

3. TTY Nonvolatile Setups (draft)

The RVP8 provides an interactive setup menu that can be accessed either from a serial TTY, or from the host computer interface. Most of the RVP8's operating parameters can be viewed and modified with this menu, and the settings can be saved in non-volatile RAM so that they take effect immediately on power-up. This permits custom trigger patterns, pulsewidth control, matched FIR filter specs, PRF, etc., to be configured by the user in the field.

The TTY menu also gives access to a collection of graphical setup and monitoring procedures that use an ordinary oscilloscope as a synthesized visual display. The burst pulse and receiver waveforms can be examined in detail (both in the time and frequency domain) and the digital FIR filter can be designed interactively to match the characteristics of the transmitted pulse.

3.1 Overview of Setup Procedures

This section describes basic operations within the setup menus such as making TTY connections, entering and exiting the menus, and saving and restoring the configurations.

The setup TTY should be plugged into the modular 6-pin phone jack located at the top edge of the RVP8 board. The electrical interface may be either RS232 or RS423. If the phone jack connection is inconvenient, the terminal may be wired directly to the TIOXMT and TIORCV signals on the P2 96-pin connector. The TTY should be configured for 7-bit or 8-bit data (the MSB is always zeroed), no parity, and either one or two stop bits.

With jumper JP4 in the "AB" position, the interface runs at 9600 baud; in the "BC" position the rate is 1200 baud (factory default), or some other rate set via the menu. Thus, the "AB" setting always makes a reliable 9600 baud connection, even if the alternate rate is accidentally set to a bad or forgotten value. Note: the reliable 9600 baud rate requires that the crystal located at X1 have a frequency of 4.9152MHz.

3.1.1 Initial Entry and Help List

The interactive setup menu is invoked by pressing the Escape key on the TTY. If that key can not be found on the keyboard, you can sometimes use Control "[" to generate the ESC code. The RVP8 then responds with the following banner and command prompt.

```
      SIGMET Incorporated, USA  
RVP8 Digital IF Signal Processor Rev.A/01  
-----  
RVP8>
```

The banner identifies the RVP8 product, and gives the hardware version of the board (e.g., Rev.A) and software version (e.g., 01). This information is important whenever RVP8 support is required, and it is also repeated in the printout of the "V" command (See below).

The "Q" command is used to exit from the menus and to restart the RVP8 with the (possibly changed) set of current values. It is important to quit from the menus before attempting to resume normal RVP8 operation. Portions of the RVP8 command interpreter remain running while the menus are active (so that the TTYOP command works properly), but the processor as a whole will not function until the menus are exited.

From the command prompt, typing “**help**” or “?” gives the following list of available commands.

Command List:

F: Use Factory Defaults
S: Save Current Settings
R: Restore Saved Settings
M: Modify/View Current Settings
 Mb - Burst Pulse and AFC
 Mc - Board Configuration
 Mf - Clutter Filters
 Mp - Processing Options
 Mt<n> - Trigger/Timing <for PW n>
 Mz - Transmitter Phase Control
 M+ - Debug Options
P: Plot with Oscilloscope
 Pb - Burst Pulse Timing
 Ps - Burst Spectra and AFC
 Pr - Receiver Waveforms
 P+ - Visual Test Pattern
V: View Jumpers and Status
?: Cmd list (?? Settings list)
***:** Reboot <Max Slaves> <+>
~: Swap Burst/IF Inputs on IFD
Q: Quit

3.1.2 Factory, Saved, and Current Settings

The *current settings* are the collection of setup values with which the RVP8 is presently operating; the *saved settings* are the collection of values stored in non-volatile RAM. The saved settings are restored (made current) each time the RVP8 is powered up. The “**S**” command saves the current settings into the non-volatile RAM, and the “**R**” command restores those non-volatile values so that they become the current settings. The “**F**” command initializes the current settings with factory default values. Thus, “**F**” followed by “**S**” saves factory defaults in non-volatile RAM, so that the RVP8 powers up in its original configuration as shipped.

The RVP8 retains all of its saved settings when new ROM upgrades are installed; the new version of code will automatically use all of the previous saved values. However, if the RVP8 detects that the new release requires a setup parameter that did not exist in the previous release, then a factory default value will automatically be filled in for that parameter. A warning is printed whenever this occurs (See also, Section 3.1.4).

There is also support for intermediate minor releases of RVP8 code. Each ROM has a major version number (the one that it always had), plus a minor version number for intermediate “unofficial” releases. The minor number starts from zero at the time of each “official” release, and then increments until the next “official” release. The RVP8 includes the minor release

number (if it is not zero) in the printout of the "V" command. Likewise, the minor release number of the code that last saved the nonvolatile RAM is also shown. This is an improvement over having to check the date of the code to determine which minor release was running.

Note that the RVP8 does not actually begin using the current settings until after the "Q" command is entered, so that the processor exits the TTY setup mode and returns to normal operation.

3.1.3 Processor Reset Command

The "*" command may be used to reset the signal processor from the TTY. This can be handy when the other methods of reset (power-up, parallel interface reset signal, or SCSI bus reset) can not easily be done. The command is robust in that pressing the Escape key followed by "*", followed by two Returns, always resets the RVP8. There are certain wait conditions from which a TTY ESC does not immediately enter the setup monitor. However, the above four-key sequence always forces a full reset.

The RVP8 diagnostics can run in a continuous loop that is useful during production burn-in testing. In this mode the complete set of powerup tests is repeated approximately once per second. The green LEDs on the RVP8/Main and RVP8/AUX boards will blink on each run as a progress indicator. All red LEDs will initially be on, but each will begin to blink if any diagnostic ever fails on that board. A line of text is also printed to the setup TTY to show the progress of the tests and a summary of any errors.

The RVP8's Perpetual Diagnostic Loop maintains a histogram of receiver IF-Input noise levels in 1dB steps from -85dBm to -72dBm. You can view the accumulated noise distribution by typing "N" while the diagnostic loop is running. This feature is intended for use during factory burn-in and testing of RVP8/IFD units.

This special test mode can be started in two ways. One is to powerup the processor with the RVP8/Main I/O jumpers JP17-JP22 in the (somewhat illegal) pattern: JP17:BC, JP18:BC, JP19:AB, JP20:AB, JP21:AB, JP22:AB. This method has the advantage of not requiring a TTY connection. The second method is to reset the processor from the local TTY monitor using the "**+" command. This is the normal reset command, but with a plus sign (debugging) suffix.

3.1.4 V — View Internal Status

The "V" command allows you to view some internal status within the RVP8. This information is available for inspection only, and can not be changed from the TTY. The view listing begins with the banner:

Jumpers and Internal Status

and then prints the following lines:

Rev.B board, ROM V14.12 from Mon Jul 12 19:29:07 1999

This line shows the revision level of the RVP8 board, the ROM code version, and the date and time that this release was compiled. This lets you know the age of the release, even if the release notes have been misplaced. The date can also be helpful in keeping track of "unofficial" interim releases.

Values were last saved using ROM version V14

This line tells which version of RVP8 code was the last to write into the non-volatile RAM. It is printed only if that last version was different from the ROM version that is currently running. The information is included so that a “smart upgrade” can often be done, i.e., values that did not exist in the prior release can be filled in with a guess that is better than merely taking the factory default.

Warning: 3 automatic defaults were inserted.

This warning will appear (accompanied by a beep) if one or more automatic factory defaults were required when the non-volatile RAM was last restored. It is likely that these automatic defaults will be acceptable operating values; but it would be wise to check the release notes to see what new parameters were added, and to decide on their proper settings. The warning will disappear once the **S** command is issued. This is because the missing saved slots are then filled in with valid values.

Diagnostics: PASS Slave DSP Count: 3

If errors were detected by the powerup diagnostics then an error bitmask will be shown on the first line. The word “PASS” indicates that no errors were detected. The slave DSP count is also shown, which is the number of processors that were detected during the powerup sequence (and which will be used during subsequent processing). The RVP8 main board has three slave DSPs, and the each RVP8/AUX board supplies ten more. Up to two RVP8/AUX boards may be attached at the same time (23 slave DSPs total) for *extremely* intensive processing applications.

An itemized list (consisting of bit pattern and text) is printed whenever any of the powerup diagnostics fail. The possible messages that might appear are:

- 0x00000001 : No fiber downlink signal detected
- 0x00000002 : 16-Bit AFC level read/write
- 0x00000004 : IF Receiver reset request not sent
- 0x00000008 : I/O FIFO full before 4096 writes
- 0x00000010 : I/O FIFO not full after 4096 writes
- 0x00000020 : Transmit phase latch bits
- 0x00000040 : Downlink local counter test
- 0x00000080 : Receiver status bits & switches
- 0x00000100 : Test byte pattern from receiver
- 0x00000200 : Test word pattern from receiver
- 0x00000400 : Non-Volatile RAM 0x00 and 0xFF flags
- 0x00000800 : UART read/write check
- 0x00001000 : External RAM check
- 0x00002000 : SCSI controller chip error
- 0x00004000 : Range mask RAM and addressing
- 0x00008000 : I&Q FIFO interrupt & trigger flags
- 0x00010000 : I&Q FIFO data bits
- 0x00020000 : FIR processing of ramp pattern
- 0x00040000 : Boot words not accepted by first slave
- 0x00080000 : No reply slave DSP count
- 0x00100000 : Invalid count of slave DSPs
- 0x00200000 : Global communication port tests

0x00400000 : Internal tests failed on some slave
0x00800000 : Trigger Generator RAM and addressing
0x01000000 : Excessive coax/fiber round trip jitter
0x02000000 : No sync found in round trip test
0x04000000 : Internal error in compile/link

Coax/Fiber/Pipeline Delay: 0.624 usec (Stdev: 0.014 usec)

During bootup the RVP8 measures the round trip delay along 1) the coax uplink to the receiver module, 2) the pipeline delays within the receiver module, 3) the optical fiber downlink to the main board, and 4) pipeline delays in the data decoding hardware. The time shown is accurate to within 14ns, and is used internally to insure that the absolute calibration of trigger and burst pulse timing remains unaffected by the distance between the main board and the receiver module. You may freely splice any lengths of coax and fiber without affecting the calibrations; the delay time will change, but the trigger and burst calibrations will remain constant.

The standard deviation of the measured delay is also shown. If the coax uplink and fiber downlink cables are run properly this variation should be less than the period of the acquisition clock, e.g., 0.028 μ sec for the standard 35.975MHz rate. Larger errors may indicate a problem in the cabling. A diagnostic error bit is set if the error is greater than two acquisition clock periods.

IFD:Okay, Burst Pwr:-48.6 dBm, Freq:35.975 MHz

This line summarizes the receiver status and Burst input signal parameters. The status may show:

Okay RVP8/IFD and connecting cables are all working properly
NoFiber Problem in DownLink fiber cable from RVP8/IFD \rightarrow RVP8/Main
UpErr Problem in UpLink COAX cable from RVP8/Main \rightarrow RVP8/IFD
NoPLL RVP8/IFD PLL is not locked to external user-supplied clock reference
DiagSW RVP8/IFD test switches are not in their normal operating position

Reset by: Software Up-time: 0-days 00:49:22

This line lists the origin of the last processor reset, as well as the total time that has elapsed since that reset occurred. The running time is given in days, followed by *hours : minutes : seconds*. The timer wraps around after approximately 180-days of continuous operation. The cause of the last reset will be one of the following:

- | | |
|--------------------------------|--------------------------------|
| 1) Power-Up | 2) External RESET line |
| 3) SCSI Bus Reset | 4) RESET OpCode with "Pwr" bit |
| 5) RESET OpCode with "Rst" bit | 6) RESET OpCode with "Dig" bit |
| 7) BOOT OpCode | 8) Internal Watchdog |
| 9) TTY "*" command | 10) IFD Power Sequencing |
| 11) Burn-In Self Tests | |

3.1.5 Burst-In / IF-In Swap Command

The "~" command swaps the Burst and IF inputs at the IFD. Requests to toggle the state are made from the top level as follows:

```
RVP8> ~  
IFD Burst/IF Inputs are: SWAPPED  
RVP8> ~  
IFD Burst/IF Inputs are: NORMAL
```

The selection remains in effect for the duration of the setup session, but then returns to NORMAL upon exiting the TTY monitor. The “~” command is very handy because it allows the **Pb**, **Pr**, and **Ps** plotting commands to easily run with one input or the other. Here are two examples of how this might be useful.

- When checking the range alignment on a Klystron system, the **Pb** plot can not be used in the usual way to center the Tx burst because a continuous-wave COHO (rather than a burst pulse) is typically used as the phase reference in these systems. However, if you swap the Burst and IF inputs, you can then use the **Pb** command to view and center the received leakage of the Tx pulse, and thus locate range zero.
- When setting up the AFC loop, you can use your RF signal generator to simulate the transmitter's frequency, and then run the loop with swapped RVP8/IFD inputs. The AFC servo will then hunt and follow the siggen frequency supplied via the receiver. You can then make step changes in that frequency to verify that the loop responds properly.

Note that the same input swapping function is also available via the RVP8/IFD toggle switches. However, those switches may be located far away from the operator's terminal; hence, the command interface is still a valuable addition. The “~” command can only be used with the new Rev.D RVP8/IFD; the command is unimplemented, and will not even show up in the “Help” list, when earlier receivers are connected.

3.2 Host Computer I/O Debugging

The RVP8 supports two very powerful monitoring functions that are helpful in debugging the I/O interface to the host computer. One examines the physical layer of the interface, i.e., the electrical handshake and data lines themselves. The other examines the application layer, i.e., the 16-bit opcodes and data that define the RVP8's application programming interface.

3.2.1 Physical-Level I/O Examiner

The RVP8 has TTY support for debugging the physical level of the host computer's SCSI or Parallel interface. The "X" (eXamine) command allows you to watch all incoming 16-bit words as they arrive from the host computer. In addition, you may also send 16-bit words back the other way. The "X" command is only available from the RS232 hardware TTY interface; it can not (obviously) be used via chat mode over the same I/O interface that it trying to examine. As such, the "X" command will not even be listed in the RVP8's top level help menu during a chat mode session.

While the "X" command is running, any words that arrive from the computer will immediately be printed in hex format, along with an "address" (word counter, starting from zero) at the start of each line. Meanwhile, the "W" subcommand can be used to write individual words back to the computer, and the "Q" subcommand will exit the I/O examiner entirely.



Note: When the "X" command is running, the RVP8 does not interpret the incoming 16-bit words as commands and arguments. Data sent to the RVP8 are discarded after being printed; and output from the RVP8 will occur only if the "W" subcommand is manually used. The "X" command is intended to debug the physical layer of the computer interface in a very controlled manner.

The following dialog was captured in response to the host computer writing 100, 200, 300 (decimal) to the RVP8. The "W" subcommand was then used twice to output a 0x4000 and 0x8000 from the RVP8, and the computer then sent the values 1, 2, 3, 4, 5.

```
RVP8> X
Host Computer I/O Debug Monitor
-----
  Q: Exit the monitor
  W: Output a word to the computer

0x0000:  0x0064 0x00C8 0x012C
Output Word : 0x4000
Output Word : 0x8000
0x0003:  0x0001 0x0002 0x0003 0x0004  0x0005
```

3.2.2 Application-Level I/O Examiner

The RVP8 has TTY support for debugging the application level of the host computer's SCSI or Parallel interface. The Real Time TTY Monitor (RTM, see Section 3.3.7) can be configured to expose the computer's complete I/O stream while the RVP8 is running and processing commands in its normal manner. Because of the enormous amount of TTY output that can be

generated by this option, all other RTM selections are disabled whenever host computer I/O is being monitored. Also, those other RTM selections would interfere with the multi-line formatting of the I/O text.

The TTY printout shows incoming opcodes called out by name, and subsequent input and output words formatted into a table. The data are printed in Hex, twelve words per line, and include a word offset (origin zero) at the start of each line. The offset is reset to zero at the start of each new input or output sequence.

Lines of data that are repeats of identical values will be skipped with a “...” indication. This shortens and simplifies the printout; but more importantly, it reduces TTY overhead so that the processor is less I/O bound. Also for this reason, the “0x” Hex prefix is omitted during the possibly lengthy printing of the data word tables.

i **Note: As with all other Real Time TTY Monitor (RTM) functions, the RVP8 remains completely functional while host computer I/O is being monitored. However, unlike all other RTM functions, the I/O monitor will stall the main processor whenever the TTY becomes I/O bound; and the performance of the RVP8 will be degraded, perhaps severely. It is recommended that you configure the TTY for 38.4-KBaud to minimize the serial I/O delays.**

The following sample transactions were captured in response to starting the IRIS/Open ZAUTO utility. An I/O RESET and diagnostic OTEST are first performed. The pulse width selection bits and maximum trigger rates are then set with PWINFO, and angle sync is disabled with LSYNC. The header words for processed data are decided using CFGHDR, operational parameters are loaded with SOPRM, and final RVP8 parameters are read back with GPARM. Finally, the trigger rate is set using SETPWF, and a dummy range mask consisting of a single bin is setup with LRMSK.

```

Opcode 0x008C (RESET)
Opcode 0x0004 (OTEST)
Output Words
  0: 0001 0002 0004 0008 0010 0020 0040 0080 0100 0200 0400 0800
 12: 1000 2000 4000 8000
Opcode 0x000F (PWINFO)
Input Words
  0: 8421 012C 0BB8 0FA0 1F40
Opcode 0x0011 (LSYNC)
Opcode 0x005F (CFGHDR)
Input Words
  0: 0001 0000
Opcode 0x0002 (SOPRM)
Input Words
  0: 0019 000F 07AE 0008 FE70 0080 00A0 0000 0003 000A AAAA 8888
 12: C0C0 C000 0000 0000 0000 AAAA 0000 2710
Opcode 0x0009 (GPARM)
Output Words
  0: 1200 0001 0960 FFFF FFFF 0D5B 0000 0000 0000 4284 0000 0000
 12: 0019 743D 0007 0000 0000 230B 0032 5DC0 0BB8 1770 1D4C 2EE0
 24: 8421 0000 2EE0 2EE0 0960 0960 000F 07AE 0008 FE70 0080 00A0
 36: 0000 0000 0000 0000 0000 0000 0001 000E 0000 000E 0000 0D5B
 48: 8000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
 60: 0000 0000 0000 0000
    
```



```
Opcode 0x0010 (SETPWF)
Input Words
  0: 2EE0
Opcode 0x0001 (LRMSK)
Input Words
  0: 0001 0000 0000 0000  0000 0000 0000 0000  0000 0000 0000 0000
 12: ...
 504: 0000 0000 0000 0000  0000 0000 0000 0000
```

This RTM option to monitor computer I/O is automatically disabled at powerup, and therefore can not be saved permanently. This is to avoid confusing situations in which the monitor is accidentally left running — the RVP8 would appear to be working, but at a puzzling level of degraded performance.

