



Vaux Protocol Specification

a software developer's guide for serial control of Vaux systems

Version 3.13

August 1997

Publication VP-04

Vaux Electronics, Inc., based in Arizona, designs, manufactures and markets innovative electronic products for the consumer audio/video, home control, and disabled persons marketplaces. We are committed to providing state-of-the-art systems which fully integrate the electronic and electrical components in the home — and to delivering high-quality, user-friendly, and affordable products. Vaux Products are made in the U.S.A.

©1993-1997 Vaux Electronics, Inc.

Printed in the U.S.A.

All rights reserved.

Conventions used in this manual:

Note: Numeric values in this document may appear with or without a suffix character:

254	No suffix: decimal
254d	'd' suffix: decimal
10110001b	'b' suffix: binary
6FC5h	'h' suffix: hexadecimal
69bcd	"bcd" suffix: binary-coded-decimal (hex, without the A-F values)
dd.0, dd.1...	Refers to bit 0, 1..7 of byte dd

Altrix, Aris, MARC+, MediaMation, Vaux, VauxConfig, VauxControl, VauxNet, VauxSerial, VauxTouch, Vaux Protocol, and VoiceOver are trademarks of Vaux Electronics, Inc.

Other trademarks are owned by their respective companies.

preface

This document is a specification of the public-domain Vaux Protocol format — a Vaux-defined command language for computer interface and control of MediaMation systems such as Vaux's Aris System, and control systems, such as Vaux's Altrix System.

This document may be reproduced or copied — provided that this document is reproduced in its entirety and all copyright notices are intact. This document is believed to be accurate at the time of publication, however, Vaux Electronics makes no claim that this document is error-free.

Please direct all inquiries to:

Vaux Electronics, Inc.

Arizona, USA

PHONE: (480) 354-5556

FAX: (480) 354-5558

tech@vauxelectronics.com

www.vauxelectronics.com

contents

preface.....	3
contents.....	4
introduction.....	6
integration with touchscreen control systems.....	6
integration with lighting control keypads.....	6
a note on interface design.....	6
controller programming.....	7
backup and archiving.....	7
vauxserial interfaces.....	8
vaux protocol.....	9
first, a note on upgradability.....	9
also, a note on software revisions.....	9
interfacing to a Vaux system.....	9
languages, operating systems, and characters	11
vaux protocol command syntax.....	12
overview.....	12
generating the commands.....	13
standard (fixed-length) control command.....	15
extended (variable-length) command	15
vaux protocol message syntax.....	16
overview.....	16
informational messages.....	17
error messages.....	18
event messages	19
request messages.....	21
using the system.....	22
setting baud rate.....	22
setting configuration bytes.....	22
security ID (SecID) code	26
request command.....	27
system control command	29
using a zone group	31
the source-selection process.....	32
Auto-Source-Power Option.....	32
Auto-Receiver-Power Option.....	32
Auto-TV-Power Option.....	33
Auto-PLC-Source-Power Option.....	33
Auto-PLC-Zone-Power Option.....	34
System-Active-PLC-Power Option.....	34
Source-Select MACROs.....	34
setting source-select mode.....	35
setting A/V sources for auto-TV-powering.....	36
setting auto-PLC-powering options.....	37
infrared control (learned IR output).....	38
lighting and appliance (X-10®) control.....	40

controlling an X-10 device.....	40
adjusting X-10 dim levels.....	40
controlling an X-10 device using the low-level commands.....	41
X-10 input (2-way X-10).....	42
X-10-Activity event messages:.....	44
Aris MediaMation audio/video control (Aris only).....	46
using the zone turn-on/off default table.....	47
getting an A/V zone's status.....	49
source-selection (source-to-zone routes).....	50
relative adjustment of volume/bass/treble levels.....	51
absolute adjustment of volume/bass/treble levels.....	52
zone muting and flattening.....	53
adjusting VBT ramp rate for remote controls.....	54
HVAC devices.....	55
security devices:.....	56
MACROs.....	57
appendix 1: vaux protocol device codes.....	60
appendix 2: vaux protocol function codes.....	62
appendix 3: vaux protocol zone codes.....	65
appendix 4: Aris MediaMation control codes.....	67
appendix 5: an interactive Vaux Protocol example.....	69
appendix 6: vauxserial port rs-232 interface.....	70
rs-232 connection to the vauxserial port.....	70
vauxserial port rs-232 interface (DB-25 F) connector.....	71
vaux-to-IBM PC®/terminal (DB-25) cable connections.....	72
vaux-to-IBM AT-style (DB-9) cable connections:.....	72
vaux-to-Macintosh® (DIN-8) cable connections:.....	73
vaux-to-modem (DB-25) cable connections:.....	73
VauxSerial Port Handshaking:.....	74
RS-232 General Information:.....	76
appendix 7: VauxSerial ASCII (receive) character set.....	77
appendix 8: obsoleted commands & messages.....	79
appendix 9: revision history.....	80
appendix 10: zone- v.s. device-specific IR outputs.....	83
appendix 11: updating controller software.....	84
appendix 12: glossary.....	86

introduction

Public, external control of a Vaux control system is possible via a serial RS-232 connection. A Vaux system controller, such as an Aris MediaMation or Altrix Controller, may be connected to an external system such as an AMX, Phast, or Crestron control system, or a personal computer running a specific control program. Vaux Protocol is the command language which allows this control connection.

integration with touchscreen control systems

Vaux systems may be controlled through the serial RS-232 port. A typical example is the use of Aris Controllers with an AMX, Phast, or Crestron control system, to provide multi-source/multi-zone audio/video distribution to various rooms in a home. The control system may be programmed to display source-selection, volume, bass, treble, and mute functions, on touchscreen panels in various rooms — each touchscreen panel may display controls specific to just that zone, or they may be designed to allow access to multiple zones in the home. “Scene” buttons, such as “Entertain,” may configure whole-house audio/video selections in addition to adjusting lighting levels, etc. This type of integration is easy to implement, by simply programming the control system to send the appropriate Vaux Protocol commands in response to touchscreen button-presses, or as part of complex control sequences.

Touchscreen controls for volume/bass/treble may use drag sliders, up/down buttons with optional level-graph display, etc. If the system will also use Vaux RF or IR Remote Controls, the computer control system may periodically poll the Aris Controller for zone status information, and update sliders and other controls on touchscreens around the home. For additional flexibility, the control system may listen for specific button-presses from Vaux RF or IR remote controls or wired or wireless keypads, in order to use them as input devices. This type of integration requires additional programming to parse input streams, but is straightforward to implement.

integration with lighting control keypads

A variety of home lighting systems are available which use custom-engraved keypads. It is possible to define buttons on these keypads for access to non-lighting systems, such as whole-house audio/video control. If the lighting system allows programmable RS-232 controls strings, an Aris Controller may be directly connected to the system, and controlled from keypad buttons which invoke the appropriate Vaux Protocol commands. For lighting systems which do not allow RS-232 programmability, but do send button-press information, an AMX, Phast, or Crestron control system may provide the mapping between keypad buttons and Aris A/V control commands.

a note on interface design

When Aris A/V controllers are integrated into a larger system employing a touchscreen (or even mouse-driven computer) control system, consideration must be given to the interface design. In the case of on/off functions, such as source/zone selections, or lighting on/off, simple buttons are generally used. However, for variables such as volume/bass/treble levels, you have a choice of control type: typically, you will use sliders or up/down buttons.

If you use slider controls, you should consider whether you wish continuous adjustment as the user drags the slider knob (this would require real-time relative up/down commands to the Vaux controller), or to simply set the volume level when the knob is released (this would use a single absolute level adjustment command, generally with level-ramping applied). If you use relative adjustments, be sure to employ an appropriate hysteresis window to prevent excessive up/down command traffic.

A hysteresis window may be implemented by the control itself, or it may be added in software. The window size is selected to filter out small variations in the opposite direction of travel. For example, if the control has started moving in the Volume-Up direction, you will issue the 2-dB up commands as necessary (based on the 0 to 100% scaling you select for the slider), but ignore downward movements which are smaller than the window you have defined. If the window is too small, you will notice a jittery up/down volume level as the slider is moved slowly, which can be compounded by filling the RS-232 buffer with excessive commands. If the window is too large, you will need to drag the slider an excessive amount before the opposite-direction commands kick in. Using an adjustable window not only allows you to fine-tune the feel of the slider, but it also provides a way to maintain consistent feel as the control is resized (by applying a scale factor to the hysteresis window parameter when the control is resized).

Using level up/down buttons is an easier approach, since the press-and-hold input allows you to send a stream of up or down commands (at a ramp rate you select) until the button is released. You may optionally show a percentage gauge or dB-level display.

controller programming

Vaux system controllers may be programmed using remote controls or through the serial RS-232 port. The primary programming of a Vaux system may be performed using a Vaux remote control (e.g. infrared learning and macro programming), but some secondary programming tasks must be performed using a computer (e.g. configuring infrared output ports for zone-specific operation). Vaux Protocol defines commands for programming all system features.

backup and achiving

Vaux Protocol provides commands for upload and download of controller programming data such as learned infrared control codes, macro sequences, and A/V zone parameters. You may design a control program to build a library of frequently-used infrared devices, or to backup each client's entire system configuration. A control system can initialize a Vaux Controller to your specific configuration needs — for example, if you have written a program for controlling an Aris Controller in a retail store installation, you may quickly download your system configuration into each Aris Controller before shipping the systems off.

vauxserial interfaces

The Vaux RS-232 interface connector is referred to as a *VauxSerial Port*. There are two types of VauxSerial-controlled systems:

- 1) Two-way direct connection: Aris and Altrix Controllers have the serial interface built in to the main control system, and have a VauxSerial Port on the rear panel. These direct (non-RF) interfaces provide two-way communication with the host computer. That is, in addition to accepting control commands, these direct-connect interfaces can also provide the computer with event activity (e.g. a button press on a Remote Control), and additional system Request Messages.
- 2) One-way RF link: The Vaux PCremote accessory is an RS-232-connected black box which essentially turns a computer into a Vaux RF Remote Control. PCremote allows the computer to send all data byte combinations over the RF link, covering all current and future command definitions.

This specification defines the master set of Vaux Protocol Commands and Messages, however, a specific Vaux product may only recognize a subset of the protocol (for example, the Altrix system does not recognize the Aris A/V routing commands). We will note product-specific differences as they appear.

caution

Please note that not all RS-232 ports adhere to the electrical standard, as does a VauxSerial interface from Vaux. A true RS-232 port can withstand having any pin shorted to any other pin without damage — be careful, since your computer may not abide by this rule. See the *VauxSerial Port RS-232 Interface* appendix for more information on RS-232 interfacing.

Vaux Electronics, Inc. assumes no responsibility for any damage (except as outlined in the appropriate product warranties) due to the connection of incompatible equipment, or the improper connection of compatible equipment, to your VauxSerial Port.

first, a note on upgradability

Vaux Aris and Altrix controllers are upgradable. That is, new system software may be downloaded into the controllers using a personal computer and appropriate serial cable. Vaux Protocol versions indicated in this document are the same as controller system software versions — if you have an earlier system, and would like to use a feature added to a later revision, you may obtain a Controller Software Upgrade disk to upgrade your system, or have the factory upgrade the system for you. Contact Vaux for pricing and additional information. See the *Revision History* appendix for an overview of improvements. See the *Upgrading Controller Software* appendix for a information on the update procedure.

also, a note on software revisions

Vaux Protocol is designed to be extensible, with the addition of new commands, as needed by new features and/or controllers. We strive to maintain the highest reasonable level of compatibility with software interfaces written by third-party developers — you should not have to change your code frequently to adapt to new Vaux software revisions. There are a couple of exceptions we have made to this policy, the most notable being a change from a factory default setting of 2400 baud to 9600 baud. This was in response to developer requests; we also allow the systems to be programmed to other baud rates. There are a couple of instances where a newer (preferred) command has been introduced, although the earlier command is still accepted. There have also been a couple of cases where protocol changes have been necessary — see the *Obsolete Commands & Messages* appendix for more information.

As always, if you have a desire for a feature you would like to see added, let us know. The Vaux Protocol interface is for your benefit, and we implement good ideas whenever possible.

interfacing to a Vaux system

Vaux Protocol is intended for interfacing to a computer program or external control system such as an AMX, Phast or Crestron control system; therefore, a primary design goal is ease of generation and parsing of the Command and Message data by popular development languages. A secondary feature is the ability to use an ASCII terminal for testing or manual control. The design criteria included: I/O over a serial data stream, short commands and messages which can be easily generated and parsed by simple application coding, provision for command checksums (and the ability to disable them), definition of various levels of messages (and the ability to selectively disable them), independence from various white space, newline, and control characters (space, tab, CR, LF...), as well as a comprehensive (and extensible) control command and message structure.

Information transfer between a controlling computer and the VauxSerial Port is in the Vaux Protocol format — a Vaux-defined command language. The Vaux Protocol format is comprised of various serial information streams, (i.e. short sequences of ASCII characters).

A Control Command is initiated by the computer and generally requests that some sort of action be taken by the control system — for example, to press Play on a VCR (send an infrared control code), or to Turn On Porch Light (via an X-10® module). Some commands obtain a reply via a Request Message and do not result in an action. There are also Event Messages sent by the control system; for example, to announce that a specific Vaux Remote Control button was pressed.

The host computer will initiate a control system action by sending a Vaux Protocol Command. For example, the Vaux Command stream (of ASCII characters):

VP 0A 87 14 00

will control X-10 (device 0Ah) module E2 (zone 87h) and turn it On (function 14h).

The Vaux Command stream:

VP 02 00 0D 00

will control the CD player (device 02h), and press Play (function 0Dh).

The syntax of the Vaux Command streams is presented shortly. The basic scheme defines a header (the VP characters above), and some bytes of control information, formatted as two ASCII characters (representing the hex value) per data byte. The application program does not require complex coding in order to drive the Vaux system.

languages, operating systems, and characters

The Vaux Protocol Commands will be sent, from an application program, out the Serial Port of the computer. This program may be in any language that provides serial output capability (for example, C, BASIC, or assembler may be used, or perhaps an object-oriented or graphical programming environment such as Visual Basic for Windows, or HyperCard for Macintosh). The definition of the Vaux Protocol Commands are simple enough to allow easy integration into any target application language. Note that serial port control may not be immediately available from your programming environment; some compilers require the inclusion (and possible purchase) of optional libraries or objects which perform the port configuration and handle the serial data stream.

Operating system differences should be transparent to the Vaux Controller. The primary differences in the serial data streams relate to the various end-of-line (or “newline”) characters used:

- Macintosh® computers use a CR (Carriage Return) character (0Dh)
- UNIX™ systems use an LF (Line Feed) character (0Ah)
- IBM PC® computers use a CR (0Dh) followed by an LF (0Ah)

Since both CR and LF characters are ignored by the Vaux Controller, the computer system may send any of the above newline characters in the output streams, if desired. Since Vaux Protocol Commands do not use CR/LF characters as end-of-line delimiters, the command stream is processed as soon as the final character is received.

In addition to ignoring the newline character(s), the Vaux Controller ignores spaces, tabs, and all non-printing characters except three: **ESCAPE** (1Bh), **Control-C** (03h), and **Control-X** (18h). Any one of these three characters, as well as the % (25h) character, are recognized by the Vaux Controller as a **Command ABORT** — this is handy for manual testing of the Controller with an ASCII terminal. If Informational Messages are enabled (discussed shortly), a “Command Aborted” message will be returned after the ABORT is received.

See the *VauxSerial ASCII Character Set* appendix for the receive ASCII Character Set.

vaux protocol command syntax

overview

Vaux Protocol *Commands* are sent to the Vaux system by the host computer to initiate an action, such as turning on a light. There are two types of Commands: Standard (fixed-length) and Extended (variable-length).

The *Standard (fixed-length) Vaux Protocol Commands* begin with a prefix comprised of the two ASCII characters VP, followed by four or five bytes of data; these bytes are sent as ASCII-encoded hex values (e.g. the byte 1Eh will be sent as the two ASCII chars '1' and 'E', most-significant nibble first). An example Standard Vaux Protocol Command line is:

VP 09 05 45 CF

The *Extended (variable-length) Vaux Protocol Commands* begin with a prefix comprised of the three ASCII characters VPX, followed by a variable number of data bytes, and terminated with a ; (semicolon) character to delimit the end-of-data.

Example Extended Vaux Protocol Command lines are:

VPX FC 16 30 00 06 02 14 92 01 ;

***VPX
05 9F 72 69 A8 CC 52 98
6B A4 59 32 35 ;***

As previously indicated, spaces, tabs, CR, LF, and non-printing characters (except the non-printing ABORT characters) may be embedded anywhere within the command stream; they are simply ignored by the Vaux Controller. The ABORT characters (Escape, Ctrl-C, Ctrl-X, and %) are used to abort a command line and therefore should not be placed in the output stream (they typically have no use in an automated computer interface, except perhaps as an initialization).

A prefix other than **VP** or **VPX**, or invalid ASCII-coded-hex characters (other than '0' to '9', 'a' to 'f', or 'A' to 'F'), will result in an *Bogus Character Message* (if Informational Messages are enabled, as discussed shortly).

VauxSerial interfaces receive serial data characters into a ring buffer. Since most Vaux Protocol Commands are short, the Vaux Controller will generally process the commands without any buffer overflow. Handshaking is recommended to ensure completely reliable data transmission, and is generally necessary for the longer Commands — see the *VauxSerial Port RS-232 Interface* appendix for connections and handshaking details.

Note that the Configuration, SecureID Program, and Request Commands are variants of the Basic or Extended Control Commands (the *Device* byte used to distinguish between these special Commands).

generating the commands

Since the Vaux Protocol format uses simple ASCII lines for command output from the computer, it can easily be generated by simple character output routines. The following examples presume that the computer's serial port has its handshake inputs configured to enable the port (if necessary) — see the *VauxSerial Port RS-232 Interface* appendix for information on serial port connections.

Formatted output functions may be employed as shown in the following clip of C code:

```
#include <stdio.h>      /* I/O defs */
main()
{
    int          device, zone, function, options;
    /* initialize serial port here and direct stdio to it...
       ...or use sprintf to format string, then output to port */
    device       =    9;
    zone         =    5;
    function     =   69;
    options      =  207;
    /* send command to VauxSerial */
    printf("VP %x %x %x %x\r\n", device, zone, function, options);
}
```

This produces the following Vaux Protocol Command line of ASCII characters:

VP 09 05 45 cf

Note, in the above example, that the parameters device, zone, function, and options are used as decimal values in the main program, and converted to the required hex value by the %x format option of the printf command. To allow for readability of the serial stream, spaces are inserted around the device, zone, function, and options numbers, and CR (\r) and LF (\n) characters were added at the end. This makes it easy to hook a terminal to your computer's Serial Port for testing the output stream (of course, you will need to use a null modem adapter to reverse pins 2 & 3 on the RS-232 cable).

If programming for DOS using C (Microsoft C, QuickC, Turbo C...), COM1 or COM2 port control is available using BIOS call `_bios_serialcom` (or `bioscom` under TurboC and TurboC++).

```
#include <stdio.h>      /* I/O defs */
#include <bios.h>       /* _COM_xxx defs */
#include <conio.h>      /* console & port i/o defs */
#define PORT          0      /* COM1 = 0, COM2 = 1 */
#define PORT_BASE     0x3f8  /* COM1 = 0x3f8, COM2 = 0x2f8 */

main (int argc, char *argv[])
{
    unsigned init_data, ret_val;
    /* init port: */
    init_data = _COM_9600 | _COM_NOPARITY | _COM_CHR8 | _COM_STOP1;
    ret_val = _bios_serialcom ( _COM_INIT, PORT, init_data);

    /* send command to VauxSerial port: */
    serial_str_out ("\r\n VP 09 05 45 CF ");
}
```

```

serial_str_out (char *strg)
{
    char  chr;
    while (chr = *strg++)          /* output chars until the null (\0) found */
    {
        /* wait for empty tx buffer: */
        while ((0x2000 & _bios_serialcom (_COM_STATUS, PORT, 0)) != 0x2000);
        outp (PORT_BASE, chr); /* send char */
    }
}

```

To use HyperCard or SuperCard on the Macintosh to send Vaux Protocol Commands, you will need a serial port driver. The HyperCard Serial Port Toolkit, available from APDA™ (the Apple Programmers and Developers Assoc.), provides an interface to the Mac serial port — it provides HyperCard XCMDs/XFCNs (external command/function resources) which allow port control in HyperTalk stacks. The interface between HyperCard and the Toolkit is not particularly fast, however it does provide easy-to-use buffered I/O. The example below shows SuperCard scripts for Window, Card, and Button levels, to send a “Play, Device 2, Zone 1” command in Vaux Protocol.

```

on openWindow
configureSPort modemPort, baud9600, data8, stop10, parityOff
configureSPort modemPort, stripOn, stripControlsOn, CTSOutOn
setSPortBufferSize 10240
end openWindow

on closeWindow
closeSPort -- Important to close driver and buffers on exit!
end closeWindow

on openCard
global currentDevice, currentZone
put "02" into currentDevice
put "01" into currentZone
sendSPort "vp ff 3a 00 3a"—turn off echo and prompting (if on)
end openCard

on mouseUp
global currentDevice, currentZone, func, VpLine
put "0d" into func -- "Play" key
put "vp" && currentDevice && currentZone && func && "00" into VpLine
sendSPort VpLine
end mouseUp

```

standard (fixed-length) control command

Added to Protocol Version: 1.0

Applicable Controllers: All

Purpose: Provides primary control of infrared devices, X-10 light and appliance modules, etc.

