

## C-Bus Multi Room Audio

### Application Note #3 – Configuring MRA with PICED Software

If you have a touchscreen in your installation that will be used to control the MRA system, you can modify the existing PICED project to include MRA parameters.

#### Example Project

Before we begin, we need to compile a list of each **Source** (an input into the MRA system) and **Zone** (an area where an amplifier is installed). The tables below show the sources and zones for the example project. These examples are used throughout the series of MRA programming application notes to help you understand the MRA product.

In the Inputs table, the left column shows the source name, which is fixed in the MRA world. The middle column shows the type of input, while the right column shows a 'real world' name or description applied for easy identification. The Outputs table shows eight zones (or locations) for the example project.

#### Inputs (Sources)

Source	Type	Description
Local*	Line-level input on amplifier	iPod
Analogue 1	Line-level input #1 into Matrix Switcher	Cable TV
Analogue 2	Line-level input #2 into Matrix Switcher	DVD
Analogue 4/ AUX**	RCA sockets on the rear panel of the Matrix Switcher 3.5mm AUX input socket on front panel of Matrix Switcher	AUX
Tuner 1	Internal AM/FM tuner	Radio 1
Tuner 2	Internal AM/FM tuner	Radio 2
Streaming	Internal Streaming Module (deluxe model of Matrix Switcher only)	MP3

\* The local input on the amplifier does not need to be configured for operation, but its position is reserved. This allows for a total of six additional source inputs to be configured on the Matrix Switcher

\*\* The 3.5mm AUX socket on the front of the matrix switcher is internally (electrically) connected to Analogue Input 4. Either the AUX socket OR Analogue input 4 may be used, not both.

#### Outputs (Zones)

Zone No	Description (location)
Zone 1	Living
Zone 2	Bathroom
Zone 3	Bedroom
Zone 4	Ensuite

Zone No	Description (location)
Zone 5	Patio
Zone 6	Pool area
Zone 7	Dining
Zone 8	Study

## PICED Programming

This Application Note assumes that the installer has a basic level of understanding of PICED programming. The Application Note tries to cover the majority of fundamentals used for programming the C-Bus Multi-Room Audio (MRA) system. It does not cover any hardware configuration. Please refer to the *C-Bus Multi Room Audio Matrix Switcher and System Installation Instructions* for hardware installation and configuration.

**Prerequisite:** You must have an existing PICED project for either a B&W MKII touchscreen, C-Touch Spectrum Touchscreen or C-Touch Colour Touchscreen.

### Volume Control

Before we look at controlling the sources of the MRA system, we'll take a quick look at on/off and volume control. These are all C-Bus controls. The example used below has an On button, an OFF button, a volume + (increase) button, a volume – (decrease) button and a slider control.



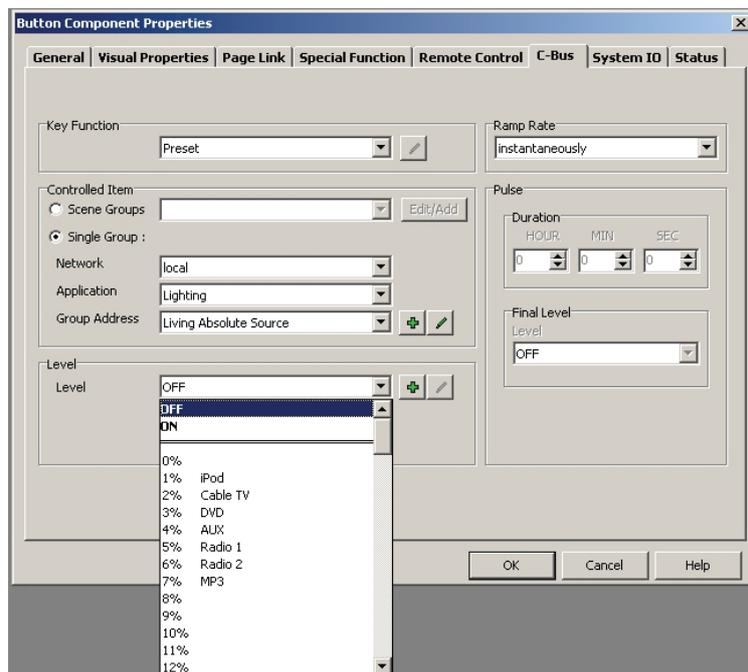
A great feature of the MRA is that when an On command is issued to the volume group, it doesn't jump to the maximum level. It goes to the most recent level. Below is a simple screen created with buttons and a slider.

### Source Selection

Source selection in the matrix switcher is performed using a lighting group address configured in the amplifier using C-Bus Toolkit. Source selection can be performed by either cycling through the sources (in the same manner as with a DLT switch) or by selecting an absolute source. The term absolute is better described as being a preset. When the amplifier for a zone is configured, the option is presented for applying a group to the 'Next Source', 'Previous Source' and 'Absolute Source'. The absolute value applies a percentage to each source. When the absolute group is set to the corresponding level, the corresponding source is selected.

#### Example:

Create a C-Bus button on your screen. On the C-Bus tab, use a 'Preset' as the key function. The example below uses the Living Absolute Source group (created previously in the application note for configuring MRA in Toolkit). Notice that you can set the source to a level (with the name adjacent to it).



You can create an individual button for each source (level) to offer a series of buttons on the screen with all the sources available for that zone.



