# BluOS Player Module v2.0.0 Guide

## Description

This module is compatible with devices running BluOS software version 2.13.7 or later.  It provides playback control, now playing information, service browsing (with search where capable), preset recall, and current play queue information. A single instance of this module can be used to control a single statically-assigned player, or it can be used to select different players at run time – allowing the programmer to add a copy of the module per user interface instead of per player.  The latter method is illustrated in the accompanying example program.  The module was designed for use with a Subpage Reference List Smart Graphics Object in order to enhance the user's browsing experience.  An additional Dynamic Button List Object is used for a popup-style context menu under certain conditions.  The module package includes the BluOS Player SIMPL Windows module (.umc), the BluOS Device SIMPL+ module (.usp/ush), the BluOS SIMPL# library (.clz), an example program (.smw), and example VTPro-e user interface projects (.vtp).  The module files (.umc, .usp/ush, and .clz) should either be copied into the same directory as the program you are working on, or placed into the appropriate centralized user modules folder so that they are available in SIMPL Windows.  The module is supported on 3-series processors, and has been tested on firmware v1.502.3079.17847 (Jun 07 2017).

## Equipment Setup

The native BluOS App for iOS/Android must be used to initially configure the player, set up Presets, save Playlists, add Network Shares, Upgrade the BluOS software, etc.  The player must be upgraded to BluOS version 2.13.7 or later in order for the module to communicate with the player properly.  The module requires you to know the IP address of the player.  You can find this by selecting the player and going to Help->Diagnostics from within the BluOS App.  It is highly recommended that DHCP reservations are set up for all players on the network so that new addresses are not automatically assigned in the future by the network's DHCP server/router.

# Module Implementation Tips

## VTPro-e

It is recommended that you examine and even copy from the included example user interface projects when implementing in your own projects. Take care when adjusting the press and visibility joins to fit your join scheme. Make sure the Main List and Context Menu List are copied into your projects and compiled before synchronizing the Smart Graphics Extenders in SIMPL Windows.

## SIMPL Windows

The module requires that the IP Address parameter be filled out, but this address can be overridden at run time by the **[Device_IP_Address]** serial input. The value of this input must include the port number as shown in the example (i.e. <IP Address>:<Port Number>). The TCP port number determines which player the module will control within units that contain multiple outputs (like the CI850). For single-output units, port 11000 (Output 1) should be used. For outputs 2 thru 4, use ports 11010, 11020 and 11030, respectively.

It is recommended that the **Enable_Polling** input be driven low when the player is not in use by any user interfaces. While this signal is high, the module will perform a Status poll on the player once per second in order to populate the user interface with playback metadata. Note that the **[Device_IP_Address]** can be changed while the **Enable_Polling** signal is being held high.

To quickly have SIMPL fill out all the signal names for you, open the module in detail view and press Ctrl-Shift-B. You can then use F9 to change the signal prefix. When connecting the module to your user interface hardware definition, carefully attach the standard VTPro-e joins as defined by your project. Then expand each List section of the module one-by-one. Highlight all signals in the group, and copy/paste them onto the corresponding Smart Graphic Extender section. Click on the first item in the group, hold Shift, and press the End key to highlight all visible signals from the first item to the end of the module definition. To adjust your selection to include only the items in the group you want to copy, continue to hold shift and click on the last item in the group.

# Module Definition Details

**Parameters**

| Name | Description | Notes/Examples |
|---|---|---|
| IP Address | String parameter that specifies the IP address of the player this module will control by default. | *Note that since players use DHCP by default, it is highly recommended to set up DHCP leases in the router or DHCP server so that this address does not get changed at run time. This parameter will be overridden by the [Device_IP_Address] input if it ever changes at run time.* |
| Output Port | String parameter with a list of values specifying which TCP port the module will communicate to the specified IP address on. This determines which Output the module will control on multi-zone units such as the CI580. | *This parameter will be overridden by the [Device_IP_Address] input if it ever changes at run time.* |
| Reference | String parameter for reference only. | Entering "DEBUG" in this parameter will enable debug mode on the module which will print statements to the processor's console output at run time. It is highly recommended to NOT leave a system in debug mode. |

