zero G tm

Connecting the Internet of Things

The 4th Age of Wireless™

Console CLI commands

Summary - This document illustrates the ZeroG command line interface (CLI). It starts with the information needed load CLI commands in an application. It then describes the commands with some simple illustrations. Finally, it takes the user through an example of an iperf application that has the command lines implemented. In this environment, more elaborate network examples are described starting with a simple 'ping' verification of the ZG2100 in managed mode to the establishment of an adhoc connection.

CLI-101.01

Revision 1.0

MEMBER Wi Fi ™ ALLIANCE

Zero G Wireless · 255 San Geronimo Way · Sunnyvale · CA · 94085 · 408 738 7600 · 408 738 7601 fax
www.zerogwireless.com

Contents

Table of Contents

Using CLI in applications

The following are the steps necessary to compile, link and use the ZeroG CLI code in a customer application. AppX.c is a hypothetical file.

1) The user should #include "TCPIP Stack/ZGConsole.h" at the top of their application's appX.c file

```
#include "TCPIP Stack/ZGConsole.h
```

2) The user should place the following data structures at the top of appX.c file. Here is an example:

```
#if defined( ZG_CONFIG_CONSOLE )
        static ROM tZGU8 cmdStrings[2][12] = { {"iperf"}, {"kill"} };
        ROM tZGU8 *p_cmdStringPtrList[3] = { cmdStrings[0], cmdStrings[1]};
        #define kZGNumCmdsInList (sizeof(p_cmdStringPtrList) / sizeof(tZGU8 *))
#endif
```

3) The user should add this code to the call appX.c file, preferably in their init code or routine:

```
#if defined( ZG_CONFIG_CONSOLE )
  ZGConsoleInit( p_cmdStringPtrList, kZGNumCmdsInList);
#endif
```

4) The user should put the following code in the main thread of execution in appX.c , for example:

```
while (1)
{
/* system tasks */

  #if defined( ZG_CONFIG_CONSOLE )
   ZGConsoleProcess();
#endif

}
```

5) The user should include the following files in their project workspace:

Mandatory Files

```
ZGConsoleMsgHandler.c
ZGConsoleMsgHandler.h
```

ZGConsoleMsgs.c
ZGConsoleMsgs.h
ZGConsole.c
ZGConsole.h
ZGConsoleShared.h

Optional Files (extra CLI applications)

ZGConsoleIfconfig.h
ZGConsoleIfconfig.c
ZGConsoleIwconfig.h
ZGConsoleIwconfig.c
ZGConsoleIwpriv.h
ZGConsoleIwpriv.c

6) In TCPIPConfig.h, the user should uncomment this line:

```
/* "DOS like" command line interface and iperf - network tools */
  #define ZG_CONFIG_CONSOLE                 y
```

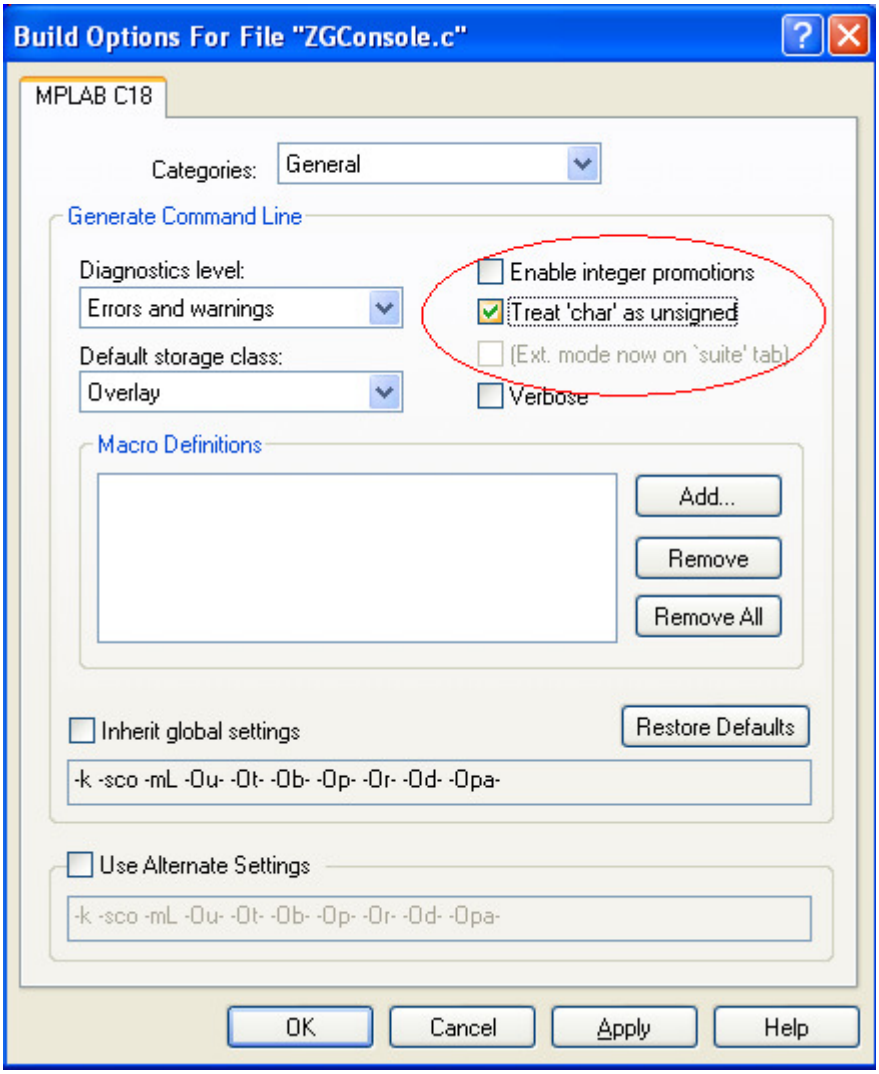
7) For the PIC18 & C18 tool chain, the link script may require modification. The following line should be added to the link script to provide the necessary space for the CLI data structures:

```
DATABANK   NAME=gpr2a      START=0x200          END=0x2CF    PROTECTED
DATABANK   NAME=gpr2       START=0x2D0          END=0x2FF

SECTION    NAME=zgcli_section   RAM=gpr2a
```

8) The Microchip TCPIP stack, on the PIC18 & C18 tool chain, assumes a project wide setting where signed integers are forced to unsigned integers, and this may cause a functional problem with the console. The workaround is to click on the ZGConsole.c and ZGConsoleMsg.c files in the workspace editor, then deselect the "Treat 'char' as unsigned" checkbox. The console is sensitive to signed chars.