## Media Control Application

The MRA system includes control using a Media Control Application. This application is significantly different from conventional C-Bus Applications and therefore, requires some explanation.

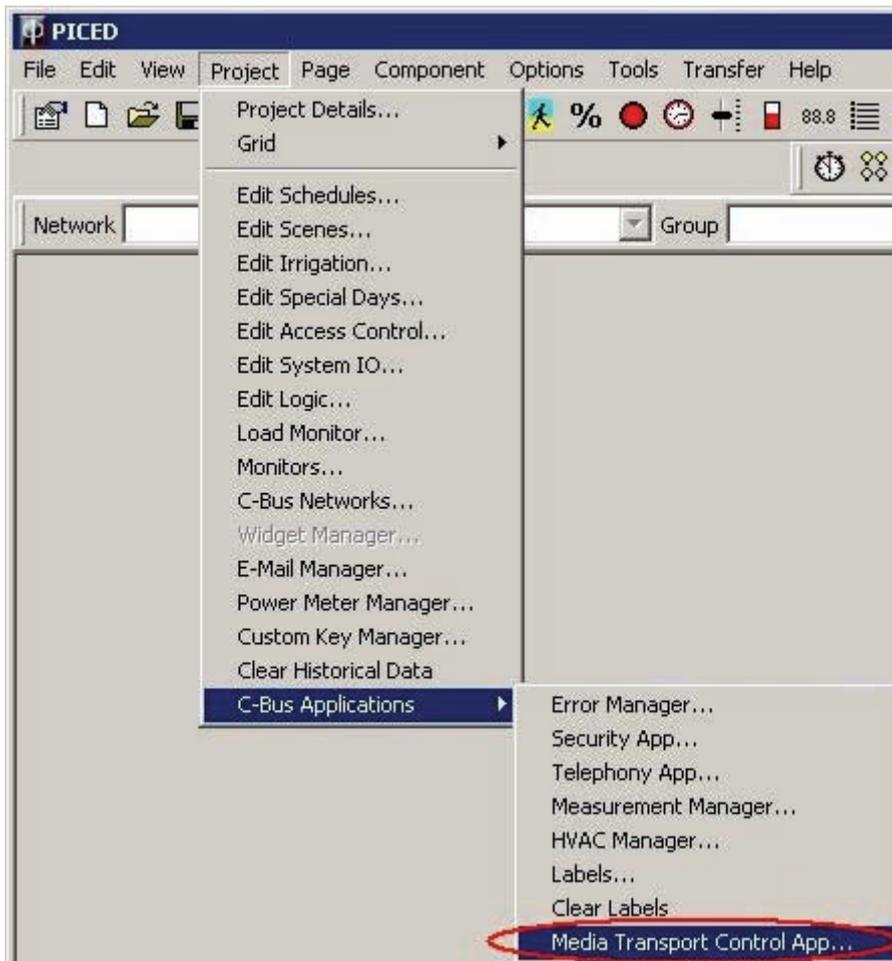
The two aspects to the Media Control Application are:

- feedback of information – radio station name, album title, artist name, playlist, etc.
- control using commands – play, stop, pause, next track, previous track, etc.

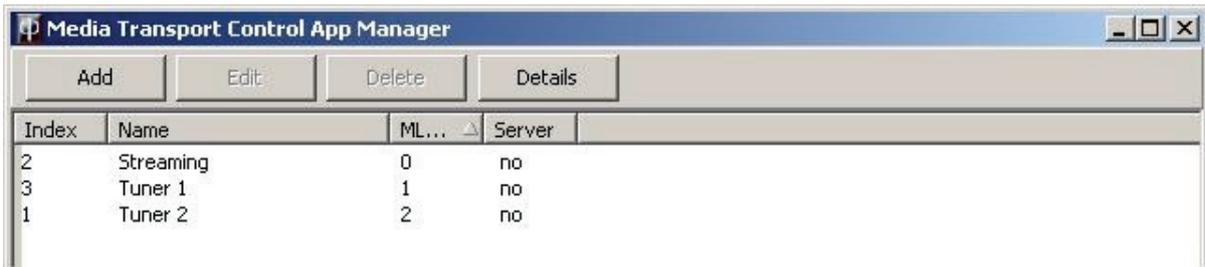
Before we dig too deeply into the specifics of the Media Control application, we need to configure the Media Link Groups. If you have previously used *Application Note #2- Configuring MRA with MARPA Software*, you will remember the general rule of thumb, to use the media link groups as follows:

- Media Link Group 000 – Streaming Audio
- Media Link Group 001 – Radio Tuner 1
- Media Link Group 002 – Radio Tuner 2

Open the Media Transport Control App manager window.



You will be asked if you want Media Link Groups to be automatically created for you. Assuming you have created your groups previously using Toolkit or MARPA, select 'yes'. The groups are extracted from the database for you.



**Note:** After the groups are created, you will see an option for the group(s) to act as servers. This is an advanced feature that is not supported on MRA and therefore should be set to No.

