Document: VIARADAR Serial Output Format

VIARADAR firmware release = version 004

POWER / DATA CONNECTOR

DB-9 Connector Pinout Pin #	Signal	Cable
-	Input Power (8 – 16 VDC)	Red
-	Ground	Black
-	Open Collector Alarm	Yellow
2	RS232 TX	DB9
3	RS232 RX	DB9
5	Ground	DB9

Byte Format:
The output byte format is one start bit, 8 data bits, no parity bit, and one stop bit.

Output Baud Rate:
The user can select the desired baud rate using the VIARADAR configuration software. The available selections are: 19200, 9600, 4800, 2400, or 1200. The default setting from the factory is 9600.

Output Update Rate:
The user can select the desired update rate using the VIARADAR configuration software. The available selections are: 8 times per second, 4 times per second, 2 times per second, 1 time per second, Upon speed change or every 2 seconds (In this setting the radar will send an output data packet every time the target speed changes or every 2 second if there is no change in the target speed), Polled ["*P"] (In this mode the interfaced equipment will poll for an output data packet by sending the two ASCII character '*' and 'P' to the VIARADAR unit. The default setting from the factory is 4 times per second.

Output Protocols:
The VIARADAR- can output data in HEX or ASCII format. The user can select the desired output protocol format using the VIARADAR configuration software. He default setting from the factory is HEX format 0, VIARADAR I standard.
Note: The VIARADAR configuration software can be used to limit targets to approaching only or receding only direction.

Alarm Output:
The open-collector Alarm output line is tied into the Target Speed Low-Limit setting in firmware version 004. The Alarm output line will pull low when a target speed is greater than 5 times the Target Speed Low-Level setting. Example: If the Low-Limit setting is 8 mph, the Alarm output line will pull low when ever the strongest target is greater than 40

mph. The output will stay at a low level for 5 seconds after the strongest speed is no longer above the threshold.

Hex format 0 -- VIARADAR I standard --

Byte:
STX (02)
Number 1 strongest target speed
Number 1 strongest target direction
: : : :
: : : :
Number 8 strongest target speed
Number 8 strongest target direction
ETX (03)

The number of bytes in each packet of this format is determined by how many valid targets are found. The values in the speed bytes are the speed of the targets. The value in the direction bytes is the direction of travel of each target, the value of 1 is an approaching target, the value of 255 is a receding target, and a value of 0 is a target that has no direction such as a tuning fork.

An example of the HEX packet where two targets are present, the strongest target approaching at 35 MPH and a weaker target receding at 50 MPH is as follows:

02 23 01 32 FF 03

Hex Format 1—{02 <strongest speed> <direction> 03}

Byte:
STX (02)
Strongest target speed
Strongest target direction
ETX (03)

The format of this protocol is similar to Hex 0 format, but only the strongest target speed is output. The value in the speed byte is the target speed. The value in the direction byte is the direction of travel of the target, (the value of 1 is an approaching target, the value of 255 is a receding target, and a value of 0 is a target that has no direction such as a tuning fork).

An example of the HEX packet where the strongest target is approaching at 35 MPH:

02 23 01 03

Hex Format 2—{02 <strongest speed> <direction> <signal-to-noise ratio> 03}

Byte:
STX (02)
Strongest target speed
Strongest target direction
Strongest target SNR
ETX (03)

The format of this protocol is similar to Hex 1 format, but the signal-to-noise ration (SNR) of the strongest target speed is output. The value in the speed byte is the target

speed. The value in the direction byte is the direction of travel of the target, (the value of 1 is an approaching target, the value of 255 is a receding target, and a value of 0 is a target that has no direction such as a tuning fork). The value of the SNR byte indicates the ration of the strongest target above the noise floor.

