



Digital Binloop and Video Binloop

User's Manual

Document Revision 3.1

August 24, 2009

Copyright © 1996-2009 Alcorn McBride, Inc. All rights reserved.

Every effort has been made to assure the accuracy of the information contained in this manual, and the reliability of the Alcorn McBride Digital Binloop hardware and software. Errors can sometimes go undetected, however. If you find one, please bring it to our attention so that we can correct it for others. Alcorn McBride welcomes comments and suggestions on the content and layout of its documentation.

Applications described herein are for illustrative purposes only. Alcorn McBride Inc. assumes no responsibility or liability for the use of these products, and makes no representation or warranty that the use of these products for specific applications will be suitable without further testing or modification. Alcorn McBride products are not intended for use in applications where a malfunction can reasonably be expected to result in personal injury. Customers using or selling Alcorn McBride products for use in such applications do so at their own risk, and agree to fully indemnify Alcorn McBride for any damages resulting from such improper use or sale. Alcorn McBride Inc. reserves the right to make changes to these products, without notice, in order to improve their design or performance.

Digital Binloop™ and Video Binloop™ are trademarks of Alcorn McBride Inc., all rights reserved.

Dolby is a trademark of Dolby Laboratories.

CobraNet is a trademark of Cirrus Logic.

Hardware Design:	Jim Carstensen, Scott Harkless, and Joy Burke
Firmware Design:	Jim Carstensen, Scott Harkless
Software Design:	Adam Rosenberg
Documentation:	Jim Carstensen, Scott Harkless, John Conley and Adam Rosenberg
Mechanical Design:	Martin Chaney

Alcorn
McBride
Inc.

Alcorn McBride Inc.
3300 S. Hiawassee, Bldg. 105
Orlando, Florida 32835
TEL: (407) 296-5800
FAX: (407) 296-5801
<http://www.alcorn.com>
info@alcorn.com

Contents

WELCOME	1
Organization of this Manual	1
Technical Support	1
GETTING STARTED.....	3
Installation.....	3
Unpacking Instructions	3
Installation Instructions	3
Playing a Test File	4
Preparation	4
Formatting the Compact Flash	5
Loading the Test File	5
Playing the Test File	6
ABOUT THE BINLOOP	7
What is the Binloop?.....	7
What's inside the Binloop?	7
The Card Cage	8
The Controller Card	8
The Reproducer Cards	8
The CobraNet Card	9
How the Binloop Works	10
Control System	11
A/V Sync System	12
SMPTE Interface	13
CONTROLLING THE BINLOOP	17
Serial Control (RS-232)	17
MIDI Control	18
Ethernet Control	18
Parallel Control	19
Parallel Inputs	19
Parallel Outputs	20
Using the Group Control Inputs	20
Using the Start SMPTE Input	21
Using the Mute Input.....	21
Timecode Control	22

CONTROL PROTOCOLS	23
AMI/Pioneer Control Protocol	23
Get Firmware Version	23
Get SMPTE Firmware Version	23
Get/Set Unit ID	24
Get/Set IP address	24
Get/Set Subnet Mask number	25
Get/Set Gateway address.....	25
Enable SMPTE	26
Disable SMPTE.....	26
Pause SMPTE	26
Idle SMPTE.....	26
Get/Set SMPTE time	27
Get/Set SMPTE Mode	27
Get/Set SMPTE Frame Rate Mode	28
Get/Set SMPTE Preroll time	28
Get/Set SMPTE Start time	29
Get/Set SMPTE End time	29
Get/Set SMPTE Loop Mode	30
Get/Set SMPTE Powerup Mode	30
Get/Set SMPTE Restart Mode.....	31
Get/Set SMPTE Idle Mode	31
Get/Set Video Format	32
Get/Set Synchronous Play Delay	32
Search Clip	33
Play / Resume.....	33
Loop / Resume.....	33
Play Clip.....	34
Play/Loop Clip.....	34
Synchronously Play Clip	35
Synchronously Play/Loop Clip	35
Play Next.....	36
Loop Next.....	36
Stop Playback	37
Still Playback.....	37
Pause Playback	37
Mute/Unmute Video	38
Mute/Unmute Audio	38
Play Independent (Audio Only).....	39
Loop Independent (Audio Only).....	39
Stop Independent (Audio Only)	40
Get/Set Video Colorspace	40
Get Reproducer Firmware Version	41
Get Reproducer Status	41
Get Reproducer Filename	42
MIDI Show Control Protocol	43
General Structure	43
Message Format.....	44
Message Examples	45
Acknowledge Message Format	46
BINLOOP CONFIGURATION SOFTWARE	47

Introduction to the Binloop Software	47
Requirements.....	47
Installing the Binloop Software.....	47
Connecting your PC to the Binloop.....	48
Connecting with RS-232 Serial.....	48
Connecting with Ethernet	48
Running Binloop Software	49
Controlling the Binloop with the Binloop Software	51
Navigating through the Menus.....	52
File Menu	52
Binloop Menu.....	54
View Menu	61
Window Menu	61
Help Menu	62
Timecode Trigger List.....	63
A Word About Trigger Latency:	66
COBRANET	67
Description	67
Details.....	67
CobraNet "Y" Cable	67
Reproducer Configuration	68
CobraNet Configuration.....	69
CobraNet Advanced Configuration.....	70
Bundle.....	71
HARDWARE DESCRIPTION.....	73
Controls and Indicators	73
Unit Status LED's.....	73
Control Buttons	74
Display and Menu Wheel.....	75
Reproducer Status LED	76
Reproducer Test Button.....	76
Connectors.....	77
Connector Summary.....	77
Digital Audio (AES/EBU).....	79
Balanced Audio.....	79
Unbalanced Audio and Composite Video.....	80
Analog Component Video.....	80
Show Control Port.....	80
Programmer Port	81
Ethernet	81
MIDI In	81

MIDI Out.....	82
V-Sync	82
SMPTE.....	82
Parallel Control	83
MEDIA FILES	85
File Types	85
File Naming	86
Video Formats.....	87
MPEG-2 Video Specifications.....	87
Audio Formats	88
MPEG2 Program Stream Audio Formats	88
Uncompressed Audio Formats	89
File System Limitations – 2GB File Size.....	90
INSTALLATION GUIDELINES	91
Grounding and System Performance	91
A Word on Temperature.....	92
Output Connections	93
Compact Flash Media Reliability.....	93
Removing a Card from the Cage.....	93
SPECIFICATIONS.....	95
Physical	95
Control Information	95
Video	95
Audio.....	96
SMPTE.....	96
APPLICATION NOTES	97
Note 1: Automatically Play on Power-Up	97
Preparation	97
Connecting to the Binloop.....	97
Modifying the Binloop Configuration	98
Downloading the New Configuration	101
Note 2: Controlling the Binloop with an Alcorn McBride Show Controller	102

Preparation	102
Configuring the Binloop	103
Programming the Show Controller	103
Testing the Show Control Script	108
Note 3: Connecting to the Binloop using Ethernet	109
Preparation	109
Changing the Network Settings of your PC	110
Configuring the Network Settings of the Binloop	111
Connecting over Ethernet with the Binloop Software	112
APPENDIX A - TROUBLESHOOTING GUIDES	115
APPENDIX B - COMPACT FLASH MEDIA CAPACITIES	117
APPENDIX C - UPDATING FIRMWARE	119
Reproducer Card Firmware	119
Controller Card Firmware	120
APPENDIX D – MENU WHEEL MAP	123
INDEX	125

Welcome

Organization of this Manual

Before you use the Binloop, it is suggested that you read this instruction manual thoroughly. The Getting Started section contains some simple operational examples and explains how to quickly get up and running with the Binloop hardware. Later sections contain more detailed information on programming and operation, including how to integrate your Binloop into a control system and the various hardware configuration options that you'll need to set up.

Technical Support

Should questions or issues come up, which are not covered in this User's Manual, please do not hesitate to contact us. We are here to help!

You can contact us:

On The Web:	http://www.alcorn.com
By E-mail:	support@alcorn.com
By Phone:	(407) 296-5800 (M-F 9-6 EST)
By Fax:	(407) 296-5801

Getting Started

In this section, you will learn how to:

- Install the Binloop
- Play a test clip

Installation

These instructions assume that you have received your Binloop from Alcorn McBride with the Controller card and the appropriate reproducers already installed. For detailed instructions and information on installing the Binloop see the **Installation Guidelines** section later in this manual.

Unpacking Instructions

When you unpack this product you may want to consider holding on to the original packing material for shipment to the installation site, or in case you need to return it to the factory for repair or exchange.
PLEASE DON'T THROW AWAY THIS MANUAL!

Installation Instructions

The Binloop is designed to be installed in an equipment rack. It is recommended that you leave an empty space (1U) above and below the unit for free-air circulation, and install it in a climate-controlled room. **Failure to do so could result in damage or intermittent operation of the Binloop.**

Note: *It is always ideal to use power protection such as surge protection and/or an uninterruptible power supply to ensure a reliable power source for the Binloop.*

1. Place the Binloop in the appropriate position on the rack.
2. Insert screws into the rack-mounting holes on both sides of the unit and tighten.
3. Connect the power plug to an appropriate outlet.
4. The Binloop will operate with 100-240VAC, 6-3A Max. 60-50 Hz.
5. Connect the audio/video cables from the Reproducer outputs to the A/V equipment (i.e. displays, amplifiers, etc.)

Playing a Test File

Now it's time to put that nice shiny new Binloop to work and make it play some media. The Binloop is commonly used to either play video or audio, so this section provides information on playing both types of media. Be sure to follow the instructions that are more suitable for your Binloop application.

Preparation

Before you can play media on the Binloop, you need to load the audio or video content onto a Compact Flash card. To do this, you'll need the following items:

1. **Compact Flash card**
2. **Compact Flash card reader**
3. **Test Media File (Video or Audio)**

Compact Flash Card

If you did not purchase Compact Flash cards with your Binloop, we highly recommend you purchase **SanDisk®** brand cards. There are many different Compact Flash devices out there with different specs and compatibility issues. We recommend SanDisk® cards because we can guarantee that they work in all of our Compact Flash products. Also, it's important to mention that Compact Flash cards come in a variety of capacities. To help you select a size that suits your needs, be sure to read over **Appendix B: Compact Flash Media Capacities**.

Compact Flash Card Reader

The Compact Flash card reader is required to interface the Compact Flash device with your PC. The most common card reader connects to your PC using USB. These are widely available in consumer electronics stores, and can also be purchased from Alcorn McBride.

Binloop Test Files

The Binloop test files can be downloaded from the product support pages on our website. Please download either the Audio or Video test clip. Here are the support page links:

Video Binloop

<http://www.alcorn.com/products/videobinloop/support.html>

Digital Binloop

<http://www.alcorn.com/products/binloop/support.html>

Formatting the Compact Flash

All Compact Flash Sockets have ejector buttons that make it easier to insert and remove the flash media.

When you insert a flash card, do so with the manufacturer's label to the right. The card should insert easily and require very little force to fully seat.

Once you have the necessary materials, you are ready to get started. First, it is recommended that you format the Compact Flash card in the Binloop. This can be accomplished by following these instructions:

With the Compact Flash card removed, power up the Binloop using the power switch on the back of the unit.

Press and hold the Test Button of channel #1. You may see the Reproducer indicate a fault (flashing red LED), but this is normal.

While holding the Test Button, insert the Compact Flash card into the Compact Flash socket of channel #1. Wait a second or two for the status LED to turn orange to indicate that the card is being formatted.

When the status LED turns off, the formatting is complete. You may now remove the card.

Loading the Test File

Now that the Compact Flash has been properly formatted, we're ready to load the Binloop Test file onto the card. Here's what you do:

1. Insert the Compact Flash card into the Compact Flash card reader.
2. Connect the card reader to your PC. If the card reader is working properly, the Compact Flash card will show up as an additional drive (i.e. Drive F:) in your computer.
3. Copy and Paste the Binloop Video or Audio Test File to the flash card.
4. If you are loading the test video clip, verify that the name of the test clip is **VID00001.MPG**. If you are loading the test audio clip, verify that the name of the test clip is **SND00001.WAV**.

Playing the Test File

All of your hard work is about to pay off. Follow these step-by-step instructions to play the test file in the Binloop.

1. If you are playing the video test clip, connect a display to one of the A/V outputs of channel #1 (YPbPr, RGB, or Composite).
2. Insert the Flash card that is loaded with the Binloop audio or video test clip into the Compact Flash socket of channel #1.
3. Power-up the Binloop.
4. Wait for the Binloop to finish initializing. You can determine this by watching status display and the 'READY' status LED.
5. Press the channel #1 test button. The channel #1 status LED should turn green to indicate that the clip is playing.

At this point, channel #1 should be outputting video and/or audio on the rear-panel connectors, and this should be apparent on your A/V equipment. If it is not, please consult the **Troubleshooting Guide** for help.

About the Binloop

What is the Binloop?

Put simply, the Binloop is a solid-state multi-channel video and audio player that provides up to 16 separate channels of standard-definition video or 32 channels of uncompressed audio. These channels can be played together synchronously or independently from one another. For stand-alone applications, the Binloop can be configured to play files automatically based on a programmable timeline. For more complex applications, this product can be controlled externally by SMPTE/EBU timecode, MIDI, RS-232, Ethernet or contact closures.

That being said, the Binloop is not your everyday multi-channel media player. Its ability to provide synchronous playback, impeccable precision, and a variety of easy-to-use control interfaces requires unique system architecture. The purpose of this section is to familiarize you with how the Binloop works so that you can gain a better understanding of how best to use it to meet your goals.

What's inside the Binloop?

The Alcorn McBride Binloop consists of four main types of components –

1. **Card Cage**
2. **Controller Card**
3. **Reproducer Card(s)**
4. **CobraNet Card (optional)**

The Card Cage

The Reproducer slots and CobraNet slot may or may not be populated depending on how your Binloop was ordered.

The Card Cage is the 'skeleton' that integrates the components of the Binloop. This cage contains a Power Supply, a status display, a Controller card slot, and 16 Reproducer card slots. The Reproducer card slots are numbered from 1 to 16, from left to right as viewed from the front of the Binloop.

The back of the Card Cage contains all of the connectors for the Binloop control interfaces, as well as Audio/Video outputs for all 16 Reproducer card slots.

The Controller Card

The Controller Card is responsible for the core functionality of the Binloop. This card updates the Status Display with information and provides front-panel buttons for controlling basic Binloop functions. It also handles all external communication from the remote control interfaces of the Binloop, as well as internal communication with the Reproducer cards. In addition, it provides the Binloop with the ability to either Generate or Read SMPTE timecode. It also has the ability to store a list of SMPTE triggers that can be precisely executed at specified times. To maintain synchronization between Binloop components, this card also supplies all audio and video sync signals to the Reproducer cards.

The Reproducer Cards

The main function of each Reproducer is to play video files from a removable Compact Flash card. During playback, the video and audio output generated by the reproducer is available on the connectors found on the back of the Binloop cage. Each reproducer has output connectors for YPbPr/RGB Component Video, Unbalanced Audio, Balanced Audio, and Digital Audio (AES/EBU or S/PDIF).

The reproducer cards receive all playback commands (i.e. Play, Stop, Pause, etc.) from the controller card. To ensure perfect synchronization, the reproducers share common Video and Audio clocks that are supplied by the controller card as well.

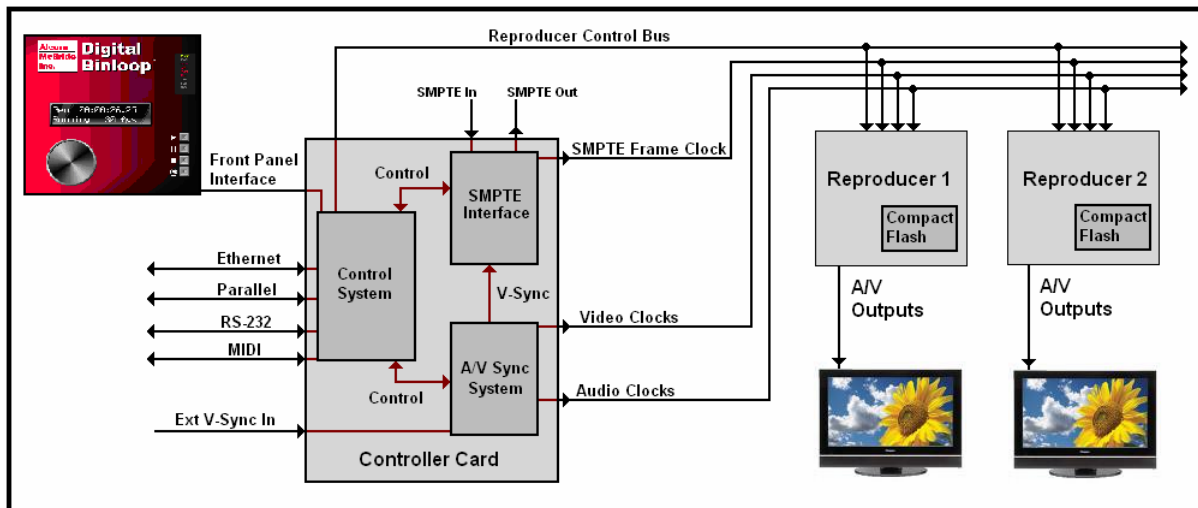
The video files themselves can be encoded using one of several popular video and audio codecs that are supported by the reproducers. For more specific information regarding the type of media content that the reproducer cards can play, refer to the chapter that covers **Media Files**.

The CobraNet Card

CobraNet is a common industry standard for distributing audio over an Ethernet-based network. When a Binloop is equipped with the CobraNet option, you can distribute the stereo audio outputs of up to 16 reproducers (that's 32 channels of audio) over a CobraNet network. When this option is installed, an additional RJ-45 Ethernet jack is located on the back of the Binloop cage. For more information about CobraNet, including how to configure and use this feature, refer to the **CobraNet** section of this manual.

How the Binloop Works

Now that you're more familiar with the various hardware components that comprise the Binloop, let's get more familiar with how those components actually work together. The diagram below outlines the different components we've covered as well as their functional relationship with one another.



The sections that follow will cover the various systems you see in this diagram, so please use it as a visual reference.

Control System

The Control system of the controller card is the heart of the Binloop. This system handles all of the Binloop's control interfaces. In addition, the Control system is responsible for configuring and controlling both the SMPTE Interface and the A/V Sync system that also reside on the controller card. This system's final significant role is to configure and control the reproducers over a shared communications line.

Control Interfaces

The Control system monitors all of the control interfaces, which include Ethernet, Parallel, RS-232, MIDI, and the Front Panel interface. Anything that happens on these interfaces, in one way or another, is translated by the Control system into a command to do something.

For example, the Control system may receive an RS-232 command to play video 1 on reproducer 1. It must interpret this command, and then send the appropriate control signal to the reproducer to make it happen. Another common example would be using the Parallel control port to start the SMPTE generator. The Control system would sense a contact closure on the Parallel control port, and then instruct the SMPTE Interface to start generating timecode.

You can read more about these control interfaces, and how they work in a later chapter in this manual titled **Controlling the Binloop**.

Reproducer Control

Without the Control system, the reproducers would do nothing except generate black video and waste electricity. They have no knowledge of the SMPTE clock value, RS-232 commands, Parallel input status, or anything of the sort. They are simply designed to obey basic playback commands and, in that regard, the Control system is their boss. It is responsible for issuing configuration and playback commands to the reproducers, and is capable of doing so with tremendous precision.

A/V Sync System

The A/V Sync system plays a critical role in the operations of the Binloop. This system is responsible for generating the Audio and Video clocks used throughout the Binloop. Most notably, these clocks are used directly by the reproducers to ensure that the video and audio outputs are precisely synchronized with one another.

Configuration Options

This system has a few straight-forward, yet important, configuration settings. The first of these settings, Video Format, configures the type of video clocks to generate. There are only two options that cover the primary video systems used throughout the world; NTSC and PAL. The other setting is the audio sample rate, which determines how to generate the audio clocks. Here's a table that shows the settings that are available to you.

Setting	Options
Video Format	NTSC, PAL
Sample Rate	32kHz, 44.1kHz, 48kHz, 96kHz

These settings can be configured using the Menu Wheel interface on the front of the Binloop, as well as the Binloop Configuration software.

External V-Sync

If you refer to the diagram earlier in this section, you will see that the A/V Sync system allows for an optional external video sync source. When a supported sync source is connected to V-Sync input of the Binloop, the A/V clocks are locked directly to the source.

This input is designed to accept a Blackburst video sync signal. When the A/V Sync system is successfully locked on to an external sync input, the V-Sync status LED on the front of the Binloop will illuminate. Keep in mind that the Video Format setting of the A/V Sync system must match the video format of the Blackburst sync source. If it does not, the A/V Sync system will not attempt to lock to the source.

The A/V Sync system always generates the A/V clocks regardless of whether or not an external sync source is present.

SMPTE Interface

The SMPTE Interface is what gives the Binloop the ability to generate and read SMPTE and EBU timecode. Whether it is generating or reading, this interface provides the Control system with the current SMPTE time. The most important purpose for this is to execute precisely timed user-programmed commands, or triggers, to the reproducer cards. In addition, it provides a SMPTE Frame Clock which is used throughout the Binloop for synchronization purposes.

Configuration Options

The SMPTE Interface is a very powerful and flexible system and, as a result, it has a variety of configurable settings. Here is a summary of those settings:

Setting	Description
SMPTE Mode	Determines whether the SMPTE Interface will generate timecode, or read external timecode.
Framerate	The framerate in which SMPTE is generated.
Preroll Time	The initial time of SMPTE generator when it is first enabled.
Start Time	The time that the SMPTE generator returns to after it reaches the End Time (when looping)
End Time	The time when the SMPTE generator either stops or loops back to the Start Time.
Loop Mode	Determines whether the SMPTE generator loops continuously from Start to End Time.
Powerup Mode	Determines whether SMPTE is Enabled or Disabled when the Binloop powers up.
Restart Mode	Determines how the SMPTE generator responds when it is told to start again. It can either restart or ignore the start command.
Idle Mode	Determines if the SMPTE Output is muted or active when the SMPTE Interface is idle.
Self-Gen Mode	Determines how long the SMPTE reader will free-wheel when external SMPTE is lost.
Output Level	Configures the voltage level (in dB) of the SMPTE output.

These settings can all be configured using the Menu Wheel interface on the front of the Binloop, or with through the Binloop Config software. Please read on to learn more about these settings and how they can be used to configure the SMPTE Interface to meet your needs.

SMPTE Mode

There are 3 basic modes that the SMPTE Interface can operate in. It can be configured to **Read** timecode, **Generate** timecode, or generate timecode in reference to the Binloop's video clocks (**Generate /w V-Sync**).

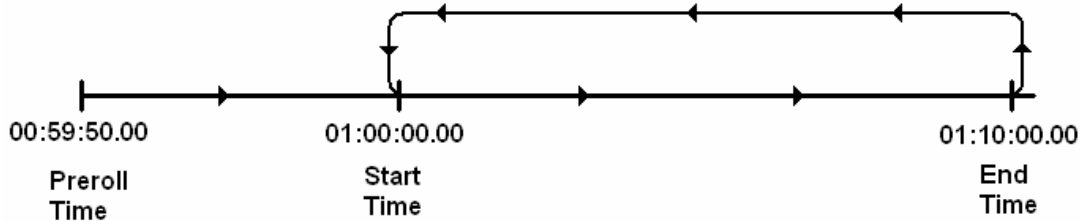
In **Read** mode, the SMPTE module will listen for incoming timecode on the Binloop's SMPTE In connector whenever the SMPTE module is enabled. While operating in this mode, the SMPTE Interface will repeat the incoming timecode on the SMPTE Out connector.

In **Generate** mode, the SMPTE Interface will generate timecode when it is enabled (or started). This timecode is used internally, and is also generated on the SMPTE Out connector.

The **Generate /w V-Sync** mode operates exactly like the normal Generate mode, except that the SMPTE clock is generated synchronously with the vertical sync video signal of the Binloop. For example, if the Binloop's A/V Sync system is configured for NTSC (29.97fps), the SMPTE module can generate 29.97fps timecode that is perfectly synchronous. This means that each 'tick' of the SMPTE clock will coincide with the beginning of each video frame, even after a long period of time. This feature is useful when you want the SMPTE clock to remain in perfect sync with a video presentation, or with an external video sync source.