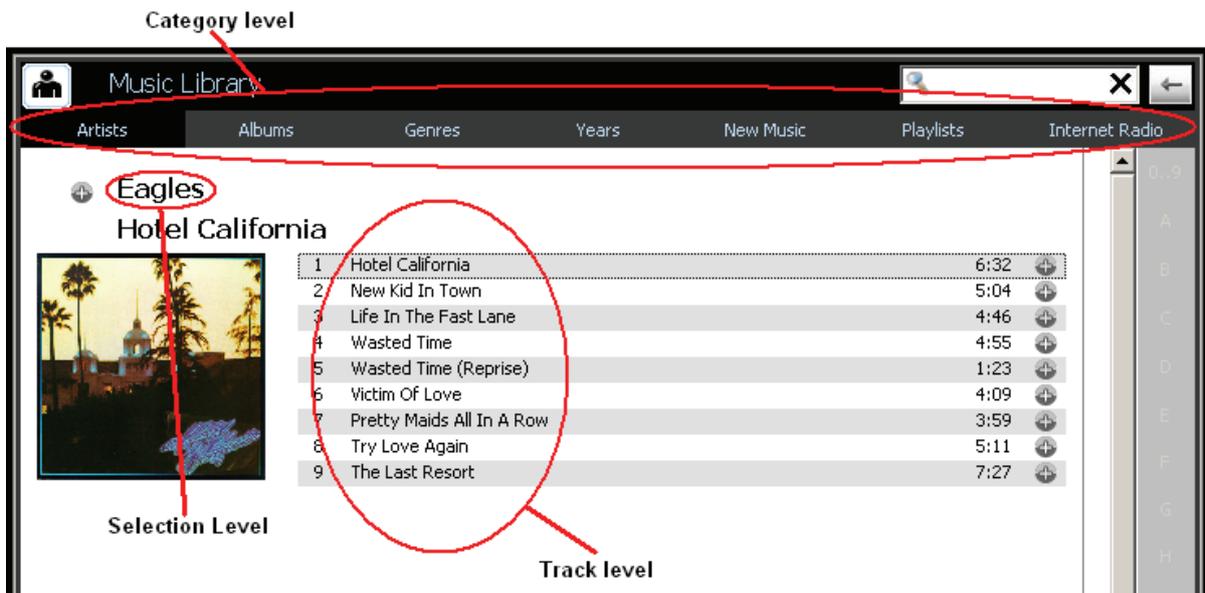
If you haven't previously created the Media Link Groups in Toolkit or MARPA, you can add them manually now. Note that the groups you add here will NOT be exported to Toolkit/MARPA.

### Media Control Hierarchy

The three levels of hierarchy for feedback and control of Media are:

1. Category
2. Selection
3. Track

When the MRA system is used in conjunction with C-Bus Ripple software (covered in Application Note #4), the hierarchy outlined is applied as follows:



Depending on the Category selected, the Selection and Track information changes. In the example above, the (current) Category is 'Artists' (darker tab on the left of the highlighted area), the (current) Selection is 'Eagles' and the (current) Track is 'Hotel California'.

The text for each of these hierarchal levels will be transmitted over C-Bus. Furthermore, the application supports more than just the current item text on each hierarchal level. The application supports the broadcast of the

- Current Item
- Next item
- Next Item +1
- Previous Item
- Previous Item -1

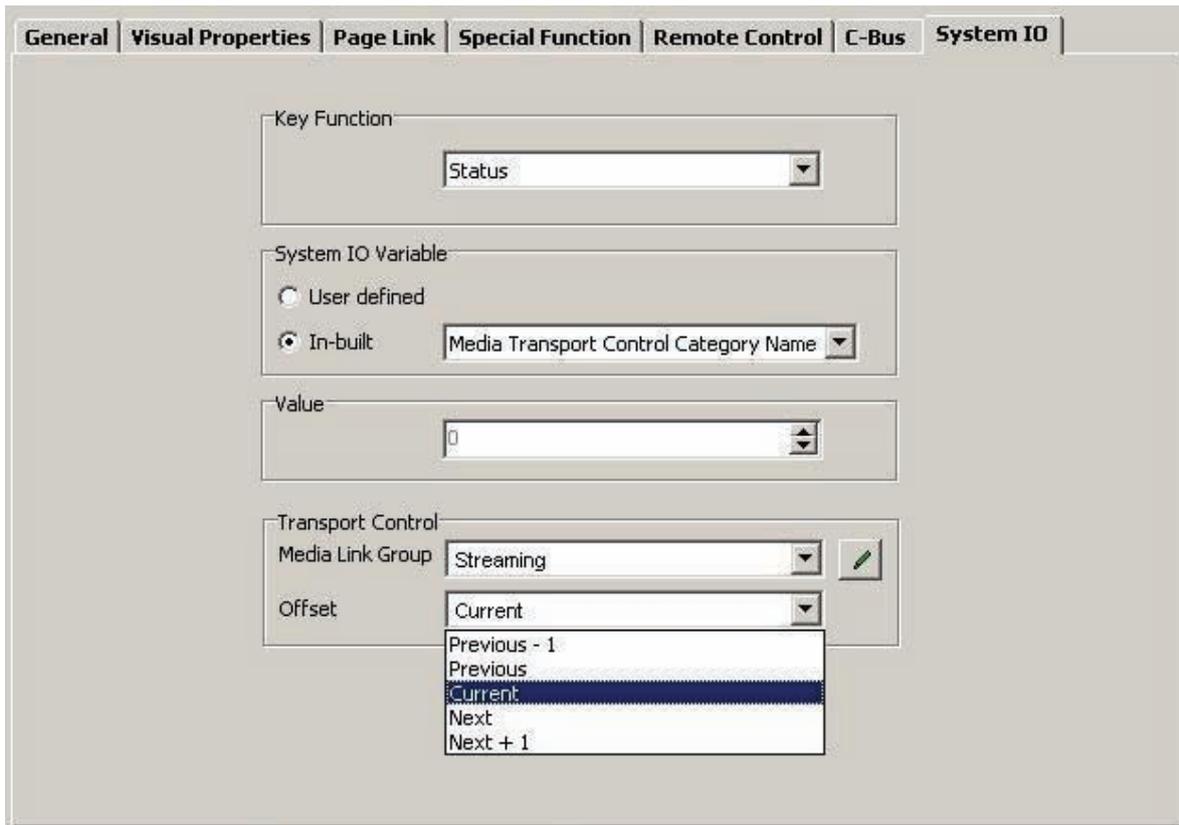
There can be a total of five strings of information per hierarchal level.

