Introduction

Bluefish444, the professionals' choice, supports Epic® Unreal Engine® to bring true uncompressed 12-bit HD-SDI and SMPTE IP I/O to users of Epic Games Unreal Engine. Bluefish has been the choice of professionals requiring the highest quality SDI and SMPTE IP input and output solutions with hardware options supporting 4:2:2 YUV, 4:4:4 RGB, 3D, and 4K HD-SDI, 12-bit video processing. Bluefish has a long history of providing OEM solutions to industry-leading developers requiring high-quality I/O solutions capable of performing day in day out, all year round. The range of Bluefish video hardware caters to video professionals that require tried and tested reliability, combined with the lowest latency of true uncompressed input and output solutions. Supporting 4K/2K/HD/SD capture & playback, 12-bit video processing, and a quality HDMI video preview.

Installation guide

Bluefish444 supports the Unreal Engine with the Plugin and Source code made available via the Unreal Engine Marketplace.

Install Bluefish444 Professional IO for use with Unreal Engine

1. Install Unreal Engine software and third-party applications as required.
2. Physically install the Bluefish444 HD-SDI IO hardware in the recommended PCI express slot. Refer to the hardware user manual for details on physical card installation. The Bluefish444 user manual can be downloaded from the Manual section of the Bluefish444 website.
3. Download, save, and run the Bluefish444 Driver, specific Unreal Engine and Bluefish444 Driver compatibility can be checked here.
4. The Bluefish444 installer package may request that you update your hardware to the latest firmware. Bluefish444 recommends running the firmware included in the latest installer for use with all third-party applications.
5. Your Bluefish444 hardware should now be installed and ready for use.
6. Now you can install the Bluefish444 Unreal Engine Plugin from the UE Marketplace.

Unreal Engine Configuration

Requirements

In order to use Bluefish444 professional Video hardware within the Unreal Engine there are a number of prerequisites listed below;

- Bluefish444 compatible Hardware
- Bluefish444 Driver and Firmware
- Bluefish444 Unreal Engine Plugin (Available from the Marketplace)
- Supported Unreal Engine version

With the above requirements met and the Bluefish444 hardware confirmed as configured and working correctly, we can progress to configure an Unreal Engine Project.

Unreal Engine Project

Open the Project that you want to use with Bluefish444 video I/O in the Unreal Editor then;

- From the main menu, select Edit > Plugins.
- In the Plugins window, find the BF444 Media Player plugin under the Media Players category. Check the Enabled checkbox.
- Find the Media Framework Utilities Plugin under the Media Players category. Check the Enabled checkbox, if it's not already checked.
- Click Restart Now to restart the Unreal Editor and reopen your Project.

Your Project is now ready to accept video from the Bluefish444 compatible hardware, and to send the rendered output out the cards interfaces.

In the next section, we will connect it within the project to allow Bluefish444 inputs to be used within Unreal Engine and to play the final rendering out of the Bluefish444 Output interfaces.

Rendering Video Input in the Unreal Engine

To bring an input from the Bluefish444 interfaces into the Unreal Engine we will need to use the Media Bundle asset.

Once you bring the Media Bundle asset into the Content Editor, and it is named appropriately you can doubleclick to edit it's properties.

- Select your Bluefish444 Device, and the channel that you want to capture from.
- Further properties can be set depending on the projects requirements, refer to the Bluefish444 Media Reference
- Save your changes and close the media Bundle.
- The Media Bundle itself can now be placed within your level and the Video Input will be displayed on this surface

If your Media Bundle doesn't start playing automatically, select it, then click the Media Bundle > Request Play Media button in the Details panel.

Capture Rendering for Output

In order to output the renderings from Unreal Engine to the Bluefish444 Output Interfaces we will use the Media Captures panel.

- From the content Browser add a BF Media output asset from the Media submenu.
- Name the new asset, and double click to configure its properties.
- Set the Bluefish444 Device, Destination, Resolution, Standard and Frame Rate.
- Further properties can be set depending on the projects requirements, refer to the Bluefish444 Media Reference.
- If you do not have a Cinema Camera actor, add one now from the Cinematic Tab.
- From the Main Menu select the Media Capture window.
- Expand the Media Viewport Capture, Viewport Captures, Index.
- At the Cameras element select Add Element, from the Index below select the Cinema Camera Actor that you have used in your level.