SMPTE Times

The Preroll, Start, and End times are all used by the SMPTE Interface when it is programmed to generate timecode. This graphic demonstrates how these times are used by the generator.



As you can see, the timecode starts running from the Preroll Time when the SMPTE Interface is first enabled. The timecode then continues to run until the End Time is reached. What happens at this point depends on the next setting; Loop Mode.

Loop Mode

The Loop mode setting determines what the timecode generator does when the End Time is encountered. If the SMPTE module is configured to loop, the timecode jumps back to the Start Time. If loop mode is disabled, the timecode stops running.

Powerup Mode

The Powerup setting simply tells the SMPTE Interface whether or not to start running when the Binloop first turns on. When the Binloop is configured to start on powerup, the SMPTE Interface will begin generating/reading timecode as soon as the system is 'ready' (this is indicated by both the 'ready' output and 'ready' unit status LED).

Restart Mode

The Start SMPTE Command can come from any of the Binloop's control interfaces, including Parallel, RS-232, Ethernet, MIDI, or the front-panel.

The Restart Mode setting only applies when the SMPTE Interface is configured to generate timecode. This setting determines how the generation process is affected if a 'Start SMPTE' command occurs when timecode is currently running. If the Restart Mode is enabled, another Start command causes the timecode to jump back to the Start Time. If the Restart Mode is disabled, any start commands received while the timecode is running will be completely ignored.

Idle Mode

The Idle Mode determines the state of the SMPTE Output when the SMPTE Interface is Stopped. If the Idle Mode is configured to mute SMPTE, the SMPTE output will be silent when the SMPTE clock is stopped. If the Idle Mode is configured for SMPTE to be active, the SMPTE Output will continuously repeat the timecode value where the clock stopped running on the SMPTE Output.

Self-Gen Mode

If external SMPTE drops out, the SMPTE reader will automatically re-lock when it is restored. The Self-Gen Mode simply determines what to do in the meantime.

This setting applies only when the SMPTE Interface is configured to Read external timecode. It determines how the SMPTE reader should react in the event that the external SMPTE signal drops out, or is lost completely. When this occurs, the SMPTE reader has the ability to continue generating (free-wheel) SMPTE on its own. This allows the Binloop to 'get by' until the external SMPTE clock is restored. If you want to implement this feature, this setting can configure the SMPTE Reader to self-generate timecode for a period of 1-8 seconds, or indefinitely. If you don't want to use this feature, Self-Gen mode can be disabled completely so that the timecode within the Binloop always represents the timecode coming in on the SMPTE Input.

Controlling the Binloop

To ensure that the Binloop can meet the needs of almost any application, it has been designed with several different control interfaces. These interfaces include:

1. **Serial (RS-232)**
2. **MIDI**
3. **Ethernet**
4. **Parallel**
5. **Timecode**

Serial Control (RS-232)

The Binloop provides a dedicated **Show Control** RS-232 port to allow external RS-232 devices to control the product. For added flexibility, this port can be configured to use a variety of serial protocols. Here's a table that outlines the differences between these protocols:

Protocol	Baud Rate	Description
Alcorn 9-bit	9600	For use with Alcorn Show Controllers.
AMI/Pioneer	9600	Generic ASCII-based protocol intended for use by almost any serial control device.
MIDI	38400	For use with MIDI Show Control systems.
Debug	38400	Debug output only. Not used for control.

Once you determine which protocol is best for your system, you must configure the **Show Control** port for one of these 4 protocols. This can be done using the front-panel Menu Wheel, or by downloading a configuration from the Binloop Configuration software. When you are ready to control the Binloop, you must connect a NULL RS-232 cable between the controlling device and the Binloop's **Show Control** port. Once you have done this, read on to the **Control Protocols** section of this manual to obtain the full list of commands you can send to the Binloop.

MIDI Control

The Binloop has a dedicated MIDI port for use with MIDI-based control systems. This port operates at the MIDI standard baud rate of 31250. Any MIDI command that is received is repeated on to the MIDI OUT connector.

Refer to the **Control Protocols** section of this manual for the complete list of MIDI Show Control commands that are supported by the Binloop.

Ethernet Control

For Ethernet control, the Binloop uses a standard Ethernet Protocol called UDP (User Datagram Protocol).

UDP is a very simple Ethernet protocol that allows an Ethernet device to transmit a generic string of data to another Ethernet device. Since UDP is such a broad standard, UDP packets can be generated by a large variety of controllers and PC applications. The UDP packet handles all of the complexities of getting the message where it needs to go in one piece. The good news is that you do not have to understand those complexities. As a matter of fact, you only need to know the following information to send a UDP message.

Destination IP Address – This needs to be the IP address of the Binloop you intend to control. The default IP address of the Binloop is **192.168.0.254**.

Destination Port – This value needs to be the UDP port that the Binloop is listening to for UDP commands. This value is always **2638**.

Source Port - This is the UDP port that your PC or controller will use to send the UDP message. It does not matter what port you choose, as long as it is a valid UDP port. If you have no idea what a valid UDP port is, just use **2638**.

Data – This is the actual message that you are sending to the Binloop. This message needs to be an AMI/Pioneer Protocol command. (i.e. 1R1PL<CR>).

Since the UDP commands use the AMI/Pioneer Protocol, they are ASCII based and are terminated with a carriage return (hexadecimal value 0x0D). The complete list of supported commands can be found in the **AMI/Pioneer Protocol** section of this manual.

Parallel Control

The Binloop's Parallel Control interface is designed to provide a very basic control from simple voltage and contact closure inputs. This interface also has 3 dry-contact relay outputs and 2 TTL level outputs that indicate the status of the Binloop. All of these signals are available on the DB-37 **Parallel Control** connector located on the back of the Binloop.

Parallel Inputs

The following table lists the available inputs of the Binloop's Parallel control interface, as well as description of their functions:

Function	Contact Closure Pins	Voltage Pins	Description
*Start		7(+),8(-)	Start SMPTE
*Pause		9(+),10(-)	Pause SMPTE at next SMPTE loop point
*Stop		11(+),12(-)	Stop SMPTE, Stop all reproducers.
*Mute Audio		13(+),14(-)	Mute Audio on all reproducers
Play Group #1	20, GND**		Play and Loop clip #1 on Group #1
Play Group #2	21, GND**		Play and Loop clip #2 on Group #2
Play Group #3	22, GND**		Play and Loop clip #3 on Group #3
Play Group #4	23, GND**		Play and Loop clip #4 on Group #4
Play Group #5	24, GND**		Play and Loop clip #5 on Group #5
Play Group #6	25, GND**		Play and Loop clip #6 on Group #6
Play Group #7	26, GND**		Play and Loop clip #7 on Group #7
Play Group #8	27, GND**		Play and Loop clip #8 on Group #8
Reset SMPTE	28, GND**		Stop SMPTE
Idle SMPTE	29, GND**		Pause SMPTE immediately
Stop Group	31, GND**		Changes the behavior of pins 20-27 to Stop the group (instead of play and loop)
Loop Inhibit	32, GND**		Changes the behavior of pins 20-27 to Play the group (instead of play and loop)

*These functions are identical to the front-panel control buttons of the Binloop.

**GND can be any one of the ground pins on the Binloop's Parallel Control Connector. This includes pins 16, 17, 18, and 19.

Parallel Outputs

The following table lists the status outputs that are provided by the Parallel control interface of the Binloop:

Function	TTL Output Pins	Contact Closure Pins	Description
*Running		1,2	CLOSED when SMPTE is running
*Fault		3,4	CLOSED when Binloop is in Fault state
*Ready		5,6	CLOSED when Binloop is ready to receive commands
SMPTE Lock	33, GND**		HIGH (5V) when locked on to external SMPTE
VSync Lock	34, GND**		HIGH (5V) when locked on to external Video Sync

*These relay outputs directly mimic the **Run**, **Fault**, and **Ready** unit status LED's on the front panel of the Binloop.

**GND can be any one of the ground pins on the Binloop's Parallel Control Connector. This includes pins 16, 17, 18, and 19.

Using the Group Control Inputs

The Binloop has contact-closure inputs on the rear panel that can be used to command groups of Reproducers to Play or Stop. When used on their own, these 8 'Play Group' inputs cause the associated group to play and loop the associate file. For example, activating the 'Play Group 2' input will cause all reproducers assigned to group 2 to play and loop file #2. Activating 'Play Group 4' will cause all members of group 4 to play and loop file #4.

There are some additional inputs that can be used in conjunction with the 'Play Group' inputs to change their behavior. These include the 'Stop Group' input and the 'Loop Inhibit' input. As the name implies, activating the 'Stop Group' input simultaneously with one of the 'Play Group' inputs will cause that group to stop (instead of play and loop). Activating the 'Loop Inhibit' input simultaneously with a 'Play Group' input causes the associated file to play through once (instead of loop indefinitely).

Using the Start SMPTE Input

*Please refer to the **Hardware** section of this manual for pin assignments and other electrical details for the Parallel Control connector.*

The Parallel Control connector also contains inputs for starting or stopping the SMPTE Interface. Unlike the Group Control inputs, the SMPTE Control inputs are Opto-Isolated and require a voltage (24V) to be input to them from an external source.

The way to start playback using the SMPTE Control inputs is to configure the Binloop to generate SMPTE, program a playback trigger in the SMPTE Trigger Table and start SMPTE using the Start SMPTE input. When the timecode reaches the programmed trigger time, playback begins.

Single Reproducers, Groups of Reproducers, or All Reproducers can be triggered. It's all in how you program the trigger.

Using the Mute Input

The Parallel Control connector also contains an input that can Mute the audio output on all of the Reproducers. This input mimics the behavior of the front panel mute button. When this input is active, all reproducer audio outputs are muted. The audio will remain muted until this input is released.

Timecode Control

In this method of control, the commands are pre-programmed ahead of time using the Binloop Configuration software, and then downloaded and stored in the Binloop.

The concept is pretty simple. When you program the commands, you must specify a SMPTE timecode value for that command. If the SMPTE clock ever equals that timecode value, the command (or trigger) is executed. This concept applies whether the SMPTE is being read from an external source, or generated internally by the Binloop.

There are several benefits to using this control interface. First, this interface allows the Binloop to operate completely on its own. For example, you could program the Binloop to automatically generate SMPTE timecode when it powers-up. You could then create a 'Play' command that gets triggered by the timecode shortly after it starts running. The result is that the Binloop automatically starts playing after power-up with no external control required!

The other benefit to this interface is precision. Each command's SMPTE execution time is accurate down to the frame. For example, you could make a video start playing at exactly 01:02:05.21. Again, it doesn't matter if that timecode is being read from an external source or generated internally. This precision also helps with tweaking the playback timing. Is the video starting a little too soon? Just add 1 or 2 frames to the SMPTE execution time to make it start a little later.

For more information about using this method of control, please refer to the **Binloop Configuration Software** section of this manual. If you would like to configure the Binloop to automatically play upon power up, please see the **Application Notes** section of this manual.

Control Protocols

AMI/Pioneer Control Protocol

*For specific information about the RS-232 and Ethernet interfaces, refer to the **Controlling the Binloop** section of this manual.*

The AMI/Pioneer protocol commands listed in this section are available for use on both the RS-232 **Show Control** port and the **Ethernet** port of the Binloop. This section discusses the most commonly used commands in detail.

This protocol inherits its name from the fact that it closely resembles the popular Pioneer protocol used throughout the A/V industry.

The protocol is ASCII-based. Upper or lower case characters can be used interchangeably. **All commands and responses are terminated with the Carriage Return character. This character is represented by the hexadecimal value 0x0D.**

Get Firmware Version

Description: This command will return the controller firmware version number.

Command: **?V**
Response: **DBAP4 Vx.xx**

Example: Command: **?V**
Response: **DBAP4 V1.23**

Get SMPTE Firmware Version

Description: This command will return the firmware version of the SMPTE module.

Comments: It is possible for the SMPTE firmware to be corrupted. If this occurs, this command will return a hardware error (E01).

Command: **?S**
Response: **SMPTE Vx.xx**

Example: Command: **?S**
Response: **SMPTE V1.23**

Get/Set Unit ID

Description: This command will get or set the unit ID number.

Parameters: **xx** represents a decimal number between 1 – 49.

Set Command: **xxID**

Response: **R**

Example: Command: **1ID**
Response: **R**

Get Command: **ID**

Response: **xx**

Example: Command: **ID**
Response: **1**

Get/Set IP address

Description: This command will get or set the unit IP address.

Parameters: **xxx.xxx.xxx.xxx** represents a standard IPv4 IP address.

Get Command: **IP**

Response: **xxx.xxx.xxx.xxx**

Example: Command: **IP**
Response: **192.168.0.254**

Set Command: **xxx.xxx.xxx.xxxIP**

Response: **R**

Example: Command: **192.168.0.254IP**
Response: **R**

Get/Set Subnet Mask number

Description: This command will get or set the number used to isolate the subnet.

Parameters: **xxx.xxx.xxx.xxx** is a standard IPv4 subnet mask.

Get Command: **SM**

Response: **xxx.xxx.xxx.xxx**

Example: Command: **SM**
Response: **255.255.255.0**

Set Command: **xxx.xxx.xxx.xxxSM**

Response: **R**

Example: Command: **255.255.255.0SM**
Response: **R**

Get/Set Gateway address

Description: This command will get or set the IP address of the network gateway.

Parameters: **xxx.xxx.xxx.xxx** is a standard IPv4 IP gateway address.

Get Command: **GW**

Response: **xxx.xxx.xxx.xxx**

Example: Command: **GW**
Response: **192.168.0.1**

Set Command: **xxx.xxx.xxx.xxxGW**

Response: **R**

Example: Command: **192.168.0.1GW**
Response: **R**

Enable SMPTE

Description:	This command will Enable the SMPTE interface.
Comments:	If the SMPTE module is configured to Generate, this command will cause the clock to start generating at the configured Preroll time. If the SMPTE module is configured to Read, it will start listening for SMPTE time code on the SMPTE Input. If the SMPTE clock is in a paused state, this command will cause it to resume from its current position.
Command:	ES
Response:	R

Disable SMPTE

Description:	This command will disable the SMPTE interface.
Comments:	If the SMPTE module is configured to Generate, this command will cause the SMPTE clock to stop at its current time. If the SMPTE module is configured to Read, this command will cause the SMPTE clock to stop running and ignore any incoming time code.
Command:	DS
Response:	R

Pause SMPTE

Description:	This command will pause the SMPTE clock at the next loop point.
Comments:	This command only applies when the SMPTE module is in Generate mode and is also configured to loop. When paused, the SMPTE clock can be resumed by sending an Enable SMPTE command.
Command:	PS
Response:	R

Idle SMPTE

Description:	This command will pause the SMPTE clock immediately.
Comments:	When paused, the SMPTE clock can be resumed by sending an Enable SMPTE command.
Command:	IS
Response:	R

Get/Set SMPTE time

Description: This command will get or set the current SMPTE time.

Parameters: **hh** = hours
 mm = minutes
 ss = seconds
 ff = frames

Get Command: **CT**
Response: **hh:mm:ss.ff**

Example: Command: **CT**
 Response: **01:59:59.30**

Set Command: **hh:mm:ss.ffCT**
Response: **R**

Example: Command: **01:59:59.30CT**
 Response: **R**

Get/Set SMPTE Mode

Description: This command will get or set the operating mode of the SMPTE module.

Parameters: **x** = mode number:
 0 = read
 1 = generate
 2 = generate with V-sync

Get Command: **SO**
Response: **x**

Example: Command: **SO**
 Response: **2**

Set Command: **xSO**
Response: **R**

Example: Command: **2SO**
 Response: **R**

Get/Set SMPTE Frame Rate Mode

Description: This command will get or set the SMPTE frame rate.

Parameters: **x** = mode number:
0 = 23.976 fps
1 = 24 fps
2 = 25 fps
3 = 29.97 fps
4 = 30d fps
5 = 30 fps

Get Command: **FR**

Response: **x**

Example: Command: **FR**
Response: **5**

Set Command: **xFR**

Response: **R**

Example: Command: **5FR**
Response: **R**

Get/Set SMPTE Preroll time

Description: This command will get or set the SMPTE preroll time.

Parameters: **hh** = hours
mm = minutes
ss = seconds
ff = frames

Get Command: **PT**

Response: **hh:mm:ss.ff**

Example: Command: **PT**
Response: **00:01:30.10**

Set Command: **hh:mm:ss.ffPT**

Response: **R**

Example: Command: **00:01:30.10PT**
Response: **R**

Get/Set SMPTE Start time

Description: This command will get or set the SMPTE start time.

Parameters: **hh** = hours
mm = minutes
ss = seconds
ff = frames

Get Command: **RT**
Response: **hh:mm:ss.ff**

Example: Command: **RT**
Response: **00:00:01.00**

Set Command: **hh:mm:ss.ffRT**
Response: **R**

Example: Command: **00:00:01.00RT**
Response: **R**

Get/Set SMPTE End time

Description: This command will get or set the SMPTE end time.

Parameters: **hh** = hours
mm = minutes
ss = seconds
ff = frames

Get Command: **ET**
Response: **hh:mm:ss.ff**

Example: Command: **ET**
Response: **01:00:00.00**

Set Command: **hh:mm:ss.ffET**
Response: **R**

Example: Command: **01:00:00.00ET**
Response: **R**

Get/Set SMPTE Loop Mode

Description: This command will get or set the current SMPTE loop mode.

Parameters: **x** = mode number:
 0 = SMPTE looping is Disabled on power up
 1 = SMPTE looping is Enabled on power up

Get Command: **LM**
Response: **x**

Example: Command: **LM**
 Response: **1**

Set Command: **xLM**
Response: **R**

Example: Command: **1LM**
 Response: **R**

Get/Set SMPTE Powerup Mode

Description: This command will get or set the current SMPTE power up mode.

Parameters: **x** = mode number:
 0 = SMPTE Disabled on power up
 1 = SMPTE Enabled on power up

Get Command: **PM**
Response: **x**

Example: Command: **PM**
 Response: **1**

Set Command: **xPM**
Response: **R**

Example: Command: **1PM**
 Response: **R**

Get/Set SMPTE Restart Mode

Description: This command will get or set the current SMPTE restart mode.

Parameters: **x** = mode number:
 0 = SMPTE restart is Disabled on power up
 1 = SMPTE restart is Enabled on power up

Get Command: **RM**
Response: **x**

Example: Command: **RM**
 Response: **1**

Set Command: **xRM**
Response: **R**

Example: Command: **1RM**
 Response: **R**

Get/Set SMPTE Idle Mode

Description: This command will get or set the current SMPTE idle mode.

Parameters: **x** = mode number:
 0 = SMPTE output is active when idle
 1 = Mute SMPTE output when idle

Get Command: **IM**
Response: **x**

Example: Command: **IM**
 Response: **1**

Set Command: **xIM**
Response: **R**

Example: Command: **1IM**
 Response: **R**

Get/Set Video Format

Description:	This command will get or set the current video mode. This command sets the A/V system to the appropriate frequency of 29.97 or 25 Hz. The reproducers will also receive the command to change the frequency as well.		
Parameters:	x = mode number: 0 = NTSC 1 = PAL		
Get Command:	VF		
Response:	x		
Example:	Command:	VF	
	Response:	0	
Set Command:	xVF		
Response:	R		
Example:	Command:	0VF	
	Response:	R	

Get/Set Synchronous Play Delay

Description:	This command configures the synchronous play delay of the Binloop reproducers. By default, this delay is 16 frames , but this setting allows you to override that default if necessary. Be warned that decreasing this delay too much <u>will</u> result in synchronization problems. The reproducers must have adequate time to load the video or audio clip before it is capable of starting playback.		
Parameters:	nnn is a number between 0-127 that represents the S-Play delay setting 0 – Use Reproducer Default Delay (16 Frames) 1-127 – Delay (in frames)		
Get Command:	SD		
Response:	nnn		
Set Command:	nnnSD		
Response:	R		
Example:	Set S-Play Delay to 8 frames:	8SD	
	Set S-Play Delay to the Reproducer Default :	0SD	

Search Clip

Description:	This command causes the Binloop to preload the specified clip. If it is a video clip, the reproducer(s) will pause on the first frame of the video. After this happens, the next Play command causes the clip to start playing.	
Command:	nnnnaaaSE	
Parameters:	nnnn is an ASCII file number between 0-1023 aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16 Gxx – Group address between G1-G13 * - Wildcard address for all reproducers	
Response:	R	
Examples:	Search File 1 on Reproducer #2: 1R2SE Search File 2 on Group #1: 2G1SE Search File 52 on All Reproducers: 52*SE	

Play / Resume

Description:	This command will start playing a clip that was previously loaded by a Search Clip command. This command will also resume playback when the reproducer has been stilled or paused.	
Command:	aaaPL	
Parameters:	aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16 Gxx – Group address between G1-G13 * - Wildcard address for all reproducers	
Response:	R	
Example:	Search File 1 on Reproducer #2: 1R2SE Play pre-searched file R2PL	

Loop / Resume

Description:	This command functions just like the Play / Resume command except that the video loops once the end of the clip is reached.	
Command:	aaaLP	
Parameters:	aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16 Gxx – Group address between G1-G13 * - Wildcard address for all reproducers	
Response:	R	
Example:	Search File 1 on Reproducer #2: 1R2SE Loop pre-searched file: R2LP	

Play Clip

Description:	This command will play the specified clip on the addressed reproducer card(s). Playback starts as soon as possible after the command is received.		
Command:	nnnnaaaPL		
Parameters:	nnnn is an ASCII file number between 0-1023 aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16 Gxx – Group address between G1-G13 * - Wildcard address for all reproducers		
Response:	R		
Comments:	Since playback starts as quickly as possible once this command is received, it should never be used for synchronous playback. For this, you should use the Synchronous Play command for a consistent synchronized start of playback. Using a Search command along with a timed Play/Resume command works well for synchronous playback too.		
Examples:	Play File 3 on Reproducer #1:	3R1PL	
	Play File 4 on Group #2:	4G2PL	
	Play File 52 on All Reproducers:	52*PL	

Play/Loop Clip

Description:	This command functions exactly like the Play Clip command, except that the video loops once the end of the clip is reached.		
Command:	nnnnaaaLP		
Parameters:	nnnn is an ASCII file number between 0-1023 aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16 Gxx – Group address between G1-G13 * - Wildcard address for all reproducers		
Response:	R		
Examples:	Loop File 3 on Reproducer #1:	3R1LP	
	Loop File 4 on Group #2:	4G2LP	
	Loop File 52 on All Reproducers:	52*LP	

Synchronously Play Clip

Description:	This command functions exactly like the Play Clip command, except that the playback does not begin as quickly as possible. Instead, the reproducer waits a consistent amount of time (16 frames by default) after the command is received before it starts playback. This ensures that all reproducers that receive this command at the same time, start playing at the same time.
Command:	nnnnaaaSP
Parameters:	nnnn is an ASCII file number between 0-1023 aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16 Gxx – Group address between G1-G13 * - Wildcard address for all reproducers
Response:	R
Comments:	This command is almost always addressed to either a Group or All reproducers to ensure that they all start playing synchronously.
Examples:	Synchronously Play File 2 on Group 1: 2G1SP Synchronously Play File 2 on All Reproducers: 2*SP

Synchronously Play/Loop Clip

Description:	This command functions like the Synchronously Play Clip command, except that the clip loops once the end of the clip is reached.
Command:	nnnnaaaSL
Parameters:	nnnn is an ASCII file number between 0-1023 aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16 Gxx – Group address between G1-G13 * - Wildcard address for all reproducers
Response:	R
Examples:	Synchronously Loop File 2 on Group 1: 2G1SL Synchronously Loop File 2 on All Reproducers: 2*SL
Comments:	If you plan to synchronously loop files on multiple reproducers, the clips MUST be identical in length. If it's a video, they must be identical down to the frame. If it's audio, they must be identical down to the sample. If they are not, the clips will not remain in sync after the loop point.