3.3 View/Modify Dialogs

The **M** command may be used to view, and optionally to modify, all of the current settings. The current value of each parameter is printed on the screen, and the TTY pauses for input at the end of the line. Pressing Return advances to the next parameter, leaving the present one unchanged. You may also type **U** to move back up in the list, and **Q** to exit from the list at any time.

Typing a numeric or YES/NO response (as appropriate to the parameter) changes the parameter's value, and displays the line again with the new value. All numbers are entered in base ten, and may include a decimal point and minus sign. In some cases, several parameters are displayed on one line, in which case, as many parameters are changed as there are new values entered. In all cases, the numbers are checked to be within reasonable bounds, and an error message (listing those bounds) is printed if the limits are exceeded. Note that changes to the settings (generally) do not take effect until after the **Q** command is typed, at which point the RVP8 exits the local TTY menu and resumes its normal processing operations.

Since the number of setup questions is large, follow the **M** command with a second letter to select the subcategory, i.e., **Mb** (Burst Pulse and AFC), **Mc** (Board Configuration), **Mf** (Clutter Filters), **Mp** (Processing Options), **Mt** (Triggers and Timing), **Mz** (Transmitter Phase Control), **M*** (Stand-alone Settings) or **M+** (Debug Options). The **M** command by itself prints the entire set of questions so that you can make a hard copy.

The **M** command always works from the current parameter values, not from the saved values in non-volatile RAM. If the host computer has modified some of the current values, then you will see these changes as you skip through the setup list. However, typing **S** at that point would save all of the current settings and would, perhaps, make many changes to the original non-volatile settings. In general, to make an incremental change to the saved settings, first type **R** to restore all of the saved values, then use **M** to make the changes starting from that point, and **S** to save the new values.

A listing of the parameters that can be viewed and modified with the **M** command is detailed in the following subsections. In each case, the line of text is shown exactly as it appears on the TTY with the factory default settings. A definition of each parameter is given and, if applicable, the lower and upper numeric bounds are shown.

3.3.1 Mc — Board Configuration

This set of commands configure general properties of the RVP8/IFD and RVP8/Main boards.

Acquisition clock: 35.9751 MHz

This is the frequency of the oscillator at U5 in the IF receiver module. Except for custom receivers, this will always be 35.9751 MHz; which gives a fundamental sample spacing of 1/240 km (approximately 4.17 meters).

Limits: 33.33 to 41.67 MHz

Dual simultaneous receivers are being used: NO

Answer this question "Yes" if the RVP8 will be processing simultaneous signals from two separate receivers. Answering "No" will revert to normal operation with just a single receiver.

Dual-LNA/Rcvr single-channel switched mode: NO

For dual-polarization single-receiver systems, this question decides whether you have a single LNA and IF-Amplifier that switches between H&V (the typical case); or two separate receivers, each hard wired to H and V, with switching performed after the IF amplifiers. The question affects how noise levels are measured and applied to the data.

Synthesize LOG video output waveform: YES

Upper 100.0 dB will occupy 85.0% of voltage span

Force freerunning video mode: NO

Plot data from secondary receiver: NO

The RVP8 supports the option of sourcing a LOG Video analog output signal from the backpanel of the main chassis. There are two ways that this signal can be configured:

- ***Self-Triggering, Free-Running Mode***

This is the default mode that is available on all RVP8 boards. The output signal is periodic at approximately the PRF of the radar, but is free-running, i.e., not actually synchronized with the radar trigger. A synthetic 1.0 μ sec wide, full scale, "trigger" pulse is embedded at the zero-range start of each LOG Video waveform. This marker can easily trigger an oscilloscope if the scope's trigger level is set just below the maximum LOG Video voltage level.

- ***Waveform Locked to Radar Trigger***

This mode requires a (one-wire) hardware modification to the Rev.B RVP8/Main board. The LOG Video waveform then becomes locked to the radar trigger, so that the LOG signal can be displayed on any device that already receives the radar trigger.

In either case, the LOG Video output signal is unipolar, ranging from approximately 0.0V to 3.0V. It is active during all data processing modes that the host computer might request, as well as during the idle time between scans. The signal is absent (zero), however, during the short intervals of time that the RVP8 is being reconfigured by the host computer, or when the RVP8's local TTY setups are being used.

The time resolution of the synthesized LOG Video signal is fixed at 1.0 μ sec per bin. This is independent of the actual range resolution of the FIR matched filter. Whatever (I,Q) data are actually being computed by the FIR front end are selected for a nearest fit to each 1.0 μ sec synthetic output cell. The maximum number of incoming FIR range bins that can be selected among is 5460. Thus, for example, the maximum range of the LOG Video signal would be 682km when the FIR range resolution is 125-meters.

Answer the first question "Yes" if you would like the RVP8/Main board to synthesize and drive the LOG Video output signal. The cost of doing this is that one of the "slave" DSP chips will be removed from the normal Doppler processing chain, and dedicated to the task of LOG Video generation. On a single-board system, the

three available slave DSPs would be reduced to two; whereas on a dual-board system, the 13 available DSPs would be reduced to 12. Obviously, the percentage penalty is less in a larger system.

The second question decides how the overall dynamic range of the receiver will fit into the 12-bit unipolar output voltage span of the DAC that produces the LOG Video waveform. The default setting calls for the upper 100dB of dynamic range to occupy 85% of the output voltage span. This means that the strongest IF input signal would produce 85% of the maximum DAC voltage (approximately 2.55 Volts); 50dB down would be 42.5%, and 100dB down would be 0%, i.e., zero volts.

If you are using a self-triggering LOG Video waveform, then the 15% of headroom provided by the default settings leaves room for the embedded trigger pulse. However, if your RVP8 has the hardware modification required to synchronize the LOG Video to the system trigger, then the full 100% of the DAC voltage span can freely be used. The third setup question can be used to force freerunning mode on an RVP8 that has the hardware modification. This question is included mostly for testing purposes.

The last question only appears in dual-receiver mode. Answer "Yes" if you would like the LOG video analog output signal to be based on the data from the secondary receiver rather than from the primary receiver.

Scope plots- Holdoff ratio: 0.50, Stroke: 1000.0 usec

The oscilloscope plotting commands are described in Chapter 4. This question allows you to vary the amount of holdoff time that is inserted between each drawing stroke, as well as the stroke length itself. Try increasing the holdoff if your scope is not triggering reliably. Longer holdoffs make it easier for the scope to find the initial trigger point, but may introduce visible flicker. To reduce flicker, try decreasing the stroke duration from its default value of 1000 microseconds.

Limits: Holdoff 0.05 to 5.00, Stroke 100 to 10000 μ sec.

PWINFO command enabled: No

The "Pulsewidth Information" user interface command can be disabled, thus further protecting the radar against inappropriate combinations of pulsewidth and PRF. This is a more safe setting in general, and is even more important when DPRT triggers are being generated. It can also be useful when running user code that is not yet fully debugged.

TRIGWF command enabled: NO

The "Trigger Waveform" user interface command can be disabled if you want to prevent the host computer from overwriting the RVP8's stored trigger specifications. This is the default setting, based on the assumption that the built-in plotting commands would be used to configure the triggers. Answering "YES" will allow new waveforms to be loaded from the host computer.

RVP7 Emulation: No

The RVP8 implements a reasonably precise emulation of the RVP7 command set. This mode is useful because it allows an RVP8 to be plugged directly into a software system that used to run with an RVP7. All of the configuration steps that are new and

unique to the RVP8 can be handled by the local TTY and Scope setups, thus making no demands on the user's system code for support. Answer this question "YES" for maximum compatibility with old driver software. However, if you are running IRIS version 6.11 or higher, then answer "NO" to enable using new RVP8 features as they are developed.

The RVP8 returns a version number of 35 when the processor is running in RVP7 compatibility mode. This fudged value will appear in the SCSI Inquiry Command reply, and in the GPARM parameter packet. Elsewhere, the correct RVP8 ROM version number will always appear. The reason for doing this is so that the RVP8 appears (to the host computer) to be a modern RVP7 with all of the latest opcodes and features.

3.3.2 Mp — Processing Options

Major Mode- 0:User, 1:PPP, 2:FFT : 0

The top level RVP8 operating modes are described in the documentation of SOPRM command word #9. This question allows you to use the mode that has been selected by that command, or to force the use of a particular mode.

Window- 0:User, 1:Rect, 2:Hamming, 3:Blackman : 0

Whenever power spectra are computed by the RVP8, the time series data are multiplied by a (real) window prior to computation of the Fourier Transform. You may use whichever window has been selected via SOPRM word #10, or force a particular window to be used.

R2 Processing- 0:Never, 1:User, 2:Always : 1

Controls R0/R1 versus R0/R1/R2 processing. Selecting "0" unconditionally disables the R2 algorithms, regardless of what the host computer requests in the SOPRM command. Likewise, selecting "2" unconditionally enables R2 processing. These choices allow the RVP8 to run one way or the other without having to rewrite the user code. This is useful for compatibility with existing applications.

Clutter Microsuppression- 0:Never, 1:User, 2:Always : 1

Controls whether individual "cluttery" bins are rejected prior to being averaged in range. Same interpretation of cases as for "R2 Processing" above.

2D Final Speckle/Unfold - 0:Never, 1:User, 2:Always : 1

The Doppler parameter modes (PPP, FFT, etc) include an optional 3x3 interpolation and speckle removal filter that is applied to the final output rays. This 2-dimensional filter examines three adjacent range bins from three successive rays in order to assign a value to the center point. Thus, for each output point, its eight neighboring bins in range and time are available to the filter. Only the *dBZ*, *DBT*, *Vel*, and *Width* data are candidates for this filtering step; all other parameters are processed using the normal 1-dimensional (three bins in range) speckle remover. See Section 5.3.3 for more details.

Unfold Velocity (Vh-Vl) - 0:Never, 1:User, 2:Always : 0

This question allows you to choose whether the RVP8 will unfold velocities using a simple ($V_{high} - V_{low}$) algorithm, rather than the standard algorithm described in Section 5.6. Bit-11 of SOPPRM word #10 is the host computer's interface to this function when the "1:User" case is selected (See Section 6.3).



Note: This setup question is included for research customers only. The standard unfolding algorithm should still be used in all operational systems because of its lower variance. For this reason, the factory default value of this parameter is "0:Never".

Process w/ custom trigs - 0:Never, 1:User, 2:Always : 0

This question allows you to choose whether the RVP8 will attempt to run its standard processing algorithms even when a custom trigger pattern has been selected via the SETPWF command. Generally it does not make sense to do this, so the default setting is "0:Never". Bit-12 of OPPRM word #10 is the host computer's interface to this function when the "1:User" case is selected (See Section 6.3).

Minimum freerunning ray holdoff: 100% of dwell

This parameter controls the rate at which the RVP8 processes free-running rays in the FFT, DPRT, and Random Phase modes. This prevents rays from being produced at the full CPU limit or I/O limit of the processor (whichever was slower); which could result in highly overlapping data being output at an unusably fast rate. Note that this behavior will only occur when one of these non-PPP modes is chosen, and is then allowed to run without angle syncing. Such is likely the case for IRIS manual scans or during Passive IRIS mode.

To make these free-running modes more useful, you may establish a minimum holdoff between successive rays, expressed as a percentage of the number of pulses contributing to each ray. Choosing 100% (the default) will produce rays whose input data do not overlap at all, i.e., whose rate will be exactly the PRF divided by the sample size. Choosing 0% will give the unregulated behavior in which no minimum overlap is enforced and rays may be produced very quickly.

Limits: 0 to 100%

Linearized saturation headroom: 4.0 dB

The RVP8 uses a statistical saturation algorithm that estimates the real signal power correctly even when the IF receiver is overdriven (i.e., for input power levels above +4dBm). The algorithm works quite well in extending the headroom above the top end of the A/D converter, although the accuracy decreases as the overdrive becomes more severe. This parameter allows you to place an upper bound on the maximum extrapolation that will ever be applied. Choosing 0dB will disable the algorithm entirely.

Limits: 0 to 6dB

Apply amplitude correction based on Burst/COHO: YES

Time constant of mean amplitude estimator: 70 pulses

The RVP8 can perform pulse-to-pulse amplitude correction of the digital (I,Q) data stream based on the amplitude of the Burst/COHO input. Please see Section 5.1.6 for a complete discussion of this feature.

Limits: 10 to 500 pulses

IFD built-in noise dither source: -57.0dBm

This question will only appear if the processor is attached to a Rev.D RVP8/IFD that includes an out-of-band noise generator to supply dither power for the A/D converters. The available power levels are { Off, -57dBm, -37dBm, -32dBm, -27dBm, -22dBm, -19dBm }. The closest available level to your typed-in value will be used. You can observe the band-limited noise easily in the **Pr** plot to confirm its amplitude and spectral properties.

For standard operation, we recommend running at -57dBm. The problem higher levels of dither level is that, for certain choices of (I,Q) FIR filter, the stopband of the filter may not give enough attenuation to preserve the RVP8/IFD's inherent noise level. For example, the factory default 1MHz bandwidth Hamming filter has a stopband attenuation near DC of approximately 43dB. You can see this graphically at the right edge of the **Ps** menu. The in-band contribution of dither power is therefore approximately $(-37\text{dBm}) - 43\text{dB} = -80\text{dBm}$, which exceeds the A/D converter's 1MHz bandwidth noise of -81.5dBm.

TAG bits to invert	AZ:0000	EL:0000
TAG scale factors	AZ:1.0000	EL:1.0000
TAG offsets (degrees)	AZ:0.00	EL:0.00

The incoming TAG input bits may be selectively inverted via each of the 16-bit words. The values are displayed in Hex. Setting a bit will cause the corresponding AZ (bits 0-15) or EL (bits 16-31) lines to be inverted. Note that the SOPRM command also specifies TAG bits to invert. Both specifications are XOR'ed together to yield the net inversion for each TAG line.

The overall operations are performed in the order listed. Incoming bits are first inverted according to the two 16-bit XOR masks. This yields an unsigned 16-bit integer value which is then multiplied by the signed scale factor. The result is interpreted as a 16-bit binary angle (in the low sixteen bits), to which the offset angle is finally added.

As an example, suppose that the elevation angle input to the RVP8 was in an awkward form such as unsigned integer tenths of degrees, i.e., 0x0000 for zero degrees, 0x000a for one degree, 0x0e06 for minus one degree, etc. If we apply a scale factor of $65536/3600 = 18.2044$ to these units, we will get 16-bit binary angles in the standard format. If we further suppose that the input angle rotated "backwards", we could take care of this too using a multiplier of -18.2044.

Interference Filter- 0:None, Alg.1, Alg.2, Alg.3: 1
Threshold parameter C1: 10.00 dB
Threshold parameter C2: 12.00 dB

The RVP8 can optionally apply an interference filter to remove impulsive-type noise from the demodulated (I,Q) data stream. See Section 5.1.4 for a complete description of this family of algorithms.

Polarization Params - Filtered:YES NoiseCorrected:YES
PhiDP - Negate: NO , Offset:0.0 deg
KDP - Length: 5.00 km
T/Z/V/W computed from: H-Xmt:YES V-Xmt:YES
T/Z/V/W computed from: Co-Rcv:YES Cx-Rcv:NO

The first question decides whether all polarization parameters will be computed from filtered or unfiltered data, and whether noise correction will be applied to the power measurements.

The second and third questions define the sign and offset corrections for ϕ and the length scale for *KDP*.

The fourth and fifth questions control how the standard parameters (Total Reflectivity, Corrected Reflectivity, Velocity, and Width) are computed in a multiple polarization system. Answering *YES* to *H-Xmt* and/or *V-Xmt* means that data from those transmit polarizations should be used whenever there is more than one choice available. Thus, these selections only apply to the Alternating and Simultaneous transmit modes. Likewise, answering *YES* to *Co-Rcv* and/or *Cx-Rcv* means to use the received data from the co-channel or cross-channel. The receiver question will only appear when dual simultaneous receivers have been configured.

A typical installation might use *H-Xmt:YES*, *V-Xmt:YES*, *Co-Rcv:YES*, *Cx-Rcv:NO*. This will compute (T/Z/V/W) from the co-polarized receiver using both H&V transmissions. Including both transmissions will decrease the variance of (T/Z/V/W); although some researchers prefer excluding *V-Xmt* because that is more standard in the literature. Also, if your polarizations are such that the main power is returned on the cross channel, then you will probably want *Co-Rcv:NO* and *Cx-Rcv:YES*.

DualRx - Sum H+V Time Series: NO

In dual-receiver systems, you may choose whether the (H+V) time series data consist of the sum of the “H” and “V” samples or the concatenation of half the “H” samples followed by half the “V” samples. The later is more useful when custom software is being used to analyze the data from the two separate receive channels.

3.3.3 Mf — Clutter Filters

Doppler Filter Set- 0:40dB, 1:50dB, 2:Saved : 0

The RVP8 has two built-in IIR Doppler clutter filter sets; one set having 40dB of stopband attenuation, and the other having 50dB. This question chooses which set is loaded on powerup.

Spectral Clutter Filters

```

-----
Filter #1 - Type:0 (Fixed)      Width:1      EdgePts:2
Filter #2 - Type:0 (Fixed)      Width:2      EdgePts:2
Filter #3 - Type:0 (Fixed)      Width:3      EdgePts:3
Filter #4 - Type:0 (Fixed)      Width:4      EdgePts:3
Filter #5 - Type:1 (Variable)   Width:1      EdgePts:2      Hunt:2
Filter #6 - Type:1 (Variable)   Width:2      EdgePts:2      Hunt:2
Filter #7 - Type:1 (Variable)   Width:3      EdgePts:3      Hunt:3
  
```

These questions define the heuristic clutter filters that operate on power spectra during the FFT-type major modes. Filter #0 is reserved as “all pass”, and is not redefinable here. For filters #1 through #7, enter a digit to choose the filter type, followed by however many parameters that type requires.

