

## 6. Host Computer Commands

This chapter describes the digital commands that the host computer must use to set up and control the RVP8 processor for recording data. Each command is described in detailed in a separate section of this chapter. Note that a command mnemonic, or shorthand reference name, is given in each section heading. These names are frequently used to refer to particular commands.

The write-up for each command includes a description of what the command does and a pictorial layout of the bits in the 16-bit command word. Commands consist of an initial command word containing an opcode in the low five bits. If additional arguments are required, they are listed as "Input 1", "Input 2", etc. Finally, if the command produces output, those words are listed as "Output 1", "Output 2", etc. Often each word is broken down into several independent fields, each consisting of one or more bits. In such cases, the pictorial layouts show the placement of the bit fields within the word, and each field is described individually. All data transferred to or from the RVP8 are in the form of 16-bit words.

Before attempting to program the RVP8, it is a good idea to at least skim through the descriptions of every command. The instruction set has been designed to be as concise and orthogonal as possible. User programs should always execute the IOTEST command on power-up to ensure that the interface connections are all intact. The diagnostic result registers from GPARM should also be checked initially to verify that the RVP8 passed all internal checks. Since all internal RVP8 tables and parameters are set to reasonable values on power-up, it is conceivable that PROC commands could be issued immediately to acquire and process radar data. More realistically, however, the default information is first modified to meet the users needs.

To set up for data acquisition and processing the following sequence of commands might be executed. Trigger and pulse width are first established using the SETPWF commands. Range bin placement and processor options are then chosen using LRMSK, and SOPRM, and receiver noise samples are taken with SNOISE. The noise levels are not automatically sampled on power-up, so SNOISE must be issued at least once by the user. LFILT is executed if clutter filters are needed. If data rays are to be synchronized with antenna motion, then LSYNC is used to specify a table of antenna angles. After all setups are complete, PROC commands are issued to actually collect, process, and output the data. Errors detected during the execution of commands are noted by the RVP8 and can be monitored using GPARM.

The RVP8 contains a 4096-word first-in-first-out (FIFO) buffer through which all output data flow. This buffer is included to simplify the requirements of the user's interface hardware. The FIFO holds each sequential word generated by the RVP8 until such time as the user is ready to accept it. Thus, when reading from the processor, it is permissible to fall behind by as many as 4096 words before any slowdown in performance occurs. The RVP8 writes to the FIFO at full speed as long as it is not full, and the internal processing is not affected by the exact speed at which user I/O actually occurs. This continues as long as the average I/O rate on, perhaps 10ms intervals, matches the average rate at which data are being produced.

The sequence of events described above is altered when the FIFO becomes completely full. Then, when the processor generates the next output word, it waits in an idle loop until the user makes room in the FIFO by reading out one or more words. Until this space becomes available, the RVP8 simply waits and does not proceed any further with its internal processing. This, of course, leads to a slowdown in performance, but it is not a disastrous one. The user always obtains correct data no matter how long it takes to read it. One could take advantage of this fact to synchronize the acquisition of data by the RVP8 with the post-processing and display of that data by the user. In this case, RVP8 would be instructed to output data at the maximum rate, the user would read these words at the user's maximum rate, and the overall system would automatically run at the slower of those two speeds.

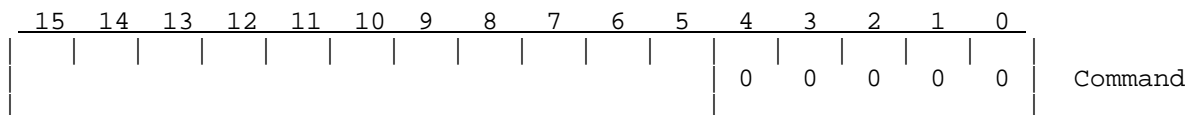
When the output FIFO is full and the RVP8 has the next word ready for output, there is another way that the idle wait loop can be exited, that is, if the processor detects that the user is performing a write I/O cycle. Since the user should have been reading data by now, the presence of a write cycle is taken to mean that some more important condition has arisen. As such, the wait loop is terminated and the RVP8 accepts the write data soon afterward. If the new data are commands, they are executed right away, but any output they try to produce may be lost in a similar manner. The net effect is that the processor continues to execute all commands correctly, but that their output is discarded.

The discarded output data are not in fact lost. Rather, the data are eventually replaced with an equal number of zeros. Each time the RVP8 discards an output word, it also increments an internal 24-bit count. When FIFO space becomes available in the future, the processor replaces all of the missing data with zero-valued placeholders.

Writing when the FIFO is full can be particularly useful if the new command is a RESET which calls for clearing of the output FIFO. When the RESET is processed, all past and present output data are discarded, leaving the RVP8 output section completely empty. This is useful whenever the processor has pending output data which the user wants to truly throw away.

## 6.1 No-Operation (NOP)

This single-word instruction is simply ignored by the the Signal Processor. The NOP is useful when a number of words are to be flushed through the RVP8 with no side effects.



## 6.2 Load Range Mask (LRMSK)

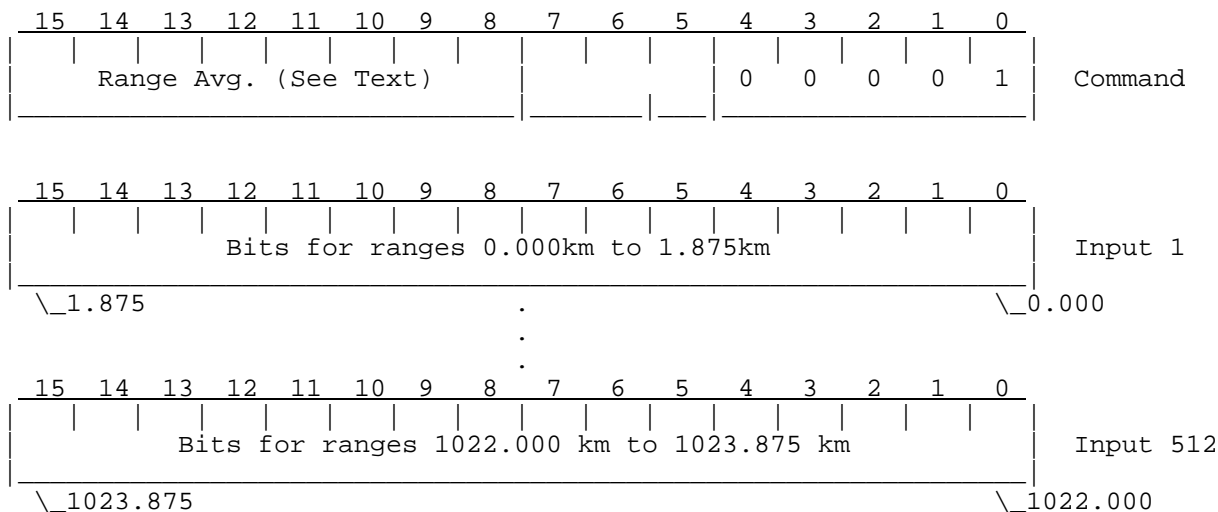
This command informs the signal processor of the ranges at which data are to be collected. An arbitrary set of range bins are selected via an 8192-bit mask. The Nth bit in the mask determines whether data are acquired and processed at a range equal to  $RES \times (N-1)$ . The Range resolution is specified by a TTY setup question (see section 3.2.5), in the range 25 through 1000 meters. Any collection of ranges may be chosen from integer multiples of that distance. The example below is given for the default resolution of 125 meters. The range mask is passed to the RVP8 packed into 512 16-bit words. The least significant bit of each packed word represents the nearest range, and the most significant bit represents the furthest range in each group of 16.

According to the range bins that are selected in the mask, the signal processor computes and stores internally a range normalization table which is later used to convert receiver intensity levels into reflectivity levels in dBZ. Note that the LRMSK command implicitly specifies the number of bins to be processed and output. The maximum bin count is 3072, though depending on the computational intensity of the configuration, the RVP8 may be able to compute fewer bins. If the number of bins selected in the bit mask exceeds this maximum, the trailing bins are truncated. If the new mask does not specify any active bins, then a single bin at range zero is forced on. The default power-up mask selects 256 bins equally spaced by 1.0km starting from zero range.

Range averaging is also determined by LRMSK. The upper byte of the command controls how many consecutive bins are grouped together. A value of zero means no averaging; one means that pairs of samples are averaged; 255 means that 256 terms are summed, etc. The individual samples that go into each average are still taken according to the bits that are set in the mask, except that they are now grouped together so that only one net bin results from the several data samples. Note that the limitation of 3072 sampled ranges applies to the bin count prior to averaging.

For example, suppose 100 bits are selected in the range mask and no averaging is elected. Then parameters are computed at those 100 ranges, and 100 bins of data are output. If the averaging were set to one, rather than zero, samples would still be taken at the same ranges, but pairs of bins would be averaged together and only 50 ranges would result. Note that the parameters are averaged by summing the autocorrelations for each bin. The range normalization value associated with the averaged bin is computed according to the midpoint of the first and last sample.

Incompletely averaged bins are discarded by the LRMSK command. In the above example, if the averaging were set to two so that triples of samples were summed, then only 33 bins would be output. This is because the 100-bit mask left a dangling 100th sample. In the extreme case where there are not enough mask bits to result in even one complete bin, the RVP8 forces the averaging to zero and turns on a single bin at zero range.



## 6.3 Setup Operating Parameters (SOPRM)

This command is used to configure the Signal Processor. The command should be issued whenever any of the parameters in the list change. The default parameter list consists of twenty 16-bit input words. These can be followed by optional XARG parameters as needed.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
							NTh				0	0	0	1	0	Command

**NTh** If 1, then no threshold values are set. This means ignore input words 4, 5, 6, 7, 11, 12, 13, 14, and 18. This is usually used in conjunction with the THRESH command (see section 6.29), when setting individual thresholds.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
																Input 1
Sample Size																

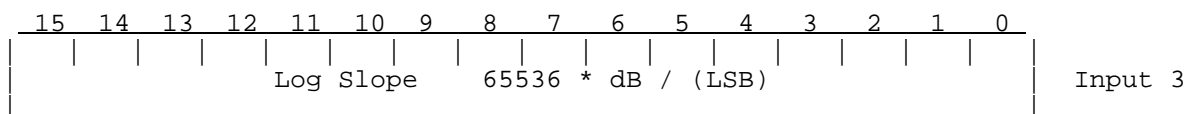
The sample size is continually adjustable from 1 to 256 pulses. However, during the alternating polarization mode, the sample size must be even. If an odd value is entered it is rounded up by one in that case.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
		Polar	NHD	ASZ	16B	CMS	R2		3x3			Lsr	Dsr	Rnv		Input 2

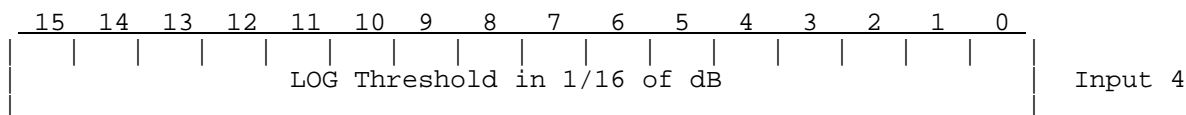
Each of the single-bit fields selects whether the given processing or threshold option is enabled (1) or disabled (0).

- Polar** Configures transmit polarization and Zdr processing:
  - 00 – Fixed polarization, Horizontal
  - 01 – Fixed polarization, Vertical
  - 10 – Alternating polarization pulse-to-pulse
  - 11 – Dual simultaneous transmission
- NHD** Disables inclusion of header words in the processed data that are output by the PROC command (See also, CFGHDR command).
- ASZ** The “Any Spectrum Size” bit requests that DFT processing algorithms, clutter filters, spectral output, etc. all operate on spectra whose size exactly matches the number of available pulses (rather than rounding the spectrum size down to the next lower power-of-two).
- 16B** Configures for 16-bit (rather than 8-bit) data output from the PROC command. This bit affects the single-parameter versions of Reflectivity, Velocity, Width, and Zdr data. However, the PROC command’s archive format always holds 8-bit data, regardless of the setting of 16B. This gives the option of extracting both 8-bit and 16-bit data simultaneously from each ray.
- CMS** Enables Clutter Microsuppression, in which individual range bins are rejected (based on excessive clutter) prior to being averaged together in range.
- R2** Use three lag (R0/R1/R2) algorithms for width, signal power, and clutter correction.

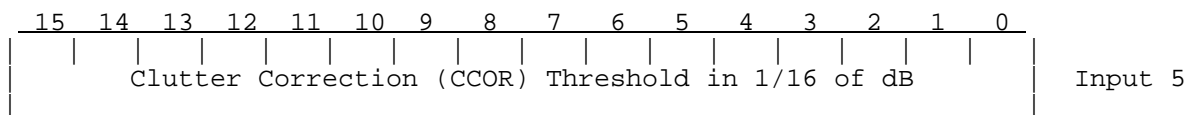
- 3x3 Switches on the 3x3 output filter (See Section 5.3.3). The RVP8 automatically handles all of the pipelining overhead associated with running the 3x3 filter, i.e., valid output data are always obtained in response to every PROC command.
- Lsr Reflectivity speckle remover. When set, range speckles in the corrected and uncorrected reflectivity data are removed.
- Dsr Doppler speckle remover. When set, range speckles in the velocity and width data are removed.
- Rnv Range normalization of reflectivity data. This bit also enables intervening gas attenuation correction.



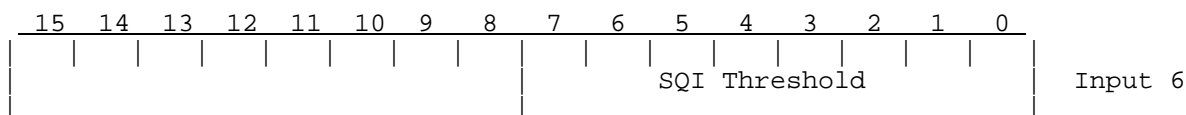
This number defines the multiplicative constant that converts the signal power in dB to the units of the 12-bit “Log of power in sample” time series outputs. One fourth of this slope is used to generate the “Log of Measured Noise Level” output from GPARM (word 6). The recommended value to use here is 0.03 (1966). This gives a dynamic range of 122 dB in 12 bits.



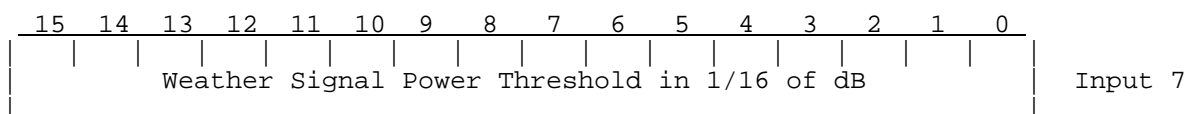
Reflectivity values below this level can result in thresholding of data, if the threshold control flags (see below) include LOG Noise bits. The threshold value is always non-negative, and the comparison test is described in Section 5.3.



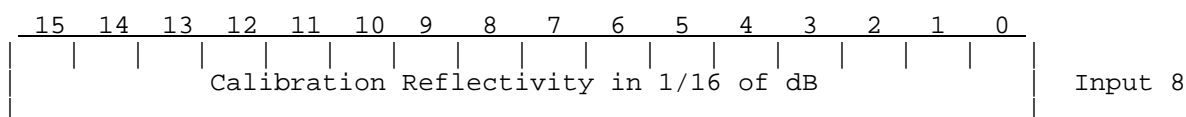
The clutter correction threshold is a bound on the computed log receiver adjustment for clutter. These corrections (in dB) are always negative. Any clutter correction which is more negative than the above value can result in thresholding of data.



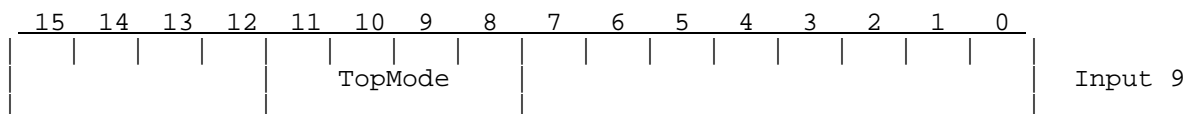
The Signal Quality Index (SQI) threshold is an unsigned binary fraction in the range 0 to 255/256. When the SQI for a range bin falls below the stated value it may result in thresholding of data.



Weather Signal Power (SIG) is an estimate of the SNR of the weather component of the received signal. When the SIG (see Section 5.2.12) falls below this comparison value it may result in thresholding of data.



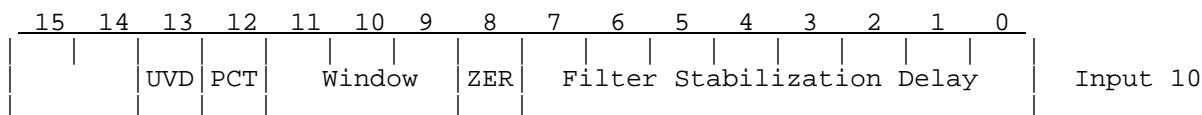
The calibration reflectivity is referenced to 1.0 kilometers.



The TopMode bits select the overall data acquisition and processing mode for the RVP8.

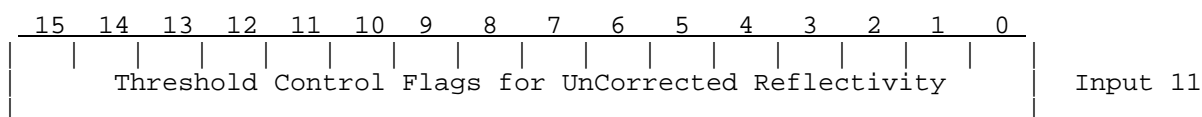
Although the processing algorithms that are used in each top level mode are quite different, the RVP8 command set works in a uniform way in all modes.

- 0000      Pulse Pair Processing Mode. Doppler clutter filters are 4th-order IIR high pass; data are processed one pulse at a time as each pulse arrives (see Section 5.2.3).
- 0001      FFT Processing Mode. Doppler clutter filters use nonlinear frequency-domain approach; data are processed in batches of pulses (see Section 5.2.2).
- 0010      Random Phase Processing Mode. Data from first and second trips are dealiased in range based on knowledge of the radar transmitter phase (see Section 5.7).
- 0100      DPRT-1 Processing Mode. The trigger generator produces alternate short and long pulses, and Doppler autocorrelations are computed using only the short pairs (see Section 5.5).
- 0101      DPRT-2 Processing Mode. The trigger generator produces alternate short and long pulses, and Doppler autocorrelations are computed using both pairs (see Section 5.5).
- 11XX      Four codes reserved for custom user modes.

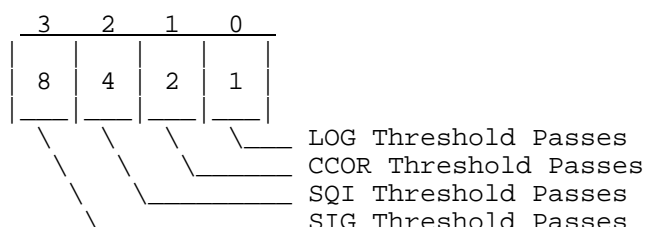


The RVP8 clutter filters are controlled by this word.

- Delay      This delay is introduced prior to processing the next ray of data whenever Dual-PRF velocity unfolding is enabled or the RVP8 has been reconfigured by user commands. The delay permits the clutter filter transients to settle down following PRF and gain switches. The value is specified as the number of pulses, and hence, the number of filter iterations, to wait.
- ZER      If set, then the clutter filter's internal state variables are zeroed prior to waiting the delay time. For some signal conditions, this may give better results than allowing the filter to naturally flow into the new data.
- Window    Selects the type of window that is applied to time series data prior to computing power spectra via a DFT. Choices are: 0:Rectangular, 1:Hamming, 2:Blackman, 3:Exact Blackman, 4:VonHann.
- PCT      If set, the RVP8 will attempt to run its standard processing algorithms even when a custom trigger pattern has been selected via the SETPWF command.
- UVD      Unfold velocities using a simple ( $V_{high} - V_{low}$ ) algorithm, rather than the standard algorithm described in Section 5.6.