Play Next

Description:	This command causes the specified file to be "queued" for playback at the completion of the current clip. If a file is currently playing, the queued transition to the selected file will be seamless. If a file is currently looping, the seamless transition will occur at the next loop point. If a clip is not currently playing, the clip specified in this command will begin playing immediately.		
Command:	nnnnaaaPN		
Parameters:	nnnn is an ASCII file number between 0-1023 aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16 Gxx – Group address between G1-G13 * - Wildcard address for all reproducers		
Response:	R		
Example:	Play File #2 on Reproducer 1:	2R1PL	
	Queue File #3 to play after file #2 finishes:	3R1PN	

Loop Next

Description:	This command functions exactly like the Play Next command except that the queued file loops after it finishes playing.		
Command:	nnnnaaaLN		
Parameters:	nnnn is an ASCII file number between 0-1023 aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16 Gxx – Group address between G1-G13 * - Wildcard address for all reproducers		
Response:	R		
Example:	Play File #2 on Reproducer 1:	2R1PL	
	Queue File #3 to loop after file #2 finishes:	3R1LN	

Stop Playback

Description:	This command will stop currently playing clip. If a video was playing, the last frame that was displayed will remain on screen.	
Command:	aaaRJ	
Parameters:	aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16 Gxx – Group address between G1-G13 * - Wildcard address for all reproducers	
Response:	R	
Examples:	Stop Reproducer #1:	R1RJ
	Stop Group #2:	G2RJ
	Stop All Reproducers:	*RJ

Still Playback

Description:	This command will pause the currently playing clip at its present position. If a video was being played, the stilled video frame will remain on screen.	
Command:	aaaST	
Parameters:	aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16 Gxx – Group address between G1-G13 * - Wildcard address for all reproducers	
Response:	R	
Example:	Still Reproducer #1:	R1ST
	Resume Reproducer #1:	R1PL

Pause Playback

Description:	This command works exactly like the Still command except that the video output will mute to black while the video is paused. When the Play/Resume command is used after this command, the video is unmuted and restored automatically.	
Command:	aaaPA	
Parameters:	aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16 Gxx – Group address between G1-G13 * - Wildcard address for all reproducers	
Response:	R	
Example:	Pause Reproducer #1:	R1PA
	Resume Reproducer #1:	R1PL

Mute/Unmute Video

Description:	This command controls muting on the video outputs of the Binloop. When muted, the video outputs are black. When unmuted, the video outputs display video normally.		
Command:	naaaVD		
Parameters:	n is a number between 0-1 that represents the mute state 0 – Muted (black output) 1 – Unmuted (normal output) aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16 Gxx – Group address between G1-G13 * - Wildcard address for all reproducers		
Response:	R		
Example:	Mute Video on Reproducer #1:	0R1VD	
	Unmute Video on Reproducer #1:	1R1VD	

Mute/Unmute Audio

Description:	This command controls muting on the audio outputs of the Binloop. When muted, all audio outputs are silent. When unmuted, the audio outputs are normal.		
Command:	naaaAD		
Parameters:	n is a number between 0-1 that represents the mute state 0 – Muted (silent) 1 – Unmuted (normal output) aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16 Gxx – Group address between G1-G13 * - Wildcard address for all reproducers		
Response:	R		
Example:	Mute Audio on Reproducer #1:	0R1AD	
	Unmute Audio on Reproducer #1:	1R1AD	

Play Independent (Audio Only)

Description: This command causes the specified mono audio file to be played on the specified audio channel. If a video file is playing, or an audio file is playing 'normally' on both channels, playback of that file will be interrupted. If a file is being played independently on the opposite channel, it will not be interrupted.

Command: **nnnnaaacPI**

Parameters: **nnnn** is an ASCII file number between 0-1023
aaa is the address of the card(s) you want to process the command
 Rxx – Reproducer address between R1-R16
 Gxx – Group address between G1-G13
 * - Wildcard address for all reproducers
c is an ASCII character representing the channel
 L – Left Channel
 R – Right Channel

Response: **R**

Example: Play File #2 on Left Channel of Reproducer 1: **2R1LPI**

Loop Independent (Audio Only)

Description: This command functions exactly like the Play Independent command except that the file loops after it finishes playing.

Command: **nnnnaaacLI**

Parameters: **nnnn** is an ASCII file number between 0-1023
aaa is the address of the card(s) you want to process the command
 Rxx – Reproducer address between R1-R16
 Gxx – Group address between G1-G13
 * - Wildcard address for all reproducers
c is an ASCII character representing the channel
 L – Left Channel
 R – Right Channel

Response: **R**

Example: Loop File #3 on Right Channel of Reproducer 2: **3R2RLI**

Stop Independent (Audio Only)

Description: This command stops independent playback on the specified channel.

Command: **aaacSI**

Parameters: **aaa** is the address of the card(s) you want to process the command
 Rxx – Reproducer address between R1-R16
 Gxx – Group address between G1-G13
 * - Wildcard address for all reproducers
 c is an ASCII character representing the channel
 L – Left Channel
 R – Right Channel

Response: **R**

Get/Set Video Colorspace

Description: This command configures the colorspace of the analog component video output of the specified reproducer(s).

Parameters: **n** is a number between 0-1 that represents the colorspace setting
 0 – RGB
 1 – YPbPr
 aaa is the address of the card(s) you want to process the command
 Rxx – Reproducer address between R1-R16
 Gxx – Group address between G1-G13
 * - Wildcard address for all reproducers

Get Command: **aaaVC**

Response: **n**

Set Command: **naaaVC**

Response: **R**

Example: Set Reproducer #1 to YPbPr colorspace: **1R1VC**
 Set Reproducer #1 to RGB colorspace: **0R1VC**

Get Reproducer Firmware Version

Description:	This command requests the firmware version of the specified reproducer. If the specified reproducer is not installed in the Binloop, there will be no response. This command requires DBAP4 Controller firmware V1.17 or newer.	
Command:	aaa?V	
Parameters:	aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16	
Response:	REPRO-16V Vx.xx Or REPRO-24D Vx.xx	
Example:	Request firmware version of Reproducer #1:	R1?V
Example Responses:	Digital Binloop Reproducer with V1.59: Video Binloop Reproducer with V1.61:	REPRO-24D V1.59 REPRO-16V V1.61

Get Reproducer Status

Description:	This command requests the playback status of the specified reproducer. If the specified reproducer is not installed in the Binloop, there will be no response. Keep in mind that using this command will briefly tie up the Reproducer communication interface, so it is not recommended for use immediately before or after time-sensitive commands (like Synchronous Play). This command requires DBAP4 Controller firmware V1.17 or newer, and Repro16V or Repro24D firmware V1.61 or newer.	
Command:	aaa?P	
Parameters:	aaa is the address of the card(s) you want to process the command Rxx – Reproducer address between R1-R16	
Response:	Pnn	
Parameters:	nn is a number that represents the status of the reproducer P00 – Error (Fault) P01 – Stopped P04 – Playing P05 – Stilled P06 – Paused	
Example:	Request status of Reproducer #1:	R1?P
Example Responses:	Reproducer is Playing: Reproducer is Stopped: Reproducer has an Error:	P04 P01 P08

Get Reproducer Filename

Description: This command requests the filename of the clip currently playing in the specified reproducer. If the specified reproducer is not installed in the Binloop, there will be no response. Keep in mind that using this command will briefly tie up the Reproducer communication interface, so it is not recommended for use immediately before or after time-sensitive commands (like Synchronous Play). This command requires DBAP4 Controller firmware V1.17 or newer, and Repro16V or Repro24D firmware V1.61 or newer. **If a clip is not playing, the response will only be a Carriage Return (0x0D) character.**

Command: **aaa?C**

Parameters: **aaa** is the address of the card(s) you want to process the command
Rxx – Reproducer address between R1-R16

Response: **xxxxxxxx.xxx**

Example: Request clip filename of Reproducer #1: **R1?C**

Example Responses: Filename is SND00020.S24: **SND00020.S24**
Filename is VID00001.MPG: **VID000001.MPG**

MIDI Show Control Protocol

The MIDI Show Control protocol is primarily used when controlling the Binloop over its dedicated MIDI port. However, it is also available as one of the configurable protocols available on the RS-232 Show Control port. If your intention is to control the Binloop with a non-MIDI based system, you should consider using the Pioneer/AMI protocol instead.

It should be noted that MIDI Show Control is a very wide-ranging protocol used for many different applications. Most of the protocol information is beyond the scope of this manual since it is not critical to controlling the Binloop.

General Structure

MIDI Show Control uses a single Universal Real Time System Exclusive ID number (sub-ID #1 = 02H) for all Show commands (transmissions from Controller to Controlled Device). In this version of MIDI Show Control, command responses (from Controlled Devices to Controller) are specified in order to optimize bandwidth requirements, system response time and system reliability in the event of communication difficulties with one or more Controlled Devices. The guiding philosophy behind live performance control is that, as much as possible, failures of individual Controlled Devices should not impair communications with other Controlled Devices. This concept has been a part of MIDI system design from the beginning and MIDI Show Control continues to use an "open-loop" design in order that standard MIDI practices may continue to be successfully utilized in applications using all types of standard Channel and system messages.

Message Format

The MIDI Show Control message format is given below:

F0 7F <ID> 02 <fmt> <cmd> <Tmsb> <Tlsb> 00 <Pmod> 00 <CLh> <CLt> <CLo> F7

The following table explains each of the optional HEX bytes in this message:

Data Byte	Entry/Description
<ID>	unit address + 9 (7FH is a wild card for all units)
<fmt>	= 10 (sound) = 30 (video) = 7F (wild card) (Note: since the Binloop responds to both the 'sound' and 'video' parameter equally, normally the 'wild card' is used here)
<cmd>	= 01 (to Play Synchronized) = 02 (to Stop Sound) = 03 (to Play Unsynchronized) = 04 (to Still Video – still frame on screen) = 05 (to Pause Video – video output goes black) = 15 (to Start SMPTE)** = 16 (to Idle SMPTE)** = 17 (to Reset SMPTE)** = 18 (to Pause SMPTE)**
Tmsb Tlsb*	= track ID 30 30 - 31 35 for Reproducers 1 through 16 respectively = group ID 35 30 - 36 32 for Groups 1 through 13 respectively = all tracks ID 36 33 for accessing all Reproducers in the Binloop
Pmod	= 31 (Normal Play) = 34 (Independent Play – Right Channel) = 37 (Independent Play – Left Channel)
CLh CLt CLo*	= file number 30 30 31 - 35 31 31 for file numbers 1 through 511 respectively = file number 35 31 33 - 39 39 39 for looping file numbers 1 - 487 respectively

*Although they are showed in this table in hexadecimal form, both the Track and Clip are ASCII numbers. 2 Bytes make up the Track number, and 3 bytes make up the Clip number.

**SMPTE commands do not require the Track or Clip bytes.

Message Examples

1. Play unsynchronized clip 1 on Reproducer 1 using wild card device ID and wild card command format code:
hf0 h7f h7f h02 h7f h03 h30 h30 h00 h31 h00 h30 h30 h31 hf7
2. Play and loop unsynchronized clip 3 on Reproducer 5 using wild card device ID and wild card command format code:
hf0 h7f h7f h02 h7f h03 h30 h34 h00 h31 h00 h35 h31 h35 hf7
3. Play synchronized clip 21 on all Reproducers using wild card device ID and wild card command format code:
hf0 h7f h7f h02 h7f h01 h36 h33 h00 h31 h00 h30 h32 h31 hf7
4. Stop playback all channels (clip number is ignored):
hf0 h7f h7f h02 h7f h02 h36 h33 h00 h31 h00 h30 h30 h31 hf7
5. Still playback of video on Reproducer 1 (clip number is ignored):
hf0 h7f h7f h02 h7f h04 h30 h30 h00 h31 h00 h30 h30 h31 hf7
6. Start SMPTE:
hf0 h7f h7f h02 h7f h15 hf7
7. Reset SMPTE:
hf0 h7f h7f h02 h7f h17 hf7

Using the MIDI Show Control protocol, the Binloop can play clips numbered from 1 to 511 (i.e. VID00001.MPG to VID00511.MPG). These files can either be played from the beginning and stop at the end, or they can be commanded to play and loop at the end. Looping files play continuously, seamlessly starting over at the beginning from the end, until commanded to stop. To cause a video file to loop, you must add 512 to the clip number used in the command. Clip numbers 1 through 511 will be interpreted as non-looping by default, while 513 through 999 are looping. For example, to play clip 1 once you would send a 1 as the clip number in the MIDI show control command. To play and loop clip 1, you would use a clip number of 513 in the MIDI command (even though the actual file number is still 1 on the reproducers flash card).

Since the clip number field is only 3 digits, the maximum file number supported by the MIDI protocol is 999. That means that you can only send a loop command for clips 1-487 (999-512=487).

Acknowledge Message Format

In response to a MIDI command, the Binloop will send an acknowledge message. The format of the acknowledge message is as follows:

F0 7F <ID> <msc> F7

Data Byte	Entry/Description
<ID>	= unit address + 9
<msc>	= 02

Binloop Configuration Software

Introduction to the Binloop Software

The Binloop Configuration Software is a Windows application that provides an easy-to-use interface for programming and configuring the Binloop. While it's true that most of the Binloop's configuration settings can be accessed from the front-panel Menu Wheel interface, the Binloop application provides a more visual alternative for configuring these settings. In addition, this software is necessary to program SMPTE timecode triggers for applications where you want the Binloop to operate without any external control system.

Requirements

The Binloop software requires that you have PC running Microsoft Windows XP or Windows Vista. This PC must have either an Ethernet port and/or an RS-232 Serial port so that it may communicate with the Binloop.

Installing the Binloop Software

This software can be installed directly from the CD/USB drive that was packaged with the Binloop. If you have Internet access, we recommend downloading the latest version from the Alcorn McBride website at **<http://www.alcorn.com>**. Once you locate the installation file, simply run it on your Windows PC and follow the on-screen instructions.

Connecting your PC to the Binloop

There are two ways in which the Binloop software can interface to the Binloop from your PC:

1. **Ethernet**
2. **RS-232 Serial**

Connecting with RS-232 Serial

If your PC is not equipped with an RS-232 serial port, you can use a USB-to-Serial adaptor. Just make sure that the adaptor will support "Mark-Space Parity" operation.

For a list of USB-to-Serial adaptor manufacturers that support this protocol visit www.alcorn.com and use our Knowledge Base.

The Binloop software can communicate with the Binloop through a PC Serial Port via a 9-pin straight-through serial cable (included with the unit). Before this can happen, you must first connect the serial cable between a serial port on your PC and the **Programmer** port of the Binloop.

Once you make this connection, you are ready to use the Binloop Configuration software to program and communicate with the your Binloop.

Connecting with Ethernet

Ethernet is also a common method of interfacing the Binloop software to the Binloop. This requires a PC to be connected to the Binloop directly, or over a network, using a CAT5 (or better) rated Ethernet cable. If you want to connect directly to the Binloop from your PC (no Ethernet switch or hub) you must use a crossover Ethernet cable. When the Binloop has a good network connection, the green Link LED will illuminate.

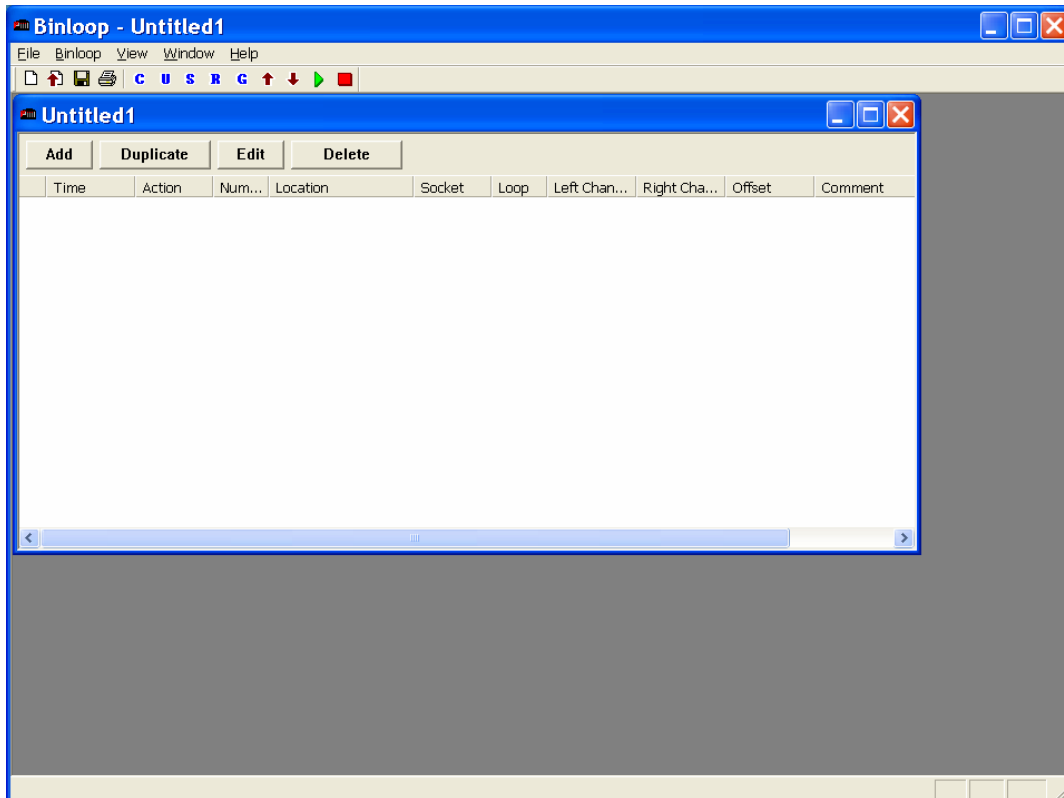
Beyond a good physical connection, Ethernet communication requires properly configured network settings on both the Binloop and your PC. Although Ethernet network design and administration is beyond the scope of this manual, we have provided a step-by-step guide in this manual that will walk you through connecting directly to the Binloop over Ethernet using the Binloop's default settings. Please refer to the **Application Notes** section if you would like more information on this.

Running Binloop Software

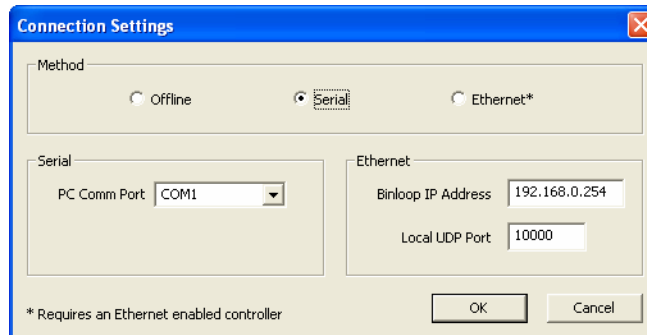
The Binloop icon on your Desktop looks like this:




Start the program by double-clicking the Binloop icon that was put on your desktop during the installation. The program can also be launched from the Windows Start menu from the **Programs→Alcorn McBride Inc** folder. When the program opens, you'll see this window...

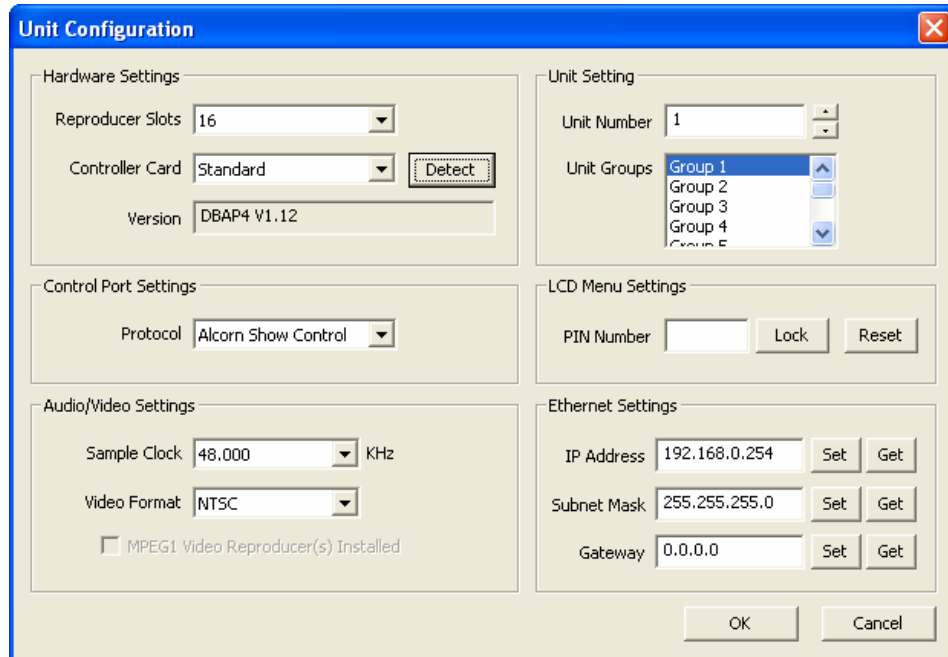


Before we get started, you'll need to configure your PC's connection with the Binloop. To do this, click on the **C** icon at the top of the Binloop window. A window will appear that will allow you to choose your preferred connection method.



If you intend to communicate with RS-232, you must select the **Serial** option and specify the COM port in your PC that is connected with the Binloop. If you are using Ethernet, you must select **Ethernet** as the communication method and then enter the Binloop's IP address in the provided box. Once you have setup the appropriate connection, click the **OK** button to close this window.

To make sure that the connection is working properly, click on the  icon at the top of the Binloop window. This will open the Unit Configuration window you see below.



The Unit Configuration window is a software interface for setting up a Binloop unit. It features a blue title bar with the text "Unit Configuration" and a close button. The window is divided into several sections: Hardware Settings, Control Port Settings, Audio/Video Settings, Unit Setting, LCD Menu Settings, and Ethernet Settings. The Hardware Settings section includes a dropdown for Reproducer Slots (set to 16), a dropdown for Controller Card (set to Standard), a Detect button, and a Version text box (displaying DBAP4 V1.12). The Control Port Settings section has a dropdown for Protocol (set to Alcorn Show Control). The Audio/Video Settings section includes a dropdown for Sample Clock (set to 48.000 KHz), a dropdown for Video Format (set to NTSC), and a checkbox for MPEG1 Video Reproducer(s) Installed. The Unit Setting section has a dropdown for Unit Number (set to 1) and a list box for Unit Groups (showing Group 1 through Group 5). The LCD Menu Settings section has a PIN Number text box and Lock and Reset buttons. The Ethernet Settings section has text boxes for IP Address (192.168.0.254), Subnet Mask (255.255.255.0), and Gateway (0.0.0.0), each with Set and Get buttons. At the bottom right are OK and Cancel buttons.

Click on the **Detect** button. If the connection is working properly, the **Version** box will display the current version of the Binloop's controller card. If this is not the case, please refer to the **Troubleshooting Guide** for assistance.

Controlling the Binloop with the Binloop Software

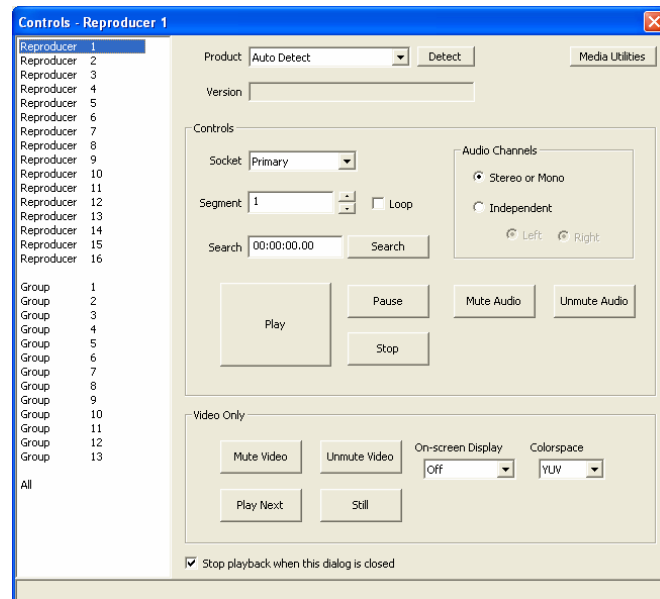
Note: Please refer to the **Media** section of this manual for more information on creating your own videos and file naming conventions for the Binloop.