**Example**

Select a 'Level Indicator' from the components list and place it on your screen.

1. On the C-Bus tab, set the Key Function to 'unused.'
2. Select the System IO tab.
3. Set the Key Function to 'Status.'
4. Choose the In-built option from the System IO Variable section.
5. Scroll through the list to find and select 'Media Transport Control Category Name.'
6. Select 'Streaming' in the Media Link Group.
7. Select 'current' in the Offset list.

The following shows the configuration of the component described in the example.



**Note:** The Offset field shows the 'Current' Category Value. It is possible to configure this value to show the next or previous if desired.

**Media Transport Control – Category**

The Level 1 (Category) discussed in the previous section can be used with an absolute number. That is, instead of cycling through the list one at a time, you can go directly to the one you want. The Media Transport Control Category number is that absolute number. This field shows the current Category number. The screen below shows the tabs found in C-Bus Ripple software with their respective (absolute) numbers.



**Note:** Although it is possible to hide some of the Category contents in C-Bus Ripple software, the absolute numbers will not change. All category items will be broadcast out onto C-Bus regardless of whether they are visible in C-Bus Ripple or not.

## Media Transport Control – Category Name

This displays the Category Name. This can be the current Category item in the list, or previous and next items in the list, for example:

If the Current Category item is 'Years', the following would apply when using C-Bus Ripple software.

Previous -1:	Albums
Previous:	Genres
Current:	Years
Next:	New music
Next +1:	Playlists

## Media Transport Control – Selection

Selection forms the 2nd level of information for broadcast onto C-Bus. What is currently present in the Selection field depends upon what is selected as the current field in the Category (this doesn't apply for radio tuners, but we'll cover that later).

Like with Category, you can have a Selection number representing the current item in the list.

If we have chosen 'Years' as the current Category item, the Selection might look something like this:

- 1) 1981 (Previous -1)
- 2) 1982 (Previous)
- 3) 1983 (Current)
- 4) 1984 (Next)
- 5) 1985 (Next +1)

As 1983 is the current Selection, then the Media Transport Control Selection value would be 3.

## Media Transport Control – Selection Name

The Selection Name is the text of the selected item in the list. Using the previous example, the current Selection is 1983. If the Category is changed to 'Artists', the Selection list might look like this:

- 1) AC/DC (Current)
- 2) Eagles (Next)
- 3) Fallout Boy (Next +1)

Due to the dynamic nature of this field, there is no previous item or previous item -1 until you scroll through the list. For example, if you request 'Next Selection,' you will see the following:

- 1) AC/DC (Previous)
- 2) Eagles (Current)
- 3) Fallout Boy (Next)
- 4) Snow Patrol (Next +1)

'Eagles' was the 'Next' but becomes 'Current'. 'AC/DC' was 'Current', but now becomes 'Previous,' and so on.

## Media Transport Control – Track

'Track' forms the 3rd level. Like Selection, this is dynamic and will change depending upon the current Category and Selection. As the name suggests, it is usually used to display and control a music track associated with the MRA. The Media Transport Control Track (found on the In-built System I/O) displays the current track number.

## Media Transport Control – Track Name

The Media Transport Control Track Name is fairly self-explanatory. Like Category and Selection, it is possible to display the 'current' Track as well as the 'Previous', 'Previous -1', 'Next' and 'Next +1'.

Note: While it appears that only a small number of tracks (5) can be displayed, this is due to the ability of C-Bus to handle large amounts of data. If you consider that every time you change a Category, C-Bus needs to handle 5 x Category Names + 5 x Selection Names + 5 x Track Names, C-Bus needs to handle the broadcast of 15 strings of information.

