Using the kismet_drone to do channel hopping with the prism0 driver

As I understand it, the prism0 (wl) driver on the wrt54g does not allow the kismet_drone to control channel hopping directly.

http://www.renderlab.net/projects/wardrive/wrt54g/openwrt.html suggests the use of a small script running on the wrt to manually channel hop.

```
#!/bin/sh
X=1
while [ $X -eq 1 ] ; do
   sleep 1
   wl scan > /dev/null 2>&1 &
done
```

The 'wl scan' command appears to increment the channel number each time it is executed... so if there are 14 channels, it will take 14 seconds to go through all of the channels 'sleep'ing for one second on each.

On my WRT54GL with DD-WRT the 'wl scan' command does not hop channels as expected. That is why I made my own little hopper script including channel range and dwell time options. The script shows the current channel after hopping.

```
#!/bin/sh
#
# hop.sh    Hop true all channels
#
# Version:  @(#)hop  1.78  06-Feb-2006  ramiro.rikkert [AT] hva.nl
#

# USA channels
#CHANNELS=1,6,11,2,7,3,8,4,9,5,10
#NBHOPS=11

# EU channels
CHANNELS=1,7,13,2,8,3,14,9,4,10,5,11,6,12
NBHOPS=14

# How many seconds to dwell on channel
SECONDS=2

# End of settings

INDEX=0
while true
do
    let INDEX=INDEX+1
    [ $INDEX -gt $NBHOPS ] && INDEX=1
    CURRENT=`echo $CHANNELS | cut -d ',' -f $INDEX`
    echo -n -e "\r\33[KCurrent channel: $CURRENT\r"
    wl channel $CURRENT
    sleep $SECONDS
done
```

You could use the vi editor to enter one of the above scripts. Remember to save the file in /tmp or /jffs (where tmp will delete on router reboot, so it's great for testing) as these are the only parts of the filesystem which are writeable. Also remember to make the script executable with

```
chmod +x <filename>
```

For more details about the wl driver commands, see
http://www.devicescape.com/docs/uwp/package_guide/pkg_broadcom-wl.php

see also http://www.kismetwireless.net/Forum/General/Messages/1099670776.2806771