These flags select which threshold comparisons result in unCorrected reflectivity being accepted or rejected at each bin. There are four test comparisons that are made at each range, as described above for input words 4, 5, 6, and 7. Each test either passes and produces a code of 1, 2, 4, and 8 respectively, or fails and produces a code of zero. The sum of the codes for each of the four tests is a number between 0 and 15, which can also be interpreted as the following four-bit binary number:



The individual bits of the Threshold Control Flag word each specify whether data are to be accepted (1) or rejected (0) in each of the sixteen possible combinations of threshold outcomes. Thus, the pattern of bits in the flag word actually represents a truth table for a given logical function of the four threshold outcomes.

The following examples show actual values of the Flag word for the stated combinations of acceptance criteria:

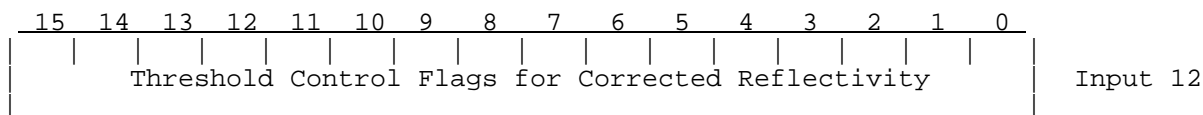
Value	Criteria
FFFF	All Pass (Thresholds disabled)
0000	All Fail (No data are passed)
AAAA	LOG
8888	LOG and CSR
A0A0	LOG and SQI
8080	LOG and CSR and SQI
F0F0	SQI
FAFA	SQI or LOG
C0C0	SQI and CSR
F000	SQI and SIG
C000	SQI and SIG and CSR
FFF0	SQI or SIG
CCC0	(SQI or SIG) and CSR

A simple way to generate these values is to imagine four 16-bit quantities having the following names and values: LOG=AAAA, CSR=CCCC, SQI=F0F0, SIG=FF00. The flag value needed to represent a given logical combination of threshold outcomes is obtained as the result when that same logical combination is applied to these special numbers.

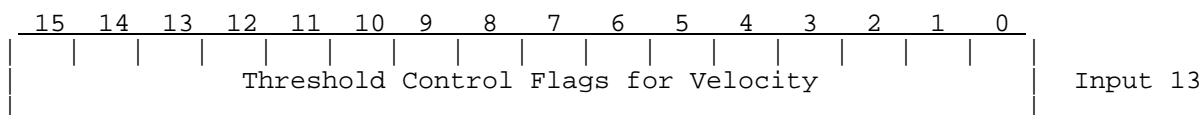
For example:

(SQI or SIG) and CSR = (F0F0 or FF00 ) and CCCC  
= (FFF0) and CCCC  
= CCC0

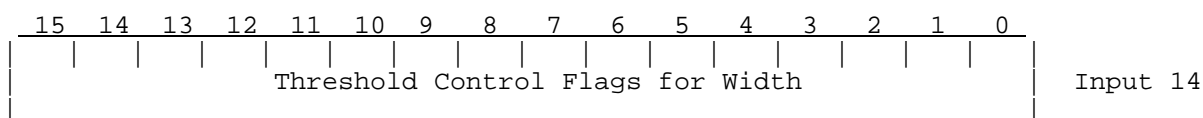
which corresponds with one of the examples given above.



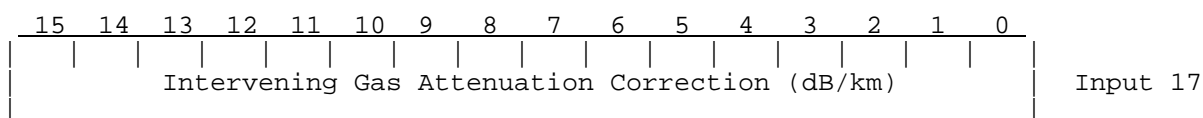
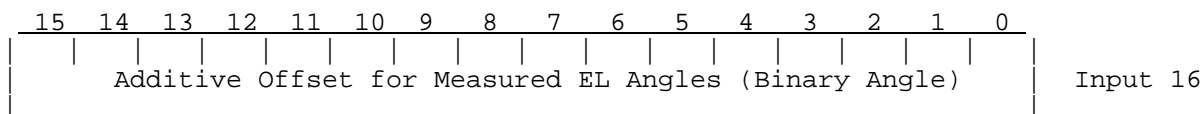
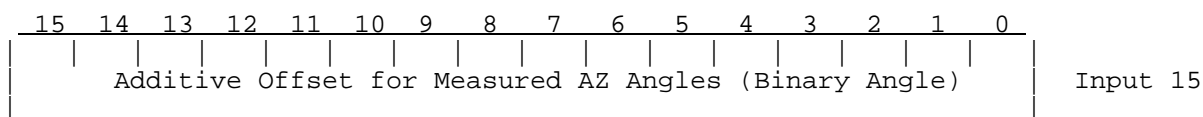
See Description for Input #11.



See Description for Input #11.



See Description for Input #11.

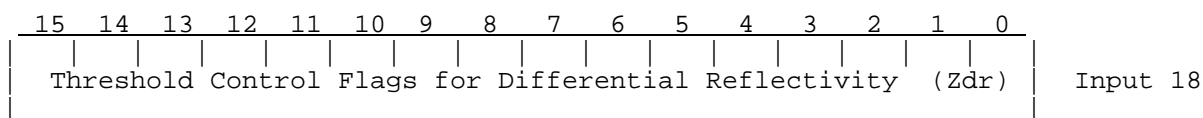


Gas attenuation correction attempts to compensate for overall (two-way) beam losses due to absorption by atmospheric gasses. The correction is linear with range, and is added to the data along with range normalization. Therefore, clearing the RNV bit in Word #2 above disables the correction. Of course, gas attenuation compensation can still be turned off even when RNV is on, simply by setting a slope of 0.0 dB/km.

An attenuation of G db/km is encoded into the unsigned 16-bit word N as follows:

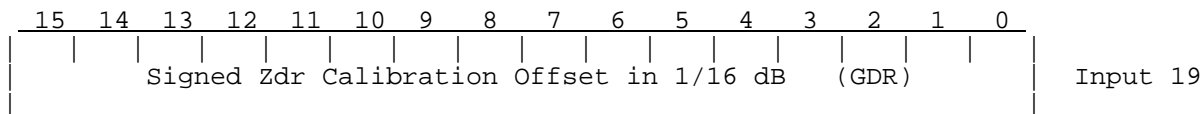
$$\begin{aligned} 0 \leq N \leq 10000 & \quad G = N / 100000 \\ \text{else} & \quad G = 0.1 + (N - 10000)/10000 \end{aligned}$$

This format is backward compatible with the previous linear format for all values between 0.0 and 0.1dB/km; but it extends the upper range of values from 0.65535 up to 5.6535. These larger attenuation corrections are needed for very short wavelength radars.

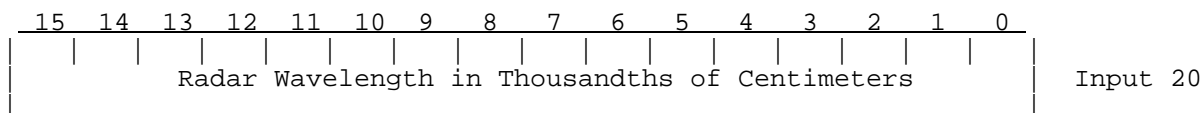




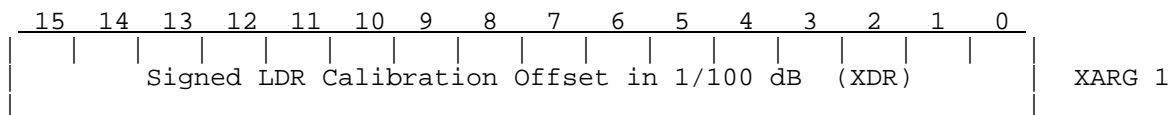
See Description for Input #11.



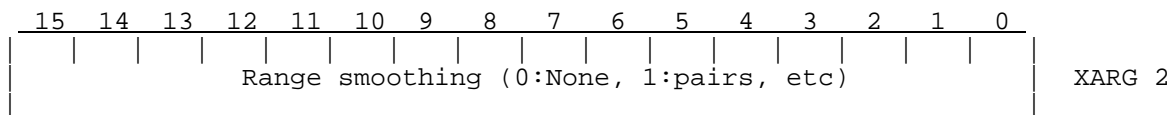
When differential reflectivity is computed there is a possibility that radar asymmetries will introduce a bias in the Zdr values, i.e., that Zdr will be non-zero even when observing purely spherical targets. This calibration offset permits nulling out this effect. The GDR offset accounts for the overall Tx/Rx gain imbalance between the two channels of the radar.



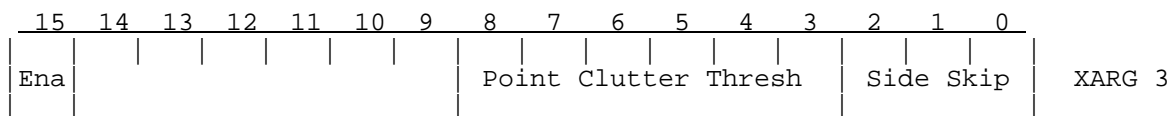
The radar wavelength is used in the calculation of 16-bit velocity and width data, to convert from Nyquist units to absolute physical units.



The XDR offset is used in the Linear Depolarization Ratio equations, and is the differential receiver gain between the two channels. Note that unlike the GDR offset (used for ZDR), the gain difference does not depend on differential transmit power.

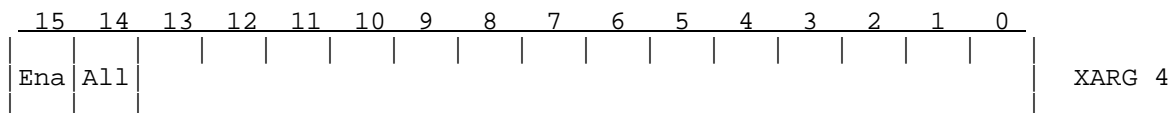


Range smoothing can be performed on raw moment data prior to the computation of scientific parameters. The number of bins to sum together is given here. This should generally be an odd integer so that no range bias is introduced by the smoothing operation.



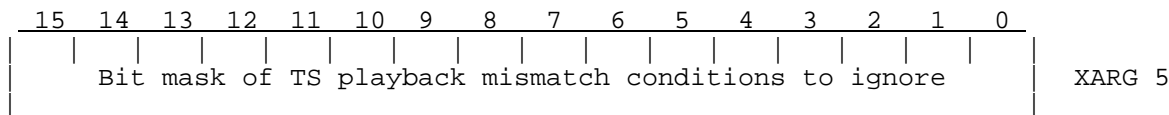
Point clutter detection is configured with this word. A bin will be flagged as containing clutter if its power exceeds that of its two neighboring bins by more than the detection threshold (in deciBels). Up to seven bins may optionally be skipped on each side of the central bin prior to making these two comparisons.

**Ena** This bit is set to enable point clutter detection. Flag bits will then be reported in the "Flg" output data type of the PROC command.

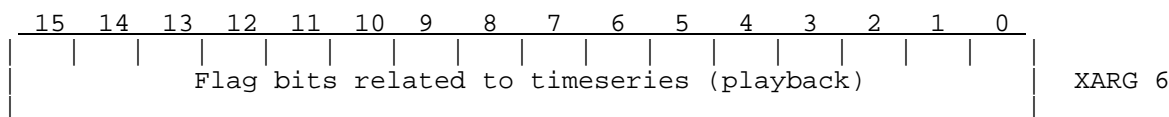


Point clutter censoring is configured with this word.

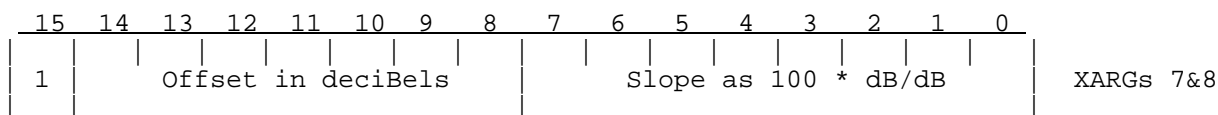
- Ena** This is set to enable point clutter censoring. Raw moment data containing point clutter will be interpolated from valid signal levels on either side.
- All** Optionally expand the reported detection flags to show the entire replaced interval, not just the original detected bins. This gives a more honest view of the data bins that have been altered.



This word is a combination of MMTS\_XXX bits specifying what types of mismatches are okay (do not cause an all-zero ray to be produced) during PROC command processing of timeseries data that are played back from an external source into the RVP8.



Combination of OPTS\_XXX bits which modify details of timeseries behavior.



These two words allow you to set the breakpoints and slopes that modify the LOG threshold according to the Clutter-to-Noise ratio of the target. This makes the LOG threshold behave properly even as the noise floor becomes elevated due to very strong clutter targets. A value of zero will restore the RVP8 defaults from the **Mf** menu.

The default (power-up) values for the above parameters are listed below. Both the scientific units and the integer-input required by the command to set up that value are given. Most of these defaults will likely be reasonable for a wide variety of radars.

**Table 6–1: Default Values For Operating Parameters**

Parameter	Scientific Units	Input
Sample Size	25 pulses	25
Flag Word		0007 Hex
Log Slope	0.03 dB/LSB	1966
LOG Threshold	0.5 dB	8
CCOR Threshold	–25.0 dB	–400
Signal Quality Index Threshold	0.5 (dimensionless)	128
SIG Threshold	10.0 dB	160
Calibration Reflectivity	–22.0 dBZ	–352
Gas Attenuation	0.016 dB/km	1600
Zdr Offset (GDR)	0.0 dB	0

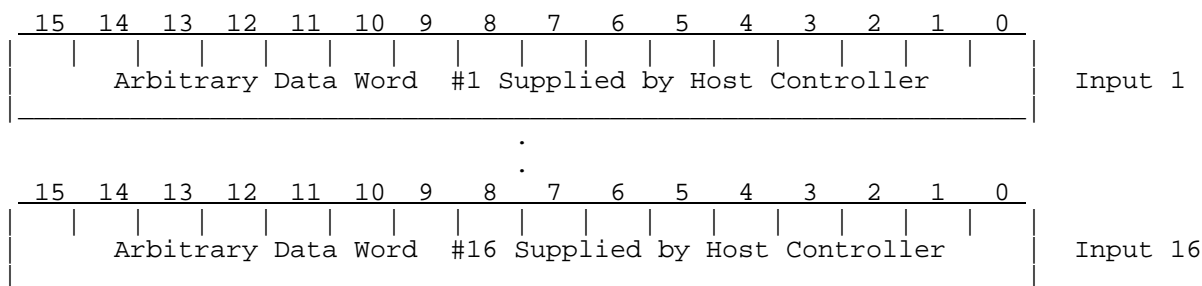
**Table 6–1: Default Values For Operating Parameters (cont.)**

Parameter	Scientific Units	Input
LDR Offset (XDR)	0.0 dB	0
AGC Integration Period	8 pulses	8
Radar Wavelength	5.3 cm.	5300
Dual PRF Filter Stabilization	10 pulses	10
UnCor Refl. Thresh. Control Flag	LOG	AAAA Hex
Cor Refl. Thresh. Control Flag	LOG & CSR	8888 Hex
Velocity Thresh. Control Flag	SQI & CSR	C0C0 Hex
Width Thresh. Control Flag	SQI & CSR & SIG	C000 Hex
Zdr Refl. Thresh. Control Flag	LOG	AAAA Hex
AZ/EL Angle Offsets	0 degrees	0000 Hex

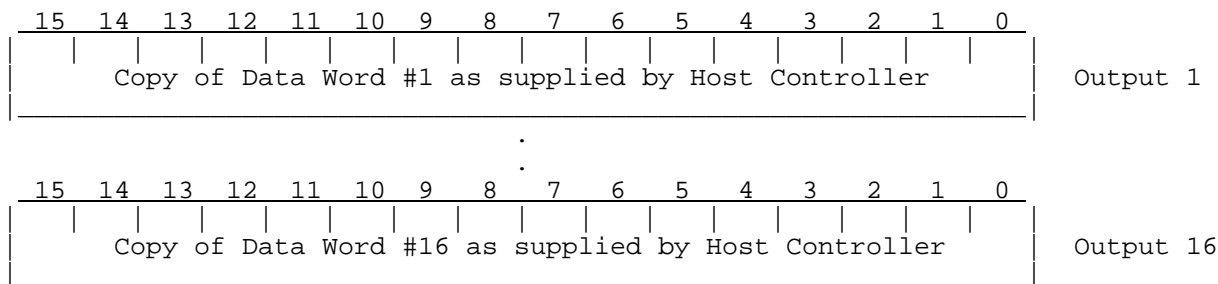
## 6.4 Interface Input/Output Test (IOTEST)

This command is used to test both the input and output data busses of the signal processor interface. When issued, the command causes sixteen words to be read from the host controller, after which those same sixteen words are written back out. Typically, the controller supplies a “barber pole” input sequence consisting, for example, of successive powers of two. If all of the output words are correct, one may conclude that there are no malfunctioning bits in the interface hardware.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
											0	0	0	1	1	Command

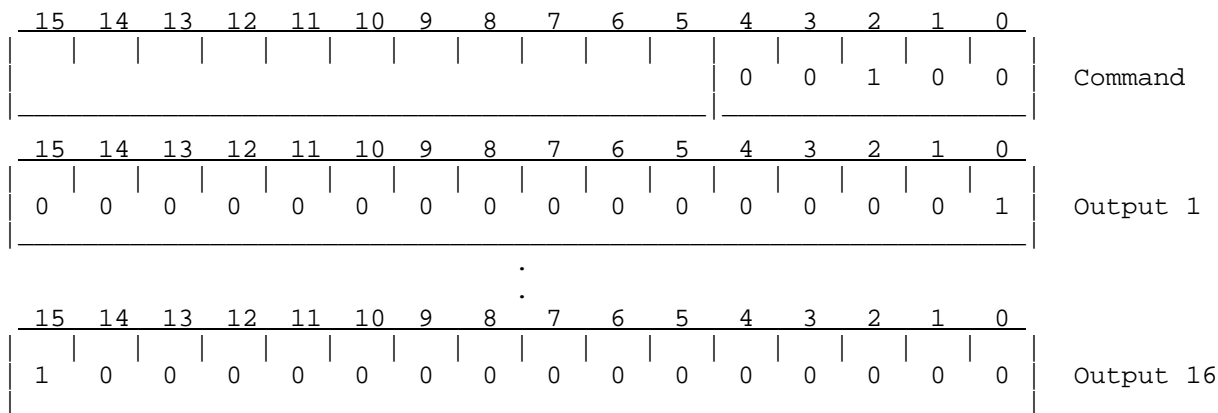


**Note: The IOTEST command can also process and echo up to 128 additional XARGS data words (See Section 6.20).**



## 6.5 Interface Output Test (OTEST)

This command is used to test the integrity of the data being output by the signal processor. The command causes sixteen words to be output consisting of successive powers of two starting from one. By verifying whether each output word is correct, malfunctioning bits in the interface data bus can easily be isolated. This test is less stringent than the input/output test IOTEST, since the input data paths to the processor are not being checked. Typically, the OTEST is performed only when the IOTEST fails, and then to determine whether the fault was on input or output.



## 6.6 Sample Noise Level (SNOISE)

This command is used to estimate the current noise level from the receiver, so that the noise can be subtracted from subsequent measurements. Data are sampled for 256 pulses at 256 bins, beginning at a selectable range and spaced by the range resolution at that pulse width. The

internal trigger generator is temporarily set to a special noise rate (usually much lower than the operating rate) during the process. It is ultimately the user's responsibility to insure that no returned power is present within the approximately 32km sampling interval. In some cases it may be necessary to raise the antenna during the noise measurement to avoid thermal noise pickup from the ground, or from weather targets.