Zero Wireless · 255 San Geronimo Way · Sunnyvale · CA · 94085 · 408 738 7600 · 408 738 7601 fax
www.zerogwireless.com

ZeroG Console CLI commands



Console commands

The following console commands can be used to readily configure the wireless network, interface, and encryption settings of the ZG2100. Below the standalone commands are shown followed by the configuration commands that take arguments.

Standalone commands

The following standalone commands do not take any arguments. They are described followed by a typical example in terminal with the command and a response shown in the box.

help

This command lists all the available CLI commands.

```
> help
help         Lists all commands
getzgver     Gets ZG2100 version
reset        Reset host MCU
cls          Clears screen
ifconfig     see documentation
iwconfig     see documentation
iwpriv       see documentation
```

getzgver

This command shows the firmware version and SDK version.

```
> getzgver
Firmware version    0x1103
SDK version         1.3.4.10
```

reset

This command resets the host MCU. If the AP and ZG2100 have the same SSID and settings by default, the reset command will establish a connection. In the example below, they were not the same, and the connection attempt failed.

```
> reset

ZeroG v1.3.4.10
```

```
ZeroG Firmware Version 0x1100
New IP Address: 192.168.1.65
Set regional domain ...
   Region = fcc
Scan ... for ssid “0GAP”
   AP is configured on different channel
```

cls

This command will clear the screen. In the current version, the whole screen is not cleared. This and other errata are listed in section 5 as errata.

```
 >  cls

================================================================================
* ZeroG Host Interface Monitor
* (c) 2008 -- ZeroG, Inc.
*
* Type 'help' to get a list of commands.
================================================================================
getzgver
Firmware version   0x1103
SDK version        1.3.4.10
>
>
```

Configuration commands

The iwconfig, ifconfig, and iwpriv commands control the wireless (w), network interface (if), and the privacy (priv) settings of the wireless network. They are described below followed by typical examples.

iwconfig commands

Use this command to configure the ZG2100 wireless interface and operation. Iwconfig without any parameters will display the current settings.

SYNOPSIS

iwconfig [ssid <name>]
[mode ('idle' | 'managed' | 'adhoc')]
[channel (<channel list> | 'all')]
[power ('reenable' | 'disable' | 'unicast' | 'all')]
[domain <name>]
[rts <length>]
[txrate (0 | 1 | 2)]

IWCONFIG EXAMPLES

The examples below illustrate the various options of iwconfig.

DISPLAY CURRENT STATUS.

Iwconfig without any parameters will display the current settings.

```
> iwconfig
        channel:  1,6,11,2,7,3,8,4,9,5,10
        domain:   fcc
        rts:      2347
        mode:     managed
        ssid:     0GAP-KS
        pwrsave:  disabled
```

iwconfig domain (Configure regulatory domain)

Set the domain associated to a geographical region & channelization scheme. The domain establishes the characteristics for the 802.11 radio and it's channel masks. Valid domain names and corresponding channel range are shown below.

a. fcc (North American domain) - 1 to 11
b. ic ( ) - 1 to 11
c. etsi (Europe) - 1 to 13
d. spain - 10 to 11
e. france - 10 to 13
f. japana - 14
g. japanb - 1 to 13

For example, set to domain 'japana'

```
> iwconfig domain japana
> iwconfig
        channel:  14
        domain:   japana
        rts:      2347
        mode:     idle
        ssid:     0GAP-KS
        pwrsave:  disabled
```

iwconfig channel (Set channels)

Set the active channels. The G2100 in managed mode will reject any SSID's that fall outside the specified channel range. The channel range must be valid for the domain/channelization scheme shown in the above table. A user can specify "all" for all valid channels in a domain, and the user may specify a channel list with comma delimiters.

```
> iwconfig
        channel:  1,6,11,2,7,3,8,4,9,5,10
        domain:   fcc
        rts:      2347
        mode:     idle
        ssid:     0GAP-KS
        pwrsave:  disabled

> iwconfig channel 1,6,11
> iwconfig
        channel:  1,6,11
        domain:   fcc
        rts:      2347
        mode:     idle
        ssid:     0GAP-KS
        pwrsave:  disabled
```

```
> iwconfig channel 6
> iwconfig
        channel:  6
        domain:   fcc
        rts:      2347
        mode:     idle
        ssid:     0GAP-KS
        pwrsave:  disabled

> iwconfig channel all
> iwconfig
        channel:  1,2,3,4,5,6,7,8,9,10,11
        domain:   fcc
        rts:      2347
        mode:     idle
        ssid:     0GAP-KS
        pwrsave:  disabled
```

iwconfig SSID (Set SSID)

Set the ssid (Network Name). The SSID identifies a network name that G2100 will match during connection establishment. The ssid is used in both adhoc and managed modes. This command allows a user defined string up to 31 characters (ie. string < 32 characters).

```
> iwconfig ssid MyAccessPoint
> iwconfig
        channel:  6
        domain:   fcc
        rts:      2347
        mode:     idle
        ssid:     MyAccessPoint
        pwrsave:  disabled
```

iwconfig mode (Set mode)

