Hellcat Utilities and Applications

Version 2.3

www.radisys.com
World Headquarters
5445 NE Dawson Creek Drive • Hillsboro, OR
97124 USA
Phone: 503-615-1100 • Fax: 503-615-1121
Toll-Free: 800-950-0044

International Headquarters
Gebouw Flevopoort • Televisieweg 1A
NL-1322 AC • Almere, The Netherlands
Phone: 31 36 5365595 • Fax: 31 36 5365620

RadiSys Microware Communications Software Division, Inc.
1500 N.W. 118th Street
Des Moines, Iowa 50325
515-223-8000

Revision C
April 2000

Copyright and publication information

Disclaimer

The information contained herein is believed to be accurate as of the date of publication. However, RadiSys Corporation will not be liable for any damages including indirect or consequential, from use of the OS-9 operating system, Microware-provided software, or reliance on the accuracy of this documentation. The information contained herein is subject to change without notice.

Reproduction notice

The software described in this document is intended to be used on a single computer system. RadiSys Corporation expressly prohibits any reproduction of the software on tape, disk, or any other medium except for backup purposes. Distribution of this software, in part or whole, to any other party or on any other system may constitute copyright infringements and misappropriation of trade secrets and confidential processes which are the property of RadiSys Corporation and/or other parties. Unauthorized distribution of software may cause damages far in excess of the value of the copies involved.

April 2000

Table of Contents

Chapter 1: Connecting and Configuring

Setting up the DAVID Install Pak and DAVID Application Pak includes seven basic steps. The following instructions and examples explain these steps.

Step 1. Connecting the hardware

- Connect a serial cable from a COM port on your PC (or a dumb terminal) to the port labeled "SERIAL 1" on the rear of your Hellcat. This is the connection for your system console. The baud rate for "SERIAL 1" is 9600.
- Audio connection: Line-level output is available from the two RCA jacks on the rear of the Hellcat marked "LEFT" and "RIGHT".
- Video connection: There are two options:

  For S-Video, there is a connector on the IGS video card inside the Hellcat.

  For composite video, there is an RCA connector on the rear of the Hellcat above the label "SURROUND".
- Do not connect the ethernet at this time.

Step 2. Testing the HOST/Target Connection

- Open your system console window.
- Turn on your Hellcat.
- You should hear three beeps, and then you should see the following (boot times are longer at first since the network information is not configured):

```
OS-9000 Bootstrap for the PowerPC™
1 devices online
```

After seeing this, you should be able to log in by pressing the return key and using the user name "super" and the password "user".

Step 3. Configuring the Hellcat

- Before your Hellcat can be put on the network, your IP and MAC addresses (among others) need to be configured. The MAC address is on the rear of your Hellcat.
- Turn on your Hellcat.
- When you hear a beep, press the `Escape` key.
- You will be presented with a menu of booting options:

```
Serial console selected by operator.
BOOTING PROCEDURES AVAILABLE - <INPUT>
Boot from ROM ---------------- <ro>
Load from ROM ---------------- <lr>
Boot over Ethernet ----------- <eb>
Restart the System ----------- <q>
Select a boot method from the above menu:
```

- Select `lr`.
- After booting, use the utility `syscfg` to set your networking parameters.
- Reboot your Hellcat. You do not need to press the Escape key for a normal boot.
- You can now connect the Hellcat to your ethernet network.

Step 4. Testing the network connection

To determine that your Hellcat is configured correctly, try to connect to it from your host machine using either `telnet` or `ftp`.

Step 5. Adding DAVID utilities to your Hellcat

- To use a utility on the Hellcat, it must first be brought to the Hellcat if it is not already in the boot image.

The simplest method is to use the `ftp` utility to transfer the utility to the Hellcat, then load it into memory using the `load` command. This requires that the Hellcat has enough space available on its RAM disk.

- If the utility is one you use frequently, you can burn it into your FLASH stick using the `pmod` utility.

Step 6. Replacing the boot image

A copy of the boot image is included with your Application Developers Pak in the event that you need to replace yours for some reason.

Note

The following step is intended for DAVID Installation Pak users only.

