Best NMEA simulator?

Original Message: "Powell, Christopher M. (ODU) wrote on Fri, 27 Aug 2010

I often find myself in need of a simulated NMEA input (bench testing), what is everyone using for NMEA simulation (Windows and/or Linux)?

I've used NMEAWiz(Win) in the past, as well gpsfeed+(linux), and an old Garmin eTrex in simulator mode.

Any thoughts are appreciated!

later
cp

Christopher Powell
Technical Services
Dept of Ocean, Earth and Atmospheric Sciences
Old Dominion University

Reply From: Brent Evers on Fri, 27 Aug 2010

Plus one for this question. I am also interested in the same, especially one that can be configured to simulate any generic feed (or comes with lots of different data string types) - USBL, DVL, depth (Paroscientific, GP50, etc), altitude (trimech), etc...

So far I have only looked at gpsfeed+, but I was going to test an even older garmin in simulator mode today, so this is very timely...

Brent Evers

Reply From: Val Schmidt (UNH) on Fri, 27 Aug 2010

I typically save example log files of this kind and then when I need them I write a little python script to read a log file and send the data out over a serial line (or to a port via UDP) at some user-defined intervals.

Without any testing and off the top of my head, the script might look like this:

```
#!/usr/local/python

import serial
import time

s = serial.Serial('/dev/ttyS0')
```

```
for line in file('filename'):
s.write(line)
time.sleep(1)
```

-Val

Val Schmidt
CCOM/JHC
University of New Hampshire
Chase Ocean Engineering Lab
24 Colovos Road
Durham, NH 03824

From: Toby Martin (OSU) on Fri, 27 Aug 2010

On the Linux side, I use a real GPS or a script that sends out whatever I tell it.

Toby

```
#!/usr/bin/perl -w

#
# heartbeat: write out a UTC timestamp every second.
#

require "/usr/local/lib/GMTdate";

$VERSION =3D "15 June 2010 by Toby Martin";
$PROGRAM_NAME =3D "heartbeat";

# 15 June 2010 Toby Martin: initial version.

while (1)
{
my $timestamp =3D GMTdate ("%D %H:%M:%S UTC");
print "Vangas heartbeat: $timestamp\n";
sleep 1;
}
~
```

From: Dale Chayes (LDEO) on Fri, 27 Aug 2010

I'm a fan of the script approach and have not switched away from Perl
First and foremost you need to _understand_ exactly what you need.....

In many of the test situations I open a file (or more than one) of real data from real devices and "play it back" with a suitable time interval.

Anthony Johnson has a simulator that handles this well.

You have to watch out for a couple of things in Perl at least:
1) by default output is buffered, so if you want discrete messages at some time (interval) you need to turn off output buffering otherwise you get a bunch of them back-to-back

2) I've had good performance w/ serial ports by using Device::Serial

-Dale

Reply from: Anthony Johnson (LDEO) on Tue, 31 Aug 2010

Christopher,

The script Dale mentioned is a piece of perl code which will 'replay' recorded nmea (or any serial) strings. Our logging system (used on both Langseth and Healy) logs each sentence of a NMEA feed with a user-defined label and a timestamp, like this:

```
cnav 2009:050:23:59:58.7133 $GPVTG,289.1,T,,M,4.47,N,8.28,K*67
```

The code I wrote will play these files back, using the unix time stamp to properly time the data. It is perl, has only been tested on linux, and does not provide any serial interfacing. I pipe it to another program to write to the serial port.

Most of the time I am only interested in the output of our own instruments, and we have nearly complete records for all of them, so this approach works well for us.

When I need something with a current timestamp, I use a custom perl script to generate a sentence, and pipe it to a C program which computes nmea checksums.

These probably aren't what you are looking for, but if you're interested in any of this code, send me a note and I can put them together for you.

Anthony

Reply from: Kurt Schwehr (UNH) on Wed, 1 Sep 2010

For those who don't want to deal with networking or serial code directly, you might check out netcat or socat (a much enhanced netcat like beast).

Here is a quick one liner that will dump a file at whatever connects to a port:

```
socat -T 1 -d -d TCP-L:31414,reuseaddr,fork SYSTEM:"cat my-nmea-ais-log.2008-03-24"
```

If you replace the contents between the quotes with a script that prints out log lines slowly (like the perl script below), you will have a simple replay tool.

socat: http://www.dest-unreach.org/socat/

-kurt