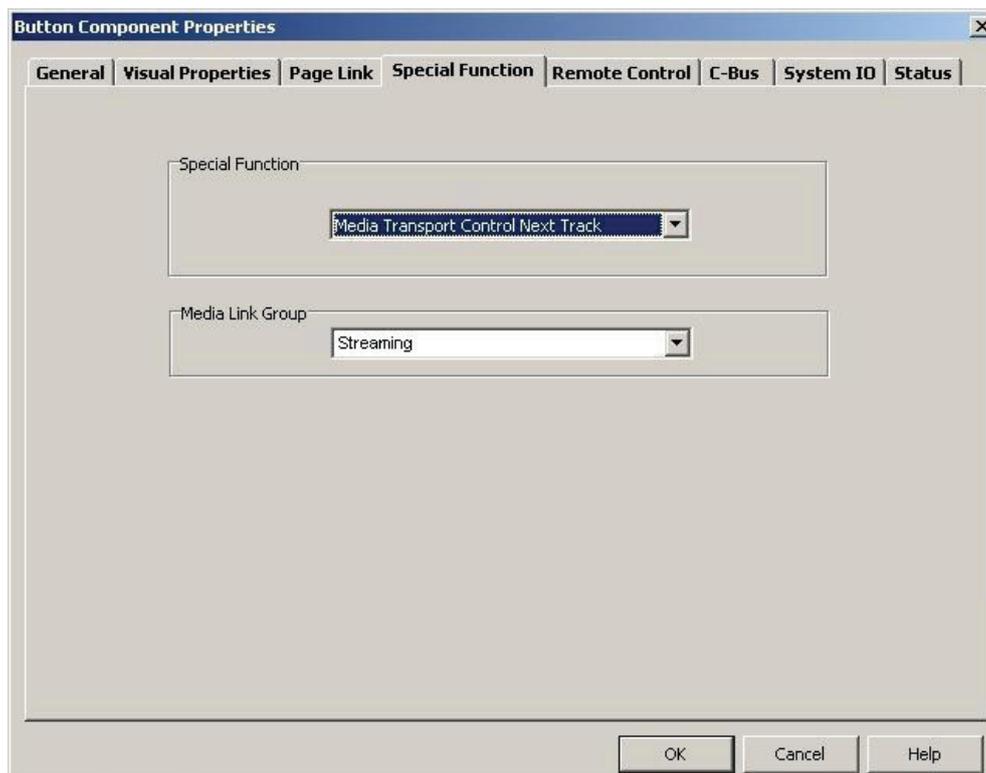
## Media Transport Control – Track Count

Track (Level 3) has one additional item, Count, that is not found in Category or Selection. This offers the ability to display how many tracks are present in the current play Queue. Refer to the Application note for C-Bus Ripple for further information about queues. This is useful if you wish to display both the track number and the total number of tracks. For example, if you have 12 tracks in the current queue and track 4 is playing, you could display the following:

Track [Media Transport Control Track] of [Media Transport Control Track Count], or  
Track 4 of 12

## Controlling Media Transport Control Levels 1, 2 and 3

Although we have covered creating fields for a Category, Selection and Track, we still need to be able to cycle through them. The commands for this navigation are found on the Special Function tab of the 'level indicator' component. Typically, you create a button to handle this. After creating a button, go to the Special Function tab and look through the Special Function list (see below). You must also set the Media Link Group you wish to control.



There are six controls found in the Special Function area. They relate directly to cycling through the Category, Selection and Tracks fields covered previously. The special functions are:

- Media Transport Control Next Category
- Media Transport Control Previous Category
- Media Transport Control Next Selection
- Media Transport Control Previous Selection
- Media Transport Control Next Track
- Media Transport Control Previous Track

## Controlling Radio Tuners

The two tuners in the Matrix Switcher are controlled using the Media Transport Control Application and can have information about them displayed in much the same way as streaming media. The tuners use only the 2nd level of Media Transport Control hierarchy (Selection) to display their data.

The actual data to be displayed is determined by the setup of the tuner presets configured in MARPA. It is possible to display either (1) the Station Name (the description given in MARPA), such as, XXXFM, or (2) the Station Frequency without kHz, MHz or the band (AM/FM), for example, 99.1 or (3) the Station Name and the Frequency, separated by a space XXXFM 99.1.

It is suggested that the two radio tuners occupy Media Link Groups 001 for Tuner 1 and 002 for Tuner 2 (with 000 for streaming). Using Next Selection and Previous Selection cycles through the radio stations.

## Control using Absolute Commands

Sometimes, you want to go directly to a Category or maybe a radio station without cycling through all of the options. You can navigate quickly by setting a preset to the Media Transport Control number (instead of getting the status) as shown in the screen below.

The screenshot shows a software interface with several tabs: General, Visual Properties, Page Link, Special Function, Remote Control, C-Bus, System IO, and Status. The 'System IO' tab is active. The configuration is as follows:

- Key Function:** A dropdown menu set to 'Preset'.
- System IO Variable:** Two radio buttons: 'User defined' (unselected) and 'In-built' (selected). Below 'In-built' is a dropdown menu set to 'Media Transport Control Selection'.
- Value:** A numeric input field set to '5'.
- Transport Control:** A dropdown menu set to 'Media Link Group Tuner 1' with a small edit icon to its right.

As you can see, we are setting the tuner to preset number 5, as previously defined in MARPA. You can then hard-code the name of the radio station on the button if you choose. Alternatively, you could put buttons across the top of the screen with the Category names (with hard-coded presets behind) instead of cycling through them.

## Real Media Transport Control

You have additional control and display options in the Category, Selection and Track components using C-Bus Ripple software. The following transport control functions are also found on the Inbuilt button under System IO tab/ System IO Variable.

- Media Transport Control Play (un-mute function for tuners)
- Media Transport Control Pause
- Media Transport Control Stop (mute function for tuners)
- Media Transport Control Fast Forward
- Media Transport Control Rewind

Media Transport Control Repeat

Media Transport Control Shuffle

Media Transport Control Power (not used in C-Bus Ripple but may be of use for 3rd party applications; the option is included for completeness of the protocol.)

## Writing Complex Code Using Logic

The logic engine available in the Colour C-Touch, C-Touch Spectrum and B&W MKII expands even further the power of the features of the MRA system. If you regularly program MRA systems, chances are you will create a template for your touchscreen(s) that you will want to re-use every time.