SNOISE has the option of setting up a new sampling range and trigger generator rate each time it is called. Two bits in the command word determine which (if any) of the new values overrides the current values stored in the RVP8. The power-up sampling range is 250km (input value of 250), and the power-up trigger rate is 200Hz (input value of 30000). These initial values persist until such time as they are altered here. Note that both input words must always be supplied after the command, even if the command calls for ignoring one or both of them. The range is supplied directly in kilometers up to a maximum of 992km. The trigger rate resulting from a given input is 6MHz divided by the input value, i.e. the input value is the trigger period in 0.1667 microsecond increments. Keep in mind that the given rate is bounded against the minimum PRT allowed for the current radar pulse width.

The SNOISE command bounds the requested starting range of the noise sampling interval. This is to insure that the noise samples will fit within the specified PRT, and within the range mask hardware RAM. The RVP8 sets an error bit when an improper range is requested.

The SNOISE command should be re-issued now and then to compensate for drift in the RF and A/D systems. However, because DC offsets do not propagate into the "I" and "Q" values, reissuing the command is much less critical than with the RVP6. The noise levels must be measured for the RVP8 to properly process data. This can be done by issuing the SNOISE at least once after power-up, or by setting the correct values for the powerup noise levels with the "mt" setup command, see section 3.2.5. The RVP8 does not automatically take a noise sample as part of its initialization procedure.

The measured offsets are stored internally for all subsequent uses inside the RVP8. The offset values may be inspected via the GPARM command, as may the current range and rate values themselves. Of course, whenever the range or rate are changed the user must ensure that the new trigger rate allows at least 32km following the new noise range. If this requirement is not met, or if other failures are detected during the noise measurement, appropriate bits are set in the GPARM latched status word. This word should generally be checked after SNOISE to make sure that everything worked properly.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
				Action		Rat	Rng				0	0	1	0	1	Command

- Rng      If 1, then the range in input word 1 is taken as the starting noise range for this and all subsequent SNOISE calls.
- Rat      If 1, then the trigger rate in input word 2 is taken as the noise rate for this and all subsequent SNOISE calls.
- Action   Specifies what action is carried out by the command.
- 0        Compute a new noise sample based on the present IFD input signals.
- 1        Do not compute a noise sample, but rather, read new noise values from the host computer and use them for subsequent processing. Four additional optional input

words supply the noise information, and GPARM words 6, 9, and 44-50 will be changed to reflect the new noise settings.

2 Do not compute a noise sample, but rather, restore the powerup noise defaults.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Starting Range in km (Max 992km) of 32km Sampling Interval																Input 1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Internal Trigger Rate (6Mhz/N) to use During Noise Sampling																Input 2

The following input words are optional, only if Action=1.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0	0	(MSB)	Log of Measured Noise Level											(LSB)		Input 3

The is the same number as GPARM output 6. See the discussion in Sections 6.7 and 6.9.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Noise Level Standard Deviation (in 1/100 of a dB)																Input 4

This is the same number as the GPARM output 49.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Ratio of Horizontal/Vertical Noise Power in Hundredths of dB																Input 5

This is the same number as the GPARM output 50.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
<Spare>													Err	Ttf	Ntg	Input 6

These fault bits will then be output in the latched status GPARM word 9 .

- Ntg No Trigger during noise measurement.
- Ttf Trigger too fast during noise measurement, i.e., some of the noise sample bins were positioned past the trigger range.
- Err Error detected during the SNOISE command.

## 6.7 Initiate Processing (PROC)

The PROC command controls the actual processing and output of radar data. The operating modes and types of data available from the RVP8 are described in detail in Chapter 1. That section also describes the proper use and application of the RVP8 to different radar environments.

PROC is a single-word command that specifies the type of processing to be performed, and the type of output to be generated. The two mode bits in the command word select either

- Synchronous mode — The processor acquires, processes, and outputs one ray in response to each PROC command. Processing is begun only after each command is actually received.
- Free running mode — A single PROC command is issued and rays are continually output as fast as they can be produced and consumed. This continues until any other command is written, e.g., a NOP could be used to terminate the free running mode with no other consequences.
- Time Series mode — Always produced in a synchronous manner, this mode requires a new PROC command to initiate each new set of samples. Data are output as 8-bit time series, 16-bit time series, or 16-bit power spectra.

Optional Dual-PRF velocity unfolding is chosen by command bits eight and nine. For Doppler data either a 2:3, 3:4, or 4:5 PRF unfolding ratio may be selected. The RVP8 carries out all of the unfolding steps internally, so that mean velocity is now output with respect to the larger unambiguous interval. There is no additional velocity processing needed by the user, except of course, to change the velocity scale on any displays being generated. Furthermore, spectral widths are scaled consistently with respect to the higher PRF, and require no user modification before being plotted.

When unfolding is selected, the internal trigger generator automatically switches rates on alternate rays. The switch over occurs immediately after the last pulse of the current ray has been acquired; thus overlapping the internal post-processing and output time, with transmitter stabilization and data acquisition at the new rate.

Output data are selected by the upper six bits of the PROC command. Packed archive output is selected by setting the ARC bit. Individual byte or word display output is selected by setting any or all of the Z, T, V, W, Zdr, and KDP command word bits. When more than one of these bits is set, the output array consists of all of the bins for the leftmost selected parameter, followed by all of the bins for the next selected parameter, etc. Bits selected in XARG #1 behave the same way, except that the output order is right-to-left. Both archive and display formats can be selected simultaneously, in which case the archive format is output first, followed by whichever individual display format values were also selected. The archive format is not recommended for use with new drivers because it can only handle four of the many possible output parameter types.

When time series mode is selected there are three output data formats available. For backwards compatibility, there is an 8-bit integer format in which the eight most significant bits from the I, Q, and LOG signals are represented in a byte. This format is not recommended because it will generally miss weak signals. We recommend the floating-point format that uses 16-bits per A/D sample. There is also a 16-bit power spectrum output that is accurate to 0.01dB. (See also GPARM output word #10).

In addition to the above output data, the first words of each ray optionally contain additional information about the ray itself. These header words are configured by the CFGHDR opcode, and are included only if the NHD (No-Headers) bit in SOPRM Input #2 is clear. For example, if TAG angle headers are requested, if the ARC, Z and V bits are all set, and if there are 100 bins selected in the current range mask, then each RVP8 output ray consists of the following:

- 1] TAG15 - TAG0 \ From Start of Acquisition
- 2] TAG31 - TAG16 / Interval
- 3] TAG15 - TAG0 \ From End of Acquisition
- 4] TAG31 - TAG16 / Interval
- \* 200 words of packed archive data,
- \* 100 words of Corrected Reflectivity data in low byte only.
- \* 100 words of Velocity data in low byte only,

The Command word format for Synchronous Doppler Mode is:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ARC	Z	T	V	W	ZDR	Unfold		KDP	0	1	0	0	1	1	0	Command

The Command word format for Free Running Doppler Mode is:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ARC	Z	T	V	W	ZDR	Unfold		KDP	1	0	0	0	1	1	0	Command

Either of these may be augmented by an optional XARG word (See Section 6.20)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
						Flg	(Tx Vert) Phi Rho Ldr			(Tx Horz) Phi Rho Ldr			SQI	RHV	PDP	XARG 1

Unfold Selects Dual-PRF unfolding scheme:

- 00 : No Unfolding
- 01 : Ratio of 2:3
- 10 : Ratio of 3:4
- 11 : Ratio of 4:5

ARC Selects archive output format in which four data bytes (see 8-Bit descriptions below) are packed into two output words per bin as follows:

High Byte		Low Byte		
V		Z		First Word
W		T		Second Word

The remaining data parameters are available in both 8-Bit and 16-bit formats, according to SOPRM Command input word #2 (See Section 6.3). The same SOPRM word configures the RVP8 for Single or Dual polarization. The later is required for KDP, PDP, and RHV to be computed properly.

V Selects radial velocity data.

**8-Bit Velocity Format** — Mean velocity, expressed as a fraction of the unambiguous velocity interval, is computed from the unsigned byte N as:

$$V_{m/sec} = V_{Nyquist} \times (N-128) / 127.5$$

- 0 : Indicates velocity data is not available at this range
- 1 : Maximum velocity towards the radar



128 : Zero velocity

255 : Maximum velocity away from the radar

When velocity unfolding is selected, the output is still interpreted as above, except that the unambiguous interval is increased by factors of 2, 3, and 4 for 2:3, 3:4, and 4:5 unfolding.

**16-Bit Velocity Format** — Mean velocity in meters/second is computed from the unsigned word N as:

$$V_{\text{m/sec}} = (N - 32768) / 100$$

The overall range is from -327.67m/sec to +327.66m/sec in one centimeter/second steps as follows:

0 : Indicates velocity data is not available at this range

1 : -327.67 m/sec (towards the radar)

32768 : 0.00 m/sec

65534 : +327.66 m/sec (away from the radar)

65535 : Reserved Code

W Selects spectral width data.

**8-Bit Width Format** — Spectral width is computed from the unsigned byte N as:

$$W_{\text{Nyquist}} = N / 256$$

The overall range is a fraction between 1/256 to 255/256 of the unambiguous interval. The code of zero indicates that width data was not available at this range.

**16-Bit Width Format** — Spectral width in meters/second is computed from the unsigned word N as:

$$W_{\text{m/sec}} = N / 100$$

The overall range is from 0.01m/sec to 655.34m/sec in one centimeter/second steps as follows:

0 : Indicates width data is not available at this range

1 : 0.01 m/sec

65534 : 655.34 m/sec

65535 : Reserved Code

Z Selects clutter corrected reflectivity data.

**8-Bit deciBel Format** — The level in decibels is computed from the unsigned byte N as:

$$\text{dBZ} = (N - 64) / 2.$$

The overall range is therefore from -31.5 dBZ to +95.5 dBZ in half-dB steps as follows:

0 : Indicates no reflectivity data available at this range

1 : -31.5 dBZ

64 : 0.0 dBZ

128 : 32.0 dBZ

255 : +95.5 dBZ

**16-Bit deciBel Format** — The level in decibels is computed from the unsigned word N as:

$$\text{dBZ} = (N - 32768) / 100$$

The overall range is from -327.67dB to +327.66dB in 1/100dB steps as follows:

0 : Indicates no reflectivity data available at this range  
1 : -327.67 dBZ  
32768 : 0.00 dBZ  
65534 : +327.66 dBZ  
65535 : Reserved Code

T Selects total reflectivity. Same 8-bit and 16-bit coding formats as for clutter corrected reflectivity above.

ZDR Selects differential reflectivity data.

**8-Bit ZDR Format** — The level in decibels is computed from the unsigned byte N as:

$$\text{dB} = (N - 128) / 16$$

The overall range is from -7.935dB to +7.935dB in one-sixteenth dB steps as follows:

0 : Indicates no reflectivity data available at this range  
1 : -7.9375 dB  
128 : 0.0000 dB  
255 : +7.9375 dB

**16-Bit ZDR Format** — Same as 16-bit deciBel format.

KDP Selects dual polarization specific differential phase data.

**8-Bit KDP Format** — Values are coded into an unsigned byte using a logarithmic scale. The KDP angles are multiplied by the wavelength in cm. (to reduce dynamic range) and then converted to a log scale separately for both signs. The minimum value is 0.25 deg\*cm/km, and the maximum value is 150.0 deg\*cm/km. A code of zero represents no data, and a code of 128 represents 0 deg\*cm/km. The conversion equation for positive values (codes from 129 to 255) is:

$$KDP \times \lambda = 0.25 \times 600 \left[ \frac{N - 129}{126} \right]$$

The conversion equation for negative values (codes from 1 to 127) is:

$$KDP \times \lambda = -0.25 \times 600 \left[ \frac{127 - N}{126} \right]$$

**16-Bit KDP Format** — Same as 16-bit deciBel format, except that the units are hundredths of degrees per kilometer. No weighting by wavelength is introduced.

PDP Selects dual polarization differential phase  $\Phi_{DP}$  data.

**8-Bit  $\Phi_{DP}$  Format** — The phase angle in degrees is computed on a 180-degree interval from the unsigned byte N as:

$$\Phi_{DP} (mod\ 180) = 180 (N - 1) / 254$$

- 0 : Indicates no  $\Phi_{DP}$  data available at this range
- 1 : 0.00 deg
- 254 : 179.29 deg
- 255 : Reserved Code

**16-Bit  $\Phi_{DP}$  Format** — The phase angle in degrees is computed on a 360-degree interval from the unsigned word N as:

$$\Phi_{DP} (mod\ 360) = 360 (N - 1) / 65534$$

- 0 : Indicates no  $\Phi_{DP}$  data available at this range
- 1 : 0.000 deg
- 65534 : 359.995 deg
- 65535 : Reserved Code

RHV Selects dual polarization correlation coefficient  $\rho_{HV}$  data.

**8-Bit  $\rho_{HV}$  Format** — The correlation coefficient is computed on the interval 0.0 to 1.0 using a square root weighting of the unsigned byte N as:

$$\rho_{HV} = \sqrt{\frac{(N - 1)}{253}}$$

- 0 : Indicates no  $\rho_{HV}$  data available at this range
- 1 : 0.0000 (dimensionless)
- 2 : 0.0629
- 253 : 0.9980
- 254 : 1.0000
- 255 : Reserved Code

**16-Bit  $\rho_{HV}$  Format** — The correlation coefficient is computed on the interval 0.0 to 1.0 linearly from the unsigned word N as:

$$\rho_{HV} = (N - 1) / 65533$$

- 0 : Indicates no  $\rho_{HV}$  data available at this range
- 1 : 0.0 (dimensionless)
- 65534 : 1.0
- 65535 : Reserved Code

SQI Selects Signal Quality Index data. This dimensionless parameter uses the same 8-bit and 16-bit data formats as RHV ( $\rho_{HV}$ ).

LDR Selects Linear Depolarization Ratio, measured either on the horizontal receive channel while transmitting vertically, or on the vertical receive channel while transmitting horizontally.

**8-Bit LDR Format** — The level in decibels is computed from the unsigned byte N as:

$$dB = -45.0 + (N - 1) / 5$$

This spans an asymmetric interval around zero decibels, and allows for cross channel isolation as large as 45dB. The overall range is from  $-45.0\text{dB}$  to  $+5.6\text{dB}$  in 0.2dB steps as follows:

- 0 : Indicates no LDR data available at this range
- 1 :  $-45.0\text{ dB}$
- 226 :  $0.0\text{ dB}$
- 254 :  $+5.6\text{ dB}$
- 255 : Reserved Code

**16-Bit LDR Format** — Same as 16-bit deciBel format.

- RHO** Selects the cross channel correlation coefficient. This dimensionless parameter uses the same 8-bit and 16-bit data formats as RHV ( $\rho_{HV}$ ).
- PHI** Selects the cross channel differential phase. This parameter uses the same 8-bit and 16-bit angular data formats as PDP ( $\Phi_{DP}$ ).
- Flg** Selects flag word output, bits defined as follows:
- 0 Reflectivity obscured at this bin
  - 1 Velocity obscured at this bin
  - 2 Width obscured at this bin
  - 3 Point clutter detected at this bin

The Command word format for Time Series Mode is:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
TSOUT		Spec Type				Unfold			1	1	0	0	1	1	0	Command

**TSOUT** Selects type of data to be output.

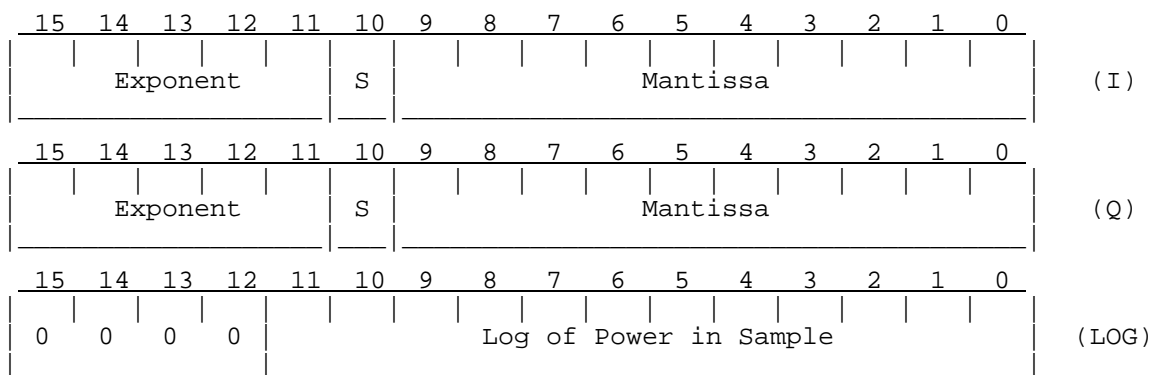
- 00 : 8-bit Time Series
- 01 : Power Spectrum
- 10 : 16-bit Time Series
- 11 : Unused

When the TSOUT bits select “Power Spectrum” then, depending on the current major mode, a further choice may be needed to select one of several spectral view points. For the Random Phase major mode the possible values of “Spec Type” are:

- 0: Raw First Trip
- 1: Whitened First Trip
- 2: Cleaned First Trip
- 3: Final First Trip
- 4: Raw Second Trip
- 5: Whitened Second Trip
- 6: Cleaned Second Trip
- 7: Final Second Trip

When time series output is selected the output data consist either of  $(3 \times B \times N)$  or  $(2 \times B \times N)$  words, depending on the output format, where B is the number of bins in the current range mask, and N is the number of pulses per ray. Data samples for each bin of pulse #1 are output first, followed by those for each bin of pulse #2, etc. up to pulse #N. In other words, the data are output in the same time-order that they were acquired.

In the floating point format, three words are used for each bin:

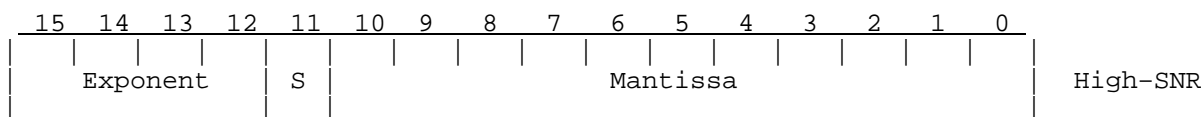


To convert these “legacy format” floating I and Q samples to voltages: First create a 12-bit signed integer in which bits zero through nine are copied from the Mantissa field, and bits ten and eleven are either 01 or 10 depending on whether S is 0 or 1. Then, multiply this number by  $2^{*(\text{exponent}-40)}$ , where the exponent field is interpreted as an unsigned 5-bit integer. Finally multiply by the maximum voltage. The resulting value has 12-bits of precision and a dynamic range of approximately 190dB. The large dynamic range is necessary to cover the full range of data. In summary:

$$\text{Voltage} = V_{MAX} \times (\text{Sign}, \text{Mantissa}) \times 2^{[\text{Exponent} - 40]}$$

Note that the resulting voltage span is actually  $\pm 4 \times V_{MAX}$ . The extra factor of four is built into the format so that transient excursions above the full scale input voltage can still be encoded properly. These may arise for time series data that have been processed by an IIR clutter filter.

An improved “High-SNR” packed floating format is also available that offers nearly the same dynamic range but provides a 6dB improvement in SNR, i.e., a commensurate improvement in sub-clutter visibility of  $-78\text{dB}$  versus  $-72\text{dB}$ .



The High-SNR packed format is similar to the legacy packed format except that it uses one extra mantissa bit and one fewer exponent bit. The dynamic range lost in the exponent is recovered through a formatting trick known as “soft underflow”, i.e., the mantissa is allowed to become unnormalized when the exponent is zero.

To decode this format when the exponent is non-zero, first create a 13-bit signed integer in which bits zero through ten are copied from the Mantissa field, and bits eleven and twelve are either 01 or 10 depending on whether S is 0 or 1. Then, multiply this by  $2^{*(\text{exponent}-25)}$ , where the exponent field is interpreted as an unsigned 4-bit integer.