Now that you are connected to the Binloop using the Binloop Software, let's go through an example of how you can use this software to control the Binloop in real-time. Specifically, this example will demonstrate how to command the Binloop to play a video file. Before we get started, it is assumed that you already have a known-working video assigned as clip #1 loaded onto the Compact Flash card of reproducer #1. It is also assumed that the outputs of reproducer #1 are properly connected to a video display. If these assumptions frighten you, then you should refer to the **Playing a Test File** guide located in the **Getting Started** section of this manual before moving on.

1. Make sure that Reproducer 1 has a flash card loaded with a video file named as clip 1 (i.e. VID00001.MPG).
2. Make sure that the Binloop, your video display, and your audio system are powered on.
3. Select *Binloop*→*Reproducer*→*Controls* from the menu of the Binloop Software.

This window serves as an interface for manually controlling the Binloop. This lets you test the functionality of the Binloop before you apply that functionality to any of the Binloop's other control interfaces.

If you try to play a file that does not exist on the media the Red Error LED on the Reproducer Flashes rapidly



4. Select *Reproducer 1* from the list on the left-hand side. Assuming that the number of the video file you want to play is 1, you can leave the *Segment* number as the default (1).
5. In the *Reproducer 1 Controls* window, use the buttons to control the playback.
6. Click on the *Play* button to start the video. The status LED of reproducer 1 will turn green.
7. Click on the *Stop* button to stop the video.

Navigating through the Menus

File Menu

The first selection in the main menu is the **File** menu, which allows you to access **.AMB** Binloop files. These files hold all of the Binloop configuration and playback parameters including address, sample clock frequency, SMPTE frame rate, and timecode triggers.

New

The first selection, **New**, creates a new Binloop configuration with default values and an empty timecode trigger list.

Open...

The second selection is **Open...** which pops up a window allowing you to open a previously saved file containing all of your program settings. Previously-stored files are recalled by browsing to the correct directory location and selecting the saved file name.



Clicking on **OK** without selecting a file does not initiate any action. Clicking on **Cancel** closes the window without opening any file or changing any settings.

Save

The **Save** option causes one of two actions. If you are using a file which was previously saved, choosing this option will immediately save the file

with the same name and update the saved version with the new information, replacing the old version. If you are using a new file which has not yet been saved, will present the **Save** window. This window allows you to name your configuration file and choose a location to store it. The **Save** window is identical to the **Save As** window.

Save As...

The **Save As...** option displays a window allowing you to save your current settings in a file with a new name.

The file name defaults to the name of the file you are currently using. The name can be changed from the default by typing a new name into the **File name** box.

Print Setup...

The **Printer Setup...** selection allows you to select and configure your printer.

Once you've programmed some timecode triggers (explained later in this chapter) options for printing the table appear in this menu.

Print...

The **Print** selection prints the timecode trigger list contained in the AMB file that you're currently viewing.

Print Preview...

The **Print Preview** option shows what the timecode trigger list printout will look like before you actually print it.

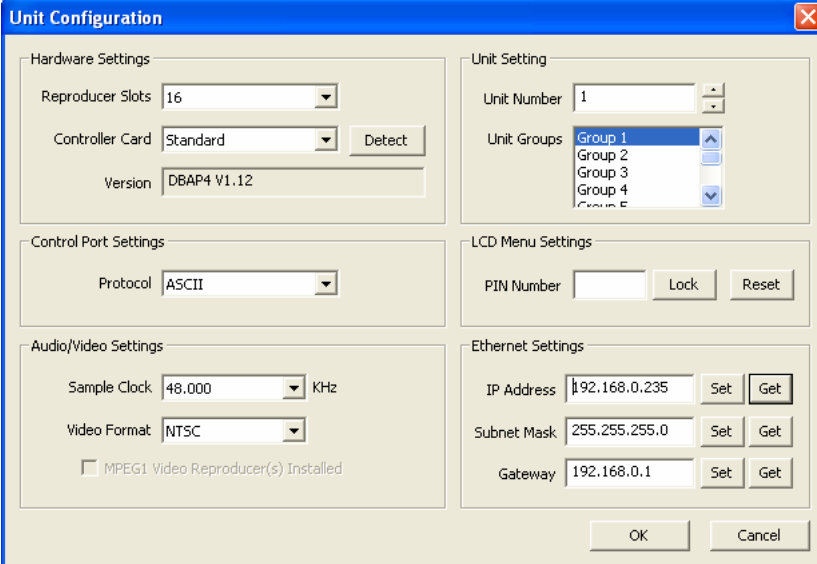
Exit

The final option in the **File** sub-menu is **Exit**. This allows you to exit the program and returns you to Windows. If you have made any changes to the settings, you will be asked if you want to update the unit and if you want to save the file with the current settings before you exit the Binloop application.

Binloop Menu

The next selection in the main window is *Binloop*. It contains two groups containing selections called *Settings...*, *SMPTE...*, *Reproducer*, and *Download to Binloop* and *Upload to Binloop*. Each of these selections is explained in what follows.

Settings...Unit...

The image shows a 'Unit Configuration' dialog box with a blue title bar and a close button. It is divided into several sections: 'Hardware Settings' with 'Reproducer Slots' (16), 'Controller Card' (Standard), a 'Detect' button, and 'Version' (DBAP4 V1.12); 'Unit Setting' with 'Unit Number' (1) and a list of 'Unit Groups' (Group 1 to Group 5); 'Control Port Settings' with 'Protocol' (ASCII); 'LCD Menu Settings' with a 'PIN Number' field, 'Lock', and 'Reset' buttons; 'Audio/Video Settings' with 'Sample Clock' (48,000 KHz), 'Video Format' (NTSC), and a checkbox for 'MPEG1 Video Reproducer(s) Installed'; and 'Ethernet Settings' with 'IP Address' (192.168.0.235), 'Subnet Mask' (255.255.255.0), and 'Gateway' (192.168.0.1), each with 'Set' and 'Get' buttons. 'OK' and 'Cancel' buttons are at the bottom right.

Reproducer Card Slots

Many previous generations of Binloop products are supported by the Binloop Software.

The Unit Configuration window permits the user to select the number of reproducers in the Binloop. The Digital Binloop and Video Binloop both have 16 slots.

Control Port Protocol

The *Control Port Protocol* determines the protocol that is used on the Show Control Port. This settings can be either *Alcorn Show Control*(Alcorn 9-bit), *MIDI*, *ASCII* (Pioneer/AMI), and *Debug*. For more information on these protocols, please refer to the **Control Protocols** section of this manual.

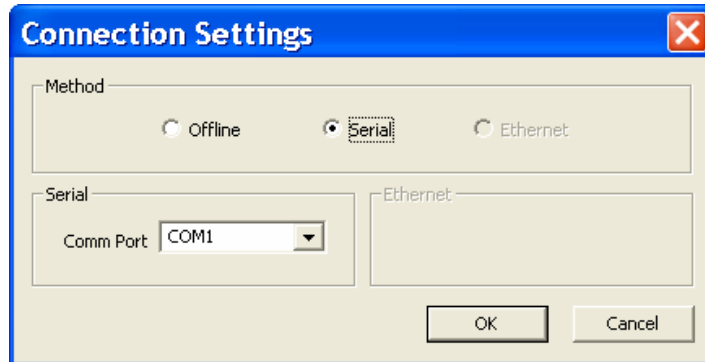
Unit Number

The *Unit Number* is the device address used for the MIDI protocol and when addressing multiple units.

Unit Groups

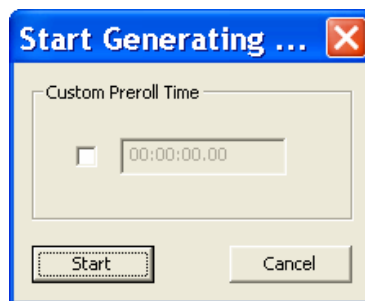
You can place the Binloop into a Group with other Binloops and address them as a single group. The *Unit Groups* selects which groups this Binloop is a member of.

Settings...Connection...



This Connection Settings window provides settings for the serial COM port your using on your PC to communicate to the Binloop. It defaults to *Offline*.

SMPTE...Start...



You can also set a Custom Preroll Time where SMPTE Generation can begin earlier the first time it is started.

This window presents you with a Start button for starting the SMPTE interface. This window will change depending on whether you have the SMPTE interface configured to Generate or Read SMPTE.

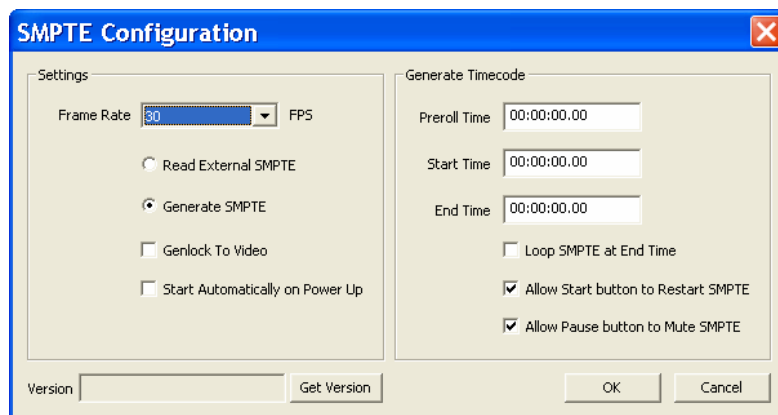
SMPTE Stop...

This option stops the SMPTE interface as well as the reproducers. A confirmation window appears with *OK* and *Cancel*.



SMPTE Configuration...

The SMPTE Configuration window allows you to set all of the parameters associated with the Binloop's SMPTE interface.



Frame Rate

This is the Frame Rate used for generating SMPTE. The choices are 23.976, 24, 25, 29.97, 30-drop, and 30 frames-per-second.

Read External SMPTE

This button configures the SMPTE interface to Read incoming SMPTE from the SMPTE In connector.

Generate SMPTE

This button configures the SMPTE interface to Generate timecode. This timecode is used internally to process the timecode triggers, and is also fed to the SMPTE Out connector.

Genlock to Video

Only 23.976, 25 (PAL) and 29.97(NTSC) frame rates are supported when in Genlock to Video mode.

Since the A/V clock interface of the Binloop can lock to an external V-Sync source, this means that you can generate SMPTE synchronously with your video sync source as well.

When this option is selected, the SMPTE interface will generate timecode synchronously with the internal V-Sync clock of the Binloop. This ensures that the SMPTE clock and reproducers operate at the same exact rate.

Since the SMPTE clock is used as a timing reference for timecode triggers, it's recommended that this option is enabled whenever you are synchronizing multiple video tracks. This will ensure that the reproducers start reliably in sync with one another.

Start Automatically on Power UP

If this checkbox is active the Binloop will begin Generating or Reading timecode as soon as the system reaches the 'Ready' state.

Preroll Time

This is the timecode value at which the SMPTE generator will begin when first started.

Start Time

This is the time at which SMPTE timecode will loop back to when it is looping, or when a restart condition has occurred.

End Time

This is the time at which SMPTE will stop or loop (depending on the Loop setting).

Loop SMPTE at End Time

If this option is enabled, then the SMPTE generator will loop back to the *Start Time* when the *End Time* is reached.

Allow Start Input to Restart SMPTE

When this option is enabled, a SMPTE Start command from any control interface will cause the SMPTE generator to jump back to the *Start Time*.

SMPTE Muted When Stopped or Paused

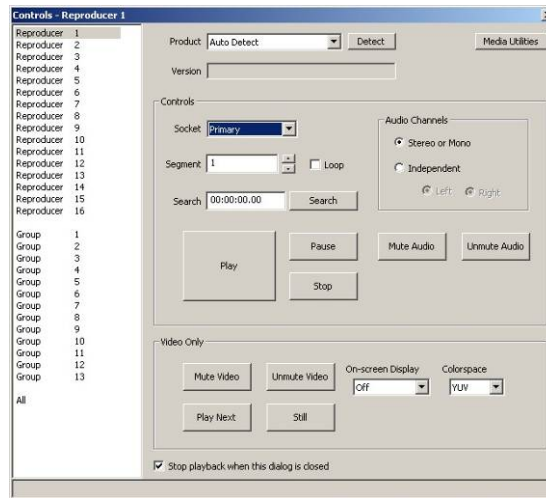
When SMPTE is paused or stopped, SMPTE output will be muted when this checkbox is active. If this checkbox is inactive, the current frame will be repeated over and over again when timecode is paused or stopped.

Version – Get Version

This allows you to get the current firmware version of the SMPTE interface on your Binloop's Controller card.

Reproducer...Controls...

When you select this menu item, the Reproducer Controls window appears.



This window presents many controls that are good for playback testing and troubleshooting. Each is explained below. The next menu item allows you to select an individual Reproducer to control.

Socket

This setting is intended only for legacy Binloop systems with dual flash sockets, and does not apply to the modern Digital Binloop or Video Binloop. This setting should be left at the default value of 'Primary'.

Segment

*Please refer to the **Media Files** section of this manual for the correct way to name and number files.*

The Segment refers to the particular video or audio file that you want to play. It is a number referring to the file number on the media.

Search

You can *Search* a video file so that when you click *Play* it will begin playing immediately. Entering an time in the adjacent box will search to an offset within the video or audio file. The default value of 00:00:00.00 will search to the beginning of the clip.

Audio Channels

This setting is used for audio-only playback from the left or right channel. If you use the default setting of *Stereo or Mono*, audio and video clips play normally on both channels (mono clips are duplicated on the left and right audio outputs). However, if you wish to play a mono audio clip on only one of the audio outputs, you may select the *Independent* option.

Colorspace

RGB is actually RGsB /w sync on green.

This setting allows you to configure the component video output *Colorspace* as either *RGB* or *YUV*.

Play

This will cause the selected Reproducer to *Play* the selected file segment. Clicking *Play* again will start playback over from the beginning.

Play Next

If a file is currently playing, clicking the *Play Next* button will queue the specified file segment to play immediately after the current file is finished playing.

Pause

This causes the currently playing file to stop. Clicking *Play* will cause it to continue playback from where it was paused.

Mute Audio, Mute Video

These buttons cause the audio and video to be muted, respectively. When video is muted, the video outputs are black.

Un-Mute Audio, Un-Mute Video

These buttons cause the audio and video to be un-muted, respectively.

Media Utilities

Important: You should always Format the media in the Binloop before you use it.

This button accesses another window that allows you to Format the compact flash media on the selected Reproducer. Formatting will erase the contents of the flash card.

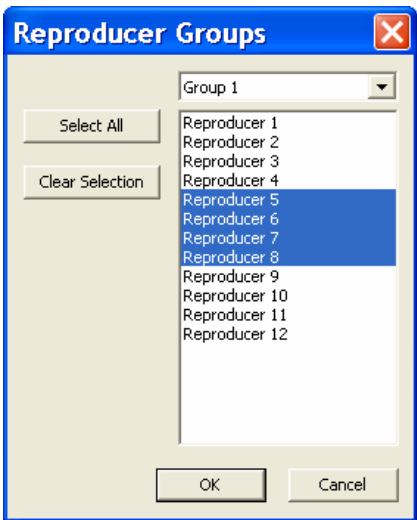


Reproducer Groups

Note: By grouping the reproducers together, Reproducers can be simultaneously commanded using any of the Binloop's control interfaces.

Reproducers can be assigned to Groups using this selection. A Reproducer that is a member of a Group will respond simultaneously with other members to any command that is addressed to the Group. This provides the ability for one Play command to trigger synchronized multi-channel video playback.

You can assign a Reproducer to be a member of more than one Group. Below is a screenshot showing the Group assignment window. This example assigns Reproducers 5, 6, 7, and 8 to be members of Group 1.

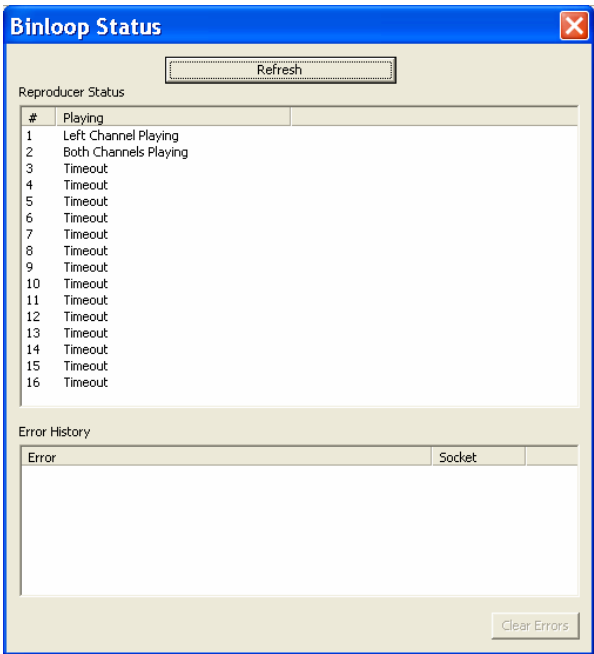


Status...

You can determine that current playing status of each Reproducer by opening the *Status* window. When this window opens, select *Refresh* and after a brief moment each Reproducer will report their current status.

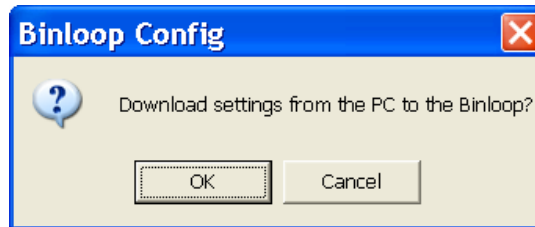
A timeout message appears if no Reproducer is installed in the slot.

Any errors that exist will be reported in the lower part of this window.



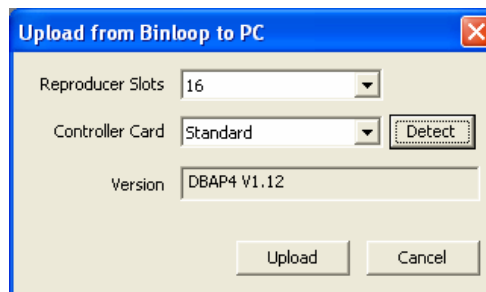
Download Configuration ...

This option allows you to download the configuration you are currently viewing. This includes all of the SMPTE interfaces settings, unit configuration settings, as well as any programmed timecode triggers. A confirmation window will appear.



Upload Configuration

This option allows you to retrieve the configuration from a Binloop, view it within the Binloop software. When you click the Upload Configuration button you will be presented with a window asking you to select your Binloop and Reproducer type.



View Menu

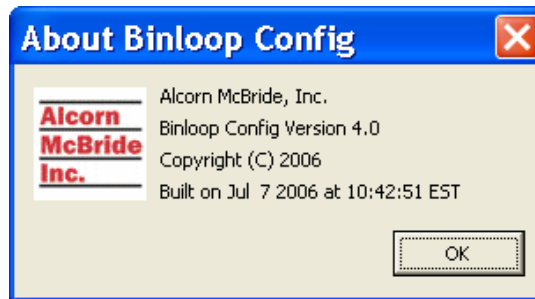
The *View* button allows you to turn on or off the Tool Bar at the top of the main window, or the Status Bar at the bottom.

Window Menu

The *Window* button is a standard Windows control that allows you to open a *New Window* for Triggers, *Cascade* or *Tile* windows, and *Arrange Icons*.

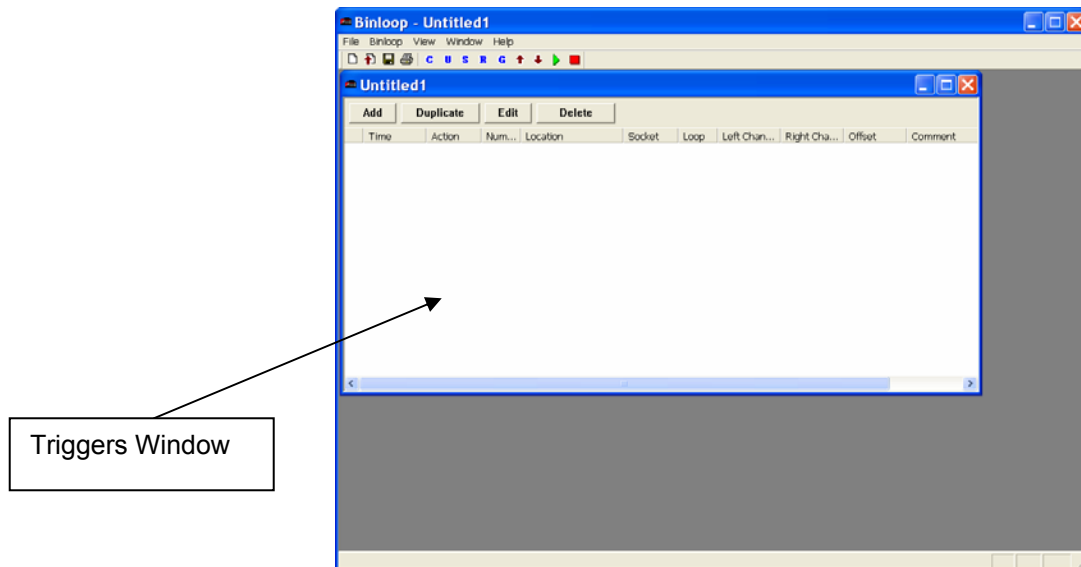
Help Menu

The **Help** sub-menu shows a window containing current revision information.



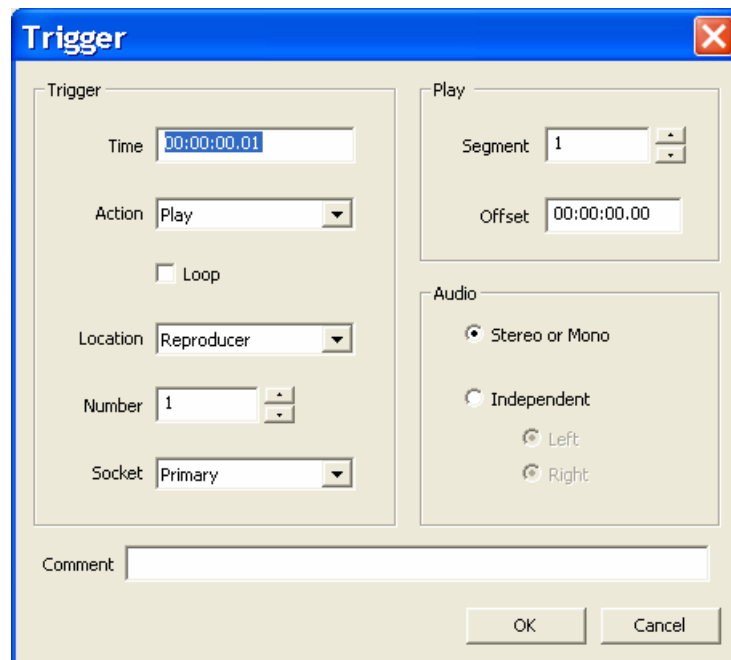
Timecode Trigger List

The Timecode Trigger List is the portion of the Binloop software that allows you to program precisely timed commands that are triggered based on the Binloop's timecode clock. It has a row of buttons at the top that allow you to **Add**, **Duplicate**, **Edit**, or **Delete** triggers.



Add

Clicking **Add** brings up a window that allows you to add a trigger.



Time

This is the timecode value at which you want the trigger to occur.

Action

This is the action (command) you want to have happen at the trigger *Time*. Available selections are *Play*, *Play Next*, *Pause*, *Still*, *Stop*, *Search*, *Start Sequence*, and *Set Variable*.