It is likely that for each project you will require different radio presets. Because there are multiple screens, the programming time for radio presets could be large. If you change any of the presets (say add one) using MARPA, you then need to go to each touchscreen and add the preset. There has to be a better way; in fact, there is!

Your system can be programmed to access the matrix switcher and then dynamically populate the screen with information retrieved. This process uses a command called 'enumerate' in logic. Below is an explanation of how this can be achieved. In order to see enumeration of radio presets, first create a trigger. Enumeration causes a lot of traffic on C-Bus and should be used sparingly.

Create a C-Bus Group called 'Get Radio Presets', then write the following code in logic:

```
once (getlightingstate("Get Radio Presets") = ON) then
begin
  GetTransportControlData(1,1,0);
  setlightingstate("Get Radio Presets", off);
end;
```

The above logic code looks for the group to be ON. If it is ON, the logic sends a request for information using the 'GetTransportControlData (a, b, c)' command, to the matrix switcher. There are some options (a, b and c) after the command that need to be explained.

'a' is the Media Link Group...for Tuner 1, this is 001 (or 1)

'b' is the data type... AKA the hierarchy level where...

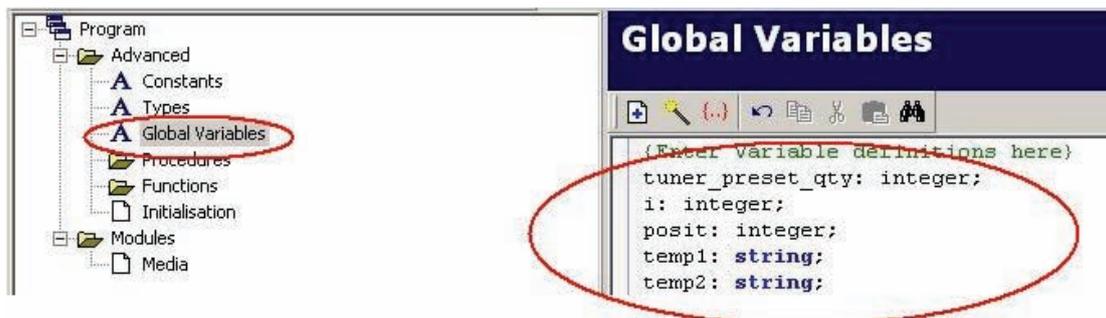
- 0 = Category
- 1 = Selection
- 2 = Track

'c' is the starting point from where to get data (0 is actually the first point).

**Note:** We are working with radio presets and need the 'Selection', so 'b' will be 1. If we wanted to start from the 5<sup>th</sup> preset, then this number would be 4.

Finally, we need to set the 'Get Radio Presets' group back to off. The request for data has been sent and (hopefully) the data received back.

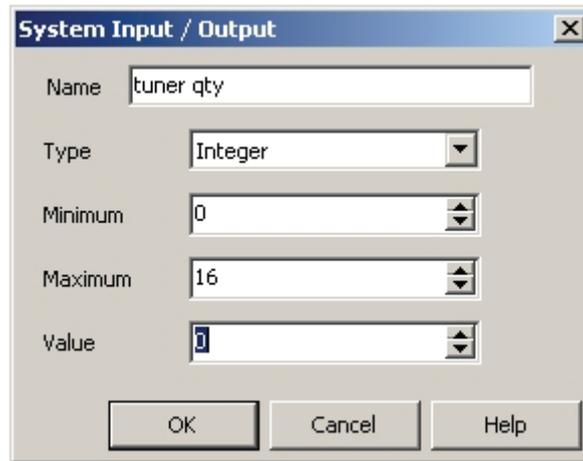
We now would like to know how many presets we have received back in our blast of data. First, we need to create an integer variable (we'll call it 'tuner\_preset\_qty') in the global variables definition section. While we're creating this, we should create all of the variables we need for this example, at the same time (see these and their types below in the right pane).



By using the following code, we can store the number of presets received, into the 'tuner\_preset\_qty' variable for further use.

```
tuner_preset_qty:= TransportControlDataCount;
```

We use the TransportControlDataCount command to get the information. It can be helpful to display this information on the screen. We can create a System Input / Output integer variable and place it there so we can display on the screen as shown below.



Use the code

```
SetIntSystemIO ("tuner qty", tuner_preset_qty);
```

To set the System IO 'tuner qty' to the variable 'tuner\_1\_qty' we now need to (1) write a chunk of code to loop through the data, (2) extract the parts we want and (3) display the information on the screen for some useful purpose.

```
If transportControlDataValid then
```

**Note:** This first line confirms that a complete lot of data has been received from the matrix switcher to process. Use the TransportControlDataValid command as a condition (outlined above). When there is valid data, the following occurs.

```
begin
```

```
for i:= 0 to tuner_preset_qty -1 do
```

Set a loop to count from 0 to the number of data records (tuner presets) that we discovered previously.

```
begin
```

```
TransportControlData(i,temp1);
```

Use the TransportControlData command to extract the first record and store it in the variable 'temp'.

**Note:** On the first loop, 'i' will be 0, so the first record will be extracted. 'i' will increment on each loop (until we reach the value in 'tuner\_preset\_qty'). Before we loop through to get the second record though, we need to do something with the first.