Step 7. Building a boot image

- To bring all of your binaries in the port directory up-to-date, go to `$MWOS\OS9000\821\PORTS\HELLCAT` and type `os9make`.
- The files specifying which modules are to be placed in the boot image are in `$MWOS\OS9000\821\PORTS\HELLCAT\BOOTLIST.` These files may be modified to add additional or modified modules to your boot image.
- To build a new boot image, go to `$MWOS\OS9000\821\PORTS\HELLCAT\ROM\BOOTROM` and type `os9make`.
- Once the make finishes, your boot files are located in `$MWOS\OS9000\821\PORTS\HELLCAT\CMDS\BOOTOBJS\ROM`.

Chapter 2: Using the fptest Utility

This chapter describes how to use the `fptest` utility to test the front panel functionality on the Hellcat set-top box.

Introduction

The `fptest` utility tests the front panel functionality of the Hellcat set-top box.

Following is an example.....

```
fptest <options>
```

options:

```
    led <1-3> red/green/yellow/<rgb color>
    reset
    power on/off
    selftest
    sendir
    read
```

Testing Front Panel Functionality

To test the front panel functionality of the Hellcat set-top box, perform the following steps:

Step 1. Make sure `fptest` is in the module directory.

Step 2. Run `fptest` with one of the above options.

Functionality and Interaction of Options

led <ledid 1-3> color

This sets the color of the corresponding led to the color specified.

reset

Resets the Hellcat set-top box.

power <on/off>

Sets the power led to green for on and yellow for off.

selftest

Continuously tests all of the possible led color combinations.

sendir

Sends the specified string(s) to the IR blaster.

read

Displays the ascii value of the front panels keys when pressed.

Chapter 3: Using the fwrite Utility

This chapter describes how to use the `fwrite` utility to write images into FLASH.

Introduction

The `fwrite` utility is similar to the `pmod` utility except that it is usually used to write images into FLASH rather than single modules.

The `fwrite` utility must be loaded into RAM before it can be used. The Hellcat development system must be booted using the `<lr>` option.

Note

`fwrite` is functional after booting using the `eb` boot selection, but this procedure is not recommended.

Following is an example.....

```
fwrite -a <address> -m <data module> file
```

`-a0x20120000`	Changes the start location of where `fwrite` looks at FLASH
`-e`	Erases the selected FLASH area
`-f`	Descriptor for the FLASH area to be modified
`-i`	Turns off the automatic reboot after `fwrite` has completed FLASH modifications
`-m`	Module name of data modules to be added to FLASH

For the Hellcat set-top box, `-f /sysflash` is the descriptor for the FLASH ROMs and `/flash` is the descriptor for an 8-MB FLASH simm.

Writing Images into FLASH

To burn a new rom image into FLASH, perform the following steps:

Step 1. Reboot the system. Press `esc` while the system is booting/beeping and a boot menu appears.

Step 2. Boot the Hellcat box in `<lr>` mode.

Step 3. `chd` to `Usr/sysadmin` on the Hellcat development system.

Step 4. From the host, FTP to the Hellcat development system and transfer the file bootrom from `$MWOS/OS9000/821/PORTS/HELLCAT/CMDS/BOOTOBJS/ROM`.

Step 5. Type `fwrite -f /sysflsh bootrom` on the Hellcat development system.

This process starts burning the image into FLASH. If system state debugging is off, the Hellcat box reboots itself after the image is burned into FLASH.

If system state debugging is on, the process appears to fail. This is not the case. The system is actually in the debugger. Wait three minutes and reset the Hellcat box.

Writing the First Image into FLASH

If this is the first time an image has been burned into FLASH on your system, you must perform the following steps:

Step 1. Press `esc` while the system is booting/beeping and a boot menu appears.

Note

Step 1 occurs after the procedure described above.

Step 2. Boot the Hellcat box in `<lr>` mode.

Step 3. Type `syscfg -i` to create a parameter file on NVRAM.

Step 4. Type `syscfg -rn=1` to make NVRAM the primary configuration device.

Step 5. Type `syscfg -d`. The current settings display.

Change the settings as appropriate to your environment.