Set the 802.11 network connection for BSS (managed), IBSS (adhoc), or idle mode of operation. For example, switch to managed mode:

```
> iwconfig mode managed
```

```
Set MAC addr ...
    mac = 00:BA:BE:11:00:00
Set regional domain ...
    region = fcc
Scan ... for ssid "0GAP-KS"
    [0] bssid = 00:1B:2F:5A:BC:A2
        strength = ###..
        indicator = 158
        ssid = 0GAP-KS
        channel = 6
    AP selected = [0]
Join ...
    succeeded
Authenticate ...
    succeeded
Associate ...
    echo AP basic rates
    succeeded
Connected!
DHCP discover 1 (method 0)
DHCP discover 2 (method 0)
DHCP Client Lease Acquired.
  ip addr = 192.168.1.5
  netmask = 255.255.255.0
  router = 192.168.1.1
```

Switch to idle mode:

```
> iwconfig mode idle
Disconnect...
    succeeded
```

Switch to adhoc mode:

```
> iwconfig mode adhoc
Set MAC addr ...
    mac = 00:BA:BE:11:00:00
Set regional domain ...
    region = fcc
Scan ...
    no network found
    starting network ...
Start ...
    succeeded
```

iwconfig power (Set power save mode)

Set the power profile for the G2100. In power save mode, the G2100 radio may be placed into a power conservation state; this state can prolong battery life significantly. The power savings mode is alternately referred to as PS-POLL mode. The G2100 active mode is a normal state, where the device is always ready to receive and transmit. The SDK defaults the G2100 into active mode, and the user must explicitly "reenable" power savings. The parameter "disable" will subsequently turn off power savings mode. The G2100 power saving mode has two flavors: "all" and "unicast." The "all" parameter configures the G2100 to wake up and poll for broadcast/multicast pkts buffered at an AP (ie. DTIM). Alternately, the "unicast" parameter will place the G2100 into it's deepest sleep state, and the device periodically wakes up to poll for unicast traffic (ie TIM) in a beacon pkt.

```
> iwconfig
        channel:  1,6,11
        domain:   fcc
        rts:      2347
        mode:     managed
        ssid:     0GAP-KS
        pwrsave:  disabled

> iwconfig power reenable
> iwconfig
        channel:  1,6,11
        domain:   fcc
        rts:      2347
        mode:     managed
        ssid:     0GAP-KS
        pwrsave:  enabled
        dtim rx:  enabled

> iwconfig power unicast
> iwconfig
        channel:  1,6,11
        domain:   fcc
        rts:      2347
        mode:     managed
        ssid:     0GAP-KS
        pwrsave:  enabled
        dtim rx:  disabled

> iwconfig power disable
> iwconfig
        channel:  1,6,11
        domain:   fcc
        rts:      2347
        mode:     managed
```

```
        ssid:     0GAP-KS
        pwrsave:  disabled
```

iwconfig rts (Configure RTS)

Set the requested #bytes to send. The default is max at 2347, yielding high throughput for single AP/client pair. For AP with many clients, the RTS may be set smaller.

```
> iwconfig
        channel:  1,6,11
        domain:   fcc
        rts:      2347
        mode:     managed
        ssid:     0GAP-KS
        pwrsave:  disabled

> iwconfig rts 500
> iwconfig
        channel:  1,6,11
        domain:   fcc
        rts:      500
        mode:     managed
        ssid:     0GAP-KS
        pwrsave:  disabled
```

iwconfig txrate (Configure TX rate)

The tx rate can be set to auto-rate adaptation by the ZG2100 (0), or to fixed datarates of 1 and 2 Mbps.

```
> iwconfig txrate 2
> iwconfig txrate 1
> iwconfig txrate 0
```

ifconfig commands

Use the ifconfig command to configure a network interface. Some ifconfig parameters are committed during the initial establishment of connection state, and these parameters are cached until a transition from idle->managed or idle->adhoc state. One such parameter is the hw_mac_address, which may only be modified in idle state. Other settings may be modified & committed in any state. For example, the internet ipv4 address & netmask may be modified in idle, adhoc, or managed states. The inet_address and netmask are immediately committed.
If no ifconfig parameters are specified, the tool will return the user's requested configuration. If ZG2100 is in a idle state, the hw_mac_address is not committed until the system transitions into a managed or adhoc mode.

SYNAPSIS

ifconfig [<'hw-mac-addr'>]
[<'static ip'>]
[netmask <'netmask'>]
[gateway <'gateway'>]
[auto-dhcp (' start' |'drop')]

Ifconfig (Display ifconfig status)

```
> ifconfig
        IP addr:  192.168.1.5
        MAC addr: 00:BA:BE:11:00:00
        Netmask:  255.255.255.0
        Gateway:  192.168.1.1
        DHCP:     Started
```

Ifconfig ipaddress (Configure static IP)

Set the node's layer 3 address. The inet_address is configured to the uIP stack, and is used to filter unicast packets. The node's inet_addr is necessary for TCP/IP and UDP/IP sockets, as well as, network application programs. The inet_address is formatted as 4 octets separated by periods: aa.bb.cc.dd. This param is for static ipv4 address only, and the default value of

192.168.1.11 may be updated manually. If the DHCP client is compile time enabled, the user should not manually assign a static ip address, unless the DHCP client is stopped (ie. ifconfig auto-dhcp drop). A DHCP client will negotiate with a DHCP server for the inet_addr lease.

```
> ifconfig 192.168.1.3
Static IP address should not be set with DHCP enabled.

> ifconfig auto-dhcp drop
> ifconfig 192.168.1.3
> ifconfig
        IP addr:  192.168.1.3
        MAC addr: 00:1E:C0:00:00:00
        Netmask:  255.255.0.0
        Gateway:  192.168.1.1
        DHCP:     Stopped
```

Ifconfig netmask (Configure IP netmask)