The data comes in a string that includes more than just the radio station id description. There are actually three pieces of information in a single line, with a comma separating them. The string will look like the following:

```
FM,XXXFM,97.8
```

Where the first part is the band (AM or FM), the second part is the station id (in this case, XXXFM) and the third is the radio frequency (97.8).

We are only interested in the station id and need to extract it from the string. First, we need to remove the band (and the comma between the band and id). This is done by using the following command:

```
copy (temp1, temp1, 4, 20);
```

This action takes data in the string 'temp', from the 4th character and discards the first three characters. In this example FM,XXXFM,97.8 is shortened to XXXFM,98.7 and stored back into the string 'temp' (over-writing the old string).

Now, we need to know the position of the comma in the string so we can take only the portion of the string up to the character before the comma. This is done by using the 'posit' command.

```
posit:=pos(',',temp1);
```

We store the position of the character (',') found in the data stored in 'temp1' in the global variable 'posit', so we can grab the data before this position. The value (obtained and stored in 'posit') in this case would be 6.

We again, use the copy command to extract the data we need.

```
copy(temp2, temp1, 1, (posit-1));
```

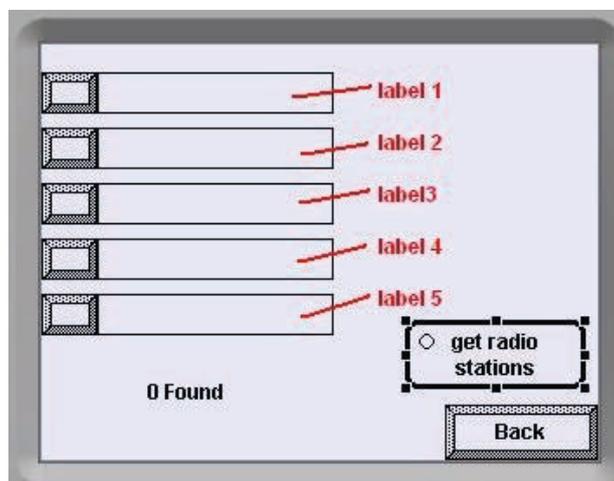
This takes all of the data from string position1, up to the position before the comma (posit-1) and stores it into the variable string called temp2.

We now need to take the data and display it on the screen. You can do this by copying the data from temp2, into a user System IO (create as many of these as needed). The below example has five of these created with the user systemIO's named label1, label2, label3, label4, label5. When 'i' = 0, then the extracted string data is put into label1, when 'i' is 1, then the data is put into label 2 and so on.

```
if i=0 then SetStringSystemIO("label1", temp2);
if i=1 then SetStringSystemIO("label2", temp2);
if i=2 then SetStringSystemIO("label3", temp2);
if i=3 then SetStringSystemIO("label4", temp2);
if i=4 then SetStringSystemIO("label5", temp2);
end;
```

Create a screen in your project (in this example, a B&W MKII touchscreen) with buttons(assign the preset number to the button) and labels as outlined above.

Your project screen could look something like this:



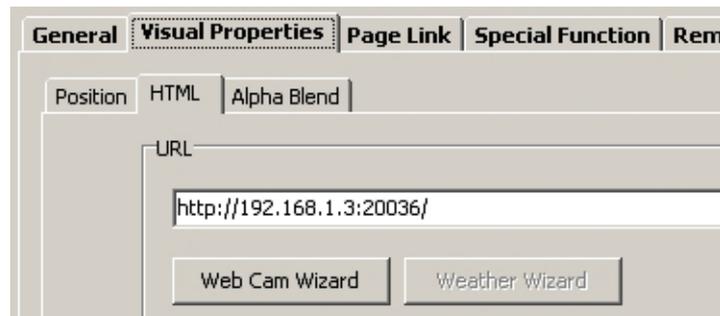
Enumeration could also be applied to Playlists too, if this is a desired way of simply navigating to absolutes.

If you are programming a colour C-Touch touchscreen, it's possible in logic to hide and reveal buttons and labels based upon data being present. For example, you can create buttons for 16 radio presets but only make the buttons visible for the number that actually exist. It's also possible to change the text on the button dynamically on logic on the colour touchscreen so the label could actually be on the button with the colour of the text.

## Viewing Cover Art

The colour touchscreen also offers the ability to show cover art using the webcam component. The URL for the cover-art needs to be the IP address of the PC running the C-Bus Ripple software. You also need to include the port address of the image of the appropriate audio stream (C-Bus Ripple supports up to 4 matrix switchers). If you are using only one matrix switcher, the port address will be 20036.

**Hint:** If you are not sure of the IP address or the port number, you can discover it quite easily by double clicking the cover art image in the 'now playing' window of C-Bus Ripple. This causes a web browser session to open, with the complete URL needed for the web cam component displayed as the address. Copy this address and paste it into the URL field of the web cam component.