The next time you reburn a bootimage into FLASH, all you need to do is type `syscfg -rn=1`. The old parameters that you set remain in FLASH.

Step 6. Reset the Hellcat box.

Note

The MAC address must be set each time a new boot ROM image is put into FLASH. This must be done while the set top box is in `lr` mode.

Chapter 4: Using the pmod Utility

This chapter describes how to use the pmod utility to make individual module or multiple module changes to FLASH memory.

Introduction

pmod makes individual module or multiple module changes to flash on a Hellcat development system. To use `pmod,` the Hellcat must be booted using the `<lr>` option.

Note

`pmod` is functional after booting using the `eb` boot selection, but this procedure is not recommended.

Following is an example:

```
pmod -bcelrstz=<filename> module module
```

```
pmod
```

`-l`	Listed modules resident in flash
`-c`	Show a `*` before name if bad `crc in list`
`-b`	Burn listed modules into flash
`-e`	Erase listed modules
`-f`	Flash descriptor
`-r`	Replace listed modules
`-s`	Define a new flash start address
`-t`	Define a new flash end address
`-z=<file>`	Use list of modules for specified operations

Making Module Changes to FLASH

To make individual or multiple module changes to FLASH on your Hellcat development system, perform the following steps:

Step 1. Boot the system.

Step 2. During a series of beeps, while the system is booting, press the `esc` key.

Step 3. When the boot menu appears, type `lr` to load from ROM.

Booting in lr Mode

If the `sys_parm` module is located in non-volatile RAM, it is only necessary to boot into `lr` mode to change the MAC address and to enable/disable ssm and cache.

To place a `sys_parm` module in `/nvr`, perform a `syscfg -i`.

Note

If you replace the boot image on the set top box, a `syscfg -rn=1` points `sysgo` to the parameter file already in non-volatile RAM.

Functionality and Interaction of Parameters

Changing the Flash Address (-s and -t)

The pmod default operating range is from 0x20000000 to 0x200fffff. It is restricted to this address space without additional options.

To change the area in which pmod operates, the following two options are available:

- `-s` `0x20120000` changes the start location of where pmod looks at FLASH.
- `-t` `0x201fffff` changes the stop location of where pmod looks at FLASH.

If you specify a new start location that is not a modules sync bytes, pmod searches flash for a module header, and when found, designates the remainder of the first module it just passed through as available flash. This works because pmod does not write to a non-module area that is not erased (all `0xff`).

pmod searches FLASH and locates all resident modules, their position, and size. Non-module areas are named `nonmodulex`, with `x` an incrementing number reflecting the number of non-module areas found in the specified search area.

List Modules (-l and -c)

pmod finds all modules with a valid header `crc` even if the module `crc` is corrupt. This is evident when the Hellcat systems boots up. A specific module is not listed during an `mdir` command, but is listed during a `pmod -l`. Also, executing a `pmod -l -c` checks the module `crc` and places a `*` in front of corrupt module names.

Listing and Erasing Modules (-z and -e)

Specific modules can be erased by executing pmod -e <module name> <module name>. Module names to be erased can be placed in a file, one name per line, and the following command executed:

```
pmod -e -z=filename
```

The -z option can have individual module names appended, such as:

```
pmod -e -z=filename module1 module2
```

Burning Modules to FLASH (-b)

The -b option specifies burning a module into FLASH that resides on the Hellcat file system. All capabilities of the -z option are functional with the -b option.

pmod searches FLASH in the default area or the user-specified area for nonmodule areas. It makes a list of modules to be burned and serially looks through the available non-module areas for a space large enough to burn the current module. It repeats this process until all modules are burned in FLASH or it fails to find space.

When pmod attempts to write a module to FLASH, it first checks to ensure that the area located is erased, all FF's. It does not burn to an area of FLASH that is nonmodule but not erased. A list of targets must all be file system resident.

Replace Listed Modules (-r)

The -r option replaces a module in FLASH if the replacement module is the same size or smaller. The module is also replaced if the module in FLASH immediately precedes a non-module area where the replacement is smaller or equal to the combined area of the target module plus the area of the non-module.