This parameter is used to define how much of the inet_address is used to divide the network into classA, classB, and classC subnetworks. The tool expects a decimal value (1-255) separated by periods: aa.bb.cc.dd. The netmask parameter currently may not be specified with inet_addr or hw_mac_addr, and must be specified by itself.

```
> ifconfig netmask 255.255.255.0
> ifconfig
        IP addr:  192.168.1.6
        MAC addr: 00:BA:BE:11:00:00
        Netmask:  255.255.255.0
        DHCP:     Stopped
```

Ifconfig gateway (Configure gateway IP address)

Set the gateway router IP address.

```
> ifconfig gateway 192.168.1.2
> ifconfig
        IP addr:  192.168.1.6
        MAC addr: 00:BA:BE:11:00:77
        Netmask:  255.255.255.0
        Gateway:  192.168.1.2
        DHCP:     Stopped
```

Ifconfig MAC (Configure hardware MAC address)

Set the layer 2 MAC address. This address may only be set in idle mode. The uIP stack and G2100 device are updated during connection establishment, not immediately. The MAC address should be formatted with semicolons, so the tool can properly parse the string: xx:xx:xx:xx:xx:xx. The tool expects 6 hexidecimal values, separated by semicolons. A leading '0x' (ie 0xAA) is not expected for the hex values.

```
> ifconfig
Why did this all of a sudden get a 0.0.0.0 IP address?
        IP addr:  0.0.0.0
        MAC addr: 00:BA:BE:11:00:77
        Netmask:  0.0.0.0
        DHCP:     Started

> ifconfig 00:23:45:67:89:ab
> ifconfig
        IP addr:  0.0.0.0
        MAC addr: 00:23:45:67:89:AB
        Netmask:  0.0.0.0
        DHCP:     Started

> iwconfig mode managed
> ifconfig
        IP addr:  192.168.1.145
        MAC addr: 00:23:45:67:89:AB
        Netmask:  255.255.255.0
        DHCP:     Started
```

Ifconfig auto-dhcp (Control DHCP client)

This parameter is only available if the DHCP client is compile time enabled. If the DHCP client is not compile time enabled, the command is undefined, and the CLI will return "Param 1 invalid." The "auto-dhcp" param follows a SUN-Solaris O/S nomenclature. This parameter may be used to start and stop the SDK's DHCP client task. A started DHCP client (ie "start") will endlessly attempt to find a DHCP server, or renew it's DHCP lease. A stopped (ie "drop") DHCP client will not search for DHCP server. Therefore, the user must assign a static IP address.

```
> ifconfig auto-dhcp stop
> ifconfig
        IP addr:  192.168.1.65
        MAC addr: 00:1E:C0:00:00:00
        Netmask:  255.255.0.0
        Gateway:  192.168.1.1
```

```
        DHCP:     Stopped
> ifconfig auto-dhcp start
> ifconfig
        IP addr:  192.168.1.65
        MAC addr: 00:1E:C0:00:00:00
        Netmask:  255.255.0.0
        Gateway:  192.168.1.1
        DHCP:     Started
```

iwpriv commands

Iwpriv is an implementation specific tool, provided by ZeroG. This tool augments the capabilities of iwconfig. Specifically, iwpriv is a command line tool for the configuration of a G2100 security context. This tool can only modify context in the idle state: "iwconfig mode idle." The context is committed when the Adhoc or Managed networks are started. However, iwpriv status may be displayed in any state. When no parameter is specified, iwpriv will display status for the current security context and mode.

SYNOPSIS

iwpriv [auth (open|shared)]
[enc (wep | wpa-psk)]
[key <key-index> <key>]
[key <key-index>]
[phrase <passphrase>]

IWPRIV EXAMPLES

The following examples illustrate the configuration of the encryption settings.

Display encryption settings by iwpriv

```
> iwpriv

Encryption: none
```

Iwpriv auth (Configure authentication)

This parameter allows the user to select the WEP authorization mode: open or shared. This parameter is meaningful only with WEP encryption.

```
> iwpriv auth open
> iwpriv auth shared
```

Iwpriv enc (Configure encryption mode)

This parameter allows the user to select the encryption mode: wep or wpa-psk. This param allows the user to switch between "wep", "wpa-psk", “wpa-phrase” or "none". WPA key management allows either a pre-calculated: "wpa-psk", or the G2100 may make the calculation using a passphrase: "wpa-phrase". This parameter may be useful, if the user has already entered a passphrase, WEP key, or PSK. Users can change security context, without re-entering the prior data.

```
> iwpriv enc wep
> iwpriv enc wpa-psk
> iwpriv enc wpa-phrase
> iwpriv enc none
```

Iwpriv key (Configure WEP key)

ZG2100 accepts WEP keys only in hex mode. So key must be either 10 (64 bit key)or 26 (128 bit key) hex characters.

```
> iwpriv key [1] 0x1234567890
> iwpriv key [2] 1234567890
> iwpriv key [3] 0xabcdefabcd
> iwpriv key [4] abcdefabcd
> iwpriv key [1]
>
>iwpriv

Encryption:wep
Auth:open
*wep key[1]:0x1234567890
 wep key[2]:0x1234567890
 wep key[3]:0xabcdefabcd
 wep key[4]:0xabcdefabcd
>
```

Iwpriv key [index] (Configure WEP key index)

```
> iwpriv key [4]
>
```

```
>iwpriv

Encryption:wep
Auth:shared
 wep key[1]:0x1234567890
 wep key[2]:0x1234567890
 wep key[3]:0xabcdefabcd
*wep key[4]:0xabcdefabcd
>
```

Iwpriv phrase (Configure pass phrase)

This parameter configures a pneumonic string, which the G2100 uses to calculate a PSK. Both passphrase and the SSID are used in the PSK calculation and a user friendly SSID may be set with "iwconfig ssid <name>". The user may pass a string with whitespace, but this string must be enclosed by double quotes: "ZeroG G2100 passphrase". Otherwise, the passphrase is assumed to be one run-on word. The pass phrase must be at least 8 chars as a min, and no greater than 64 chars as a max. The G2100 will take roughly 30secs to make the PSK calculation. If a valid passphrase is provided, the encryption mode is automatically set to "wpa-phrase", so there is no need to enter "iwpriv enc wpa-phrase" at the command line.

```
>
>iwpriv

Encryption:none

> iwpriv enc wpa-psk
> iwpriv phrase abcdefghijklm

> iwpriv
Encryption: wpa-phrase
  Phrase: “abcdefghijklm”
  SSID:  zgap
```

WiFi iperf example

The following example illustrates the use of several of the CLI commands.