The Standard Vaux Protocol Control Command syntax is:

VP dd zz ff oo (cs)

where data bytes are as follows:

dd	Device Code — see Appendix (00h-FFh)
zz	Zone Code — see Appendix (00h-FFh)
ff	Function Code — see Appendix (00h-FFh)
oo	Options (00h-FFh — if not used, set to zero)
(cs)	Optional Checksum (if enabled — see Configuration Command)

Returned: Command Accepted (VI 80) or Invalid Control Cmd (VE 03) Message.
An Event Message may also result.

Example: This example line, explained later, will turn on X-10 module E2 (checksumming is off):

VP 0A 87 14 00

extended (variable-length) command

Added to Protocol Version: 2.0

Applicable Controllers: All except PCremote

Purpose: Used for downloading MACROs, etc.

The Extended Vaux Protocol Control Command syntax is:

VPX dd xx xx ... xx (cs) ; (Note trailing semicolon)

where the data bytes are as follows:

dd	Device Code — see Appendix (00h-FFh)
xx	Data Bytes (00h-FFh) (variable number of bytes — 255 bytes max)
(cs)	Optional Checksum (if enabled — see Configuration Command)
;	<u>End-of-Data delimiter</u> (semicolon character)

Returned: Command Accepted (VI 80) or Invalid Ext. Control Cmd (VE 08) Message.
An Event Message may also result.

Example: This example line programs MACRO 7 with three steps — turn off A2, dim A2 down slightly (1 of these dims = 5 X-10 dims of 32 total), and put the laserdisc (device 6) in Play (no checksumming):

VPX FB 07 03 0A 1C 67 00 0A 12 67 00 06 0D 00 00 ;

vaux protocol message syntax

overview

Vaux Protocol Messages are sent to the host computer by the Vaux system to provide status and other information. The protocol format uses simple ASCII lines for message input to the computer. These messages can easily be interpreted by simple character input routines. Formatted input functions (such as scanf in C) may be employed also, if desired.

There are four types of messages produced: Informational, Error, Event, and Request Messages.

All Vaux Protocol format messages begin with a prefix comprised of two ASCII characters Vx, followed by one or more data values. The “x” value depends on the type of message being sent. To aid in formatted parsing, there are leading and trailing space characters around the “Vx” (i.e. SPACE,“Vx”,SPACE).

The data values are 2-digit ASCII-encoded-hex values (e.g. the two ASCII chars ‘C’ and ‘4’, represent the hexadecimal value C4h). However, to provide more flexibility for computer input routines, we avoid the hex codes Ah to Fh when possible — this allows you to use ASCII-to-Decimal routines, or ASCII-to-Hex, as you desire. These binary-coded-decimal (BCD) codes will have range of 00 - 99 (BCD). Some messages, such as Request, are returning full byte values; if you request these messages you will still need to parse the hex.

One or more spaces will be added around the data bytes (as white space) to aid in formatted input parsing.

Note: If checksumming is enabled, it only applies to Extended Request Messages.

informational messages

Added to Protocol Version: 1.0 up

Applicable Controllers: All; messages vary

Purpose: Informational Messages are Vaux Protocol Messages which provide general information. They are sent from the Vaux Controller to the computer to relate non-critical events (e.g. "Command Accepted"). These messages may be disabled by changing the appropriate configuration flag. Info Messages are prefixed with a VI string, have one data byte, and a trailing CR and LF sequence.

The Vaux Protocol Informational Message syntax is:

VI xx

where the data byte, xx, is an Informational Message code ranging from 00bcd to 99bcd.

<u>"xx"</u>	<u>Meaning</u>
80bcd	Command Accepted
81bcd	Command Aborted (an ABORT character was received — buffer flushed)
82bcd	Bogus Character (an invalid character was received) Note: Prior to Version 2.1, a bogus character would also flush the buffer.
85bcd	Empty MACRO
86bcd	Invalid SecurityID Code
90bcd	IR Learn was successful
91bcd	IR Learn timed-out
98bcd	Controller System Program Load Successful

Info Codes 85bcd and up: Not applicable to PCremote.

Info Codes 95bcd and 96bcd removed in Version 2.2 (IR/X10 Repeats — see Event Mssgs).

Info Code 99bcd removed in Version 3.11 (Vaux Boot — see Event Mssgs).

error messages

Added to Protocol Version: 1.0 up

Applicable Controllers: All; messages vary

Purpose: Error Messages are Vaux Protocol Messages which provide fault information. They are sent from the Vaux Controller to the computer to relate critical events (e.g. “Bad Checksum”). These messages may be disabled by changing the appropriate configuration flag. Error Messages are prefixed with a VE string, have one data byte, and a trailing CR and LF sequence.

The Vaux Protocol Error Message syntax is:

VE xx

where the data byte, xx, is an Error Message code ranging from 00bcd to 99bcd.

<u>“xx”</u>	<u>Meaning</u>
00bcd	Invalid Checksum (when checksumming is enabled)
01bcd	Invalid Configuration Command (its mandatory checksum was incorrect)
02bcd	Invalid Request Command (an invalid request number was found)
03bcd	Invalid Control Command (an invalid device number was found)
04bcd	Invalid Security ID Program Command (its mandatory checksum was bad)
08bcd	Invalid Extended Control Command (invalid device or >255 data bytes)
10bcd	IR Learn or Playback error (invalid device or function, no IR data...)
12bcd	Invalid X10 Command or X-10 interface not found
20bcd	RS-232 buffer overflow — cmd lost/buffer flushed (bad handshake?)
22bcd	Action queue overflow (should not occur — queue was flushed)
30bcd	Invalid MACRO number requested
31bcd	Invalid Device (Source) number requested
32bcd	Invalid Function number requested
33bcd	Invalid Zone number (or Group) requested
41bcd	Backup battery is low (10 years typical life) — service required
60bcd	Vaux internal error code — system will hang (contact factory)
70bcd	Vaux internal error code (contact factory)
90bcd	Controller System Program not loaded — download now
91-95bcd	Program download error (contact factory)

Error Codes 08bcd and up: Not applicable to PCremote.

Error Code 40bcd removed in Version 2.4 (clock/cal).

event messages

Added to Protocol Version: 1.0 up

Applicable Controllers: All except PCremote; messages vary

Purpose: Event Messages are Vaux Protocol Messages which provide information to the computer on a variety of outside events. An event may be a button press on a remote control, for example. Often, these messages correspond to VauxNet activity. For example, a Vaux remote control-initiated temperature setpoint (for a particular zone) will get broadcast on VauxNet for future expansion module use — an Event Message is also sent to the computer to allow the computer to control HVAC directly, using Vaux remote control input. These messages may be disabled by changing the appropriate configuration flag. Event Messages are prefixed with a VV string, have seven data bytes, and a trailing CR and LF sequence.

The Vaux Protocol Event Message syntax is:

VV aa ss dd ff zz oo rr

where the data bytes are as follows:

aa	VauxNet Address Byte (00h-FFh) Currently, is always FFh (broadcast)
ss	Status Code Byte (00h-FFh)
dd	Device Code (source code for A/V routes) — see Appendix (00h-FFh)
ff	Function Code — see Appendix (00h-FFh)
zz	Zone Code — see Appendix (00h-FFh)
oo	Options Byte (00h-FFh) (if not used, is set to zero)
rr	Reserved Byte (00h-FFh) (reserved for future use — is set to zero)

Currently, the only VauxNet Address we are displaying is the TWIMC (To-Whom-It-May-Concern) broadcast packet — this address is FFh.

For the Device, Function, and Zone code displayed, the Status Code is as follows:

<u>“ss”</u>	<u>Meaning</u>
00h	Function Not Performed by Controller
01h	Function Performed
10h	Invalid X-10 Function — Not Performed
11h	X-10 Interface Not Connected
15h	X-10 Input packet received (Altrix 3xxx and Aris 720i only) (dd = 0Ah, ff = Func/Unit-Code, zz = House-Code)
21h	No Infrared data stored for this device/function
22h	No Infrared repeat data stored for this device/function
23h	IR data stored for this device/function is corrupt (should not occur)
24h	Unknown Infrared Function code
25h	Unknown Infrared Device code (00h or > 08h)
26h	Infrared Learn was successful
27h	Infrared Learn timed-out
28h	Infrared Learn internal error 0
29h	Infrared Learn internal error 1
30h	MACRO playback requested (Macro number, in hex, is in options byte)
40h	A/V Route was performed
41h	A/V Route was not performed
86h	Invalid SecID Code (dd/ff=received SecID, zz/oo=expected SecID)
99h	System Boot (dd = Ctrl Model, ff = Protocol Version, zz = Subver, oo = system cksum)
EEh	Debug error message (should not occur)

Status Code 01h: For IR and X-10 Devices, the Options Byte contains the number of RF Remote Control repeats performed (except PCremote) (Version 2.2 Addition).

Status Codes 20h and up: Not applicable to PCremote.

Status Codes 40h and up: Not applicable to (discontinued) MARC II.

All external (remote-control) button-press activity gets reported to the computer by this message. Additionally, when the computer requests an X-10 or Infrared operation, an event message will be returned to indicate what happened.

request messages

Added to Protocol Version: 1.0 up

Applicable Controllers: All; messages vary

Purpose: Request Messages are Vaux Protocol Messages which provide computer-requested reply (upload) information. They are sent from the Vaux Controller to the computer to respond to a variety of information requests. These messages are always enabled. Request Messages are prefixed with a VS, VR, or VX string, have one, two, or more data byte(s), and a trailing CR and LF sequence.

Request Messages are invoked when the computer sends a Vaux Protocol Request Command of the form:

VP FE rr b1 b2 (cc) (this command is described in a later section)

The Short Vaux Protocol Request Message syntax is:

VS xx

where data xx, is a Return Value ranging from 00h to FFh.

The Regular Vaux Protocol Request Message syntax is:

VR xxxx

where data xxxx, is a Return Value ranging from 0000h to FFFFh.

The Extended Vaux Protocol Request Message syntax is:

VX xx xx xx ... xx (cs) ; (Note trailing semicolon)

where the data is as follows:

xx	Data Bytes (00h-FFh) (variable number)
(cs)	Optional Checksum (if enabled — see Configuration Command)
;	End-of-Data delimiter (semicolon character)

Note: Spaces are inserted between data bytes for readability and parsing. CR/LF pair(s) may appear in the data byte stream for longer outputs.

Example: This example shows a single-line reply (checksumming is off):

VX 1B 03 58 48 F7 9A D1 ;

using the system

setting baud rate

Altrix, Aris, and MARC II Controllers provide non-volatile storage of their baud rate setting. The factory setting is 9600 baud, but it may be set to 1200, 2400, 4800, or 9600 baud. The new rate stays in effect until changed, or the factory settings are restored from a reset button or control command.

To Set a new Baud Rate:

(not applicable to PCremote)

VP F0 90 br 00 (cs)

Returned: VI 80
(at new baud rate)

where br is Baud-Rate parameter:

12 = 1200 baud

24 = 2400 baud

48 = 4800 baud

96 = 9600 baud (Factory setting)

and (cs) is the optional checksum.

Example: This example sets baud rate to 2400.

VP F0 90 24 00

setting configuration bytes

Also see: *Restore Factory Settings* System Control Command.

The Altrix, Aris, and MARC II Controllers provide non-volatile storage of their configuration parameters. They are initially configured to the factory settings; new settings will remain in effect until changed, regardless of any power cycling.

The Vaux PCremote accessory, powers-up in a fixed factory setting configuration. The host computer may change configuration settings at any time; the new settings will remain in effect until changed or the unit is powered-up again.

request the current configuration bytes

The host computer may obtain the current settings by issuing a Request Command (see Request Commands/Messages sections).

To Request Configuration Byte A :

VP FE 01 00 00 (cs) Returned: VS xx

where xx is Configuration Byte A, and (cs) is the optional checksum).

To Request Configuration Byte B : (Altrix, Aris only)

VP FE 04 00 00 (cs) Returned: VS xx

where xx is Configuration Byte B, and (cs) is the optional checksum).

The following parameters may be enabled or disabled:

Configuration Byte A is a collection of 1-bit flags:

Added to Protocol Version: 1.0 up Applicable Controllers: All

<u>Bit</u>	<u>Purpose</u>	<u>Factory Setting</u>
ConfigByteA.7 (MSB)	(reserved for future use)	0
ConfigByteA.6	Zone-specific IR outputs *	0 = device-specific
ConfigByteA.5	Enable Event Messages	1 = on
ConfigByteA.4	Enable Error Messages	1 = on
ConfigByteA.3	Enable Informational Messages	1 = on
ConfigByteA.2	Enable Command Line Prompts	0 = no prompts
ConfigByteA.1	Ignore Data Checksumming	1 = no checksums
ConfigByteA.0 (LSB)	Echo Characters	0 = no echo

* Not applicable to PCremote.

Informational Messages relate non-critical events (e.g. command accepted) to the computer, and may be turned off if desired.

Error Messages relate critical events (e.g. Bad Checksum) to the computer, and may also be turned off if desired.

Event Messages relate other system events (e.g. Button press on Remote Control) to the computer, and may also be turned off if desired.

Checksumming, if enabled, will require the computer to send a checksum of the data bytes — this is defined as an 8-bit binary addition of all data bytes, ignoring overflow. This addition sums the data bytes in their original binary form (not the ASCII-encoded string data). When checksumming is enabled, the Vaux Controller will reject a command having an incorrect checksum, and will generate a Bad Checksum Error Message (if Error Messages are enabled). Reception of a command with a proper checksum will result in processing of the command, and generation of an Informational Message indicating Command Accepted (providing that Informational Messages are enabled). Enabling checksumming will also cause the Extended Request Message to append a checksum byte to the end of its data stream.

Character Echoing, if enabled, echos received characters (except the control characters 00h to 1Fh and 7Fh) back to the computer. Note: since tab is not echoed, the space character is a better choice if white space is to be added to the command lines for readability. Echoing is intended for interactive testing with a terminal, and is normally turned off during computer control.

Prompting, if enabled, prompts for a command line with the string, “V> “, prefaced with two CR,LF sequences (i.e. CR,LF,CR,LF, “V> “). Prompting is intended for interactive testing with a terminal, and is normally turned off during computer control.

Device-specific, or Zone-specific Infrared outputs are available on Altrix, Aris, and MARC II Controllers. When configured for Device-specific operation (the factory setting), each of the six infrared emitter ports will be activated only when the corresponding device is selected via the Vaux Remote Control — for example, the second device (CD) will activate only the infrared emitter 2 output port. When configured for Zone-specific operation, each of the six infrared emitter ports will be activated only when the corresponding zone is selected via the Vaux Remote Control — for example, the Master Bedroom zone (5) will activate only the infrared emitter 5 output port. The IR blaster port is the logical combination of all IR ports, in both configurations. Additional information may be found in *Zone-Specific versus Device-Specific Infrared Outputs* appendix.

Configuration Byte B is a collection of 1-bit flags:

Added to Protocol Version: 2.0 up

Applicable Controllers: all except PCremote

Bit	Purpose	Factory Setting
ConfigByteB.7 (MSB)	(reserved for future use)	0
ConfigByteB.6	(reserved for future use)	0
ConfigByteB.5	(reserved for future use)	0
ConfigByteB.4	(reserved for future use)	0
ConfigByteB.3	(reserved for future use)	0
ConfigByteB.2	Enable Auto-Answer Modem	1 = enabled
ConfigByteB.1	Use Software Handshaking	1 = enabled
ConfigByteB.0 (LSB)	Disable User Programming	0 = enabled

Disable User Programming, if enabled, prevents all remote controls from programming macros or initiating infrared learning mode.

Software Handshaking, if enabled, uses XON/XOFF flow control handshaking between the computer and the Vaux system. If this config bit is disabled, the system will use hardware handshaking. See the VauxSerial Port RS-232 Interface appendix.

Auto Answer Modem, if enabled, will initialize a compatible modem (operating at the baud rate of the Vaux Controller) to automatically answer the phone line after one ring. A computer may then dial-in to the modem, and begin a remote Vaux Protocol session to configure or control the Vaux system. Any modem which recognizes the ATSO command (e.g: most Hayes-compatible modems) will allow the auto-answer operation. Specifically, the Vaux system will send ATSO=1 at power-up time instead of the normal VI 99 Vaux Boot message. To use the auto-answer feature, the modem must power-up before, or with, the Vaux system, in order to receive the ATSO=1 command and be placed in auto-answer mode. When in auto-answer state, the VauxSerial interface ignores all characters until the first ‘V’ is received; this prevents unwanted error messages when the modem returns strings such as RING and CONNECT. Disabling Info Messages (which disables VI99) will not disable modem.

configuration command

Purpose: Initializes or changes the some of the system's programmable features.

The Vaux Protocol Configuration Command syntax is:

VP FF aa bb cs

where data bytes are as follows:

aa	Configuration Byte A (00h-FFh)
bb	Configuration Byte B (00h-FFh)
cs	Mandatory Checksum of aa and bb (required for configuration integrity)

The checksum cs is always mandatory for a configuration — it is defined as the 8-bit unsigned integer addition of the two data bytes, discarding any overflow.

Returned: Command Accepted (VI 80) or Invalid Configuration Cmd (VE 01) Message.

Example: This example configures the system to the factory setting shown above.

VP FF 3A 06 40

Example: This example disables info/error/event messages and disables auto-answer modem initialization. This is a typical setting for computer control where “silent” operation is desired to eliminate or minimize serial receive parsing by the computer system (ie: received messages are only obtained using polling, if at all):

VP FF 02 02 04

Example: This example configures the Vaux Controller to use zone-specific infrared outputs. This is a typical setting for a zoned Altrix system:

VP FF 7A 06 80

Example: This example configures the Vaux Controller to lock-out system programming by remote controls (disables Macro and IR learning):

VP FF 3A 07 41

Example: This example configures the Vaux Controller to echo characters, disable checksumming, enable prompts, enable all messages, use zone-specific infrared outputs, use software handshaking, and disable the modem. This is a typical setting for interactive ASCII terminal use:

VP FF 7F 02 81

security ID (SecID) code

(for RF remote control ID)

To Request Current SecureID Code:

VP FE 03 00 00 (cs) Returned: VR xxxx

where xxxx is the current Security ID Code, and (cs) is the optional checksum.

Security ID Code Program Command

Purpose: This command is used to set the current Security ID Code. The new ID Code will be stored in the non-volatile parameter memory of the Vaux controller. This provides the same functionality as changing your Security ID Code with a Vaux RF Remote Control; note that all Remote Controls in use will need to be manually programmed to the new code. While Vaux Remote Controls can only set the Security ID Code to a 4-digit BCD value, this Vaux Protocol command can set the system to the full hex range of codes. There may be occasions (e.g. a party) where it is desirable to temporarily disable the Remote Controls by setting a non-BCD ID Code (then, only the computer can restore the ID Code to a BCD value).

For Altrix, Aris, and MARC II Controllers, the new ID Code will be stored in the non-volatile parameter memory of the Controller; note that all Remote Controls in use will need to be manually programmed to the new ID Code.

For the Vaux PCremote Accessory, the new ID Code will be stored in the (volatile) Security ID Code memory of the PCremote interface. The new Security ID Code is now the default for subsequent transmissions (the power-up value is 1070h — the Vaux standard factory setting). Also, the Change Security ID Code RF Command is broadcast to all Vaux Altrix, Aris, and MARC II Controllers in the house; they will be updated and will recognize the new ID Code as indicated above. Note that all Remote Controls in use will need to be manually programmed to the new code. As is the case with Vaux Remote Controls, the original ID Code of the PCremote must match the Controller's ID Code, or the Controller will not accept the (RF broadcast) Change Security ID Code Command.

The Vaux Protocol Security ID Program Command syntax is:

VP FD aa bb cs

where data bytes are as follows:

aa	Security ID Code high byte (normally 00bcd-99bcd; 00h-FFh is legal)
bb	Security ID Code low byte (normally 00bcd-99bcd; 00h-FFh is legal)
cs	Mandatory Checksum of aa and bb (required for Security ID integrity)

Returned: Command Accepted (VI 80) or Invalid Sec ID Prog Cmd (VE 04) Message.

