

GPS on a mesh node

Written by Jim Kinter, K5KTF - Last Updated Sunday, 23 December 2012 11:50

I had received a small Pharos360 GPS unit from a friend of mine who received it free with a software mapping program, and he didn't need it.

It turns out that it is native 3.3V serial output, with a 3-5VDC input, which means you can wire it directly to COM2 of a mesh node, and it fits right in the case (the clear plastic doo-hickey is the GPS unit, wired with regular CAT5 conductors).



GPS on a mesh node

Written by Jim Kinter, K5KTF - Last Updated Sunday, 23 December 2012 11:50

So I wired it in and made that my Mobile node (K5KTF-Mobile). Luckily for me, my 2001 Dodge Ram had very brittle plastic in the dash and a large chunk cracked and fell in, so I have a nice place to set my node and have the longer rubber duckie antenna without hitting the windshield.

So now, what to do software-wise?

First thing is set the serial port to 4800 baud, the output of the GPS.

/etc/init.d/gps:

```
#!/bin/sh /etc/rc.common
# Copyright (C) 2006 OpenWrt.org
START=51

start() {
    stty -F /dev/tts/1 4800
}

stop() {
    killall gps
}
```

That does it every time the unit boots.

Then I made a script to poll the GPGLL line of the GPS output and write that line to a file called "latlon":

/etc/init.d/read-gps:

```
head /dev/tts/1|grep GPGLL >/www/cgi-bin/latlon
```

I then did 'crontab -e' and added a line to run 'read-gps' every minute (the first line is the ID'er, per Part 97):

```
*/5 * * * * /usr/local/bin/fccid  
*/1 * * * * /etc/init.d/read-gps
```

So that updates the 'latlon' file every minute with my current coordinates.

Then I added a chunk to /www/cgi-bin/status (the main status page for a mesh node) that would read 'latlon', scrape out the latitude and longitude, divide the numbers by 100 to get the standard XX.XXXX format, and then print it to the status screen.

```
# page header  
print "<h1><big>$node";  
print " / $tactical" if $tactical;  
print "</big></h1>";  
print "<hr>n";  
  
#lets play with GPS!  
open FILE, "</www/cgi-bin/latlon" or die $!;  
$latmung = substr(<FILE>,7,9);  
close FILE;  
open FILE, "</www/cgi-bin/latlon" or die $!;  
$longmung = substr(<FILE>,20,9);  
close FILE;  
$latitude = ($latmung / 100);  
$longitude = ($longmung / 100);  
  
print "Im currently at $latitude / -$longitude n";  
print "<hr>n";
```

nav buttons

For some reason I had to close and reopen the file again to get the second parameter, as running through it once would not return any value for Longitude. --Im open to suggestions how to fix it.

For those looking at it, this would only work for those between Greenwich U.K, and American Samoa. I cheated, and dont expect to ever drive my truck any further east than the Carolinas or west of California.....

And it looks like this:

K5KTF-Mobile

Im currently at 30°31.1591 / -97°52.7284

Refresh

Mesh Status

OLSR Status

WiFi Scan

Setup

GPS on a mesh node

Written by Jim Kinter, K5KTF - Last Updated Sunday, 23 December 2012 11:50

This may help when your out working on setting up links and need GPS coordinates (and dont have your cell phone handy...).

I will eventually take the data and craft an ax.25 packet to send over the network to an APRS server for inclusion onto the APRS network.

UPDATE 12-23-2012

I have played with the GPS data and the formatting of the page a bit more.

I changed from using the GPGLL stream to the GPGGA, which outputs more info.