To decode the High-SNR format when the exponent is zero simply interpret the mantissa as a 12-bit signed integer and multiply by  $2^{*-24}$ .

A complete analysis of the noise properties of the floating point codes would be fairly tricky. For the High-SNR format, the 12-bit mantissa with hidden normalization bit will vary from 2048 to 4095. The SNR will therefore vary from 66dB to 72dB and we can assign a mean value of 69dB. Another 9dB of useful range is contained within the code as follows:

- In a floating point encoding format, the notion of fixed additive quantization noise is not really correct. For a signal having a given power, the additive noise within each instantaneous sample will scale down according to the magnitude of that sample. The ensemble of noise terms thus contributes an RMS power that is smaller than the Peak-to-Noise ratio would imply. In the case of a sinusoidal input, this gives a 3dB boost in effective SNR.
- The format, of course, also represents negative amplitudes with the same relative precision as positive values. In a fixed-point format this would add 6dB (one more bit) to the overall dynamic range and large-signal SNR. In the floating format we really only gain 3dB (half a bit) because the RMS noises add independently on the positive and negative excursions.
- The packed format is used to encode timeseries (I,Q) pairs, and it's the SNR properties of these pairs that we're really concerned about. To a first approximation, having a pair of values roughly doubles the information content and adds another 3dB to the SNR.

The last of the three timeseries output words, the "Log of Power in Sample", is provided mainly for backwards compatibility. It can be calculated from the I and Q numbers. To convert to dBm it requires a slope and offset as follows:

$$dBm = P_{MAX} + Slope \times [Value - 3584]$$

Where:

$P_{MAX}$  = +4.5dBm for 12-bit IFD, +6.0dBm for 14-bit IFD

$V_{MAX}$  = 0.5309 Volts for 12-bit IFD, 0.6310 Volts for 14-bit IFD

$Slope$  = "Log Power Slope" word 3 of SOPRM command. 0.03 recommended.

For backwards compatibility the RVP8 produces a 8-bit fixed point time series format. Because of the limited dynamic range available, this will only show strong signals, and is not recommended for use. The I, Q, and Log power triplets are packed into two 16-bit output words as follows:

High Byte	Low Byte	
Q Sample	I Sample	First Word
Zero	Log Power	Second Word

The "Log Power" value is the upper 8 bits of the long format. The other numbers are produced by the equation:

$$Voltage = V_{MAX} \times \left[ \frac{Sample}{128} \right]$$

When Power Spectrum output is selected, the spectrum size is chosen as the largest power of two (N2) that is less than or equal to the current sample size (N). When the sample size is not a power of two, a smaller spectrum is computed that by averaging the spectra from the first N2

and the last  $N_2$  points. The data format is one word/bin/pulse, in the same order as for time series output. Each word gives the spectral power in hundredths of dB, with zero representing the level that would result from the strongest possible input signal ( $P_{MAX}$ ). Thus, the spectral output terms are almost always negative.

The time series that are output by the RVP8 are the filtered versions of the raw data, when available. If a non-zero time-domain clutter filter is selected at a bin, then the I and Q data for that bin show the effects of the filter. Whenever you need to observe the raw samples, make sure that no clutter filters are being applied.

In pulse pair time series mode with dual receivers, selecting (H+V) will produce data in one of two formats according to the “Sum H+V Time Series” question in the **Mp** setup section:

- Answering “Yes” will result in summed time series from both channels, but spectra from the DSP will be the averaged spectra from each channel individually. This allows the **IRIS ascope** utility to display either the spectrum-of-sum or sum-of-spectra according to whether the “Spectra from DSP” button is pressed in the *Processing/Gen-Setup* window.
- Answering “No” will still produce the usual (BxN) time series output samples, except that the first half of these samples will be the first half of the “H” data in their normal order. This will be followed by a zero sample if (BxN) is odd; followed by the first half of the “V” data, also in their normal order.
- In other words, only the first halves of the individual “H” and “V” sample arrays are output by the RVP8. As an example, if you select 25 bins and 100 pulses, then the output data will consist of 1250 “H” samples (from all bins in the first 50 pulses), followed by 1250 “V” samples from the exact same set of bins and pulses. This is the more useful option when custom algorithms are being run on the data from the two separate receivers.

When the number of output words is large there is a possibility that the internal buffering within the RVP8 may overflow and data may be lost. Due to internal memory limitations, the product (BxN) must be less than 12000. A bit in the latched status word (See GPARM) indicates when time series overflows occur. In such cases, the correct number of words are still output, but they are all zero after the point at which overflow was detected.

## 6.8 Load Clutter Filter Flags (LFILT)

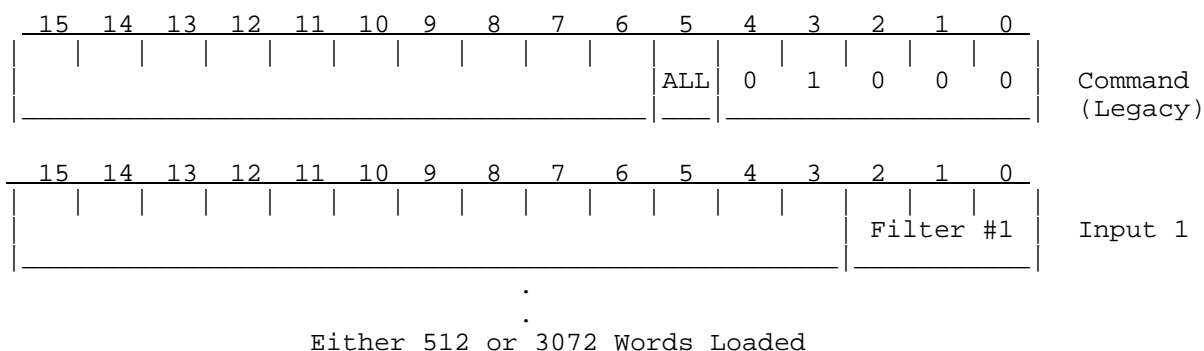
A special feature of the RVP8 processor is that any of the available clutter filters may be chosen independently at each selected range. This range-dependent clutter removal is useful when the clutter characteristics vary with increasing range. Typically, clutter interference is most severe in the immediate vicinity of the radar. Thus, a highly rejective filter might be chosen for near ranges, and a less rejective or perhaps no filter could be used at far ranges.

### Legacy Version

In the legacy version of LFILT, the input words following the command specify the choice of filter to be applied at each of the selected range bins. A fixed size filter table is always loaded, regardless of whether the range mask (See LRMSK) is using the full number of bins. In such cases, the later filter codes are simply ignored for the current range mask. However, if a longer

range mask is loaded in the future, then those later codes would apply to the correspondingly numbered bins. Put another way, each filter code is associated with a particular bin number, not with a particular range. The correspondence between bin numbers and actual ranges is made only through the range mask.

Only the low three bits are used in each word to specify the filter number. If the ALL bit is set in the Command, then 3072 words are loaded, corresponding to the maximum number of range bins that are allowed. Otherwise only 512 words are loaded, and the 512th filter choice is replicated for all bins further in range.



### **Enhanced Version**

The RVP8 supports an enhanced version of the LFILT command that provides “Clutter Maps”, i.e., much greater flexibility by allowing filter choices to depend on antenna angle as well as range. This lets you specify a 2D or even 3D table of clutter filter selections that are dynamically selected during live data processing.

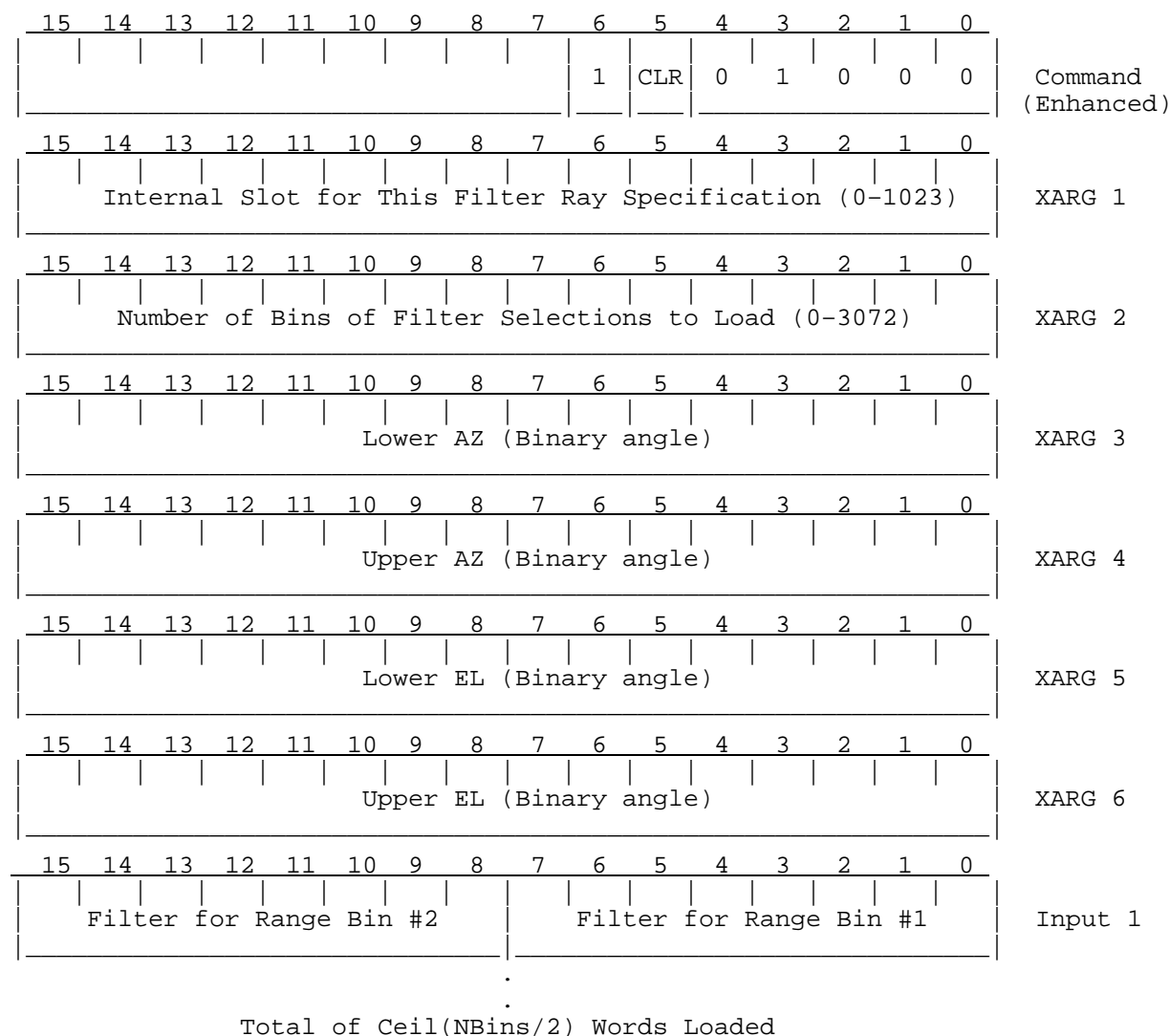
The RVP8 maintains an internal array of up to 1024 different filter-versus-range tables, each of which is keyed to a particular solid angle AZ/EL sector. Each enhanced LFILT command fills in one of these slots with a filter selection table similar to that of the legacy command, except that the number of range bins is specified explicitly and eight bits are used for filter selection rather than three. Then, for each live ray being processed, the RVP8 applies clutter filters according to the filter slot whose solid sector includes the midpoint AZ/EL of the ray. If the antenna angle of the ray does not fall within any of the defined filter sectors, then the all-pass filter (#0) will be used at all ranges.

The low and high angle limits in each filter slot are inclusive; hence, the pair (0x000, 0xFFFF) would span the full 360-degree circle with no gaps. Also, the filter array can be sparse (not all slots filled in), and have overlapping sectors (in which case the highest numbered slot that spans a ray's AZ/EL midpoint will be used). Choosing the highest numbered encompassing slot is a subtle but important detail that allows complex regions to be defined as a layered hierarchy of overlapping sectors. For example Slot-0 might define a default 360-degree filter-versus-range table, while Slot-1 defines special filtering within 0-90 degrees that is further modified by Slot-2 filters in a 40-50 degree span. A 45-degree ray would then be filtered according to Slot-2, a 60-degree ray would use Slot-1, and a 100-degree ray would use Slot-0.



If the CLR bit is set in the opcode, then no additional arguments follow and the entire internal filter array (all slots) will be invalidated. The result is that no clutter filter will be applied to any of the processed data, regardless of Range, AZ, or EL. Moreover, loading a given slot with a table consisting of zero bins of filter data will invalidate just that one slot. This allows some data to be removed from the table without having to resort to a complete CLR operation.

The legacy LFILT command is equivalent to calling the enhanced command first with the CLR bit set, followed by a second call that writes the legacy filter choices into slot #0 using AZ/EL limits that cover all of space. Thus, the legacy behavior is obtained as a special case of the enhanced mechanism.

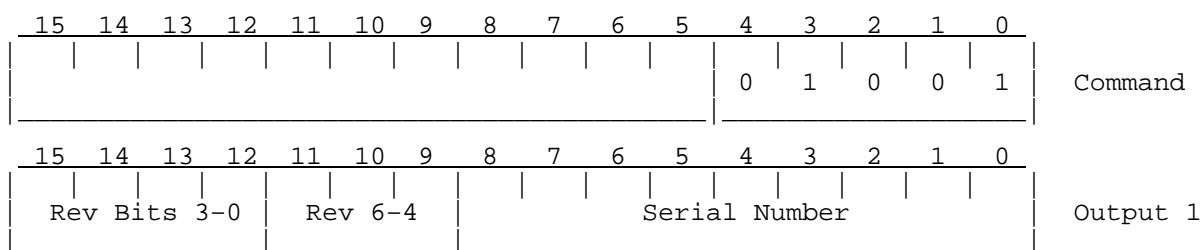


## 6.9 Get Processor Parameters (GPARM)

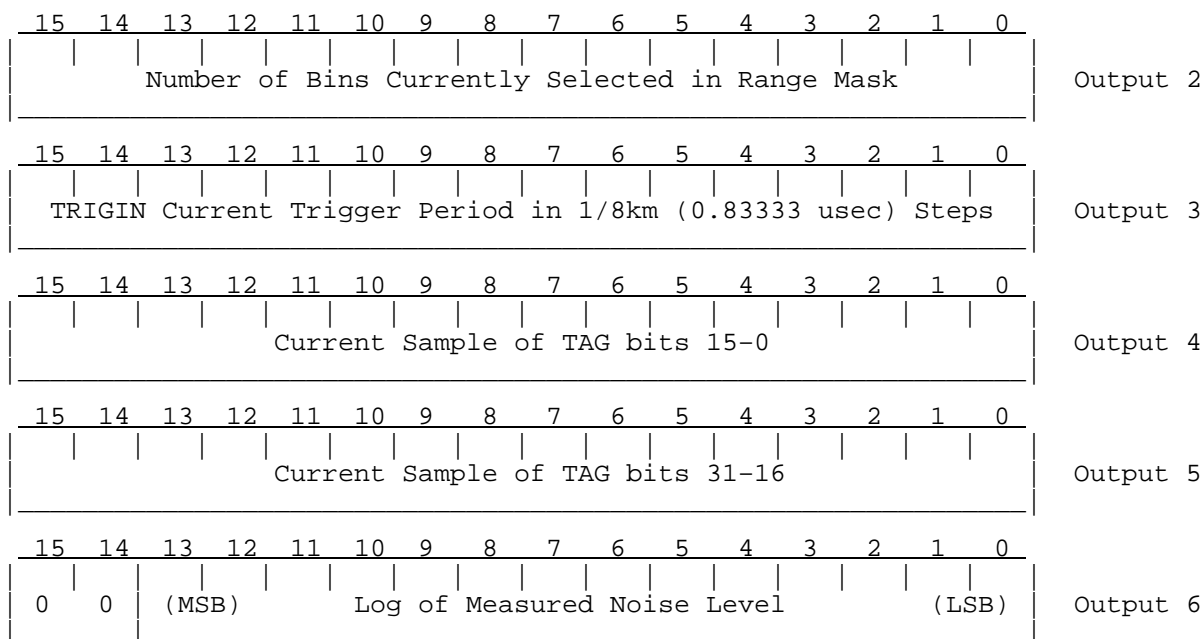
This command is used to access status information from the RVP8 processor. Sixty-four words are always transferred, some later words are reserved for future compatibility and are read as zeros. For convenience, a shorthand table of the output words is given in Table 6-2.

**Table 6–2: RVP8 Status Output Words**

Word	Description	Word	Description
1	Revision / Serial number	33	LOG Threshold
2	Number of Range Bins	34	CCOR Threshold
3	Current trigger period	35	SQI threshold
4	Current TAG00 – TAG15	36	SIG Threshold for Width
5	Current TAG16 – TAG31	37	Calibration Reflectivity
6	Log of Measured Noise Level	38	— Reserved —
7	“I” Channel DC Offset	39	— Reserved —
8	“Q” Channel DC Offset	40	Range Averaging Choice
9	Latched Processor Status	41	— Reserved —
10	Immediate Status Word #1	42	— Reserved —
11	Diagnostic Register A	43	Header configuration of PROC data
12	Diagnostic Register B	44	I-Squared Noise (Low 16-bits)
13	Number of Pulses / Ray	45	I-Squared Noise (High 16-bits)
14	Trigger count (Low 16-bits)	46	Q-Squared Noise (Low 16-bits)
15	Trigger Count (High 8-bits)	47	Q-Squared Noise (High 16-bits)
16	No. of Properly Acquired Bins	48	Log of Measured Noise Level
17	No. of Properly Processed Bins	49	LOG Noise Standard Deviation
18	Immediate Status Word #2	50	Horizontal/Vertical Noise Ratio
19	Noise Range in Km	51	AFC/MFC Control Value
20	Noise Trigger Period	52	Interference Filter Select
21	Pulse Width 0 min. Trig. Period	53	Interference Filter C1 Constant
22	Pulse Width 1 min. Trig. Period	54	Interference Filter C2 Constant
23	Pulse Width 2 min. Trig. Period	55	Immediate Status Word #3
24	Pulse Width 3 min. Trig. Period	56	Burst Tracking Slew
25	Pulse Width Bit Patterns	57	Polarization Algorithm Choices
26	Current /Pulse Width	58	Range Mask Spacing
27	Current Trigger Gen. Period	59	Immediate Status Word #4
28	Desired Trigger Gen. Period	60	— Reserved —
29	PRT at Start of Last Ray	61	— Reserved —
30	PRT at End of Last Ray	62	— Reserved —
31	Processing/Threshold Flags	63	— Reserved —
32	Log Slope	64	— Reserved —

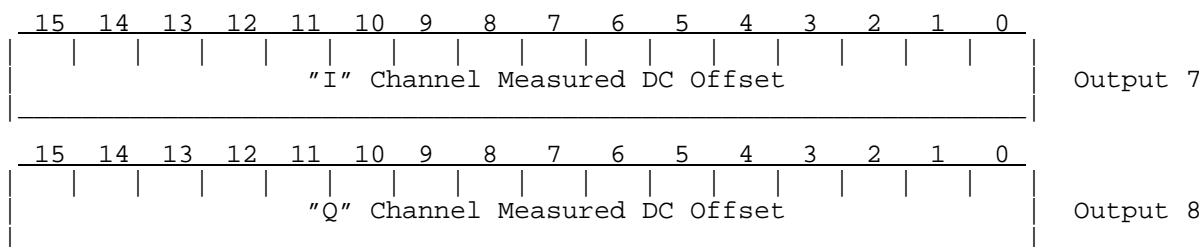


The revision and serial numbers of the particular RVP8 board are accessible here. This information is useful when computer software is being designed to handle a variety of signal processor revisions. The revision number is seven bits total; four of which are still in the high four bits of the word for compatibility with an older format.