Example: This example line will set the new Security ID Code to 8497bcd:

VP FD 8497 1B

request command

Added to Protocol Version: 1.0 up

Applicable Controllers: All; messages vary

Purpose: Request Messages (VS xx, VR xxxx, or VX xx xx ... xx ;) are returned by Vaux Controller upon receipt of a Request Command Line. The various Vaux Controllers support these requests as appropriate for the hardware: for example, the Vaux PCremote accessory cannot provide upload or download of MACRO data, and the audio/video routing is only available in Aris systems. An unsupported request will return an Invalid Request (VE 02) Message.

The Vaux Protocol Request Command syntax is:

VP FE rr b1 b2 (cs)

where data bytes are as follows:

rr Request Code (00h-FFh)
b1 Options Byte 1 (00h-FFh) (if not used, set to zero)
b2 Options Byte 2 (00h-FFh) (if not used, set to zero)
(cs) Optional Checksum (if enabled — see Configuration Command)

NOTE: Some of the following commands are described in greater detail in other sections of this manual.

<u>Request Code</u>	<u>Purpose</u>	<u>Resultant Message</u>
00h	Get VauxProtocol/Software Version	VS aa
09h	Get VauxProtocol/Software Sub-Version e.g. aa=27, bb=05, is Version: 02.75	VS bb
01h	Get Configuration Byte A	VS xx
04h	Get Configuration Byte B	VS xx
02h	Get Vaux Controller Model (xx: 30 = Altrix 3000, 34 = Altrix 3400, 66 = MarcPCI, 67 = PCremote, 68 = MARC II, 70 = Aris 700, 71 = Aris 710, 72 = Aris 720i)	VS xx
03h	Get Current Security ID Code (FS = 1070)	VR xxxx
41h	Get MACRO Definition b1 - Macro (01h-14h; 1 to 20 resp.)	VX xx xx ... xx (cs) ;
48h	Get IR Dev/Func Memory b1 - Device (01h-08h) b2 - Function (00h-1Dh)	VX xx xx ... xx (cs) ;
30h *	Get Altrix Zone Status b1 - Zone (01h-08h)	VX state source zone (cs) ;
50h **	Get VBT Ramp Rate for Remote Controls	VS xx

51h **	Get Aris Zone Turn-On Default Table b1 - Zone (01h-06h)	VX xx xx ... xx (cs) ;
52h **	Get Aris Zone Status b1 - Zone (01h-06h)	VX xx xx ... xx (cs) ;
53h ***	Get Zone Group (FS = 00h = Group 0)	VS xx
54h ***	Get Source-Change parameter (FS = 0Fh)	VS xx
63h ***	Get Power-Up timer (FS = 00h = 0 sec)	VS xx
64h ***	Get Power-Down timer (FS = 05h ~ 5 secs)	VS xx
66h ***	Get A/V Source Definition (FS = BCh)	VS xx
67h ***	Get X-10 Sys-Active-Pwr HC (FS = 00h = disable)	VS xx
68h ***	Get X-10 Source-Pwr HC (FS = 00h = disable)	VS xx
69h ***	Get X-10 Zone-Pwr HC (FS = 00h = disable)	VS xx
70h ****	Get X-10 Input Map Table b1 (varies)	VX xx xx ... xx (cs) ;

* Altrix 3xxx only.

** Aris 7xx only.

*** Altrix 3xxx and Aris 7xx only.

**** Altrix 3xxx and Aris 720i only (2-way X-10 support).

Request Codes 00h to 03h: These are the only cmds valid for PCremote.

Request Code 47h removed in Version 2.3.

Request Codes 40h and 45h removed in Version 2.4.

Returned: A Request Message (VS, VR, or VX) will result from a valid request.
Or, an Invalid Request (VE 02), or Invalid MACRO Number (VE 30)
Message may result from an unrecognized request.

system control command

Added to Protocol Version: 2.0

Applicable Controllers: All except PCremote

Purpose: This command performs miscellaneous control tasks.

The Vaux Protocol System Control Command syntax is:

VP F0 cc b1 b2 (cs)

where data bytes are as follows:

cc Control Code (00h-FFh)
b1 Options byte 1 (00h-FFh — if not used, set to zero)
b2 Options byte 2 (00h-FFh — if not used, set to zero)
(cs) Optional Checksum (if enabled — see Configuration Command)

NOTE: Some of the following commands are described in greater detail in other sections of this manual.

<u>Control Code</u>	<u>Purpose</u>	<u>Returns</u>
11h	Clear Infrared Memory	VI 80
12h	Clear MACRO Memory	VI 80
30h	Play a MACRO b1 MACRO Number (01h - 14h) b2 unused (set to 00h)	VI 80, VI 85 or VE 30
50h	Enter/Exit Learn Mode	VI 80
51h	Enter Learn Mode and Learn a Code b1 Device (01h - 08h) b2 Function (00h - 1Dh)	VI 90 or VI 91
59h **	Set Zone Turn-On-Volume-Limit b1 Zone (01h to 06h, 11h to 16h...91h to 96h)	VI 80
60h **	Set VBT Ramp Rate for Remotes (FS = 80h = 128ms) b1 Delay parameter (01h - FFh)	VI 80
61h ***	Set Zone Group (FS = 00h) b1 Group base (00h, 10h, 20h...)	VI 80
62h ***	Set Source-Select mode (bit-addressable) (FS = 0Fh) b1 Mode	VI 80
63h ***	Set Power-Up Delay (FS = 00h = 0 sec) b1 Delay (seconds) after device pwr-up (00h to FFh)	VI 80
64h ***	Set Power-Down Delay (FS = 05h ~ 5 secs) b1 Delay (approx) before device pwr-off (01h to FFh)	VI 80

66h ***	Set A/V Source Definition (bit-addressable) (1=A/V, 0=Audio-only; FS = BCh)	VI 80
	b1 A/V-Source-Definition byte	
67h ***	Set X-10 Sys-Active-Pwr HC (FS = 00h = disable) (00=disable, 01=A, 02=B...10h=P)	VI 80
	b1 House Code for System-Active	
68h ***	Set X-10 Source-Pwr HC (FS = 00h = disable) (00=disable, 01=A, 02=B...10h=P)	VI 80
	b1 House Code for Source-On	
69h ***	Set X-10 Zone-Pwr HC (FS = 00h = disable) (00=disable, 01=A, 02=B...10h=P)	VI 80
	b1 House Code for Zone-On	
90h	Set new Serial Baud Rate	VI 80 (at new baud)
	b1 12=1200, 24=2400, 48=4800, 96=9600	
99h	Restore Factory Settings (reset Config bytes, SecID to 1070, clears IR and MACRO memory, enables software handshaking, enables auto-answer modem, resets A/V defaults, disables AV routes, resets baud rate to 9600, resets Zone Group to 0, restores any other factory settings, and flashes the front-panel light bar). This is the same as pressing the RESET button.	VI 80

* Altrix 3xxx only.

** Aris 7xx only.

*** Altrix 3xxx and Aris 7xx only.

**** Altrix 3xxx and Aris 720i only (2-way X-10 support).

Control Code 10h removed in Version 2.4.

using a zone group

(multiple-Aris-controller installations)

Added to Protocol Version: 2.6 Applicable Controllers: Aris 7xx

The host computer may change an Aris Controller's default Zone Group. By programming different zone groups, multiple Controllers may be configured together in a single installation, providing additional zones. The Zone Group factory setting for a controller is Group 0, but it may be changed to Group 1, Group 2...up to Group 9 — a controller with a Zone group of 0 is called the Master controller.

Vaux RC-8 remote controls are programmed with a Default Zone: this is a two-digit number reflecting the zone group (the actual Controller), and that controller's zone (1 to 6). The first digit of the RC-8 Default zone is the zone group (0 to 9), and the second digit is the actual zone (1 to 6, but whole-house zone 0 is also permitted); a total of ten controllers may be configured this way, for a total of 60 zones. For example, to access the six zones in zone group (controller) 0, the RC-8 remotes are programmed with default zones of 00, 01, 02...06. Similarly, to access the six zones in zone group (controller) 1, the RC-8 remotes are programmed with default zones of 10, 11, 12...16.

The Zone Group stored in an RC-8 remote control must match the controller's zone group in order for the controller to recognize routes, volume/bass/treble, and mute functions for the physical zones. However, the master controller (Zone Group 0) recognizes all remotes (any zone group) for playback of infrared codes, requiring IR storage in only one controller. The Zone Group factory setting for a controller is Group 0 (00h), but it may be changed to Group 1 (10h), Group 2 (20h), etc. Note that the zone group is actually the upper nibble of the stored zone group byte.

To Change the Zone Group:

VP F0 61 zg 00 (cs)

where zg is Zone Group base (00h, 10h...90h):

00h => Zones 01 to 06 (factory setting)

10h => Zones 11 to 16

(etc.)

and (cs) is the optional checksum.

To Get Current Zone Group:

VP FE 53 00 00 (cs)

(cs) is the optional checksum.

Returned: VS zg

Example: This example sets the controller to Zone-Group 1 (recognizing zones 11, 12...16):

VP F0 61 10 00

the source-selection process

Pressing a Source-Device button on a Vaux remote control or keypad (e.g.: CD) not only changes the “current device” to CD, but also transmits a “Source-Select” command, informing the Controller that the CD button was pressed. There are a variety of features which can be enabled as part of this Source-Selection process, using a combination of learned IR codes, MACRO sequences, and computer-port programming.

Auto-Source-Power Option

The Auto-Source-Power feature is an integral, and optional, step in the Source-Selection process. Auto-Source-Powering automatically tracks the state of infrared-controlled source devices (CD, Laser...) and powers them up or down as necessary. In a multi-zone system, the system keeps track of all zones and only powers a device up if it is not currently on. Likewise, it only powers a device down when it is not in use by any other zone. You are not required to use this feature, but you may enable auto-source-powering of one or more of the source devices by simply programming a couple of IR power codes.

In operation, Aris and Altrix Controllers use the ON and OFF buttons of the remote control in special ways: For an Aris System, ON and OFF perform an A/V route (of the remote’s Current-Device, to the remote’s Default-Zone). For an Altrix System, ON and OFF presses invoke MACROs 19 and 20 (resp.) However, the IR ON/OFF locations are programmable in infrared learn mode, and are used to store these optional device power commands, for Auto-Source-Powering of one or more source-devices.

There are actually sixteen (IR) storage locations associated with the ON and OFF buttons, two for each of the eight Devices (Tuner, CD...) — these locations are only used for the Auto-Source-Power feature, although they can be tested by pressing Shift-ON or Shift-OFF (which also track states). To enable Auto-Source-Powering for a particular Device, you simply program the appropriate ON and OFF locations with the appropriate on/off IR codes for the device. Generally, only one IR power code is used for an A/V device, so simply program it into both the ON and OFF locations (both locations are used by the Aris or Altrix system to keep the system synchronized, as well as to handle the occasional device which has two individual On/Off codes).

The source-devices will then automatically power-up as needed when sources are selected. When you select a different source, or switch the system off, the device will turn off after about five seconds. In a multi-zone system, the source turns off only if no other zone is using the source device.

This five-second “Power-Off-Delay” is actually programmable via the computer port. Also, there is a computer-programmable “Power-On-Delay,” which is normally zero seconds — it may be programmed to add a fixed delay after each IR and X-10 device turns on, to allow the device to stabilize before receiving configuration commands, etc.

Note: the power state is managed intelligently by the controller, but the user may manually turn a device on or off and get the system out-of-sync. It is obvious when a device is in the wrong state; it is easily rectified by manually correcting the power state of the device in question.

Auto-Receiver-Power Option

A refinement of the above concept provides an additional feature: Auto-Receiver-Power. This allows one of the devices (Device 1: Tuner) to power an A/V receiver for one of the zones, and intelligently turn the receiver on when the system is in use. Enabling the Receiver powering simply requires programming a couple of IR power codes for Device 1. This option provides global or zoned powering of A/V receivers (or other IR devices). If desired, this feature may be disabled via the computer port, allowing normal Tuner source device powering.

To use this feature, the device's IR power commands must be learned into the Device-1 (Tuner) bank ON and OFF locations, the system configured for device- or zone-specific operation, and the IR emitter for the device(s) connected as follows: In device-specific IR mode, a global "system-active" IR device is connected to the Tuner port (1) or Blaster port. This single IR device is powered up when system is active, and off when inactive -- e.g. a single-zone Altrix system's A/V receiver, or an Aris system's multi-source input device (e.g. CD/tape unit). In zone-specific IR mode, zoned IR devices (one or more identical devices) are connected to Zone ports 1, 2, 3... (not Blaster port) -- e.g. a multi-zone Altrix system using identical receivers in different zones. Another option, while in zone-specific IR mode, is to program the Tuner device to force global operation—e.g. an Aris system in zoned-IR mode (perhaps for multi-zone-TV powering), but needing global IR powering for a multi-source input device (e.g. CD/tape unit). In this case, the "Tuner" device powers up when system is active, and off when the system goes inactive—the IR device must be connected to the Blaster port in this case, since the other IR emitter ports are zoned.

Auto-TV-Power Option

Another refinement of the above concept provides an additional feature: Auto-TV-Power. This allows one of the devices (Device 8: TV/Tape-2) to power a television monitor for one of the zones, and intelligently turn the TV on only for audio/video sources, and off for audio-only sources. Enabling the TV powering simply requires programming a couple of IR power codes for Device 8. This option provides zoned powering of TVs, based on sources defined as Audio-only or A/V (this definition is programmable, via the computer port). This feature may be disabled via the VauxSerial port, allowing normal Tape-2 source powering.

To use this feature, the TV's IR power commands must be learned into the Device-8 (TV/Tape-2) bank ON and OFF locations, the system must be configured for zone-specific operation, and the IR emitter for the TV(s) connected as follows: In zone-specific IR mode, zoned IR TVs (one or more identical TVs) are connected to Zone ports 1, 2, 3... (not Blaster port) -- e.g. a multi-zone Aris or Altrix system using one or more identical TVs in different zones. In device-specific IR mode, TV powering is disabled (there is no global source-specific TV powering possible, since different zones may have A/V or Audio-only sources activated) -- in this case, the system provides normal Tape-2 source powering.

The factory setting source definition is:

Tuner:	Audio-only (TV off)
CD:	Audio-only (TV off)
Cab/Sat:	A/V (TV on)
VCR-1:	A/V (TV on)
VCR-2:	A/V (TV on)
Laser:	A/V (TV on)
Tape-1:	Audio-only (TV off)
TV/Tape-2:	A/V (TV on)

Auto-PLC-Source-Power Option

For non-IR-controlled source devices, you may also opt to enable Auto-PLC-Source-Powering, which sends X-10 commands for turning on and off appliance or relay modules — unique for each source. Like the IR powering discussed above, the X-10 modules will turn on or off only as necessary — the power state of the source device is tracked by the controller, across all zones. This feature is programmed via the computer port.

This option, if enabled, sends source-specific X10 ON/OFF codes. The House-Code used for this function may be programmed, via the VauxSerial port, to a number (1 to 16) corresponding to the desired House-Code, as follows: 1=A, 2=B, 3=C, ... 16=P, 0=disabled. The factory-setting is 0, which disables this feature. The Unit-Code corresponds to the Source that is turning on or off. For example, if the system is programmed to used House-Code K for the X-10 Source Power option, then a K6-ON X-10 command is sent when Source 6 (Laser) turns on, and K6-OFF is sent when the source turns off. Note: if you will be enabling the "Auto-Receiver-Powering" option, you should not use an X-10 device on source-1 (tuner); if you will be enabling the "Auto-TV-Powering" option, you should not use an X-10 device on source-8 (tv/tape-2).

Auto-PLC-Zone-Power Option

For non-IR-controlled zone devices, such as audio power amplifiers used with an Aris System, you may also opt to enable Auto-PLC-Zone-Powering, which sends X-10 commands for turning on and off appliance or relay modules — unique for each zone. The X-10 modules will turn on or off only as necessary, as the zone turns on or off. Lamp modules may also be used to turn zone-specific effect lighting on and off. This feature is programmed via the computer port.

This option, if enabled, sends zone-specific X10 ON/OFF codes. The House-Code used for this function may be programmed, via the VauxSerial port, to a number (1 to 16) corresponding to the desired House-Code, as follows: 1=A, 2=B, 3=C, ... 16=P, 0=disabled. The factory-setting is 0, which disables this feature. The Unit-Code corresponds to the Zone that is turning on or off. For example, if the system is programmed to use House-Code M for the X-10 Zone Power option, then an M3-ON X-10 command is sent when Zone 3 turns on, and M3-OFF is sent when the zone turns off.

System-Active-PLC-Power Option

If you wish to have a global “any-zone-on” indication, you may enable System-Active-PLC-Powering, which sends a unique X-10 command for turning on and off appliance or relay modules. This is a useful method for powering a multi-zone audio power amplifier. Lamp modules may also be used to turn effect lighting on and off. This feature is programmed via the computer port.

This option, if enabled, controls a specific X10 device, and provides a master system-is-active control function. The House-Code used for this function may be programmed, via the VauxSerial port, to a number (1 to 16) corresponding to the desired House-Code, as follows: 1=A, 2=B, 3=C, ... 16=P, 0=disabled. The factory-setting is 0, which disables this feature. The Unit-Code is forced to 16 for this feature. For example, if the system is programmed to use House-Code L for the X-10 System-Active Power option, then an L16-ON X-10 command is sent when the system first becomes active (a zone turns on from an off state, and all other zones are currently off), and L16-OFF is sent when the system becomes inactive (a zone turns off, and all other zones are currently off).

Source-Select MACROS

Ten MACROS, 11 to 20, are used by the Altrix and Aris Controllers for customizing source-selection operations. The first eight of these ten MACROS are associated with the Source-Change commands (they begin playback after all source/zone power-up, and routing for the Aris, is complete), the last two MACROS are available for general use:

MACRO 11:	Tuner
MACRO 12:	CD
MACRO 13:	Cab/Sat
MACRO 14:	VCR 1
MACRO 15:	VCR 2
MACRO 16:	Laser
MACRO 17:	Tape 1
MACRO 18:	Tape 2
MACRO 19:	ON button (Altrix general-purpose use)
MACRO 20:	OFF button (plays after zone turn off)

With an Aris system, MACROS 11 to 18 play back after the zone route is performed, and may be used to fine-tune the effect desired when a source is selected. For example, pressing Laser will route the laser to your zone, but MACRO 16 may also be used to drop a projection screen, dim lights, and put the laserdisc in Play. For an Altrix system, MACROS 11 to 18 will at least be used to play the appropriate IR source-selection command for the A/V receiver (an Altrix “route”).

setting source-select mode

Added to Protocol Version: 2.6 Applicable Controllers: Altrix 3xxx and Aris 7xx

To Get Current Source-Select Mode:

VP FE 54 00 00 (cs)

(cs) is the optional checksum.

Returned: VS ss

To Set the Source-Select Mode:

VP F0 62 ss 00 (cs)

where ss is Source-Selection Mode byte (bit-addressable) (FS = 0Fh):

ss.7	(spare) (FS=0)
ss.6	1 = disable Auto-TV-Powering (FS=0)
ss.5	1 = ARP global, 0 = ARP zoned if IR is zone-specific (FS=0)
ss.4	1 = disable Auto-Receiver-Powering (ARP) (FS=0)
ss.3	1 = enable OFF MACRO (Macro 20) (FS=1)
ss.2	1 = enable ON MACRO (Macro 19) (Altrix-only) (FS=1)
ss.1	1 = enable SrcSel MACROs (Macros 11-18) (FS=1)
ss.0	1 = Source-Selection master enable (FS=1)

and (cs) is the optional checksum.

Example: This example sets the Source-Select features to the Factory Setting — all SrcSel options are enabled; MACROs 11 to 18 are used to customize each selected source; MACRO 20 is used for a general-purpose OFF-button sequence, occurring at the end of zone turn-off; for an Altrix system, MACRO 19 is used for a general-purpose sequence for the ON button (for an Aris system, the ON button is used for Current-Device Source-Selection); Auto-Receiver-Powering is enabled and is global when Device-specific IR output is used (e.g. single-zone Altrix system), or is zoned when Zone-specific IR output is used (e.g. multi-zone Altrix system using a bank of receivers); Auto-TV-Powering is enabled, which is zoned when Zone-specific IR output is used:

VP F0 62 0F 00

Example: This example is similar to the one above, but forces Auto-Receiver-Powering to always be global — this allows Zone-specific IR output to be used (e.g. for zoned TV powering), while providing a global “system-active” IR device to be powered on when any zone is active, and off when all zones go inactive (e.g. IR-controlled multi-zone amplifier, a multi-source device, or any global IR device):

VP F0 62 2F 00

Example: This example disables an Altrix system’s ON macro:

VP F0 62 0B 00

setting A/V sources for auto-TV-powering

Added to Protocol Version: 3.00 Applicable Controllers: Altrix 3xxx and Aris 7xx

A Source-Device is defined as Audio/Video or Audio-only — the TV will turn on and off as necessary when source selections are made, provided Auto-TV-powering is enabled (see Source-Select mode), and the TV's power codes have been programmed in the TV/Tape-2 ON and OFF locations. If Zone-specific IR output is enabled, multiple identical TVs may be auto-powered.

