

The HYPACK® Generic Parser, a user-configurable device driver, can be a very handy driver for some unique situations. The driver will read most ASCII strings of data. It can be used to read **Position, Depth, Heading, Speed, Heave Tide** and **Draft.** If you are recording depth, it can also be used to paint the matrix in SURVEY.

Hypack Configuration	Mobile Device Advanced Functions Postion Postion Depth Postion Postio	Offsets Staboard 0.00 NUS Yaw 0.00 deg
🚽 Boat	ic Parser ♥ Position ♥ Depth ♥ Heading ♥ Speed	
5α Hypeok Configuration → Boat └ @ (0) HYPACK® Generic Parser	© Heave ♥ Drat Options □ Use for matrix update	Forward 0.00 HUS Roll 0.00 deg Vertical 0.00 HUS Pitch 0.00 deg Vertical Postive Downward Latency 0.000 sec.
	Setup Test	Connect Seriel Par Seriel Parameters Port CDM8  Speed 9500
		Data bits     B     V     Parity     None     Flow Control     None     V

The connection for this driver must be a serial configuration. The offsets should be set in the same manner that would be used for a standard device driver:

- Latency: The delay time between the GPS and echosounder is only required if you are recording positions.
- Starboard: The distance to the starboard of the origin.
- **Forward**: The distance forward of origin
- **Height**: The distance below the static waterline.

**NOTE:** The GPS antenna height should be negative most of the time to signify it is above the waterline.

## Driver Setup

## FIGURE 2. GenDevParse Driver Setup

On the left you select the driver output items. There must be a checked item for each item in the data string, in the order that it appears in the string. Items in this field can be dragged and dropped in the list to reorder them. Should the string have data that you do not wish to view, you can select **Ignore Field**. The ignore field button will create additional "Ignore Field" items if needed.

On the right, define for the driver what to expect in the format of the incoming data string

**Message Header** is the leading tag that begins each output string to be read. This enables the driver to select

🛲 GenDevParse Setup	_ <b>_</b> X
Ignore Field  Ignore Field  Ignore Field  Ignore Field  Latitude  Y Heading Speed Heave Pitch Roll Depth (Low Frequency) Tide Draft X	Fields delimited by         C Space         Comma         Tab         Other         End of Message         < <cr><lf>         Carriage Return <cr>         Line Feed <lf>         Other         Message Header         \$PSIMGPS         Image: Header is a separate field</lf></cr></lf></cr>
Add Ignore Field Latitude/Longitude Options Degree and Minute Symbols N/S and E/W GPS	Depths in     Speed in <ul> <li>Meters</li> <li>Decimeters</li> <li>Centimeters</li> <li>Feet</li> <li>Fathoms</li> </ul> <ul> <li>MPH</li> <li>Knots</li> </ul> OK

one string type from multiple output string types that may be output by the same device.

**Fields delimited by** is where you will enter the delimitation of the string, be it by space comma, tab or another character.

End of Message is used to signal that the data string being read has ended.

In the example, we are reading a data string from which we want to read position. The raw string looks like this:

\$PSIMGPS,100308190031.50,G,2907.57150,N,09012.99659,W\*0C

This is a GPS string that is comma delimited giving us latitude and longitude. Our header is \$PSIMGPS and our end of message is carriage return, line feed.

The **Driver Output** is a 3-column display in the device window. The first two columns show the items that you selected to display from the data string. The right-most column displays all of the data that is being read.

The **GenDevParse.dll** can be an excellent choice of device driver when non-standard data strings are received into HYPACK®. As long as you know the information contained in your device output data, the GenDevParse driver can make a tricky device setup much less difficult.