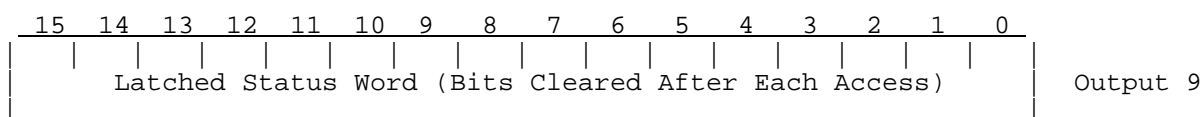


This value is scaled 4 times higher than the time series LOG format (see the discussion in Section 6.7). To convert to dBm, use the equation:

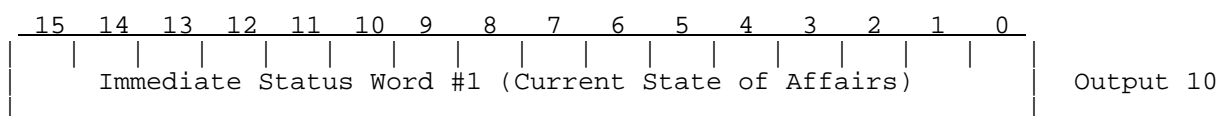
$$dBm = P_{MAX} + Slope \times [(Value/4) - 3584]$$



These two words convey the measured "I" and "Q" DC offsets from the last noise sample. The output format is either signed 16-bit values in which  $\pm 32767$  represent  $\pm 1.0$  (legacy format), or packed timeseries values using the High-SNR encoding format. Bit-9 of GPARM Word-59 tells which format to use.



- Bit 0 No Trigger during noise measurement.
- Bit 1 Trigger too fast during noise measurement, i.e., some of the noise sample bins were positioned past the trigger range.
- Bit 2 No trigger during PROC command.
- Bit 3 PRT varied by more than 10 microseconds within portions of a processing interval that should have been at a fixed rate.
- Bit 5 FIFO overflow during last PROC command.
- Bit 6 Command received while waiting for output FIFO space. The command was processed but some output data has been lost (zeroed).
- Bit 7 Error detected during last SNOISE command.
- Bit 9 Error in last Load Range Mask (LRMSK) Command. This generally means that too many range bins were selected.
- Bit 10 Error in LSIMUL command protocol.
- Bit 11 Measured phase sequence is incorrect.
- Bit 15 Invalid processor configuration. This bit is set if the last PROC command called for an illegal combination of parameters. The possible causes are:
  - Spectrum size greater than 128 or less than 4
  - More than 342 bins/slave in FFT modes
  - (bins/slave) x (4 + sample size) exceeds 26200 in FFT modes
  - (bins/slave) x (sample size) exceeds 3000 for Time Series or Spectra output
  - Odd number of bins selected during fast polarization switching
  - Bad combination of polarization parameters



- Bit 0 No trigger, or, more than 50ms. since last trigger.
- Bit 1 Error in loading trigger angle table (See LSYNC Command).
- Bit 2 PWINFO command is disabled.
- Bit 3 Angle sync input is BCD (Else binary angle).
- Bit 4 Angle sync is on elevation axis (Else azimuth axis).
- Bit 5 Angle sync is enabled.
- Bit 6 Angle sync is not interruptible.
- Bit 7 Angle sync is dynamic (else rays begin on sync angles).
- Bit 8 DSP has full IAGC hardware and firmware configuration.
- Bit 9 DSP supports 16-bit floating time series.
- Bits 11,10 Current unfolding mode.
- Bits 13,12 Number of RVP8/PROC compute processes minus one
- Bit 14 DSP supports Power Spectrum output

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Diagnostic Result Register A															

Output 11

- Bit 0 RVP8/Rx card #1 failure
- Bit 1 RVP8/Rx card #2 failure
- Bit 2 RVP8/Tx card #1 failure
- Bit 3 RVP8/Tx card #2 failure
- Bit 4 IO62 card #1 failure
- Bit 5 IO62 card #2 failure
- Bit 6 Error loading config/setup files
- Bit 7 Error attaching to antenna library
- Bit 8 Problem when forking compute processes
- Bit 9 Error(s) in softplane configuration
- Bit 10 Signals raised during startup
- Bit 11 RVP8 running without 'root' privileges
- Bit 12 Problem creating daemon process
- Bit 13 Inconsistent setup values detected

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Diagnostic Result Register B															

Output 12

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Number of Pulses Being Integrated															

Output 13

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Trigger Count (Low 16-bits)															

Output 14

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Trigger Count (high 8-bits)															

Output 15

The trigger count is a running tally of the number of triggers received by the RVP8 on the TRIGIN line. It is a full 24-bit counter.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Number of Properly Acquired Bins for Current Range Mask & PRT															

Output 16

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
No. of Valid Bins in Initial Part of Ray From Last PROC Cmd															

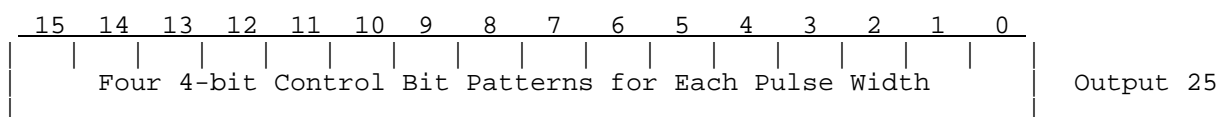
Output 17

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Immediate Status Word #2 (Current State of Affairs)															

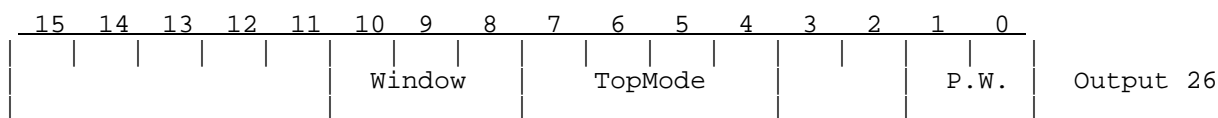
Output 18

- Bit 0 Processor supports FFT algorithms
- Bit 1 Processor supports Random Phase algorithms
- Bit 2 Reserved (zero)
- Bit 3 Processor supports DPRT-1 (Dual-PRT) algorithms
- On dual IFD systems: Bits 4,5,7, and 11 are set if either IFD fails:
- Bit 4 Problem in UpLink cable from RVP8/Rx → IFD
- Bit 5 Problem in DownLink cable from IFD → RVP8/Rx
- Bit 7 IFD PLL is not locked to external user-supplied clock reference
- Bits 8–10 Status of burst pulse and AFC feedback
- |                            |                               |
|----------------------------|-------------------------------|
| 1: AFC Disabled            | 2: Manual Frequency Control   |
| 3: No burst pulse detected | 4: AFC is waiting for warm-up |
| 5: AFC is locked           | 6: AFC is tracking            |
- Bit 11 IFD test switches are not in their normal operating position
- Bit 12 Set according to whether the RVP8 is performing trigger blanking. This allows the host computer to decide whether to interpret the End-TAG-0 bit in the output ray header as a blanking flag, or as a normal TAG line.
- Bit 13 Missing signal at IFD #1 Burst Input
- Bit 14 Reserved (zero)
- Bit 15 Set when valid burst power is detected but the center-of-mass lies outside of the aperture sub-window that defines the portion of the pulse used for AFC analysis. This error bit effectively flags when the burst pulse has drifted out of its optimal placement within the sampling window.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Starting Range in Km at Which Noise Sample Data are Taken																Output 19
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Trigger Period (0.16667usec Increments) During Noise Sampling																Output 20
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Min Trig Period (0.16667usec Increments) for Pulse Width 0																Output 21
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Min Trig Period (0.16667usec Increments) for Pulse Width 1																Output 22
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Min Trig Period (0.16667usec Increments) for Pulse Width 2																Output 23
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Min Trig Period (0.16667usec Increments) for Pulse Width 3																Output 24



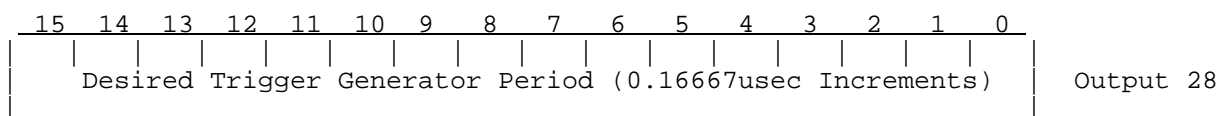
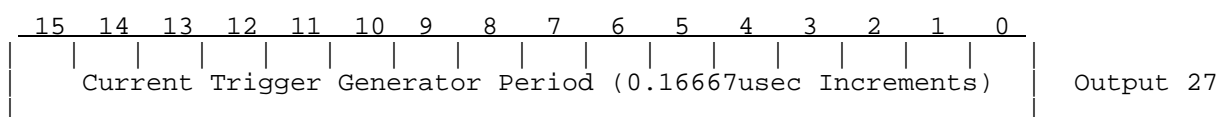
See PWINFO command, input word #1, for definition of these bits.



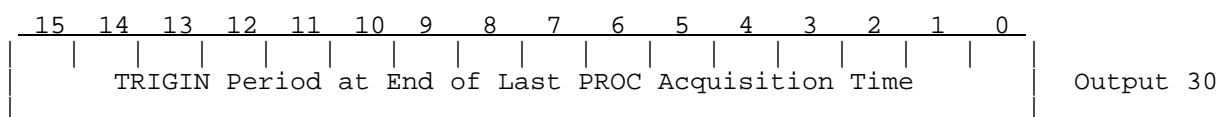
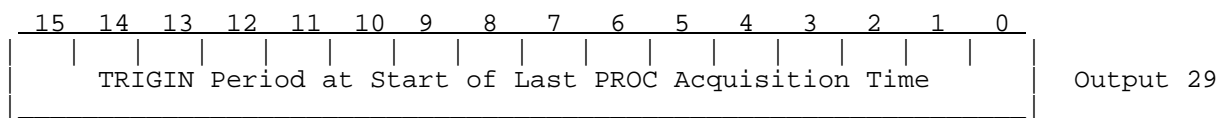
P.W. Currently selected radar pulse width

TopMode Major Mode (See SOPRM Input #9)

Window Spectral Window Choice (See SOPRM Input #10)

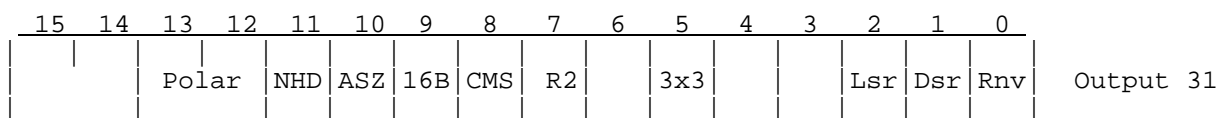


The desired trigger generator rate is that which was selected in the most recently issued SETPWF command (or power-up rate if SETPWF was never issued). The current rate may be different from the desired rate due to bounding against limits for the current pulse width, or being in an odd ray cycle during dual-PRT processing. The measured PRT's are forced to 0xFFFF (the maximum unsigned value) whenever the external trigger is expected but missing.



The PRTs from the start and end of the last ray are the actual measured values whenever possible, i.e., when non-simulated data are being processed, and we either have an external trigger, or an internal trigger that is not in any of the Dual-PRT modes. The units are the same as for the measured current trigger period in Output #3.

Outputs 31 through 37 are the current processing and threshold parameters set by SOPRM. See Section 6.3 for additional notes on each of these parameters. Since the threshold levels for each data parameter can be different (See THRESH command, Section 6.29), words 33-36 are taken from the velocity parameter.



15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	Log Slope 65536 * dB / LSB	Output 32
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	LOG Noise Threshold in 1/16 of dB	Output 33 (From V)
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	Clutter Correction (CCOR) Threshold in 1/16 of dB	Output 34 (From V)
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	SQI Threshold	Output 35 (From V)
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	SIG Threshold in 1/16 of dB	Output 36 (From V)
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	Calibration Reflectivity in 1/16 of dB	Output 37
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	Reserved (Zero)	Output 38
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	Reserved (Zero)	Output 39
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	Range Avg (From LRMSK Command)	Output 40
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	Reserved (Zero)	Output 41
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	Reserved (Zero)	Output 42
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	Header Config of PROC data (CFGHDR Input #1, Section 6.22)	Output 43



15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Noise Sum of I Squared								MSB=2** <sup>-16</sup>				LSB=2** <sup>-31</sup>				Output 44
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Noise Sum of I Squared								MSB=1				LSB=2** <sup>-15</sup>				Output 45
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Noise Sum of Q Squared								MSB=2** <sup>-16</sup>				LSB=2** <sup>-31</sup>				Output 46
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Noise Sum of Q Squared								MSB=1				LSB=2** <sup>-15</sup>				Output 47

To compute the noise power in dBm from Words 44-47, first calculate:

$$N_I = (\text{Word 45}) \times 2^{-15} + (\text{Word 44}) \times 2^{-31}$$

$$N_Q = (\text{Word 47}) \times 2^{-15} + (\text{Word 46}) \times 2^{-31}$$

From which we obtain:

$$dBm = P_{MAX} + 10 \log_{10} (N_I + N_Q) - 3dB$$

Note that the four integer values become rather small and severely quantized when the noise power drops to low values. Historically, these four words were used to balance the individual gain of the “I” and “Q” channels in the RVP6 in the presence of a strong test signal. Since “I” and “Q” are inherently balanced in the RVP8, these output words are no longer of much value.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Log of Measured Noise Level (same as word 6)																Output 48
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Vert Noise Stdev in dB*10								Horiz Noise Stdev in dB*10								Output 49

The noise standard deviations for each receive channel are normalized to the mean power. The values reported here will therefore hover around 0dB for ordinary exponentially distributed noise in which the standard deviation scales directly with the mean.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Ratio of Horizontal/Vertical Noise Power in Hundredths of dB																Output 50
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
16-Bit AFC/MFC Value (-32768 through +32767)																Output 51
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PhaseSeq				IFD Sat.Power				MinRev				Inter.F				Output 52

- Inter.F Specifies which interference filter is running. Zero means “none”; see Section 5.1.5 for a description of the interference filter algorithms.
- MinRev Minor revision level of the RVP8 code that is currently running.
- IFD Sat.Power ( $P_{MAX}$ )  
Input power required to saturate the IF-Input A/D converter for the RVP8/IFD receiver that is currently attached.  
0: +4.5dBm 1: +6.0dBm

PhaseSeq Tx Phase modulation sequence (See Section 6.27)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Interference Filter Parameter “C1” in Hundredths of deciBels																Output 53
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Interference Filter Parameter “C2” in Hundredths of deciBels																Output 54
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Immediate Status Word #3 (Current State of Affairs)																Output 55

- Bit 0 Burst pulse timing adjustments can be made
- Bit 1 Burst pulse frequency adjustments can be made
- Bit 2 Burst pulse hunting is enabled
- Bit 3 Burst pulse hunt is running right now
- Bit 4 Last burst pulse hunt was unsuccessful
- Bit 5 Processor supports DPRT-2 (Dual-PRT) algorithms
- Bit 6 Could not generate the requested phase sequence
- Bit 7 Problem with digital transmitter clock
- Bits 8-11 User-defined Major Modes 1-4 are supported

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Signed trigger slew in hundredths of microseconds																Output 56

This is the same format that is used by the SETSLEW command to set the current trigger slew (See Section 6.25)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Polarization Algorithm Choices																Output 57

- Bit 0 Use H transmissions for (T,Z,V,W)
- Bit 1 Use V transmissions for (T,Z,V,W)
- Bit 2 Use Co-Pol reception for (T,Z,V,W)
- Bit 3 Use Cross-Pol reception for (T,Z,V,W)
- Bit 4 Correct all polar Parameters for noise

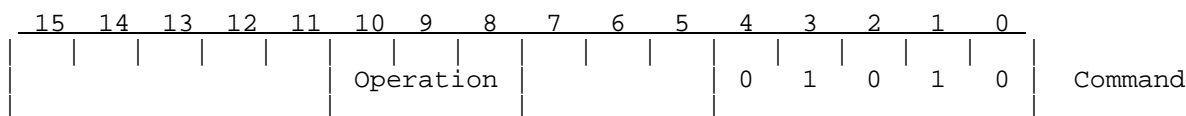
Bit 5	Use filtered data for all polar Parameters															
Bit 6	Sign convention for PHIdp															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Range mask spacing for current pulsewidth (centimeters)																Output 58
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Immediate Status Word #4 (Current State of Affairs)																Output 59
Bit 0	Internal power spectra size matches sample size (else power-of-2)															
Bit 1	PROC command output spectra match sample size (else power-of-2)															
Bit 2	Trigger pattern has been altered to fit within the desired PRT															
Bit 3	PRT has been altered to preserve the desired trigger pattern															
Bit 4	Using High-SNR packed (I,Q) format															
Bit 5	Trigger sequence truncated due to insufficient pattern memory															
Bit 6	TimeSeries data source is external to the RVP8															
Bit 7	WSR88D “Batch” mode is supported															
Bit 8	Major mode refuses to use external trigger															
Bit 9	GPARM outputs #7 and #8 use Hi-SNR format, else linear															
Bit 10	Receiver protection fault															
Bit 11	IFD dual-channel inconsistency (e.g., power and/or phase out of bounds for ratio of HiGain-to-LoGain channels)															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Reserved (Zero)																Output 60
.																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Reserved (Zero)																Output 64

## 6.10 Load Simulated Time Series Data (LSIMUL)

This command is provided as a diagnostic for proper functioning of the RVP8 algorithms. It permits arbitrary simulated data samples to be input to the processing routines, rather than sampled data from the A/D converters as is ordinarily the case. Since the properties of the simulated data are known exactly, it is possible to verify that the calculations within the RVP8 are proceeding correctly.

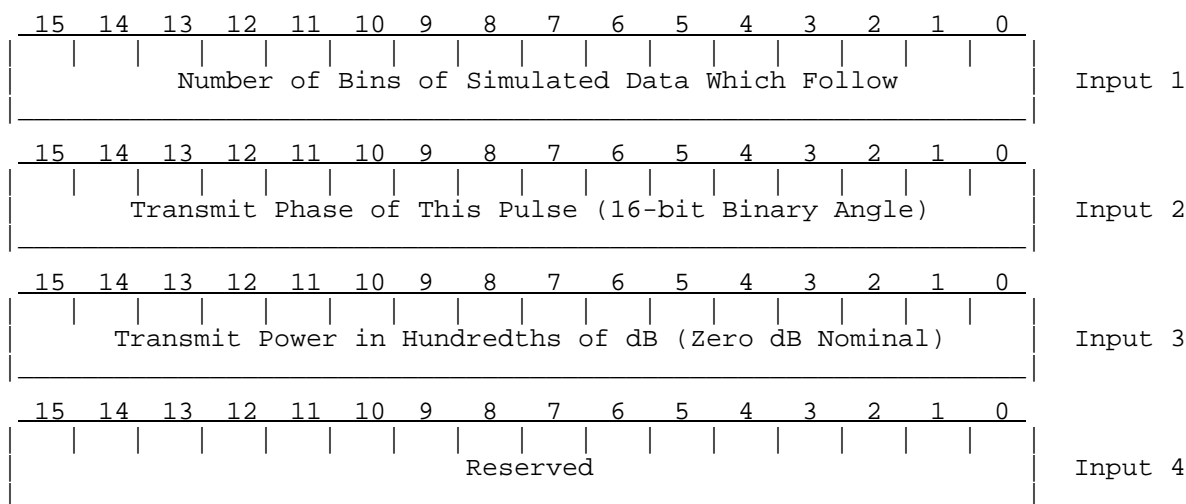
The LSIMUL command (with operation=1) should be issued prior to the PROC command which is being tested. This enables the simulated data mode. The next PROC command will then wait for  $N$  ( $N$  = sample size) LSIMUL commands (with operation=2) prior to outputting each ray. The arrival of any other command during that time will cause the simulated data mode to be exited, and error bit #10 will be set in the GPARM latched status word. The error bit is also set if an LSIMUL command with operation=2 is received while simulated data mode is disabled.