Loading the WiFi iperf application

The following instructions assume that you have the hardware connected up (AP and PiCTAIL), and have run the Demo in the "ZGS2101 ZeroG System Development Kit for WiFi PICtail using PICDEM.net 2 and Explorer 16."

If you are running the Demo from the above document, then stop it by clicking the 'pause' button in the MPLAB IDE. Then ooen a different project by going to "Project>Open…"

The files displayed are located in:

C:\Microchip Solutions\WiFi Iperf App\

There are 2 project files:

WiFi Iperf App-C18
Open this file if your Microchip board is using a PIC18 microcontroller
WiFi Iperf App-C30
Open this file if your Microchip board is using a PIC24 or the dsPIC microcontroller

Select the TCPIP WiFi Demo App appropriate for your development board.

Compile the project, Program to target, and Release from reset

Check in your hyperterminal (from Demo 1). You should now have a simple command line interface. If SSID of your AP and device match, the device will automatically connect and you should see several messages including the following. Notice the new IP Address – it should match the IP address on the LCD screen of your setup.

```
…
Associate ...
   succeeded
Connected!
New IP Address: 192.168.1.100
================================================================================
* ZeroG Host Interface Monitor
* (c) 2008 -- ZeroG, Inc.
*
* Type 'help' to get a list of commands.
================================================================================
>
```

Check wireless connection with 'ping'

This is an example to measure the latency of the wireless connection between the AP and the ZG2100.

Verify connection from the hyperterminal

Start with the reset command in the hyperterminal. If the default settings are the same as the access point, it may connect automatically. If it does, skip ahead to 4.2.2. A possible failure is shown below.

```
>  reset

ZeroG v1.3.4.10

ZeroG Firmware Version 0x1103

New IP Address: 192.168.1.65
Set MAC addr ...
   mac = 00:1E:C0:00:00:00
Set regional domain ...
   region = fcc
Scan ... for ssid "MKT_DemoAP4"
   AP is configured on different channel
```

In the case of this connection failure, we did not have the ssid correct. By connecting up the PC to the access point directly with an Ethernet cable, we can log into the access point and find the ssid. In our case, we need to set the ssid to MKT_DemoAP5 as shown below. We can then try connecting again with the iwconfig command shown.

```
>  iwconfig ssid MKT_DemoAP5
> iwconfig mode managed

ZeroG Firmware Version 0x1103
Set MAC addr ...
   mac = 00:1E:C0:00:00:00
Set regional domain ...
   region = fcc
Scan ... for ssid "MKT_DemoAP5"
   [0] bssid = 00:23:69:3F:E6:9E
      strength = ####.
      indicator = 176
      ssid = MKT_DemoAP5
      channel = 11
   AP selected = [0]
Join ...
   succeeded
```

```
Authenticate ...
   succeeded
Associate ...
   succeeded
Connected!

New IP Address: 192.168.1.100
```

This is a successful connection, and we are given a new IP address.

Verify connection from PC/access point

With the access point connected to the PC via an Ethernet cable, go to 'Start/Run' from the PC. Then type 'cmd' and press OK. This should bring up a C:\WINDOWS\system32\cmd.exe window. In this window, type in a ping command that continuously sends data to the device under test. The transfer time reported should be in the range 1-100ms. The following commands are from the cmd window on the pc – not from the hyperterminal.

```
> ping 192.168.1.100 –t

Reply from 192.168.1.100: bytes=32 time=8ms TTL=100
Reply from 192.168.1.100: bytes=32 time=13ms TTL=100
```

Verify ZeroG power save with 'ping'

After the command sequences shown in 4.2.1-4.2.2, set the device in power save mode as shown in the hyperterminal window below. The iwconfig command should then display the pwrsave enabled and the dtim rx enabled as shown.

```
> iwconfig power reenable

> iwconfig
      channel:  1,6,11
      domain:   fcc
      rts:      2347
      mode:     managed
      ssid:     MKT_DemoAP5
      pwrsave:  enabled
      dtim rx:  enabled
```

By observing the ping (still going on in the cmd window of the pc), you should notice that the transfer time is now increased a wider range (10 – 500ms). This is due to the fact that the device only wakes up at certain intervals of the beacon to check for data. This could be affected by settings on the access point (e.g. advanced wireless settings – beacon interval and dtim interval on a Cisco Linksys router). Notice that that the increased latency is only over the air. In power save, you can still wake up the host

from the hyperterminal with a short latency.

WiFi adhoc network examples

The following examples illustrate how to set up an adhoc network with the ZG2100 and another client. The requirement is to have a similar setup to that used in section 4 with the ZeroG PICtail. In addition, you need to have another WiFi device. It could be a laptop with WiFi, a phone with WiFi (e.g. iPhone), or a WiFi wireless USB adapter connected to a PC. In the example below, a Netgear wireless USB adapter (WG111T) was used.