An example of the HEX packet where the strongest target is approaching at 35 MPH:

02 23 01 12 03

Hex Format 3—{02 <strongest speed> <direction> <signal-to-noise ratio> <phase> 03}

Byte:
STX (02)
Strongest target speed
Strongest target direction
Strongest target SNR
Strongest target phase
ETX (03)

The format of this protocol is similar to Hex 2 format, but the phase relationship of the two receivers at the strongest target frequency is output. The value in the speed byte is the target speed. The value in the direction byte is the direction of travel of the target, (the value of 1 is an approaching target, the value of 255 is a receding target, and a value of 0 is a target that has no direction such as a tuning fork). The value of the SNR byte indicates the ration of the strongest target above the noise floor. The phase reading indicates relative direction of motion. Note: this is typically a test protocol.

An example of the HEX packet where the strongest target is approaching at 35 MPH:

02 23 01 12 55 03

Hex Format 4—{02 <MSB strongest speed tenths> <LSB strongest speed tenth> <direction> 03}

Byte:
STX (02)
High byte of strongest target speed tenths
Low byte of strongest target speed tenths
Strongest target direction
ETX (03)

The format of this protocol indicates the speed of the strongest target in tenths of units. The value of the MSB byte is the high byte of the strongest target speed. The value of the LSB is the low byte of the strongest target speed. The value in the direction byte is the direction of travel of the target, (the value of 1 is an approaching target, the value of 255 is a receding target, and a value of 0 is a target that has no direction such as a tuning fork).

An example of a packet where the strongest target is approaching at 35.3 MPH:

02 01 61 01 03

Hex format 28 -- VIARADAR I standard with SNR average --

Byte:
STX (02)

Number 1 strongest target speed
Number 1 strongest target direction
Number 1 strongest target average Signal-to-Noise Ratio (SNR)
: : : :
: : : :
Number 8 strongest target speed
Number 8 strongest target direction
Number 8 strongest target average Signal-to-Noise Ratio (SNR)
ETX (03)

The number of bytes in each packet of this format is determined by how many valid targets are found. The values in the speed bytes are the speed of the targets. The value in the direction bytes is the direction of travel of each target, the value of 1 is an approaching target, the value of 255 is a receding target, and a value of 0 is a target that has no direction such as a tuning fork. The value in the SNR byte is the averaged signal-to-noise ration for that target.

An example of the HEX packet where two targets are present, the strongest target approaching at 35 MPH and a weaker target receding at 50 MPH is as follows:

02 23 01 12 32 FF 09 03

Hex format output #29: -- VIARADAR I standard with amplitude in dB.

STX (02)
Number 1 strongest target speed
Number 1 strongest target direction
Number 1 strongest target amplitude in dB
: : : :
: : : :
Number 8 strongest target speed
Number 8 strongest target direction
Number 8 strongest target amplitude in dB
ETX (03)

The number of bytes in each packet of this format is determined by how many valid targets are found. The target data is output by target signal strength, with the strongest target output first. The data for each valid target is in three byte groups, the speed of the target, the direction of the target (0x01 = approaching, 0xFF = receding, 0x00 = non-moving), and the signal amplitude of the target signal in dB.

Hex format output #30: -- VIARADAR I standard with duration tracking

STX (02)
Number 1 strongest target speed
Number 1 strongest target direction
Number 1 strongest target duration tracking
: : : :
: : : :
Number 8 strongest target speed
Number 8 strongest target direction

Number 8 strongest target duration tracking
ETX (03)

The first byte in the message is the start of transmission 'stx' 0x02. The target data is output by target signal strength, with the strongest target output first. The data for each valid target is in three byte groups, the speed of the target, the direction of the target (0x01 = approaching, 0xFF = receding, 0x00 = non-moving), and the accumulated signal-to-noise ratio of the target signal.

Hex format output #31: -- Strongest speed with log status
STX (02)
Strongest target speed
Strongest target direction
Strongest target duration tracking
Log status
ETX (03)

The target data is output for the strongest target only. The data for a valid target is in a four byte group, the speed of the target, the direction of the target (0x01 = approaching, 0xFF = receding, 0x00 = non-moving), and the accumulated signal-to-noise ratio of the target signal, and the log status. The log status byte is used to track multiple targets and send a message as to when a new target could be stored for data gathering. The radar will send a single 0x01 byte for the log status when it has tracked a target over a set DuraTrak level. Prior to and following this, the log status will be 0x00, indicating to not log the target yet or that the target has already been logged.