You may specify a single simulated data sample for every range bin, or a pattern or simulated samples to be replicated over the range of bins. Most RVP8 algorithms are independent of range, and can be tested with identical data at every bin. Notable exceptions, however, are the “pop” clutter filter, and range bin averaging procedures. In its full generality, the LSIMUL command permits independent I and Q samples to be simulated at every bin of every pulse. If this results in more host computer I/O than is practical, then specify fewer simulated bins and allow the RVP8 to replicate them internally.



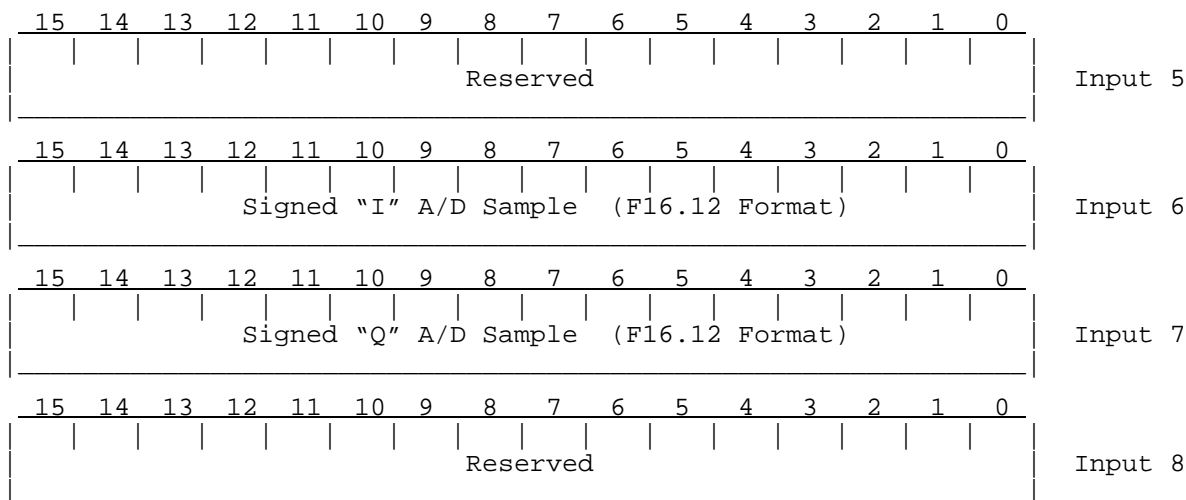
The available operations are:

- 0        Disable the use of simulated data. RVP8 returns to acquisition and processing of live data from the A/D converters.
- 1        Enable processing of simulated data. Subsequent PROC commands will use the data supplied in the next  $N$  ( $N$  = sample size) LSIMUL commands with Operation=2.  
           The receiver noise and offset levels which are internally maintained by the RVP8 are set to their special simulated values (from the **M+** setup menu) by this command. This is because the measured offsets are not relevant to the simulated data, and must not be used in the subsequent computations. Thus, it is important to issue the SNOISE command before resuming the acquisition and processing of live radar data.
- 2 or 3   Load one pulse of data samples beginning with the following 4-word header, and continuing with an array of items each representing a single instantaneous sample of (I,Q) data. You may specify one or more bins to be loaded, and the RVP8 will replicate these data as necessary in order to fill out the entire count of acquired bins. If the number of bins is zero, then a zero-valued sample is applied for all channels.

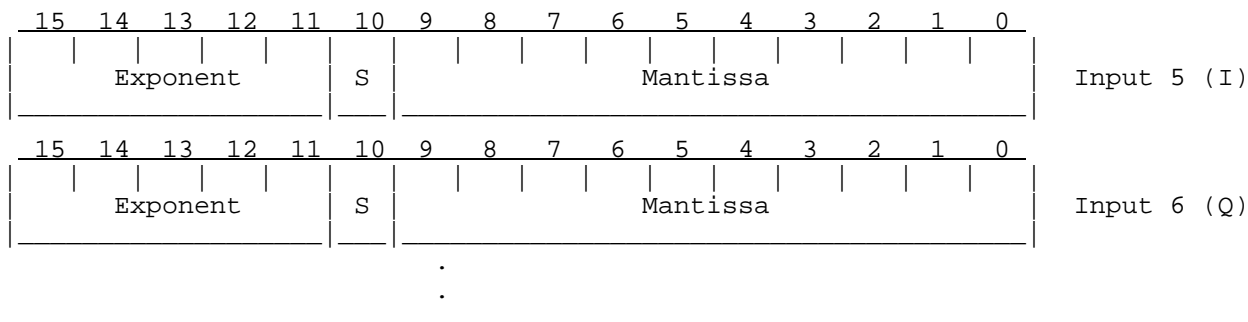


In the legacy Format #2 (RVP5-RVP8) each bin within the pulse is represented by four 16-bit fixed point words. Thus, the total number of words loaded is  $(4+4B)$ ,

where  $B$  is the bin count specified in Word #1. This takes account of the four "header" words, plus four words for every bin being defined.

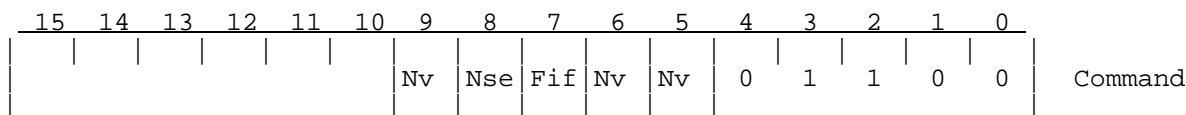


In Format #3 (new in the RVP8) each bin within the pulse is represented by two 16-bit floating point words having the exact same format as the packed (I,Q) time-series data that are output by the PROC command (See Section 6.7). Compared to the legacy 4-word format, this 2-word format uses half the I/O bandwidth, has superior dynamic range, and allows data to be fed back into the signal processor in their native packed format. The total number of words loaded (including the initial header section) is  $(4+2B)$ , where  $B$  is the bin count specified in Word #1.



## 6.11 Reset (RESET)

The RESET command permits resetting either the entire RVP8 processor, or selected portions thereof. Flags within the command word determine the action to be taken.



Nv Reloads configuration from the saved nonvolatile settings. For compatibility with RVP6 and RVP7, any of 3 bits will trigger this response.

- |     |  |
|-----|--|
| Nse | Reset the receiver noise levels to the power-up default value for all pulsewidths as defined in the <b>Mt</b> setup questions (See Section 3.2.5).   |
| Fif | Remove any data currently in the output FIFO's. This permits flushing output data that was left from a previous command, so that new output can be read from scratch. See notes in the <i>Introduction</i> to this chapter concerning actions taken by the RVP8 when the output FIFO becomes full. |

## 6.12 Define Trigger Generator Waveforms (TRIGWF)



**Note:** This opcode is obsolete, and is included only for backward compatibility with the RVP6. The opcode is disabled by default (See Section 3.2.1), because the interactive trigger setup procedure described in Section 4.3 is the preferred method of defining all RVP8 triggers and timing. TRIGWF should not be used in any new code applications that drive the RVP8.

The RVP8 has a built-in trigger generator that can synthesize six independent digital output waveforms, each having arbitrary shape and being active anywhere in a window centered around zero-range. The six trigger outputs can be defined by a 2048-word by 6-bit table which is loaded from the user computer. The patterns are automatically read from the table and output to the six trigger lines during each radar pulse. The six outputs can be used for transmitter triggers, scope triggers, range strobes, PLL gates, etc. The writable waveform table makes the RVP8 unique, in that the detailed timing of trigger and related control signals can be easily adjusted in software, without having to resort to reprogramming PROMs. This makes it possible for user software to edit the trigger timing in a convenient interactive manner.

Trigger waveforms are loaded using the TRIGWF command. Four bits in the command word (PW0 through PW3) select which pulsewidths will receive the new waveforms. On power-up, all four pulsewidths are initialized to user-selected waveforms.

The first word following the TRIGWF command specifies the transition point of the POLAR0 polarization control signal. This control signal is either held low or high for the cases of fixed horizontal or vertical polarization, or it alternates from pulse to pulse for fast-switching polarization measurements such as Zdr. The transition point is specified as a value between 0 and 2047, where 1024 represents range zero. These units are the same as the time units for the waveforms which follow, i.e., a 2048-word array holding 6-bit trigger patterns. Bit 0 in each of these words affects the TGEN0 digital output line, bit 1 affects TGEN1, etc. The bits are output at a 7.195MHz rate, and the beginning of the 1024th array word (1025th word following the command) corresponds exactly to the instant at which data at range zero are sampled by the RVP8. Note that the output rate can also be interpreted as a new bit coming every 1/48 km. In some cases this is a more useful view.

As an example, suppose we wish to make the TGEN0 output be a 0.42 microsecond pretrigger pulse, with a rising edge exactly five microseconds prior to range zero. This would be done by setting bit 0 in input words 988, 989, and 990 following the TRIGWF command, and leaving all other bit 0's clear. Further, if TGEN1 was to be a 0.14 microsecond marker strobe at 20km, we would simply set bit 1 of input word 1984.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
				PW3	PW2	PW1	PW0				0	1	1	0	1	Command

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
H																Input 1
Transition Point of POLAR0 Control Line																

H This bit defines the sense of the control line when horizontal polarization is selected.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
																Input 2
										Bits for -142.22 usec						

.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
																Input 1025
										Bits for zero range						

.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
																Input 2049
										Bits for +142.08 usec						

## 6.13 Define Pulse Width Control and PRT Limits (PWINFO)

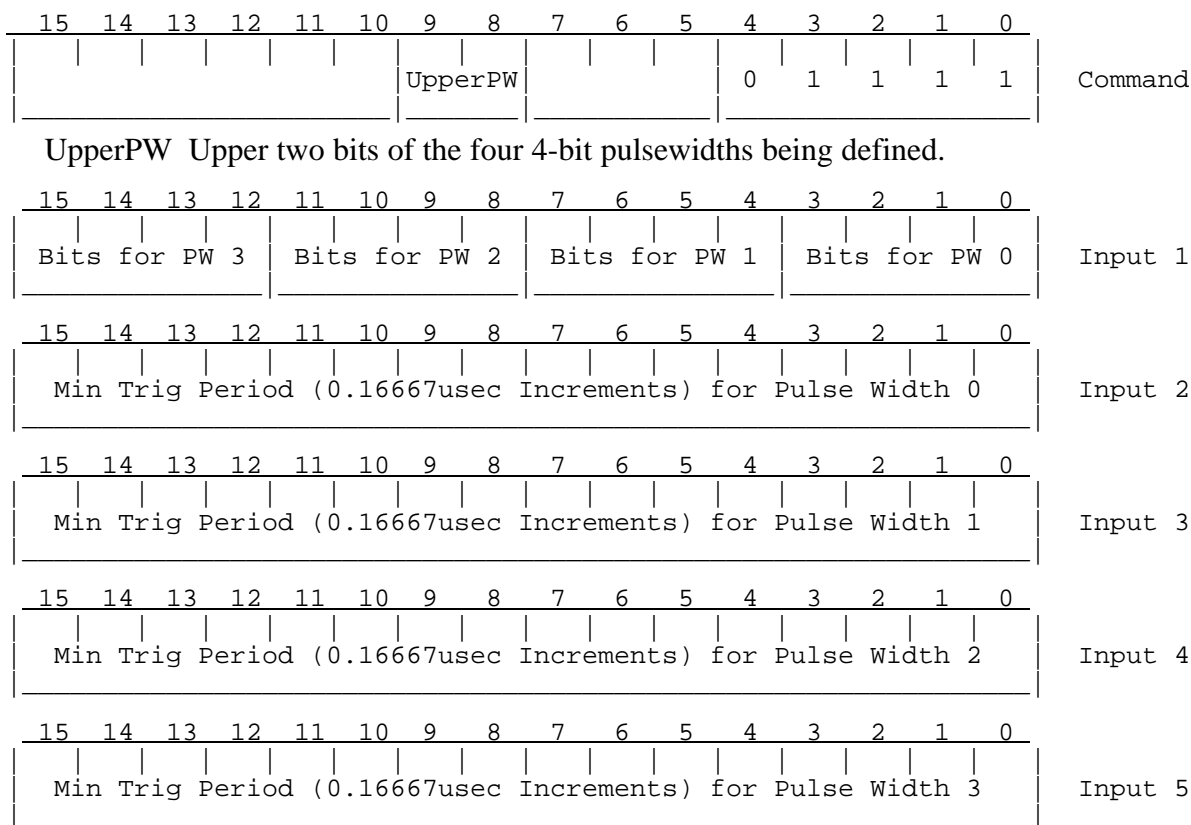
The RVP8 is equipped to control the radar transmitter's pulse width and corresponding receiver bandwidth. There are four pulse/bandwidth codes, numbered simply 0 through 3. The association between codes and pulse widths is completely determined by the needs and capabilities of the particular radar on hand. In some cases, the zero code might represent 0.25 microsecond pulse width, and in other cases it may represent 2.0 microseconds. Likewise, some radars may use all four codes, whereas others have fewer options from which to choose. The PWINFO command defines what happens for each of the four possible codes, but does not actually select which code is being used. The later function is performed by SETPWF.

The RVP8 drives four TTL output lines (PWBW0 – 3) which are intended to control the radar pulse/bandwidth hardware. Typically this control is via relays or solid-state switches in the transmitter and receiver. The user decides what state the four lines assume for each pulse width code. This is done using word #1 following the command, which contains four codes packed into one 16-bit word. The power-up default is to drive output line N low for a code of N, keeping all other lines high (Input of 7BDE Hex). The flexibility in defining the output bits usually makes the radar hardware connections very simple. For example, if pulsewidth selection relied on choosing one of four relays, then each PWBWn line could serve directly as a relay driver using the default pattern.

For each pulse width there is a corresponding minimum trigger PRT permitted. This bound is intended to limit the transmitter duty cycle to a safe value under all conditions. PWINFO sets up these minimum PRT's using words 2 through 5 following the command. The maximum

frequency of the internal trigger generator is then constrained at each pulse width to the indicated rate. This protection applies at all times, i.e., during noise sampling, during ray processing, and during the standby time between rays. The default PRT bounds are 2000, 1000, 750, and 500 Hertz (Inputs of 3000, 6000, 8000, and 12000). If your radar does not use all four pulse width codes, it is still a good idea to set the unused PRT limits to reasonable values. This way protection is still provided in the event that SETPWF accidentally selects one of the unused states. If the internal trigger generator is not being used, then the PRT limits no longer affect the actual trigger rate and transmitter protection becomes the responsibility of the the user hardware. Finally, note that the entire pulse/bandwidth mechanism can be effectively turned off by setting the four bit patterns and the four PRT limits all to the same value.

The PWINFO command can be disabled (for transmitter safety), so that PRT limits cannot accidentally be changed by the host computer. When this is done the RVP8 still reads the five input words, but no changes are made to the pulse width and PRT information. Thus, the command I/O behaves the same way, whether enabled or disabled.

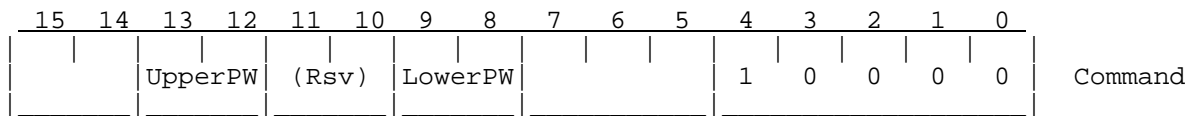


## 6.14 Set Pulse Width and PRF (SETPWF)

This command selects the pulsewidth and trigger rate. A 2-bit pulse width code is passed in bits 8 and 9 of the command word, and selects one of four pulse widths as described under PWINFO. The new radar PRT is passed in word #1. For all processing modes that use a fixed trigger rate, this value defines the trigger period that is output at all times except during noise measurements.

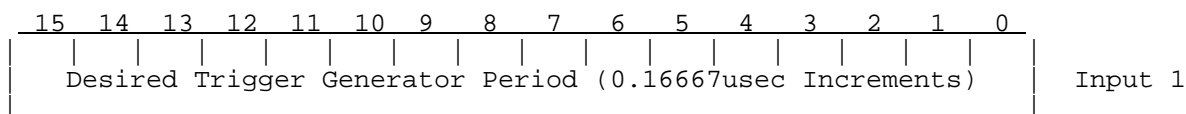


For Dual-PRF applications, this word defines the short period (high PRF) rate. The long period is internally computed as either 3/2, 4/3, or 5/4 the short period, and the trigger generator alternates between the short and long rates on each successive ray.

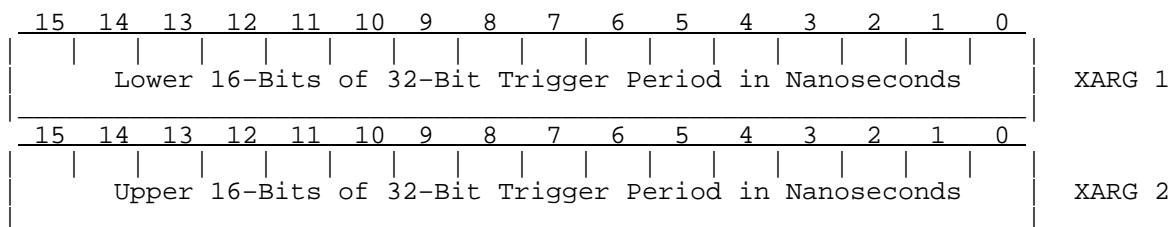


UpperPW Upper two bits of overall 4-bit pulsewidth selection

LowerPW Lower two bits of overall 4-bit pulsewidth selection



When Input #1 is zero, then the arguments take on an alternate form that allows an array of  $N$  (up to 64) trigger periods to be specified, and also gives much finer time resolution in the choice of each period. The XARGS command is first used to load an array of  $N$  32-bit words that define the trigger period(s) in nanoseconds. The RVP8 will then generate triggers whose shapes (relative starts and widths) are identical for each pulse, but whose periods follow the selected sequence. Trigger patterns such as these are intended to support research customers who use the real-time (I,Q) data stream directly.



⋮

## 6.15 Load Antenna Synchronization Table (LSYNC)

The RVP8 can operate in a mode wherein radar data are acquired in synchronization with the antenna motion along either the azimuth or elevation axis. This special feature frees the user computer from having to separately monitor the antenna angles and request each data ray individually. To use this mode, it is assumed that TAG0-15 are wired to receive azimuth angles, and that TAG15-31 are wired to receive elevation. Angle input may be in the form of either 16-bit binary angles, or four-digit BCD. This synchronization mode is the only one which ascribes any meaning to the TAG inputs; ordinarily they are merely passed on to the user computer as ancillary information.

Antenna synchronization is accomplished by way of a table of trigger angles. This table, which contains between three and 1024 angles, is used to define the angle boundaries for each processed ray. The trigger angles need not be uniformly spaced, nor must they span the full 360-degrees of rotation. This gives considerable flexibility in the choice of angles. For example, if local obstructions cause shadows in the radar image, then those regions can be

skipped merely by omitting table entries in their vicinity. Likewise, as the antenna rotates data can be acquired within one or more sectors by simply specifying the appropriate sets of contiguous bearings at whatever angular resolution is desired. Note that on power-up the angle table is initialized to 360 values corresponding to integer-valued degrees from zero to 359.

The synchronization algorithm works automatically with either clockwise or counterclockwise antenna rotation, and can tolerate any sequence of changes in direction, e.g., if the antenna itself is scanning a sector, or if it is turning erratically. Moreover, the trigger angles do not have to be hit exactly in order to start each new ray — the antenna need only move across them. This minimizes the possibility of losing data due to missing codes in the angle encoders. The RVP8 will automatically produce an output ray after one second of waiting, even if no trigger angles have been crossed. This is to avoid timeouts with the host computer when the antenna is not moving at all.