- Then from the Media Output below select the BF Media Output Asset that you added and named earlier.
- When the Capture button is selected then the rendering from Unreal will be output via the Bluefish444 interface as configured earlier.

Timecode and Genlock

Bluefish444 provide a Blueprint "Bf444TimecodeProvider" that can be used to provide Unreal Engine with a Timecode from a Bluefish device.

- The Bf444TimecodeProvider can be found in the Blueprints, expand All Classes and start to type the name Bf444TimecodeProvider.
- Name the new asset and double click to set the properties of the Bf444TimecodeProvide asset.
- Choose the Bluefish444 device and channel that you would like to recieve the Timecode from.
- Below select the Timecode Source:

```
ANC VITC (SDI RP188 VITC Ancillary Timecode)
ANC LTC (SDI RP188 LTC Ancillary Timecode)
ANC Ext LTC (Ext LTC captured with Video Channel)
Ext LTC (Ext LTC captured Directly from LTC Source)
```

- If Ext LTC is used, then you must select a frame rate from the below Ext LTC Frame Rate drop down.

- Select Edit > Project Settings from the main menu and search for Timecode
- In the TimecodeProvider select the Bf444TimecodeProvider asset that you have added to your project.

Bluefish444 provide a Blueprint "Bf444CustomTimestep" that can be used to lock the rendering frequency of the Unreal Engine to one of the Bluefish444 interfaces.

- The Bf444CustomTimestep can be found in the Blueprints, expand All Classes and start to type the name Bf444CustomTimestep.
- Name the new asset and double click to set the properties of the Bf444CustomTimestep asset.
- Choose the Bluefish444 device and channel that you would like to lock the engine to.
- Select the Time Step Source:

```
SDI IN
REF IN
Ext LTC In
```

- Select the Timecode Format:

```
LTC
VITC
NONE
```

  - Select Edit > Project Settings from the main menu and search for Timestep
  - From the dropdown choose the Bf444CustomTimeStep asset you created earlier

Bluefish444 Media Reference

Versions

Bluefish444 hardware is compatible with the following versions of Epic Games Unreal Engine software:

- Unreal Engine 5.0
- Unreal Engine 5.1
- Unreal Engine 5.2

Unreal Engine compatible Bluefish444 hardware

KRONOS K8	Epoch Supernova	Epoch Neutron
KRONOS Optikos3G	Epoch Supernova S+	Epoch Neutron LP
		Epoch 4K Neutron
		Epoch 4K Neutron LP

Bluefish444 Media Source Settings

When creating a Bf444 Media Source you can configure the input with the following settings.

Property	**Description**
Device	Select the Bluefish444 Device to attach to
Source	Set Single Link, Dual Link, Quad Link SDI Interface
Resolution	Set Video Modes Resolution
Standard	Set Progressive or Interlaced
Frame Rate	Set the Video Modes Frame Rate
Auto	Auto will set the input based on the selected input channel
Timecode Format	Select the Timecode format from the list
Ancillary	
Capture Ancillary	Enable or Disable the Capture of Ancillary Data
Max Number Ancillary Frame Buffers	Set the max number of buffers for Ancillary
EIA 708	
Capture EIA708	Enable or Disable the Capture of EIA708 Closed Captions
Audio	
Capture Audio	Enable or Disable the Capture of Audio
Audio Channels	Set the number of Audio Channels to Capture

Audio	
Max Number of Audio Frame Buffers	Set the Max number of Audio buffers
Video	
Capture Video	Enable or Disable the Capture of Video
Color Format	Set the Pixel format to capture to
Is sRGB	Select if the Input is in sRGB Color Space
Max Number of Video Buffers	Set the Max number of Video buffers
Debug	
Log Dropped Frames	Enable or Disable the Logging of Dropped Frames, note that this may affect performance
Burn Frame Timecode	Enable or Disable the Burning in of Timecode to the Input
synchronization	
Syncronize with Engines Timecode	Syncronize the media with the Engines Timecode
Frame Delay	Set the number of Frames to Delay the Media
Time Delay	Set the amount of Time to Delay the Media

Bluefish444 Media Output Settings

Each Bluefish444 Media Output object that you create exposes the following configuration settings.