Chapter 5: Using the syscfg Utility

This chapter describes how to use the `syscfg` utility to change system parameters.

Introduction

The `syscfg` utility is used to change system parameters associated with the the `sysgo` module as it is used with the Hellcat set-top box (STB).

The `sysgo` module for this version of DAVID is derived from the `sysgo` example provided in the ***OS-9 Technical Manual***. It is enhanced to handle most of the parameters specified in the `syscfg` example listed below.

For More Information

See the ***OS-9 Technical Manual*** for an example of the `sysgo` module.

syscfg Example

```
Nvr params:
Host Parameters
   name (hn)hc6
   domain name (he)microware.com
Ethernet Parameters
   enable (ee)enabled
   MAC address (em)41:00:00:00:00:00
   IP address (ei)172.16.2.148
   gateway address (eg)172.16.1.254
   broadcast address (eb)172.16.255.255
   subnet mask (es)255.255.0.0
   dns server (en)172.16.2.128
   dns server (eo)172.16.1.32
   dns server (eq)0.0.0.0
   uplink server (eu)172.16.2.11
Misc Parameters
   RAM disk size (mk)6144K
   rombug (mr)disabled
   sys state debug (ms)disabled
      hlproto (mh)disabled
      sndp (mj)disabled
   mbuff size (mb)256K
   init ma (ma)enabled
   init mv (mv)enabled
   start inetd (mi)enabled
   start ndpd (mu)enabled
   start nppd (mc)disabled
   start maui_inp (mm)enabled
   nvr startup (mn)disabled
   dd startup (me)disabled
   t1 tsmon (mt)disabled
   pal decoding (mp)disabled
   ntsc decoding (mx)enabled
Flash only Parameters
   cache on|off (rc)disabled
   ssm on|off (rs)disabled
```

Changing System Parameters

To change system parameters in the `sysgo` module, perform the following steps:

Step 1. Boot the system.

Step 2. During a series of beeps, while the system is booting, press the `esc` key.

Step 3. When the boot menu appears, type `lr` to load from ROM.

Booting in lr Mode

If the `sys_parm` module is located in non-volatile RAM, it is only necessary to boot into `lr` mode to change the MAC address and to enable/disable ssm and cache.

To place a `sys_parm` module in `/nvr`, perform a `syscfg -i`.

Note

If you replace the boot image on the set top box, a `syscfg -rn=1` points `sysgo` to the parameter file already in non-volatile RAM.

Functionality and Interaction of Parameters

The functionality and interaction of each parameter listed in the `syscfg` example is described below.

Host Parameters

Name (hn)

Sets the host name of the set top box.

Domain Name (he)

Sets the domain name for the network.

Ethernet Parameters

Enable (ee)

If enabled (ee) starts the isp system. This flag loads the `inetdb` module required by the isp system.

MAC Address (em)

The MAC address (hardware ethernet address) can only be changed when the system is booted in `<lr>` mode.

Standard Ethernet Parameters

The standard ethernet parameters must be set according to your particular network. They include the following:

- IP address (ei)
- Gateway address (eg)
- Broadcast address (eb)
- Subnet mask (es)
- dns server (en)
- dns server (eo)
- dns server (eu)

For More Information

See ***Using LAN Communications Pak*** for additional information about setting the standard ethernet parameters.

Miscellaneous Parameters

RAM disk size (mk)

This parameter controls the size of the ram disk. The ram disk is the default file system for the set-top box. The value must be given in kbs and be a multiple of 256.

rombug(mr)

This parameter turns on the rombug debugger.

sys state debug (ms)

Setting this constant to enabled initializes the modules required for system state debugging. None of the ISP ethernet modules is started.