To use the synchronization mode, the trigger angle table is first loaded using the LSYNC command. The user chooses the number of table entries and then writes the required number of words to the RVP8. The angles must be supplied in a clockwise strictly increasing order, and they must neither reach nor pass zero degrees by the table's end. The first value, however, may be zero. Binary angle representation is used wherein Bit 15 represents 180 degrees, Bit 14 represents 90 degrees, etc. The Ld bit must be set in the command word to indicate that a new table size and set of angles are being loaded. A flag bit is to be set (See GPARM) if errors are detected when loading the table of angles.

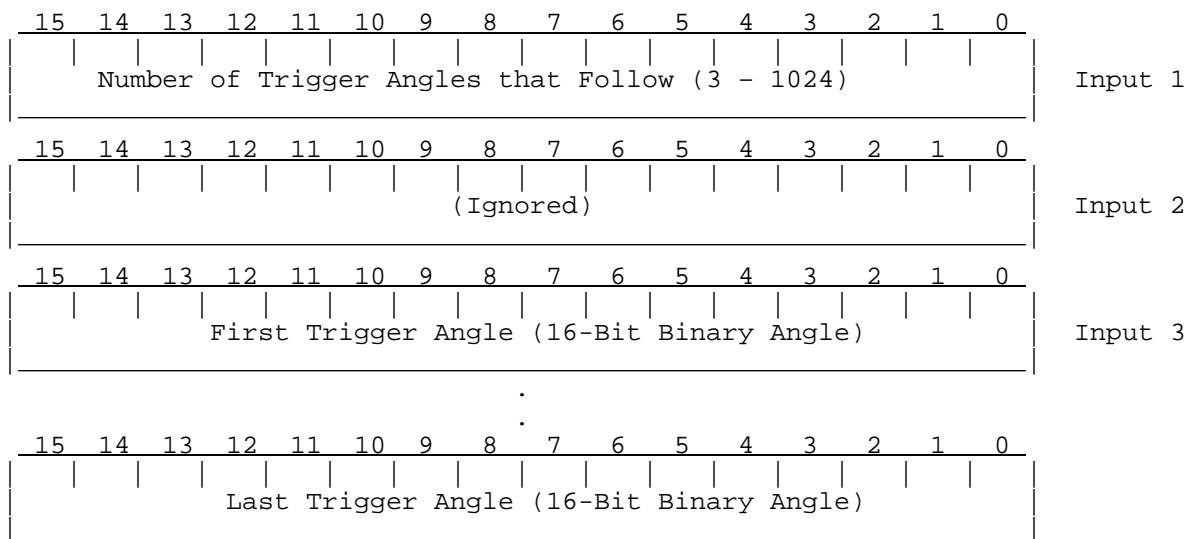
To actually enable synchronized operation the Ena command bit must eventually be set, and EL and BCD should be either set or cleared according to the user's needs. These bits may be used independent of reloading the actual table values. Thus, antenna synchronization may be turned on and off without having to reload the table each time. However, if there were errors when the table was last loaded, the processor ignores the Ena bit and synchronization is forced off. Once enabled, PROC commands are then issued in the usual manner to acquire and process the radar data. Either the single-cycle or free-run PROC mode may be used. Data collection proceeds as usual, except that the rays are now automatically aligned with the trigger angles.

The angle sync algorithm is dynamic and works as follows. Each ray begins immediately upon the user's request, or upon completion of the previous ray when in continuous processing mode. At the start of the ray, the RVP8 finds the pair of sync angles that enclose the previous trigger angle. The current ray then runs until the antenna passes outside of either limit, at which point processing for that ray is terminated. Once this happens, a new trigger angle is assigned based on which limit was crossed.

The maximum number of pulses that will be present in each ray during angle syncing is still given by the Sample Size field of the SOPRM command. The actual number of pulses will be less only if a trigger angle is crossed before the full pulse count is reached. In general, you should set the Sample Size somewhat larger than the expected pulse count so that the trigger angle crossings make the best use of every available pulse when the antenna is scanning at the expected rate.

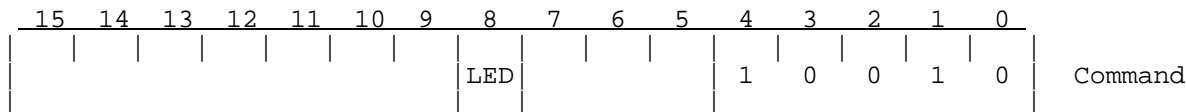
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
			NoI	Ena	El	BCD	Ld				1	0	0	0	1	Command

- NoI** Ordinarily, the potentially lengthy sync wait loop is terminated if the user writes additional words to the RVP8. Setting this bit prevents such interrupts. Beware that the processor loop can not be broken in this case except by moving the antenna across a trigger angle, or cycling the RESET\_ line.
- Ena** Enables antenna synchronization.
- El** Synchronization is based on TAG15–31 (Elevation) inputs, else TAG0–15 (Azimuth) is used.
- BCD** Specifies that TAG angle input is in the form of 4-digit Binary Coded Decimal; otherwise, a 16-bit binary angle is assumed.
- Ld** Indicates that a new table size and array of values follow the command. If Ld = 0, then LSYNC is a one-word command only.



## 6.16 Set/Clear User LED (SLED)

This command simply turns the red user LED on and off under program control. The LED is on during the initial running of internal diagnostics, and then remains off unless changed by this command. Note that the red LED can be configured to serve as an internal activity indicator (see TTY setups), in which case this command has no effect.



## 6.17 TTY Operation (TTYOP)

This command controls the TTY “chat mode” interface to the host computer. The command can simulate the typing of characters on the RVP8’s setup TTY. Characters entered in this manner are indistinguishable from those typed on the actual TTY; hence, whatever one can do via the TTY, one can also do via this command. The RVP8 sends all TTY output to whichever stream (TTY, or host computer) provided the most recent input character. This command is also used to monitor the graphical data from the special scope plotting modes.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
								Operation			1	0	0	1	1	Command

The operation codes are as follows:

- 0 Sends the ASCII character in the upper byte of the word to the RVP8 as if it had been typed on the setup TTY's keyboard.
- 1 Allow scope plotting data to be output whenever a plot is being drawn. All relevant status and data words are output once upon each receipt of this command. Subsequently, status and data will be output only when a change has taken place.
- 2 Disable the scope plotting output data.

Any of the following types of data may be output by the RVP8 while the TTY monitor is running. The order of arrival of each data type is indeterminate, but all multi-word sequences will always be output as contiguous words.

Individual "TTY" characters generated by the RVP8 are output in the low byte of the word, with the upper byte set to zeros.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	ASCII Character								TTY Char

The status of the plotting modes is given in the following word.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	0	0	0													PLT Status

PLT Indicates that a scope plot is being drawn now.

The 2-bit intensities of each of 16 possible strokes of data is given in the following 4-word sequence. An intensity of zero represents "OFF"; one, two and three are successively brighter.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	0	0	1	0	0	0	0	Int 3		Int 2		Int 1		Int 0		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	0	0	1	0	0	0	1	Int 7		Int 6		Int 5		Int 4		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	0	0	1	0	0	1	0	Int 11		Int 10		Int 9		Int 8		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	0	0	1	0	0	1	1	Int 15		Int 14		Int 13		Int 12		

The data for each stroke of the plot is given by the following sequence of 501 words.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	0	1	0													Plot Data
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	0	1	1													Word #1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
1	0	1	1													Word #500

## 6.18 Load Custom Range Normalization (LDRNV)

Reflectivities computed by the RVP8 are ordinarily corrected for range effects by adding an offset in deciBels equal to  $20 \log(R / 1\text{km})$ , where R is the range in kilometers. This correction is based on a simple filled beam geometry, and is sufficiently accurate for most meteorological observations. The LDRNV command is provided for applications in which an alternate custom range correction is required, for example, if the radar receiver's LNA were to be driven by an external user-supplied STC waveform.

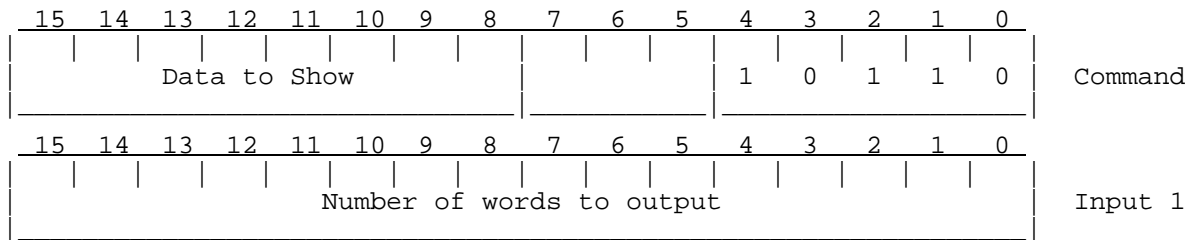
LDRNV loads a 251-word custom correction table holding values in hundredths of deciBels over five decades of  $\log(\text{range})$  from 0.01km to 1000km. There are 50 table entries per decade of range. Thus, the range in kilometers corresponding to an input word #N is  $10^{\left[\frac{N-1}{50}-2\right]}$ , and the default correction table (automatically used on power-up) is simply  $40(N - 101)$ . The table values are stored and interpolated whenever the RVP8 loads a new range mask (See LRMSK), at which point custom values for the actual user ranges are computed. The LDRNV command need be issued only once, but it must be done prior to choosing the working set of range bins.

The linear intervening gas attenuation correction (See SOPRM) is always added to the reflectivity data, regardless of whether default or custom range normalization is in effect. If this is undesirable, the intervening gas slope should be set to zero.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
											1	0	1	0	1	Command
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
																Input 1
Signed Hundredths dB of Range Normalization for Range 0.01 km																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
																Input 251
Signed Hundredths dB of Range Normalization for Range 1000 km																

## 6.19 Read Back Internal Tables and Parameters (RBACK)

This command permits some of the RVP8 internal tables to be read back for confirmation and diagnostic purposes. This command would not generally be used during normal data acquisition and processing.



The data that can be returned are:

- 0 Full operational parameter table from last SOPRM command.
- 1 Ray history array consisting of six words per ray for the last 40 rays (in reverse time order) that were processed. Each six-word group holds
  - a. Actual number of samples that went into the ray
  - b. Time since the last ray (in tenths of ms)
  - c. Ending azimuth TAG bits
  - d. Ending elevation TAG bits
  - e. Starting azimuth TAG bits
  - f. Starting elevation TAG bits
- 2 Angle sync table from last LDSYNC command.
- 3 *Reserved (was AGC table)*
- 4 Filter selection array from the last LFILT command. This returns the filter selection codes, one per word, for all range bins described by filter slot #0.
- 5 *Reserved (was STC table)*
- 6 Custom range normalization from last LDRNV command.
- 7 Samples of the TAG input lines at 4ms intervals. The sampling begins at the moment the RBACK command is received, and continues until the output count is reached. Each 32-bit sample is output as a pair of 16-bit words:
  - a. Azimuth (TAG bits 0 – 15)
  - b. Elevation (TAG bits 16 – 31)
- 8 Doppler clutter filter coefficients (Same format as for LFCOEFS command)
- 9 *Reserved (was LOG clutter filter coefficients)*
- 10 Range mask spacing in cm for each pulsewidth
- 11 Current value of UIQ bits from Set/Clr all prior operations
- 12 Individual threshold configuration for each data type. This allows read back of the threshold table set with the THRESH command (section 6.29). Outputs 7 words per data type. Datatypes in the order specified in the selection mask.
- 13 Extended parameter information defined in *struct dspExParmIO*.
- 14 Minimum and Maximum values of the optional I/O-62 A/D converter, sampled over at least one complete pulse period. Sixteen bit signed outputs represent the

full range of the A/D converter. When the *RVP88D* backpanel is connected, the first Min/Max pair samples the *LOGVideo* input, the second pair samples the *CathodePulse* input, and the third and fourth pairs sample internal levels.

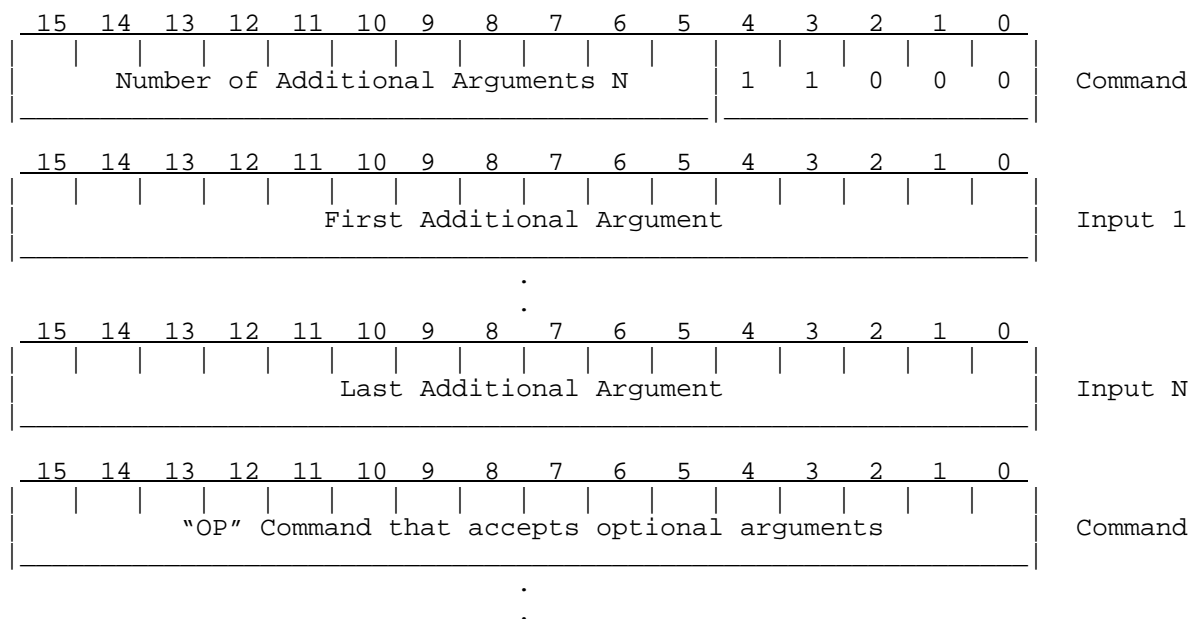
- 15 Returns an array of *struct rvp8SpecFiltIO* structures for each of the non-zero clutter filter definitions, beginning with #1. This is the same format used by the LFSPECS command to define each individual clutter filter. The order is as defined in the `PPRMS_N_* #defines` in *rvp8.h*.

## 6.20 Pass Auxiliary Arguments to Opcodes (XARGS)

This command provides a backward compatible mechanism for supplying additional (optional) arguments to other opcodes. The command may be used freely in the RVP8's instruction stream, even if the opcode being modified does not expect any optional arguments. XARGS will be a NOP in that case.

To supply optional arguments to another opcode "OP", the XARGS command is first executed with the additional argument count encoded in its upper 11-bits. This is followed by the array of between 0 and 2047 additional arguments. At this point the XARGS command is finished and the "OP" command is fetched as the next instruction. "OP" will execute normally, except that the additional arguments from XARGS can be picked up after its own input list has been read to completion.

XARGS affects only the opcode that immediately follows it. The entire list of optional arguments is discarded after "OP" executes, even if "OP" did not use some or all of the list. However, if "OP" is yet another XARGS command, then the additional arguments that it supplies will be appended to the first set. In this way, XARGS can supply an arbitrarily large number of additional arguments.



The RVP8 allows seven different clutter filters (plus the fixed all-pass filter) to be resident at once, so that an appropriate filter can be selected and applied to each processed ray based on Range, Azimuth, and/or Elevation. The LFSPECS command allows this suite of filters to be redefined on the fly.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
				0	0	0	0	0	0	0	1	1	1	1	1	Command

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
												Filter #			Input 1	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
										Filter Type						XARG 1

### Variable number of XARGS depending on Filter Type

The “Filter Number” tells which filter definition slot is being modified, and the “Filter Type” conveys the type of clutter filter to construct. The command is then followed by additional XARGs that give the specific filter parameters. Beginning with the Filter Type, the complete XARG list is a *struct rvp8SpecFiltIO* ( Please see *include/rvp8.h* ) for each of the following filter types.

Legacy clutter filter inherited from the RVP6/7, specified by a width parameter telling how many points to remove (center zero velocity point, plus one side), plus an “Edge Points” parameter telling how many points to minimize on each side of the gap to compute the end points of a linear interpolation to fill the gap.

Similar to the fixed width filter except that the width parameter is interpreted as a minimum width, and a third parameter indicates the maximum width. The actual clutter gap width will be dynamically determined at each bin based on the slopes of the spectral terms. Linear interpolation of the gap (based on “Edge Points”) is the same as above.

Similar to the variable width filter except that quadratic gap interpolation is used. This filter is experimental and should not be used.

This is the RVP8's most advanced clutter filter, combining the best techniques for determining the clutter gap width and restoring whatever low-velocity spectral points are removed. This filter is characterized by a single parameter, which is the assumed clutter width expressed as a physical velocity.



## 6.22 Configure Ray Header Words (CFGHDR)

The processed data that are output by the PROC command may contain optional header words that give additional information about each ray. This command configures the set of words that makeup each header. There are (up to) thirty two different choices of words or groups of words to include, as indicated by the bit mask following the command. Setting a bit requests that those words be included in the header, and be placed in the order implied by the sequence of the bits. Leaving all bits clear will suppress the header entirely; though this can also be done without changing the configuration via the NHD (No-Headers) bit in SOPRM Input #2.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
				0	0	0	0	0	1	0	1	1	1	1	1	Command
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
					TID	PBN	SYT	MMT	UTC	Flg	Gpm	Tim	Pul	PRT	Tag	Input 1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
																Input 2

**Tag** Four words containing two 32-Bit TAG samples, one from the beginning and one from the end of the ray:

Word #1	TAG15–0	Start of Ray
Word #2	TAG31–16	Start of Ray
Word #3	TAG15–0	End of Ray
Word #4	TAG31–16	End of Ray

- When the RVP8 is operating in dual PRF mode, bit zero of the “start” TAG word is replaced with a flag indicating that the ray’s PRF was low (0) or high (1).
- When trigger blanking is enabled, bit zero of the “end” TAG word is replaced with a flag indicating that the trigger was blanked (0) or normal (1). Note that the data within a ray are considered to be invalid if any of the pulses that were used to compute the ray were blanked. Also, the RVP8 will output all zeroed data whenever a ray contains any blanked pulses.

**PRT** PRT (Pulse Repetition Time) measured at the end of the ray. Same format as GPARM Word #30. The measured PRT’s are forced to 0xFFFF (the maximum unsigned value) whenever the external trigger is expected but missing.

**Pul** Number of pulses that were used to compute the ray.

**Tim** Milliseconds universal time (0–999), sampled at the end of the ray.

**Gpm** GPARM. Sends a copy of the 64-word GPARM output with each ray.

**Flg** Ray Flag word:

- Bit 0: Dual PRF is in the low PRF state
- Bit 1: Trigger is blanked for this ray
- Bit 2: This ray is from one of the PRFSECT special sectors
- Bits4–6: Tells which PRFSECT when Bit-2 is set

UTC	3-word universal time, sampled at the beginning of the ray Word 1: Milliseconds (0–999) Word 2: Low 16-bits of 32-bit UTC time Word 3: High 16-bits of 32-bit UTC time
MMT	MisMatched Timeseries bits (playback versus RVP8 configuration). See the MMTS_* flags in dsp.h.
SYT	IFD system clock time at the beginning of the ray Word 1: Low 16-bits of 32-bit clock counter Word 2: High 16-bits of 32-bit clock counter
PBN	Timeseries playback version number
TID	Task ID encoded as <i>struct rvp8TaskID_IO</i> (14 words total)

## 6.23 Configure Interference Filter (CFGINTF)

The RVP8 can optionally apply an interference filter to its incoming (I,Q) data stream, with the goal of rejecting occasional and sparse interference from other (usually man-made) signal sources. The CFGINTF command is used to choose which filtering algorithm will be applied, and to configure its operation via additional XARGS parameters (See Section 6.20).