To Get Current A/V-Source-Definition:

VP FE 66 00 00 (cs)

(cs) is the optional checksum.

Returned: VS sd

To Set the A/V-Source-Definition:

VP F0 66 sd 00 (cs)

where sd is A/V-Source-Definition byte (bit-addressable):

(1=A/V, 0=Audio-only; FS = 10111100b = BCh)

sd.7	Device 8 (Tape-2/TV)
sd.6	Device 7 (Tape-1)
sd.5	Device 6 (Laser)
sd.4	Device 5 (VCR-2)
sd.3	Device 4 (VCR-1)
sd.2	Device 3 (Cab/Sat)
sd.1	Device 2 (CD)
sd.0	Device 1 (Tuner)

and (cs) is the optional checksum.

Example: This example sets the VCR-2 device to audio-only (no TV powering):

VP F0 66 AC 00

setting auto-PLC-powering options

Added to Protocol Version: 2.81/3.10 Applicable Controllers: Altrix 3xxx and Aris 7xx

X-10 appliance module may be used to switch amplifiers, etc. The System-Active control provides global switching when any zone is on, and is useful for multi-channel amplifiers. The zoned commands may be handy for stereo zone amps, and the source commands may be used to power a source which has no infrared control.

To Get Current X-10 System-Active-Power House-Code (Unit-Code is always 16):

VP FE 67 00 00 (cs)

(cs) is the optional checksum.

Returned: VS hc

To Set the X-10 System-Active-Power House-Code (Unit-Code is always 16):

VP F0 67 hc 00 (cs)

where hc is desired House-Code (FS = 00h = disable) (00=disable, 01=A, 02=B...10h=P)

and (cs) is the optional checksum.

Example: This example sets the System-Active HC to F (F16 will be used):

VP F0 67 06 00

And, likewise:

To Get Current X-10 Source-Power House-Code (Unit-Code corresponds to source, e.g. CD = 2):

VP FE 68 00 00 (cs)

To Set the X-10 Source-Power House-Code (Unit-Code corresponds to source, e.g. CD = 2):

VP F0 68 hc 00 (cs)

To Get Current X-10 Zone-Power House-Code (Unit-Code corresponds to zone number):

VP FE 69 00 00 (cs)

To Set the X-10 Zone-Power House-Code (Unit-Code corresponds to zone number):

VP F0 69 hc 00 (cs)

infrared control (learned IR output)

To Control an Infrared Device:

VP dd zz ff nn (cs)

where data bytes are as follows:

dd Device Code — see Appendix (01h-08h) (01h-06h for pre-2.3 revs)
zz Zone Code (if used — see Configuration Command) (01h-06h)
ff Function (key) Code — see Appendix (00h-1Dh) (00h-27h for pre-2.3 revs)
nn Number of repeats (00h-FFh) (added in ver 2.80)
(repeats are useful to ramp volume, etc.)
(cs) Optional Checksum (if enabled — see Configuration Command)

Returned: Command Accepted (VI 80) or Invalid Control Cmd (VE 03) Message.
An Event Message may also result.

Example: This example line will send Play (Function 0D), for CD (Device 2), and specifies Zone 5 (checksumming is off). If this controller is configured for Device-specific infrared output, IR emitter port 2 will be used, or, if it is configured for Zone-specific Infrared output, IR emitter port 5 will be used. Note that Device 2 is actually specifying the IR memory bank (page) where the code is stored, and the Zone information is only required when the controller is programmed for Zone-specific Infrared output instead of the factory-setting of Device-specific (see Configuration). No repeats are sent.

VP 02 05 0D 00

To Learn an Infrared Device Code: (All except PCremote)

VP F0 51 dd ff (cs)

where data bytes are as follows:

dd Device Code — see Appendix (01h-08h) (01h-06h for pre-2.3 revs)
ff Function (key) Code — see Appendix (00h-1Dh) (00h-27h for pre-2.3 revs)
(cs) Optional Checksum (if enabled — see Configuration Command)

Returned: Command Accepted (VI 80) or IR Error (VE 10) Message.

To Exit (or Enter) Infrared Learn Mode: (All except PCremote)

VP F0 50 00 00 (cs)

Example: This example line will enter learn mode for Function Skip Fwd (16), on Device Laser (06), will continue learning for Function Pause (0E), on Device Laser (06), and then exit learn mode(checksumming is off):

**vp F0 51 06 16
vp F0 51 06 0E
vp F0 50 00 00**

To Clear Infrared Storage Memory: (All except PCremote)

VP F0 11 00 00 (cs)

To Upload Infrared Data from IR Device/Function Memory: (All except PCremote)

VP FE 48 dd ff (cs)

Returned: VX h1 h2 h3 h4
xx xx ... xx (cs) ;
or VI 80, VE 31, or VE 32.

where dd is Device Code (01h - 08h), ff is the Function (button) Code (00h - 1Dh), h1-h4 is a 4-byte header, xx, xx... is the stored Infrared Data (128 bytes), and (cs) are the optional checksums.

To Download Infrared Data to IR Device/Function Memory: (All except PCremote)

VPX F9 dd ff xx xx xx xx xx (cs) ; (Note trailing semicolon)

where xx, xx... is the new Infrared Data (128 bytes), dd is Device Code (01h - 08h), ff is the Function (button) Code (00h - 1Dh), and (cs) is an optional checksum.

Returned: VI 80 or VE 03.

lighting and appliance (X-10®) control

controlling an X-10 device

(All controllers)

VP 0A zz ff 00 (cs)

where data bytes are as follows:

zz Zone (House/Unit) Code — see Appendix
ff Function Code — see Appendix (14h, 1Ch, 12h, or 1Ah)
(cs) Optional Checksum (if enabled — see Configuration Command)

Returned: Command Accepted (VI 80) or Invalid Control Cmd (VE 03) Message.
 An Event Message may also result.

Example: This example line will turn on X-10 module E2 (checksumming is off):

VP 0A 87 14 00

adjusting X-10 dim levels

The Dim and Bright commands, when used with the above command, will result in 5 actual X-10 dim/brt steps. This granularity offsets the latency of RF packet transmission with interactive control. There are 32 steps of dim/brt levels available with an X-10 lamp module. For complete control over dim levels (single steps) see the next section on Low-Level X-10 control.

NOTE: a module which is OFF, will assume full brightness upon receipt of a DIM command, and will proceed to dim down from this point. The opposite is not true — a BRIGHT command will not bring an OFF module up gradually (as might be desired).

NOTE: a dimmed module does not assume full brightness upon receipt of an ON command (it stays at the same level — it is already “on”). The OFF command is the only command which puts the module in a definite state.

To keep module states predictable, you can:

- 1) Pre-DIM modules to zero brightness by first turning them OFF, and then dimming them down with 32 or more DIM commands for a standby state. Then, brighten from the standby state when desired, and, later, dim down when desired.
- 2) Control multiple modules simultaneously if they all share the same House Code. By using the Vaux Protocol command for Low-Level X-10 control, you can exercise a greater degree of control. See the next section.

You may Control multiple modules simultaneously if they all share the same House Code. The Vaux Protocol command for Low-Level X-10 control lets you select one or more Unit Codes of modules on a specific House Code. This is accomplished by individually “arming” the modules first, and then sending a global command to all of the armed modules.

controlling an X-10 device using the low-level commands

(All except PCremote)

VP 10 hh fu nn (cs)

where data bytes are as follows:

hh	House Code — see Appendix (00h-0Fh)
fu	Function/Unit Code — see Appendix
nn	Number of Repeats (00h - FFh)
(cs)	Optional Checksum (if enabled — see Configuration Command)

Returned: Command Accepted (VI 80) or Invalid Control Cmd (VE 03) Message.
An Event Message may also result. Note: bogus functions are not trapped.

This Command acts differently based on the number of Repeats requested:

- 1) Zero repeats are requested: HH/FU packets are sent twice with required gaps. This version of the command is used to arm modules (by specifying House and Unit codes), or to Command the previously-armed set of modules ON or OFF (by specifying House and Function codes).
- 2) 'N' repeats are requested: HH/FU packets are sent 'N' times with no gaps. This version of the command is used to dim or brighten modules (by specifying House and Function codes). The modules will be fully dimmed or brightened after 32 steps, but a few steps may get missed due to powerline noise — send a few extra if you are dimming completely off. This command allows 1 to 255 steps.

Example: This example will control X-10 modules F2, F3, and F10:

```
VP 10 09 07 00    arm F2 (09h/07h)  
VP 10 09 04 00    arm F3 (09h/04h)  
VP 10 09 0F 00    arm F10 (09h/0Fh)  
VP 10 09 12 20    dim them all 32 steps (to 0%)  
VP 10 09 1A 10    brt them all 16 steps (~50%)
```

X-10 input (2-way X-10)

(Altrix 3xxx and Aris 720i only)

Vaux Aris 720i and Altrix 3000 and 3400 Controllers support two-way X-10 powerline control (as opposed to the one-way X-10 control found on the Aris 700 and MARC+ systems). In addition to X-10 output for controlling powerline devices, the Controllers also “listen” to the powerline for X-10 commands generated by other systems. The systems are configured with factory defaults, but can be reprogrammed as desired.

A two-way Vaux Controller is listening for one specific X-10 House-Code to which it will respond. All other House-Codes are ignored by the Controller, although RS-232 messages are sent out the VauxSerial port — for an external control program running on an attached computer system, the Controller provides a full two-way X-10 interface for sending X-10 commands and monitoring powerline traffic.

The factory setting X10-Input-House-Code is “G” and may be changed to be any other House-Code from A to P, or disabled. For the desired House-Code, there are 36 individual commands which get mapped to an action in the Vaux system: Unit-Code-1-On, 1-Off, 2-On, 2-Off...16-On, 16-Off, All-Lights-On, All-Units-Off, Dim, and Bright codes, each have a map-table entry. The factory settings provide a minimal map table which provides out-of-the-box capability by programming MACROs 1 to 10, as well as demonstrating A/V switching and volume control, using X-10 codes on House-Code G. Note that volume control using X-10 Bright/Dim commands is not particularly fast, but press-and-hold adjustment is supported.

The X-10 Input factory settings are:

Active-House-Code:	G
Unit-1-On (G1-On):	MACRO-1
Unit-2-On (G2-On):	MACRO-2
Unit-3-On (G3-On):	MACRO-3
Unit-4-On (G4-On):	MACRO-4
Unit-5-On (G5-On):	MACRO-5
Unit-1-Off (G1-Off):	MACRO-6
Unit-2-Off (G2-Off):	MACRO-7
Unit-3-Off (G3-Off):	MACRO-8
Unit-4-Off (G4-Off):	MACRO-9
Unit-5-Off (G5-Off):	MACRO-10
All-Lights-On (G):	Source-Select: Tuner to Zone-1
All-Units-Off (G):	Zone-1 Off
Bright (G):	Volume-Up on Zone-1 (Tuner vol for Altrix)
Dim (G):	Volume-Down on Zone-1 (Tuner vol for Altrix)

The factory setting X10-Input-House-Code is “G” and may be changed to be any other House-Code from A to P, or disabled.

To Get Current X-10-Input-House-Code (Controller will respond only to this HC):

VP FE 70 13 00 (cs)

(cs) is the optional checksum.

Returned: VS hc

To Set the X-10-Input-House-Code (Controller will respond only to this HC):

VPX F7 13 hc (cs) ; (note trailing semicolon)

where hc is desired House-Code (FS = 07h = G) (00=disable, 01=A, 02=B...10h=P)

and (cs) is the optional checksum.

Example: This example sets the X-10-Input-HC to F:

VPX F7 13 06 ;

For the desired House-Code, there are 36 individual commands which get mapped to an action in the Vaux system: Unit-Code-1-On, 1-Off, 2-On, 2-Off...16-On, 16-Off, All-Lights-On, All-Units-Off, Dim, and Bright codes, each have a map-table entry.

The “Misc Map” defines actions that take place when BRIGHT, DIM, ALL-LIGHTS-ON, and ALL-UNITS-OFF functions are received:

To Get Current X-10-Input-Misc-Map (Controller will respond only for defined HC):

VP FE 70 10 00 (cs)

(cs) is the optional checksum.

Returned: VX bd bf bz bo dd df dz do nd nf nz no fd ff fz fo ;

where: there are the four fields (device, func, zone, options) of the map entry defining the Vaux operation performed when the specific X-10 function is received (for the House-Code expected):
bd/bf/bz/bo is the map entry for a BRIGHT function (FS = 01,18,01,00 = Zone-1/Tuner Vol-Up)
dd/df/dz/do is the map entry for a DIM function (FS = 01,19,01,00 = Zone-1/Tuner Vol-Dn)
nd/nf/nz/no is the map entry for an ALL-LIGHTS-ON function (FS = 01,14,01,00 = Zone-1/Tuner ON)
fd/ff/fz/fo is the map entry for an ALL-UNITS-OFF function (FS = 01,1C,01,00 = Zone-1/Tuner OFF)

To Set the X-10-Input-House-Code (Controller will respond only for defined HC):

VPX F7 10 bd bf bz bo dd df dz do nd nf nz no fd ff fz fo (cs) ;

The “On Map” defines actions that take place when UNIT-1-ON, UNIT-2-ON...UNIT-16-ON functions are received:

To Get Current X-10-Input-On-Map (Controller will respond only for defined HC):

VP FE 70 11 00 (cs)

(cs) is the optional checksum.

Returned: VX 1d 1f 1z 1o 2d 2f 2z 2o ... 16d 16f 16z 16o ;

where: there are the four fields (device, func, zone, options) of the map entry defining the Vaux operation performed when the specific X-10 function is received (for the House-Code expected):

1d/1f/1z/1o is the map entry for a UNIT-1-ON function (FS = EF,01,00,00 = MACRO-1)

2d/2f/2z/2o is the map entry for a UNIT-2-ON function (FS = EF,02,00,00 = MACRO-2)

3d/3f/3z/3o is the map entry for a UNIT-3-ON function (FS = EF,03,00,00 = MACRO-3)

4d/4f/4z/4o is the map entry for a UNIT-4-ON function (FS = EF,04,00,00 = MACRO-4)

5d/5f/5z/5o is the map entry for a UNIT-5-ON function (FS = EF,05,00,00 = MACRO-5)

6d/6f/6z/6o is the map entry for a UNIT-6-ON function (FS = 00,00,00,00 = nothing)

(etc)...

16d/16f/16z/16o is the map entry for a UNIT-16-ON function (FS = 00,00,00,00 = nothing)

To Set the X-10-Input-On-Map (Controller will respond only for defined HC):

VPX F7 11 1d 1f 1z 1o 2d 2f 2z 2o ... 16d 16f 16z 16o (cs) ;

The “Off Map” defines actions that take place when UNIT-1-OFF, UNIT-2-OFF...UNIT-16-OFF functions are received:

To Get Current X-10-Input-Off-Map (Controller will respond only for defined HC):

VP FE 70 12 00 (cs)

(cs) is the optional checksum.

Returned: VX 1d 1f 1z 1o 2d 2f 2z 2o ... 16d 16f 16z 16o ;

where: there are the four fields (device, func, zone, options) of the map entry defining the Vaux operation performed when the specific X-10 function is received (for the House-Code expected):

1d/1f/1z/1o is the map entry for a UNIT-1-OFF function (FS = EF,06,00,00 = MACRO-6)

2d/2f/2z/2o is the map entry for a UNIT-2-OFF function (FS = EF,07,00,00 = MACRO-7)

3d/3f/3z/3o is the map entry for a UNIT-3-OFF function (FS = EF,08,00,00 = MACRO-8)

4d/4f/4z/4o is the map entry for a UNIT-4-OFF function (FS = EF,09,00,00 = MACRO-9)

5d/5f/5z/5o is the map entry for a UNIT-5-OFF function (FS = EF,0A,00,00 = MACRO-10)

6d/6f/6z/6o is the map entry for a UNIT-6-OFF function (FS = 00,00,00,00 = nothing)

(etc)...

16d/16f/16z/16o is the map entry for a UNIT-16-OFF function (FS = 00,00,00,00 = nothing)

To Set the X-10-Input-Off-Map (Controller will respond only for defined HC):

VPX F7 12 1d 1f 1z 1o 2d 2f 2z 2o ... 16d 16f 16z 16o (cs) ;

X-10-Activity event messages:

Vaux Event Messages are generated when X-10 activity is present on the powerline, due to X-10 controllers other than the Vaux system (a standard event message is generated for Vaux-invoked X-10 output). These messages are generated even for

X-10 functions received for the X-10-Input-House-Code that the Vaux controller is looking for (followed by event message(s) of any actions performed by the Vaux controller, in response to the function.

There is no indication on the front of the controller for X-10 powerline activity (just event messages), except in the case of the X-10-Input-House-Code that the Vaux controller is looking for — these functions will result in lights of mapped functions, as they are performed (e.g. Macro...), or an Error light flash if the map table is empty for a specific X-10 function.

VV FF 15 0A fu hc 00 00

where: fu is the Function or Unit Code

(Func: 14h = ON, 1Ch = OFF, 12h = DIM, 1Ah = BRIGHT, 18h = ALL-LIGHTS-ON, 10h = ALL-UNITS-OFF)

(or Unit: 01h = 1, 02h = 2...10h = 16), and

hc is the House Code for the received X-10 transmission

(01h = A, 02h = B...10h = P)

Note that some activity results in a single message, such as dim/bright commands — there may be repeating dim/brights if the user is holding the button down, although the messages will appear only about every second, not a message for each individual dim/bright on the powerline.

Example: This is an example of an X-10 “DIM-house-code-F” event message:

VV FF 15 0A 12 06 00 00 DIM-F

Most X-10 activity will result in two messages, representing the actual low-level X-10 commands which first “arm” a module, and then perform a function on it.

Example: This is an example of an X-10 “DIM-house-code-F” event message:

VV FF 15 0A 04 06 00 00 arm F4
VV FF 15 0A 1C 06 00 00 ON-F

Aris MediaMation audio/video control (Aris only)

The Type 1 Audio/Video System is used in the Aris Controllers Models 7xx, which 8/4 audio/video input sources; 6/4 audio/video output zones (the model 710 does not support video).

See the Appendix for control codes applicable to the Type 1 A/V system.

Aris MediaMation Controllers, Model 700 and 720i, have an audio/video matrix switching system which provides eight stereo, line-level audio source inputs (four of the eight sources also include composite video inputs), and six zone outputs (four of the six zones also include composite video outputs). The model 710 does not support video. The video switching follows along with the audio switching, and the zone outputs provide volume, bass, and treble adjustment. A host computer has control over all source selection, volume/bass/treble levels, and muting.

The Volume control provides adjustment from -64 dB to $+6$ dB, in 2 dB steps. The Mute level may be programmed to be any volume level down to -64 dB, or the next step down, which is -80 dB. The Bass control provides a -12 dB to $+15$ dB adjustment range, centered at 40 Hz, while Treble may be varied from -12 dB to $+12$ dB, centered at 15 KHz.

When a source is routed to a previously-Off zone, the zone turns On using preset turn-on characteristics; a zone turns Off using preset turn-off characteristics. When a Zone is switched On, it turns on to a fixed -44 dB Volume level, and then ramps up to the programmed Turn-On Volume Level. The Turn-On Volume Ramp is programmable for each zone, and gives the zone turn-on a “soft-start” — there is an corresponding Turn-Off Volume Ramp which tapers the volume down at zone turn-off, and is generally set to a rate faster than turn-on. When a zone turns Off, it ramps down from the Current Volume to -44 dB, and then switches off (to -80 dB). Note: starting/ending the volume ramp at -44 dB allows for a linear ramp effect without wasting time at very-low volume levels below this point. Also note that if the Turn-On Volume Level is set to a value below -44 dB, the system simply turns on to this level (with no ramping applied). In addition to programming via the serial port, a zone’s Turn-On Volume Level may also be set by using an RC-8 Remote Control: adjust the zones’s volume to a desired level and then press the Limit button (Shift-FF) — the current volume level is then stored as the new Turn-On Volume level for this zone.

The zone’s Bass and Treble Turn-On levels are programmable, but they also track user adjustment (via remote control) to allow easy “Zone EQ.”

The Turn-On/Off Ramps are controlled by “delay parameters” — a delay is inserted between each 2 dB volume level step. The Turn-On Volume Ramp is programmable and has a Factory Setting of 20 ms-per-2dB step, and the The Turn-Off Volume Ramp has a Factory Setting of 5 ms-per-2dB step. For example, with the Factory Setting for the Turn-On Volume Level of -20 dB, this gives a total ramp-up time (from -44 dB) of 240 ms (twelve 2-dB steps times 20 ms/step). A ramp down from -20 dB (to -44 dB) takes 48 ms. You may adjust as desired.