Two values used for ISP operation, `ndpd` and `nppd`, are examined. If the values are set, the system state debugging counterparts are started. For example, if `ndpd` is enabled during system state debugging, operation `undpd` is started. To resume user state debugging, issue the command `syscfg -ms=disabled` and reboot the set-top box.

mbuff size (mb)

This parameter controls the amount of memory allocated for mbuffers. The system default is 128k. Values smaller than 128k can cause system instability.

init ma (ma)

This parameter must be initialized by `sysgo` in order to gain access to non-volatile storage.

init mv (mv)

If enabled, this constant causes `sysgo` to iniz `/mv`.

start inetd (mi)

If enabled, `sysgo` runs `inetd` in the background. The demon is not run if Ethernet is disabled or system state debugging is enabled. `inetd` handles incoming FTP and telnet connections.

start ndpd (mu)

If enabled this constant causes `sysgo` to start `ndpd` while in user state debugging and `undpd` while in system state. If in user state mode and Ethernet is disabled, `ndpd` is not started.

start nppd (mc)

Not implemented at this time.

start maui_inp (mm)

If enabled, `maui_inp` is started.

nvr startup (mn)

If enabled, a startup file present on `/nvr` is executed. Standard error during this time is redirected to a status file, called `stat`, on `/dd`.

dd startup (me)

If enabled, a startup file present on `/dd` is executed. Standard error is redirected to a status file, called `stat`, on `/dd`. Only one startup file can be enabled at a time. If both are enabled, only `nvr startup` is run. Both may be disabled.

t1 tsmon (mt)

If enabled, the time-sharing monitor (`tsmon`) is started for the serial port `/t1,` allowing logins on this port.

pal decoding (mp)

Not implemented at this time.

ntsc decoding (mx)

Enables NTSC display.

FLASH Parameters

cache enable (rc)

To enable or disable cache, the system must be booted in `<lr>` mode.

ssm enable (rs)

To enable or disable ssm, the system must be booted in `<lr>` mode.

Note

To change these constants, use `syscfg` to set them to an appropriate value and reboot. A number of automatic defaults occur if improper or mismatched conditions exist.

Default Configuration

`sysgo` uses a default configuration and forks a shell—instead of `tsmon`—if the parameter file is missing from `/nvr` or the file system is unavailable. `tsmon` requires that a login be completed, which is not possible under these conditions.

If this default condition occurs, use the supplied shell to resolve any problems. The `sysgo` startup log in `/dd/stat` may indicate the problem(s) that stopped the normal boot process. Any communication capability required to correct the problem must be started by you in this situation.

System State Debugging

You can start system state debugging by using the `syscfg -ms=enabled` command. After rebooting, the set-top box starts with system state debugging functional.

There is no need to disable the ISP settings. `sysgo` recognizes the operation mode and does not start any ISP functionality. The two flags examined are `ndpd` and `nppd`. If `ndpd` is enabled, `sysgo` starts `undpd`. If `nppd` is enabled, `sysgo` starts `unppd`. To resume user state debugging, set the `-ms=disabled` flag and the pre-existing setup is used.

SPF System

During normal operation, one of the following options must be enabled to keep the stacked protocol file (SPF) system active:

- `inetd`
- `ndpd`

If none of these functions is selected, SPF operation will be indeterminate.

System Startup File

Startup file execution has three options:

- Execute the startup file located on `/nvr`
- Execute the startup file located on `/dd`
- Do not execute a startup file

Enabling both startup files is an invalid condition. If you attempt to enable both startup files, the one in `/nvr` is executed.

Configuring your Hellcat with syscfg

To change these constants use syscfg to set them to an appropriate value and reboot. There are a number of automatic defaults that occur if improper or mismatched conditions exist.

An important consideration is what action sysgo takes if the parameter file is missing from /nvr or the file system is unavailable. When this occurs, sysgo uses a default configuration and a shell is forked instead of tsmon. (tsmon requires that a login be completed and is that is not possible under these conditions.) If this default condition occurs, use the supplied shell to resolve any problems. The sysgo startup log in /dd/stat may offer some indication to the problem which stopped the normal boot process. Any communication capability required to correct the problem must be started by the user in this situation.

You may elect to start system state debugging by using the syscfg -ms=enabled command. After rebooting, the box starts with system state debugging functional. There is no need to disable the isp settings, sysgo recognizes the operation mode and does not start any isp functionality. The two flags examined are ndpd and nppd. If ndpd is enabled, sysgo starts undpd. If nppd is enabled, unppd is started. To resume user state debugging, set the -ms=disabled flag and the pre-existing setup is used.