If the XARGS are not supplied, then the filter parameters will simply retain their previous values. Thus, CFGINTF with no XARGS can be used to turn the interference filters On/Off without making any other changes to their threshold constants. Likewise, if only XARG 1 is supplied, then that single threshold value will be used for both C1 and C2.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Filter				0	0	0	0	0	1	1	1	1	1	1	1	Command

**Filter** Chooses which interference algorithm should be run. See Section 5.1.5 for a description of the available algorithms.

- 0: None (Interference filtering is disabled)
- 1: Alg.1 (Traditional JMA Algorithm)
- 2: Alg.2 (Alg.1 optimized for additive interference)
- 3: Alg.3 (Alg.2 with better statistics)

We recommend that you choose Alg.3 for general operational use. The other algorithms are included mostly for historical reasons.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Threshold Parameter "C1" in Hundredths of deciBels																XARG 1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Threshold Parameter "C2" in Hundredths of deciBels																XARG 2

## 6.24 Set AFC level (SETAFC)

This command sets the AFC level to a given value. The signed 16-bit span is identical to GPARM Output #51 which shows the present AFC level, i.e., corresponding to the –100% to +100% AFC range that is defined in the **Mb** menu. The RVP8 will automatically convert the

new level into whatever analog or digital AFC output format has been configured. The only exception is for the Motor/Integrator type of AFC loop, in which case this command does nothing.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
				0	0	0	0	1	0	1	1	1	1	1	1	1
				16-Bit AFC/MFC Value (-32768 through +32767)												
																Input 1

## 6.25 Set Trigger Timing Slew (SETSLEW)

The **Mt** menu allows you to select a subset of triggers that can be slewed “left” and “right” in order to place the burst pulse accurately at range zero. This command allows you to manually set the present amount of slew. The input argument is in hundredths of microseconds, i.e., ranging from  $-327.68\mu\text{sec}$  to  $+327.67\mu\text{sec}$ . The actual span permitted by the RVP8 is  $\pm 20\mu\text{sec}$ . This is the same format used in GPARM Output #56 which shows the present slew value.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
				0	0	0	0	1	1	0	1	1	1	1	1	1
				Signed trigger slew in hundredths of microseconds												
																Input 1

## 6.26 Hunt for Burst Pulse (BPHUNT)

This command starts up the internal procedure to hunt for a missing burst pulse when we are uncertain of both its time and frequency. Depending on how the hunting process has been configured in the **Mb** menu, the whole procedure may take several seconds to complete. The RVP8's host computer interface remains completely functional during this time, but any acquired data would certainly be questionable. GPARM status bits in word #55 indicate when the hunt procedure is running, and whether it has completed successfully.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
			Now	0	0	0	0	1	1	1	1	1	1	1	1	1
																Command

**Now** Forces the hunt procedure to be started even if the burst pulse is already present. Normally the procedure will only be started when the burst pulse is missing at the time BPHUNT is given.

## 6.27 Configure Phase Modulation (CFGPHZ)

This command configures the RVP8 phase control output lines, which determine the relative phase of each transmitted pulse. In some cases the phase sequence that is chosen will also have side effects elsewhere in the processor, e.g., different algorithms may be used in Random Phase mode according to the transmit sequence that is requested.

Some of the phase sequences chosen by CFGPHZ also expect additional arguments to have been supplied by the XARGS command. Phase sequences are expressed as a list of  $N$  16-bit binary angles representing the desired phase sequence. The sequence is assumed to be periodic with period  $N$ . The **Mz** command defines the correspondence between phase codes and phase angles, and is described in Section 3.2.8.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
		PhSeq		0	0	0	1	0	0	0	1	1	1	1	1	Command

- PhSeq=0 Selects **No Modulation**. The RVP8 outputs a constant default phase request as defined in the **Mz** menu.
- PhSeq=1 Selects a **Random Phase** sequence. This is also the default phase modulation that will be output following power-up. From the set of valid phase codes that are defined in the **Mz** setup section, a random code is automatically chosen for each pulse. Each code has an equal probability of being chosen each time, and the choice is independent of any previous state. No XARG words accompany this command.
- PhSeq=2 Selects a **User Defined** sequence. If no XARGS have been supplied, then the RVP8 outputs the default idle phase that is defined in **Mz**. If XARGS are supplied, then they are interpreted as a sequence of 16-bit binary angles. The RVP8 will make the best match between each desired angle and the closest realizable angle that the phase modulation hardware can produce. The maximum length of the sequence is 1024 pulses.
- PhSeq=3 Selects the **SZ(8/64)** sequence. This is a systematic code due to Sachidananda and Zrnic, which does a nice job separating and recovering first and second trip echoes in “Random Phase” mode. It will usually perform better than a truly random transmit sequence, especially when the processing interval is fairly short (as little as 32-pulses). With no XARGS, the RVP8 automatically generates the phase sequence using the closest realizable angles that the phase modulation hardware can produce. This is the recommended way to invoke SZ(8/64) coding. However, you may also supply your own 32-pulse angle sequence.

## 6.28 Set User IQ Bits (UIQBITS)

Load user-specified bits that will be included with the pulse headers in the RVP8 TimeSeries API data stream. The permanent Set/Clr bits are updated in the signal processor and retain their value from the last time they were defined. These bits are then repeated into all pulse headers. The ONCE bits, however, are transitory and will appear in only one pulse header each time they are set.

A FIFO history of the permanent bits is maintained so that the bits can be associated with the data being acquired right now as the UIQBITS opcode is executed. Each 16-bit command arg specifies bits to Set/Clr in successive bytes of the structure. This allows user code to safely change some bits without affecting others.

The user bits from separate calls will never be collapsed into a single pulse header, even if the header and bit times indicate that they could. This means that each UIQBITS opcode will always result in at least one pulse header being tagged with exactly those data. This is generally what you want, since no other exact outcome could be guaranteed based on time-of-arrival alone.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
				0	0	0	1	0	0	1	1	1	1	1	1	Command
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
				64 Permanent User Bits to SET												Inputs 1-4
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
				64 Permanent User Bits to CLEAR												Inputs 5-8
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
				64 User Bits to Apply ONCE												Inputs 9-12

## 6.29 Set Individual Thresholds (THRESH)

The SOPRM command in section 6.3 allows you to configure four threshold numbers used by all data types, and to select the threshold control flags for five of the data types. See that section for detailed documentation on how the thresholds work. Use the THRESH command if you wish to apply different threshold numbers to different data types. Using this command you can individually set the thresholds and mask used for each data type, or for groups of data types. Note that the GPARM command will read out the threshold numbers set for velocity. To read back the numbers for each data type use the RBACK command (section 6.19).

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
				0	0	0	1	0	1	0	1	1	1	1	1	Command

The first three words supply a mask that indicates which data types are being set:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	Z	T	V	W	ZDR			KDP								Input 1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
						Flg	(Tx Vert)	(Tx Horz)								Input 2
							Phi	Rho	Ldr	Phi	Rho	Ldr	SQI	RHV	PDP	

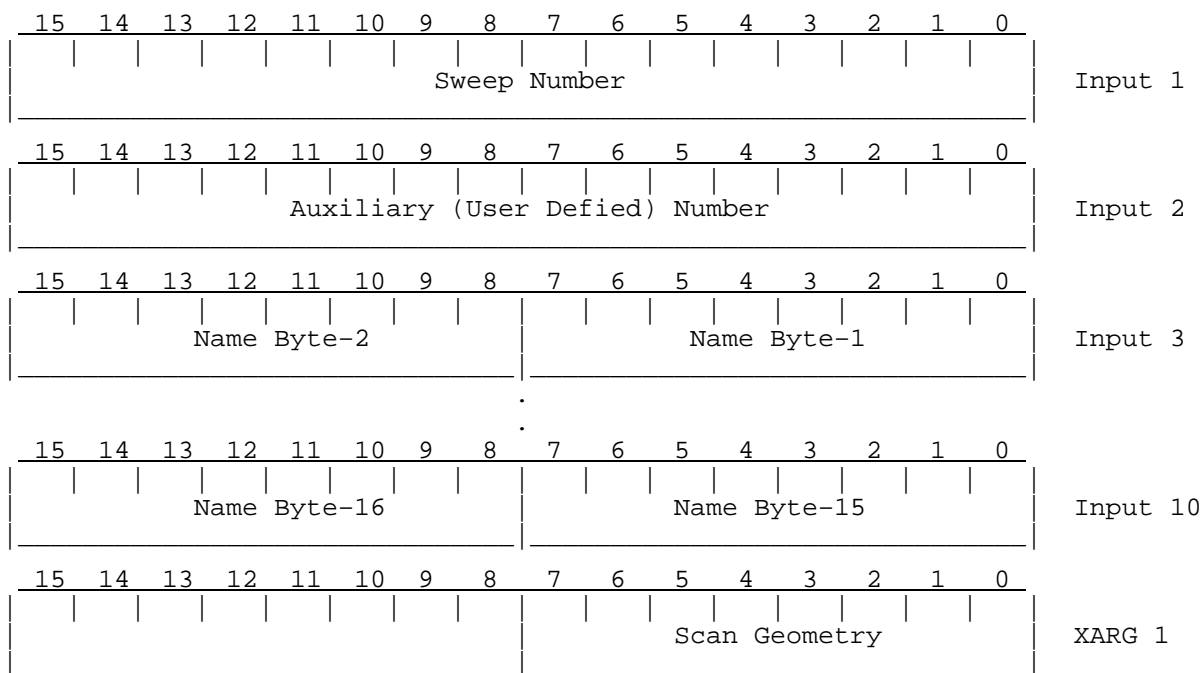
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
							<all spares>									Input 3
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
			LOG Threshold in 1/16 of dB													Input 4
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
			CCOR Threshold in 1/16 of dB													Input 5
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
								SQI Threshold (SOPRM Input 6)								Input 6
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
			Weather Signal Power Threshold in 1/16 of dB													Input 7
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
			Threshold Control Flags													Input 8
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
							<spare>									Input 9
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
							<spare>									Input 10

## 6.30 Set Task Identification Information (TASKID)

This command allows the user to “name” the (I,Q) data that are currently being acquired by the RVP8. This naming information then becomes associated with these data, and is available in the pulse information structures (*struct rvp8PulseInfo*) that are read from the Timeseries API. The *iAqMode* field of the pulse headers (*struct rvp8PulseHdr*) will be incremented each time a TASKID opcode is received, but the continuous flow of (I,Q) data from the RVP8/Rx card(s) will not be disturbed in any way.

The TASKID command defines a 16-character Null-terminated name, along with a 16-bit sweep number and 16-bit auxiliary (user defined) number. You may use all sixteen characters of the name, as it is stored internally in seventeen slots. The “Sweep Number” and “Scan Geometry” (one of SCAN\_xxx parameters) should be filled in with values best approximating those notions. “Auxiliary Number” may be filled in with any value that you find meaningful.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
				0	0	0	1	0	1	1	1	1	1	1	1	Command

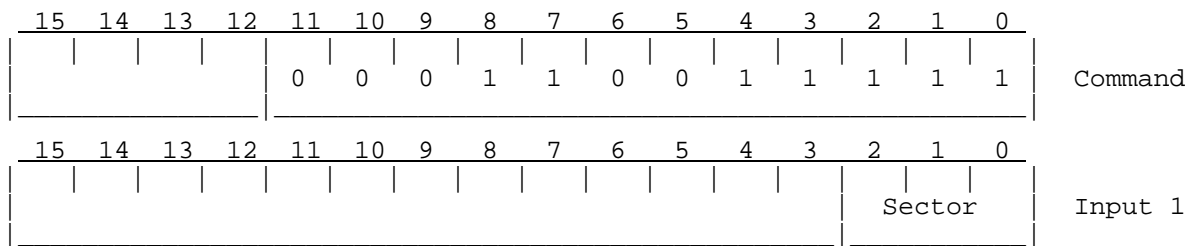


### 6.31 Define PRF “Pie Slices” (PRFSECT)

This command supplements the SETPWF command (Section 6.14) and allows an alternate trigger PRF to be generated within prescribed AZ/EL sectors. As many as eight different trigger sectors can be defined by invoking PRFSECT for each separate region. The trigger pattern will then automatically change whenever the antenna enters any of these regions, but the timeseries data will remain continuous and uninterrupted throughout each change. The motivation behind PRFSECT is that it allows a complete volume scan to run with PRFs that have been optimized to the radar echoes in all directions. Some caveats should be observed:

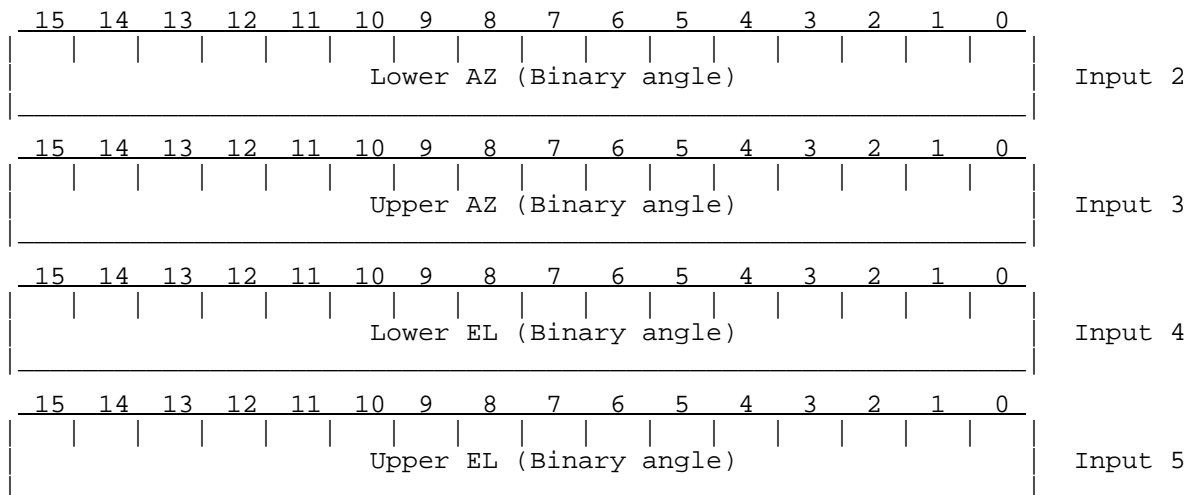
- Dual-PRF unfolding can not possibly work properly at PRF sector boundaries, so we recommend not using Dual-PRF and PRFSECT at the same time.
- When PRF sectors are used in conjunction with angle sync'ing, it is best to set the PRF sector boundaries at the midpoint between individual sync angles. This will prevent the PRF seam from bobbling between two adjacent sync angles.

The SETPWF opcode completely erases any alternate sectors that have been setup so far. Thus, the PRFSECT command can only be used after SETPWF has established the pulsewidth and default trigger rate for the entire AZ/EL scan volume.

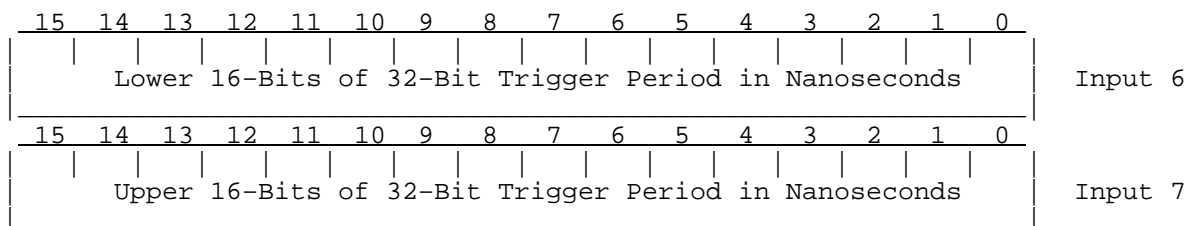


**Sector**      Selects which sector number is being defined to have an alternate trigger pattern. This is simply an arbitrary index from 0-7.

The following four arguments define a solid sector in azimuth and elevation within which the alternate trigger pattern will be used in preference to the default (SETPWF) pattern. When the current AZ/EL angle pair is contained in more than one defined sector, then the trigger pattern from the lower numbered sector will be used. Note that the sector bounds are inclusive, i.e., they include the Upper/Lower AZ/EL boundaries themselves. This convention makes it simpler to define several contiguous regions without generating slivers in between.



The following two arguments specify a trigger period in the same manner as the optional form of the SETPWF command.



.

## 6.32 Configure Target Simulator (TARGSIM)

The RVP8 contains a built-in target simulator tool that can test and debug processing algorithms that work with multiple trip returns. Several real physical targets can be simulated, each having a range span measured in kilometers, a Doppler shift in Hertz, and an echo power relative to the saturation level of the receiver. The echoes are placed in range exactly according to how they have been illuminated by whatever sequence of pulses have been transmitted so far. Multiple trip returns and range folding are all modeled correctly.

The target simulator can be used with both live and simulated (I,Q) data (See LSIMUL opcode in Section 6.10). In the former case, it allows you to overlay simulated physical targets on top of real physical targets from the radar receiver.

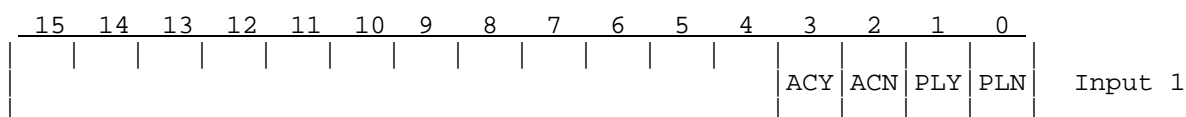


15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
			Oper	0	0	0	1	1	0	1	1	1	1	1	1	Command
Oper=0    Disable all target simulation activity (no additional args) Oper=1    Enable simulation of all defined targets (no additional args) Oper=2    Define a new simulated target (arg list follows)																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
								Co	Cx						Targ	Input 1
Targ        Which target is being defined Co/Cx       Place simulated target in Co-Pol and/or Cross-Pol Rx channels																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
																Input 2
Target Starting Range in Tenths of Kilometers																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
																Input 3
Target Length in Tenths of Kilometers																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
																Input 4
Target Power in Tenths of dB Relative to Saturation (signed)																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
																Input 5
Target Power Delta over Range Span in Tenths of dB (signed)																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
																Input 6
Target Doppler Shift in Hertz (signed)																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
																Input 7
Target Doppler Delta over Range Span in Hertz (signed)																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
																Inputs 8-11
Four Spare Words																

## 6.33 Set Burst Pulse Processing Options (BPOPTS)

Some burst pulse processing options can be set by this command.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
				0	0	0	1	1	1	0	1	1	1	1	1	Command

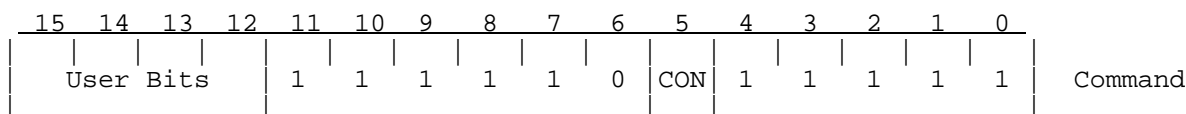


**PLY/N** These bits affect whether the RVP8 will phase lock its (I,Q) data to the measured burst pulse. The “PLY” and “PLN” bits force “Yes” and “No” responses. If both bits are clear or both bits are set, then no change will be made.

**ACY/N** These bits affect whether the RVP8 applies pulse-to-pulse amplitude correction to its (I,Q) data. The Yes/No bits behave the same as PLY/PLN.

## 6.34 Custom User Opcode (USRINTR and USRCONT)

These opcodes are part of the open software extensions to the RVP8, which allow custom opcodes to be defined for each major mode of operation. Arguments may be passed into a custom opcode handler as an XARG list. Likewise, an optional array of words returned from that handler will appear after the command executes.



**UserBits** Four additional bits defined by the user to help subdivide the opcode functions if desired.

**CON** If set, then the RVP8's IQ data acquisition thread proceeds continuously while the opcode is executed. If clear, then the IQ stream is interrupted prior to handling the call.