Hex format output #32: -- Log strongest speed
STX (02)
Strongest target speed
Strongest target direction
ETX (03)

The target data is output only when the strongest target should be logged. The data for a valid target is in a two byte group, the speed of the target, the direction of the target (0x01 = approaching, 0xFF = receding, 0x00 = non-moving). The packet will be sent when it has tracked the target over a set DuraTrak level. Prior to and following this, no packet will be sent, indicating the target has not reached the log setting yet or that the target has already been logged. Note: This is a special order protocol.

ASCII Format 64 – {<dir> [sss] <cr>}
This output is in Ascii format. The <dir> byte character can be '0' (no target), '?' (non-directional target), '+' (approaching target), or '-' (receding target). The three sss bytes are

the hundreds, tens, and units characters of the target speed. <cr> is the carriage return character (0x0D).

ASCII Format 65 – { [<d>['S'][sss](CR)<7 bit checksum>]}
This output is in ASCII format. The <dir> byte character can be '0' (no target), '?' (non-directional target), '+' (approaching target), or '-' (receding target). The 'S' character is next. The three sss bytes are the hundreds, tens, and units characters of the target speed. <cr> is the carriage return character (0x0D). The <7 bit checksum> is the truncated 7 bit checksum value of the proceeding bytes.

ASCII Format 66 – {<dir> [sss'.'s] <cr>}
This output is in ASCII format. The <dir> byte character can be '0' (no target), '?' (non-directional target), '+' (approaching target), or '-' (receding target). The five sss.s bytes are the hundreds, tens, units, decimal point and tenths characters of the target speed. <cr> is the carriage return character (0x0D).

ASCII Format 67 – {<dir> [sss'.'s] ',' [AAA] <cr>}
This output is in ASCII format. The <dir> byte character can be '0' (no target), '?' (non-directional target), '+' (approaching target), or '-' (receding target). The five sss.s bytes are the hundreds, tens, units, decimal point and tenths characters of the target speed. The comma character byte ',' separates the speed block from the amplitude block. The three AAA bytes are the hundreds, tens, and units characters of the target amplitude. <cr> is the carriage return character (0x0D).

ASCII Format 68 – {[sss] <cr>}
This output is in ASCII format. The three sss bytes are the hundreds, tens, and units characters of the target speed. <cr> is the carriage return character (0x0D).

ASCII Format 69 – { [<d>['S'][xss](CR)<7 bit checksum>]}
This output is in ASCII format. The <dir> byte character can be '0' (no target), '?' (non-directional target), '+' (approaching target), or '-' (receding target). The 'S' character is next. The target speed is in two or three bytes dependent on its' value. The bytes are hundreds, tens, and units characters of the target speed. <cr> is the carriage return character (0x0D). The <7 bit checksum> is the truncated 7 bit checksum value of the proceeding bytes

ASCII Format 70 – {[sss'.'s] <cr>}
This output is in ASCII format. The five SSS.S bytes are the hundreds, tens, units, decimal point and tenths characters of the target speed. <cr> is the carriage return character (0x0D).

ASCII Format 71 – {<'*'> [sss'.'s] ',' [AAA] <cr>}
This output is in ASCII format. The <*> byte character is the start of message. The five sss.s bytes are the hundreds, tens, units, decimal point and tenths characters of the target speed. The comma character byte ',' separates the speed block from the amplitude block.

The three AAA bytes are the hundreds, tens, and units characters of the target amplitude. <cr> is the carriage return character (0x0D).

ASCII Format 72 – {<'*'> [sss'.'s] ',' [AAA] <cr>}
This output is similar to output 71. Except the packet is sent only if there is a strongest speed, no data is sent if there is no target. The <*> byte character is the start of message. The five sss.s bytes are the hundreds, tens, units, decimal point and tenths characters of the target speed. The comma character byte ',' separates the speed block from the amplitude block. The three AAA bytes are the hundreds, tens, and units characters of the target amplitude. <cr> is the carriage return character (0x0D).