Fixed Width Filters (Type 0)

These are defined by two parameters. The “Width” sets the number of spectral points that are removed around the zero velocity term. A width of one will remove just the DC term; a width of two will remove the DC term plus one point on either side; three will remove DC plus two points on either side, etc. Spectral points are removed by replacing them with a linear interpolating line. The endpoints of this line are determined by taking the minimum of “EdgeMinPts” past the removed interval on each side.

Variable Width, Single Slope (Type 1)

The RVP8 supports variable-width frequency-domain clutter filters. These filters perform the same spectral interpolation as the fixed-width filters, except that their notch width automatically adapts to the clutter. The new filters are characterized by the same *Width* and *EdgePts* parameters in the **Mf** menu, except that the *Width* is now interpreted as a minimum width. An additional parameter *Hunt* allows you to choose how far to extend the notch beyond *Width* in order to capture all of the clutter power. Setting *Hunt=0* effectively converts a variable-width filter back into a fixed-width filter.

The algorithm for extending the notch width is based on the slope of adjacent spectral points. Beginning (*Width-1*) points away from zero, the filter is extended in each direction as long as the power continues to decrease in that direction, up to adding a maximum of *Hunt* additional points. If you have been running with a fixed *Width=3* filter, you might try experimenting with a variable *Width=2* and *Hunt=1* filter. Perhaps the original fixed width was actually failing at times, but you were reluctant to increase it just to cover those rare cases. In that case, try selecting a variable *Width=2* and *Hunt=2* filter as an alternative. In general, make your variable filters “wider” by increasing *Hunt* rather than increasing *Width*. This will preserve more flexibility in how they can adapt to whatever clutter is present.

Residual clutter LOG noise margin: 0.15 dB/dB

Whenever a clutter correction is applied to the reflectivity data, the LOG noise threshold needs to be increased slightly in order to continue to provide reliable qualification of the corrected values. The reason for this is that the uncertainty in the corrected reflectivity becomes greater after the clutter is subtracted away.

For example, if we observe 20dB of total power above receiver noise, and then apply a clutter correction of 19dB, we are left with an apparent weather signal power of +1dB above noise. However, the uncertainty of this +1dB residual signal is much greater than that of a pure weather target at the same +1dB signal level.

The “Residual Clutter LOG Noise Margin” allows you to increase the LOG noise threshold in response to increasing clutter power. In the previous example, and with the default setting of 0.15dB/dB, the LOG threshold would be increased by $19 \times 0.15 = 2.85$ dB. This helps eliminate noisy speckles from the corrected reflectivity data.

Whitening Parameters

Noise threshold for replacing a point: 1.20
Replacement value multiplier: 0.5000
SNR in tails, for determining width: 0.25

These questions control the adaptive whitening filter that is used by the Random Phase processing algorithms. A spectral point will be whitened if the ratio of its power to the noise power exceeds the “Noise threshold for replacing a point.” The whitened point will consist of a complex value whose ARG is identical to that of the original point, and whose MAG is the product of the noise level with the “Replacement value multiplier” term. The nominal spectral width of the whitened region is a function of the power and width of the coherent signal, and the noise level. Assuming a Gaussian model, the “SNR in tails...” value is the ratio of the coherent power in the tails of the distribution to the noise level.

RPhase SQI Threshold Slope:0.50 Offset:-0.05

The two values in this question define a secondary SQI threshold that is used to qualify the LOG data during Random Phase processing. The secondary SQI level is computed by multiplying the primary user-supplied SQI threshold by the SLOPE, and adding the OFFSET. See also Section 5.9.3.

Limits: SLOPE: 0.0 to 2.0, OFFSET -2.0 to 1.0

3.3.4 Mt — General Trigger Setups

These questions are accessed by typing “Mt” with no additional arguments. They configure general properties of the RVP8 trigger generator

Pulse Repetition Frequency: 500.00 Hz

This is the Pulse Repetition Frequency of the internal trigger generator.

Limits: 50 to 6000Hz.

Transmit pulse width: 0

Limits: 0 to 3

Use external pretrigger: NO

PreTrigger active on rising edge: YES

PreTrigger fires the transmitter directly: NO

When an external pretrigger is applied to the TRIGIN input of the RVP8, either the rising or falling edge of that signal initiates operation. This decision also affects which signal edge becomes the reference point for the pretrigger delay times given in the “Mt<n>” section.

Answer the second sub-question according to whether the radar transmitter is directly fired by the the external pretrigger, rather than by one of the RVP8's trigger outputs. In other words, answer "YES" if the transmitter would continue running fine even if the RVP8 TRIGIN signal were removed. This information is used by the "L" and "R" subcommands of the "Pb" plotting command, i.e., when slewing left and right to find the burst pulse, the pretrigger delay will be affected rather than the start times of the six output triggers.

2-way (Tx+Rx) total waveguide length: 0 meters

Use this question to compensate for the offset in range that is due to the length of waveguide connecting the transmitter, antenna, and receiver. You should specify the total 2-way length of waveguide, i.e., the span from transmitter to antenna, plus the span from antenna to receiver. The RVP8 range selection will compensate for the additional waveguide length to within plus-or-minus half a bin, and works properly at all range resolutions.

POLAR0 is high for vertical polarization : NO

POLAR1 is high for vertical polarization : NO

These questions define the logical sense of the two polarization control signals POLAR0 and POLAR1. In a dual-polarization radar POLAR0 should be used to select one of two possible states (nominally horizontal and vertical, but any other polarization pair may also be used). The control signal will either remain at a fixed level, or will alternate from pulse to pulse with a selectable transition point (See Section 3.3.5). POLAR1 is identical to POLAR0, but may be configured with a different polarity and switch point. This second signal could be used if the radar's polarization switch required more than one control line transition when changing states.

Quantize trigger PRT to ((1 x AQ) + 0) clocks

It is possible to control the exact quantization of the PRT of the internal trigger generator. Normally the trigger PRT is chosen as the closest multiple of AQ (the acquisition clock period) that approximates the requested period. This question allows the possible PRT's to be constrained to $((N \times AQ) + M)$ clock cycles. This feature can be useful for synchronous receiver systems in which the trigger period must be some exact multiple of the COHO period.

Blank output triggers according to TAG#0 : NO

Blank when TAG input is high : NO

Blank triggers 1:YES 2:YES 3:YES 4:YES 5:YES 6:YES

These questions control trigger blanking based on the TAG0 input line. You first select whether the trigger blanking feature is enabled; and then optionally choose the polarity of TAG0 that will result in blanking, and which subset of the six user definable triggers are to be blanked.

Blank output triggers during noise measurement : NO

The RVP8 can inhibit the subset of blankable trigger lines whenever a noise measurement is taken. This will be forced whenever trigger blanking (based on TAG0) is enabled, but it can also be selected in general via this question. Since noise triggers must be blanked whenever trigger blanking is enabled, this question only appears if trigger blanking is disabled.

This question permits the state of the triggers during noise measurements to be consistent and known, regardless of whether the antenna happens to be within a blanked sector; and you have the additional flexibility of choosing blanked noise triggers all the time.

Rx-Fixed Triggers: #1:N #2:N #3:N #4:N #5:N #6:N P0:N P1:N Z:N

You have explicit control over which RVP8 trigger outputs are timed relative to the transmitter pre-fire sequence, versus those which are relative to the actual received target ranges. Triggers in the first category will be moved left/right by the "L/R" keys in the **Pb** plot, and will also be slewed in response to Burst Pulse Tracking. Triggers in the second category remain fixed relative to "receiver range zero", and are not affected by the "L/R" keys or by tracking.

This question specifies which triggers are Tx-relative and which are Rx-relative. Answer with a sequence of "Y" or "N" responses for each of the six trigger lines, for the two polarization control lines, and for the timing of the phase control lines. You should answer *No* for any trigger that is involved with the pre-fire timing of the transmitter. If you enable the Burst Pulse Tracker (Section 5.1.3) you will probably want to assign a *Yes* to some of your triggers so that they remain fixed relative to the burst itself.

It is very helpful to have these two categories of trigger start times. Triggers that fire the transmitter, either directly or indirectly, should all be moved as a group when hunting for the burst pulse and moving it to the center of the FIR window. However, triggers that function as range strobes should be fixed relative to range zero, i.e., the center of that window, and the center of the burst. This distinction becomes important when the transmitter's pre-fire delay drifts with time and temperature.

Replace triggers with alternate waveforms: YES

```
Trigger #1 - 0:Normal, 1-2:Pol0-1, 3-6:PW0-3 : 0
Trigger #2 - 0:Normal, 1-2:Pol0-1, 3-6:PW0-3 : 0
Trigger #3 - 0:Normal, 1-2:Pol0-1, 3-6:PW0-3 : 0
Trigger #4 - 0:Normal, 1-2:Pol0-1, 3-6:PW0-3 : 1
Trigger #5 - 0:Normal, 1-2:Pol0-1, 3-6:PW0-3 : 0
Trigger #6 - 0:Normal, 1-2:Pol0-1, 3-6:PW0-3 : 4
```

These questions make it possible to reassign the waveforms that are driven onto the six user trigger (TRIG1-6) BNC outputs on the backpanel of the RVP8. This makes it easier to adapt the external cabling of the RVP8 so as to make better use of the available BNC connectors and related 15V drivers. You may substitute either of the two polarization control lines or the four pulsewidth control lines in place of any of the six normal triggers.

In the example above, triggers #1, #2, #3, and #5 are all driven with their normal waveforms. However trigger #4 will have a copy of the POLAR0 polarization control line, and trigger #6 will have a copy of the PWBW1 pulsewidth control line. Neither POLAR0 nor PWBW1 themselves are changed by these assignments.

Whenever any of the six user trigger lines is reassigned from its normal setting, the plot of that trigger within the **Pb** command will show a hashed line across the screen. This is a graphical reminder that that trigger has been replaced by some other waveform.

Merge triggers to create composite waveforms: YES

```
Merge Trigger #1 into : #1: #2: #3: #4: #5: #6:
Merge Trigger #2 into : #1: #2: #3: #4: #5: #6:
Merge Trigger #3 into : #1:Y #2: #3: #4: #5: #6:
Merge Trigger #4 into : #1: #2:Y #3: #4: #5: #6:
Merge Trigger #5 into : #1: #2:Y #3: #4: #5: #6:
Merge Trigger #6 into : #1: #2: #3: #4: #5: #6:
```

These questions allow you to merge the six user triggers together; resulting in trigger patterns that can be much more complex. In this example, Trigger #3 will be merged into Trigger #1; Trigger #3 will be unaltered, and Trigger #1 will be the “OR” of itself with Trigger #3. Likewise, Triggers #4 and #5 will be merged into Trigger #2 so that the later will contain three distinct pulses within each PRT. Answer each question with a sequence of up to six “Y” or “N” responses in order to set the merged destinations for each trigger line.

Note that the six triggers are still defined in the usual way in the **Mt<n>** menu, i.e., start time, width, etc. The only change is that you may now combine these individual pulse definitions into a more complex composite output waveform.

3.3.5 Mt<n> — Triggers for Pulsethwidth #n

These questions are accessed by typing “Mt”, with an additional argument giving the pulsethwidth number. They configure specific trigger and FIR bandpass filter properties for the indicated pulsethwidth only.

```
Trigger #1 - Start:    0.00 usec
              #1 - Width:    1.00 usec      High:YES
Trigger #2 - Start:    0.00 usec + ( 0.500000 * PRT )
              #2 - Width:   10.00 usec      High:YES
Trigger #3 - Start:   -3.00 usec
              #3 - Width:    1.00 usec      High:YES
Trigger #4 - Start:   -2.00 usec
              #4 - Width:    1.00 usec      High:YES
Trigger #5 - Start:   -1.00 usec
              #5 - Width:    1.00 usec      High:YES
Trigger #6 - Start:   -5.00 usec + (-0.001000 * PRT )
              #6 - Width:    2.00 usec      High:NO
```

These parameters list the starting times (in microseconds relative to range zero), the widths (in microseconds), and the active sense of each of the six triggers generated by the internal trigger generator. Setting a width to zero inhibits the trigger on that line.

The Start Time can include an additional term consisting of the pulse period times a fractional multiplier between -1.0 and +1.0. This allows you to produce trigger patterns that would not otherwise be possible, e.g., a trigger that occurs half way

between every pair of transmitted pulses, and remains correctly positioned regardless of changes in the PRF. Enter this multiplier as "0" if you do not wish to use this term, and it will be omitted entirely from the printout..

In the above example, Trigger #2 is a 10.0 μ sec active-high pulse whose leading edge occurs precisely halfway between the zero-range of every pair of pulses. Likewise, Trigger #6 is a 2.0 μ sec active-low pulse whose falling edge is nominally 5.0 μ sec prior to range zero, but which is advanced by 1.0 μ sec for every millisecond of trigger period. All other triggers behave normally, and have fixed starting times that do not vary with trigger period.

Some subtleties of these variable start times are:

- The PRT multipliers can only be used in conjunction with the RVP8's internal trigger generator. The PRT-relative start times are completely disabled whenever an external trigger source is chosen from the **Mt** menu.
- When PRT-relative triggers are plotted by the **Pb** command, the active portion of the trigger will be drawn cross-hatched and at a location computed according to the current PRF. The cross-hatching serves as a reminder that the actual location of that trigger may vary from its presently plotted position.
- The PRT multiplier for a given pulse is applied to the interval of time between that pulse and the next one. This distinction is important whenever the RVP8 is generating multiple-PRT triggers, e.g., during DPRT mode, or during Dual-PRF processing. Multipliers from 0.0 to +1.0 are generally safe to use because they shift the trigger into the same pulse period that originally defined it. For example, a start time of $(0.0 \mu\text{sec} + (0.98 * \text{PRT}))$ would position a trigger 98% of the way up to the next range zero. But, if -0.98 were used, and if the period of the previous pulse was shorter than the current one, then that shorter period would become incorrect (longer) as a result of having to fit in the very early trigger.

A small but important detail is built into the algorithm for producing the six user trigger waveforms. It applies whenever a) the trigger period is internally determined, i.e., the external pretrigger input is not being used, and b) the overall span of the six trigger definitions combined does not fit into that period. What happens in this case is that any waveforms that do not fit will be zeroed (not output) so that the desired period is preserved. This means that you can define triggers with large positive start times, and they will pop into existence only when the PRF is low enough to accommodate them.

For example, if Trigger #2 is defined as a 200.0 μ sec pulse starting at +400.0 μ sec, then that trigger would be suppressed if the PRF were 2000Hz, but it would be present at a PRF of 1000Hz. Whenever a trigger does not completely fit within the overall period it is suppressed entirely. Thus, even though the +400.0 μ sec start time is still valid at 2000Hz, the entire 200.0 μ sec pulse would not fit, and so the pulse is eliminated altogether.

Start limits: -5000 to $5000 \mu\text{sec}$. Width limits: 0 to $5000 \mu\text{sec}$.

Maximum number of Pulses/Sec: 2000.0
Maximum instantaneous 'PRF' : 2000.0 (/Sec)

These are the PRF protection limits for this pulsewidth.

The wording of the "Maximum number of Pulses/Sec" question serves as a reminder that the number shown is not only an upper bound on the PRF, but also a duty cycle limit when DPRT mode is enabled.

The "Maximum instantaneous 'PRF'" question allows you to configure the maximum instantaneous rate at which triggers are allowed to occur, i.e., the reciprocal of the minimum time between any two adjacent triggers. This parameter is included so that you can limit the maximum DPRT trigger rate individually for each pulsewidth. Note that the maximum instantaneous PRF can not be set lower than the maximum number of pulses per second.

PRF limits: 50 to 20000Hz.

External pretrigger delay to range zero: 3.00 usec

Range Zero is time at which the signal from a target at zero range would appear at the radar receiver outputs. This parameter adjusts the delay from the active edge of the external trigger to range zero. It is important that this delay be correct when the RVP8 is operating with an external trigger, since the zero range point is a fixed time offset from that trigger. When the transmitter is driven from the internal trigger signals, those signals themselves are adjusted (see Burst Pulse alignment procedures) to accomplish the alignment of range zero.

Limits: 0.1 to 500 μ sec.

Range resolution: 125.00 meters

The range resolution of the RVP8 is determined by the decimation factor of the digital matched FIR filter that computes "I" and "Q". This decimation factor is the ratio of the filter's input and output data rates, and can be any integer from six to sixteen. The Acquisition Clock (See **Mc** Section) sets the input data rate. At its standard frequency of 35.9751MHz, the available range resolutions (in meters) are: 50.0, 58.3, 66.7, 75.0, 83.3, 91.7, 100.0, 108.3, 116.7, 125.0, and 133.3.

The ranges that are selected by the bit mask in the LRMSK command are spaced according to the range resolution that is chosen here. Also, the upper limit on the impulse response length of the matched FIR filter (see below) is constrained by the range resolution. If you choose a range resolution that can not be computed at the present filter length, then a message of the form: "Warning: Impulse response shortened from 72 to 42 taps" will appear.

Limits: 50.0 to 133.3 meters.

FIR-Filter impulse response length: 1.33 usec

The RVP8 computes "I" and "Q" using a digital FIR (Finite Impulse Response) matched filter. The length of that filter (in microseconds) is chosen here. At the standard Acquisition Clock rate of 35.9751MHz, a 1.00 microsecond impulse response corresponds to a filter that is 36 taps long.

The filter length should be based on several considerations:

- It should be at least as long as the transmitted pulsewidth. If it were shorter, then some of the returned energy would be thrown away when “I” and “Q” are computed at each bin. The SNR would be reduced as a result.
- It should be at least as long as the range bin spacing. The goal here is to choose the longest filter that retains statistical independence among successive bins. If the filter length is less than the bin spacing, then no IF samples would be shared among successive bins, and those bins would certainly not be correlated.
- It should be “slightly longer” than either of the above bounds would imply, so that the filter can do a better job of rejecting out-of-band noise and spurious signals. The SNR of weak signals will be improved by doing this.

