired output to control

b is a number 0-8 selects the input to switch the output to

c is either "D" or "O": D selects digital coax, O selects digital toslink

Oabc<cr> : Digital audio toslink output control

a is a number 1-6 selects the desired output to control

b is a number 0-8 selects the input to switch the output to

c is either "D" or "O": D selects digital coax, O selects digital toslink

Caabb<cr> : Digital audio coax output control (CMX3838A2)

aa is a number 01-16 selects the desired output to control

b is a number 0-38 selects the input to switch the output to

Oaabb<cr> : Digital audio toslink output control (CMX3838A2)  
aa is a number 01-06 selects the desired output to control  
bb is a number 0-38 selects the input to switch the output to

Naabb<cr> : Network audio switch control (NAB100)  
aa is a number 01-08 selects the desired output to control  
bb is a number 0-38 selects the input to switch the output to

Audio volume Command set : (only for CMX84,86,88,1616A)

Gaabb<cr> : Audio output gain control  
aa is a number 01-16 selects the desired output to control  
bbb is a number 00-255 selects the output gain  
For bbb = 000 output is muted  
Else gain(dB) =  $31.5 - [0.5 * (255 - bbb)]$   
NOTE: on firmware versions 1.5 and up bbb may also be  
bbb = U Volume steps up  
bbb = D Volume steps down  
bbb = MT Mute on  
bbb = MF Mute false  
bbb = M Mute toggle

Eaabb<cr> : Audio input gain control  
aa is a number 01-MAX selects the desired input zone to control  
bbb is a number 00-255 selects the input gain  
For bbb = 128 input is nominal  
Else signal = signal \* bbb/128

Maabb<cr> : Audio output balance control  
aa is a number 01-MAX selects the desired input zone to control  
bbb is a number 00-255 selects the balance setting  
For bbb = 128 input is nominal  
if (bbb > 128) then right\_vol = right\_vol \* (256 - bbb) / 128  
else left\_vol = left\_vol \* (bbb) / 128

Baabb<cr> : Audio bass control  
aa is a number 01-MAX selects the desired input zone to control  
bbb is a number 00-255 selects the input gain  
For bbb = 128 input is nominal  
Else signal = signal \* bbb/128

Taabb<cr> : Audio treble control  
aa is a number 01-MAX selects the desired input zone to control  
bbb is a number 00-255 selects the input gain  
For bbb = 128 input is nominal

Else signal = signal \* bbb/128

Xaabbb<cr> : Audio input delay control

aa is a number 01-16 selects the desired input to control

bbb is a number 00-255 selects the input delay

Each delay step is 0.25 video frames (60Hz), or 4.1666ms

bbb = U Delay steps up

bbb = D Delay steps down

Daabbb<cr> : Audio output delay control

aa is a number 01-16 selects the desired output to control

bbb is a number 00-255 selects the output delay

Each delay step is 0.25 video frames (60Hz), or 4.1666ms

bbb = U Delay steps up

bbb = D Delay steps down

BPFbbb<cr> : Audio bass pole frequency

bbb is a number 00-255 adjusts the bass pole frequency (128 nominal)

RTaa<cr> : Audio internal ramp speed

aa is a number 00-06 adjusts the audio ramping speed

00 is the slowest and 06 is the fastest

## Response strings

CMX devices have two possible response strings:

1. **ACKNOWLEDGE** (OK<cr>) : This is sent to indicate the previous command was executed
2. **STATUS** This is sent whenever explicitly requested by STAT command
3. **NOT ACKNOWLEDGE** (NAK<CR>) This is returned when an invalid command is received
4. **ACKNOWLEDGE for volume** (OK:###(M):##<cr>) : When sending a volume command, the CMX will return the current setting after the OK. If the "M" is present, that output is currently muted. 2<sup>nd</sup> ## is the zone number.
5. **Input status notification:** Whenever input status changes, CMX will send the following string (note STATUS command will also return this)

IShhhhhh<cr>

hhhhhh : Six hexadecimal strings corresponding to the input state of all inputs. 1 is locked/present, 0 is no signal. Input 1 is bit 0 of the last character.

The format of the status response is:

```

Video Storm LLC CMX##### switch (with HDMI##-##)<cr>
Version 1.X<cr><cr>
V01bb<cr>
V02bb<cr>
(V03-Vmax)           Same as above
A01bb<cr>
(A02-Amax)
P01bb<cr>
(P02-Pmax)
C1bc<cr>
(C2-Cmax)
O1bc<cr>
(O2-Omax)
G01ddd(M)<cr>
(G02-Gmax)
E01ddd<cr>
(E02-Emax)
H01bb<cr>
(H02-Hmax)
CF2#<cr>
CF3#<cr>
CF4#<cr>
CF5#<cr>
CF6#<cr>

```

The first line will give information on the model number and if there is a support HDMI matrix attached. The numbers following CMX indicate the model number. The numbers following the HDMI string are the number of outputs and inputs on the attached HDMI matrix. If no HDMI matrix is attached, the HDMI string in parenthesis will not be sent.