Action	Description
Play	Plays a file (Synchronous Play). Also used to resume from a search, still, or pause. (segment number is ignored when used to resume)
Play Next	Queue a file to play when the current clip is finished.
Pause	Pause video and audio playback, and mute video output
Still	Pause video and audio playback, still frame remains on-screen.
Stop	Stop playback
Search	Prepare a video for playback, first frame appears on-screen when search is complete.
Start Sequence	Send a Start Sequence command to an Alcorn McBride Show Controller out of the Show Control Port. (Show Control Port must be set to Alcorn 9-bit)
Set Variable	Send a Set Variable command to an Alcorn McBride Show Controller out of the Show Control Port. (Show Control Port must be set to Alcorn 9-bit)

Loop

Check this box if you want the Play trigger to be a Play-and-Loop command where playback will loop back to the beginning of the clip after the clip ends.

Location

You can have the *Action* apply either to an individual *Reproducer*, a *Reproducer Group*, or *All Reproducers*.

Number

This is either the *Number* of the individual *Reproducer* or *Reproducer Group* selected in the *Location*. This field is disabled when *All Reproducers* is selected.

Socket

This field is used for older Binloop systems that have multiple flash memory sockets. This value is not relevant for the modern Binloop.

Segment

This is the number of the video clip that you want to play (for Play triggers).

Offset

This field specifies a starting offset into a video or audio file. For example, entering 00:00:05.00 would skip the first 5 seconds of the file. If you use this feature with a video, keep in mind that it can take a significant amount of time for the reproducer to locate the correct point within the file. For that reason, you should always use the 'Search' command, and you should always allow for adequate search time before sending the 'Play' command in synchronous applications.

Stereo or Mono versus Independent

This setting is used for audio-only playback from the left or right channel. If you use the default setting of *Stereo or Mono*, audio and video clips play normally on both channels (mono clips are duplicated on the left and right audio outputs). However, if you wish to play a mono audio clip on only one of the audio outputs, you may select the *Independent* option.

A Word About Trigger Latency:

The Binloop uses two different types of Play commands. The first one, called “Unsynchronized Play” is used mostly for “point-source” or other audio/video applications where synchronization between channels is not important.

The other type of Play command is called “Synchronous Play”. This command needs to be used in those situations where absolute frame synchronization is required across channels. This is the type of play command issued by the *Play* action. There are important differences between how the Binloop reacts to these two commands.

Unsynchronized Play Latency

When this type of play command is sent to a reproducer, it starts playing the file as soon as it possibly can. The latency is determined only by the amount of time it takes the reproducer to process the play command, load the clip from the compact flash card, and begin decoding and playing the file. This latency is very short, but it is not predictable. For that reason, this command is not recommended for use when the goal is to synchronize playback between multiple reproducer cards. The unsynchronized play command is normally used for applications that require multiple channels of video that are unrelated to one another (like individual kiosks, etc.).

Synchronized Play Latency

The exact amount of latency time will depend on the SMPTE frame rate being used.

When using Play commands from the Timecode trigger list, it is highly recommended that the SMPTE clock is locked to V-Sync. This ensures that the timecode framerate and the video framerate of the reproducers is synchronous.

A Synchronized Play is the same as an Unsynchronized Play with one key difference. Rather than playing the video as quickly as possible, the reproducer will wait a consistent amount of time after the play command was received before playback starts. This ensures that every reproducer that received this command starts playing at the same time.

The default latency between when the play command occurs and when the file actually starts playing is exactly **16 frames** for a **Synchronized Play**. This delay provides the reproducer with more than adequate time to process the command and load the video from the compact flash card so that it's ready to start playing when the consistent wait period is over.

Use the Synchronized Play command for playing groups of Reproducers, which must start playback in perfect synchronization (i.e. multi-channel music tracks or video wall presentations).

CobraNet

Description

CobraNet is an industry standard interface for distributing multiple channels of audio over an Ethernet based network. This is useful for both transmitting audio over long distances, as well as easily interfacing to CobraNet compatible systems using a simple Ethernet cable rather than many individual analog connections.

Details

CobraNet is digitally encoded audio over Ethernet, and as such it does not suffer the effects of long distance analog transmission issues. It is not within the scope of this manual to provide a detailed installation and implementation of CobraNet; however, feel free to contact Alcorn McBride for additional details.

CobraNet uses standard Ethernet packets and network infrastructure (controllers, hubs, repeaters, switches, cabling, etc.).

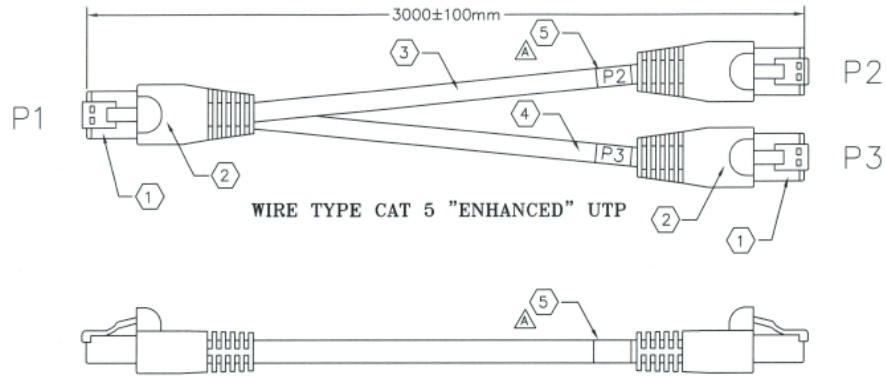
CobraNet "Y" Cable

There are two, separate CobraNet interface components to the CobraNet card. The first set of Binloop reproducers (1-8) are on one of the CobraNet interfaces. The second set of reproducers (9-16) are connected to the second, slave CobraNet interface.

These two, separate CobraNet interfaces are merged into a single Ethernet jack on the rear of the Binloop. Reproducers 1-8 are available only on the 1st half of the "Y" cable labeled "2". Reproducers 9-16 are available on the second half of the cable labeled "3".

The reason for this is that it allows 24bit audio for all 16 channels in a single CobraNet interface. CobraNet interfaces supports 7 channels of audio at 24bit rates per bundle for a maximum of 4 bundles. The CobraNet interface does support 8 channels of 20bit audio per bundle.

A diagram and pin out of the cable is shown in the following diagram.



PINOUT CONFIGURATION

P3 RJ45 8P8C	P1 RJ45 8P8C	P2 RJ45 8P8C	WIRE COLOR CODE
	1	1	WHITE /ORANGE
	2	2	ORANGE
	3	3	WHITE /GREEN
	6	6	GREEN
6	4		BLUE
3	5		WHITE /BLUE
1	7		WHITE /BROWN
2	8		BROWN

Note: the second, slave CobraNet interface for reproducers 9-16 must be connected to the same CobraNet network as the first, master CobraNet interface for reproducers 1-8 for proper operation.

Reproducer Configuration

Edit CobraNet Reproducer Assignment

Reproducer 1

Left Channel

Bundle Number 256

Bundle Channel 1 [current]

Right Channel

Bundle Number 256

Bundle Channel 2 [current]

OK Cancel

CobraNet Configuration

The image shows the 'Binloop CobraNet Configuration' window. At the top, the 'Version' field displays 'Cobranet Module v 1.0.0' with a 'Detect' button to its right. Below this, a checkbox labeled 'CobraNet Module Installed' is checked. Further down, there are three dropdown menus: 'Resolution' set to '20 bit', 'Latency' set to '5-1/3 ms', and 'Conductor Priority' set to 'Never Conduct'. The main area is divided into two sections for 'Reproducers 1 - 8' and 'Reproducers 9 - 16'. Each section contains an 'IP Address' field (192.168.0.252 and 192.168.0.253 respectively) and a 'Bundles' table. The 'Reproducer Assignment' table on the right lists 16 reproducers with their 'Left' and 'Right' channel assignments. At the bottom, there are buttons for 'Advanced...', 'Upload', 'Download', and 'OK'.

Version: Cobranet Module v 1.0.0 [Detect]

☒ CobraNet Module Installed

Resolution: 20 bit Latency: 5-1/3 ms Conductor Priority: Never Conduct

Reproducers 1 - 8

IP Address: 192.168.0.252

Bundles

Bundle Number	Channels Assigned
256	8
257	8
258	0
259	0

Edit

Reproducers 9 - 16

IP Address: 192.168.0.253

Bundles

Bundle Number	Channels Assigned
260	8
261	8
262	0
263	0

Edit

Reproducer Assignment [Bundle:Channel]

Repro Number	Left	Right
1	256:1	256:2
2	256:3	256:4
3	256:5	256:6
4	256:7	256:8
5	257:1	257:2
6	257:3	257:4
7	257:5	257:6
8	257:7	257:8
9	260:1	260:2
10	260:3	260:4
11	260:5	260:6
12	260:7	260:8
13	261:1	261:2
14	261:3	261:4
15	261:5	261:6
16	261:7	261:8

Edit

Advanced... Upload Download OK

Detection – The configuration process starts by locating the CobraNet hardware. Upon locating the CobraNet interface the hardware will report back the revision number for identification purposes.

CobraNet Advanced Configuration

Binloop CobraNet Advanced Configuration

☒ CobraNet Chip 1 ☐ CobraNet Chip 2

Settings

IP Address: 192.168.0.252 Resolution: 20 bit Latency: 5-1/3 ms Conductor Priority: Never Conduct

Bundle 1

Number: 256

Channel	Repro	Left / Right
1	1	Left
2	1	Right
3	2	Left
4	2	Right
5	3	Left
6	3	Right
7	4	Left
8	4	Right

Bundle 2

Number: 257

Channel	Repro	Left / Right
1	5	Left
2	5	Right
3	6	Left
4	6	Right
5	7	Left
6	7	Right
7	8	Left
8	8	Right

Bundle 3

Number: 258

Channel	Repro	Left / Right
1		
2		
3		
4		
5		
6		
7		
8		

Bundle 4

Number: 259

Channel	Repro	Left / Right
1		
2		
3		
4		
5		
6		
7		
8		

OK

CobraNet Clip -

IP Address – This is the network address, which is used to identify the source of the session. The function of this address is the same as any other IP address used on the network

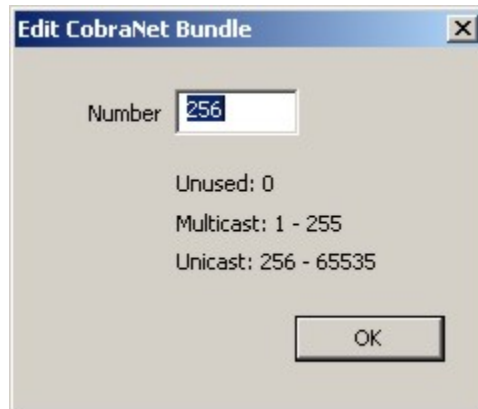
Resolution – This is the encoded resolution of the audio source. 20 or 24 bits are the standard options

Latency – This option allows the operator to set the maximum delay tolerated in the communications channel

Conductor Priority – This establishes the point of control for the communications session.

Bundle Assignments – The bundle number is assigned to the communications channel as well as the audio channels to be carried with in the bundle. Note: Left and right stereo channels are identified as individual channels in a bundle.

Bundle



The “Bundle” is the term used for the network channel in which the sending device will connect with the receiving unit. Many audio channels can be transmitted within a bundle. Bundle numbers 0-255 are defined as “Multicast” point-to-many channels and bundles 256-65279 are “Unicast” or point-to-point transmissions. For additional information on CobraNet, contact Cirrus Logic support at the link below:

<http://www.cirrus.com/en/support/cobranet/>

Hardware Description

Controls and Indicators

The Binloop is normally controlled either internally using a configuration loaded from our Binloop software, or remotely from an external control system. However, the product does have several controls and status indicators located on the front panel.

Unit Status LED's

On the front of the Binloop are nine LED's corresponding to the status of the unit. Below is a brief description of their meaning:



Power (Green) - Power is being applied to the unit.

Fault (Yellow) - The unit has detected a fault condition.

VSynC (Red) - The unit has detected an external video sync source and is locked to it.

Gen (Red) - The unit is actively generating SMPTE timecode.

Ready (Red) - The unit is ready to accept commands. This LED should almost always be on.

Run (Red) – This indicator is illuminated whenever the SMPTE output is active.

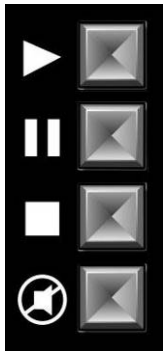
Pause (Red) - This LED is illuminated during SMPTE generation when the Pause button has been pressed and a Pause at the SMPTE End or Loop Time is pending.

Stop (Red) - This is illuminated whenever the SMPTE Generator/Reader is idling and ready to begin SMPTE generation or external SMPTE synchronization.

Mute (Red) - This LED is illuminated when the Mute button on the unit has been pressed. This indicates that the audio outputs of all reproducers are muted.

Control Buttons

Directly beneath the Unit Status LED's, there are four control buttons. Their functions are defined briefly here but they will be discussed in more detail over the course of the next two chapters:



▶ Start - If the Binloop is programmed to generate SMPTE, pushing this button starts the generation of timecode. If the Binloop is programmed to read external timecode, this causes the unit to start listening for external timecode. If SMPTE is already being generated, this button can be optionally configured to restart the SMPTE clock.

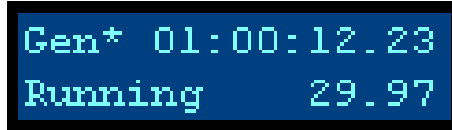
⏸ Pause – This button is only applicable when the Binloop is configured to generate and loop SMPTE. Pushing this button causes the SMPTE generation to pause the next time the configured End Time (the loop point) is reached. After this button is pressed, the Pause status LED will illuminate.

■ Reset – If the Binloop is configured to generate timecode, pushing this button halts the generation. If the Binloop is configured to read external timecode, this button causes the Binloop to ignore any incoming timecode. In addition, pressing this button stops playback on all reproducers.

🔇 Mute – Pressing this button toggles between muting and unmuting the audio outputs of all reproducers. When this button is in the mute position, the reproducer status LED's will be red to indicate that they are muting the audio output. The Mute status LED will also be illuminated in this case.

Display and Menu Wheel

The Binloop is equipped with a 2x16 character Vacuum-Fluorescent Display. This display is primarily used to indicate the current status of the Binloop. Here's an example of that status display:



```
Gen* 01:00:12.23
Running 29.97
```

The current SMPTE mode is indicated at the top-left of this screen. The 3 possible modes are displayed as **Gen**, **Gen*** (Generate /w VSync), and **Read**. The current state of the SMPTE generator/reader is displayed at the bottom left, and the SMPTE frame rate is displayed at the bottom right. The time at the top-right of the display represents the current SMPTE timecode value in hours, minutes, seconds, and frames. In the event that there is no current timecode value (meaning timecode is not currently being read or generated), this time will be displayed as dashes like the following example:



```
Gen --:--:--.--
Idle 29.97
```

In addition to providing system status, the display can also be used in conjunction with the nearby Menu Wheel to access the configuration settings of the Binloop. To enter this menu, simply press in the menu wheel.



```
---MAIN MENU---
Unit Setup >
```

Note: The menu wheel 'clicks' when rotated to provide tactile feedback between selections.

After you do this, you can navigate through the various settings of the Binloop by rotating the menu wheel to change the selection, and pressing it to select the current item. When you are finished, simply browse to the 'Exit' selection and push the menu wheel to return to the status screen. To see a complete map of this menu interface, refer to **Appendix D - Wheel Menu Map**.

Reproducer Status LED

Each reproducer has a single multi-color LED located directly above the Compact Flash socket. The color and blink pattern of this LED signifies the current state of the reproducer. Here's a table of those states and the corresponding LED behavior:

LED State	Reproducer State
Off	Stopped
Green	Playing
Orange	Paused, Stilled, or Searched
Red	Audio Muted
Flashing Red	Fault
Flashing Green	Power-on Initialization, Searching

Reproducer Test Button

Each reproducer has a test button located directly above the Compact Flash socket. As you can probably tell from the name, this button is intended only for testing purposes and not for normal operational control. When this button is pressed, the reproducer reacts differently depending on its current state.

Note: File Number 1 is VID00001.XXX or SND00001.XXX. Please refer to the **Media Files** chapter in this manual for further details on file naming.

1. If the reproducer is stopped, pressing this button will cause the lowest numbered file to start playing.
2. If the reproducer is playing, pressing this button will cause it to stop playing.
3. If the reproducer is in a fault condition (status LED flashing red), pressing this button will clear the fault.

Connectors

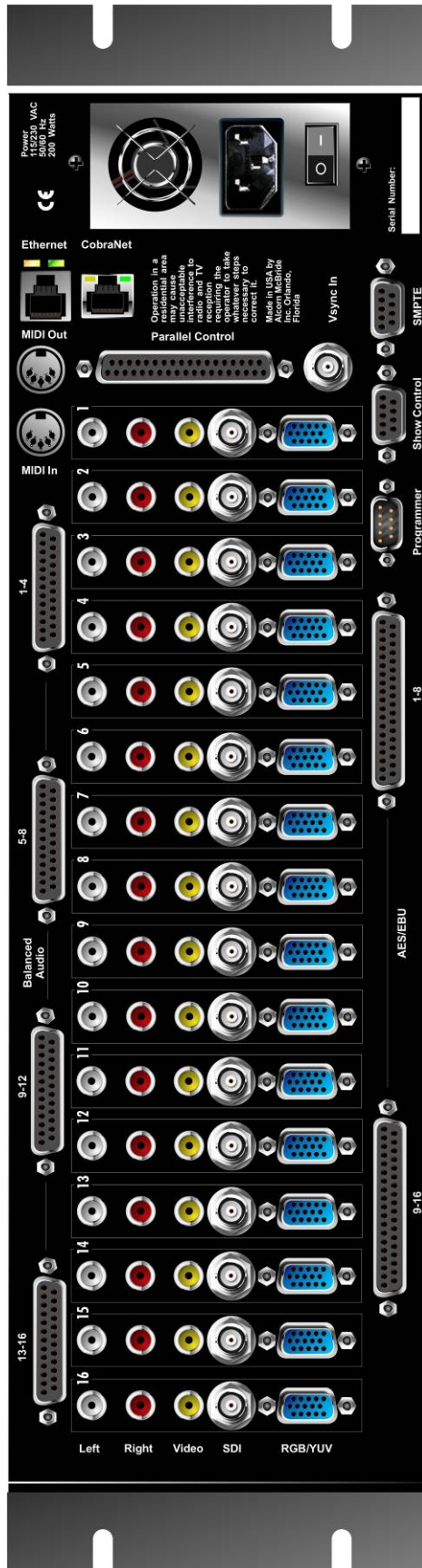
This table lists all connectors on the Binloop Card Cage and their related function. The connectors are all located on the rear of the Binloop. If the connector is a single signal, then the signal is listed. If the connector contains multiple signals, the connector's pins are identified in further tables.

Connector Summary

Connector	Type	Function
Video	RCA/F (Yellow) x 16	Composite Video Outputs
Component	HD-15 x 16	Component Video outputs (YUV/RGB)
*SDI	BNC Female x 16	SMPTE 259M SDI Digital Video Outputs
Unbalanced Audio	RCA/F (Red-Wht) x 32	Unbalanced Stereo Audio Outputs (Right-Left)
Show Control	DB-9/M	RS-232 to Control System
Programmer Port	DB-9/M	RS-232 to Programmer
VSYNC	BNC Female	Blackburst Video Sync Input
SMPTE	DB9/F	SMPTE Input/Output
MIDI In	DIN5/F	MIDI In
MIDI Out	DIN5/F	MIDI Out
Parallel Control	DB37/F	Control and I/O
Balanced Outputs 1-4, 5-8, 9-12, 13-16	DB25/F x 4	Balanced Stereo Audio Outputs
AES/EBU & S/PDIF I/O	DB37/F x 2	AES/EBU or S/PDIF Digital Audio I/O
**CobraNet	RJ-45	CobraNet Audio Network
Ethernet	RJ-45	Ethernet Control

** Although the SDI connectors are always present on the back of the Binloop, they are not active unless the Binloop's reproducers are equipped with the **optional** SDI upgrade.*

*** The CobraNet connector is only available if the **optional** CobraNet Card is installed in the Binloop.*



Digital Binloop and Video Binloop Rear Panel

Digital Audio (AES/EBU)

There are two connectors that supply the AES/EBU, S/PDIF, or AC3 digital audio data for the Reproducers 1 – 16.

AES/EBU 1-8

Reproducer	Output
1	1(+), 20(-)
2	2(+), 21(-)
3	3(+), 22(-)
4	4(+), 23(-)
5	5(+), 24(-)
6	6(+), 25(-)
7	7(+), 26(-)
8	8(+), 27(-)

AES/EBU 9-16

Reproducer	Output
9	1(+), 20(-)
10	2(+), 21(-)
11	3(+), 22(-)
12	4(+), 23(-)
13	5(+), 24(-)
14	6(+), 25(-)
15	7(+), 26(-)
16	8(+), 27(-)

Balanced Audio

Each of these connectors contains balanced analog audio outputs for groups of four Reproducers. Refer to the table below.

Balanced Output #	Reproducer Slot	Left		Right		Analog Gnd
		+	-	+	-	
1 - 4	1	1	2	14	15	3, 16
	2	4	5	17	18	6, 19
	3	7	8	20	21	9, 22
	4	10	11	23	24	12, 25
5 - 8	5	1	2	14	15	3, 16
	6	4	5	17	18	6, 19
	7	7	8	20	21	9, 22
	8	10	11	23	24	12, 25
9 - 12	9	1	2	14	15	3, 16
	10	4	5	17	18	6, 19
	11	7	8	20	21	9, 22
	12	10	11	23	24	12, 25
13 - 16	13	1	2	14	15	3, 16
	14	4	5	17	18	6, 19
	15	7	8	20	21	9, 22
	16	10	11	23	24	12, 25

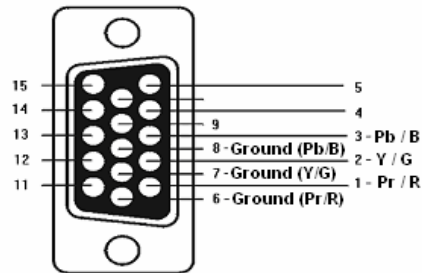
Unbalanced Audio and Composite Video

Sixteen groups of RCA connectors provide stereo unbalanced analog audio, and composite video outputs from each reproducer. They are color-coded and labeled from the top of the cage down, in the following order:

Color	Signal
White	Left Unbalanced Audio
Red	Right Unbalanced Audio
Yellow	Composite Video

Analog Component Video

The component video outputs can be configured to output in either the RGB or YPbPr colorspace. The composite video (CVBS) connector is always active no matter how the component video output is configured.



Show Control Port

The Show Control serial port is intended for applications where the Binloop is to be controlled remotely via an RS-232 capable control system. This could be an Alcorn McBride show controller, a PC, or practically any device with an RS-232 serial port. Since this interface is designed to be flexible, it supports a variety of different serial protocols. These protocols include **Alcorn 9-bit**, **Pioneer/AMI**, and **MIDI**. For more information on how to use the serial control interface, please refer to the section titled **Controlling the Binloop**.

Pin	Signal
2	RXD
3	TXD
5	GND
1, 4, 6 – 9	Not Used

Programmer Port

The Programmer serial port is specifically intended to connect with a PC serial port for configuring and programming the Binloop using the Binloop Configuration software. This connection should be made with a straight-through (not null) 9-pin serial cable, which is included with the Binloop.

Pin	Signal
2	TXD
3	RXD
5	GND
1, 4, 6 – 9	Not Used

Ethernet

The Ethernet jack of the Binloop is intended to connect the Binloop to a 10/100 Ethernet network. This interface can be used by the Binloop Configuration software to control the Binloop as well as download new configurations. For other methods of Ethernet control, this interface also provides a simple control interface that uses the Pioneer/AMI protocol over the common Ethernet protocol called UDP. For more information on how to use the Ethernet interface of the Binloop, please read the section of this manual called **Controlling the Binloop**.

MIDI In

The MIDI In connector accepts standard MIDI protocol commands for control of the Binloop. Please refer to the section titled **Controlling the Binloop** for details on how to use the MIDI port.

Pin	Signal
1	N/C
2	GND
3	N/C
4	MIDI In +
5	MIDI In -

MIDI Out

The MIDI Out connector repeats any input received through the MIDI In connector. For more information on the MIDI control and Timecode control interfaces, please refer to the **Controlling the Binloop** section of this manual.