The Ethernet parameters for system state debugging are stored in the system module cnfgdata. This module does not show up in an mdir, and resides in FLASH. To set the system state debugging parameters, set the Ethernet parameters using syscfg. They may be set to different values than the box used during normal isp operation. Reboot the box using the lr boot selection, which is accessible by entering an escape character during the three beeps that occur during the boot. Once the box has completed startup, enter the command pmod -u. This transfers the Ethernet parameters stored in non-volatile to the cnfgdata data module in FLASH. You may reboot and restore the Ethernet parameters required for normal isp operation.

During normal operation, one of the following options must be enabled to keep the isp system active: inetd, routed, ndpd, or nppd. If none of these functions are selected, isp operation is indeterminate.

Startup file execution has three options:

- Execute the startup file located on `/nvr`
- Execute the startup file located on `/dd`
- Do no execute a startup file

Enabling both startup files in an invalid condition, and the one in `/nvr` is executed.

Chapter 6: Using the Xdmod Utility

This chapter describes how to use the Xdmod utility to pack a file in a data module or remove the contents of a data module to a file.

Introduction

Xdmod is a bi-directional command used to pack a file in a data module or remove the contents of a data module to a file. You must specify a file name and a module name. The option determines if a file is created from the contents of the module or a data module is created containing the specified file. Only data modules made by xdmod can be unpacked. If a data module created using mkdatmod is unpacked, the file will be corrupt.

An example is:

```
Xdmod -f|-m <file name> <module>
```

Appendix A: Hellcat Specific Programming Reference

This section provides programming reference information about the Hellcat Rear Panel and Front Panel.

Hellcat Rear Panel

The Hellcat rear panel device `/rp` controls several distinctly different pieces of hardware. These are all controlled via `SetStat`/`GetStat` calls to the driver. An API library called `rearpanl.l` is provided for that purpose. The following rear panel functional blocks are controlled by the rear panel driver.

- Audio input volume control
- Audio input digital sample rate control
- Audio output mute control
- RF output switching
- RF output TV channel control

The following library functions are provided in `rearpanl.l`. These are declared in `rearpanl.h`, which should be included by all applications that use the rear panel device.

In addition to calling these library functions, it is possible for an application to make `GetStat`/`SetStat` calls directly to the driver. This is not recommended, since the library functions provide a much simpler interface and calling the driver directly gives you no added capabilities over those provided by the library functions. In addition, the library functions provide range checking on passed arguments. If you need to call the driver directly, however, there are definitions in `rearpanl.h` for the parameter blocks and function codes that are passed to the driver. Also, see the source to `rearpanl.l` to see how it operates.

_gs_rp_set_audclk()

Set Input Audio Sample Rate

Syntax

```
#include <rearpanl.h>
error_code _gs_rp_set_audclk(
     path_id      path,
     u_int32      sr);
```

Description

`_gs_rp_set_audclk()`sets the input audio sample rate to 44.1 Kbits/sec or 48.0 Kbits/sec.

Parameters

`path`	An open path to the `/rp` device
`sr`	The desired audio sample rate, `RP_AUDCLK_44` for 44.1 Kbits/sec or `RP_AUDCLK_48` for 48.0 Kbits/sec.

Errors

`EOS_PARAM`	Returned if you pass a value for `sr` other than `RP_AUDCLK_44` or `RP_AUDCLK_48`

_gs_rp_set_tvchan()

Select RF Output TV Channel

Syntax

```
#include <rearpanl.h>
error_code _gs_rp_set_tvchan(
     path_id      path,
     u_int32      chan);
```

Description

`_gs_rp_set_tvchan()` controls the channel setting. The RF modulator can be set to TV channel 3 or channel 4.

Parameters

`path`	An open path to the `/rp` device
`chan`	The TV channel selection, `RP_CHNL_3` for channel 3 or `RP_CHNL_4` for channel 4

Errors

`EOS_PARAM`	Returned if you pass a value for `chan` other than `RP_CHNL_3` or `RP_CHNL_4`

_gs_rp_set_rfoutput()

Select RF Output

Syntax

```
#include <rearpanl.h>
error_code _gs_rp_set_rfoutput(
     path_id      path,
     u_int32      output);
```

Description

`_gs_rp_set_rfoutput()` controls RF output routing. The RF output on the back of the box can be set either to loop through the RF input or to the output of the RF modulator.