Adhoc network without security

Start with the reset command in the hyperterminal. If the default settings are the same as an access point in range, it may connect automatically. If it does connect, then change the mode to idle as shown below. If it doesn't connect, you can omit the 'iwconfig mode idle' command, although it doesn't hurt to issue it twice.

```
> reset
> iwconfig mode idle
```

Make sure that the encryption is turned off.

```
> iwpriv enc none
> iwpriv

Encryption: none
```

Before starting the adhoc network, view the available wireless networks from the PC, laptop or phone. Then pick a unique name for your new adhoc wireless network. Let's say that 'MyAdHoc' is not currently used. The status from an iwconfig command should be as shown below.

```
> iwconfig ssid MyAdHoc
> iwconfig
      channel:  1,6,11
      domain:   fcc
      rts:      2347
      mode:     idle
      ssid:     MyAdHoc
      pwrsave:  disabled
```

Start the new adhoc network by the following command:

```
> iwconfig mode adhoc
>
```

```
Set MAC addr ...
   mac = 00:1E:C0:00:00:00
Set regional domain ...
   region = fcc
Install WEP...
   succeeded
Scan ...
   starting network ...
Start ...
   succeeded
```

Look at your available wireless networks from your PC, laptop, or phone – refresh the network list. You should be able to see your new network SSID. In the USB/WiFi with a PC, it is shown as an 'unsecured computer-to-computer network'.
Try connecting to this network with your PC, laptop, or phone. If these steps were followed, it should connect. This example is not complete. Since there is not an AP giving out IP addresses (DHCP), we need to do it. This is covered in the next example along with WEP encryption.

Adhoc network with WEP encryption

Start with the reset command in the hyperterminal. If the default settings are the same as an access point in range, it may connect automatically. If it does connect automatically, then change mode to idle as shown below.

```
> reset
> iwconfig mode idle
```

Change encryption to WEP.

```
> iwpriv enc wep
> iwpriv

Encryption: wep
  Auth: open
* Wep key[1]:  0x000102030405060708090a0b0c
  Wep key[2]:  0x1011121314151617181 91a1b1c
  Wep key[3]:  0x2021222324252627282 92a2b2c
  Wep key[4]:  0x3031323334353637383 93a3b3c
```

Change the WEP key to something that is easier to remember or type. Also, set the key index to 1. Notice that we do not need to enter the 0x before the hex values.

```
> iwpriv key [1] 1234567890
> iwpriv key [1]

Encryption: wep
  Auth: open
* Wep key[1]:  0x1234567890
  Wep key[2]:  0x1011121314
  Wep key[3]:  0x2021222324
  Wep key[4]:  0x3031323334
```

Set the SSID to MyAdHoc and turn off the auto-dhcp so that we can give a static IP address. The address below is arbitrarily chosen.

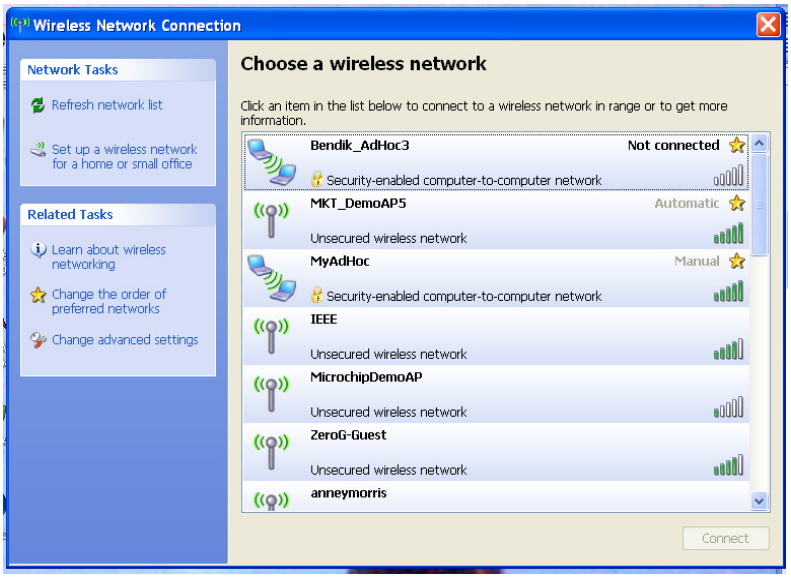
```
> iwconfig ssid MyAdHoc
> ifconfig auto-dhcp drop
> ifconfig 192.168.12.34
> ifconfig
        IP addr:  192.168.12.34
        MAC addr: 00:1E:C0:00:00:00
        Netmask:  255.255.0.0
        Gateway:  192.168.1.1
        DHCP:     Stopped
```

Start the new adhoc network by the following command:

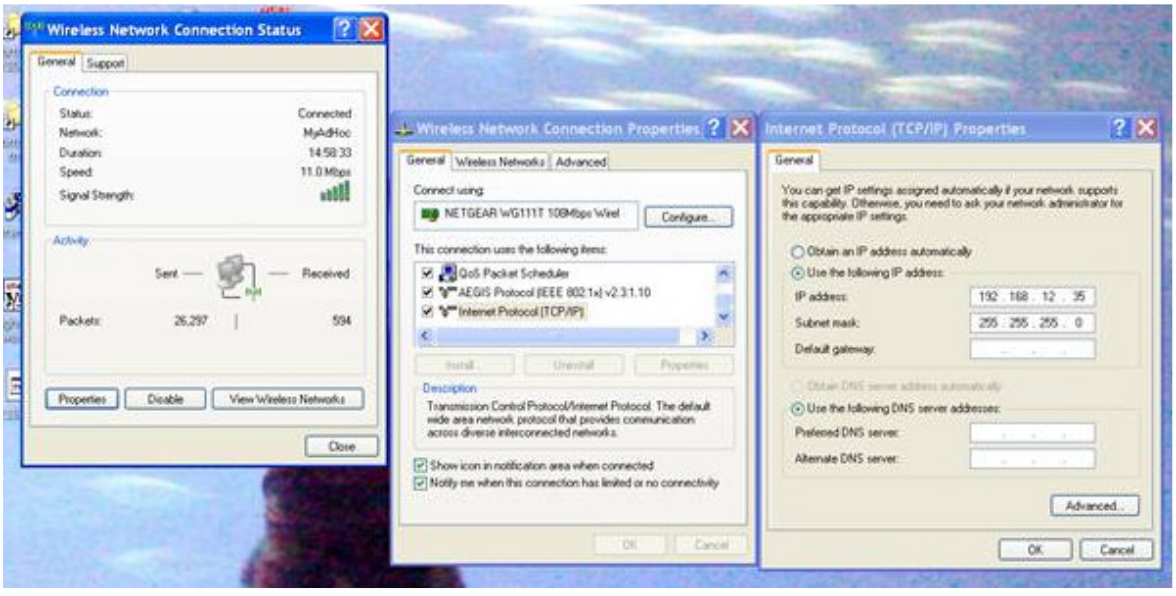
```
> iwconfig mode adhoc

>
Set MAC addr ...
   mac = 00:1E:C0:00:00:00
Set regional domain ...
   region = fcc
Install WEP...
   succeeded
Scan ...
   starting network ...
Start ...
   succeeded
```