## Inputs

**Standard Digital Inputs**

| Name | Description | Notes/Examples |
|---|---|---|
| Enable_Polling | Latch this input high to enable Status polling of the player at an interval of once per second. | *It is recommended that this signal only be latched high while a user interface is actually viewing the player's controls. Note that when this signal goes high, the module performs a Services query and an initial Browse query before starting the polling cycle, so the first poll request will not be sent for a few seconds after setting this signal high.* |
| Browse_Navigation | This signal puts the module into Browse mode which updates the Main List and other visibility elements on the user interface. | *Browse mode is the default navigation mode which is set the very first time the Enable_Polling signal is driven high.* |
| Presets_Navigation | This signal puts the module into Presets mode which updates the Main List and other visibility elements on the user interface. | |

| | | |
|---|---|---|
| Queue_Navigation | This signal puts the module into Queue mode which updates the Main List and other visibility elements on the user interface. | *When selected from a different mode, the current Queue is queried and if the currently playing track is in the Queue, the Main List will scroll to that item. If not, the top 50 items of the Queue are displayed in the Main List. The Browse_Next and Browse_Back signals are used to scroll between "pages" of the Queue – 50 items at a time.* |
| Play/Pause | Pulse to send either the play or pause command to the player based on feedback as to the unit's current state. | |
| [Play] | Pulse to send a discreet Play command to the player. | *Optional signal.* |
| [Pause] | Pulse to send a discreet Pause command to the player. | *Optional signal.* |
| Back | Pulse to send the playback Back command (previous track) to the player. | *The Back command does not work under certain player conditions. The Back_Button_Visibility output indicates whether or not the Back command can be issued.* |
| Skip | Pulse to send the playback Skip commands (next track) to the player. | *The Skip command does not work under certain player conditions. The Skip_Button_Visibility output indicates whether or not the Skip command can be issued.* |
| Love | Pulse to send the Love command to the player. Often triggers the Notification Popup subpage to appear. | *The Love command only applies to specific services. The Love_Button_Visibility output indicates that the currently playing service can accept the Love command.* |
| Ban | Pulse to send the Ban command to the player. Often triggers the Notification Popup subpage to appear. | *The Ban command only applies to specific services. The Ban_Button_Visibility output indicates that the currently playing service can accept the Ban command.* |
| Shuffle | Pulse to send the Shuffle command based on feedback from the player. | |
| Repeat | Pulse to send the Repeat command based on feedback from the player. | *This input will cycle between Off, Repeat All, and Repeat One.* |
| Now_Playing_Toggle | Pulse to toggle visibility of the Now Playing popup subpage. | |
| Browse_Home | Pulse to perform a top-level Browse query which updates the Main List to display the root | |

| Name | Description | Notes/Examples |
|---|---|---|
| | browsing level consisting of the services set up on the player. | |
| Browse_Next | Pulse to display the next page of Browse results. | *This signal only pertains to certain browsing contexts. The Browse_Next_Button_Visibility output indicates that the current context has a next page.* |
| Browse_Back | Pulse to display the previous page of Browse results. | *This signal only pertains to certain browsing contexts. The Browse_Back_Visibility output indicates that the current context has a previous page.* |
| Search_Toggle | Pulse to toggle the Search Text Entry popup subpage. | |
| Search_Subpage_Transition_Complete | Input coming from the user interface indicating that the Search Text Entry subpage has appeared. | *Used to set the focus to the text entry box, popping up the user interface's keyboard.* |
| Search_Enter | Pulse to send the current Search_Text_Entry serial input in a search query to the player. | |
| Search_Clear | Pulse to clear the current Search_Text_Entry serial input. | *Also clears the Search_Text_Feedback output for visual confirmation.* |
| Context_Menu_Close | Pulse to close the Context Menu popup subpage if open. | |
| Error_OK | Pulse to clear the Error Message popup subpage. | *Often during error conditions, polling is interrupted. This input will restart the polling cycle.* |
| [Volume_Up] | Momentary input to raise the volume of the player. | *Optional signal.* |
| [Volume_Down] | Momentary input to lower the volume of the player. | *Optional signal.* |

**Standard Analog Inputs**

| Name | Description | Notes/Examples |
|---|---|---|
| [Volume_Slider] | Analog input for setting the volume of the player to a specific percentage. | *Value range (0%-100% or 0d-65535d). Optional signal.* |

**Standard Serial Inputs**

| Name | Description | Notes/Examples |
|---|---|---|
| Search_Text_Entry | Serial input intended to be driven by the Output Text Serial Join of a Text Entry object in VTPro-e. This serial input is sent in a search query when the Search_Enter digital input is pulsed. | |
| [Device_IP_Address] | Overrides the IP Address parameter (or any previously sent address on this signal) and will cause the module to start | *Must be of the format: <IP Address>:<TCP Port> Optional signal that does not need to be defined if not controlling* |

| | communicating with the new address. | *more than one player from the module.* |
|---|---|---|