Parameters

`path`	An open path to the `/rp` device
`output`	The routing selection, `RP_RF_PASS` to pass the input through to the output; or `RP_RF_MOD` to route the modulator output to the RF output

Errors

`EOS_PARAM`	Returned if you pass a value for `output` other than `RP_RF_PASS` or `RP_RF_MOD`

_gs_rp_set_atten()

Set Audio Input Volume

Syntax

```
#include <rearpanl.h>
error_code _gs_rp_set_atten(
     path_id     path,
     u_int32     left,
     u_int32     right)
```

Description

`_gs_rp_set_atten()` sets the audio input levels (separate settings for left and right channel).

The levels may be set to any value between 0 and `RP_VOL_MAX` (192). Each increment in the level corresponds to a 0.5 dB change in volume, giving a total adjustment range of 96 dB. A constant called `RP_VOL_MAX`, which is the maximum legal level, is defined in `rearpanl.h`.

Parameters

`path`	An open path to the `/rp` device
`left`	The left channel input level, 0 to `RP_VOL_MAX`
`right`	The right channel input level, 0 to `RP_VOL_MAX`

Errors

`EOS_PARAM`	Returned if you pass a level greater than `RP_VOL_MAX` to this function

_gs_rp_set_attenlr()

Set Audio Input Volume

Syntax

```
#include <rearpanl.h>
error_code _gs_rp_set_attenlr(
     path_id     path,
     u_int32     vol)
```

Description

`_gs_rp_set_attenlr()` sets both left and right audio inputs to the same level.

The levels may be set to any value between 0 and `RP_VOL_MAX` (192). Each increment in the level corresponds to a 0.5 dB change in volume, giving a total adjustment range of 96 dB. A constant called `RP_VOL_MAX`, which is the maximum legal level, is defined in `rearpanl.h`.

Parameters

`path`	An open path to the `/rp` device
`vol`	The input level setting for both the left and the right channel, 0 to `RP_VOL_MAX`

Errors

`EOS_PARAM`	Returned if you pass a level greater than `RP_VOL_MAX` to this function

_gs_rp_set_mute()

Mute or Unmute Audio Output

Syntax

```
#include <rearpanl.h>
error_code _gs_rp_set_mute(
     path_id      path,
     u_int32      mute)
```

Description

`_gs_rp_set_mute()` turns the audio output on or off.

Parameters

`path`	An open path to the `/rp` device
`mute`	The mute setting, `RP_MUTE` to turn off the output or `RP_UNMUTE` to turn on the output

Errors

`EOS_PARAM`	Returned if you pass a value for `mute` other than `RP_MUTE` or `RP_UNMUTE`

_gs_rp_get_audclk()

Get the Current Input Audio Sample Rate

Syntax

```
#include <rearpanl.h>
error_code _gs_rp_get_audclk(
     path_id      path,
     u_int32      *sr);
```

Description

`_gs_rp_get_audclk()` returns the current input audio sample rate.

Parameters

`path`	An open path to the `/rp` device.
`sr`	A pointer to the location where the current audio sample rate will be stored by this function; after this function returns `SUCCESS`, that location will contain `RP_AUDCLK_44` or `RP_AUDCLK_48`

_gs_rp_get_tvchan()

Get the Current TV Channel Selection

Syntax

```
#include <rearpanl.h>
error_code _gs_rp_get_tvchan(
     path_id      path,
     u_int32      *chan);
```

Description

`_gs_rp_get_tvchan()` returns the current TV channel setting.

Parameters

`path`	An open path to the `/rp` device
`chan`	A pointer to the location where the current TV channel selection will be stored by this function; after this function returns `SUCCESS`, that location will contain `RP_CHNL_3` or `RP_CHNL_4`

_gs_rp_get_rfoutput()

Get the Current RF Output Setting

Syntax

```
#include <rearpanl.h>
error_code _gs_rp_get_rfoutput(
     path_id      path,
     u_int32      *output);
```

Description

`_gs_rp_get_rfoutput()` returns the current RF output setting.