In practice, a small degree of bin-to-bin correlation is acceptable in exchange for the filter improvements that become possible with a longer impulse response. The FIR coefficients taper off to zero on each end; hence, the power contributed by overlapping edge samples is minimal. SIGMET recommends beginning with an impulse response length of 1.2–1.5 times the pulsewidth or bin spacing, whichever is greater.

The maximum possible filter length is bounded according to the range resolution that has been chosen; a finer bin spacing leaves less time for computing a long filter. For the RVP8 Rev.A processor, the filter length must be less than 2.92 μ sec at 125-meter resolution; for Rev.B and higher this limit increases to 6.67 μ sec.

NOTE: Cascade filter software is being contemplated that will extend the maximum impulse response length to at least 50 μ sec. This is of interest when very long (uncoded CW) transmitted pulses are used.

FIR-Filter prototype passband width: 0.503 MHz

This is the passband width of the ideal lowpass filter that is used to design the matched FIR bandpass filter. The actual bandwidth of the final FIR filter will depend on 1) the filter's impulse response length, and 2) the design window used in the process. The actual 3dB bandwidth will be:

- Larger than the ideal bandwidth if that bandwidth is narrow and the FIR length is too short to realize that degree of frequency discrimination. In these cases it may be reasonable to increase the filter length.
- Smaller than the ideal bandwidth if the FIR length easily resolves the frequency band. This is because of the interaction within the filter's transition band of the ideal filter and the particular design window being used. For example, for a Hamming window and sufficiently long filter length, the ideal bandwidth is an approximation of the 6dB (not 3dB) attenuation point. Hence, the 3dB width is narrower than the ideal prototype width.

This parameter should be tuned using the TTY output and interactive visual plot from the “Ps” command. The actual 3dB bandwidth is shown there, so that it can be compared with the ideal prototype bandwidth.

Limits: 0.05 to 10.0 MHz.

Output control 4-bit pattern: 0001

These are the hardware control bits for this pulsewidth. The bits are the 4-bit binary pattern that is output on PWBW0:3

Bit Limits: 0 to 15 (input must be typed in decimal)

Current noise level: -75.00 dBm

Powerup noise level: -75.00 dBm

-or-

Current noise levels - PriRx: -75.00 dBm, SecRx: -75.00 dBm

Powerup noise levels - PriRx: -75.00 dBm, SecRx: -75.00 dBm

These questions allow you to set the current value and the power-up value of the receiver noise level for either a single or dual receiver system. The noise level(s) are shown in dBm, and you may alter either one from the TTY. The power-up level(s) are assigned by default when the RVP8 first starts up, and whenever the RESET opcode is issued with Bit #8 set. Likewise, the current noise level is revised whenever the SNOISE opcode is issued. These setup questions are intended for applications in which the RVP8 must operate with a reasonable default value, up until the time that an SNOISE command is actually received. They may also be used to compare the receiver noise levels during normal operation, which serves as a check that each FIR filter is behaving as expected when presented with thermal noise.

Transmitter phase switch point: -1.00 usec

This is the transition time of the RVP8's phase control output lines during random phase processing modes. The switch point should be selected so that there is adequate settling time prior to the burst/COHO phase measurement on each pulse. This question only appears if the PHOUT[0:7] lines are actually configured for phase control (See Section 3.3.1).

Limits: -500 to 500 μ sec.

Polarization switch point for POLAR0: -1.00 usec

Polarization switch point for POLAR1: 1.00 usec

The RVP8's POLAR0 and POLAR1 digital output lines control the polarization switch in a dual-polarization radar. During data processing modes in which the polarization alternates from pulse to pulse, the transition points of these control signals are set by these two questions. The values are in microseconds relative to range zero; the same units used to define the start times of the six user triggers. The logical sense of POLAR0 and POLAR1 is set by questions described in Section 3.3.4.

Limits: -500 to 500 μ sec.

3.3.6 Mb — Burst Pulse and AFC

These questions are accessed by typing "Mb". They set the parameters that influence the phase and frequency analysis of the burst pulse, and the operation of the AFC feedback loop.

Receiver Intermediate Frequency: 30.0000 MHz

This is the center frequency of the IF receiver and burst pulse waveform. The RVP8 can operate at an intermediate frequency from any of the three alias bands 22-32MHz, 40-50MHz, and 58-68MHz. These bands are delineated by 4MHz

safety zones on either side of integer multiples of half the RVP8/IFD's 36MHz sampling frequency. The value entered here implicitly defines the band, and hence, the boundaries of the 18MHz window in which the IF is assumed to fall.

Limits: 22 to 68 MHz.

Primary Receiver Intermediate Frequency: 30.0000 MHz
Secondary Receiver Intermediate Frequency: 24.0000 MHz

These alternate questions will replace the previous question whenever the RVP8's dual-receiver mode is selected. You should enter the two intermediate frequencies for your primary and secondary (nominally horizontal and vertical polarized) receivers. Note that you can easily swap receiver channels merely by exchanging the two frequency values.

IF increases for an approaching target: YES

The intermediate frequency is derived at the receiver's front end by a microwave mixer and sideband filter. The filter passes either the lower sideband or the upper sideband, and rejects the other. Depending on which sideband is chosen, an increase in microwave frequency may either increase (STALO below transmitter) or decrease (STALO above transmitter) the receiver's intermediate frequency. This question influences the sign of the Doppler velocities that are computed by the RVP8.

PhaseLock to the burst pulse: YES

This question controls whether the RVP8 locks the phase of its synthesized "I" and "Q" data to the measured phase of the burst pulse. For an operational magnetron system this should always be "YES", since the transmitter's random phase must be known in order to recover Doppler data. The "NO" option is appropriate for non phase modulated Klystron systems in which the RVP8/IFD sampling clock is locked to the COHO. It is also useful for bench testing in general. In these "NO" cases the phase of "I" and "Q" is determined relative to the stable internal sampling clock in the RVP8/IFD module.

Minimum power for valid burst pulse: -15.0 dBm

This is the minimum mean power that must be present in the burst pulse for it to be considered valid, i.e., suitable for input into the algorithms for frequency estimation and AFC. The reporting of burst pulse power is described in Section 4.4; the value entered here should be, perhaps, 8 dB less. This insures that burst pulses will still be properly detected even if the transmitter power fades slightly.

The mean power level of the burst is computed within the narrowed set of samples that are used for AFC frequency estimation. The narrow subwindow will contain only the active portion of the burst, and thus a mean power measurement is meaningful. The full FIR window would include the leading and trailing pulse edges and would not produce a meaningful average power. Since radar peak power tends to be independent of pulse width, this single threshold value can be applied for all pulsewidths.

Limits: -60 to +10 dBm.

Design/Analysis Window- 0:Rect, 1:Hamming, 2:Blackman : 1

You may choose the window that is used in 1) the design of the FIR matched filter, and 2) the presentation of the power spectra for the various scope plots. Choices are rectangular, Hamming, and Blackman; the Hamming window being the best overall

choice. The Blackman window is useful if you are trying to see plotted spectral components that are more than 40dB below the strongest signal present. It is especially useful in the "Pr" plot when a long span of data are available. FIR filters designed with the Blackman window will have greater stopband attenuation than those designed with the Hamming window, but the wider main lobe may be undesirable. The rectangular window is included mostly as a teaching tool, and should never be used in an operational setting.

Settling time (to 1%) of burst frequency estimator: 5.0 sec

The burst frequency estimator uses a 4th order correlation model to estimate the center frequency of the transmitted pulses. Each burst pulse will typically occupy approximately one microsecond; yet the frequency estimate feeding the AFC loop needs to be accurate to, perhaps, 10KHz. Obviously this accuracy can not be achieved using just one pulse. However, several hundred of the (unbiased) individual estimates can be averaged to produce an accurate mean. This averaging is done with an exponential filter whose time constant is chosen here.

Limits: 0.1 to 120 seconds.

Lock IFD sampling clock to external reference: NO

This question determines the usage of the shared SMA connector that is labeled "AFC/(CLK)" on the RVP8/IFD. It is generally *not* necessary to phase lock the IFD sampling clock to the radar system clock, since very good stability is obtained from the burst phase measurements during normal operation. However, two cases that benefit from clock locking are 1) using the RVP8 in a klystron system where an external trigger is provided, and 2) dual-receiver systems in which computation of ϕ is important.

The following two questions will appear only if you have requested that the IFD sampling clock be locked to an external clock reference. See Section 2.2.11 for a description of the hardware setups that must accompany this selection.

PLL ratio of (1/1) ==> Input reference at 17.9876 MHz

The VCXO phase-locked-loop (PLL) in the RVP8/IFD can work with any input reference clock whose frequency is a rational multiple (P/Q) of half the desired sampling frequency, i.e., center frequency of the VCXO. This question allows this ratio to be established. In general, the best PLL performance will be attained when the ratio is reduced to lowest terms, e.g., use a ratio of 6/5 rather than 12/10.

Limits: 1 to 128 for both numerator and denominator.

VCXO has positive frequency deviation: YES

Most VCXOs have positive frequency deviation, i.e., their output frequency increases with increasing input control voltage. This question will generally be answered "yes", but is included to accommodate the other case as well. The PLL will not lock, and will be completely unstable, if the wrong choice is made.

Enable AFC and MFC functions: YES

AFC is required in a magnetron system to maintain the fixed intermediate frequency difference between the transmitter and the STALO. AFC is not required in a klystron system since the transmitted pulse is inherently at the correct frequency.

The following rather long list of questions will appear only if AFC and MFC functions have been enabled.

AFC Servo- 0:DC Coupled, 1:Motor/Integrator : 0

The AFC servo loop can be configured to operate with an external Motor/Integrator frequency controller, rather than the usual direct-coupled FM control. This type of servo loop is required for tuned magnetron systems in which the tuning actuator is moved back and forth by a motor, but remains fixed in place when motor drive is removed. These systems require that the AFC output voltage (motor drive) be zero when the loop is locked; and that the voltage be proportional to frequency error while tracking. Please see Section 3.3.6.1 for more details.

Wait time before applying AFC: 10.0 sec

After a magnetron transmitter is first turned on, it may be several seconds or even minutes until its output frequency becomes stable. It would not make sense for the AFC loop to be running during this time since there is nothing gained by chasing the startup transient. This question allows you to set a holdoff delay from the time that valid burst pulses are detected to the time that the AFC loop actually begins running.

Limits: 0 to 300 seconds.

AFC hysteresis -- Inner: 5.0 KHz, Outer: 15.0 KHz

These are the frequency error tolerances for the AFC loop. The loop will apply active feedback whenever the outer frequency limit is exceeded, but will hold a fixed level once the inner limit has been achieved. The hysteresis zone minimizes the amount of thrashing done by the feedback loop. The AFC control voltage will remain constant most of the time; making small and brief adjustments only occasionally as the need arises.

AFC outer tolerance during data processing: 50.0 KHz

In general, the AFC feedback loop is active only when the RVP8 is not processing data rays. This is because the Doppler phase measurements are seriously degraded whenever the AFC control voltage makes a change. To avoid this, the AFC loop is only allowed to run in between intervals of sustained data processing. This is fine as long as the host computer allows a few seconds of idle time every few minutes; but if the RVP8 were constantly busy, the AFC loop would never have a chance to run. This question allows you to place an upper bound on the frequency error that is tolerated during sustained data processing. AFC is guaranteed to be applied whenever this limit is exceeded.

Limits: 15 to 4000 KHz.

AFC feedback slope: 0.0100 D-Units/sec / KHz
AFC minimum slew rate: 0.0000 D-Units/sec
AFC maximum slew rate: 0.5000 D-Units/sec

These questions control the actual feedback computations of the AFC loop.

The overall span of the AFC output voltage is set by Gain and Offset potentiometers on the RVP8/IFD module (See Section 2.2.10). The control level that is applied to the AFC's 16-bit Digital-to-Analog converter is specified here in "D-Units", i.e.,

arbitrary units ranging from -100 to +100 corresponding to the complete span of the D/A converter. Since the D-Unit corresponds in a natural way to a percentage scale, the shorter “%” symbol is sometimes used.

AFC feedback will be applied in proportion to the frequency error that the algorithm is attempting to correct. The feedback slope determines the sensitivity and time constant of the loop by establishing the AFC's rate of change in (D-Units / sec) per thousand Hertz of frequency error. For example, a slope of 0.01 and a frequency error of 30KHz would result in a control voltage slew of 0.3 D-Units per second. At that rate it would take approximately 67 seconds for the output voltage to slew one tenth of its total span (20 D-Units / (0.3 D-Units / sec) = 67 sec). AFC is intended to track very slow drifts in the radar system, so response times of this magnitude are reasonable.

Keep in mind that the feedback slew is based on a frequency error which itself is derived from a time averaging process (see Burst Frequency Estimator Settling Time described above). The AFC loop will become unstable if a large feedback slope is used together with a long settling time constant, due to the phase lag introduced by the averaging process. Keep the loop stable by choosing a small enough slope that the loop easily comes to a stop within the inner hysteresis zone.

See Section 3.3.6.1 for more information about these slope and slew rate parameters.

AFC span- [-100%,+100%] maps into [-32768 , 32767]
AFC format- 0:Bin, 1:BCD, 2:8B4D: 0, ActLow: NO
AFC uplink protocol- 0:Off, 1:Normal, 2:PinMap : 1

The RVP8's implementation of AFC has been generalized so that there is no difference between configuring an analog loop and a digital loop. The AFC feedback loop parameters are setup the same way in each case; the only difference being the model for how the AFC information is made available to the outside world. Many types of interfaces and protocols thus become possible according to how these three questions are answered. AFC output follows these three steps:

- The internal feedback loop uses a conceptual [-100%,+100%] range of values. However, this range may be mapped into an arbitrary numeric span for eventual output. For example, choosing the span from -32768 to +32767 would result in 16-bit AFC, and 0 to 999 might be appropriate for 3-digit BCD; but any other span could also be selected from the full 32-bit integer range.
- Next, an encoding format is chosen for the specified numeric span. The result of the encoding step is another 32-bit pattern which represents the above numeric value. SIGMET will make an effort to include in the list of supported formats all custom encodings that our customers encounter from their vendors.

Available formats include straight binary, BCD, and mixed-radix formats that might be required by a specialized piece of equipment. The “8B4D” format encodes the low four decimal digits as four BCD digits, and the remaining upper bits in binary. For example, 659999 base-10 would encode into 0x00419999 Hex.

- Finally, an output protocol is selected for the bit pattern that was produced by encoding the numeric value. The bits may be written to the eight RVP8/Main

backpanel RS232 outputs, or sent on the uplink as a value to be received by the RVP8/IFD and converted to an analog voltage. Yet another option is for the bits to be sent on the uplink and received by the RVP8/DAFC board, which supports arbitrary remapping of its output pins.

To summarize: the internal AFC feedback level is first mapped into an arbitrary numeric span, then encoded using a choice of formats, and finally mapped into an arbitrary set of pins for digital output. We are hopeful that this degree of flexibility will allow easy hookup to virtually any STALO synthesizer that one might encounter.

PinMap Table (Type '31' for GND, '30' for +5)

```
-----  
Pin01:00   Pin02:01   Pin03:02   Pin04:03   Pin05:04  
Pin06:05   Pin07:06   Pin08:07   Pin09:08   Pin10:09  
Pin11:10   Pin12:11   Pin13:12   Pin14:13   Pin15:14  
Pin16:15   Pin17:16   Pin18:17   Pin19:18   Pin20:19  
Pin21:20   Pin22:21   Pin23:22   Pin24:23   Pin25:24  
FAULT status pin (0:None): 0, ActLow: NO
```

These questions only appear when the “PinMap” uplink protocol has been selected. The table assigns a bit from the encoded numeric word to each of the 25 pins of the RVP8/DAFC module. For example, the default table shown above simply assigns the low 25 bits of the encoded bit pattern to pins 1-25 in that order. You may also pull a pin high or low by assigning it to +5 or GND. Note that such assignments produce a logic-high or logic-low signal level, not an actual power or ground connection. The latter must be done with actual physical wires.

One of the RVP8/DAFC pins can optionally be selected as a Fault Status indicator. You may choose which pin to use for this purpose, as well as the polarity of the incoming signal level. Note that the standard RVP8/DAFC module only supports the selection of pins 1, 3, 4, 13, 14, and 25 as inputs. This setup question allows you to choose any pin, however, because it does not know what kind of hardware may be listening on the uplink and what its constraints might be.

Burst frequency increases with increasing AFC voltage: NO

If the frequency of the transmit burst increases when the AFC control voltage increases, then answer this question “Yes”; otherwise answer “No”. When this question is answered correctly, a numerical increase in the AFC drive (D-Units) will result in an increase in the estimated burst frequency. If the AFC loop is completely unstable, try reversing this parameter.

Mirror AFC voltage on- 0:None, 1:I, 2:Q : 0

AFC/MFC can be mirrored on a backpanel output of the main chassis using this question. When either “I” or “Q” is selected, the AFC/MFC voltage will be present on the corresponding BNC output, and the other output will be used for scope plotting. This configuration would be useful, for example, in a dual-receiver magnetron system that needs a phase locked acquisition clock in the RVP8/IFD, but also needs an AFC tuning voltage to control the transmit frequency. When “None” is selected, scope plotting will revert to its normal “Q” output.

The voltage range of the “I” and “Q” outputs is approximately ± 1 Volt, and is not adjustable. When AFC/MFC is mirrored on these lines, you will probably need to add an external Op-Amp circuit to adjust the voltage span and offset to match your RF components. We also recommend that you add significant low-pass filtering (cutoff at 3Hz) to remove any power line noise or crosstalk that may be originating within the RVP8/Main chassis.

Enable Burst Pulse Tracking: YES

This question enables the Burst Pulse Tracking algorithm that is described in Section 5.1.3. Remarkably, for such an intricate new feature, there are no additional parameters to configure. The characteristic settling times for the burst are already defined elsewhere in this menu, and the tracking algorithm uses dynamic thresholds to control the feedback.

