

BMXdigital Server 3.24

Release 3.24 of the BMXdigital Server software adds several new features and enhancements, especially in regards to network control. In conjunction with these new features a new function, License Codes, is being introduced in this software version. License Codes unlock proprietary or unique software features on a console by console basis.

This release also increases the number of router sources the console can address. Previously, only 256 sources could be addressed, but now up to 999 source names can be addressed by the console. These sources can come from one or multiple routers, since currently only the SAS 32KD router offers this quantity of sources in a single router. Each SAS 64000 router is limited to a maximum of 256 sources.

With this new addressing functionality, a new global Router command `Include` has been added to easily limit all controller modules in a console to a subset of possible sources.

To take advantage of these new network control features, a BMXdigital Automation Protocol document is available that lists the commands that are built into the BMXdigital server. All communications and control to the console is done through a TCP/IP client/server protocol.

The console functions that can be remotely controlled using this protocol include controlling most of the input channel switches along with the ability to set the individual fader levels.

The two-line channel displays can also be controlled to display 10-character static information (like the current source name in the top line), or scroll a text message of up to 160 characters in the bottom line. One specialized channel display function that could be easily implemented in the bottom line is a “countdown bar” to cue the board operator to the upcoming end of event.

The BMXdigital Automation Protocol is available at <ftp://ftp.pre.com>. Refer to Maintenance (page 5-1) for log in information. The PDF file of the protocol is inside the “BMXdigital Automation Protocol” folder, which is inside the main BMXdigital folder.

SERVERID.TXT

A new file, SERVERID.TXT, has been added to the SYS FILES folder on the BMXdigital Storage Card (Storage Card/Data/SysFiles). The file lists the Server ID number of the console’s Single Board Computer (SBC) that is located on the Session module. This file can be opened with Notepad, or any text editor, and simply consists of a single line of text:

```
Server ID is 014-000-188
```

The number (014-000-188 in this example) is the Server ID for that particular SBC. This number is used to generate a License Code for unlocking Automation and Router support features in the BMXdigital Server software.

NEW ENTRIES IN NQX.INI

There are two new entries in the NQX.INI file that relate to router and automation control. The first is the implementation of License Codes that unlock certain software features. The second is the Deadman control command line.

License Codes

There are two License Codes (Automation and Router) that unlock specific functionality in the BMXdigital Server software. A License Code is a nine digit number generated by a proprietary Harris software program from the unique nine digit Server ID number that identifies a specific Session Module's SBC.

When Automation and/or Router support was unlocked (purchased) prior to console shipment, the nine digit License Code number will appear in the appropriate License Code command line. If this automation and/or router support was not purchased prior to console shipment, then the features will be locked (the nine digit numbers will be all zeros). The License Code section of a locked NQX.INI file is shown below:

```
;-----  
;License Codes. Please Consult Customer Service  
AUTOMATION = 000-000-000  
ROUTER = 000-000-000  
;-----
```

To add automation and/or router support onto a console in the field, the NQX.INI file must be manually edited, to replace 000-000-000 with the License Code number, using Notepad, or other text-only editor, after the License Codes have been purchased.

To obtain a License Code, the Server ID number for the console (listed in the SEVERID.TXT file) must be supplied to a Harris Technical Services or Sales Representative so that they can generate the License Code for that BMXdigital console. An example of an unlocked NQX.INI file, with the License Codes entered, is shown below:

```
;-----  
;License Codes Please Consult Customer Service  
AUTOMATION = 493-865-799  
ROUTER = 594-875-899  
;-----
```

Deadman

This command line is an optional entry. In almost all cases, the setting will be DEADMAN = 1 (this is the default setting which is invoked if there is no entry in the NQX.INI file). This command reverts, to local control, those console features that have been "taken over" by an automation system if that automation system does not respond after a set amount of time. The command ensures that if the automation system crashes or is otherwise off-line, that immediate control of those automated functions can again be accomplished by a board operator.

When the Deadman command line is set to 1, or enabled, then control is automatically regained by the board operator after about 10 seconds of communication failures with the automation system. When the Deadman command line is set to

0, and the automation fails to respond, the automated functions will not revert to local control. This setting is intended for test or troubleshooting purposes only.

When the Deadman command is missing, the enabled state (1) is automatically invoked.

```
;-----  
; Automation Server Deadman control, Enable = 1 Disable = 0  
DEADMAN = 1  
;-----
```

NEW ENTRY IN THE ROUTERS.INI FILE

A global Include statement has been added to the [Routers] section of the ROUTERS.INI file. This functions just like the Include statement that is available under the [RouterControl] section, but it is applied to all router communication, whereas the [RouterControl] Include statement only affects a single controller. This means that individual Include statements are no longer required for each router controller module in order to limit the same sources from all the controller modules in a console.

Implied within this global Include statement is an exclude function, in that any sources that are not listed in the statement are excluded from being seen by any controller module in the console. [RouterControl] Include statements can then be used to further limit the sources for each controller module in the console.

[Routers]

Include_x=a-b,c-d,e,f,...

x is the number of the router. Each router has a unique number (starting with 1), that is assigned in this section of the ROUTER.INI file (e.g. Type_1=sas32kd, Type_2=sas64k, Type_3=sas64k).

a-b and *c-d* represent a range of sources for a particular SAS router that are available to any controller module in the console. A range can span both even and odd numbered sources, even when they are linked as stereo pairs. To insure stereo pairs are properly selected, make sure the last number is even. Masked sources are automatically excluded from the include list. If the Include entry is missing, then all possible unmasked sources (up to 999 in a 32 KD) will be alphabetically displayed on any controller module.

e,f,... represent individual SAS router sources that are available to any controller module in the console. Typically these will always be odd numbers for stereo pairing. The only time an even number would be listed is for a mono source.

An example of using this command is shown on the next page.

Example:

```
Include_2=1-126,337,339
```

This entry says: include these sources from Router 2, from source 1 up to source 126, and sources 337 and 339. Note that any, or all, of these inputs could be stereo-paired (Stereo linkage information is defined through the SAS software configuration, consult the SAS manual for additional details).

This command is especially useful when multiple stations share routers. By adding an Include statement for each router into each console's ROUTERS section, each station can have a preset range of sources available specifically to them that are effectively "invisible" to all of the other stations sharing the router(s). For instance, in the above example, inputs 127 - 250 could be used by another station, inputs 251-300 could be used by a third station, with inputs above 300 being shared as required by all stations.