If an active (On) zone gets a new source routed to it, the same ramp-down and ramp-up tapers apply, to provide a smooth transition to the new source (returning to the current volume level). Note that the zone’s previous bass and treble levels will still be in effect.

If Passthrough mode is selected for a zone, the Turn-On Volume is ignored, since the zone is routing the source at a fixed 0 dB level. Passthrough mode can be further programmed to use bass and treble turn-on levels, or to force a flat zone at turn-on. When a zone is configured for passthrough operation, the remote control Volume buttons now access infrared code locations. A typical example of Passthrough use is for driving a surround sound preamplifier — volume/bass/treble adjustment on the surround sound input may affect steering and separation. By learning the preamplifier’s IR commands (globally, by storing them in all eight device locations), this Passthrough zone now appears to the user as any other zone. Likewise, pressing the remote control Mute button typically affects the zone directly, but the zone may be programmed to allow the Mute button to access infrared codes instead.

A Zone’s Audio Mode is generally normal stereo (Linear Stereo), but may also be set to Forced Mono, Psuedo-Stereo (from a mono source), or Spatial-Stereo (from a stereo source).

The Factory Settings for Zone Turn-On/Off characteristics are as follows:

Turn-On Volume Level	-20 dB
Turn-On Bass Level	0 dB
Turn-On Treble Level	0 dB
Turn-On Volume Ramp Delay	20 ms (per 2 dB Volume step)
Turn-Off Volume Ramp Delay	5 ms (per 2 dB Volume step)
Mute Level	-80 dB
Passthrough Operation	Off
Audio Mode	Linear Stereo

A host computer may also poll periodically for Zone Status, and update its own state tables. The Status command returns the State of the Zone (Off, On, or Muted), the Current Source, and Current Volume/Bass/Treble Levels.

There is also an additional delay parameter which may be programmed: VBT Ramp Rate for Remote Controls. This allows adjustment of Volume/Bass/Treble ramping when the remote control's buttons are pressed and held. This is a global — not zone-specific.

using the zone turn-on/off default table

Added to Protocol Version: 2.5

The host computer may access a table which provides control over a zone's turn-on characteristics. These settings do not get immediately applied to an active zone — they are applied when the zone is switched On, from an Off state. Refer to the opening discussion for additional information.

To load a new Zone Turn-On/Off Default Table:

VPX 21 10 zz vv bb tt ru rd mm cf am xx (cs) ; (note trailing semicolon)

where (FS = Factory Setting),

zz	Zone	(01h to 06h for group-0, 11h to 16h for group-1, etc.)
vv	Turn-On Volume Level	(1Bh to 3Fh) 32h = -20 dB (FS)
bb	Turn-On Bass Level	(02h to 0Bh) 06h = 0dB is initial FS; tracks user
tt	Turn-On Treble Level	(02h to 0Ah) 06h = 0dB is initial FS; tracks user
ru	Volume Ramp-Up Delay	(01h to FFh) 14h = (20 ms-per-2dB-step) (FS)
rd	Volume Ramp-Down Delay	(01h to FFh) 05h = (5 ms-per-2dB-step) (FS)
mm	Mute Level	(1Bh to 3Fh) 1Bh = -80dB (FS)
cf	Zone Control Flags	cf.7 to cf.3 (reserved — set to 0)
		cf.2 :
		0 = Normal Muting (internal) (FS)
		1 = Use infrared codes for mute
		cf.1
		0 = Passthrough uses bass/treble (FS)
		1 = Passthrough flat
		cf.0
		0 = Normal Volume Control (FS)
		1 = Passthrough at 0 dB (IR Vol codes)
am	Audio Mode	00h = Forced Mono
		01h = Psuedo Stereo (from mono source)
		02h = Linear Stereo (FS)
		03h = Spatial Stereo (from stereo source)
		04h to FFh (reserved)
xx	(reserved — set to 00h)	

Note that the Turn-on bass and treble levels are actually the same as the Current bass and treble levels (unlike volume), to allow a user to EQ a zone from a remote control. The other parameters are static, and are only changed by this command, except volume-limit, which can be changed using the RC-8 remote control.

The Volume Ramp-Up/Down delay parameters correspond to approximately 1 ms resolution, and therefore range from 01h (1 ms) to FFh (255 ms) per step.

Example: This Zone 4 example loads the factory settings:

VPX 21 10 04 32 06 06 14 05 1B 00 02 00 ;

Example: This Zone 2 example shows flat passthrough use:

VPX 21 10 02 32 06 06 14 05 1B 03 02 00 ;

To get a Zone's Turn-On/Off Table:

VP FE 51 zz 00 (cs)

where *zz* is Zone (01h to 06h for group-0, 11h to 16h for group-1, etc.),
and (cs) is the optional checksum.

Returned: VX vv bb tt ru rd mm cf am xx (cs) ;

where the bytes are defined as above.

To set a Zone's Turn-On-Volume-Level only (added ver 3.13)

(handy for on-site system setup use: adjust vol, then set):

VP F0 59 zz 00 (cs)

where *zz* is Zone (01h to 06h for group-0, 11h to 16h for group-1, etc.),
and (cs) is the optional checksum.

getting an A/V zone's status

Added to Protocol Version: 2.5

The host computer may request status of a Zone (e.g.: poll periodically for Zone Status, and update its own state tables). The Status command returns the State of the Zone (Off, On, or Muted), the Current Source, and Current Volume/Bass/Treble Levels. If the Zone is Off, the other parameters reflect a state just before final turn-off. If the Zone is muted, the Current Volume is the level the zone will unmute to.

To get Zone Status information:

VP FE 52 zz 00 (cs)

where zz is Zone (01h to 06h for group-0, 11h to 16h for group-1, etc.),
and (cs) is the optional checksum.

Returned: VX st sr vv bb tt zz (cs);

st	Zone State	00h = Off 01h = On 86h = Muted
sr	Current Source	00h = None; 01h to 08h = Tuner to Tape 2
vv	Current Volume Level	1Bh to 3Fh
bb	Current Bass Level	02h to 0Bh
tt	Current Treble Level	02h to 0Ah
zz	Zone	01h to 06h (added ver 2.80)

source-selection (source-to-zone routes)

Added to Protocol Version: 2.5 for VP commands (2.1 for VPX commands)

(Zone 0 support added ver 2.6/3.11)

The host computer may perform a source-to-zone route (switch), to turn a zone On with a selected source, change the source of an active zone, or turn a zone Off. The Zone Turn-On parameters in effect for that zone will be applied; for example, Turn-On Volume Ramping, which gives the route a softer transition. Refer to the opening discussion for additional information. As of Version 3.11, Zone-0 routes ignore the Zone-Group, for whole-house routing with one command (the same is true for remote-control or keypad operation).

The analogous extended (VPX) command is maintained for compatibility with earlier protocol versions — the VP command is preferred.

To perform a source-to-zone route:

VP 21 zz 01 ss (cs) or VPX 21 20 ss zz (cs) ;

zz	Zone	01h to 06h for group-0, 11h to 16h for group-1, etc — or 00h for all zones.
ss	Source	00h = Zone Off; 01h to 08h = Tuner to Tape 2
(cs)	optional checksum	

Example: This example routes source 8 to zone 2:

VP 21 02 01 08

Example: This example routes source 5 to all zones (Zone = 00h):

VP 21 00 01 05

Example: This example turns off zone 2 (Source = 00h):

VP 21 02 01 00

Example: This example turns off all zones (Source = 00h and Zone = 00h):

VP 21 00 01 00

relative adjustment of volume/bass/treble levels

Added to Protocol Version: 2.5 for VP commands (2.1 for VPX commands)

The current volume/bass/treble levels in effect for a zone may be increased/decreased in single steps (volume steps are 2 dB; bass/treble steps are 3 dB). As of Version 3.11, Zone-0 is allowed, and the Zone-Group is ignored, for whole-house volume-adjustments (the same is true for remote-control or keypad operation).

The analogous extended (VPX) commands are maintained for compatibility with earlier protocol versions — the VP commands are preferred.

To adjust Volume Up one step:

VP 21 zz 18 00 (cs) or VPX 21 E4 zz 00 (cs) ;

To adjust Volume Down one step:

VP 21 zz 19 00 (cs) or VPX 21 E5 zz 00 (cs) ;

To adjust Bass Up one step:

VP 21 zz E0 00 (cs) or VPX 21 E0 zz 00 (cs) ;

To adjust Bass Down one step:

VP 21 zz E1 00 (cs) or VPX 21 E1 zz 00 (cs) ;

To adjust Treble Up one step:

VP 21 zz E2 00 (cs) or VPX 21 E2 zz 00 (cs) ;

To adjust Treble Down one step:

VP 21 zz E3 00 (cs) or VPX 21 E3 zz 00 (cs) ;

where zz is the zone number (01h to 06h for group-0, 11h to 16h for group-1, etc; 00 valid for volume-only) and (cs) is the optional checksum.

Example: This example adjusts zone 4's bass level up by 3 dB:

VP 21 04 E0 00

absolute adjustment of volume/bass/treble levels

Added to Protocol Version: 2.5

The current volume/bass/treble levels in effect for a zone may be set to specific absolute levels.

Volume ramping is NOT applied.

To adjust Volume to absolute Level:

VP 21 zz 02 vv (cs)

zz Zone (01h to 06h for group-0, 11h to 16h for group-1, etc)
vv Volume Level — see Appendix (1Bh-3Fh)

To adjust Bass to absolute Level:

VP 21 zz 03 bb (cs)

zz Zone (01h to 06h for group-0, 11h to 16h for group-1, etc)
bb Bass Level — see Appendix (02h-0Bh)

To adjust Treble to absolute Level:

VP 21 zz 04 tt (cs)

zz Zone (01h to 06h for group-0, 11h to 16h for group-1, etc)
tt Treble Level — see Appendix (02h-0Ah)

zone muting and flattening

To Flatten Zone (set bass and treble to 0dB): Added to Version: 2.5

VP 21 zz E4 00 (cs)

zz Zone (01h to 06h for group-0, 11h to 16h for group-1, etc)

To Mute Zone (to Zone's Mute Level): Added to Version: 2.5

VP 21 zz 1B rr (cs)

zz Zone (01h to 06h for group-0, 11h to 16h for group-1, etc)

rr Ramping (00h = No Ramping, 01h = Ramping)

To Unmute Zone (to return to pre-mute level): Added to Version: 2.5

VP 21 zz 28 rr (cs)

zz Zone (01h to 06h for group-0, 11h to 16h for group-1, etc)

rr Ramping (00h = No Ramping, 01h = Ramping)

To Toggle Mute/Unmute Zone: Added to Version: 2.6

VP 21 zz 30 rr (cs)

zz Zone (01h to 06h for group-0, 11h to 16h for group-1, etc)

rr Ramping (00h = No Ramping, 01h = Ramping)

NOTE: as of ver 3.11, whole-house toggle mute may be performed across multiple Aris controllers on different zone groups (no ramping applied in this case) (ie: group ignored and no ramping for zones 00, 10, 20, 30, 40, 50, 60, 70, 80, or 90).

adjusting VBT ramp rate for remote controls

Added to Protocol Version: 2.5

The host computer may change a delay parameter which affects the rate at which volume, bass, and treble levels will ramp up and down when using a remote control. This parameter may be changed if a faster or slower VBT rate is desired when using the remote control. The factory setting is 128 ms (80h) — this is inserted between each step (volume is 2 dB/step; bass/treble are 3 dB/step). The delay parameter corresponds to approximately 1 ms resolution, and therefore ranges from 01h (1 ms) to FFh (255 ms). This ramp rate is global for all zones.

To Change the VBT rate for Remote Controls:

VP F0 60 rr 00 (cs)

where rr is Ramp delay (01h - FFh) (FS = 80h),

and (cs) is the optional checksum.

To Get Current VBT rate for Remote Controls:

VP FE 50 00 00 (cs)

(cs) is the optional checksum.

Returned: VS rr

HVAC devices

Future Energy Management Expansion Modules can plug in to VauxNet. They are controlled interactively from the Vaux remote control using a Temperature button, which provides zone-specific setpoints of:

- Relative temperature set: $\pm 1^\circ$ to $\pm 9^\circ$ F
- Absolute temperature set: 00 to 99° F

Remote Control-initiated commands are broadcast on VauxNet and sent as Vaux Event Messages (with a Function Not Performed status) by the Vaux Controllers. The RC-8 Remote Control provides temperature setpoint commands. VauxSerial-initiated commands are also broadcast on VauxNet.

The host computer may listen for HVAC Event Messages and control the HVAC system directly (via an appropriate interface), or the computer may send HVAC Commands to the VauxSerial port and control the system through future Vaux Expansion Modules.

To Control an HVAC Device (invokes a VauxNet broadcast packet):

VP 09 zz ff 00 (cs)

where data bytes are as follows:

zz	Zone Code — see Appendix (00h-09h)
ff	Function Code — see Appendix (00h-99h, E1h-E9h, F1h-F9h)
(cs)	Optional Checksum (if enabled — see Configuration Command)

Returned: Command Accepted (VI 80) or Invalid Control Cmd (VE 03) Message.
An Event Message may also result.

Example: This example line will set the master bedroom (zone 5) setpoint to +2° F from the current setpoint (checksumming is off):

VP 09 05 E2 00

security devices:

Future Security Expansion Modules can plug in to VauxNet. They are controlled interactively from the Vaux Remote Controls using the Security button to initiate commands for zone-based arming/disarming of a security system (Zone, digit, On/Off). There are nine zones plus a whole-house zone defined. Note that this scheme may not be appropriate for security use if a) the Vaux Remote Control is located in an unsecured area (there is no password access protection), or, b) you are uncomfortable with a non-encrypted and predictable RF transmission controlling your security system.

Remote Control-initiated commands are broadcast on VauxNet and also sent as Vaux Event Messages (with a Function Not Performed status) by the Vaux Controllers. VauxSerial-initiated commands are also broadcast on VauxNet.

These (Remote-initiated) codes are also available as general input to the computer to provide On and Off control of up to ten arbitrary devices — in this case the computer can simply monitor for the Event Messages and take action on the desired codes.

The computer may listen for Security Event Messages and control the Security system directly (via an appropriate interface), or the computer may send Security Commands to the VauxSerial port and control the system through future Vaux Expansion Modules.

To Control a Security Device (invokes a VauxNet broadcast packet):

VP 0B zz ff 00 (cs)

where data bytes are as follows:

zz	Zone Code — see Appendix (00h-09h)
ff	Function Code — see Appendix (On/Arm = 14h, Off/Disarm = 1Ch)
(cs)	Optional Checksum (if enabled — see Configuration Command)

Returned: Command Accepted (VI 80) or Invalid Control Cmd (VE 03) Message.
An Event Message may also result.

Example: This example line will arm the whole house (checksumming is off):

VP 0B 00 14 00

MACROs

(not applicable to PCremote)

The Altrix, Aris, and MARC II Controllers each store twenty 25-step MACRO functions, which may be played from the Vaux remote control MACRO buttons, or from a Vaux Protocol command (listed below). Note: the systems originally had ten MACROs; this was increased to twenty MACROs in version 2.75.

When the MACRO is played by the user, or computer, the events (steps) defined in the MACRO list get copied to the system action queue, where they are dispatched (executed) sequentially. The events are executed in the order in which they were entered in the MACRO list definition. For example, MACRO 1 may be defined to dim down several lamps, turn on the stereo, and then play CD track 5.

The user may program general key-sequence MACROs from the remote control keyboard — also, there are additional system events which may be included in a MACRO definition by programming the MACRO using a Vaux Protocol command (listed below). Each of the 20 MACRO definitions can hold 25 events — an event in the Vaux domain is a four-byte structure: Device, Function, Zone, and Options (see the appendix for an overview). This covers all devices included in the current implementation of Vaux Protocol (A/V routing/Volume/Bass/Treble/Mute, learned infrared commands, lights/appliances...), and other devices (security, HVAC...) which may be system options and whose control commands will get broadcast on VauxNet (for optional expansion module interception). Note that Zone and Options information may not apply to some Device/Function commands, and may be entered as 00h.

What comprises a single event (or step)?

- An infrared command (e.g. 02h 0Bh 05h 00h is CD, Disc Skip, Zone 5)
- A lighting command (e.g. 0Ah 14h 69h 00h is Turn ON module A6)
- (etc.)

MACRO Definition Data in uploads and downloads:

The definition of a MACRO consists of a byte indicating the number of events (steps) in the MACRO (01h to 19h — 1 to 25 steps), followed by a four-byte structure (Device, Function, Zone, Options) for each event in the MACRO.

MACRO Event Messages:

When a MACRO is invoked, it will result in an Event message indicating which MACRO is playing, followed by Event messages for each of the MACRO's steps. The MACRO playback Event Message has the form:

VV FF 30 EF 00 00 mm 00

where mm is the number (in hex) of the MACRO.

To Clear MACRO Memory:

VP F0 12 00 00 (cs)

Returned: Command Accepted (VI 80) Message.

To Play a MACRO:

VP F0 30 mm 00 (cs)

where data bytes are as follows:

mm MACRO Number (01h to 14h — MACRO 1 to 20)
(cs) Optional Checksum (if enabled — see Configuration Command)

Returned: Command Accepted (VI 80), Empty MACRO (VI 85),
or Invalid MACRO Number (VE 30) Message.
Event Message(s) will also result from each event in the MACRO.

Example: This example line will play MACRO 6 (checksumming is off):

VP F0 30 06 00

To Upload (get) a MACRO Definition:

VP FE 41 mm 00 (cs)

where data bytes are as follows:

mm MACRO Number (01h to 14h — MACRO 1 to 20)
(cs) Optional Checksum (if enabled — see Configuration Command)

Returned: MACRO Definition (VX xx xx xx xx...xx (cs) ;) and Command Accepted
(VI 80) Message, Empty MACRO (VI 85), or Invalid MACRO Number
(VE 30) Message.

Example: This example line gets MACRO 4's definition which has two steps —
turn on light A2 and CD (IR Dev 2) Play (no checksumming):

**VP FE 41 04 00
VX 02 0A 14 67 00 02 0D 00 00 ;**

To Download (program) a MACRO Definition:

VPX FB mm xx xx xx xx...xx (cs) ; (Note trailing semicolon)

where data bytes are as follows:

mm MACRO Number (01h to 14h — MACRO 1 to 20)
xx... MACRO Definition Data
(cs) Optional Checksum (if enabled — see Configuration Command)

Returned: Command Accepted (VI 80), or Invalid MACRO Number (VE 30) Message.
 Bogus data is not trapped.

Example: This example line programs MACRO 7 with three steps — turn off A2,
 dim A2 down slightly (1 of these dims = 5 X-10 dims of 32 total),
 and put the laserdisc in Play (no checksumming):

VPX FB 07 03 0A 1C 67 00 0A 12 67 00 02 0D 00 00 ;

M7 3 steps A2 OFF DIM A2 CD Play

Example: This example line programs MACRO 5 with three steps — turn on K15,
 put the CD in Play for Zone 2, and skip a track (no checksumming):

VPX FB 05 03 0A 14 C2 00 E0 02 00 00 06 0D 00 00 ;

M5 3 steps K15 ON Delay Laserdisc Play

appendix 1: vaux protocol device codes

Device Labeling:

To select IR Devices (or Aris Sources), the the RC-8 Remote Control uses buttons, while the earlier Vaux Console and RC-1 used a slide switch. With an Aris Controller, the device also corresponds to the input source to be selected when turning a zone on (the Aris sources marked with an * include video input as well as audio).