Enable Time/Freq hunt for missing burst: No

Number of frequency intervals to search: 5

Settling time for each frequency hop: 0.25 sec

Automatically hunt immediately after being reset: YES

Repeat the hunt every: 60.00 sec

These questions configure the process of hunting for a missing burst pulse. The trigger timing interval that is checked during Hunt Mode is always the maximum $\pm 20\mu\text{sec}$; hence no further setup questions are needed to define the hunting process in time. The hunt in frequency is a different matter. The overall frequency range will always be the full -100% to $+100\%$ AFC span; but the number of subintervals to check must be specified, along with the STALO settling time after making each AFC change. With the default values shown, AFC levels of -66% , -33% , 0% , $+33\%$, and $+66\%$ will be tried, with a one-quarter second wait time before checking for a valid burst at each AFC setting.

You should choose the number of AFC intervals so that the hunt procedure can deduce an initial AFC level that is within a few megaHertz of the correct value. The normal AFC loop will then take over from there to keep the radar in tune. For example, if your radar drifts considerably in frequency so that the AFC range had to be as large as 35MHz, then choosing fifteen subintervals might be a good choice. The hunt procedure would then be able to get within 2.3MHz of the correct AFC level. The settling time can usually be fairly short, unless you have a STALO that wobbles for a while after making a frequency change. Note that hunting in frequency is not allowed for Motor/Integrator AFC loops, and the two AFC questions will be suppressed in that case.

The RVP8 can optionally begin hunting for a missing burst pulse immediately after being reset, but before any activity has been detected from the host computer. This might be useful in systems that both drift a lot and generally have their transmitter *On*. However, this option is really included just as a work around; the correct way for a burst pulse hunt to occur is via an explicit request from the host computer which “knows” when the pulse really should be present. Blindly hunting in the absence of that knowledge can not be done because there are many reasons why the burst pulse may legitimately be missing, e.g., during a radar calibration.

The automatic hunt for the burst pulse will always run at least once whenever the feature is enabled. The automatic hunting ceases, however, as soon as any activity is detected from the host computer. Only use this feature on radars with a serious drift problem in their burst pulse timing.

Simulate burst pulse samples: NO

The RVP8 can simulate a one microsecond envelope of burst samples. This is useful only as a testing and teaching aid, and should never be used in an operational system.

A two-tone simulation will be produced when the RVP8 is setup in dual-receiver mode. The pulse will be the sum of two transmit pulses at the primary and secondary intermediate frequencies. To make the simulation more realistic, the two signal strengths are unequal; the primary pulse is 3dB stronger than the secondary pulse.

Frequency span of simulated burst: 27.00 MHz to 32.00 MHz

The simulated burst responds to AFC just as a real radar would. The frequency span from minimum AFC to maximum AFC is given here.

3.3.6.1 AFC Motor/Integrator Option

The question "*AFC Servo- 0:DC Coupled, 1:Motor/Integrator*" selects whether the AFC loop runs in the normal manner (direct control over frequency), or with an external Motor/Integrator type of actuator. The question "*AFC minimum slew request:...*" provides additional control when interfacing to mechanical actuators whose starting and sustaining friction needs to be overcome.

The DC-Coupled AFC loop questions (changes shown in bold) are:

```
AFC Servo- 0:DC Coupled, 1:Motor/Integrator : 0  
Wait time before applying AFC: 10.0 sec  
AFC hysteresis- Inner: 5.0 KHz, Outer: 15.0 KHz  
AFC outer tolerance during data processing: 50.0 KHz  
AFC feedback slope: 0.0100 D-Units/sec / KHz  
AFC minimum slew rate: 0.0000 D-Units/sec  
AFC maximum slew rate: 0.5000 D-Units/sec
```

and the Motor/Integrator loop questions are:

```
AFC Servo- 0:DC Coupled, 1:Motor/Integrator : 1  
Wait time before applying AFC: 10.0 sec  
AFC hysteresis- Inner: 5.0 KHz, Outer: 15.0 KHz  
AFC outer tolerance during data processing: 50.0 KHz  
AFC feedback slope: 1.0000 D-Units / KHz  
AFC minimum slew request: 15.0000 D-Units  
AFC maximum slew request: 90.0000 D-Units
```

Notice that the physical units for the feedback slope and slew rate limits are different in the two cases. In the DC-Coupled case the AFC output voltage controls the frequency directly, so the units for the feedback and slew parameters use *D-Units/Second*. In the Motor/Integrator case, the AFC output determines the rate of change of frequency; hence *D-Units* are used directly.

The above example illustrates typical values that might be used with a Motor/Integrator servo loop. The feedback slope of 1.0 *D-Units/KHz* means that a frequency error of 100KHz would produce the full-scale (100 *D-Units*) AFC output. But this is modified by the minimum and maximum slew requests as follows:

- A zero *D-Unit* output will always be produced whenever AFC is locked.
- When AFC is tracking, the output drive will always be at least ± 15 *D-Units*. This minimum non-zero drive should be set to the sustaining drive level of the motor actuator, i.e., the minimum drive that actually keeps the motor turning.
- When AFC is tracking, the output drive will never exceed ± 90 *D-Units*. This parameter can be used to limit the maximum motor speed, even when the frequency error is very large.

The AFC Motor/Integrator feedback loop works properly even if the motor has become stuck in a “cold start”, i.e., after the radar has been turned off for a period of time. The mechanical starting friction can sometimes be larger than normal, and additional motor drive is required to break out of the stuck condition. But once the motor begins to turn at all, then the normal AFC parameters (minimum slew, maximum slew, feedback slope) all resume working properly. The algorithm operates as follows:

- Whenever AFC correction is being applied, the RVP8 calculates how long it would take to reach the desired IF frequency at the present rate of change. For example, if we are 1MHz away from the desired IF frequency, and the measured rate of change of the IF burst frequency is 20KHz/sec, then it will be 50 seconds until the loop reaches equilibrium.
- Whenever the AFC loop is in Track-Mode, but the time to equilibrium is greater than two minutes, then the “Minimum Slew” parameter will be slowly increased. The idea is to gradually increase the starting motor drive whenever it appears that the IF frequency is not actually converging toward the correct value, i.e., the motor is stuck.
- As soon as the frequency is observed to begin changing, such that the desired IF would be reached in less than two minutes, then the “Minimum Slew” parameter is immediately put back to its correct setup value. The loop then continues to run properly using its normal setup values.

Manual Frequency Control (MFC) operates unchanged in both of the AFC servo modes. Whenever MFC is enabled in the **Ps** command, it always has the effect of directly controlling the output voltage of the AFC D/A converter. The MFC mode can be useful when testing the motor response under different drive levels, and when determining the correct value for the minimum slew request.

3.3.7 M+ — Debug Options

A collection of debugging options has been added to the RVP8 to help users with the development and debugging of their applications code. For the most part, these options should remain disabled during normal radar operation. These questions are included so that the RVP8 can be placed into unusual, and perhaps occasionally useful, operating states.

Noise level for simulated data: -50.0 dB

This is the noise level that is assumed when simulated “I” and “Q” data are injected into the RVP8 via the LSIMUL command. The noise level is measured relative to the power of a full-scale complex (I,Q) sinusoid, and matches the levels shown on the slide pots of the ASCOPE digital signal simulator.

Limits: -100dB to 0dB

Simulate output rays: NO

Answering “YES” to this question causes the RVP8 to output bands of simulated data. The bands can occupy a selectable range interval, and span a selectable interval of data values.

Start bin:0, Width:10 bins, Bands:16

This question is only asked if we are simulating output rays. The Start Bin chooses the bin number (origin zero) where the simulated bands will begin. The width of each band (in bins), and the total number of bands are also selected. The upper limit for all parameters is the maximum bin count for the RVP8 (which depends on board configuration, and number of attached RVP8/AUX boards).

Limits: Start: 0-Max, Width: 1-Max, Bands: 1-Max

Start data value:0, Increment:16

This question is only asked if we are simulating output rays. The data value that will be assigned to the first simulated band, and the data increment from one band to the next, are selected. The permissible values are from 0 to 65535, i.e., the full unsigned 16-bit integer range. This full range is useful when simulating 16-bit output data; for the more typical 8-bit output formats, only the low byte of the start and increment are significant.

Limits: 0 to 65535

3.3.8 Mz — Transmitter Phase Control

These questions are used to configure the 8-Bit phase modulation codes that may be used to control the phase of a coherent transmitter. The RVP8 will output a pseudo-random sequence of phase codes that are chosen from a specified set of available codes, i.e., all 8-bit patterns that are valid for the phase modulation hardware. The random sequence is output only when the RVP8 is in one of its random phase processing modes (time series or parameter). At all other times, a fixed “idle” phase code pattern is output. See also Sections 3.3.1 and 3.3.5 where related phase control questions are found.

8-Bit code to output when idle: 0x00

This is the bit pattern to be output whenever the RVP8 is not in a random phase processing mode. Note that this “idle” code does not have to be one of the “active” codes that are enabled below.

Selection of Valid 8-Bit States

00-0F: Y - - - - -

10-1F: - - - - - - - - - - - - - - - -
20-2F: - - - - - - - - - - - - - - - -
30-3F: - - - - - - - - - - - - - - - -
40-4F: - - - - - - - - - - - - - - - -
50-5F: - - - - - - - - - - - - - - - -
60-6F: - - - - - - - - - - - - - - - -
70-7F: - - - - - - - - - - - - - - - -
80-8F: - - - - - - - - - - - - - - - -
90-9F: - - - - - - - - - - - - - - - -
A0-AF: - - - - - - - - - - - - - - - -
B0-BF: - - - - - - - - - - - - - - - -
C0-CF: - - - - - - - - - - - - - - - -
D0-DF: - - - - - - - - - - - - - - - -
E0-EF: - - - - - - - - - - - - - - - -
F0-FF: - - - - - - - - - - - - - - - -

This set of questions defines the subset of active 8-bit codes that are valid states for the transmit phase modulator. Answer each line with a sequence of Y's or N's to indicate whether the corresponding 8-bit code is enabled. Only the codes that appear with a "Y" will be used by the RVP8; the "-" indicates an unused code. The "-" character was used instead of "N" so that the visual contrast of the printed table would be improved.

As an example, if your klystron transmitter has an octant phase modulator that is controlled by three digital lines, you might enable phase codes zero through seven, and then cable the modulator to the low three bits of the 8-bit code. The upper five bits would not need to be used in this case.

4. Plot-Assisted Setups

The RVP8/IFD receiver module replaces virtually all of the IF components in a traditional analog receiver. The alignment procedures for those analog components are usually very tedious, and require continued maintenance even after they are first performed. Subtle drifts in component specifications often go unnoticed until they become so severe that the radar's data are compromised.

The RVP8 makes a big improvement over this by providing an interactive graphical alignment procedure for burst pulse detection, Tx/Rx phase locking, and calibration of the AFC feedback loop. You may view the actual samples of the burst pulse and receiver waveform, examine their frequency content, design an appropriate matched filter, and observe live operation of the AFC. It is a simple matter to check the spectral purity of the transmitter on a regular basis, and to discover the presence of any unwanted noise or harmonics. Moreover, the RVP8 is able to track and modify the initial settings so that proper operation is maintained even with changes in temperature and aging of the microwave components.

The Plot-Assisted Setups are accessed using the various "P" commands within the normal TTY setup interface. These commands are described later in this chapter. For a standalone RVP8 the graphical output can be viewed as a synthesized image on an ordinary oscilloscope. Section 4.1 describes how to make the oscilloscope connections. The RVP8 also supports new opcodes that allow the host computer to monitor the data being plotted. Users with IRIS/Open software can view the plots directly on their workstation screen, and thus, can carry out the graphical checkup and alignment procedures remotely via a network.

4.1 Oscilloscope Connections

All that is required to view the graphical displays is an oscilloscope with a single vertical input channel. Setup the scope as follows:

Vertical Input	BNC cable to “Q” output of RVP8, terminated in 50Ω or 75Ω according to cable type.
Vertical Channel	Variable gain, approximately $\pm 1V$ full-scale deflection. High frequency rejection enabled (to “soften” the appearance)
Horizontal Time Base	100 μsec /division (1ms full sweep), no holdoff.
Trigger Source	From vertical channel, rising edge, DC coupled.
Trigger Level	Either $-0.3V$ or $+0.3V$ (approximately)
Trigger Mode	Use “Norm” rather than “Auto”, so that the display will be blank when nothing is being plotted.

The RVP8 synthesizes a waveform on its “Q” channel consisting of a repeating sequence of graphical “strokes”, each of which is 1ms in duration. A typical plot will have between five and fifteen strokes that blend together to form the complete image. In between each stroke is a 0.5ms idle period in which the output voltage remains at its lowest level (bottom of the screen). The idle period ends with a 2.0 μsec “trigger” pulse to the top of the screen, followed by the next 1.0ms drawing stroke.

Because the oscilloscope is set to trigger on the rising edge of its input, these brief trigger pulses will initiate the horizontal sweeps that draw each stroke of the image. Thus, the waveform is self triggering, and no external trigger signal is required. This model assumes that the 0.5ms idle period (50% of the full sweep time) is long enough for the scope’s horizontal time base to retrace and begin waiting for the next trigger event. Most scopes can easily do this; but if not, try increasing the holdoff ratio (See setup question on page 3–12) until a clean and stable display is observed.

Z-axis (intensity) modulation is not required since the oscilloscope automatically blanks its beam between sweeps. When a plot requires certain strokes to be appear brighter than others, the RVP8 accomplishes this merely by drawing that particular stroke more than once in the overall sequence.

The plotting technique relies on the luminance persistence of the oscilloscope phosphor to blend the multiple strokes into a single, flicker free, complex display. This happens automatically on an analog scope, but can be awkward if a digital sampling scope is used. This is because a digital scope is usually intended to capture and hold a single trace of data, rather than to sweep out multiple traces and allow them to blend into a single image. Digital scopes are not recommended here, but if one must be used, try experimenting with its controls that affect persistence and overwriting.

4.2 P+ — Plot Test Pattern

The RVP8 can produce a simple test pattern that is useful when the oscilloscope is attached for the first time. From the TTY monitor enter the “P+” command. This will print the message “**Plotting Test Pattern...**” on the TTY and then produce the plot shown in Figure 4–1. This display is actually an overlay of six different strokes: 1) bottom line, 2) middle line, 3) top line, 4) line sloping up, 5) line sloping down, and 6) the sine wave pattern. The later changes phase with each plot so that, with a little imagination, it appears to be radiating from the left side of the display.

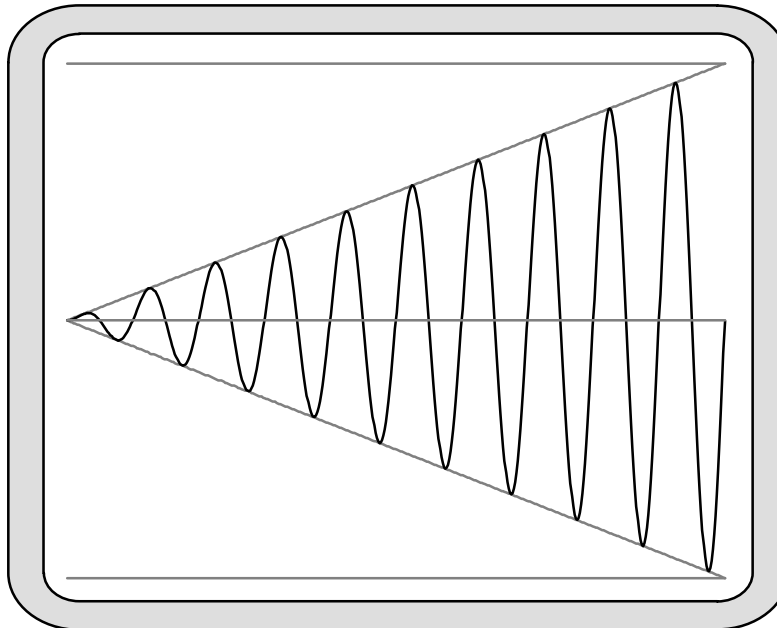


Figure 4–1: Oscilloscope Display of Test Pattern

Verify that the test pattern is stable, and adjust the vertical gain so that the top and bottom lines are exactly the full height of the oscilloscope screen. At the same time, adjust the horizontal and vertical offsets so that the image is exactly centered. When you are satisfied that the plot is being drawn correctly, type “Q” or **ESC** to return to the TTY monitor.

4.3 General Conventions Within the Plot Commands

The “**Pb**”, “**Ps**”, and “**Pr**” commands all have a similar structure to their TTY user interface. Each command begins by printing a list of subcommands that are valid in that context. These subcommands are single keystrokes that are executed immediately by the RVP8 as they are typed. The “ENTER” key is not required. The available subcommands are different for each plot command; but, as much as possible, each key has a similar meaning across all commands.

The working and measured parameters for each plot command are printed on the TTY as two lines of information following the subcommand list. The first line contains settings that only change when a subcommand is issued; but the second line is live and reflects the current status of the burst input, the IF input, or the AFC output. The first line is printed just once, but the second line is continually overprinted on top of itself. This makes it appear as a live status line whose values always remain up to date. The “Pb”, “Ps”, and “Pr” commands will report “No Trigger” on the TTY status line whenever the external trigger is expected but missing.

The TTY screen will scroll upward each time a new subcommand is executed, so that a history of information lines and command activity can be seen on the screen. You may also use the Carriage-Return key to scroll the display up at any time. If the initial list of subcommands disappears off the top, you may type “?” to force a reprint. To exit the plot command entirely and return to the TTY main menu type “Q” or **ESC**. These basic “help” and “exit” keystrokes apply everywhere within the RVP8 setup menus. To save space and minimize clutter on the TTY screen, they are not shown in the itemized list of subcommands.

Most commands have a lowercase and an uppercase version. If a lowercase command does something, then its uppercase version does the same thing but even more so (or in reverse). For example, if the “w” subcommand widens something by a little bit, then “W” would widen it a lot. This simple convention reduces the number of different subcommand keys that are needed, and makes the interface easier to memorize.