Property	Description
Output Type	Select Fill or Fill + Key output type
Device	Select the Bluefish444 Device to attach to
Destination	Set Single Link, Dual Link, Quad Link SDI Interface
Resolution	Set Video Modes Resolution
Standard	Set Progressive or Interlaced
Frame Rate	Set the Video Modes Frame Rate
Reference	Not used in Bluefish444 configuration, see Bf444GenericSettings Blueprint

Output	Description
Output with Auto Circulating	When enabled, the Unreal Engine buffers its output frames before sending them to the Bluefish444 card. This may improve the smoothness of the video signal, at the cost of some latency. Leave this option disabled to minimize latency, at the risk of seeing interruptions in the output signal.
Timecode Format	Determines whether the Unreal Engine should embed timecode in the output feed, and which timecode format it should use.
Pixel Format	Set the pixel format sent to the Bluefish444 card, for Fill+Key format must include Alpha
Output Audio	Check to enable Audio Output
Audio Buffer Size	Size of the Buffer which holds the number of audio samples
Num Output Audio Channels	Set the number of Audio Channels sent to the Bluefish444 hardware
Output Audio Depth	Set the bit depth of the Audio sent to the Bluefish444 hardware
Audio Sample Rate	Set the Sample Rate of the Audio sent to the Bluefish444 hardware
ADVANCED	
Output in 3GLevel B	Check to enable 3G Signals to be sent as 3G Level B
Invert Key Output	Invert the Key channel
Number Of BF444Buffers	The number of Bluefish444 Buffers used, min 1 for lowest latency
Interlaced Fields Timecode Need to Match	When producing an interlaced video feed, this setting determines whether the timecode values for both fields in a single interlaced frame need to match.

Output	Description
Number Of Texture Buffers	The number of Texture Buffers used, min 2

Debug	Description
Log Drop Frame	When enabled dropped frames will be displayed in the UE terminal
Burn Frame Timecode	Encode Timecode in Texel

HW Keyer	Description
HWKeyer Enable	When enabled the Bluefish444 hardware keyer will composite the RGBA output with the corresponding Input signal
HWKeyer Key Over Input	When using the HW Keyer Enable Key over Input
HWKeyer Use Input ANC	When using the HW Keyer Enable Ancillary data from the SDI input
HWKeyer Data Is Premul	When using the HW Keyer decide if the Key is Premultiplied or not

Bluefish444 Generic Settings BluePrint

Bluefish444 provide a Generic Settings Blueprint to allow the generic/global card settings to be setup.

Property	Description
Device	Select the Bluefish44 hardware to configure

Bypass Relay Watchdog	Description
Enable Bypass Relay Watch Dog	Enable the Bypass Relay Watchdog Timer, if this does not respond in the period defined then the Bypass Relays will be enabled
Watch Dog Wait Interrupts	Number of Interrupts to wait until the Watchdog is triggered

Timecode External Ltc Source	Description
External Ltc Source	Set the External LTC timecode source connector; Leave As Is / Ref In / Interlock

Reference Options	Description
Reference Source	Set the Reference source connector; Leave As Is / Free Run / External Ref / External Ref Aux / Interlock / Input
Reference Source HOffset	Set a Horizontal Offset Value
Reference Source VOffset	Set a Vertical Offset Value
Reference Input Connector	Set the Input connector to use as a Reference
Reference Out Connector	Configure the Reference Out Connector function; Leave As Is / SPG / Interlock / LTC / Genlock Pass-through
Spg Signal	Set the SPG video standard
Interlock Output Signal	Configure the Interlock Out Connector function; Leave As Is / SPG / Interlock / LTC