<u>Device/Source</u>	<u>RC-8 Remote</u>	<u>CON-1 Console</u>	<u>RC-1 Remote</u>
1	Tuner	Tuner	Tuner
2	CD	CD	CD
3 *	Cab/Sat	Aux	Aux
4 *	VCR 1	Video 1	Video
5 *	VCR 2	Video 2	-
6 *	Laser	Laser	-
7	Tape 1	-	-
8	Tape 2	-	-

The Device Code byte indicates the system being controlled. Current devices are defined as follows:

<u>Device Code</u>	<u>Device Addressed</u>
00h	Current Infrared/Source-Device
01h	Infrared/Source-Device 1 (RC-8 label: Tuner)
02h	Infrared/Source-Device 2 (RC-8 label: CD)
03h	Infrared/Source-Device 3 (RC-8 label: Cab/Sat)
04h	Infrared/Source-Device 4 (RC-8 label: VCR 1)
05h	Infrared/Source-Device 5 (RC-8 label: VCR 2)
06h	Infrared/Source-Device 6 (RC-8 label: Laser)
07h	Infrared/Source-Device 7 (RC-8 label: Tape 1)
08h	Infrared/Source-Device 8 (RC-8 label: Tape 2)
09h	HVAC (Heating, Ventilation, and Air Conditioning)
0Ah	Lighting and Appliances (X-10)
0Bh	Security System

<u>Device Code</u>	<u>Device Addressed</u>
0Ch	Reserved (broadcast on VauxNet)
0Dh	Reserved (broadcast on VauxNet)
0Eh	Reserved (broadcast on VauxNet)
0Fh	Reserved (broadcast on VauxNet)
10h	Lighting and Appliances (X-10 Low-Level Commands)
11h - 20h	Reserved (broadcast on VauxNet)
21h	A/V System Type 1
22h	A/V System Type 2
23h - DFh	Reserved (broadcast on VauxNet)
E0h - FFh	Vaux Protocol Special "Devices"
E0h	Delay "Device" (mapped)
E1h	Sleep "Device"
E2h	Source-Select "Device"
EFh	Macro-Play "Device"
F0h	System Control "Device"
F1h	Net node diagnostic: Read Word
F2h	Net node diagnostic: Write Word
F3h	Net node diagnostic: Control command
F7h	X10 Input Program "Device"
F9h	Infrared Memory Program "Device"
FAh	(removed)
FBh	MACRO Memory Program "Device"
FCh	(removed)
FDh	SecureID Code Program "Device"
FEh	Request Command "Device"
FFh	Configure Command "Device"

The Vaux RF (and IR version) Remote Control RC-8 can access Infrared Devices 1 to 8.

The Vaux RF Console CON-1 can access Infrared Devices 1 to 6.

The Vaux RF Remote Control RC-1 can access Infrared Devices 1 to 4.

appendix 2: vaux protocol function codes

The Function Code byte defines the operation to be performed (on the device indicated, and for the zone indicated) — it generally has a context-sensitive interpretation depending on the device being controlled.

For Device 0Ah, X-10 lighting and appliance operation, the Function Code byte defines the X-10 operation desired:

<u>Function Code:</u>	<u>X-10 Operation</u>
14h (20d)	On (when Unit Code is 0, this becomes “All Lights On”)
1Ch (28d)	Off (when Unit Code is 0, this becomes “All Units Off”)
12h (18d)	Dim (same as interactive button-press: 5 actual X-10 dims)
1Ah (26d)	Bright (same as interactive button-press: 5 actual X-10 brights)

For Device 10h, X-10 Low-Level Commands, the Function Code byte defines the X-10 operation desired:

NOTE: Any function code may be sent, not just those noted below. Repeat code in Low-Level command must be 0 to send normal function command (such as On), and must be 1 to 255 to send a repeating function (such as Dim).

<u>Function Code:</u>	<u>Repeat Code:</u>	<u>X-10 Operation</u>
1Ch (28d)	0	Off
14h (20d)	0	On
12h (18d)	1 - 255	Dim (1 - 255 actual X-10 dims)
1Ah (26d)	1 - 255	Bright (1 - 255 actual X-10 brights)

For device codes 01h to 08h, the infrared command sets, the function code is a pointer to a memory area holding a “Learned” infrared code signal. There are 30 infrared storage locations for each device (earlier systems used 6 devices at 40 functions/device). The following table lists which remote control buttons access the same IR code:

Function Code (for Devices 1-8):	Addressable from RC-8 Remote	Addressable from CON-1 Console	Addressable from RC-1 Remote
00h	“0”	“0”	“0”
01h	“1”	“1”	“1”
02h	“2”	“2”	“2”
03h	“3”	“3”	“3”
04h	“4”	“4”	“4”
05h	“5”	“5”	“5”
06h	“6”	“6”	“6”
07h	“7”	“7”	“7”
08h	“8”	“8”	“8”
09h	“9”	“9”	“9”
0Ah	“F2/up”	“+10”	“+10/Enter”
0Bh	“Disc Skip”	“Disc Skip”	“AB/Disc Skip”
0Ch	“Stop”	“Stop”	“Stop”
0Dh	“Play”	“Play”	“Play”
0Eh	“Pause”	“Pause”	“Pause”
0Fh	“Rewind”	“Rewind”	“Rewind/Last”
10h	“Fast Fwd”	“Fast Fwd”	“Fast Fwd/Next”
11h	“Record”	“Record”	“Record”
12h	“Channel Down”	“Channel Down/-“	“Channel /Dim Dn”
13h	“F4/down”	“Sleep”	“Sleep”
14h	“On”	“On”	“Power/On”
15h	“Enter/+10”	“Enter”	-
16h	“F3/right”	“Skip Fwd”	-
17h	“F1/left”	“Skip Back”	-
18h	“Volume Up”	“Volume Up”	“Volume Up”
19h	“Volume Down”	“Volume Down”	“Volume Down”
1Ah	“Channel Up”	“Channel Up/+”	“Channel Up/Dim”
1Bh	“Mute”	“Mute”	“Mute”
1Ch	“Off”	“Off”	“Off”
1Dh	“A/B”	“A/B”	-

The following “obsoleted” buttons work with earlier MARC II/Aris systems (version 2.2 and earlier); these Controllers used 6 IR Devices @ 40 Codes/Device.

These codess should be avoided with later systems:

Function Code (for Devices 1-8):	Addressable from RC-8 Remote	Addressable from CON-1 Console	Addressable from RC-1 Remote
1Eh	-	“Repeat”	-
1Fh	-	“Scan”	-
20h	-	“Shuffle”	-
21h	-	“Shift-1”	“Source-1”
22h	-	“Shift-2”	“Source-2”
23h	-	“Shift-3”	“Source-3”
24h	-	“Shift-4”	“Source-4”
25h	-	“Shift-5”	“Source-5”
26h	-	“Shift-6”	“Source-6”
27h	-	-	-

For Device 09h, HVAC, the (Zone-specific) Function Codes are defined as:

<u>Function Code:</u>	<u>HVAC Operation</u>
00h (00d)	Absolute Temp. setpoint: 0° (°F or C, as appropriate).
01h (01d) (etc.)...	Absolute Temperature setpoint: 1°
99h (153d)	Absolute Temperature setpoint: 99°
E1h (225d)	Relative Temperature setpoint: +1°
E2h (226d) (etc.)...	Relative Temperature setpoint: +2°
E9h (233d)	Relative Temperature setpoint: +9°
F1h (241d)	Relative Temperature setpoint: -1°
F2h (242d) (etc.)...	Relative Temperature setpoint: -2°
F9h (249d)	Relative Temperature setpoint: -9°.

For Device 0Bh, Security Systems, the (Zone-specific) Function Codes are:

<u>Function Code</u>	<u>Security Operation</u>
14h (20d)	Arm (or On)
1Ch (28d)	Disarm (or Off)

appendix 3: vaux protocol zone codes

The Zone Code byte is context-sensitive — its interpretation is dependent on the Device being controlled. It is generally a Vaux Standard Zone reflecting the room being controlled (e.g. adjust temperature in master bedroom, or mute audio in entire house).

For Aris MediaMation Controllers, the Zone is affecting the internal zone control for routing and Volume/Bass/Treble/Mute adjustments.

For Controllers placed in Zone-Specific IR output mode, the Remote Control's Zone is selecting the appropriate IR emitter port.

For X-10 (Device 0Ah) only, the Zone Code byte directly addresses 1 of 240 individual X-10 modules for “On”, “Off”, “Dim”, or “Bright” commands, or 1 of 16 groups of modules for “All Lights On/All Units Off” commands. The X-10 House Codes A through P, are represented by the Vaux Zone byte upper nibble hex values 0 through F, as mapped by the (X-10-defined) codes shown below. The X-10 Unit Codes 1 through 15, are represented by the lower nibble hex values 1 through F, as shown below.

X-10 House Code	Vaux Zone Code Upper Nibble	X-10 Unit Code	Vaux Zone Code Lower Nibble	Vaux Zone Digits
A	6h	1	6h	1
B	7h	2	7h	2
C	4h	3	4h	3
D	5h	4	5h	4
E	8h	5	8h	5
F	9h	6	9h	6
G	Ah	7	Ah	7
H	Bh	8	Bh	8
I	Eh	9	Eh	9
J	Fh	10	Fh	0
K	Ch	11	Ch	
L	Dh	12	Dh	
M	0h	13	Note 1	
N	1h	14	1h	
0	2h	15	2h	
P	3h	16	3h	

Note 1: An X-10 Unit Code of 0h is a special case: it indicates an “All Lights On/All Units Off” command is being requested for this House Code (in the upper nibble). Therefore, there are 16 X-10 addresses which are not individually available (A13-P13) since they have been replaced by “All Lights On/All Units Off” commands.

For Low-Level X-10 (Device 10h) only, there are similarly-defined House and Unit Code bytes used (all 256 X-10 addresses are accessible):

X-10 House Code	Protocol Command House Code Byte	X-10 Unit Code	Protocol Command Unit Code Byte
A	06h	1	06h
B	07h	2	07h
C	04h	3	04h
D	05h	4	05h
E	08h	5	08h
F	09h	6	09h
G	0Ah	7	0Ah
H	0Bh	8	0Bh
I	0Eh	9	0Eh
J	0Fh	10	0Fh
K	0Ch	11	0Ch
L	0Dh	12	0Dh
M	00h	13	00h
N	01h	14	01h
0	02h	15	02h
P	03h	16	03h

appendix 4: Aris MediaMation control codes

These are the data values for the Type 1 system for volume, bass, and treble setting.

<u>vd</u>	<u>Volume Gain</u>
3Fh (63d)	+ 6 dB (max)
3Eh (62d)	+ 4 dB
3Dh (61d)	+ 2 dB
3Ch (60d)	0 dB (passthrough)
3Bh (59d)	- 2 dB
3Ah (58d)	- 4 dB
39h (57d)	- 6 dB
38h (56d)	- 8 dB
37h (55d)	- 10 dB
36h (54d)	- 12 dB
35h (53d)	- 14 dB
34h (52d)	- 16 dB
33h (51d)	- 18 dB
32h (50d)	- 20 dB
31h (49d)	- 22 dB
30h (48d)	- 24 dB
2Fh (47d)	- 26 dB
2Eh (46d)	- 28 dB
2Dh (45d)	- 30 dB
2Ch (44d)	- 32 dB
2Bh (43d)	- 34 dB
2Ah (42d)	- 36 dB
29h (41d)	- 38 dB
28h (40d)	- 40 dB
27h (39d)	- 42 dB
26h (38d)	- 44 dB
25h (37d)	- 46 dB
24h (36d)	- 48 dB
23h (35d)	- 50 dB
22h (34d)	- 52 dB
21h (33d)	- 54 dB
20h (32d)	- 56 dB
1Fh (31d)	- 58 dB
1Eh (30d)	- 60 dB
1Dh (29d)	- 62 dB
1Ch (28d)	- 64 dB
1Bh (27d)	< -80 dB (min — muted volume)

<u>bd</u>	<u>Bass Gain</u>
0Bh (11d)	+ 15 dB
0Ah (10d)	+ 12 dB
09h (9d)	+ 9 dB
08h (8d)	+ 6 dB
07h (7d)	+ 3 dB
06h (6d)	0 dB (flat)
05h (5d)	- 3 dB
04h (4d)	- 6 dB
03h (3d)	- 9 dB
02h (2d)	- 12 dB

(centered @ 40 Hz)

<u>td</u>	<u>Treble Gain</u>
0Ah (10d)	+ 12 dB
09h (9d)	+ 9 dB
08h (8d)	+ 6 dB
07h (7d)	+ 3 dB
06h (6d)	0 dB (flat)
05h (5d)	- 3 dB
04h (4d)	- 6 dB
03h (3d)	- 9 dB
02h (2d)	- 12 dB

(centered @ 15 KHz)

appendix 5: an interactive Vaux Protocol example

Here is an example of an interactive session using a MarcPCI and MARC System.

The underlined characters are returned from PCI to the computer:

```
V> (vp ff 3f 00 3f) VI 80                (Configuration: enable prompts and character echo)
V> vp 01 03 18 00 VI 80                (IR Device 01h, Zone 03h, "Vol Up") (VI 80 = Cmd OK)
V> vp ff 3d 00 3d VI 80                (Configuration: turn on checksumming)
V> vp 01 03 18 00 23 VE 00                (VE 00 = Invalid Checksum)
V> vp 01 03 18 00 1C VI 80                (VI 80 = Cmd Accepted)
V> vp ff 3f 00 11 VE 01                (VE 01 = Invalid Configuration Checksum)
V> vp fe 01 00 00 ff VS 3D                (Get Configuration Byte A = 3D)
V> vp fe 00 00 00 fe VS 18                (Get Protocol Version Level = 1.8)
V> vp ff 3f 00 3f VI 80                (Configuration: turn off checksumming)
V> vp fe 02 00 00 VS 66                (Get Controller Model = PCI)
V> vp fe 03 00 00 VR 1070                (Get current Security ID Code = 1070bcd)
V> vp fd 8497 1b VI 80                (Set Security ID Code to 8497bcd)
V> vp fe 03 00 00 VR 8497                (Get current Security ID Code = 8497bcd)
V> vp fe 21 00 00 VE 02                (VE 02 = Invalid Request Command)
V> vp e7 01 23 00 VE 03                (VE 03 = Invalid Control Command (dev e7))
V> vp 0a 67 14 00 VI 80                (Device = X-10 (0Ah), Module A2 (67h), "On" (14h))
V> VV FF 11 0A 14 67 00 00                (No X-10 Interface found (11h))
V> vp 0a 67 14 00 VI 80                (Plug in X-10 Interface, and try again)
V> VV FF 01 0A 14 67 00 00                (Function Performed (01h) for above module)
V> vp 0w VI 82                (Invalid character in stream)
V> vp 01 3 VI 81                (Command Aborted — pressed ESCAPE)
```

appendix 6: vauxserial port rs-232 interface

rs-232 connection to the vauxserial port

The VauxSerial Port is designed to connect directly to any computer (or asynchronous data terminal) which has a standard RS-232 serial port. An RS-232 Serial Cable is needed to connect the VauxSerial Port to the Computer Serial Port. This cable is not supplied with the Vaux interfaces due to differing serial port connectors found on various computers.

The VauxSerial Port uses a DB-25 F (female) connector and is configured as if it were a modem. In RS-232 terminology, the VauxSerial Port is wired as a DCE (i.e. it receives data on pin 2, and transmits data on pin 3 of its DB-25 F connector). If you currently have a cable for connecting your computer to a modem, it will likely work with your VauxSerial Port. Vaux also offers optional Serial Cables for IBM PC®-compatible and Macintosh® computers. You may also choose to wire your own cable — see later sections. Connection of Hardware Handshaking lines may be necessary if you choose hardware handshaking instead of the factory setting: software handshaking.

Note: If you are using hardware handshaking, the computer-generated signals (RTS and/or DTR) must be driven properly, driven always-active, or disconnected from the VauxSerial Port — otherwise, the Vaux Controller will accept and process only the first command sent to it (it will wait for handshaking in order to send the Command Accepted message back to the computer).

The VauxSerial interface generally communicates at 9600 baud (bits-per-second) using 8 bits of data, no parity, and one stop bit. The Baud Rate can be set to 1200, 2400, 4800, or 9600 baud (see the System Control Command). The Serial Port on the controlling computer or terminal must be configured according to its manufacturer's instructions.

Note: Not all RS-232 Serial Ports are the same, and some may require certain pins to be asserted before the port becomes active. If your cable is not providing Hardware Handshake signals from the VauxSerial Port, and your computer's Serial Port is requiring a handshake input (on CTS, DSR, and/or DCD), you will likely need to change your computer's Serial Port configuration to ignore these lines. The Vaux end does not need special treatment — the VauxSerial handshake inputs (from RTS and/or DTR) may be left unconnected and they will assume an *always true* state (effectively, it will presume that an infinitely-large receive buffer exists at the computer end of the cable).

Information transfer between the controlling computer and the VauxSerial Port is in the Vaux Protocol™ format — a Vaux-defined command language. The Vaux Protocol format is comprised of various command and message information streams, as defined in this Vaux Protocol Specification. These information streams are short sequences of ASCII characters which send control commands (for example, to press Play on a VCR), or commands which request a reply message. While the VauxSerial interface will usually be connected to a control computer, a simple ASCII terminal may also be used for testing or controlling the system using the Vaux Protocol Commands.

vauxserial port rs-232 interface (DB-25 F) connector

Pin	Description	Signal Direction	VauxSerial Use
1	Frame Ground	—	[not used]
2	TXD (Transmit Data)	Computer —> Vaux	Data In
3	RXD (Receive Data)	Computer <— Vaux	Data Out
4	RTS (Request To Send)	Computer —> Vaux	Handshake In*
5	CTS (Clear To Send)	Computer <— Vaux	Handshake Out**
6	DSR (Data Set Ready)	Computer <— Vaux	Handshake Out**
7	Signal Ground	—	Signal Ground
8	DCD (Data Carrier Detect)	Computer <— Vaux	Handshake Out**
20	DTR (Data Terminal Ready)	Computer —> Vaux	Handshake In*
22	RI (Ring Indicator)	Computer <— Vaux	[not used]

Note:

- TXD and Signal Ground are minimal connections for control.
- RXD is necessary for status-polling, message parsing, etc.

** At least one of the Handshake Out signals (CTS, DSR, DCD) should be connected to, and sensed by, your computer to control data flow *from the computer*, if you are using hardware handshaking.

At least one of the Handshake In signals (RTS, DTR) should be connected to, and driven by, your computer to control data flow *to the computer*, if you are using hardware handshaking.

All lines listed as not used, and other lines not listed, are not connected (to anything) in the VauxSerial interface.

The shield rim (the “D”) is connected to signal ground.

vaux-to-IBM PC®/terminal (DB-25) cable connections

This cable uses a DB-25 F connector to plug into the PC/Terminal, and a DB-25 M connector to plug into the VauxSerial Port:

PC/Term DB-25 F Pin(s)	Description	Signal Direction	VauxSerial DB-25 M Pin(s)
2	TXD	Computer → Vaux	2
3	RXD	Computer ← Vaux	3
4	RTS	Computer → Vaux	4
5	CTS	Computer ← Vaux	5
6	DSR	Computer ← Vaux	6
7	Signal Ground	—	7
8	DCD	Computer ← Vaux	8
20	DTR	Computer → Vaux	20

vaux-to-IBM AT-style (DB-9) cable connections:

This cable uses a DB-9 F connector to plug into the PC, and a DB-25 M connector to plug into the VauxSerial Port:

PC DB-9 F Pin(s)	Description	Signal Direction	VauxSerial DB-25 M Pin(s)
3	TXD	Computer → Vaux	2
2	RXD	Computer ← Vaux	3
7	RTS	Computer → Vaux	4
8	CTS	Computer ← Vaux	5
6	DSR	Computer ← Vaux	6
5	Signal Ground	—	7
1	DCD	Computer ← Vaux	8
4	DTR	Computer → Vaux	20

vaux-to-Macintosh® (DIN-8) cable connections:

This cable uses a DIN-8 M connector to plug into the Macintosh (and convert it to an RS-232-compatible port), and a DB-25 M connector to plug into the VauxSerial Port:

Macintosh DIN-8 M Pin(s)	Description	Signal Direction	VauxSerial DB-25 M Pin(s)
3	TXD	Computer → Vaux	2
5	RXD	Computer ← Vaux	3
1	Handshake Out	Computer → Vaux	4 and/or 20
2	Handshake In	Computer ← Vaux	5, 6, and/or 8
4 and 8 6	Signal Ground (leave open)	—	7

Note: The Macintosh uses RS-422 serial ports. These ports are differential, that is, instead of driving a signal line against ground (e.g. RS-232), there are signal+ and signal- lines which drive against each other, symmetrically around ground. A single-ended receiver, such as RS-232, senses whether the received signal is sufficiently negative (with respect to ground) to be a logic “1,” whereas the differential RS-422 receiver simply senses which line is more negative than the other. This makes RS-422 more immune to noise and interference, and able to drive long cables more reliably. If you leave the signal+ side of an RS-422 transmitter open (e.g. pin 6 on the Mac), and ground the signal+ side of an RS-422 receiver (e.g. pin 8 on the Mac), you have converted the port to EIA standard RS-423, which can be used to connect to most RS-232 ports. Note: at least one commercially-available Mac-to-modem cable illegally grounds pin 6, which should be open.

vaux-to-modem (DB-25) cable connections:

This cable uses a DB-25 M connector to plug into the Modem, and a DB-25 M connector to plug into the VauxSerial Port. This is a “null-modem” (reversing) cable:

Modem DB-25 M Pin(s)	Description	Signal Direction	VauxSerial DB-25 M Pin(s)
2	RXD ← TXD	Modem ← Vaux	3
3	TXD → RXD	Modem → Vaux	2
7	Signal Ground	—	7
4	RTS ← CTS	Modem ← Vaux	5
5	CTS → RTS	Modem → Vaux	4
6	DSR → DTR	Modem → Vaux	20
20	DTR ← DSR	Modem ← Vaux	6
8	DCD → N/C	Modem → Vaux	N/C

VauxSerial Port Handshaking:

The factory setting for the Aris and MARC II (Version 2.1 and up) Systems uses Software Handshaking for data flow control over the RS-232 serial link. For the earlier PCI/MARC, PCremote, and Version 2.0 MARC II Systems, the factory setting uses Hardware Handshaking.