The oscilloscope display and TTY status lines are continually updated with fresh data several times per second. Occasionally it is useful to freeze a plot so that it can be studied in more detail, or compared with earlier versions. To accomplish this, every plotting command supports a “Single Step” mode that is accessed by typing the “.” (period) key. This key causes the oscilloscope display and TTY status lines to freeze in their present state, and the message “**Paused...**” will be printed. Subsequently, typing another “.” will single step to the next data update, but the plot and printout will still remain frozen. Typing “Q” or **ESC** will exit the plot command entirely (as they normally do). All other keys return the plot command to its normal live updating, but the key is otherwise discarded (i.e., subcommand keys are not executed while exiting from single step mode).

All of the plot commands support subcommands whose only purpose is to alter the appearance of the display, e.g., zoom, stretch, etc. These subcommands make no changes to the actual working RVP8 calibrations. However, the display settings are stored in nonvolatile RAM just like all of the other setup parameters. This means that all previous display settings will be restored whenever you restart each plot command. This is very convenient when alternating among the various plots.

The “**Pb**”, “**Ps**”, and “**Pr**” commands are intended to be used together for the combined purpose of configuring the RVP8's digital front end. You may, of course, run any of the commands at any time; but the following procedure may be used as a guideline for first time setups. The full procedure must be repeated for each individual pulsewidth that the radar supports.

1. Use **Mb** to set the system's intermediate frequency (See Section 3.3.6).
2. Use **Mt** to choose the PRF and pulsewidth (See Section 3.3.4). Also, choose the range resolution now, as it may constrain the design of the matched filter later.
3. Use **Mt0**, **Mt1**, etc., to set the relative timing of all RVP8 triggers that are used by the radar. Do not worry about the absolute values of the trigger start times. Just setup their polarity and width, and their start times relative to each other (See Section 3.3.5). Make an initial guess of FIR filter length as 1.5 times the pulsewidth.
4. Use **Pb** to slew the start times of all triggers so that the burst pulse is properly sampled (See Section 4.4). Refine the impulse response length if necessary so that all samples easily fit within the display window.
5. Use **Ps** to design the matched FIR filter (See Section 4.5). Further refine the impulse response length and passband width to achieve a filter that matches the spectral width of the burst, and that has strong attenuation at DC. If the FIR length is changed, return to **Pb** to verify that the burst is still being sampled properly.
6. Continue using **Ps** and **Mb** to tune up the AFC feedback loop. The settings that work for one pulsewidth should also work for all others.
7. Use **Pr** to verify that targets are being detected with good sensitivity (See Section 4.6).

Sometimes it is useful to run the **Pb** and **Ps** commands with samples from the IF-Input of the RVP8/IFD, rather than from the Burst-Input. Likewise, it is sometimes useful to view the **Pr** plots on samples of Burst data. The top-level “~” command (See Section 3.1.5) allows you to do this easily.

4.4 Pb — Plot Burst Pulse Timing

For magnetron radars the RVP8 relies on samples of the transmit pulse to lock the phase of its synthesized “I” and “Q” data, and to run the AFC feedback loop. The “**Pb**” command is used to adjust the trigger timing and A/D sampling window so that the burst pulse is correctly measured.

4.4.1 Interpreting the Burst Timing Plot

The oscilloscope plot will ultimately resemble Figure 4–2, which shows a successful capture of the transmitter's burst pulse. The horizontal axis of the display represents time, and the overall time span from the left edge to the right edge is listed as “PlotSpan” on the TTY.

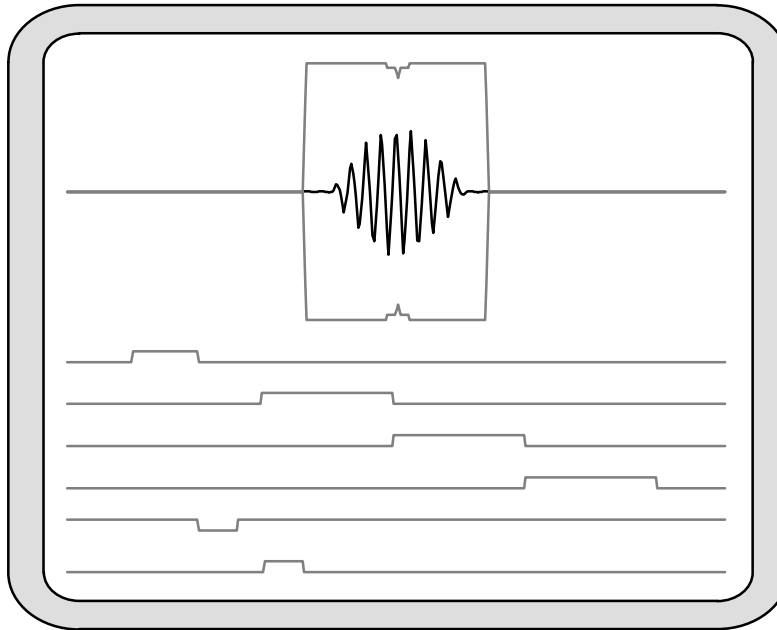


Figure 4–2: Successful Capture of the Transmit Burst

The upper portion of the plot shows the sampling window wherein the burst pulse is measured. The duration of this window is determined by the impulse response length of the matched FIR filter. This is because the same FIR coefficients that compute “I” and “Q” are also used to compute the reference phase vectors for the burst pulses. The A/D samples of the RVP8/IFD's burst input are plotted (somewhat brighter) within the sample window.

The RVP8 computes the power-weighted center-of-mass (COM) of the burst pulse envelope. This allows the processor to determine the location of the “middle” of the transmitted pulse within the burst analysis window. The **Pb** plot displays small tick marks on the top and bottom of the burst sample window to indicate the location of the COM. These markers are only displayed when valid burst power is detected. A second “error bar” is drawn surrounding the tick mark to indicate the uncertainty of the mark itself. This error interval is used by the burst pulse tracking algorithm to decide when a timing change can be made with confidence.

It is possible to independently choose a subinterval of burst pulse samples that are used by the AFC frequency estimator. Thus, the AFC feedback loop is not constrained to use the same set of samples that are chosen for the FIR filter window. The FIR window typically is longer than the

actual transmitted pulse, and thus, the samples contributing to the frequency estimate will include the leading and trailing edges of the pulse. These edges tend to have severe chirps and sidebands, compared to the more pure center portion of the pulse. The AFC frequency estimate (which is power weighted) could be misled by these edges and might not tune to the optimum center frequency if they were included.

The lower portion of the plot shows the six triggers that are output by the RVP8. Trigger #0 is at the top, and Trigger #5 is on the bottom. They are drawn in their correct polarity and timing relative to each other, and relative to the burst sample window. Note that the sample window is always drawn in the center of the overall time span. Thus, depending on the PlotSpan and location of the six trigger's edges, triggers that do not vary within the plotted time span will appear simply as flat lines.

The RVP8 defines "Range Zero" to occur at the center of the burst sample window. This also defines the zero reference point for the starting times of the six programmable triggers. For example, a trigger whose starting time is zero will be plotted with its leading edge in the exact horizontal center of the display. Knowing this convention makes the absolute value of the trigger start times more meaningful.

4.4.2 Available Subcommands Within "Pb"

The list of subcommands is printed on the TTY:

Available Subcommands within 'Pb':

I/i	Impulse response length Up/Dn
A/a & S/s	Aperture & Start of AFC window
L/l & R/r	Shift triggers left/right
T/t	Plot time span Up/Dn
Z/z	Amplitude zoom
B/b	BP Tracking On/Off (temporary)
+	Hunt for missing burst
.	Single Step

These subcommands change the matched filter's impulse response length, shift the radar triggers, and alter the format of the display.

I/i The "I" command increments or decrements the length of the matched filter's impulse response. Each keystroke raises or lowers the FIR length by one tap.

A/a & S/s These commands raise/lower the aperture/start of the subwindow of burst pulse samples for AFC. If you never use these commands, then the full FIR window will be used; however, shortening the AFC interval will result in two sample windows being drawn on the plot. The smaller AFC window should be positioned into the center portion of the transmitted pulse, where the burst amplitude and frequency are fairly stable.

L/l & R/r These two commands shift the entire group of six RVP8 triggers left or right (earlier or later in time). The lowercase commands shift in

0.025 μ sec steps, and the uppercase commands shift in 1.000 μ sec steps (approximately). The reason for shifting all six triggers at once is that the relative timing among the triggers remains preserved. However, the absolute timing (relative to range zero) will vary, and this will cause the burst pulse A/D samples to move within the sample window.

- T/t** The "T" command increments or decrements the overall time span of the plot. The available spans are 2, 5, 10, 20, 50, 100, 200, 500, 1000, 2000 and 5000 microseconds. The value is reported on the TTY as "PlotSpan".
- Z/z** The "Z" command zooms the amplitude of the burst pulse samples so that they can be seen more easily. The value is reported on the TTY as "Zoom".
- B/b** These keys temporarily disable or re-enable the Burst Pulse Tracker. The tracker must be disabled in order for the L/R keys to be used to shift the nominal trigger timing. The "b" key disables tracking and sets the trigger slew to zero; the "B" key re-enables tracking starting from that zero value.
- +** The "+" subcommand initiates a hunt for the burst pulse. Progress messages are printed as successive AFC values are tried, and the trigger slew and AFC level are set according to where the pulse was found. If no burst pulse can be found, then the previous trigger slew and AFC are not changed.

4.4.3 TTY Information Lines Within "Pb"

The TTY information lines will resemble:

```
Zoom:x2, PlotSpan:5 usec, FIR:1.36 usec (49 Taps)  
Freq:27.817 MHz, Pwr:-53.9 dBm, DC:0.14%, Trig#1:-5.00, BPT:0.00
```

- Zoom** Indicates the magnification (in amplitude) of the plotted samples. A zoom level of "x1" means that a full scale A/D waveform exactly fills the height of the sample window. Generally, the signal strength of the burst pulse will not be quite this high. Thus, use larger zoom levels to see the weaker samples more clearly. You may zoom in powers of two up to x128.
- PlotSpan** Indicates the overall time span in microseconds of the complete scope display, from left edge to right edge.
- FIR** Indicates the length of the impulse response of the matched FIR filter, and hence, the duration of the burst pulse sample window. The length is reported both as a number of taps, and as a time duration in microseconds.
- Freq** Indicates the mean frequency of the burst, derived from a 4th order correlation model. The control parameters for this model are set using the "Mb" command (Section 3.3.6).

Pwr	Indicates the mean power within the full window of burst samples. DC offsets in the A/D converter do not affect the computation of the power, i.e., the value shown truly represents the waveform's (Signal+Noise) energy.
DC	Indicates the nominal DC offset of the burst pulse A/D converter. This is of interest only as a check on the integrity of the front end analog components. The value should be in the range $\pm 2.0\%$.
Trig#1	Indicates the starting time of the first (of six) RVP8 trigger outputs. This number will vary as the "L" and "R" subcommands cause the triggers to slew left and right. Note that if the radar transmitter is directly fired by an external pretrigger, then the pretrigger delay (in the form "PreDly:6.87") will be printed instead.
BPT	This shows the present value of timing slew (measured in microseconds) being applied to track the burst. The slew will be zero initially when the RVP8 is first powered up, meaning that the normal trigger start times are all being used.

4.4.4 Recommended Adjustment Procedures

The burst pulse timing must be calibrated separately for each individual pulsewidth. Repeat the following procedure for each pulsewidth that you plan to use. Each iteration is independent.

It is first necessary to setup the proper relative timing for all RVP8 triggers that are being used. The six trigger output lines are completely interchangeable, and each may be assigned to any function within the radar system. For example, Trigger #0 might be the transmitter pretrigger, Triggers #2 and #3 might synchronize external displays, and Triggers #1, #4, and #5 might be unused.

Choose an initial impulse response length of 1.5 times the transmit pulsewidth. This length will be refined later when the matched filter is designed (See Section 4.5). Adjust the plot time span to view a small region around the sample window, and set the initial amplitude zoom to x16. This assures that the plotted waveform will still be noticeable even if the burst signal strength is very weak.

Verify that the transmitter is radiating, and observe the burst pulse samples on the display. Use the "L" and "R" commands to shift the timing of all six triggers relative to range zero. This has the effect of moving the burst sampling window relative to the transmitted pulse. Depending on whether the triggers are set properly, you may at first see nothing more than a flat line of misplaced A/D samples. However, after a few moments of hunting, the burst pulse should appear on the oscilloscope screen. Fine tune the triggers so that the burst envelope is centered in the window, and adjust the amplitude zoom for a comfortable size display.

The clean center portion of the burst pulse should then be isolated to a narrower subwindow of the overall FIR interval. Use the "A" and "S" commands to change the aperture and start of the narrowed region from which the AFC frequency estimator's data will be derived.

Check that the burst pulse signal strength is reasonably matched to the input span of the RVP8/IFD's A/D converter. The maximum analog signal level is +4dBm. Exceeding this level produces distorted samples that would seriously degrade the algorithms for phase locking and

AFC. However, if the signal is too weak, then the upper bits of the A/D converter are wasted and noise is unnecessarily introduced. We recommend a peak signal level between -6dBm and $+1\text{dBm}$, i.e., a signal that might be viewed at x2 or x4 zoom. Take note of the burst energy level reported on the TTY; it will help decide the minimum energy threshold for a valid burst pulse, which is needed in Section 3.3.6.

4.5 Ps — Plot Burst Spectra and AFC

Once the transmit burst pulse has been captured the next step is to analyze its frequency content and to design a bandpass filter that is matched to the pulse. In a traditional analog receiver the matched filters use discrete components that can not be adjusted, and the transmit spectrum can not be viewed unless a spectrum analyzer is on hand. The RVP8 eliminates all of these restrictions via its “Ps” command, which plots the burst spectrum, designs the bandpass filter, plots its frequency response, and also helps with alignment of the AFC.

4.5.1 Interpreting the Burst Spectra Plots

An example of a plot from the Ps command is shown in Figure 4–3. The oscilloscope screen is divided into two independent areas. The major portion (the lower seven eighths) is devoted to power spectrum plots of the burst pulse and/or the matched filter response. The top portion (single line) serves as a visual indicator of the present AFC level.

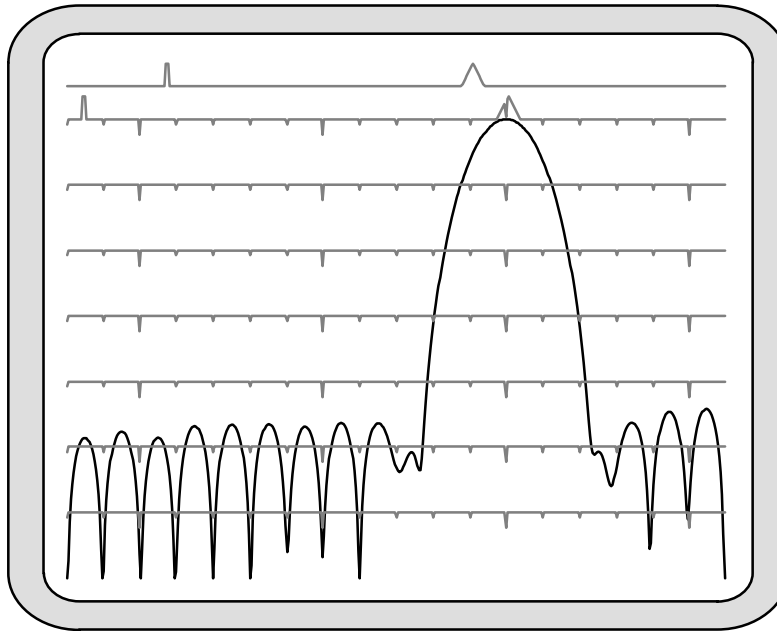


Figure 4–3: Example of a Filter With Excellent DC Rejection

The horizontal axis of the spectrum plot represents frequency. The overall span from the left edge to the right edge is always 18MHz, but the exact endpoints depend on which alias band the radar's intermediate frequency falls in. For example, a 30MHz IF would imply a horizontal axis range of 18MHz to 36MHz, whereas a 60MHz IF would make the range 54MHz to 72MHz. The frequency span is printed on the TTY when the command is first entered.

Since the left edge of the spectral plot always represents an integer multiple of 18MHz, either the left side or the right side will always be a multiple of 36MHz. This is important to remember when designing the matched filter, since fixed DC offsets in the A/D converters will appear aliased at these 36MHz multiples.

The vertical axis of the spectrum plot is logarithmic and is marked with faint horizontal lines in 10-dB increments. An overall dynamic range of 70 dB can be viewed at once. The horizontal lines also contain major and minor tick marks to help calibrate the frequency axis. Major marks are small downward triangles that represent integer multiples of 5MHz; minor marks are in between and represent 1-MHz steps. The power spectrum example in Figure 4-3 is from a system with an intermediate frequency of 30MHz. Thus, the left edge of the plot begins at 18MHz, and the graph is centered on the third major tick, i.e., 30MHz.

Two types of spectra can be plotted on the screen: 1) the frequency response of the FIR filter, and 2) the frequency content of the burst pulse itself. The burst spectrum is computed by first applying a Hamming window to the raw samples. You may choose to view either plot individually, or both at the same time.

Figure 4-3 is an example of a single filter response plot, whereas Figure 4-4 shows a combined display of both spectra. The combined display makes it easy to compare the filter being designed with the live waveform that it is intended to selectively pass. Note that the filter's frequency response is always drawn with its passband peak touching the top of the plot. The vertical height of the burst spectrum, however, will vary with signal strength but can be adjusted using the "Z" subcommand.

The horizontal line at the top of the plotting area is also marked with an upward pointing major and minor tick. These indicate the present value of the burst pulse frequency estimator. The major tick is a triangle whose position along the horizontal axis corresponds directly to the estimated frequency. It should always be positioned directly over the main lobe of spectral power. The minor tick gives finer scale resolution by indicating the fractional part of each 1-MHz multiple. For example, in Figure 4-3 the burst frequency estimate was 30.027MHz. The major tick thus appears slightly to the right of 30MHz, and the minor tick appears 2.7% across the screen. Note that when an upward tick happens to overlap a downward tick the two simply add together so that both can still be seen.

It is helpful to read the minor tick relative to the ten horizontal division lines that are present on most scopes. Motion of the minor tick is apparent even with very small changes in burst pulse frequency; a change of just 5 KHz can easily be seen. This means that you can observe the frequency drift of the magnetron in great detail, and also watch the AFC's behavior in real time.