Pin	Signal
1	N/C
2	GND
3	N/C
4	MIDI Out -
5	MIDI Out +

V-Sync

The V-Sync connector is a female BNC connector that is designed to accept a blackburst video sync signal. When a sync signal is applied to this connector, the A/V system of the Binloop will use this as a reference for all audio and video generated within the Binloop. For more details about using external video sync with the Binloop, refer to the **A/V Sync System** chapter in the **About the Binloop** section of this manual.

SMPTE

The SMPTE connector provides both an input and output for the SMPTE interface of the Binloop. The balanced input is 600 ohm transformer-isolated and will accept levels between 1- 5 Vpp. The balanced output level defaults to 0dBV into 150 ohms, but can be adjusted between 0-4dBV using the front-panel menu of the Binloop. For backward compatibility with previous Binloop systems, an alternate V-Sync input is also provided on this connector.

Pin	Signal
1	SMPTEIN +
2	SMPTEIN -
3	N/C
4	VSYN CIN
5	GND
6	N/C
7	N/C
8	SMPTEOUT-
9	SMPTEOUT+

Parallel Control

This connector contains various inputs and outputs that can be used to control and monitor the Binloop. Pins 7-14 are optically-isolated inputs designed to trigger on 24V. Pins 20-37 are designed to accept a contact closure between the specified pin and a GND pin (16, 17, 18, or 19).

Pin	Signal
1	SMPTE Running output (Contact 1)
2	SMPTE Running output (Contact 2)
3	Fault output (Contact 1)
4	Fault output (Contact 2)
5	Ready output (Contact 1)
6	Ready output (Contact 2)
7	SMPTE Start input (+)
8	SMPTE Start input (-)
9	SMPTE Pause input (+)
10	SMPTE Pause input (-)
11	SMPTE Stop input (+)
12	SMPTE Stop input (-)
13	Mute input (+)
14	Mute input (-)
15	VCC (+5VDC) supplied by Binloop power supply
16	GND
17	GND
18	GND
19	GND
20	Play and Loop clip 1 of group 1.
21	Play and Loop clip 2 of group 2.
22	Play and Loop clip 3 of group 3.
23	Play and Loop clip 4 of group 4.
24	Play and Loop clip 5 of group 5.
25	Play and Loop clip 6 of group 6
26	Play and Loop clip 7 of group 7.
27	Play and Loop clip 8 of group 8.
28	Reset SMPTE.
29	Idle SMPTE (refresh)
30	Reserved
31	Stop sounds selected by inputs 20-27 (else play).
32	Inhibit the looping of sounds selected by inputs 20-27
33	High = SMPTE Locked, Low = not locked
34	High = Video Sync locked, Low = no sync detected
35-37	(not used)

Media Files

The reproducer cards within the Binloop support a variety of common industry-standard audio formats, as well as the popular MPEG-2 video format. To play properly in the Binloop, however, the files loaded onto the Compact Flash cards must adhere to some specific encoding and file-naming requirements. This section provides detailed information on those requirements so that you can successfully create and load content into the Binloop.

File Types

The possibilities of storing audio and video content in a digital file format are nearly endless, and it's impossible for the Binloop to support everything. Instead, the Binloop supports the specific file types which are listed below:

- **MPEG2 Program Stream** (Video and Audio)
- **Windows PCM - WAV** (Audio Only)
- **Apple AIFF** (Audio Only)
- **Raw PCM** (Audio Only)

The audio-only formats are easy to explain. They are simply audio files that are created using the associated uncompressed audio format. These files can hold up to 2 channels of audio information (Stereo), and can have a variety of other parameters which are discussed later in the **Audio Formats** portion of this section.

An MPEG2 Program Stream is a little more involved, but is necessary to play video in the Binloop. A Program Stream contains both a video and audio that are intended to play simultaneously. It can do this because both the Video track (Video Elementary Stream) and Audio track (Audio Elementary Stream) are combined (multiplexed) into single file. The Video portion of the stream must be a MPEG2 video that follows the specifications listed later in **Video Formats** portion of this section. The Audio stream contained within the program stream can be one of several formats, including MPEG-1 Layer 2 audio or Dolby Digital AC3. These types are covered in more detail later in **Audio Formats** portion of this section.

File Naming

A specific file naming convention is used so that the Binloop can identify the file type, as well as associate the file with a number. Here is a table that shows the naming conventions based on the type of file that is being named:

File Type	File Naming Style
MPEG2 Program Stream (Video+Audio)	VIDnnnnn.MPG
Windows PCM (Audio)	SNDnnnnn.WAV
Apple AIFF (Audio)	SNDnnnnn.AIF
Raw PCM 16-bit Mono (Audio)	SNDnnnnn.AMS
Raw PCM 16-bit Stereo (Audio)	SNDnnnnn.SFD
*Raw PCM 24-bit Mono (Audio)	SNDnnnnn.A24
*Raw PCM 24-bit Stereo (Audio)	SNDnnnnn.S24

** Only supported by the Repro24D (Digital Binloop)*

As you can see from this table, the files are identified based on the file extension. For example, the Binloop will identify a file with an extension of **MPG** as an MPEG2 Program Stream containing video and audio. A file with the extension of **SFD** would be treated as a 16-bit Mono Raw PCM audio file.

The file number is determined by the 5-digit 'nnnnn' portion of the file name. The Binloop supports files that are assigned numbers between 0 and 1023, so this part of the filename is always represented as **00000** through **01023**.

Just as an example, let's say you wanted to create a Program Stream (Video file) that you wanted to associate as file #123. To do this, you would need to rename the video file to VID00123.MPG.

Note: Every file on the media must have a unique number in the file name, no duplications are allowed. If you have duplicate file numbers (i.e. two files associated as file #1), the Binloop would consistently choose only one of those files when told to play the associated number.

Video Formats

As stated earlier, the reproducers in the Binloop support only one commonly used MPEG2 file format; the **MPEG2 Program Stream**. This MPEG2 Program stream MUST contain a video AND an audio track. Even if you do not wish to have audio for your application, you must at least have a blank audio track. For supported audio formats, see the **Audio Formats** section. The next section describes the supported parameters for the video portion of the Program Stream.

MPEG-2 Video Specifications

The decoder within the Binloop is compliant with Main Profile at Main Level MPEG-2 specified in ISO/IEC 13818. The following table summarizes the recommended encoding parameters for creating MPEG-2 video that is compatible with the Binloop:

Setting	Values
Profile	Main
Level	Main
Max Bitrate*	15Mbps (15000kbps)
Resolutions	480p, 480i – 720x480 576p, 576i – 720x576
Framerates**	NTSC Interlaced – 29.97fps NTSC Progressive – 29.97fps , 59.94fps PAL Interlaced – 25fps PAL Progressive – 25fps, 50fps
Aspect Ratios	16:9 & 4:3
Chroma Format	4:2:0

** Both Constant (CBR) and Variable (VBR) bitrate are supported, so long as the bitrate does not exceed the maximum.*

*** The framerate is very important when encoding for the Binloop, especially in synchronous playback applications. Only these 4 settings should ever be used. Which one of the 4 depends on whether the Binloop is configured for NTSC or PAL video systems, as well as the type of frame encoding used (interlaced or progressive).*

Audio Formats

The type of audio format that you choose to use first depends on one simple question: *Will you be using the Binloop to play only audio, or will you be playing video?* If you are only playing audio, you can create one of several types of uncompressed audio files. These file types are discussed in **Uncompressed Audio Formats**. If you are creating an MPEG2 Program stream for the purposes of playing video, you must include an audio track that is compatible with the Program Stream format. These formats are discussed in **MPEG2 Program Stream Audio Formats**.

MPEG2 Program Stream Audio Formats

When encoding video files as Program Streams, it is required that you have audio along with video. Here are the audio formats that you are permitted to use within the MPEG2 Program Stream:

1. **MPEG Audio**
2. **Dolby Digital Audio (AC3)**

MPEG Audio Specification

Parameter	Options
Format:	MPEG-1 Layer I/II, MPEG-2 Layer II
Mode:	Stereo, Mono
Sample Rate:	*32kHz, *44.1kHz, *48kHz
Max Bitrate:	384kbps
Emphasis:	Off

** For synchronization, the Binloop uses a global sample rate clock used throughout the entire system. This means that it is **ABSOLUTELY CRITICAL** that the sample rate used to encode files matches the sample rate configuration of the Binloop cage. A mismatch between the two will result in both audio and video playback issues.*

Dolby® Dolby Digital Specification

Parameter	Options
Format:	Dolby® Digital AC3*
Mode:	2.0, 5.1
Max Bitrate:	448kbps
Emphasis:	Off

** AC3 audio streams are not decoded in the Binloop. They are only bitstreamed from the digital audio output of the Binloop. This bitstream output must be decoded using an external AC3 decoder or A/V receiver.*

Uncompressed Audio Formats

When creating audio-only files for playback in the Binloop, there are several common formats to choose from. They are:

1. **Windows PCM (WAV)**
2. **Apple AIFF**
3. **Raw PCM**

The following table outlines the specifications for using any of these file formats:

Parameter	Options
Format:	WAV, AIFF, Raw PCM
Mode:	Stereo, Mono
*Sample Rate:	32kHz, 44.1kHz, 48kHz, **96kHz
Quantization:	16-bit, **24-bit

** For synchronization, the Binloop uses a global sample rate clock used throughout the entire system. This means that it is **ABSOLUTELY CRITICAL** that the sample rate used to encode files matches the sample rate configuration of the Binloop cage. A mismatch between the two will result in improper audio playback.*

*** These options are only supported by the **Repro24D**, which are typically installed in the **Digital Binloop** (not the Video Binloop).*

File System Limitations – 2GB File Size

The Binloop reproducers use a special implementation of the FAT32 file system. This file system is widely used, and will allow your flash cards to be accessible from any Windows based PC or MAC computer.

If the File Splitter software is no longer available, please obtain the latest manual from the Alcorn McBride website, or contact technical support for the latest recommended method for splitting files.

Although the supported *volume* size of Compact Flash cards is much larger, the maximum *file* size supported by the FAT32 implementation in the reproducers is 2GB. Your Windows PC will allow you to copy a larger file (up to 4GB) to the flash card, but the reproducer will not be able to access this file. This will result in a fault condition (Red flashing LED) if you attempt to play the file.

If your application requires a longer playback time than a 2GB limitation will allow, the best solution is 'split' the file into several segments using a file-splitting program. One suggestion is a software package called File Splitter, which can be downloaded at <http://www.filesplitter.org/>. Using this method, a large file can be separated into several segments which a reproducer can treat as a single coherent video file. Here's how it works.

The split-file playback feature requires reproducer firmware V1.64 or newer to work.

Let's say that you have encoded a 5GB video file (Program Stream) that you want to play in the Binloop. First, you will need to use a utility (like File Splitter) to divide this file into several chunks smaller than 2GB. To be safe, you could divide the file into 1GB chunks. When the splitting process is complete, you should have 5 files. Rename those files like this:

VID00001.MPG

VID00001.001

VID00001.002

VID00001.003

VID00001.004

Keep in mind that VID00001.MPG is the first segment, VID00001.001 is the second segment, and so on. When you are finished renaming the files, copy them to the reproducer's Compact Flash card.

At this point, all you have to do is treat VID00001.MPG as you normally would. For example, if you send a command to Play File #1, the reproducer will automatically play through all of the file chunks without you having to do anything special. You can also send a command to Loop File #1, and the entire file will loop from start to finish just as you would expect. If your application requires a longer playback time than a 2GB limitation will allow, the best solution is to create separate files and 'seam' them together using the 'Play Next' command.

Installation Guidelines

Grounding and System Performance

The connection of the Binloop into an audio system completes a gain stage that may pass through equalizers (EQs), mixers, and other low level processing gear before reaching the amplifiers. Once this connection is complete, any sources of noise in the system will be unmasked, and must be resolved in order to achieve optimum performance.

The Binloop itself has a muted signal to noise ratio of 120 dB. There should, therefore, be no audible hiss in the system once the connection is made and approximate amplification levels have been set. If any hiss is present, it is likely due to excessive gain at the low level end of the circuit. Since the Binloop is capable of generating very high audio output levels (up to +4 dBm into 600 ohms), it should be possible to restructure the gain stages to eliminate any boost prior to the inputs to the amplifiers. The use of the balanced outputs, rather than the unbalanced outputs provides twice the signal amplitude for any given material, and also isolates the signal from any common mode noise that may be present. Any hiss present only during playback, of course, must be a part of the recording itself.

Hum is a far more common problem than hiss, and is unfortunately a far more difficult problem to correct. Hum is fundamentally caused because the audio signal becomes contaminated or, more often, referenced to noise from the 60 Hz AC line. Improper grounding almost always causes this, but the question is: where? Hum may be caused by grounding errors present anywhere throughout the audio chain right up to the amplifiers, or even in the building wiring external to the audio system. A systematic approach is required to eliminate any ground loops and reference the entire audio system to a single, clean ground reference point.

All audio lines should be run as balanced signals, beginning with the Binloop and ending at the amplifiers. Neither the positive nor the negative side of any of these balanced signals should be tied to ground at any point along their journey. Additionally, it is essential that each of these signal runs be shielded cable, with the shield connected at only one end. This shield connection will normally be to the chassis of the connected audio equipment. The shielding scheme should be consistent throughout, for example with the receiving equipment providing the shield connection for each cable: The shield of the cable between the Binloop and the EQs tied at the EQs, the shield of the cable between the EQs and the Amps tied at the Amps.

The Binloop's rear panel Ground studs are connected to one another by a soldered-in-place jumper, but are, by design, not connected to the chassis or the ground pin of the power cord. This provides considerable flexibility in the grounding scheme. In many systems, the power supply will provide sufficient ground reference by itself. In others, a single shield may be terminated to one of the output connectors to provide a ground reference to the EQs. In other systems, the ideal ground point will be achieved by connecting a 16 gauge (or heavier) wire from the Binloop's analog ground stud to a point in the EQ or amplifier rack which may need to be determined experimentally: that is, by finding the point which nullifies any hum in the system.

And then there is the situation where all of this has been done and the system still hums. This points to an electrical wiring problem in the building. In many systems there can be dozens of circuits feeding the audio cabinets and amp racks. To guarantee the absence of hum, these circuits must be sourced from a shielded isolation transformer. This will provide two, or more often three, phases of clean power to be divvied up between the audio gear. There will also be multiple neutrals sourced from the center of the isolation transformer, and an isolated ground. It is essential that this isolated ground be just that, isolated. It should be referenced to the isolated neutral, and nothing else. Any connection to the building's mechanical ground will wreak havoc with the system. This connection can occur through such inadvertent means as tying the power to the cabinets through metal flex conduit, or tying mechanically grounded cable trays to the cabinets through Unistrut. Any measurement made across isolated ground and the isolated neutrals should read 0.00 VAC.

And finally, when all this fails, or becomes impossible to implement, connecting the Binloop's balanced outputs to the EQs through 1:1 600 ohm matching transformers will de-reference the low level audio from all grounds, eliminating any potential for problems in the electrical wiring to induce hum.

A Word on Temperature

The Binloop requires free air ventilation. All models are designed to operate between temperatures of 32 to 100 degrees F and 0 to 99% non-condensing relative humidity. Like all other electrical equipment, this means that an air-conditioned equipment room is the optimal environment. Temperatures can easily exceed the high limit if the Binloop is mounted without proper ventilation or mounted directly above or below a piece of equipment that normally runs hot (i.e. power amplifier). Remember, high temperature is the enemy of all electronic equipment.

Output Connections

It is not recommended that both the balanced and unbalanced outputs of the Binloop be used simultaneously. Ground loops (see section on Grounding and System Performance) and poor high frequency response can result. Additional low frequency accuracy can also be obtained by using 600 Ohm (low impedance) interfaces on the Binloop outputs.

Compact Flash Media Reliability

Compact Flash MTBF (Mean-Time Between Failure) rates published by most manufacturers are in excess of 1,000,000 hours under normal use. The fact is, properly installed Compact Flash usage in the Binloop is a very benign environment for Compact Flash media and there is a very, very rare occurrence of failure.

When failures do occur they normally occur during the first few hours of use. Often this is called "infant mortality." Even this failure mode is extremely rare, however.

Nonetheless, it is always prudent to properly backup all of the audio and video data contained in your Binloop. This can easily be done by simply copying the data to a PC hard drive or by making CDR or Data-DVD/R backups.

Like any other computer system, your sound and video files for the Binloop should be treated as critical data. Successful operation of your installation depends on this data.

Removing a Card from the Cage

WARNING: The following information should only be used when a card (Controller, or Video Reproducer) has failed and must be returned to Alcorn McBride Inc. Removing a card carelessly could result in further damage, which would not be covered under your warranty. **PLEASE BE EXTREMELY CAREFUL!!!**

1. Disconnect power from the unit.
2. Using a regular Phillips screwdriver, remove the screws from the reproducer cover panel on the right side of the Binloop front-panel.
3. Be careful of the display on the left cover. It has a cable connected to it. Observe the polarity of this cable connection and be sure to reconnect it in the same way if it becomes disconnected.

4. Inside the unit should be:
 - The Power Supply (a silver box on the left side)
 - The Controller Card (the card on the left with four buttons on the lower end and an array of ten LEDs on the top.
 - Various Audio and/or Video Reproducers - Cards with one button on the top, an LED right underneath, and a slot for Compact Flash card(s).
5. Verify the stability of each card (Controller, Audio/Video Reproducers) to make sure that it has not simply become disconnected:
 - **GENTLY** push on the card to make sure that is not off of the tracks in the slot. If it is stable, proceed to step 7.
6. If the card is not stable:
 - **GENTLY** pull the card out of the slot.
 - Carefully align the card in between the guides. The solder side of the circuit board should be facing the power supply on the left.
 - **GENTLY** push the card toward the back end of the unit until you feel the pins lock into place. **DO NOT FORCE THE CARD IN!!!**
7. To Remove the Card:
 - Find the card that needs to be replaced.
 - **GENTLY** pull the card toward you. At first you will feel some resistance as you pull the card from its connectors in the back. After the card has been detached, it will slide easily.
8. Place the card in a Static Shielding bag to protect the board from static discharge.
9. Place the bagged card on a stable surface.
10. Carefully place the cover back onto the unit. Make sure that it is facing the right direction (The Alcorn McBride logo should be right-side up and the buttons should fit through the holes).
11. Replace the screws and tighten.
12. Pack the card-in-the-bag in a box, padded to protect against damage.

Please be sure to obtain an RMA number from your distributor (if outside USA) or from Alcorn McBride before returning product.

Specifications

Physical

Power:	100 - 240VAC, 6-3A Max. 60-50 Hz.
Size and Weight:	19" W x 5.25" H x 10.25" D, 16 lbs.
Environment:	0o C (32o F) to 38o C (100o F) 0-90% Relative Humidity

Control Information

Push Button Controls:	Start/Stop or Mute (1 per Reproducer) SMPTE Start, Pause, Stop, Mute All
LED Indicators:	Playing/Muted (1 per Reproducer) Power, Fault, V-Sync, Gen, Ready, Run, Pause, Stop, Mute
VFD Display:	2 x 16 Characters
Parallel Inputs:	Play, Loop, Stop (for 8 groups) Start SMPTE, Pause SMPTE, Stop, Opto- Isolated
Parallel Outputs:	Ready, Running, Fault, dry contact relay outputs
Show Control Serial:	RS-232C – Settings vary by protocol
Programmer Port Serial:	RS-232C at 9600 baud (9th-bit parity)

Video

Formats:	MPEG-2 Program Stream MP@ML
Resolution:	576p, 480p, 576i, 480i
Digital Output:	SDI (SMPTE-259M)
Analog Output:	CVBS Composite Component Video (YPbPr or RGsB)

Audio

Formats:	MPEG-1 Layer I/II, MPEG-2 Layer II, WAV*, AIFF*, Raw PCM*
Digital Outputs:	AES/EBU or S/PDIF
Unbalanced Outputs:	+4 Volts Maximum into 20K
Balanced Outputs:	+8 Volts Maximum into 150 ohms (+20dBm)
Output Impedance:	DC Coupled, less than 10 ohms
Quantization:	16-bit, 24-bit**
Dynamic Range:	104 dB
Sample Rate:	32kHz, 44.1kHz, 48kHz, 96kHz**
Frequency Response:	DC - 20,000 Hz +/- 0.5 dB
Signal-to-Noise:	120 dB Below Full Scale when muted
THD+N:	-97dB
Track Skew:	Less than 100 nanoseconds
Sync Format:	NTSC at 29.97 Hz, or PAL at 25 Hz

* Supported for Audio playback only (no video)

** Only supported by the Repro24D reproducer card installed in the Digital Binloop HD.

SMPTE

Frame Rates:	23.976, 24, 25, 29.97, 30-drop, or 30
Trigger Skew:	23 μ Sec max
Lock Range:	+/- 20%
Timing Stability:	7 ppm
Dropout Tolerance:	Configurable (None, Infinite, or 1-8 seconds)
Input:	Transformer Isolated
Input Impedance:	600 Ω balanced
Output:	Balanced 2-10 VPP selectable 0dBV to +4dBV selectable
Output Impedance:	10 Ω maximum
Load Impedance:	10K Ω

Application Notes

Note 1: Automatically Play on Power-Up

Many times, all you may want the Binloop to do is to play and loop a video clip when it is powers up. This can be accomplished using the Binloop's built-in SMPTE generator, and the Timecode control interface. This application note will lead you through the process of configuring the unit to use these features to perform this type of function.

Preparation

If you do not already have a Compact Flash card loaded with a video clip, then follow the procedures found in the **Playing A Test File** section before going any further. This section will take you through the steps needed to format a Compact Flash card and load it with a test video.

This application note also requires that you have the Binloop Configuration software installed on a PC, and that the PC is connected to the Binloop using Ethernet or RS-232. If you do not have this software or you need guidance with connecting your PC to the Binloop, refer to the **Binloop Configuration Software** section before continuing with this example.

Connecting to the Binloop

To get started, launch the Binloop software and verify that you have established a connection with the PC to the Binloop. An easy way to do this is to upload the configuration from the Binloop to your PC. To do this, click on **File→New Upload from Binloop...** Unless you have already established a connection with the Binloop, a screen will appear prompting you to select the communication method. Select the appropriate method, and the click **OK** to begin uploading the Binloop configuration to your PC.


Once the upload is complete, a window will appear within the Binloop application. This window is called the Trigger List, and contains a list of all timecode triggers that were stored in the Binloop. If there were no triggers stored in the Binloop, this list will be empty. If you have reached this point without receiving any communication errors, you have successfully established a connection with the Binloop.