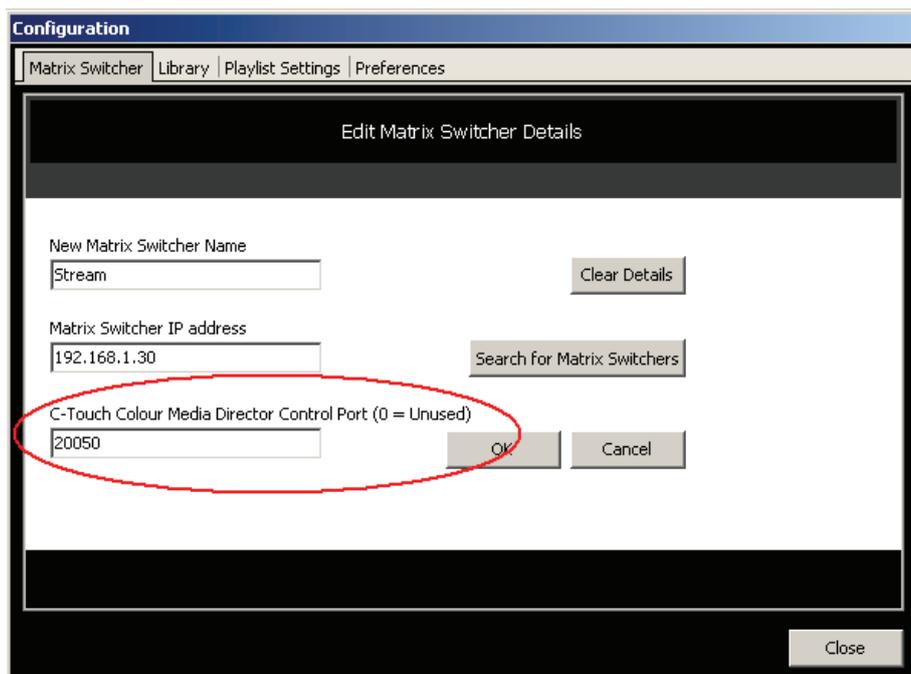


In the example below, the IP address of the PC running C-Bus Ripple is 192.168.1.3. Port numbers are assigned as follows:

20035	Cover-art for PC Queue
20036	Cover-art for Matrix Switcher 1
20037	Cover-art for Matrix Switcher 2
20038	Cover-art for Matrix Switcher 3
20039	Cover-art for Matrix Switcher 4

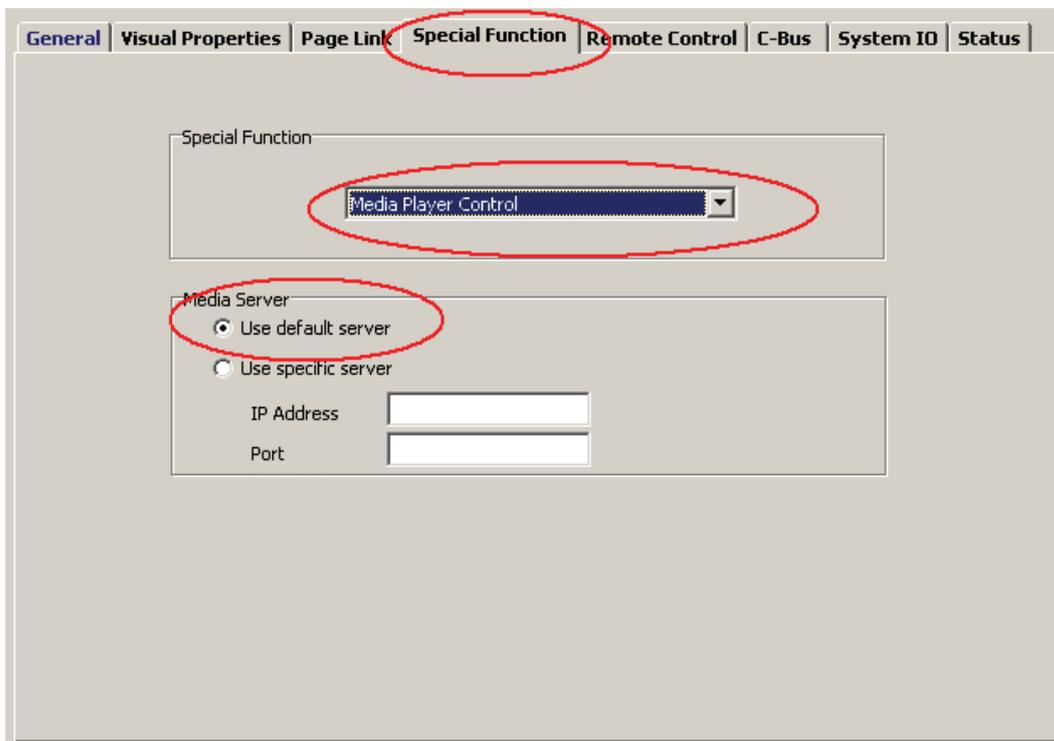
## Media Director Control

The C-Touch colour touchscreen includes a program that usually operates with a computer running Windows Media Player. This can also control C-Bus Ripple in the same way. In order to do this, a control port will need to be specified in C-Bus Ripple for the device being controlled as indicated below.



It is suggested port 20050 be used for this. If more than one is configured, use 20051 for the second device and so on.

When Media Director is configured to start in Piced, use a button programmed with a special function of 'Media Player Control'. Use the 'default server' option as highlighted in the image below. This will enable the Media Director to look for all connections on the network.



For further assistance in using this product, consult your nearest Clipsal Integrated Systems (CIS) Sales Representative or Technical Support Officer.



**Technical Support email:** cis.support@clipsal.com.au

Schneider Electric reserves the right to change specifications, modify designs and discontinue items without incurring obligation. Every effort is made to ensure that descriptions, specifications and other information in this application note are correct. No warranty is given in respect thereof, and the company shall not be liable for any error therein.

Copyright 2010. Schneider Electric. All rights reserved.