To implement, here are the changes:

=====

```
/etc/init.d/read-gps :
```

```
head /dev/tts/1|grep GPGGA >/www/cgi-bin/latlon
```

=====

AND (note that I included parts of the original code before and after so you can see the

GPS on a mesh node

Written by Jim Kinter, K5KTF - Last Updated Sunday, 23 December 2012 11:50

placement of the new code between)

```
=====
/www/cgi-bin/status

# page header
print "<h1><big>$node";
print " / $tactical" if $tactical;
print "</big></h1>";
print "<hr>n";

#lets play with GPS!

#$GPGGA,031914.046,3031.1526,N,09752.7339,W,1,04,5.8,336.5,M,-22.5,M,0.0,0000*48

open FILE, "</www/cgi-bin/latlon" or die $!;
$latdeg = substr(<FILE>,18,2);
close FILE;

open FILE, "</www/cgi-bin/latlon" or die $!;
$latminsec =substr(<FILE>,20,7);
close FILE;

$latdec = ($latdeg + ($latminsec / 60));

open FILE, "</www/cgi-bin/latlon" or die $!;
$longdeg = substr(<FILE>,30,3);
close FILE;

open FILE, "</www/cgi-bin/latlon" or die $!;
$longminsec = substr(<FILE>,33,7);
close FILE;

open FILE, "</www/cgi-bin/latlon" or die$!;
$sats = substr(<FILE>,45,2);
close FILE;

open FILE, "</www/cgi-bin/latlon" or die $!;
$altit = substr(<FILE>,52,5);
close FILE;
```

GPS on a mesh node

Written by Jim Kinter, K5KTF - Last Updated Sunday, 23 December 2012 11:50

```
$altitude = ($altit * 3.280839895);
```

```
open FILE, "</www/cgi-bin/latlon" or die $1;
```

```
$utchh = substr(<FILE>,7,2);
```

```
close FILE;
```

```
open FILE, "</www/cgi-bin/latlon" or die $1;
```

```
$utcmm = substr(<FILE>,9,2);
```

```
close FILE;
```

```
$csth = $utchh - 6;
```

```
if ($csth <0){
```

```
  $csth = $utchh + 18;
```

```
  $ampm = "AM";
```

```
};
```

```
if ($csth >12){
```

```
  $csth = $csth -12;
```

```
  $ampm = "PM";
```

```
};
```

```
$longdec = ($longdeg + ($longminsec / 60));
```

```
print "At $csth:$utcmm $ampm, Im at <a
```

```
href="https://maps.google.com/maps?q=$latdec+%2F+-$longdec+&z=15"
```

```
target="_blank">$latdec / -$longdec </a>, ALT=$altit M / $altitude', using $sats satelites.
```

```
n";
```

```
print "<hr>n";
```

```
# GPS CODE DONE
```

```
# nav buttons
```

```
print "<nobr>n";
```

```
=====
```

I found that the GPGLL may not be as "accurate" or as reliable as GPGLL, but it does give extra data, as far as how many sats used, approx elevation, etc. One reason is that the lat/lon output for GPGLL has a changing decimal point, and since this code counts characters,

sometimes it gets weird. If there is anyone out there who plays in this sandbox more and wants to improve the code and send it back to me, Ill be glad to test/use it, as well as post it here with full credit.

This new code in /www/cgi-bin/status also makes the LAT/LON on the page a link, so that if you have internet access, it will open a new tab/window to a Google map, like so:

K5KTF-Mobile

At 11:45 , Im at [30.5130116666667 / -97.87954](#) , ALT=,0.0, M / 0' , using 00 s

[Help](#)

[Refresh](#)

[Mesh Status](#)

[OLSR Status](#)

[WiFi Scan](#)

[Setup](#)

WiFi address	10.81.108.225 / 8 fe80::218:39ff:fe51:6ce1 Link	Signal/Noise/Ratio	-58 / -92 / 34
default gateway	10.244.112.228 K5KTF-TunnelHost	firmware version	0.4.3
your address	10.159.26.149 K5KTF-USB	configuration	mesh
		system time	Sun Dec 23 2012 17:45:46 UTC
		uptime	23 min
		load average	0.02, 0.05, 0.02
		free space	flash = 624 KB /tmp = 7076 KB memory = 2548 B

GPS on a mesh node

Written by Jim Kinter, K5KTF - Last Updated Sunday, 23 December 2012 11:50

73!
Jim
K5KTF