The RS-232 standard provides for communication between Data Terminal Equipment (or DTE; typically a computer or terminal) and Data Communications Equipment (or DCE; typically a modem). Handshaking (also called flow control) between the DTE and the DCE provides a mechanism by which the slower side may halt communications briefly while it attends to another task. If an RS-232 link is full duplex (both sides may transmit simultaneously), and both sides of the link have infinitely large receive buffers, there is no need for any handshaking (unless there is a need to keep the tasks at both ends in synchronization). Since many RS-232 links are half duplex (only one side may transmit at a time), and receive buffers may be as small as one character, some form of handshaking is generally necessary.

When configured to use Software Handshaking, the Vaux system will send XOFF (Ctrl-S) and XON (Ctrl-Q) characters to suspend and resume (respectively) the computer's transmission of RS-232 characters to the VauxSerial Port. The Vaux Controller will also respond to computer-generated XON/XOFF characters and will adjust data flow out of the Vaux system.

Hardware Handshaking uses additional RS-232 communication lines to provide "OK-to-Send" signals between the two sides. There are various ways of implementing Hardware Handshaking; two common forms use the RTS/CTS lines:

- 1) One-way, "computer-to-modem-only" RTS/CTS flow control: a) the computer (DTE) sets the Request To Send (RTS) line true, telling the modem (DCE) that it has a character to send, b) if the modem is ready to receive a character, it responds by setting the Clear To Send (CTS) line true. Note that this RTS/CTS handshaking only controls data flowing from the computer to the modem. Generally the computer has a large receive buffer (and is fast enough to keep up with the data) so flow control in the modem-to-computer direction is not always necessary. Some DCE devices simply connect the RTS and CTS pins together so that the computer-generated RTS signal is sent right back to it on the CTS line. Generally, the DCE does not need the RTS — it can just set the CTS line when it has buffer capacity to receive data, and clear it otherwise.
- 2) Two-way RTS/CTS flow control (not "classic" RS-232): a) the DCE (modem) handles its CTS line as would be expected — it sets the CTS line when it has buffer capacity to receive data, and clears it otherwise, b) the DTE (computer) treats its RTS line as if it were a CTS signal — it sets the RTS line when it has buffer capacity to receive data, and clears it otherwise. The Request-To-Send name may seem confusing, but this scheme provides bidirectional, peer-to-peer flow control using existing signal lines. Of course, both ends of the system must agree on the definition.

The computer side of the system (DTE) also generates the Data Terminal Ready (DTR) signal; when this signal is true, the modem may transmit data to the computer. The DTR line may also be used as a handshake line, but dropping the DTR line usually causes the DCE side to close the connection (for modem use, that is). Modem (DCE) devices also generate Data Set Ready (DSR) and Data Carrier Detect (DCD) signals which inform the computer that the modem is ready to communicate.

If the Vaux system is configured to use Hardware Handshaking, it will control flow in the computer-to-Vaux direction by sending its handshake signal (OK-to-send-to-Vaux) out on three lines: CTS, DSR, and DCD; this is done to ensure that most cables and computer Serial Port configurations will recognize at least one of these handshake inputs for flow control. Likewise, to control flow in the Vaux-to-computer direction, the Vaux Controller looks at both the RTS and the DTR lines for the handshake signal (OK-to-send-to-computer). These lines are logically combined, such that if either of them goes low, the Vaux system will suspend any data transmission. Again, this is done to ensure that most cables and computer Serial Port configurations will provide at least one of these handshake outputs for flow control. To summarize, the computer must check one or all of its CTS/DSR/DCD handshake input lines (the VauxSerial Port drives all three lines with the same signal) for a true state before sending characters to the Vaux system. When (or if) the computer's receive buffer is nearly full, the RTS and/or DTR line must be driven false to pause transmission from the Vaux system.

Hardware Handshaking Notes:

You may elect to not use Hardware (or Software) Handshaking under certain conditions. First, an explanation of Hardware Handshaking on the Vaux-side of the system is in order.

Direct-connect VauxSerial Ports: (1) the computer will send the Vaux Controller a command when the Vaux handshake line is true; (2) the Vaux system will respond by sending the computer a “Command Accepted” Informational Message (if enabled), and will set the handshake line false; (3) the system processes the command (this can take up to a second for an X-10 transmission); (4) the system sets the handshake line true and sends a “Function Performed” Event Message (if enabled) to the computer. The computer can now send another command.

PCremote Only: (1) the computer will send the PCremote unit a command when the Vaux handshake line is true; (2) PCremote will respond by sending the computer a “Command Accepted” Informational Message (if enabled), and will set the handshake line false; (3) the command is transmitted from PCremote to the Vaux Receiver (this takes about 500 msec of RF transmit time); (3) PCremote sets the handshake line true at the end of transmission (the computer can now send another command, but it will get lost if sent too soon); (4) the main Vaux system processes the command (this can take up to a second for an X-10 transmission).

If Hardware Handshaking is not used when connecting the Vaux system to the computer (i.e. only the ground, TXD, and RXD lines are connected), there are two things to consider: the commands must get into the Vaux receive buffer without overflow, and a subsequent command cannot be sent until the current one has finished. The handshaking simply holds off the computer until a new command will be accepted by the Vaux Controller.

Only the characters actually used in the Vaux Protocol Commands count towards the receive buffer limit; space, tab, CR, LF, and all non-printing characters (except an ABORT character, which flushes the buffer) are ignored and are not entered in the buffer. The computer’s serial input routine must also have adequate buffering for the Vaux Protocol Messages.

With no handshaking, most Vaux Protocol Commands will be accepted without buffer overflow, since a small receive buffer exists in the Vaux system to handle short command lines. Also, by enabling Informational Messages, the computer essentially has a form of software handshake since it can simply wait for the arrival of the “Command Accepted” message which indicates that a new command may be sent. If these messages are not enabled, then the computer must ensure that adequate delays are inserted between successive commands to allow for command processing. For example, infrared code replay will vary widely depending on the code, but a delay of at least 500 msec is probably reasonable. X-10 powerline transmission will take almost 1 second.

In addition to these delays, an additional restriction applies to PCremote: after sending a command, there must be a delay of at least 500 msec — this time is required in order for the transmission to take place over the RF link, and during this time the VauxSerial Port is ignored. The computer must provide this delay for PCremote when Hardware Handshaking is not employed. However, even with handshaking, PCremote still has no way of knowing when the command has finished processing; the computer must provide a minimum of a second or so delay between commands.

Larger Vaux Protocol data movements (e.g.: the upload/download of macros) will generally require that handshaking be used.

To summarize, if no handshaking is used, provide at least the following minimum inter-command delays to account for command processing/transmit times:

Type of VauxSerial Interface	With Handshake	Without Handshake
Direct-connect (e.g. Aris)	none	1 second
RF-link (e.g. PCremote)	1 second	1.5 seconds

RS-232 General Information:

- RS-232 is a standard published by the Electronic Industries Association (EIA) for serial data communication. When it was introduced in 1962, the main use was the connection of data terminals to modems. The 1969 revision, RS-232-C, specified a cable length maximum of 15 meters. The 1986 revision, RS-232-D, adds compatibility with other standards such as CCITT V24, V.28, and ISO IS2110. RS-232-D's major changes include a mechanical specification of the interface connector, and loopback and test modes. RS-232-D does not specify a maximum cable length, but instead defines a maximum load capacitance of 2500 pF. A typical cable will exhibit perhaps 75-150 pF/meter; more for shielded cables. Since RS-232 is a single-ended system (return currents are flowing in the signal ground line, or worse yet, on the shield), it is much more susceptible to noise pickup (and generation) than a balanced system like RS-422 or RS-485. Twisting the TXD and RXD lines with individual signal ground wires (not all together unless you want crosstalk), and connecting the shield at only one end of the cable can help on long cable runs.
- A DB-25 F (female) connector is sometimes referred to as a DB-25 S (socket).
- A DB-25 M (male) connector is sometimes referred to as a DB-25 P (plug).
- Voltage levels on RS-232 connections (with respect to signal ground, pin 7):

TXD or RXD data lines:

Logical One (aka Mark)	² -3 V
Logical Zero (aka Space)	³ +3 V

Control lines:

Logical One (True)	³ +3 V
Logical Zero (False)	² -3 V

The control lines are sometimes referred to as logical inverse levels (i.e. RTS is RTS) which will conform to the voltage states of the data lines — same thing — take your pick (RTS is still true when it's ³ +3 V, in both definitions).

appendix 7: VauxSerial ASCII (receive) character set

This is the VauxSerial ascii receive map, used to filter the incoming character stream. All characters which get mapped to IGNORE are discarded. All characters which get mapped to BOGUS will result in an error. Valid characters will be BOGUS if found in an illegal position of the data stream. All control characters are mapped to IGNORE except ESCAPE (1Bh), Ctrl-C (03h), and Ctrl-X (18h), which are mapped to ABORT and are used to abort a command line and flush the receive buffer. % is also an ABORT char. SPACE is mapped to IGNORE. The numbers 0 to 9 get mapped to 00h to 09h respectively. The upper or lowercase letters A, B, C, D, E, and F get mapped to 0Ah to 0Fh respectively. Bit 8 of the incoming character should be 0, but is also cleared on input.

ASCII Character	Vaux RX	ASCII Char	Vaux RX	ASCII Char	Vaux RX
00h NUL (Ctrl-SP)	IGNORE	30h 0	00h	60h `	BOGUS
01h SOH (Ctrl-A)	IGNORE	31h 1	01h	61h a	0Ah
02h STX (Ctrl-B)	IGNORE	32h 2	02h	62h b	0Bh
03h ETX (Ctrl-C)	ABORT	33h 3	03h	63h c	0Ch
04h EOT (Ctrl-D)	IGNORE	34h 4	04h	64h d	0Dh
05h ENQ (Ctrl-E)	IGNORE	35h 5	05h	65h e	0Eh
06h ACK (Ctrl-F)	IGNORE	36h 6	06h	66h f	0Fh
07h BEL (Ctrl-G)	IGNORE	37h 7	07h	67h g	BOGUS
08h BS (Ctrl-H)	IGNORE	38h 8	08h	68h h	BOGUS
09h TAB (Ctrl-I)	IGNORE	39h 9	09h	69h i	BOGUS
0Ah LF (Ctrl-J)	IGNORE	3Ah :	BOGUS	6Ah j	BOGUS
0Bh VT (Ctrl-K)	IGNORE	3Bh ;	S_COLON	6Bh k	BOGUS
0Ch FF (Ctrl-L)	IGNORE	3Ch <	BOGUS	6Ch l	BOGUS
0Dh CR (Ctrl-M)	IGNORE	3Dh =	BOGUS	6Dh m	BOGUS
0Eh SO (Ctrl-N)	IGNORE	3Eh >	BOGUS	6Eh n	BOGUS
0Fh SI (Ctrl-O)	IGNORE	3Fh ?	BOGUS	6Fh o	BOGUS
10h DLE (Ctrl-P)	IGNORE	40h @	BOGUS	70h p	P_CHAR
11h DC1 (Ctrl-Q)	XON *	41h A	0Ah	71h q	BOGUS
12h DC2 (Ctrl-R)	IGNORE	42h B	0Bh	72h r	BOGUS
13h DC3 (Ctrl-S)	XOFF *	43h C	0Ch	73h s	BOGUS
14h DC4 (Ctrl-T)	IGNORE	44h D	0Dh	74h t	BOGUS
15h NAK (Ctrl-U)	IGNORE	45h E	0Eh	75h u	BOGUS
16h SYN (Ctrl-V)	IGNORE	46h F	0Fh	76h v	V_CHAR
17h ETB (Ctrl-W)	IGNORE	47h G	BOGUS	77h w	BOGUS
18h CAN (Ctrl-X)	ABORT	48h H	BOGUS	78h x	X_CHAR
19h EM (Ctrl-Y)	IGNORE	49h I	BOGUS	79h y	BOGUS
1Ah SUB (Ctrl-Z)	IGNORE	4Ah J	BOGUS	7Ah z	BOGUS
1Bh ESC (Ctrl-[])	ABORT	4Bh K	BOGUS	7Bh {	BOGUS
1Ch FS (Ctrl-])	IGNORE	4Ch L	BOGUS	7Ch	BOGUS
1Dh GS (Ctrl-^)	IGNORE	4Dh M	BOGUS	7Dh }	BOGUS
1Eh RS	IGNORE	4Eh N	BOGUS	7Eh ~	BOGUS
1Fh US (Ctrl-_)	IGNORE	4Fh O	BOGUS	7Fh DEL	IGNORE
20h SPACE	IGNORE	50h P	P_CHAR		
21h !	BOGUS	51h Q	BOGUS		
22h “	BOGUS	52h R	BOGUS		
23h #	BOGUS	53h S	BOGUS		
24h \$	BOGUS	54h T	BOGUS		
25h %	ABORT	55h U	BOGUS		
26h &	BOGUS	56h V	V_CHAR		
27h ‘	BOGUS	57h W	BOGUS		

28h (BOGUS	58h X	X_CHAR	
29h)	BOGUS	59h Y	BOGUS	
2Ah *	BOGUS	5Ah Z	BOGUS	
2Bh +	BOGUS	5Bh [BOGUS	
2Ch ,	BOGUS	5Ch \	BOGUS	* XON/XOFF used during
2Dh -	BOGUS	5Dh]	BOGUS	Software Handshaking, and
2Eh .	BOGUS	5Eh ^	BOGUS	ignored during Hardware
2Fh /	BOGUS	5Fh _	BOGUS	Handshaking.

appendix 8: obsoleted commands & messages

<u>Info Code</u>	<u>Meaning</u>	
95bcd	IR Repeat (from RF remote)	(removed ver 2.2)
96bcd	X10 Repeat (from RF remote)	(removed ver 2.2)

<u>Error Code</u>	<u>Meaning</u>	
40bcd	Invalid Clock/Calendar Program Cmd	(removed ver 2.4)

<u>Request Code</u>	<u>Purpose</u>	
40h	Get Clock/Calendar Data	(removed ver 2.4)
45h	Get (upload) Event Schedule (removed ver 2.4)	
47h	Get (upload) IR Device Memory (removed ver 2.3)	

<u>System Control Code</u>	<u>Purpose</u>	
10h	Clear Event Schedule	(removed ver 2.4)

Upload Clock/Cal:

VP FE 40 00 00 (cs) (removed ver 2.4)

Download Clock/Cal:

VPX FC hh mm ss dw mo dm yr ds (cs); (removed ver 2.4)

Clear Event Schedule:

VP F0 10 00 00 (cs) (removed ver 2.4)

Append entry to Event Schedule:

VPX FA 49 46 ... (cs); (removed ver 2.4)

Upload current Event Schedule:

VP FE 45 00 00 (cs) (removed ver 2.4)

Download a new Zone Map Table:

VPX 21 01 z0 za zb zc zd ze zf (cs); (removed ver 2.5)

Download a new Source Map Table:

VPX 21 02 s0 s1 s2 s3 ... s8 (cs); (removed ver 2.5)

Download a new Current Volume Table:

VPX 21 03 z0 za zb zc zd ze zf (cs); (removed ver 2.5)

Download a new Current Bass Table:

VPX 21 04 z0 za zb zc zd ze zf (cs); (removed ver 2.5)

Download a new Current Treble Table:

VPX 21 05 z0 za zb zc zd ze zf (cs); (removed ver 2.5)

appendix 9: revision history

Note that Vaux Protocol versions correspond with system software revisions. The Altrix, Aris, and MARC II Controllers use downloadable system software, and can be upgraded — contact the factory for pricing and availability of the latest software version, if newer features are desired in an older system.

Version 3.13:

- Added Shift-Enter to enter/exit special programming mode for prog IR ports, Zone-Group, System-Active-House-Code and X10-Input-House-Code
- Added Set Zone Turn-On Volume cmd (VP F0 59 zz 00)

Version 3.12:

- Allowed Zone-0 volume/mute/route across all zone groups in multi-controller installations
- Added Invalid SecID Event Message

Version 3.11:

- Initial release for Aris Model 710
- System Boot event message replaces info mssg
- Added Altrix VP Route command (VP 21 zz 01 ss)

Version 3.10:

- Added System-Active X-10 programmability

Version 3.00:

- Added Auto-TV-Powering using TV/Tape-2 device
- Added A/V Source parameter for TV powering use

Version 2.85:

- Aris: Routing a source will unmute a muted Aris zone.
- MACRO number in event mssg is now in hex (was bcd).

Version 2.82:

- Added 250 msec delay after each IR command in MACRO
- Power-Up delay (delay added after device power-up) now programmable.
- Power-Down delay (delay before device power-down) now programmable.

Version 2.81:

- Initial release for Altrix
- Altrix: Zone Status command added (VP FE 30 zz 00).
- MACROs 1 to 10 use static zoning (as before), MACROs 11 to 20 use dynamic zoning (dynamic-zone MACROs use RC-8's transmitted zone — needed for IR zoning in Altrix).
- Added Auto-Source powering by programming IR ON/OFF codes for each device.
- Altrix: Added Auto-Receiver (Tuner device) powering (use Tuner IR ON/OFF codes).
- Shift-ON/OFF from RC-8 used to test underlying IR source-device powering codes

Version 2.80:

- RC-8 delay definitions changed: 1=1 second, 2=2 seconds, 3=4 secs, 4=8 secs, 5=12 secs, 6=20 secs, 7=30 secs, 8=45 secs, 9=60 secs.
- Zone number added to mssg returned from Aris “Get Zone Status” cmd.
- Added repeats to Vaux Protocol IR playback command.
- MACRO learning now performs each step as entered (except delays).
- MACRO learning now stores repeat estimates in each step (VBT, IR, or X-10 repeats).

Version 2.79:

- Last MARC II release.

Version 2.78:

- Added support for Aris Zone-0 volume and mute.

Version 2.75:

- 9600 baud is now factory setting (was 2400 baud).
- Added ten new macros (11 to 20).
- Source-Change parameter added (MARC II/Aris).
- Added X-10 input mapping tables (Aris 720i).
- Protocol sub-version added.

Version 2.6:

- Delays can be added to macros via RC-8 (10-90 seconds), or via serial port (1-99 secs).
- Allow Remote-Control-Lockout (from programming macros or initiating IR learning).
- Added toggle mute/unmute VP command (Aris).
- Added Zone Group for addressing multiple Aris Controllers
- Added Auto-Source-Route mode (Aris).
- Allow zone 0 (all-zones) in Vaux Protocol route command.

Version 2.5:

- Initial release for Aris Model 720i.
- Enabled RC-8 commands for zone mute/flat/turn-on-limit (Aris).
- Event light flashes on VBT/Mute/Limit/Flat (Aris).
- Added Zone Turn-On/Off Default Table (Aris).
- Added Zone Status Request (Aris).
- Added Absolute Volume/Bass/Treble Control (Aris).
- Added Zone Muting and Flattening (Aris).
- Added Adjusting VBT Ramp Rate for Remote Controls (Aris).
- Removed obsolete Current Volume/Bass/Treble tables (Aris).
- Added variable baud-rate: 1200, 2400, 4800, 9600 (see System Control Command)

Version 2.4:

- Clock/Calendar system removed to make way for other features.
- Event Scheduling and Min/Second timers removed (required clock)
- Request Codes 40h and 45h removed (clock and event sched).

Version 2.3:

- Aris: Zone turn-on Volume fixed at -20 dB (was: previous value).
- Aris: Zone Volume ramps-on at fixed 20 ms-per-2dB-step rate (approx ¼ sec total).
- Aris: Source Map and Zone Map tables removed.
- Support added for eight infrared devices (previously supported six).
- Request Code 47h removed (full device upload not compatible with devices 7/8).

Version 2.2:

- Info Messages for IR Repeat (VI 95) and X10 Repeat (VI 96) removed; repeat count now added to Event Messages.
- Info Message for Invalid SecurityID Code added.
- IR download/upload commands added.

Version 2.1:

- Initial release for Aris Models 500/700 Audio/Video routing and control.
- MARC II and Aris: system code is downloadable.
- Software handshaking (XON/XOFF) added — is now the factory setting.
- Support added for auto-answer dial-in modem; remote site configs, etc.

Version 2.0:

- Initial release for MARC II Model 2000 (non-downloadable system code).
- Added Extended (variable-length) Commands and Messages.
- Added New Request Commands/Messages.
- Added System Control Command and low-level X-10 Command.
- Added Upload/Download of Clock/Calendar Data, MACROs, and Event Schedule.
- Added Device/Zone-specific infrared outputs (see configuration).