Look at your available wireless networks from your PC, laptop, or phone – refresh the network list. You should be able to see your adhoc network SSID. In the USB/WiFi with a PC, it is shown as a 'security-enabled computer-to-computer network'.
You now need to go to the Wireless Network Connection Status or 'view wireless settings' on your PC, Laptop, or phone, you should get a similar window to that shown below.

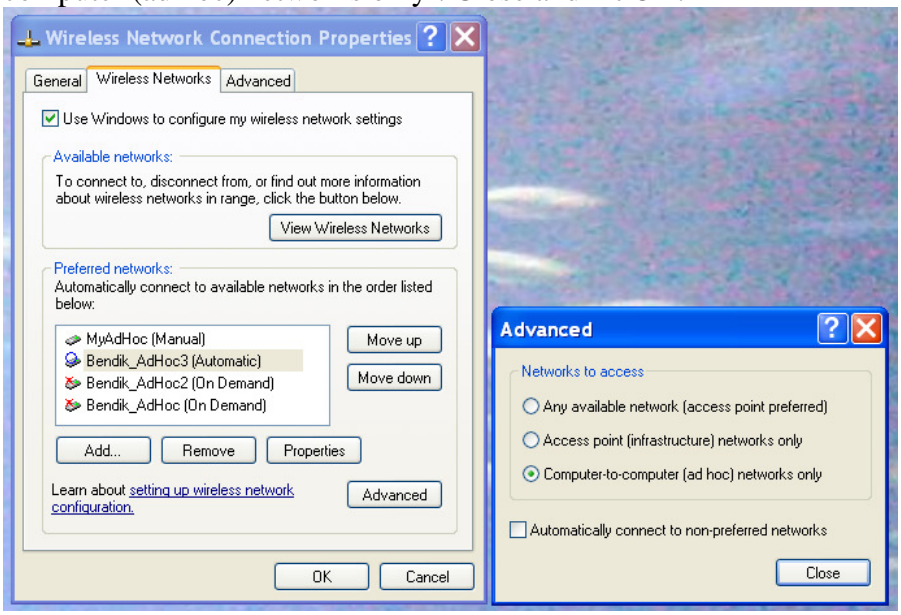
ZeroG Console CLI commands



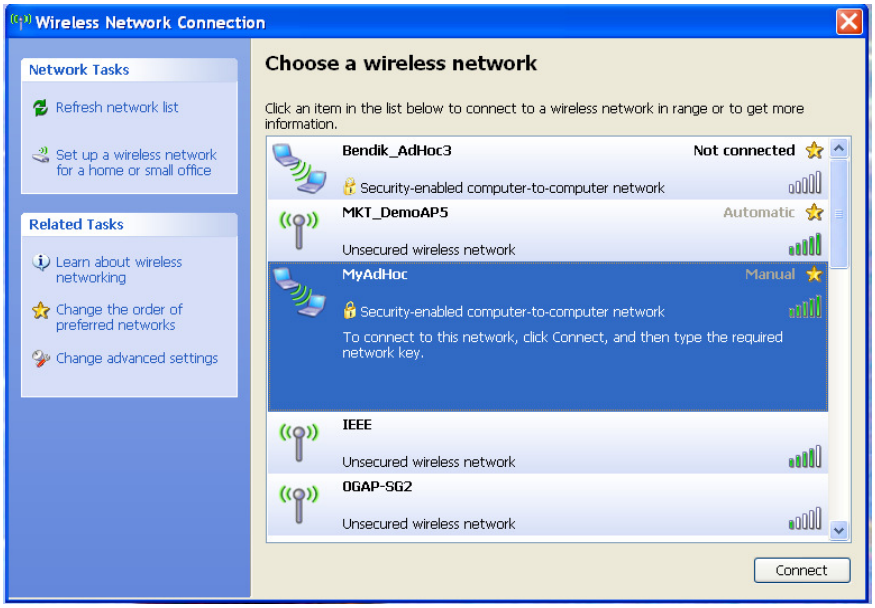
Click on ‘Change advanced settings’, and the left pop-up window below will appear. Click on ‘Properties’ and the middle pop-up window should show. Then scroll down and select Internet Protocol (TCP/IP) as shown, and click on properties. Now the rightmost window should appear. If you were using managed mode before, the ‘Obtain an IP address automatically’ is most likely selected. Change it to ‘Use the following IP address’ as shown below. Then enter the new IP address. It should be different from the one you gave your ZeroG/PIC tail. In the example below, we give it 192.168.12.35. (instead of …12.34). Then click ‘OK’.



