



Changes to the Capture Driver

By Joseph Adamski

Recently, a new setup window was added into the Capture Driver to allow user-defined end-of-lines, the ability to set a custom line length, the ability to handle binary output, and a filter if the device is somehow putting binary into your ASCII stream.

FIGURE 1. Setup Form for the Capture Driver

For example, let's say you have a device that is repeatedly outputting two lines like this:

```
$GPGGA,024056.00,3341.85937,S,15106.89860,E,1,09,0.9,160.0,M,20.4,M,,*7C
$GPVTG,289.41,T,,M,0.06,N,0.11,K,A*3D
$GPGGA,024057.00,3341.85936,S,15106.89860,E,1,09,0.9,160.0,M,20.4,M,,*7C
$GPVTG,129.68,T,,M,0.05,N,0.10,K,A*3D
```

If you are trying to capture everything this device is outputting, you should set the end-of-line method (EOL) to End Seq and enter carriage return and line feed as the end-of-line characters.

FIGURE 2. Capture Driver Output Using End Seq as End-of-Line Method

```
HYPACK Capture Driver
$GPGGA,024101.00,3341.85935,S,15106.89861,E,1,09,0.9,160.0,M,20.4,M,,*7C
$GPVTG,124.09,T,,M,0.01,N,0.02,K,A*30
$GPGGA,024102.00,3341.85934,S,15106.89861,E,1,09,0.9,160.0,M,20.4,M,,*7E
$GPVTG,203.15,T,,M,0.04,N,0.07,K,A*3B
$GPGGA,024103.00,3341.85934,S,15106.89862,E,1,09,0.9,160.0,M,20.4,M,,*7C
$GPVTG,252.05,T,,M,0.01,N,0.02,K,A*3E
$GPGGA,024104.00,3341.85934,S,15106.89862,E,1,09,0.9,160.0,M,20.4,M,,*7B
$GPVTG,20.30,T,,M,0.07,N,0.13,K,A*09
$GPGGA,024105.00,3341.85934,S,15106.89862,E,1,09,0.9,160.0,M,20.4,M,,*7A
$GPVTG,96.97,T,,M,0.12,N,0.22,K,A*0F
$GPGGA,024106.00,3341.85934,S,15106.89862,E,1,09,0.9,160.0,M,20.4,M,,*79
$GPVTG,162.17,T,,M,0.12,N,0.22,K,A*3D
$GPGGA,024107.00,3341.85934,S,15106.89863,E,1,09,0.9,160.0,M,20.4,M,,*79
$GPVTG,125.02,T,,M,0.08,N,0.15,K,A*35
$GPGGA,024108.00,3341.85934,S,15106.89863,E,1,09,0.9,160.0,M,20.4,M,,*76
$GPVTG,302.45,T,,M,0.09,N,0.17,K,A*32
$GPGGA,024109.00,3341.85934,S,15106.89863,E,1,09,0.9,160.0,M,20.4,M,,*77
$GPVTG,323.35,T,,M,0.10,N,0.19,K,A*30
$GPGGA,024110.00,3341.85934,S,15106.89863,E,1,09,0.9,160.0,M,20.4,M,,*7F
```

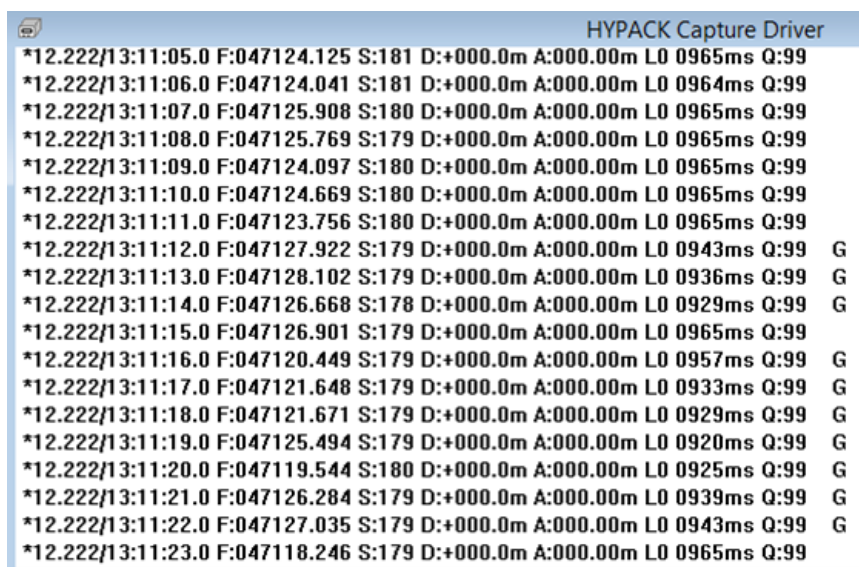
The Beginning of Sequence (Beg Seq) option makes the program start looking at the line starting at the Beg Seq character string and then go until it sees the same the character string again or reaches the “Line Len” (line length specified in number of characters) input.