Versions 1.0 to 1.8:

- Releases for MarcPCI and PCremote.

appendix 10: zone- v.s. device-specific IR outputs

The IR Emitter connectors (3.5 mm phone jacks), on the rear panel of the Altrix, Aris, and MARC II Controllers, can be programmed for device-specific or zone-specific operation. The factory setting of the emitter ports is for device-specific operation. See the Configuration section for information on selecting which mode is in use.

Device-Specific Operation: Device-specific operation allows you to use identical source components in your system, for example, identical CD jukeboxes or DSS tuners. Emitter port 1 corresponds to the Tuner device, 2 corresponds to CD, 3 corresponds to Cab/Sat, 4 corresponds to VCR-1, 5 corresponds to VCR-2, 6 corresponds to Laser, 7 corresponds to Tape-1, and port 8 corresponds to TV/Tape-2. For an Aris controller, which only has six IR ports, the Tape-1 and Tape-2 device IR output is only available at the common IR Blaster port.

Example

If you have two identical VCRs, you could use the VCR-1 and VCR-2 device pages for them. When learning IR, you would program the same IR codes in both the VCR-1 and VCR-2 device pages (e.g. Play, FF... are each programmed in two locations). The VCRs are then accessed individually based on selection of the VCR-1 or VCR-2 source-device, since the emitters are activated accordingly.

Zone-Specific Operation: The user-specified Zone information is used to select the desired IR emitter output. All Vaux remote controls have a Zone code which gets transmitted with every button pressed. A default Zone can be programmed, or a temporary zone may be explicitly selected.

Example

Using an Altrix Controller, you could design a simple 3-zone audio system by using 3 identical audio receivers and running the speaker wires of each receiver to different areas of the house (up to 8 zones could be implemented with the Altrix by adding a receiver and IR emitter for each zone). The source components' audio (CD, Cassette, DAT, etc...) will connect in parallel to the inputs of all audio receivers.

When learning IR codes for the Altrix System, store the IR commands for the audio receivers (recall that they are identical) in the Tuner device page. Connect the infrared emitters for the receivers to the desired IR Emitter ports on the Controller. Let's say that one audio receiver drives the living room and is connected to IR Emitter port (zone) 1, and another receiver drives the master bedroom and is connected to IR Emitter port (zone) 5. Common equipment (CD...) must connect to the IR Blaster port using a single or dual emitter, or a multi-emitter connecting block, as necessary.

The Vaux Remote Control in each room will have its default Zone set to the appropriate zone number for the room — in this case, 01 will be programmed into the remote control for the living room, and 05 for the master bedroom remote. The leading 0 is the Zone-Group for this typical single-controller installation. Now, upon selecting a source for the living room (Zone 01) by pressing a key such as CD, you will be controlling the receiver connected to IR port 1 (for commands such as volume...). The common source device (CD) is controllable independently of the zone, since it is connected to the common IR Blaster port. Other zones are controlled identically, although they are sharing control of the common source equipment.

appendix 11: updating controller software

Vaux Altrix, Aris, and MARC II Controllers may have their system software updated to take advantage of features in newer software releases. Some of the system is hard-coded in firmware, but the bulk of the system's features and operation are controlled by the downloadable portion of the system software.

Vaux Authorized Dealers may order a software update disk directly from Vaux; the provided disk is in 3.5", DSHD (or DSDD) IBM format. VauxConfig software for Windows may be used to load new system software into a Vaux Controller, but you may also use a computer with a terminal emulation program having ASCII transfer capability. You will also need an appropriate serial cable. Windows machines, some DOS machines, and some Macintosh computers can read the disk file and provide the downloading necessary.

This example uses the "Terminal" emulator provided with Windows 3.1:

(Windows 95 includes a more confusing and needlessly complicated application called Hyper-Terminal)

Note: Ensure that you have the correct serial cable available. Also, you must have an unused serial port available on your computer. If you are using a laptop, you may only have one serial port — enable the internal trackball if you have a mouse on your laptop's only available serial port.

- 1) Start the Terminal application on the Windows PC.
- 2) From the Settings/Communications menu, select:
Baud: 9600 (typical; earlier controllers used 2400 baud default)

(note: baud may be set to 1200, 2400, 4800 or 9600)

Data Bits:	8
Stop Bits:	1
Parity:	None
Flow Control:	Xon/Xoff
Connector:	COM2 (or whichever port you are using)

- 3) Connect the appropriate serial cable between the Computer and the Vaux Controller. Most modem cables will work fine, but not all of them are wired correctly. If your computer requires active handshake input lines to enable the port, the cable must connect these lines or the port will be permanently "waiting."
- 4) Test the serial connection:
Press 'v' (upper or lower-case), and then press Escape — you should see "VI 81" displayed in your terminal window. The VI 81 message (command aborted) is in response to the ESC character. You now know that you are properly connected to the Vaux Controller.

Note: The 'V' is necessary since the controller will ignore all characters until the first V is found. This is a result of the controller being configured for an auto-answer modem connection, to prevent error messages on normal modem strings such as "RING" and "CONNECT."

Note: The system's operating baud rate may be set (as of Version 2.5) to 1200, 2400, 4800, or 9600 baud. If you get returned characters in response to sending an Escape character, but the displayed characters are incorrect, it is likely that the baud rate is incorrect — try another. If you do not get any response to sending an Escape character, you do not have a proper connection — check your serial port and cable.

- 5) Turn OFF the power switch on the Vaux Controller.

Note: Continuing past this point will erase the system software currently in the Controller; a new system must then be downloaded. Ensure that you have the appropriate update disk and that your serial connection is working.

- 6) While pressing and holding the Controller's rear-panel Reset button, turn ON the power switch; this erases the system software currently in the Controller. Front panel lights should be flashing indicating that system software is not

installed, and that the Controller is awaiting a download — depending on the ROM chip installed in the Controller, the front panel lights will indicate the download baud rate:

- Green SERIAL/PCI and red ERROR lights both flashing: download baud rate = 9600.
- Red ERROR light only is flashing: download baud rate = 2400 (earlier Controllers)

7) From the Settings/Communications menu, select:

Baud: 9600 (or 2400, as determined in step 6 above)

Note: The download baud rate is unrelated to the operating baud rate used by the Controller after a system has been downloaded. Newer Vaux Controller ROMs use 9600 baud merely to speed up downloads.

Note: While the system's operating baud rate may be set (as of Version 2.5) to 1200, 2400, 4800, or 9600 baud, for downloading system software, the rate must always be set to 9600 or 2400 baud depending on the Controller ROM installed.

8) Insert the Vaux system disk into the floppy drive.

9) From the Transfers menu, select "Send Text File..." and find the appropriate system software file to download:

<u>Controller</u>	<u>Filename</u>
Aris Model 700:	A700.HEX
Aris Model 720i:	A720I.HEX
Altrix Model 3000:	ALT3000.HEX
Altrix Model 3400:	ALT3400.HEX
MARC II Model 2000:	M2000.HEX

e.g.: Change to the floppy drive, change the File Name to *.hex, and browse folders if necessary, select the appropriate filename, and click OK.

10) As the file is being transferred, the Controller's SERIAL/PCI light will flash. If any errors occur during download (e.g.: incorrect baud rate, or wrong file transferred), the ERROR light will begin to flash again — retry the transfer.

11) When the download is complete (a few minutes later), the Controller will display the "light bar" indication seen at power-up, and is ready for use.

12) Set up the system as usual.

Note: The version of system software you just downloaded determines the operating baud rate at this point: version 2.75 and later use 9600 baud, while earlier versions used 2400 baud.

appendix 12: glossary

ABORT characters	% (25h), ESCAPE (1Bh), Control-C (03h), and Control-X (18h). Any one of these four characters are recognized by the VauxSerial Interface as a Command ABORT — this is handy for manual testing of VauxSerial with an ASCII terminal. If Informational Messages are enabled, a “Command Aborted” message will be returned after the ABORT is received. The ABORT characters also flush the VauxSerial input buffer.
Application Program	A software program, running on a host computer, which is communicating with the Vaux system using Vaux Protocol™.
ASCII	Acronym for American Standard Code for Information Interchange. A set of 128 7-bit characters from 0 (00h, the NULL char) to 127 (7Fh, the DEL char). The most widely accepted standard for defining text and control codes.
ASCII Terminal	See Terminal.
Asynchronous	A form of serial data transmission whereby the clock timing information is not explicitly sent, but is extracted from the data signal. An RS-232 connection, for example, will use the position of the start bit to synchronize the clocking of the received data bits which follow.
Baud	A transmission rate having units of bits-per-second (bps). Usually refers to the speed of serial data communications links.
Binary Number System	The base-2 number system, using the set of symbols ‘0’ and ‘1’. In Vaux documents, binary numbers have a ‘b’ suffix (e.g. 10010001b).
Bit	A contraction of binary digit. The smallest unit of information in a base-2 (binary) number system as used in digital computers. Has only two states: True (1) or False (0).
Buffer	A temporary memory storage area. Often used to allow one device or program to store information at one rate, and allow a second device or program to read the information at different (usually slower) rate, as is done in serial communications receive buffers.
Byte	A unit of information consisting of eight bits and having a range from 0 (00h) to 255 (FFh).
Character	(1) A symbol having widely-understood meaning such as letters, numbers, and punctuation. (2) A member of the ASCII set of 7-bit text and control characters. (3) A byte-sized (8-bit) quantity used in programming languages (such as C); the eighth bit may or may not have meaning. (4) John Cleese.

Checksum	A mechanism used to ensure that a group of data bytes was transmitted without error. The transmitting system calculates a sum of the data bytes, and then appends this sum to the tail end of the transmitted data; the receiving system also calculates a sum on the received data bytes, and then compares it to the transmitted checksum to determine if there was a transmission error. The receiver may then simply ignore the bad data block, or it may have a means of requesting retransmission. See also Parity.
Computer Serial Port	Refers to an RS-232-compatible asynchronous serial interface on a computer such as an IBM PC®, Macintosh®, or Unix™ system.
Configure	To change settings of software or hardware systems in order to change operational characteristics (e.g. to configure a serial port's parameters for speed, parity, etc., or configure a Vaux parameter, such as enabling checksums).
Control Character	A non-printing ASCII character (from 00h to 1Fh, and 7Fh) for controlling the way information is transferred or displayed. These characters are ignored by Vaux systems, except for five — see ABORT and XON/XOFF.
Control Command	In Vaux Protocol™, is initiated by the controlling computer and generally requests that some sort of action be taken by the control system; for example, to press Play on a VCR (send an infrared control code), or to Turn On Porch Light (via an X-10® module). Some commands request a reply message and do not result in an action.
Control-C	An ASCII control character (03h) used by Vaux systems to abort a command stream.
Control-X	An ASCII control character (18h) used by Vaux systems to abort a command stream.
CTS	The RS-232 Clear-To-Send signal from a DCE (e.g. modem or VauxSerial) indicating that the DCE can accept characters from the DTE.
DB-9	A 9-pin physical connector design used for RS-232 serial port connections on some IBM-compatible computers.
DB-25	A 25-pin physical connector design used for RS-232 serial port connections.
DCD	The RS-232 Data Carrier Detect signal from a DCE (e.g. modem or VauxSerial) indicating that the DCE is ready to communicate. Specifically, for a modem it indicates that there is a connection established to another modem.
DCE	RS-232-connected Data Communications Equipment is typically a modem (it receives data on pin 2, and transmits data on pin 3 of its DB-25 connector). It is the slave end of an RS-232 connection. VauxSerial interfaces use DCE-configured ports.
Decimal Number System	The base-10 number system, using the set of symbols '0' through '9'. In Vaux documents, decimal numbers have no suffix (e.g. 254), or a 'd' suffix (e.g. 254d).
DIN-4 (4-pin Mini-DIN)	A 4-pin physical connector design used for the VauxNet interconnection. Also used for S-Video.

DIN-8 (8-pin Mini-DIN)	An 8-pin physical connector design used for RS-232 serial port connections on most Macintosh computers.
Download	Defined as data movement from the host computer to a Vaux system (see also Upload).
DSR	The RS-232 Data Set Ready signal from a DCE (e.g. modem or VauxSerial) indicating that the DCE is ready to communicate.
DTE	RS-232-connected Data Terminal Equipment is typically a computer (it transmits data on pin 2, and receives data on pin 3 of its DB-25 connector). It is the master end of an RS-232 connection.
DTR	The RS-232 Data Terminal Ready signal from a DTE (e.g. computer) indicating that the DTE is ready to communicate.
Error Messages	Vaux Protocol Messages which provide error information are sent from the VauxSerial Port to the computer to relate critical events (e.g. “Bad Checksum”). These messages may be disabled by the appropriate configuration flag.
Escape Character	An ASCII control character (1Bh) used by Vaux systems to abort a command stream.
Event	The occurrence of something of significance. An event may be the reaching of a specific time, or a button press on a remote control.
Event Message	In Vaux Protocol™, is sent by the Vaux system to the controlling computer; for example, to announce that a specific Vaux Remote Control button was pressed. These messages may be disabled by the appropriate configuration flag.
Field	A data item separated from other data by either its position within the data (implied delimiting), or by a delimiter such as a space, tab, comma, or other character.
Flow Control	See Handshaking.
Full duplex	A communications link over which both sides may transmit simultaneously (e.g. a telephone). In serial data communications, the receiving end will often echo characters back to the transmitting side.
Half duplex	A communications link over which only one side may transmit at a time (e.g. a push-to-talk walkie-talkie). In serial data communications, the transmitting end will often echo transmitted characters locally for screen display.
Handshaking	Also called flow control, is used between a DTE and a DCE to provide a mechanism by which the slower side may halt communications briefly while it attends to another task. See also Hardware Handshaking, and Software Handshaking.
Hardware Handshaking	Uses additional RS-232 communication lines to provide “OK-to-Send” signals between the two sides. See also Handshaking, and Software Handshaking.
Hexadecimal Number System	The base-16 number system, using the set of symbols ‘0’ through ‘9’, and ‘A’ through ‘F’ (or ‘a’ through ‘f’). In Vaux documents, hex numbers have an ‘h’ suffix (e.g. 2C7Fh).

HVAC	Heating, Ventilation, and Air Conditioning, or, Heating, Ventilation, And Cooling.
Icon	An image that graphically represents an object or action. Icons are used for pushbutton identification on computer screens, Remote Controls, and Consoles.
Informational Message	Vaux Protocol Messages which provide general information are sent from the Vaux system to the computer to relate non-critical events to the computer (e.g. "Command Accepted"). These messages may be disabled by the appropriate configuration flag.
Infrared	A region of the light spectrum just beyond the red end of visible light. Modulated infrared light beams are used in most remote control devices found in the consumer world. Limitations include line-of-sight operation, maximum distance of about 20 feet, and sunlight swamping which generally precludes outdoor operation. Although the human eye cannot see infrared, most camcorders can "see" the pulsed infrared light; handy for testing!
Installation	The process of physically locating, cabling, configuring, and teaching infrared codes to a Vaux system.
Interface	The devices, rules, or conventions by which one system connects to, and communicates with, another system. Interface often refers to only the physical and electrical characteristics of the connection, and a communications protocol generally specifies the temporal aspects of data movement.
Learning	In Vaux systems, infrared remote control codes are learned by the system during the installation process, and stored into non-volatile memory for later playback.
MarcPCI	A Vaux Expansion Module (now discontinued) which plugged into the VauxNet Expansion Port of early MARC Controllers. It allowed an RS-232-connected host computer to control the MARC system.
Memory	A computer's data storage area. See also Non-Volatile.
Modem	A contraction of modulator/demodulator. A transceiver which converts between one signal domain and another. Commonly used to refer to a device which converts between RS-232-level digital signals and one or more transmission schemes utilizing voice-grade (300-3000 Hz) telephone audio channels. The RS-232 port on a modem is configured as a DCE (it receives data on pin 2, and transmits data on pin 3 of its DB-25 connector). It is the slave end of an RS-232 connection. VauxSerial interfaces use modem-like DCE-configured ports.
Network	Usually refers to an interface/protocol for connecting multiple computers or other components together. Vaux has a short-distance network, VauxNet™, for interconnecting its control components.
Nibble	A unit of information consisting of four bits and having a range from 0 (0h) to 15 (Fh). Usually refers to the upper or lower half of a byte.
Non-Volatile	Indicates that data is permanently stored and is not subject to deletion upon power interruption. Vaux stores learned infrared codes and system parameters in non-volatile memory.
Octal Number System	The base-8 number system, using the set of symbols '0' through '7'. In Vaux documents, it's unlikely that octal numbers will be used.

Parity	A scheme for giving a modest level of data transmission error checking. A parity bit is calculated from the number of 1s which occur in a 7-bit character (or 8-bit byte); this parity bit is transmitted as the 8 th (or 9 th) bit of the word to allow the receiving system to determine if there may have been a transmission error. The parity can be defined to be either odd or even depending on whether an odd number or even number of 1 bits is tested for. Note that two dropped 1s during transmission will NOT appear as an error to the receiving system. Vaux systems do not use parity checking. See also Checksum.
PCremote	A Vaux accessory which allows an RS-232-connected host computer to control the Vaux system. This is an RF transmitter system providing one-way control from the host computer to the Vaux system; the computer may be located remotely from the Vaux system when using this product.
Protocol	A set of rules that computers must follow to exchange data over a communications link.
Radio-Frequency	Vaux Remote Controls and Consoles use radio-frequency (RF) signals to send control commands to a Vaux Receiver from anywhere in the building (or outside). Range of an RF transmission is dependent upon the surroundings and construction materials; Vaux system range is typically 125 feet.
Receive Buffer	See Buffer.
RS-232	A standard published by the Electronic Industries Association (EIA) for serial data communication. When it was introduced in 1962, the main use was the connection of data terminals to modems (see DTE and DCE). The 1969 revision, RS-232-C, specified a cable length maximum of 15 meters. The 1986 revision, RS-232-D, adds compatibility with other standards such as CCITT V24, V.28, and ISO IS2110. RS-232-D's major changes include a mechanical specification of the interface connector, and loopback and test modes. RS-232-D does not specify a maximum cable length, but instead defines a maximum load capacitance of 2500 pF. A typical cable will exhibit perhaps 75-150 pF/meter; more for shielded cables. Since RS-232 is a single-ended system (return currents are flowing in the signal ground line, or worse yet, on the shield), it is much more susceptible to noise pickup (and generation) than a balanced system like RS-422 or RS-485.

RTS	The RS-232 Request-To-Send signal from a DTE (e.g. computer) indicating that the DTE has data to transmit to the DCE. Can also be defined to be a CTS-like handshake signal to control modem-to-computer data flow.
RTS/CTS Handshaking	1) One-way: Controls data flowing from the computer to the modem. 2) Two-way: the RTS line from the DTE is used like a DCE's CTS line to control signal flow in the other direction.
RXD	The RS-232 Receive Data signal from a DCE (e.g. a modem, or VauxSerial) to a DTE (e.g. computer).
Serial Cables	RS-232 interconnect cables. Vaux supplies cables for connecting the VauxSerial Port (DB-25) to IBM PC®-compatible (DB-25 or DB-9) ports, or Macintosh® (DIN-8) ports.
Serial Port	See Computer Serial Port.
Software Handshaking	A flow control technique using, for example, the XON (Ctrl-Q) and XOFF (Ctrl-S) characters. See also Handshaking, and Hardware Handshaking.
Request Messages	Vaux Protocol Messages which provide answer requests are sent from the Vaux Controller to the computer to return a variety of information requested by the computer. These messages are always enabled.
Terminal	Refers to a simple, asynchronous, ASCII, RS-232-connected, video display terminal (or terminal emulator running on a PC). May be used for testing or controlling the Vaux system manually using Vaux Protocol commands.
TXD	The RS-232 Transmit Data signal from a DTE (e.g. a computer) to a DCE (e.g. modem, or VauxSerial).
Upload	Defined as data movement from the Vaux system to a host computer (see also Download).
Vaux Protocol	A Vaux-defined, public-domain command language for interacting with Vaux's control systems, by an RS-232-connected host computer.
Vaux Protocol Commands	Commands output from the host computer to control the Vaux system (Configuration, Request, Control...).
Vaux Protocol Messages	Messages returned from the Vaux system to the host computer (Error, Event, Informational, Request...).
VauxSerial Port	Refers to the DCE-configured RS-232 serial port on Vaux Controllers. This is a DB-25 F (female) connector. Formerly called a PCI Serial Port.
X-10®	A defacto standard for AC powerline control of remotely-located lighting and appliance control modules. Vaux control systems provide an interface to the powerline and send the appropriate X-10 commands (over the powerline) to control these modules. Appliance modules are On/Off only, whereas Lamp modules provide for dimming of incandescent lights as well.
XON/XOFF	See Software Handshaking.