Parameters

`path`	An open path to the `/rp` device
`output`	A pointer to the location where the current RF output setting will be stored by this function; after this function returns `SUCCESS`, that location will contain `RP_RF_PASS` or `RP_RF_MOD`

_gs_rp_get_atten()

Get the Current Audio Input Level Settings

Syntax

```
#include <rearpanl.h>
error_code _gs_rp_get_atten(
     path_id      path,
     u_int32      *left,
     u_int32      *right);
```

Description

`_gs_rp_get_atten()` gets the current left and right audio input level settings.

Parameters

`path`	An open path to the `/rp` device
`left`	A pointer to the location where the current left channel level setting will be stored by this function
`right`	A pointer to the location where the current right channel level setting will be stored by this function

After this function returns `SUCCESS`, `*left` and `*right` contains the current left and right audio input level settings, in the range of 0 to `RP_VOL_MAX` (192).

_gs_rp_get_mute()

Get the Current Output Mute Setting

Syntax

```
#include <rearpanl.h>
error_code _gs_rp_get_mute(
     path_id     path,
     u_int32     *mute);
```

Description

`_gs_rp_get_mute()` returns the current input audio sample rate.

Parameters

`path`	An open path to the `/rp` device
`mute`	A pointer to the location where the current audio output mute setting will be stored by this function. After this function returns `SUCCESS`, that location will contain `RP_MUTE` or `RP_UNMUTE`

The Hellcat Front Panel

The Hellcat front panel device (`/ir`) provides IR input and output, and control of other front panel functionality. The IR input is normally done using a MAUI input protocol module, which is included in the system. IR output is done by opening a path to the `/ir` device and doing `write()` operations on that path. All other front panel operations are done via `setstat` calls to the driver. An API library called `fpir_lib.l` is provided for that purpose.

The following library functions are provided in `fpir_lib.l`. These are declared in `fpir.h`, which should be included by all applications that use the front panel device.

fp_reset()

Reset the Hellcat System

Syntax

```
#include <fpir.h>
error_code fp_reset(path_id path);
```

Description

`fp_reset()` sends a `RESET` command to the front panel device. This causes a hardware reset of the Hellcat system.

Parameters

`path`	An open path to the `/ir` device

fp_self_test()

Puts the Front Panel Into Self Test Mode

Syntax

```
#include <fpir.h>
error_code fp_self_test(path_id path);
```

Description

`fp_self_test()` puts the front panel device into self-test mode. After running the front panel self-test, the Hellcat system must be reset to restore normal operation.

Parameters

`path`	An open path to the `/ir` device

fp_power_ctrl()

Switch Hellcat On or Off

Syntax

```
#include <fpir.h>
error_code fp_power_ctrl(
     path_id      path,
     u_char       power);
```

Description

`fp_power_ctrl()` sends power on/power off commands to front panel.

Parameters

`path`	An open path to the `/ir` device
`power`	The selected power state, 0 = power off, all other values = power on

fp_led_ctrl()

Control Front Panel LEDs

Syntax

```
#include <fpir.h>
error_code fp_led_ctrl(
     path_id      path,
     u_char       ledid,
     u_char       rgycode)
```

Description

fp_led_ctrl() turns front panel LEDs on or off and sets their color.

Parameters

path
An open path to the /ir device

ledid
Selects the LED to set. Legal values are FPIR_LED1, FPIR_LED2 or FPIR_LED3

rgycode
The desired color; legal values are FPIR_LED_OFF, FPIR_LED_RED, FPIR_LED_GREEN, or FPIR_LED_GREEN

Index

Product Discrepancy Report

To: Microware Customer Support

FAX: 515-224-1352

From:____

Company:____

Phone:____

Fax:______________________Email:___________________

Product Name:

Description of Problem:

____

____

____

____

____

____

____

____

____

Host Platform_______________________________________

Target Platform_____________________________________