## Main List Digital Inputs

| Name | Description | Notes/Examples |
|---|---|---|
| Main_List_SelectItemXX | Selects the corresponding item in the Main List. | *The action that follows depends on the navigation mode.  In Browse mode, the action is determined by the item type.  In the other modes, the item will start playback.  If an item's text is surround by square braces "[ ]", it is simply a text divider for readability and will not perform an action.* |
| Main_List_Play_All_ItemXX | Begins playback from the corresponding item and adds the subsequent items in the list to the Queue. | *Only functions with specific item types, such as albums, that have this capability.* |
| Main_List_Context_ItemXX | Brings up the context menu for the corresponding item. | *The options that appear in the context menu list depend on the service and item type.* |

## Context Menu List Digital Inputs

| Name | Description | Notes/Examples |
|---|---|---|
| Context_Menu_Select_ItemX | Selects the corresponding item in the Context Menu List. | *The options that appear in the context menu list depend on the service and item type.* |

## Outputs

**Standard Digital Outputs**

| Name | Description | Notes/Examples |
|---|---|---|
| Browse_Navigation_Selected | Indicates that the module is in Browse mode. | *Used to display/hide the Browse Home button.* |
| Presets_Navigation_Selected | Indicates that the module is in Presets mode. | |
| Queue_Navigation_Selected | Indicates that the module is in Presets mode. | |
| Back_Button_Visibility | Indicates that current playback supports the Back command. | *Used to display/hide the Back (previous track) button.* |
| Skip_Button_Visiblity | Indicates that current playback supports the Skip command. | *Used to display/hide the Skip (next track) button.* |
| Skip_Count_Visibility | Indicates that current playback status has a limited number of skips. | *Used to display/hide the skip count.* |
| Love_Button_Visibility | Indicates that the current playback has an associated Love command that can be sent to the service to indicate that the user likes the current track. | *Used to display/hide the Love button.* |
| Ban_Button_Visibility | Indicates that the current playback has an associated Ban command that can be sent to the service to indicate that the user dislikes the current track. | *Used to display/hide the Ban button.* |
| Shuffle_Button_Visibility | Indicates that the current playback supports the shuffle function. | *Used to display/hide the Shuffle button.* |
| Repeat_Button_Visibility | Indicates that the current playback supports the Repeat function. | *Used to display/hide the Repeat button.* |
| Now_Playing_Visibility | Indicates that the Now Playing subpage should be visible. | *Used to display/hide the Now Playing subpage.* |
| Track_Length_Visiblity | Indicates that the current playback contains track progress information. | *Used to display/hide the track progress gauge and times.* |
| Notification_Visibility | Indicates that an action has prompted a notification message to be displayed to the user. | *Used to display/hide the Notification Popup subpage.* |
| Browse_Service_Icon_Visibility | Indicates that the current Browse context has an associated Icon image that should be displayed. | *Used to display/hide the Service Icon image object.* |
| Browse_Next_Button_Visibility | Indicates that the current Browse context has a next page of additional items to be displayed. | *Used to display/hide the Browse Next button. Only functions when the module is in Browse mode.* |
| Browse_Back_Button_Visibility | Indicates that the current Browse context has a previous page of items to be displayed. | *Used to display/hide the Browse Back button. Only functions when the module is in Browse mode.* |
| Search_Button_Visibility | Indicates that the current Browse context is capable of the Search function. | *Used to display/hide the Search button. Note that not all services are searchable.* |

| Search_Entry_Visibility | Indicates that the Search Text Entry subpage should be visible on the user interface. | Used to display/hide the Search Text Entry subpage. |
|---|---|---|
| Search_Entry_Focus | Digital output used to set the focus to the Search Text Entry object when it appears on the user interface.  By default, this displays the user interface's keyboard. | Without this signal, the text entry object will have to be pressed in order for the keyboard to be displayed. |
| Context_Menu_Visibility | Indicates that the Context Menu subpage should be displayed on the user interface. | Used to display/hide the Context Menu subpage. |
| Error_Visibility | Indicates that the module or device has thrown an error that should be displayed on the user interface. | Used to display/hide the Error Message subpage. |

**Standard Analog Outputs**

| Name | Description | Notes/Examples |
|---|---|---|
| Play_Button_Mode | Analog output indicating the playback state of the player. | Used to set the mode feedback of the Play/Pause button.  There are three modes: 0=paused, 1=playing, 2=connecting. |
| Shuffle_Button_Mode | Analog output indicating the shuffle state of the player. | Used to set the mode feedback of the Shuffle button.  There are two modes: 0=shuffle off, 1=shuffle on. |
| Repeat_Button_Mode | Analog output indicating the repeat state of the player. | Used to set the mode feedback of the Repeat button.  There are three modes: 0=repeat all, 1=repeat track, 2=repeat off. |
| Track_Progress | Analog output indicating the current track's playback progress percentage. | Used to display the track progress gauge.  Visibility is determined by the Track_Length_Visibility digital output because not all playback states support the progress bar. |
| [Volume_Feedback] | Analog feedback indicating the player's current volume. | Value range (0%-100% or 0d-65535d). Optional signal. |