So let's say you have a bunch of data that looks like this:

```
*12.222/13:10:58.0 F:047117.433 S:180 D:+000.0m A:000.00m L0 0965ms Q:99
*12.222/13:10:59.0 F:047123.754 S:179 D:+000.0m A:000.00m L0 0933ms Q:99      G
*12.222/13:11:00.0 F:047120.216 S:178 D:+000.0m A:000.00m L0 0959ms Q:99      G
*12.222/13:11:01.0 F:047123.704 S:179 D:+000.0m A:000.00m L0 0930ms Q:99      G
*12.222/13:11:02.0 F:047123.933 S:177 D:+000.0m A:000.00m L0 0946ms Q:99      G
```

A good Beg Seq would be * and the Line Length would be 79 to accommodate the occasional 'G' at the end.

FIGURE 3. Capture Driver Output using Beg Seq as EOL method.



The screenshot shows the HYPACK Capture Driver window with a list of data lines. Each line follows the format: *12.222/13:11:05.0 F:047124.125 S:181 D:+000.0m A:000.00m L0 0965ms Q:99. The lines are displayed in a scrollable list, and the output is captured using the Beg Seq method.

```
HYPACK Capture Driver
*12.222/13:11:05.0 F:047124.125 S:181 D:+000.0m A:000.00m L0 0965ms Q:99
*12.222/13:11:06.0 F:047124.041 S:181 D:+000.0m A:000.00m L0 0964ms Q:99
*12.222/13:11:07.0 F:047125.908 S:180 D:+000.0m A:000.00m L0 0965ms Q:99
*12.222/13:11:08.0 F:047125.769 S:179 D:+000.0m A:000.00m L0 0965ms Q:99
*12.222/13:11:09.0 F:047124.097 S:180 D:+000.0m A:000.00m L0 0965ms Q:99
*12.222/13:11:10.0 F:047124.669 S:180 D:+000.0m A:000.00m L0 0965ms Q:99
*12.222/13:11:11.0 F:047123.756 S:180 D:+000.0m A:000.00m L0 0965ms Q:99
*12.222/13:11:12.0 F:047127.922 S:179 D:+000.0m A:000.00m L0 0943ms Q:99      G
*12.222/13:11:13.0 F:047128.102 S:179 D:+000.0m A:000.00m L0 0936ms Q:99      G
*12.222/13:11:14.0 F:047126.668 S:178 D:+000.0m A:000.00m L0 0929ms Q:99      G
*12.222/13:11:15.0 F:047126.901 S:179 D:+000.0m A:000.00m L0 0965ms Q:99
*12.222/13:11:16.0 F:047120.449 S:179 D:+000.0m A:000.00m L0 0957ms Q:99      G
*12.222/13:11:17.0 F:047121.648 S:179 D:+000.0m A:000.00m L0 0933ms Q:99      G
*12.222/13:11:18.0 F:047121.671 S:179 D:+000.0m A:000.00m L0 0929ms Q:99      G
*12.222/13:11:19.0 F:047125.494 S:179 D:+000.0m A:000.00m L0 0920ms Q:99      G
*12.222/13:11:20.0 F:047119.544 S:180 D:+000.0m A:000.00m L0 0925ms Q:99      G
*12.222/13:11:21.0 F:047126.284 S:179 D:+000.0m A:000.00m L0 0939ms Q:99      G
*12.222/13:11:22.0 F:047127.035 S:179 D:+000.0m A:000.00m L0 0943ms Q:99      G
*12.222/13:11:23.0 F:047118.246 S:179 D:+000.0m A:000.00m L0 0965ms Q:99
```