The horizontal line at the very top of the display (above the spectra plot) serves to indicate the present value of the AFC control voltage. The line contains an upward pointing major and minor tick, similar to the ones used to represent the burst frequency estimate on the line below. However, the horizontal axis now represents voltage rather than frequency, and the overall span is the complete range of the AFC's digital-to-analog converter. The major tick will move from the left edge to the right edge as the AFC varies from its minimum to maximum value. The minor tick will traverse the screen at ten times this rate.

4.5.2 Available Subcommands Within "Ps"

The list of subcommands is printed on the TTY:

Frequency span of the plot is 18.0 MHz to 36.0 MHz.

Available Subcommands within 'Ps':

I/i	Impulse response length Up/Dn
N/n & W/w	Filter bandwidth Narrower/Wider
U/u & D/d	MFC Up/Down (On/Off '=' , Test ' ')
A/a & S/s	Aperture & Start of AFC window
#	Print filter coefficients
\$	Search for an optimal filter
V/v	Number of spectra averaged
Z/z	Amplitude zoom
<space>	Alternate Plots
%	Toggle between dual receivers
.	Single Step

These subcommands change the design of the matched FIR filter, assist with calibration of the AFC loop, and alter the format of the display.

I/i The "I" command increments or decrements the length of the matched filter's impulse response. Each keystroke raises or lowers the FIR length by one tap. Often the matched filter's characteristics can be very much improved merely by changing the FIR length by one or two taps. Be sure to experiment with this as you design your filter.

N/n & W/w The "N" and "W" commands change the passband width of the matched filter, making it narrower or wider. The lower case commands make changes in 1KHz steps, and the upper case commands use 100KHz steps. The value is reported on the TTY as "BW". Often a small change in passband width will shift the exact locations of the filter's zeros, and possibly improve the DC rejection.

U/u & D/d The "U" and "D" commands implement the Manual Frequency Control (MFC) override, and allow the RVP8/IFD's AFC output voltage to be manually set to any fixed level. The lower case commands make changes in 0.05 D-Unit steps, and the upper case commands use 1.0 D-Unit steps. The value is reported on the TTY as "AFC".

= MFC mode is toggled on and off using the "=" key. A warning will be printed if the Ps command is exited while MFC is enabled, and you will be given a second chance to reenale AFC.

| The AFC test submode is entered by typing the "|" key. The following list of keybindings will be shown, and will remain in effect until the test mode is exited by typing "Q".

AFC Test Mode Subcommands	
W	Use Walking-Ones pattern
P	Toggle Pin/Bit numbering
0-9,A-O	Toggle AFC Bits 0-24 (Pins 1-25)


```

2 2 2 2 2 1 1 1 1 1 1 1 1 1 1
4 3 2 1 0 9 8 7 6 5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0
- - - - - - - - - - - - - - - - - - - - - - - - - - -
O N M L K J I H G F E D C B A 9 8 7 6 5 4 3 2 1 0
    
```

The **P**s command continues to run normally during the AFC test mode. The customary AFC information will be replaced with a hexadecimal readout of the present 25-bit value. Your live display may look something like:

```

Navg:3, FIR:1.33 usec (48 Taps), BW:1.000 MHz, DC-Gain:ZERO
Freq:26.610 MHz, Pwr:-64.6 dBm, AFC-Test:0000207F (Bits)
    
```

Initially, a walking-ones bit pattern will be output in lieu of the normal formatted AFC value. This test pattern shifts a single “1” downward through the AFC word, making a transition approximately every 4ms. It is intended to help ring out and test the wiring for digital AFC installations. The walking-ones test is handy as an oscilloscope diagnostic, and you may return to it at any time by typing “W”.

Typing any of the characters “0” through “9” or “A” through “O” will enter a new mode in which a static 25-bit digital AFC pattern is controlled directly. Each key toggles its corresponding bit, as summarized in the keybindings printout. Any 25-bit pattern can be made by toggling the appropriate bits (initially all zero) to one. Within any particular pattern, it is also easy to toggle a particular bit On/Off in order to verify its function.

The “P” command lets you decide whether the 25-bit word represents a numeric AFC span that is mapped into pins via the pin-map table in the **Mb** menu; or whether it represents those pins directly. The printed hex test value will be followed either by “(Bits)” or “(Pins)” accordingly. When in “Pins” mode, the “0” key toggles Pin-1, the “1” key toggles Pin-2, etc. When in “Bits” mode, the “0” key toggles whatever pin or pins have been designated to be driven from Bit-0 of the numeric AFC. The “Pins” mode is useful when you are doing the initial electrical tests of the wiring of each pin. After the pin wiring has been verified and the **Mb** mapping table has been created, then the “Bits” mode allows you to test the complete digital AFC interface.

The “#” command results in a printout of the coefficients of the current FIR filter. The values are scaled by the coefficient with the largest absolute value, so that they all fall within the -1 to +1 range. This detailed information may be used to model the behavior of the filter for point targets that fall in between discrete range bins, e.g., as will happen when performing a radar sphere calibration. See Section 5.1.1 for the exact definition of these coefficients.

\$ The “\$” command performs an automatic search for optimal (DC gain of zero) filters in the vicinity of the current one. As an example, suppose that we wanted an optimal filter that was approximately 2.2 usec long and 650 KHz wide. We would first use the “**I/i**” and

“**W/wN/n**” subcommands to manually move to that starting point. Typing “**\$**” would then print a dialog line in which the search span length and width are chosen. You may keep the indicated values or type in new ones, just as for all RVP8 setup questions. The search begins when the spans are accepted.

The search procedure may require a few seconds to a few minutes, depending on the length and width spans that are being scanned. During this time, a progress message is printed showing the length and width currently under examination. You may type “**Q**” to abort the search and retain the original filter settings. When the search completes normally, it will print “Done” and replace the old filter settings with the best ones that could be found.

In dual-receiver mode, the “**\$**” command will search for a filter that minimizes the maximum width and DC offset at both receiver’s intermediate frequencies. The final filter will be the one that has the best simultaneous performance at both IFs.

V/v The “**V**” command increments or decrements the number of burst pulse spectra that are averaged together to create the plot. The count ranges from one (no averaging) to 25, and is reported on the TTY as “**Navg**”.

Z/z The “**Z**” command zooms (i.e. shifts on a logarithmic scale) in 1.0-dB steps the amplitude of the burst pulse spectra. This is useful when the overall 70dB plot span is not sufficient to hold the full range. Zoom can also be used to line up the burst spectrum with the filter response so that the two can be compared. The zoom level is not printed on the TTY because there is nothing useful that could be done with it.

<space> The space bar alternates among three choices for the type of spectra that are plotted: 1) FIR frequency response, 2) Burst pulse spectrum, and 3) Both.

% In dual-receiver mode, the “**%**” command toggles between each receiver. The printed status line is prefixed with “**Rx: Pri**” or “**Rx: Sec**” according to which receiver is selected. Specifically, typing “**%**” will toggle the plot of the FIR filter’s frequency response, and the printout of its DC-Gain. However, the plotted spectrum and printed power levels are always based on the sum of all input signals, and thus do not change with “**%**”.

4.5.3 TTY Information Lines Within “Ps”

The TTY information lines will resemble:

```
Navg:3, FIR:1.33usec (48 Taps), BW:1.000, MHz, DC-Gain:ZERO
Freq:30.027 MHz, Pwr:-64.2 dBm, Loss:1.2dB, AFC:23.05% (Manual)
```

Navg Indicates the number of burst spectra that are averaged together prior to plotting. Larger amounts of averaging increase the ability to see subtle spectral components, but the display will update more slowly.

FIR	Indicates the length of the impulse response of the matched FIR filter. See description on Page 4–8.
BW	Indicates the actual 3dB bandwidth of the matched filter. This is the complete width of the passband from the lower frequency edge to upper frequency edge. Note that the filter's center frequency is fixed at the radar's intermediate frequency, as chosen in the "Mb" setup command.
DC-Gain	Indicates the filter's response to DC (zero frequency) input. The value is a negative number in decibels, or the word "ZERO" if the filter has a true zero at DC. The filter's DC gain should be kept at a minimum so that fixed offsets in the A/D converters will not propagate into the synthesized "I" and "Q" values.
Freq	Indicates the mean frequency of the burst. See description on Page 4–8.
Pwr	Indicates the average power in the full burst sample window. See description on Page 4–9.
Loss	The filter loss is a positive number in deciBels, and is only displayed if the overall burst power exceeds the minimum valid burst threshold set in the Mb command (clearly, it would not be possible to compute the filter loss when the burst waveform is missing). The filter loss is discussed further in Section 4.5.4.
AFC	<p>Indicates the level and status of the AFC voltage at the RVP8/IFD module. The number is the present output level in D-Units ranging from –100 to +100. The shorter "%" symbol is used since percentage units correspond in a natural way to the D-Units.</p> <p>An additional number in square brackets will be printed to the right of the AFC level to show the encoded bit pattern which corresponds to that level. This will only appear when the RVP8 deduces that a special digital format is being used, i.e., when the backpanel phase-out lines have been configured for AFC, or when any of the following are not true: a) the low and high numeric AFC span is –32768 to +32767, b) the uplink is enabled, c) the uplink format is binary, and d) pinmap protocol is OFF. Binary format is printed in base-10, BCD format is printed in Hex, and 8B4D format is printed with the low 16-bits (four BCD digits) in Hex and the upper bits in base-10.</p> <p>The AFC mode is shown to the right of the numerical value(s), and can take on the following states.</p> <p>(Disabled) Indicates that neither AFC nor MFC are enabled. The output voltage remains fixed at 0% (center of its range).</p> <p>(Manual) Manual Frequency Control (MFC) is overriding AFC. The "U" and "D" commands can be used to slew the voltage up and down.</p> <p>Whenever any of the following four states appears, it implies that AFC is enabled and that MFC is disabled.</p>

- (NoBurst)** The energy in the burst is below the minimum energy threshold for a valid pulse (See Page 3–26). The AFC loop remains idle.
- (Wait)** The burst pulse has become valid just recently, but the AFC loop is idle until the transmitter stabilizes (See Page 3–27)
- (Track)** The burst pulse is valid, and the AFC loop is tracking in order to bring the burst frequency within the inner hysteresis limits.
- (Locked)** The burst pulse is valid and the AFC loop is locked. The burst frequency is now within the outer hysteresis limits and has previously been within the inner limits while tracking. This is the stable operational mode in which data acquisition should take place.

4.5.4 Computation of Filter Loss

The **Ps** printout displays the power loss (calibration error) that results when the given filter is applied to the given transmit burst waveform. This allows you to correct for the difference between what a broad-band power meter measures as the overall transmit power, and what the RVP8 narrow-band receiver will detect within its passband. The filter loss is a subtle quantity that depends on the combined characteristics of both the transmit waveform and the receiver matched filter.

The filter loss is zero if the burst waveform consists of a pure sinusoid at the designated intermediate frequency. It is also very near zero as long as most of the burst energy is confined within the passband of the RVP8's filter. The filter loss will increase as the bandwidth of the burst waveform increases and begins to spill out of that passband. Typical losses for a well-matched filter are in the 0.5–1.8dB range, depending on the FIR length and other design criteria.

As an example, consider how the RVP8 filters would respond to a simple rectangular pulse of energy lasting T_o seconds. For this discussion we can ignore the sinusoidal IF carrier that must also be present within the pulse, and just focus on the rectangular envelope. This is valid because the signal bandwidth, and hence the filter loss, is determined entirely by the shape of the modulation envelope. For a pulse of length T_o to have unit-energy it must have an amplitude of $1/\sqrt{T_o}$. By centering this pulse at time zero the power spectrum is easily computed using a real-valued integral:

$$S(f) = \left[\int_{-T_o/2}^{T_o/2} \frac{1}{\sqrt{T_o}} \cos(2\pi f t) dt \right]^2 = \frac{\sin^2(\pi f T_o)}{\pi^2 f^2 T_o}$$

where f is the frequency in Hertz. This is the familiar “synch” function, whose main frequency lobe extends from $-1/T_o$ to $1/T_o$ Hertz, and whose total power integrated over all frequencies is 1.0.

We can now examine what the filter loss (dB_{loss}) would be if this pulse were applied to a bandpass filter. The filter loss is simply the ratio of the power that is passed by the filter, divided by the total input power (1.0 in this case). Assume for the moment that the filter is an ideal bandpass filter centered at zero Hertz (corresponding to how $S(f)$ was defined) and having a bandwidth B_w , then:

$$dB_{loss} = -10 \log_{10} \left[\int_{-B_w/2}^{B_w/2} S(f) df \right]$$

This integral can be computed for a few “interesting” filter bandwidths, yielding filter losses of 0.44dB, 1.11dB, and 3.31dB when B_w is $2/T_o$, $1/T_o$, and $1/2T_o$ respectively. These three example bandwidths correspond to filters that pass the entire main frequency lobe, half of that lobe, and one quarter of it.

You can experimentally verify these results using the RVP8 as follows:

- Using the **Mt0** command, setup a $T_o = 0.5$ μ sec trigger pulse from the RVP8 in the vicinity of range zero, and use that trigger to gate a signal generator whose output is applied to the RVP8/IFD Burst Input. Also setup 125-meter range resolution, and a rather long 6.0 μ sec impulse response length. The long length will make the transition edges of the matched filter as steep as possible, so that it becomes a reasonably good approximation to the ideal bandpass filter used in the above analysis.
- Use the **Pb** command to verify that the burst pulse is present, and position the triggers left and right until the pulse is centered exactly at zero.
- Use the **Ps** command to examine the frequency spectrum of the pulse. You should see a main energy lobe that is 4MHz wide and centered at the radar's IF. There should also be weaker lobes spaced 2MHz apart on both sides of the main lobe. If the lobe spacing does not look quite right, it may be because the signal generator has slightly shortened or lengthened the trigger gate.
- Continue using **Ps** to examine filters that are 4MHz, 2MHz, and 1MHz wide at their 3dB points. You should see filter losses reported that are very close to the theoretical values for the ideal bandpass filter.

In the above analysis we have assumed that $S(f)$ is the idealized power spectrum of a continuous time signal. Of course, the RVP8 filter loss algorithm can only work from an estimate of $S(f)$ that is obtained from a finite number of samples. The filter loss calculation thus becomes more complicated than the above example in which we integrated an idealized filter response over an idealized power spectrum.

Let $\hat{B}(f)$ denote the estimated power spectrum of the continuous-time Tx burst waveform, for which we have only a finite number of discrete samples $\{b_n\}$. For purposes of this discussion we can assume that the frequency variable f is continuous. Furthermore, let $\hat{C}(f)$ denote a

power spectrum estimate that is derived in an identical manner using the same number of samples, but of a pure sine wave at the radar's IF. The RVP8 determines $\hat{B}(f)$ according to its sampled measurement of the transmitted waveform; however it can calculate $\hat{C}(f)$ internally based on an idealized sinusoid. The reported filter loss is then:

$$dB_{loss} = -10 \log_{10} \left[\frac{\int |H(f)|^2 \hat{B}(f) df}{\int \hat{B}(f) df} \div \frac{\int |H(f)|^2 \hat{C}(f) df}{\int \hat{C}(f) df} \right]$$

Where $|H(f)|^2$ is the spectral response of the RVP8 IF filter, and the integrals are performed over the Nyquist frequency band that is implied by the RVP8/IFD sampling rate. Note that the two integrals involving $\hat{C}(f)$ will have constant value and need only be computed once. They serve to normalize the $\hat{B}(f)$ integrals in such a way that the filter loss evaluates to 0dB whenever the transmit burst is a pure tone at IF.

This normalization is necessary for the filter loss values to be meaningful. Regardless of the bandwidth and center frequency of $H(f)$, the filter loss should be reported as 0dB whenever the Tx waveform appears to have zero spectral width, i.e., is indistinguishable from a pure IF sinusoid. Of course, the real Tx waveform has only finite duration, and thus should never look like a pure tone as long as the RVP8 is able to "see" the entire Tx envelope. For this reason, it is important that the filter's impulse response length be set long enough (using the **Pb** plot) to insure that all of the details of the Tx waveform are being captured. If the entire Tx envelope does not fit within the FIR filter, then the filter loss will be underestimated because the Tx spectrum will appear to be narrower than it really is.

The RVP8's calculation of digital filter loss is very similar to how the loss of an analog filter would be measured on a test bench. Suppose we are given an analog bandpass filter and are asked to determine its spectral loss when a given waveform is presented. We could use a power meter to measure the waveform power before and after the filter is inserted, and compute the ratio of these two numbers. This corresponds to the first integral ratio in the above equation. However, this is not by itself an accurate measure of filter loss because it does not take into account the bandwidth-independent insertion loss. Put another way, a flat 3dB pad would seem to produce a 3dB filter loss in the above measurement, but that is certainly not the result that we desire. The remedy is to make a second pair of power measurements of the filter's response to a CW tone at the passband center. This serves to calibrate the gain of the filter, and allows us to compute a filter loss that captures the effects of spectral shape independent of overall gain. This normalization step corresponds to the second integral ratio in the above equation.

If your radar calibration was performed using CW waveforms, then the reported filter loss should either be added to the receiver calibration losses, or subtracted from the effective transmit power; the net result being that dBZ_0 will increase slightly.

In dual-receiver systems the filter loss is computed for the primary and secondary channels using only the portion of bandwidth that is allocated to that channel. For example, if the two IFs are 24MHz and 30MHz, then the filter losses for each channel would use the frequency intervals

21–27MHz and 27–33MHz respectively. This is necessary to avoid picking up energy from the other receiver and interpreting it as out-of-band input power. A consequence, however, is that the real out-of-band power is underestimated, i.e., the filter loss itself is underestimated. We recommend temporarily switching dual-receiver systems back to single-receiver mode when the filter loss is being measured. This is easily done by changing the **Mc** setup question back to “single”, and disconnecting the secondary burst input to the RVP8/IFD.

4.5.5 Recommended Adjustment Procedures

The **Ps** command should be used only after the burst pulse has been successfully captured by way of the **Pb** command. Use the <space> key to display the burst spectrum plot by itself, and use the “Z” key to shift the entire graph into view. You are now looking at the actual frequency content of the transmitted pulse. The plot should show a clean main power lobe centered at the receiver’s intermediate frequency. Check the spectrum for spurious harmonics, excessive width, and other out-of-band noise. Make any adjustments in the transmitter that might give a sharper main lobe or reduced spurious noise.