**Standard Serial Outputs**

| Name | Description | Notes/Examples |
|---|---|---|
| [Player_Name] | Serial feedback containing the name of the player that the module is currently controlling. | Optional signal. |
| Main_List_Title | Indicates the current mode (or service when in Browse mode). | |
| Now_Playing_TitleX | Serial feedback for displaying current playback information. | Titles 2 & 3 may be blank in certain playback scenarios. |
| Status_Image_Url | Serial feedback containing the URL for the Now Playing image associated with the current track or station. | |
| Station_Image_Url | Serial feedback containing the URL for the Now Playing image associated with the current station. | If the currently playing service is not a station, the service's icon will be output here instead. |

| | | |
|---|---|---|
| Service_Name | Indicates the currently playing service's name. | |
| Track_Current_Time | Indicates the currently playing track's elapsed time. | *Visibility is determined by the Track_Length_Visibility digital output because not all playback states support track progress.* |
| Track_Total_Time | Indicates the currently playing track's total time. | *Visibility is determined by the Track_Length_Visibility digital output because not all playback states support track progress.* |
| Skip_Count | Indicates the number of remaining Skips allowed within the current service, if applicable. | *The Skip_Count_Visibility signal indicates whether or not this feedback signal should be displayed.* |
| Browse_Service_Icon_Url | Serial feedback containing the URL for the service that the Browse context is currently focused on. | *The Browse_Service_Icon_Visibility signal is used to display the Browse service icon image object.* |
| Search_Text_Feedback | Serial feedback for the text that should populate the Search Text Entry object. | *This signal is only used to clear the displayed text when the Search_Clear button is pressed.* |
| Error_Message | Serial feedback indicating that the player or module has thrown an error to be displayed on the user interface. | *The Error_Visibility signal indicates that there is an error to be displayed.* |

**Main List Digital Outputs**

| Name | Description | Notes/Examples |
|---|---|---|
| Main_List_Image_VisibilityXX | Indicates that the corresponding Main List item has an associated Image that should be displayed. | *Used to display/hide the image object for the corresponding Main List Item.* |
| Main_List_Play_All_VisibilityXX | Indicates that the corresponding Main List Item has Play All capabilities. | *Used to display/hide the Play All button for the corresponding Main List Item.* |
| Main_List_Context_VisibilityXX | Indicates that the corresponding Main List Item has an associated Context Menu for further functionality. | *Used to display/hide the Context button (...) for the corresponding Main List Item.* |

**Main List Analog Outputs**

| Name | Description | Notes/Examples |
|---|---|---|
| Main_List_Scroll_To_Item | Signal to control the scroll focus of the Main List. | *Should be connected to the 'Scroll To Item' input on the Main List Subpage Reference List Smart Object extender.* |
| Main_List_Number_of_Items | Indicates the number of items that the Main List contains in the current mode/browse context. | *Should be connected to the 'Set Number of Items' input on the Main List Subpage Reference List Smart Object extender.* |
| Main_List_Type_ItemXX | Sets the mode feedback for the corresponding Main List Item. | *There are 10 modes: 0=link, 1=audio, 2=artist, 3=composer, 4=album, 5=playlist, 6=track, 7=text, 8=section, 9=folder.* |

**Main List Serial Outputs**

| Name | Description | Notes/Examples |
|---|---|---|
| Main_List_Text1_ItemXX | Serial feedback for the primary description of the corresponding Main List Item. | |
| Main_List_Text2_ItemXX | Serial feedback for the secondary description of the corresponding Main List Item. | |
| Main_List_Image_Url_ItemXX | Serial feedback containing the image URL for the corresponding Main List Item. | |

**Context Menu List Serial Outputs**

| Name | Description | Notes/Examples |
|---|---|---|
| Context_Menu_List_Text_ItemX | Serial feedback for the function description of the corresponding Context Menu List Item. | |

**Context Menu List Analog Outputs**

| Name | Description | Notes/Examples |
|---|---|---|
| Context_Menu_List_Number_of_Items | Indicates the number of items that the Context Menu List contains in the current context. | |

# Known Issues

## KKBox

Currently, it is not recommended to use the KKBox service with this module. There are problems with the multi-byte characters that are often returned by this service. These responses have been seen to 'corrupt' the text within the Main List and make the items misaligned until a reset of the user interface is performed.