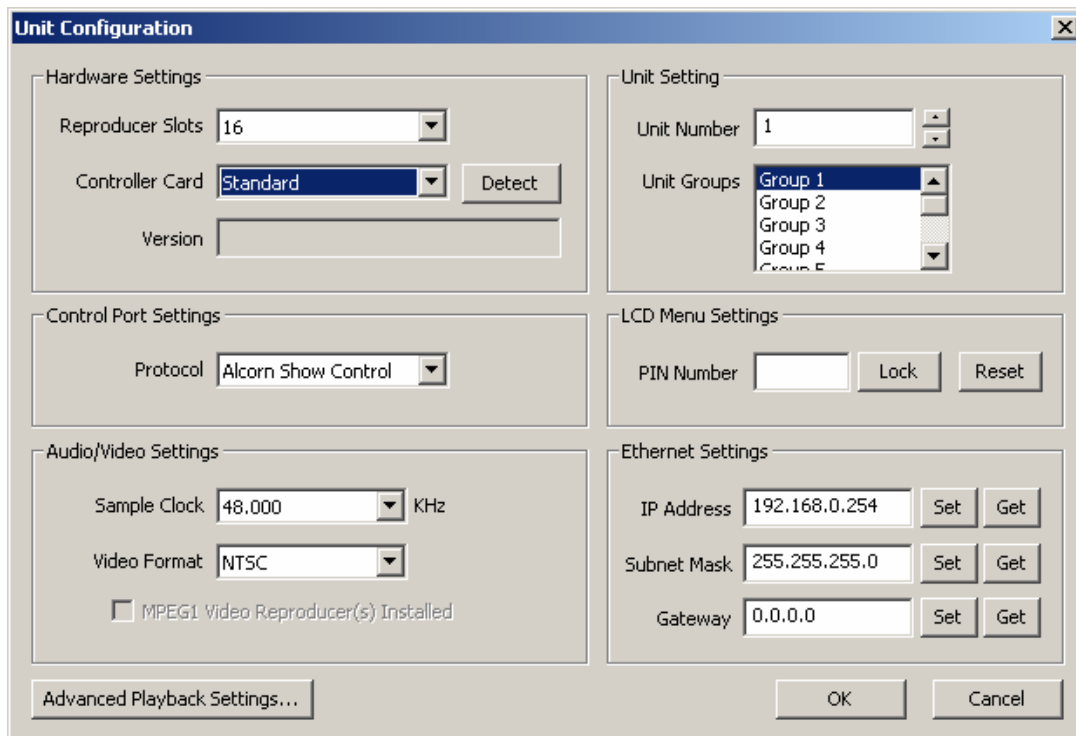
Modifying the Binloop Configuration

There are a few important things that need to happen before the Binloop can automatically play video on power-up. First, we need to configure the Binloop's SMPTE timecode interface to automatically generate timecode when the unit powers on. Next, we have to create a timecode trigger that will instruct the Binloop channels (Reproducers) to begin playing a video.

The idea is that the SMPTE clock will begin running on power up. When this clock reaches a certain value, a trigger will instruct the Binloop reproducers to play a video. When the video is finished, the SMPTE clock will loop back to its starting value and cause this process to repeat indefinitely.

Unit Configuration

Let's begin by configuring some of the basic Binloop settings. In the Binloop software, navigate to the **Unit Configuration** screen. To do this, select **Binloop→Settings→Unit...** from the application menu or by click on the  icon in the menu bar.

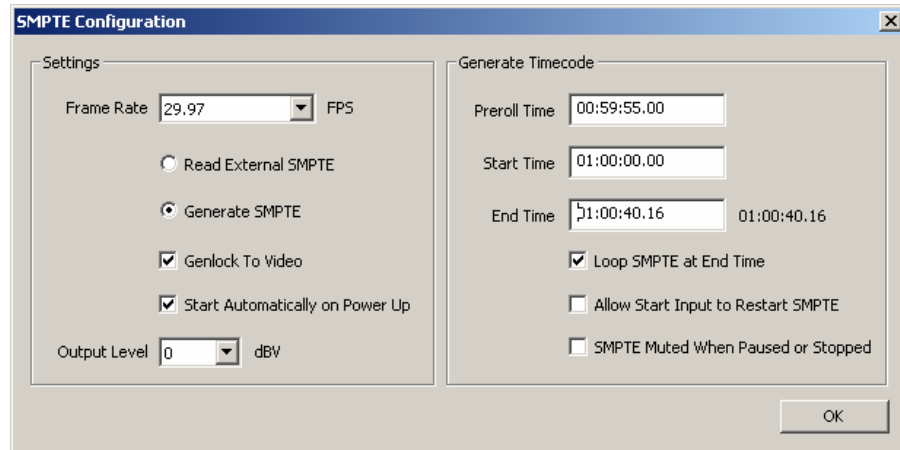


The **Unit Configuration** dialog box is divided into several sections for configuring the Binloop unit. The **Hardware Settings** section includes a **Reproducer Slots** dropdown set to 16, a **Controller Card** dropdown set to Standard with a **Detect** button, and a **Version** text field. The **Unit Setting** section has a **Unit Number** spinner set to 1 and a **Unit Groups** list box showing Group 1 through Group 5. The **Control Port Settings** section features a **Protocol** dropdown set to Alcorn Show Control. The **Audio/Video Settings** section includes a **Sample Clock** dropdown set to 48.000 KHz, a **Video Format** dropdown set to NTSC, and a checkbox for **MPEG1 Video Reproducer(s) Installed**. The **LCD Menu Settings** section has a **PIN Number** text field and **Lock** and **Reset** buttons. The **Ethernet Settings** section includes **IP Address**, **Subnet Mask**, and **Gateway** text fields, each with **Set** and **Get** buttons. At the bottom, there is an **Advanced Playback Settings...** button, and **OK** and **Cancel** buttons.

For the sake of this example, the majority of these settings are not important. Just make sure that the number of reproducer slots is configured for **16**, and that the **Standard** controller card is selected. Once you have verified these settings, click **OK** to exit this screen.

SMPTE Configuration

Next, we will setup the SMPTE interface in the **SMPTE Configuration** screen. You can access this screen by selecting **Binloop→SMPTE→Configuration...** or by clicking on the **S** icon.

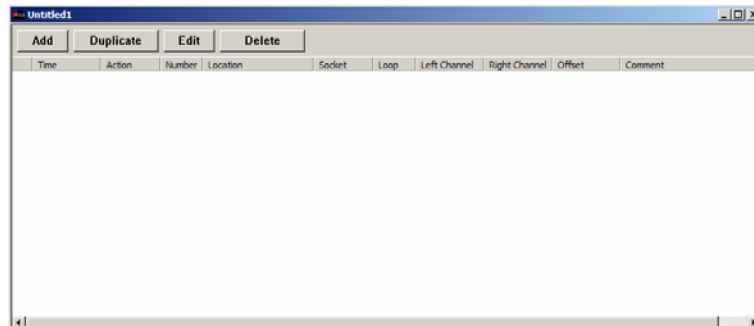


For consistency, we want the SMPTE clock to run at exactly the same rate as the video. We will be playing 29.97FPS video in this example, so select **29.97** as the frame rate and enable the **Genlock To Video** option. We also want the SMPTE generator to **Start Automatically on Power Up** and **Loop SMPTE at End Time**, so make sure those options are enabled as well.

Now, the only thing left to configure is the range of time that the SMPTE interface will generate. First, set the **Preroll Time** to **00:59:55.00** and the **Start Time** to **01:00:00.00**. For this example, the **End Time** value needs to coincide with the end of the video. If you loaded the test clip mentioned in the **Playing a Test File** section, the length of the video is 40 seconds (or 00:00:40.00). One might conclude that the End Time would then be 01:00:40.00, but not so fast! The reproducers are not capable of playing video immediately when they are commanded to do so. They need time to load the video before playback can actually begin, so there is a consistent delay associated with the Play command. This delay is exactly 16 frames. When you add this delay to the video length, you end up with an **End Time** of **01:00:40.16**. After that, click **OK** to exit this screen.

Trigger List

Now we have the important configuration screens set up, we need to add a **Play** trigger to the trigger list. This trigger will command the reproducers to play a video at the time we specify.




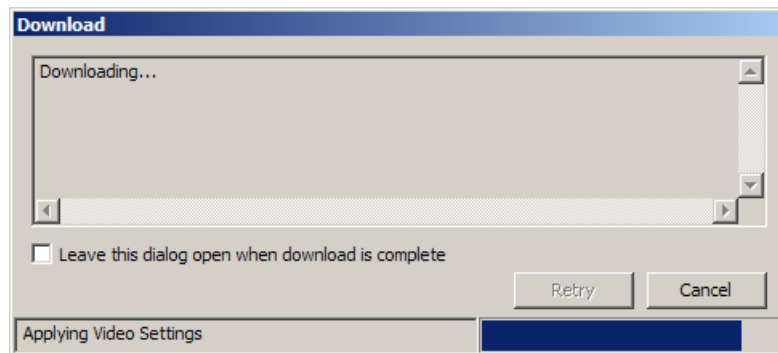
To create a new trigger, click the **Add** button at the top left of the trigger list window. The following window will appear.

A screenshot of a "Trigger" configuration dialog box. It is divided into several sections. The "Trigger" section on the left contains: a "Time" field with "01:00:00.00" and a smaller "01:00:00.00" label below it; an "Action" dropdown menu set to "Play"; a "Loop" checkbox which is unchecked; a "Location" dropdown menu set to "All Reproducers"; a "Number" field with a small increment/decrement control; and a "Socket" dropdown menu set to "Primary". The "Play" section on the right contains: a "Segment" dropdown menu set to "1" and an "Offset" field with "00:00:00.00". Below these is an "Audio" section with three radio button options: "Stereo or Mono" (which is selected), "Independent", and "Left" and "Right" (which are disabled). At the bottom of the dialog is a "Comment" text field. At the very bottom are four buttons: "Test Trigger", "Stop Reproducer", "OK", and "Cancel".

First, we need to enter the SMPTE time when we want this trigger to occur. Enter **01:00:00.00** in the **Time** box. Select **Play** as the **Action**, and make sure the **Loop** option is disabled. In the **Location** box, select **All Reproducers**. Next, verify that the **Segment** is set to **1** (this assumes that you have named your video as video #1, as instructed in the **Playing a Test File** section). Click on **OK** to add the trigger to the trigger list.

Downloading the New Configuration

We've now created a configuration that will cause the Binloop to play and loop video when it powers up. All we have to do is download that configuration to the Binloop so that it will take effect. To initiate the download, click on the  icon or select **Binloop→Download to Binloop...** A window will prompt you to make sure that you want to download the new configuration and overwrite the configuration currently in the Binloop. When you proceed, the following progress window will appear to show you the real-time progress of the download.



The configuration we just created is now stored in the Binloop. To verify that it works, turn the Binloop off and then back on again. After it finished initializing, the SMPTE clock will start running at 00:59:55.00. When the clock reaches 01:00:00.00, a synchronous play command will be sent to every reproducer. When the clock reaches 01:00:00.16, all reproducers in the Binloop with a video #1 will start playing video. When the SMPTE clock reaches 01:00:40.16, it will loop back to 01:00:00.00 and repeat the process.

Note 2: Controlling the Binloop with an Alcorn McBride Show Controller

While the Binloop has some very powerful features that allow it to operate as a stand-alone device in many circumstances, there are inevitably applications where an external control system is required. An installation may have other elements besides video such as lighting, audio, special effects, interactive interfaces, etc. In addition to making A/V products like the Binloop, Alcorn McBride also specializes in making Show Control systems to easily manage applications like these.

This goal of this application note is to show how easy it is to integrate one of our show control products with the Binloop for simple remote control. This particular example will use an Alcorn McBride InterActivator to start video playback in the Binloop from a button press. Keep in mind that, although we're using an InterActivator in this example, all of our controllers would be programmed using the exact same method.

Preparation

This application note requires that you have video clips loaded onto the Binloop. If this is not the case, then follow the procedures found in the **Playing A Test File** section before going any further. This section will take you through the steps needed to format a Compact Flash card and load it with a test video.

This application note also requires that you have the Alcorn McBride WinScript software installed on a Windows PC. This software is always available as a free download from our website at <http://www.alcorn.com>. Your PC must be connected to the Show Controller's Programmer port using an RS-232 serial cable.

Last but not least, the InterActivator and the Binloop must be connected via RS-232. In this example, we will be controlling the Binloop from Port 1 of the InterActivator. This means that a Null-model RS-232 cable must be connected between **Port 1** of the InterActivator to the **Show Control** port of the Binloop.

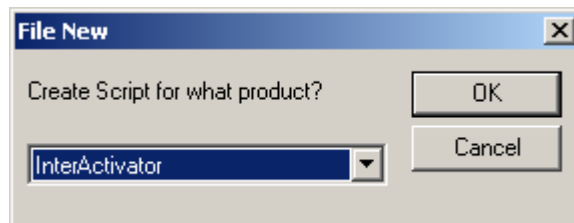
Configuring the Binloop

Before the InterActivator can properly control the Binloop, some steps must first be taken to prepare the Binloop. First, the Show Control port must be configured for Alcorn 9-Bit operation. This can be done using the front-panel menu interface of the Binloop or in the Unit Configuration screen of the Binloop software. Secondly, we must make sure that the Binloop is not doing anything that will interfere with the commands from the show control system. For example, if you followed the steps in the previous application note, the Binloop would currently be configured to trigger itself to play periodically. Connecting a show controller will not prevent this, so you must take steps to prevent any undesired behavior. This could be accomplished by deleting triggers from the Trigger List, configuring the SMPTE clock to be idle, or both. If you need assistance doing these things, you can use the previous application note as a guide on changing the Binloop's configuration.

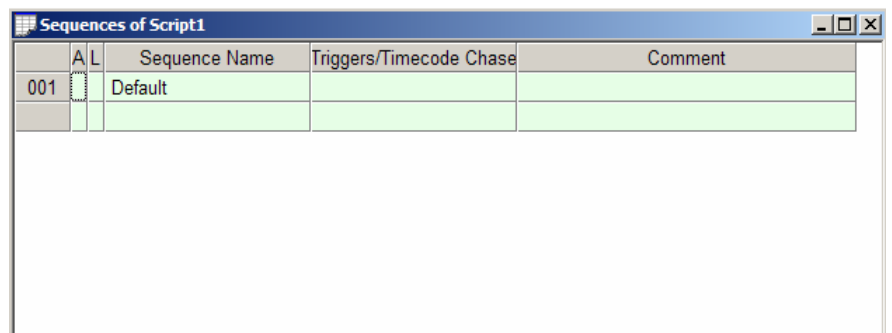
Programming the Show Controller

Alcorn McBride Show Controllers are designed to be programmed initially by a PC, and then operate independently of the PC once the programming is complete. We call this program a 'Script', and it is created and transferred using the WinScript application mentioned earlier in the application note.

To get started on writing a show control script, launch the WinScript software. WinScript may automatically start a new script called 'Script 1'. If so, close this script by clicking the **X** of the green **Sequences** screen, and then create a new one by clicking on **File→New**. The following screen will appear:



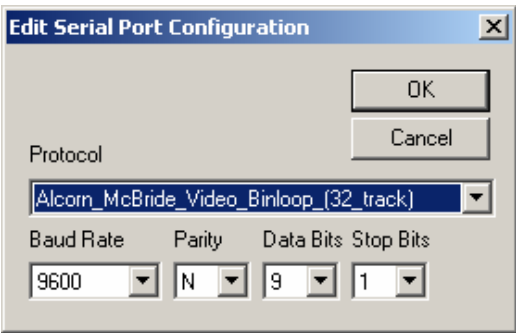
Select the show controller you wish to use, and then click **OK**. After that, a new blank script will be created that looks like this:



Before modifying the new script, click on **File→Save** to give it a descriptive name and store a copy of the script on your PC. For this example, we will save our script as **Interactivator-Binloop**.

Port Configuration

Next, we need to tell WinScript that a Binloop is connected to Port 1 of the show controller. Click on **Resources→Ports** to bring up the port configuration. Start by changing the name of **port1** to **Binloop** so that we can easily identify this connection in the script programming. Next, right-click in the **Protocol** column of the **Binloop** port and select **Protocol Wizard**. The following screen will appear, prompting you to select the protocol for the device connected to this port.

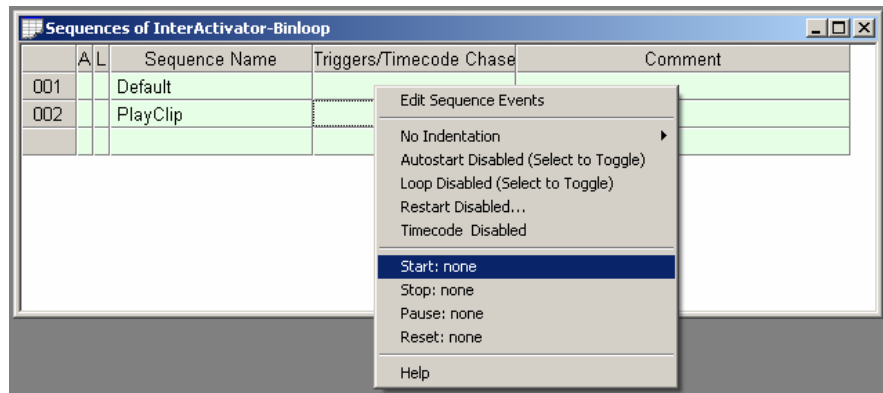


Select **Alcorn_McBride_Video_Binloop_(32_Track)** from the list and click **OK**. The Port Configuration should look like this when finished.

Ports of InterActivator-Binloop				
	Port Name	Protocol	Error Seq	Port Comment
00	Programmer	Alcorn_9_Bit_Control		
01	port1	Alcorn_McBride_Video_Binloop_(32_track)		
02	port2	None		

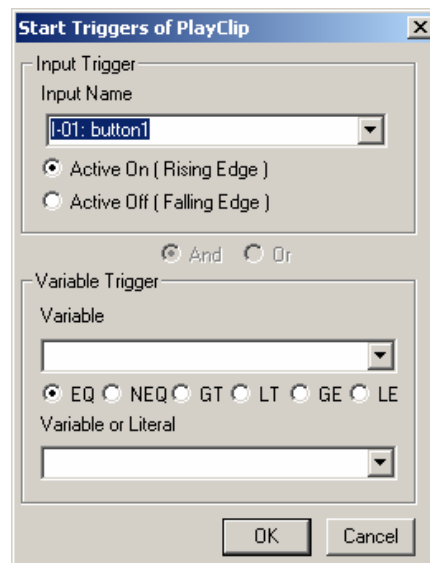
Creating a Sequence

Next, let's add a new sequence called **PlayClip**. Double-click on the empty cell just below the cell that contains **Default**. Type in the name **PlayClip**.

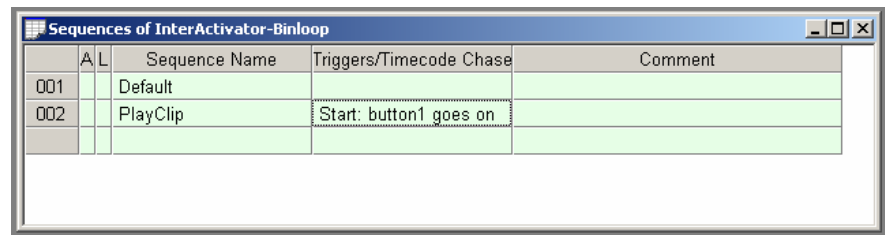


Creating the Input Trigger

We have now created a sequence that we will eventually program to start playing a file in the Binloop. Before we get to that part, let's first configure this sequence to be triggered by pressing Button #1 of the InterActivator. To do this, right-click on the **PlayClip** sequence and select **Start: none**. This will open a window that will allow you to specify a trigger to start the PlayClip sequence. Select **button1** as the input name, and then click the **OK** button.



Your script should look like this.



	A L	Sequence Name	Triggers/Timecode Chase	Comment
001		Default		
002		PlayClip	Start: button1 goes on	

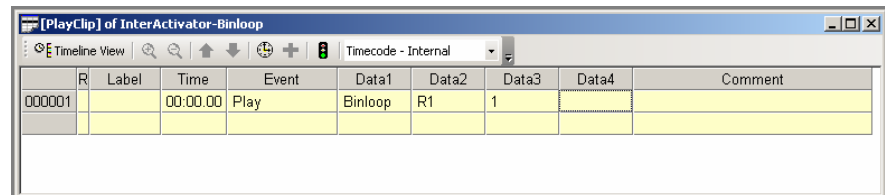
Now, the **PlayClip** sequence will be triggered when **Button #1** on the front of the controller is pushed.

Editing the PlayFile Sequence

At this point, the **PlayClip** sequence is empty, so nothing will happen when it is activated. That means that the next step is to add events in the **PlayClip** sequence so that it commands the Binloop to play a file.

To access the event list for the **PlayClip** sequence, right-click on PlayClip and select **Edit Sequence Events**. You can also double-click the sequence number on the left-hand side (i.e. **002**) to access the event list.

The event list will be mostly empty with the exception of single **nop** event (which does nothing). Right-click on the **nop** event and select **Play** from the list of events. Most events in WinScript require additional information, or data parameters, to work. For example, the **Play** event needs to know which Binloop to talk to (there might be more than one!), which reproducer to control, and what file to play. To determine what information to enter in each column, you can always right-click and select **Event Wizard** for guidance.



	R	Label	Time	Event	Data1	Data2	Data3	Data4	Comment
000001			00:00:00	Play	Binloop	R1	1		

Here's an explanation of the event fields, as well as how you should fill them out for this application.

Time – This field specifies when to execute the event in relation to the when the sequence is started. We want our **Play** event to happen immediately after the sequence is started, so leave the time value as **00:00:00.00**.

Event – This is the action that is to be taken at the specified time. In this case, we are using the Binloop **Play** event which instructs one or more channels of the Binloop to play a file.

Data1 – For the **Play** event, this data parameter needs to specify the name of the port connected to the Binloop we want to receive the command. We are using Port 1, which we named **Binloop** earlier in this application note. Enter **Binloop** in this column.

Data2 – The **Play** event uses this parameter to identify which card (or cards) to command to play. This event allows you to command a single reproducer (R1-R16), a group of reproducers (G1-G13), or all reproducers (All). Enter **R1** in this column to specify Reproducer #1.

Data3 – The **Play** event uses this field for the video clip number. Reproducers can have multiple clips stored on their Compact Flash cards, so it's important to specify which clip to play. Enter the number **1** in this field.

We have now created an event that will command the Binloop to play file #1 on Reproducer #1 as soon as the PlayFile sequence is started. You can now close the PlayFile Event list, and return to viewing the Sequence list.

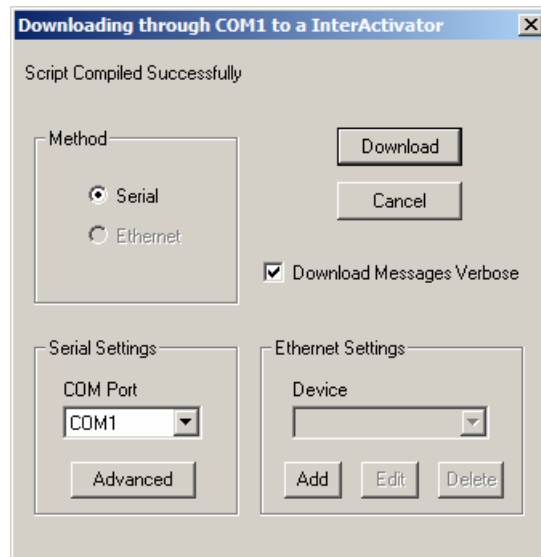
Downloading the Script to the Show Controller

Our script is now finished and ready to compile and download.

Start by clicking on the  icon in the menu bar, or by selecting **File→Compile Script and Download**. The script is then compiled and checked for errors. If no errors are found the screen will look like this.

Compilation of Interactivator-bIHD - Double Left or Right Click Column Header to Sort Up or Down				
	Compiler Message	Error	Seq/Res	Event
00	WinScript ScriptOS Incremental Compiler Version 1.02			
01	Optimizing for Speed			
02				
03	Compiling C:\User Projects\WinScript Tutorial\Temp\Interactivator-bIHD.amw...			
04				
05	Compiling Configuration...			
06				
07	Compiling Sequences...			
08	Compiling sequence Default			
09	Compiling sequence PlayClip			
10	Compiled script size: 166.			
11	Writing Interactivator-bIHD.DAT			
12				
13	Interactivator-bIHD.amw compiled with 0 Error(s), 0 Warning(s)			
14				

After the compiling process is successful, the following window will appear prompting you to download the compiled script to the show controller.



Verify that the **COM Port** setting matches the PC COM Port that you have connected to the show controller. You should then click the **Download** button to begin transferring the script to the controller. When the transfer is complete, you may be asked to compress and store the non-compiled script file. You should always do this whenever possible for permanent installations but, for this example, go ahead and click **No**.

Testing the Show Control Script

At this point, the script is now running on the show controller. You can now press Button #1 on the front of the show controller to play video #1 on reproducer #1 of the Binloop. Immediately after you press the button, you will see the serial LED 1 blink to indicate that the controller is sending a serial message on RS-232 Port 1. You will then see the green status LED of Reproducer #1 illuminate green to indicate that it is playing a video.

Now that you have the basics, you can easily modify this script to do other things more suited for your application. Perhaps you want to play different files based on which button is pressed? Maybe you want to trigger playback from a motion sensor wired into the controller? Having a Show Controller in your system provides all kinds of possibilities.