The format of the status strings is the same as the command strings before. V is video, A is analog audio, P is composite video, C is digital audio coax, O is digital audio toslink, G is audio gain, E is audio equalizer, H is HDMI, and CF is config.

**Each model will only send status on the supported outputs.** For example, CRM1616A will only send P, A, G, E, and CF. H will be sent if a supported HDMI switch is connected.

“bb” denotes the input selected (0-16). For digital audio, the “c” is D for digital coax or O or digital optical. For audio gain “ddd” denotes the gain setting (0-255).

Example 1: V0304 => Component video output 3 has input 4 selected

Example 2: C16O => Coax audio output 1 has toslink audio input 6 selected

The config settings are interpreted as follows:

C2T    C2F	LED on when F
C3T    C3F	Power on when F
C4T    C4F	User mode when F, Advanced mode when T
C5T    C5F	Debug setting 1
C6T    C6F	Debug setting 2

## Expansion port

The expansion port allows three possible connections:

1. Connecting multiple CMX devices together to be controlled by a single rs-232 port.
2. Connecting CMX supported HDMI matrix switches which will be controlled by the CMX device
3. Connecting other generic rs-232 controlled equipment which will be controlled by the host controller in pass through mode.

You can combine uses 1 & 2 or 1& 3. However, the generic equipment or HDMI switch must be connected to the last CMX unit and therefore terminates the chain.

In order to control the connected equipment, the host controller must know which devices are connected and their ordering. A simple algorithm to detect this is shown in the last section (Recommended controller driver protocol). Otherwise it can be hard coded if the configuration will not change.

The devices in the rs-232 should be numbered 0 to X-1, where X is the number of attached devices. Device 0 is addressed using the normal protocol above. All other devices are addressed by adding a prefix to their command string. The FORWARD\_COMMAND prefix is “/F” (two ascii characters). When a device receives this string immediately after a <cr>, it will forward all the following characters to the next device in the chain until it receives a <cr> (the <cr> will also be forwarded, but the first “/F” will not). Therefore, you address each device as follows:

Device 0 : normal commands  
 Device 1 : “/F” and then normal command (terminated by single <cr>)  
 Device 2 : “/F/F” and then normal command (terminated by single <cr>)  
 Same upto X-1

If present, the last HDMI device does not need to be directly controlled. It will be controlled by the last CMX in the chain.

## Recommended controller driver protocol



The following is pseudo code for a generic CMX driver which auto detects the full feature set of the connected equipment. If you are developing a fixed feature driver then this is not necessary.

```

// Connected_eq array holds to device network attached to the rs-232 port
// Connected_eq[0] is the first CMX device, Connected_eq[15] would be the 16th
// 0 = no device
// 1 = CMX44
// 2 = CMX84
// 3 = CMX86
// 4 = CMX88
// 5 = CMX1616Video
// 6 = CMX1616Audio
// 7 = supported HDMI slave device (network termination)
// 8 = generic device (network termination)

Dim int[16] Connected_eq = 0;

// at powerup, first discover the network
In_net = true
Index = 0
While (in_net){
  Cmd = repeat("/F",index) . "STAT<CR>"
  Send CMD
  Ret_string = returned status from rs232 // this string will be multiple lines of data
  If ( "CMX44" is found in first line) then Connected_eq[index] = 1
  ElseIf ( "CMX84" is found in first line) then Connected_eq[index] = 2
  ElseIf ( "CMX86" is found in first line) then Connected_eq[index] = 3
  ElseIf ( "CMX88" is found in first line) then Connected_eq[index] = 4
  ElseIf ( "CMX1616V" is found in first line) then Connected_eq[index] = 5
  ElseIf ( "CMX1616A" is found in first line) then Connected_eq[index] = 6
  ElseIf ( Ret_string is not NULL) then Connected_eq[index] = 8
  Else In_net = false // end of network
  If (in_net) then {
    // detect HDMI connection
    If ("HDMI" is found in first line of return string) then {
      Connected_eq[index+1] =7
      In_net = false
      // discover number of HDMI inputs/outputs
      Pattern match on return string "HDMI##-##" // ## will be 1-16
      HDMI_inputs = 2nd ##
      HDMI_output = 1st ## // Last CMX unit will control HDMI
    }
  }
}
}

```

Example to control CMX in position 0: cmd="V11C<CR>"

Example to control CMX in position 3: cmd="/F/F/FV11C<CR>"

Example to control generic equipment in position 1:

cmd="/F" . one\_line\_of\_command\_terminated\_by\_<CR>