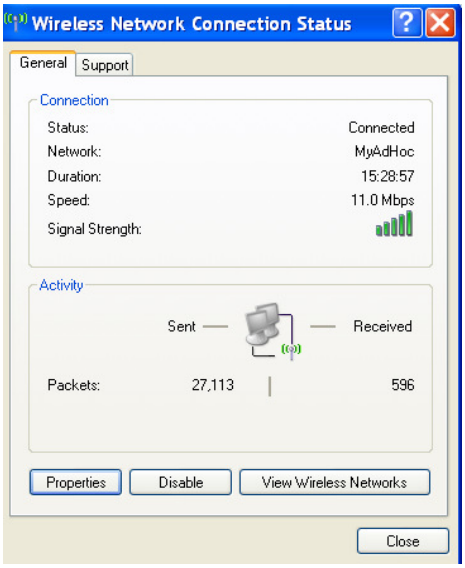
Click OK for the center and right window. Then choose properties again in the left window, and select wireless network. Select 'Advanced' below 'properties' as shown below, and a new small window should appear as shown below on the right. Change it from 'Any available network' to 'Computer-to-computer (ad hoc) networks only'. Close and hit OK.



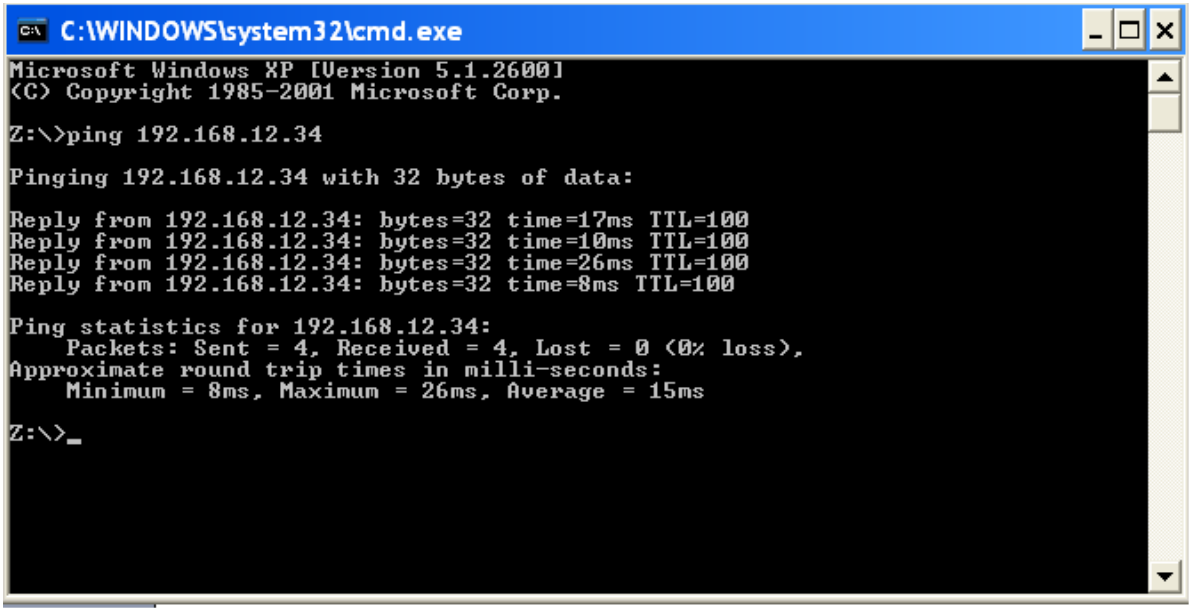
Now, view the available wireless networks, and select 'MyAdHoc' as shown below.



Connect to it, and it should ask for the network key. Use the same key that you entered in the iwpriv key [1] command in your ZG2100 PIC tail (1234567890). Now view the network connection it should show a connection as shown below.



To evaluate the connection, open a 'Run/cmd' window again and ping your ZeroG/PIC tail. You should see a response as shown below.



Congratulations! You have now successfully established a secure wireless adhoc network!

Zero Wireless · 255 San Geronimo Way · Sunnyvale · CA · 94085 · 408 738 7600 · 408 738 7601 fax
www.zerogwireless.com

Abbreviations and vocabulary

Below, a few WiFi abbreviations are included for reference. They are mainly from relevant Wikipedia entries. More details are found at http://www.wikipedia.org.

DHCP: Dynamic Host Configuration Protocol (DHCP) is a network application protocol used by devices (DHCP clients) to obtain configuration information for operation in an Internet Protocol network.

DTIM - A Delivery Traffic Indication Message is a kind of Traffic Indication Message(TIM) which informs the clients about the presence of buffered and/or multicast/broadcast data on the access point. It is generated within the periodic beacon at a frequency specified by the DTIM Interval.[1]

RTS - Request to Send (RTS) frame: The RTS and CTS frames provide an optional collision reduction scheme for access point with hidden stations. A station sends a RTS frame to as the first step in a two-way handshake required before sending data frames.

SSID: (Service Set IDentifier) is a token which identifies an 802.11 (Wi-Fi) network. You must know the SSID to join an 802.11 network. However, the SSID can be discovered by network sniffing. By default, the SSID is part of the packet header for every packet sent over the WLAN. Service set identifiers, or SSIDs, are names used to identify the particular 802.11 wireless LAN(s) to which a user wants to connect. A client device will receive broadcast messages from all access points within range advertising their SSIDs, and can choose one to connect to based on pre-configuration, or by displaying a list of SSIDs in range and asking the user to select one.

Errata

Below, a few errata or observations are included for the current release. We appreciate any feedback, so please let us know if you find any errors or have any suggestions.

1. The hyperterminal may hang if you type while it is printing to the screen.
2. Some commands return prompt '>' (e.g. getzgver, help) and some commands do nt (e.g. 'iwconfig mode idle' after you are in managed mode). You can simply hit the return key one more time to get the prompt back.
3. The 'cls' command (clear screen), does not clear the full screen.

Revision History

Document ID	CLI-101.01	
Title	Console CLI commands	
Revision History	1.0	Initial Revision