Note 3: Connecting to the Binloop using Ethernet

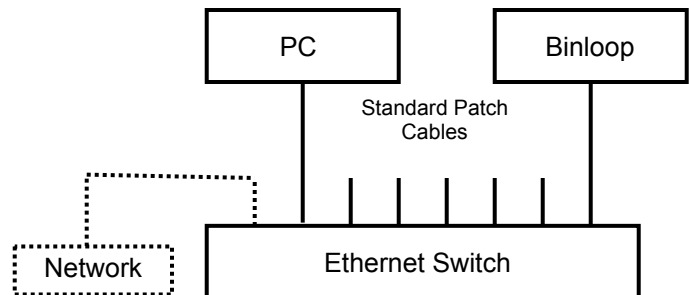
Ethernet is a very common communication interface used to connect devices together. Using the Binloop's Ethernet interface, you can download configurations and even control the Binloop across an Ethernet network.

Ethernet is a very broad standard with many complexities that go way beyond the scope of this manual. The intention of this application note is to simply demonstrate how to connect a Windows PC directly to the Binloop using Ethernet.

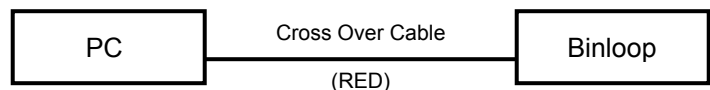
Preparation

This application note also requires that you have the Binloop Configuration software installed on a PC, and that the PC is connected to the Binloop via Ethernet. If you do not have this software, refer to the **Binloop Configuration Software** section before continuing with this example.

As for the Ethernet connection itself, there are two ways to do this. The most common way is to use standard Ethernet patch cables and a network switch or hub. When you connect the PC and the Binloop to the numbered ports of the switch, the status LED's of the switch should indicate a connection.



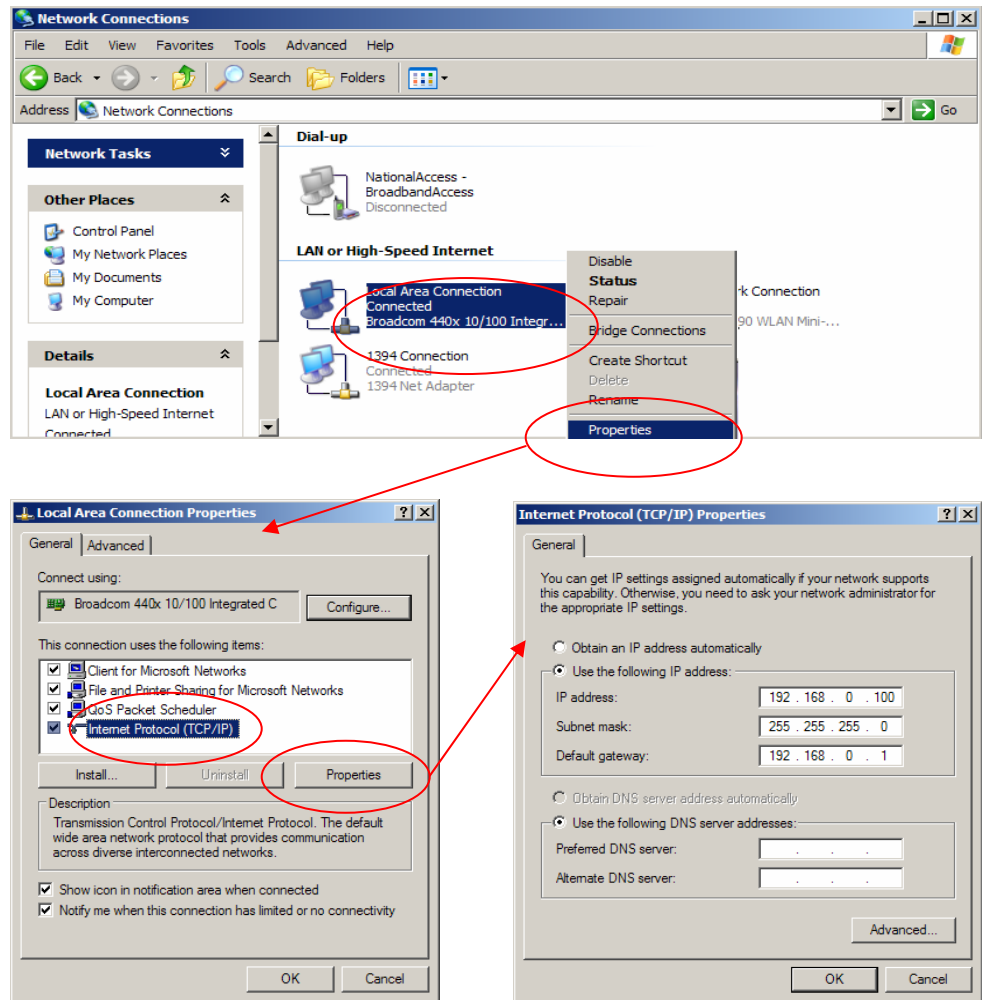
Another way to connect your PC to the Binloop over Ethernet is using an Ethernet crossover cable, like the red one included with the Binloop packaging. This cable allows you to connect the PC directly to the Binloop without the need for a switch or hub.



Changing the Network Settings of your PC

In order for a PC to communicate to a Binloop over an IP-based Ethernet network, the PC and the Binloop must be on the same Subnet. In this case, the subnet is 192.168.0.xxx. This will most likely require you to change the network settings of your PC before you are able to communicate with a Binloop that is using default network settings. The following steps will guide you through checking and setting a Windows XP-based PC correctly. Other computer operating systems may differ in method and appearance, but the concept is still the same.

1. Open the Windows **Control Panel** and open **Network Connections**. Your PC's Ethernet connection will appear in this list, and is usually named **Local Area Connection**. Keep in mind that you may have several network connections, so it's important that you select the connection that is connected to the Binloop.
2. Right-click on this connection and select **Properties**. Click on the **Internet Protocol (TCP/IP)** item then click the **Properties** button.
3. When the TCP/IP properties window opens, write down all of your current settings before making any changes. When you are done communicating with the Binloop, you will need to return to this screen and restore your previous settings.
4. Click the option **Use the following IP address**.
5. Enter **192.168.0.100** into the **IP address** box
6. Enter **255.255.255.0** in the **Subnet mask** box
7. Enter **192.169.0.1** as the **Default gateway**
8. Click **OK** to apply the new IP Address to your PC.
9. Click **OK** once again to exit the connection properties screen.



Configuring the Network Settings of the Binloop

This application note requires your Binloop is configured for the factory default network settings, which are as follows:

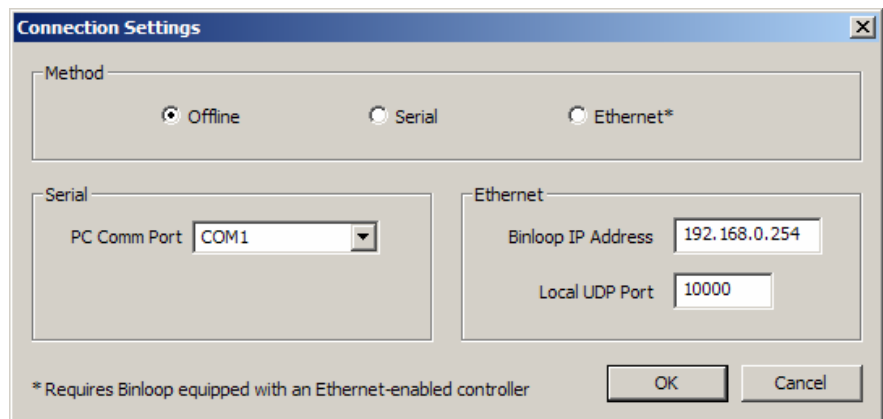
IP Address	192.168.0.254
Subnet Mask	255.255.255.0
Gateway	192.168.0.1

If you believe that these settings may have been altered, you should verify them using the front-panel menu wheel interface of the Binloop. Browse to the **Network Setup** screen and make sure that all of the settings match these values.

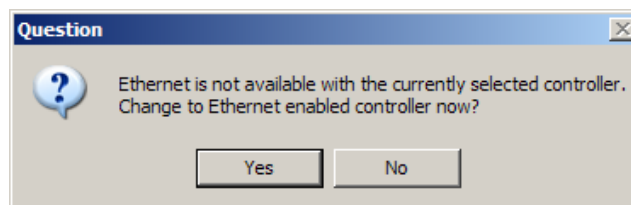
Connecting over Ethernet with the Binloop Software

At this point, your PC is now physically connected to the Binloop through Ethernet. We've also made sure that both the Binloop and your PC have network configurations that are compatible, and capable of communicating with one another. The next step is to actually use this interface to communicate with and control the Binloop.


Begin by launching the Binloop Software. When the program is loaded, open the **Connection Settings** screen by clicking on the  icon or selecting **Binloop→Settings→Connection...**

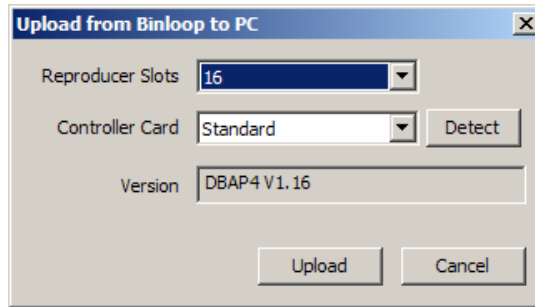


Select **Ethernet** as the communication method and verify the default address of **192.168.0.254** is in the **Binloop IP Address** box. When you select Ethernet, the following message may appear.



If it does, select **Yes** and then click **OK** to exit the Connection Settings screen.

We can now verify the Ethernet connection by uploading the configuration from the Binloop to your PC. To do this, click on the  icon or select **Binloop→Upload From Binloop...**



When this screen appears, click on the **Detect** button. If the Ethernet connection is working, the version box should fill with the current firmware version of the Binloop's controller card.

If you wish, you may click **Upload** to transfer the Binloop's configuration to your PC over the Ethernet connection. You can also use all of the features of the Binloop software to change configuration settings, and control the Binloop over Ethernet.

Appendix A - Troubleshooting Guides

Card Cage Troubleshooting Guide		
Symptom	Cause(s)	Solution(s)
1) No Front Panel display or LEDs	a) No Power b) Bad Power Supply	a) Check Power connections b) Cycle Power
2) One or more of the reproducers not responding to test button press.	a) Incompatible Video File b) Faulty Button c) Faulty Reproducer d) Faulty Controller	a) Press Reproducer Test Button to see if reproducer responds. b) Press Mute Button and make sure Reproducer LED's illuminate RED. (This verifies communication) c) Cycle Power

Video Troubleshooting Guide		
Symptom	Cause(s)	Solution(s)
1) No Video Output single track - no fault indicated. Status and/or Mute indicator off or always lit. Reproducer does not respond.	a) Reproducer microprocessor hung up due to "hot-carding" of CF cards.	a) Cycle power on unit.
2) Component video output is discolored on your display (i.e. GREEN, BLUE)	a) Component Video output wired incorrectly b) Faulty or loose cables b) Color space configured incorrectly for display	a) Check cables b) Change Colorspace configuration for affected reproducer. c) Change Colorspace setting of display (if possible)
3) Video plays fine for several seconds, and then starts stuttering – blocky artifacts appear on the video output. Audio may sound strange, or drop out.	a) Video Format of Binloop does not match framerate of video clip. b) Sample rate of Binloop does not match sample rate of the audio track in the video clip.	a) Verify that the video is encoded using a supported framerate and that the Video Format of the Binloop matches that framerate. b) Change sample rate of Binloop or audio track.

Audio Troubleshooting Guide		
Symptom	Cause(s)	Solution(s)
1) No Audio Output, Reproducer status LED is RED	a) Binloop Muted b) Reproducer Muted	a) Verify that MUTE Button on front of Binloop is not enabled. b) Send Unmute Audio command to affected reproducer. c) Power-cycle Binloop
2) No Audio Output, Reproducer status LED is not RED.	a) Audio output connector disconnected from unit or shorted.	a) Restore audio output connection to Digital Binloop HD.
3) Audio Output distorted on single reproducer card.	a) Audio output wired incorrectly - one end of balanced output might be shorted to ground.. b) Faulty reproducer	a) Rewire balanced audio output connection. Leave (-) output disconnected for connection to unbalanced inputs.
4) Audio Output distorted - multiple outputs.	a) Same as (3) above b) Interconnect ground loop. c) Faulty controller	a) Same as (3) above. b) Power cycle Binloop.
5) Audio sounds high-pitched, or low-pitched compared to how it should sound.	a) Binloop sample rate does not match sample rate of the audio file.	b) Change sample rate of Binloop or audio track.

Appendix B - Compact Flash Media Capacities

The tables below should give you an indication of the amount of audio (mono or stereo) or video information you can fit on a Compact Flash media of a given size.

Keep in mind that, even though media size can be larger, the maximum supported file size is 2GB. A computer will not allow you to copy a file larger than this to the flash card. If your application requires a longer playback time than a 2GB limitation will allow, the best solution is to create separate files and 'seam' them together using the **Play Next** command.

MPEG2 Video - 8Mbps CBR (High Quality)

Compact Flash Size (Gigabytes)	Approx. Capacity (Minutes)
1	18
2	36
4	72
8	144
16	288
32	576
64	1152

MPEG2 Video - 15Mbps CBR (Highest Quality)

Compact Flash Size (Gigabytes)	Approx. Capacity (Minutes)
1	10
2	20
4	40
8	80
16	160
32	320
64	640

Uncompressed Audio - 16-bit Stereo

Compact Flash Size (Gigabytes)	Capacity (Minutes)			
	32k	44.1k	48k	*96k
1	140	100	90	45
2	280	200	180	90
4	560	400	360	180
8	1120	800	720	360
16	2240	1600	1440	720
32	4480	3200	2880	1440
64	8960	6400	5760	2880

** Only supported by Repro24D (Digital Binloop)*

Uncompressed Audio - 24-bit* Stereo

Compact Flash Size (Gigabytes)	Capacity (Minutes)			
	32k	44.1k	48k	*96k
1	90	66	60	30
2	180	132	120	60
4	360	264	240	120
8	720	528	480	240
16	1440	1056	960	480
32	2880	2112	1920	960
64	5760	4224	3840	1920

** Only supported by Repro24D (Digital Binloop)*

Appendix C - Updating Firmware

The various components of the Binloop are all capable of accepting firmware updates. The following guides will instruct you on how to update the firmware of each component.

Reproducer Card Firmware

The Repro16V (Video Binloop) and Repro24D (Digital Binloop) Reproducers can be updated by copying a firmware update file to a Compact Flash card and inserting that card into the Reproducer's CF socket. These step-by-step instructions will walk you through this process:

1. Download the latest firmware for the Repro from:
<http://www.alcorn.com/support/firmware.html>
2. Extract the zipped firmware update file to your PC.
3. Power down the Binloop, and remove the Compact Flash card from the reproducer you wish to update.
4. Copy the **OS.NEW** file from the downloaded update to the Compact Flash Card using a card reader.
5. Re-insert the Compact Flash Card into the Reproducer.
6. Power on the Binloop.
7. The reproducer's status LED will turn ORANGE to indicate that it is updating the firmware.
8. Wait for LED to turn off, indicating that the firmware update is complete.
9. When you reach this point, you're done. The **OS.NEW** file will be renamed to **OS.SAV** to flag the successful operation.
10. Use the **Version** command to verify that the Reproducer's firmware has actually been updated to match the version on the website.
11. Repeat this process for each reproducer.

NOTE: If multiple updates are to be performed, you can rename **OS.NEW** to **FACTORY.NEW**. This causes the update to occur without the file name changing to **OS.SAV** after the update is complete. This allows you to update an entire cage of reproducer cards with a single Compact Flash card containing the **FACTORY.NEW** file.

Controller Card Firmware

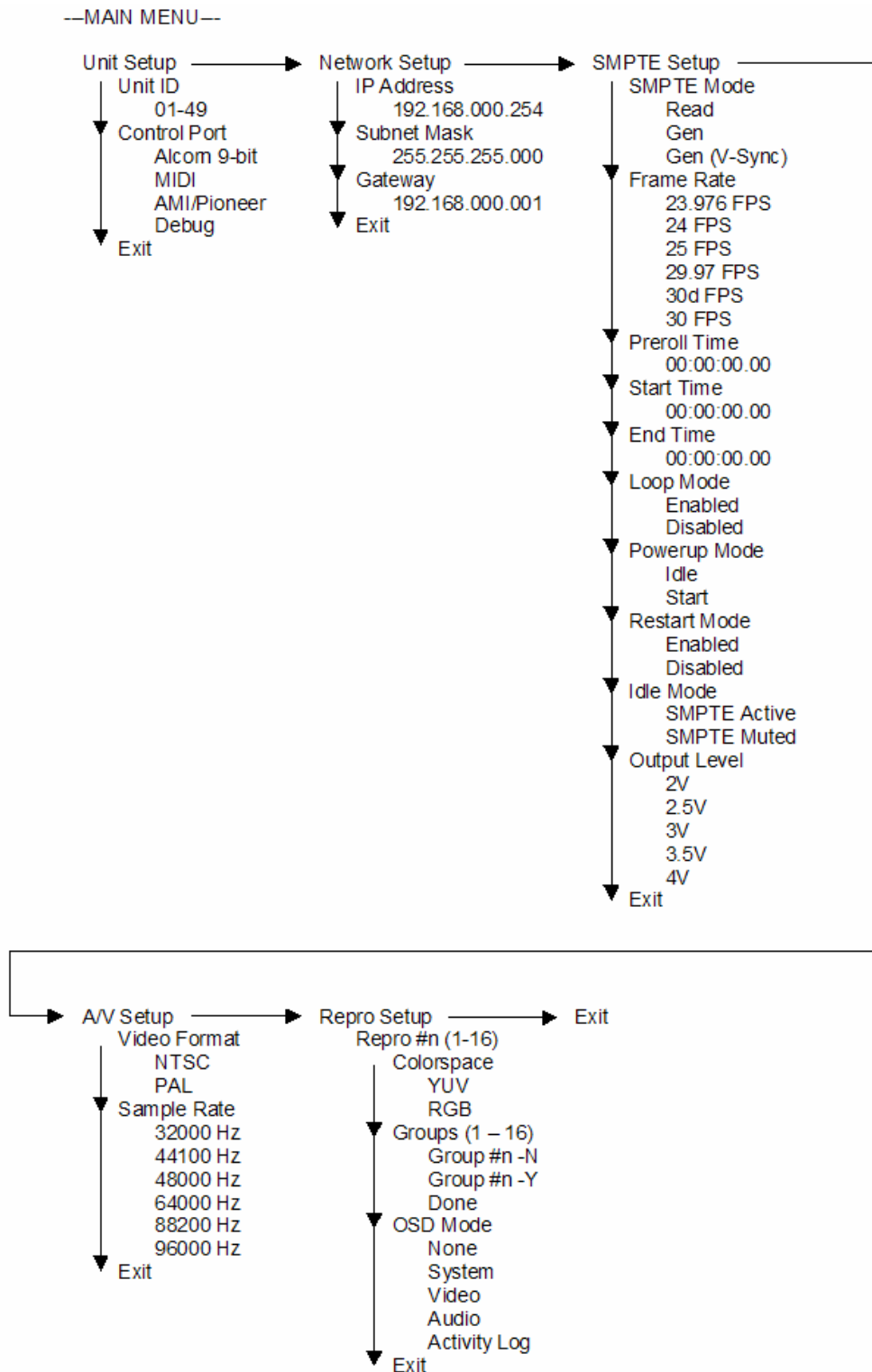
The DBAP4 Controller Card used in the Binloop must be updated over an RS-232 serial connection. This will require you to have a Windows PC with an RS-232 port (or a USB→RS-232 serial adapter). This PC should also have the latest version of the Binloop Config software installed. Once you have met these requirements, follow these instructions to update the DBAP4 Controller Card:

1. Download the latest firmware for the DBAP4 from:
<http://www.alcorn.com/support/firmware.html>
2. Extract the zipped firmware update files to your PC.
3. Make sure the Binloop is off, and then press in the MUTE button. (The MUTE button toggles each time it is pressed, so make sure it is toggled to the ON position)
4. Power up the Binloop, and verify that the display reads "Update Firmware? Press Start..."
5. Press the Start button on the front of the Binloop (top button with play icon next to it)
6. Verify that the display now reads "Firmware Update Mode".
7. Connect a straight-thru RS-232 cable between a PC and the Binloop's Programmer serial port.
8. Launch the Binloop Config software.
9. From the Binloop software menu, select **Help→Update Firmware...**
10. Select **DBAP4 Standard Controller** as the component to update.
11. Choose the COM port for the PC serial port that is connected to the Binloop.
12. Click the **Check** button and verify that the Binloop responds with "DBAP4 Bootloader".
13. Click the Browse button, and browse to the folder you extracted in step #2. Select the **DBAP4.HEX** file and click **Open**.
14. Click on the **Update** button at the bottom of the window, and wait for the update to complete. Leave the Firmware Update window open.
15. When the update finishes, press the MUTE button on the front of the Binloop to exit the firmware update mode.
16. Watch the Binloop display to verify that the firmware version corresponds to the update firmware version.
17. Wait for the READY LED to turn on, indicating that the Binloop has powered up properly.

18. In the Firmware Update window of the Binloop software, select **DBAP4 SMPTE Module** as the component to update.
19. Click the **Check** button and verify that the Binloop responds with "SMPTE Vx.xx".
20. Click the **Browse** button and browse to the folder extracted in step #2. Select the **SMPTECOR.HEX** file and click **Open**.
21. Click on the **Update** button at the bottom of the window and wait for the update to complete.

You have now successfully updated the Control System and the SMPTE Interface of the DBAP4 controller card to the latest versions.

Appendix D – Menu Wheel Map



Index

AES/EBU	79, 96	PAL	96
Audio Reproducer	7	Parallel Control	77
Balanced Audio Outputs	77, 79	pause	73
Binloop Configuration Software	47	Pause	83, 95
Bitrate	87	Play.....23, 24, 25, 26, 27, 28, 29, 30, 31, 32,	
card cage	7	33, 34, 35, 36, 37, 38, 39, 40	
Chroma Format	87	Play Next	
CobraNet	67	Seamless Playback	59
Compact Flash	119	Power	3–4, 83, 92, 115
Configuration	80, 81	Print	53
Connectors	92, 94	Program port	81
Control	17	Protocol	23, 81
Controller	3, 43, 80–82, 93	Ready	48, 73, 83, 95
Creating Alcorn Video	83–85	Rear Panel	92
Display	75	Reproducer Status LED	76
Dolby Digital	iii	Reproducer Test Button	76
Edit	63	Reset	44, 83
End Time Code	74	RS-232	77
Exit	53	Run	73, 91
Fault	73–76, 83, 95, 116	S/PDIF	79, 96
File Naming	86	Sample Rate	96
Firmware	iii, 119, 123	Save	52, 53
Format	59	Serial Port	48, 81
Frame Rate	52	Show Control	43, 67, 77, 80, 95
free-air circulation	3	SMPTE	82
Gen	73, 95	SMPTE Frame Rate	52
Grounding	91–93	SMPTE Interface	13
hard drives	51	SMPTE Start Time	74
Hardware	73	Start	44, 80, 83, 95
Installation	3, 91, 93	Start Button	74
Latency	66	Technical Support	1
LED	119	Temperature	92
Lock	94, 95, 96	Troubleshooting	115
Mechanical	iii	Unbalanced Audio Outputs	77, 80
Media Capacities	117	<i>Undo</i>	63
Media Reliability	93	Unit Address	44–46
Media Utilities		Upload Configuration	61
Format	59	USB	
Menu Wheel	75	USB-to-Serial	48
Message Format	44	Video Outputs	77, 80
MIDI In	77, 81	Video Reproducer	93
MIDI Out	77, 82	Windows Vista	47
Mute	73–74, 83, 95	Windows XP	47
NTSC	96		