## Notes for ADCPBIN2TXT Program:

Input to the adcpbin2txt program is ADCP Workhorse binary data as output in real-time or stored on the flash cards, typically in a series of files named _RDI_000.000, _RDI_001.000, etc.

The –c option selects processing of current data and the –w option selects wave data. Either or both must be selected.

Output from the adcpbin2txt program is comma-separated-values. The –h option outputs double-quote enclosed header lines to identify the data values.

Program Syntax:

```
adcpbin2txt [-c] [-w] [-h] [-e] [-d] ADCP_BINARY_FILE > OUTPUT_TEXT_FILE

where -c = process current ensembles
      -w = process wave pings
      -h = output text header lines
      -e = output error messages
      -d = output debug messages
```

A typical Windows Command Prompt session to concatenate a series of RDI files and process all of them would look something like this:

```
C:\> type _RDI*.000 > rdiall.bin
C:\> adcpbin2txt –h -c -w rdiall.bin > rdiall.csv
```

The following data type records are processed:

Current (-c) Data Type IDs Processed:

| Decimal | Hex | Data Type |
|---|---|---|
| 0 | 0x0000 | Fixed Leader Data ID |
| 128 | 0x0080 | Variable Leader Data ID |
| 256 | 0x0100 | Velocity Data ID |
| 512 | 0x0200 | Correlation Data ID |
| 768 | 0x0300 | Echo Intensity Data ID |
| 1024 | 0x0400 | Percent Good Data ID |

Wave (-w) Data Type IDs Processed:

| Decimal | Hex | Data Type |
|---|---|---|
| 259 | 0x0103 | Wave Fixed Leader ID |
| 515 | 0x0203 | Wave Ping Data ID |
| 771 | 0x0303 | Last Leader Data ID |
| 1027 | 0x0403 | Head/Pitch/Roll Data ID |

No units conversion is done on the data. All data values are output as integers. Some are signed integers. The maximum signed integer value is +/- $2^{31}$ and the maximum unsigned integer value is $2^{32}$.

## Decoding Wave Bitmap Bytes into Selected Bin Numbers:

Bitmap data = 16 bytes
Bins = 128

```
Bitmap: ____Byte-1_____   ________Byte-2________  ...  _____________Byte-16____________
        LSB         MSB   LSB                MSB  ...  LSB                          MSB
Bit:    0 1 2 3 4 5 6 7   0  1  2  3  4  5  6  7  ...    0   1   2   3   4   5   6   7
        | | | | | | | |   |  |  |  |  |  |  |  |  ...    |   |   |   |   |   |   |   |
Bin:    1 2 3 4 5 6 7 8   9 10 11 12 13 14 15 16  ...  121 122 123 124 125 126 127 128

Bit value 0 = bin not selected, 1 = bin selected
```