Once we know the power spectrum of the transmitted pulse we can begin designing the matched FIR filter. Use the <space> key to display both the filter response and the burst spectrum on the same plot. Use the “Z” key to shift the burst’s main lobe up to the top horizontal line of the graph. This makes it level with the filter’s peak lobe, which is always drawn tangent to the same top line.

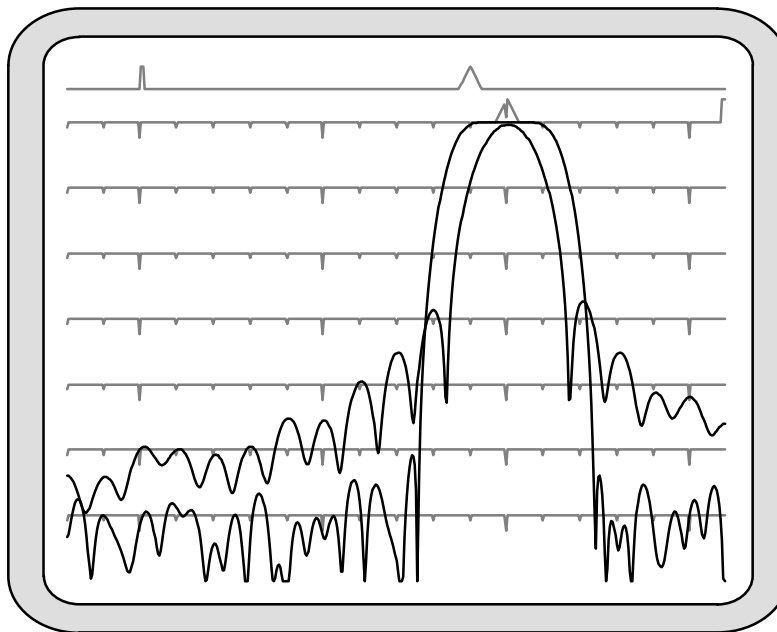


Figure 4-4: Example of a Poorly Matched Filter

Begin with the FIR length that was chosen previously in the **Pb** command, and use the “N” and “W” keys to set an initial bandwidth equal to the reciprocal of the pulsewidth. The main lobes of the two plots should more-or-less overlap. Experiment with changing the FIR length and bandwidth to achieve a filter with the following properties.

- The filter width should be no greater than the burst spectral width. A wider passband will reduce the SNR of the received signal because out-of-band noise would be allowed to pass.
- The DC gain should be as small as possible, preferably less than -64dB (See discussion below).
- If there are conspicuous interference spikes at particular frequencies, try to adjust the location of the filter's zeros so that the interference is maximally attenuated.

The filter should not pass any frequencies that do not actually contain useful information from the original transmitted pulse. If anything, choose a filter whose width is slightly narrower than the burst's spectral width. Figure 4-4 shows an example of a filter that is poorly matched to the pulse. Although the filter has fairly good DC rejection, it passes frequencies that are outside of the transmitter's broadcast range. These frequencies contribute nothing but noise to the synthesized "I" and "Q" data stream.

There are two procedures for optimizing the performance of the FIR filter:

- **Manual Method** — The process of arriving at a nearly optimal filter will require a few minutes of hunting with the "I", "W", and "N" keys. Every time you press any of these keys the RVP8 designs a new FIR filter from scratch, and displays the results. Fortunately, the DSP chips are fast enough that this can be done quickly and interactively. Even though the user must still control two degrees of freedom (length and bandwidth), the RVP8's internal design calculations are actually performing several hundred iterative steps each time, which preferentially select for the best filter. Because the FIR coefficients are quantized in the filter chips themselves, the process of finding an optimal filter becomes quite nonlinear.
- **Automatic Method** — Simply type the "\$" command and let the RVP8 do all of the work (See description in Section 4.5.2).

The offset error of the RVP8/IFD's A/D converter is at most 10mV , i.e., -27dBm into its 50Ω input. If we wish to achieve 85-dB of dynamic range below the converter's $+4\text{dBm}$ saturation level, then we expect usable "I" and "Q" values to be obtainable from a (sub-LSB) input signal at -81dBm . This is 54dB below the interference that would result from the worst-case A/D offset. But a weak input signal at -81dBm would still be damaged by even an equal level of DC interference. Therefore, adding another 10dB safety margin, we get -64dB as the recommended maximum DC gain of the matched filter. This DC gain should be reduced even further if it is known that coherent leakage is present in the receive signal at a level greater than the -27dBm worse-case A/D offset.

Figure 4-5 shows a 30MHz filter with particularly poor (-42dB) DC rejection. The frequency range of the plot is 18MHz – 36MHz ; hence, DC appears aliased at the right edge. Not only is there a peak in the filter's stopband at DC, but it is the largest stopband peak anywhere in the plot. Contrast this with the filter shown in Figure 4-3 that has a true zero at DC. In general, a poor filter can be converted into a "nearby" good filter by making only incremental changes to the impulse response length and/or desired bandwidth.

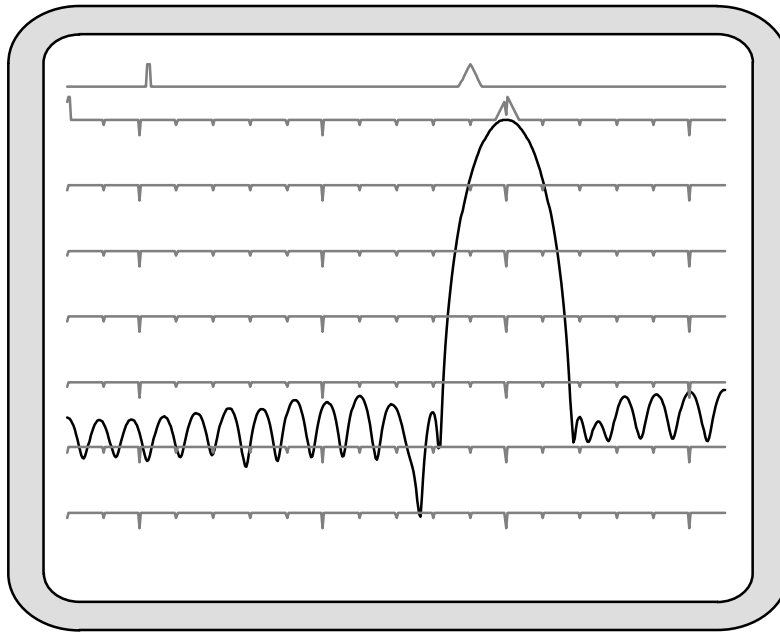


Figure 4-5: Example of a Filter With Poor DC Rejection

4.6 Pr — Plot Receiver Waveforms

The “**Pb**” and “**Ps**” commands described in the previous sections have been used to analyze the signal that is applied to the “Burst-In” connector of the RVP8/IFD receiver module. The task that remains is to checkout the actual received signal that is connected to “IF-In”. The goal is to verify that the received signal is clean and appropriately scaled, and that nearby targets can be seen clearly. The “**Pr**” command is used to make these measurements.

4.6.1 Interpreting the Receiver Waveform Plots

An example of a plot from the **Pr** command is shown in Figure 4–6. The horizontal axis represents time (range) starting from a selectable offset and spanning a selectable interval. The data are acquired from a single transmitted pulse, are plotted both as raw IF samples and as the LOG of the detected power using the FIR filter for the current pulsewidth.

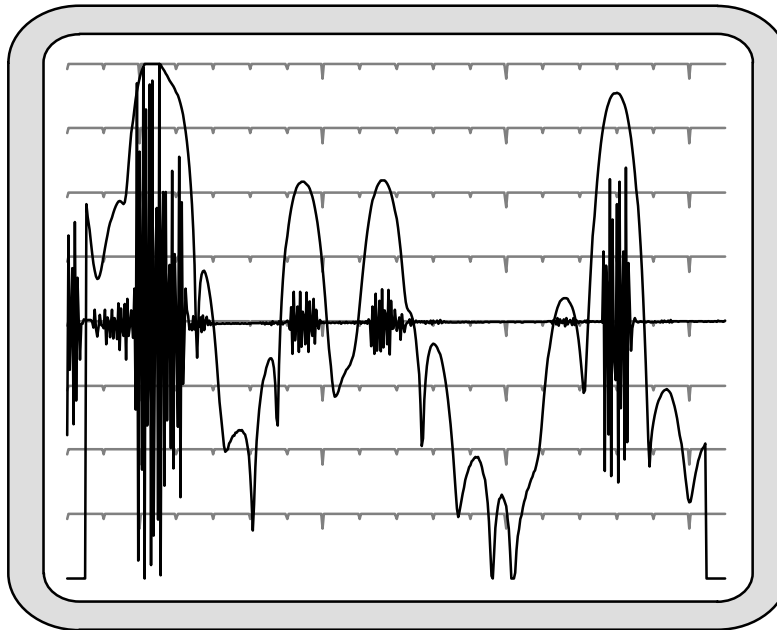


Figure 4–6: Example of Combined IF Sample and LOG Plot

The IF samples are plotted on a linear scale as signed quantities, with zero appearing at the center line of the scope. Any DC offset that may be present in the A/D converter is not removed, and will be seen as a shift in the baseline at higher zoom levels. For example, the converter’s worst case DC offset of 10mv would appear as a 91-count offset in the 12-bit range spanning –2048 to +2047. At the x32 or higher zoom scales, this offset would peg the sample plot off scale. Typically the DC offset will be much less than this worst case value; but the RVP8 preserves the DC term in the **Pr** sample plot so that its presence is not forgotten.

The “AC” amplitude of the IF samples will increase wherever targets are present. On top of these samples is drawn the detected power on a logarithmic scale. Each horizontal line represents a 10dB change in power. The graph is scaled so that the LOG power reaches the top display line when the samples occupy the full amplitude span. Using Figure 4–6 as an example,

the two equal-power targets just to the left of center are approximately 18dB down from the top. The amplitude of the samples is thus $10^{(-18/20)} = 0.13$, i.e., 13% of full scale. This correspondence between the LOG scale and the amplitude scale applies regardless of the plot's zoom level. As the IF samples are zoomed up and down by factors of two, the LOG plot will shift up and down in 6dB steps.

The LOG plot is obtained by convoluting the FIR filter coefficients with the raw IF data samples, and then plotting $\log(I^2 + Q^2)$ at each possible offset along the sampling interval. This convolution produces only $(1 + N - I)$ output points, where N is the number of sample points and I is the length of the FIR filter. For this reason the LOG plot begins approximately I/2 samples from left side and ends approximately I/2 samples from the right.

The LOG points are computed at each possible offset within the raw IF samples. At the nominal 35.975MHz sampling rate the spacing between LOG samples will be a mere 8.33 meters. Thus, the LOG plot gives a very detailed view of received power versus range. Of course, successive LOG points will be highly correlated because successive input data intervals differ by only one sample point. This is why the LOG plots appear smooth compared to the instantaneous variation of the raw IF samples.

As the starting offset of the **Pr** plot is decreased to range zero you will begin to see part of the burst pulse (the second half of it) appear at the left edge of the plot. This is because the burst data samples are multiplexed onto the same fiber cable that carries the IF data samples. Zero range is defined to occur at the center of the burst window; hence, the later half of the burst pulse will be visible when the plot begins at range zero.

A second type of **Pr** display is shown in Figure 4-7. This plot shows a frequency spectrum of the received data samples in a format that is nearly identical to the **Ps** display. The horizontal axis represents the same band of frequencies (half the sampling rate), and the vertical axis represents power in 10dB steps. The entire vertical axis is used so that an overall span of 80dB is visible. This particular plot was made with the time span set to 50 μ sec, and with a 1-meter antenna attached to the IF input so that a broad range of signals (radio stations, electrical noise, etc.) would be detected.

The purpose of the **Pr** power spectrum is to check for spurious interference in the IF signal from the radar receiver. The spectrum should be viewed with the transmitter turned off, and with the starting range moved out so that the burst samples are not mixed in with the receiver data. The power spectrum is computed using the complete interval of raw IF samples which, depending on the chosen time span, may contain many hundreds of points. The frequency resolution of the **Pr** spectrum can therefore be quite fine; making it possible to discern any interfering frequencies with some detail.

The **Pr** spectrum plot will properly show a 0-Hz peak from any DC offset in the A/D converter, and is thus consistent with how the DC offset is presented in the **Pr** sample plot. Both of these plots preserve the DC component of the IF samples so that it can be monitored as part of the routine maintenance of the receiver system. This is one of the few places in the RVP8 menus and processing algorithms where the DC term deliberately remains intact.

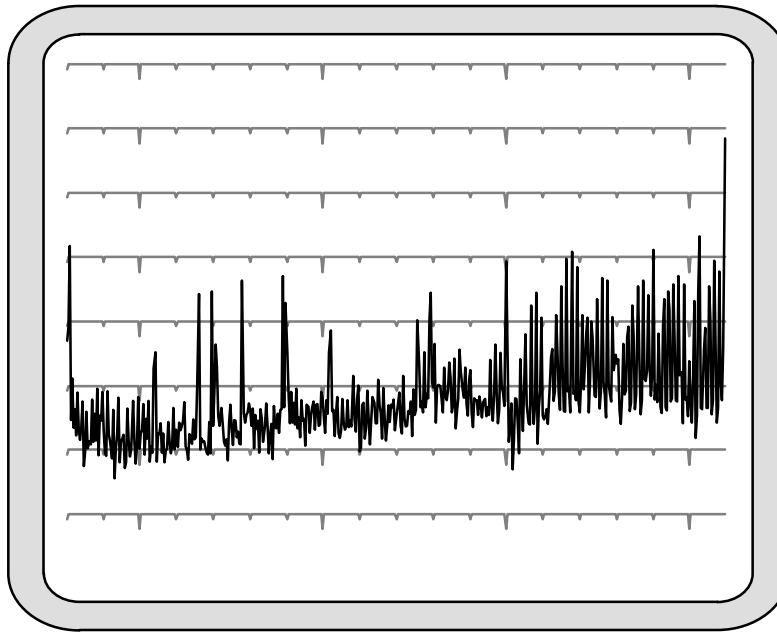


Figure 4-7: Example of a Noisy High Resolution “Pr” Spectrum

4.6.2 Available Subcommands Within “Pr”

The list of subcommands is printed on the TTY:

```
Available Subcommands within 'Pr':  
  L/l & R/r  Start range Left/Right  
    T/t      Plot time span Up/Dn  
    V/v      Number of spectra averaged  
    Z/z      Amplitude zoom  
  <space>   Alternate Plots  
    %       Toggle between dual receivers  
    .       Single Step
```

These subcommands change the start time and span of the IF sampling window, and alter the format of the display.

L/l & R/r

The “L” and “R” commands shift left and right the starting point of the window of IF samples. The lower case commands shift in 0.25 μ sec steps, and the upper case commands use 10 μ sec steps. The starting point is displayed both in microseconds and kilometers on the TTY, and is not allowed to be set earlier than range zero.

T/t

The “T” command increments or decrements the time duration of the window of IF samples. The window is not allowed to become shorter than the impulse response length of the FIR filter, since that would preclude calculating even a single LOG power point. The value is reported in microseconds on the TTY, and the largest permitted span is 50 μ sec.

V/v	The "V" command increments or decrements the number of power spectra that are averaged together to create the plot. The count ranges from one (no averaging) to 25, and is reported on the TTY as "Navg".
Z/z	The "Z" command zooms the amplitude of the IF samples by factors of two from one to 128. The LOG plots are shifted in corresponding 6dB increments as the amplitude is zoomed up and down. The zoom level is reported on the TTY so that absolute power levels and A/D usage can be assessed.
<space>	The space bar alternates among three choices for the type of data that are plotted: 1) Received Samples, 2) Received Samples and LOG Power, and 3) Received Power Spectrum.
%	In dual-receiver mode, the "%" command toggles between each receiver. The printed status line is prefixed with "Rx:Pri" or "Rx:Sec" according to which receiver is selected. Specifically, typing "%" will toggle the LOG plot of the received power, and the printout of the "Total", "Filtered", and "Midpoint" powers. However, the plots of power spectra and raw IF data samples are always based on the sum of all input signals, and thus do not change with "%".

4.6.3 TTY Information Lines Within "Pr"

The TTY information lines will resemble:

```
Zoom:x1, Navg:4, Start:0.00 usec (0.00 km), Span:5 usec  
Total:-63.3 dBm, Filtered:-77.6 dBm, MidSamp:-77.4 dBm
```

Zoom	Indicates the magnification (in amplitude) of the plotted samples. A zoom level of "x1" means that a full scale A/D waveform exactly fills the vertical height of the plot. Generally, the IF signal strength will not be quite this high. Thus, use larger zoom levels to see the weaker samples more clearly. You may zoom in powers of two up to x128.
Navg	Indicates the number of spectra and/or LOG powers that are averaged together prior to plotting. Larger amounts of averaging increase the ability to observe subtleties of the signals, but the display will update more slowly.
Start	Indicates the starting time of the IF sample window relative to range zero. The time is shown both in microseconds and in kilometers.
Span	Indicates the time span of the IF sample window in microseconds.
Total	Indicates the total RMS power that is being detected by the IF-Input A/D converters. This total is computed using all of the raw IF samples in the chosen interval, and is the sum of power at all frequencies other than 0 Hz (and its aliases).
Filtered	Indicates the RMS power that falls only within the passband of the FIR filter for the current pulsewidth. This is computed using all of the raw IF samples in the chosen interval.

MidSamp

Also indicates the RMS power within the passband of the FIR filter, but using only the raw IF samples in the exact center of the chosen interval.

The computation of “Total Power” is performed using the same subset of central IF samples that are used to compute “Filtered Power”. This smaller subset of IF samples comes about because filtering the data requires a convolution with the current FIR filter, and this computation does not produce results all the way to the edges of the input data. This is the same reason that the LOG plots do not extend across the full screen.

Because of this definition, it is valid to intercompare the “Total Power” and “Filtered Power”. The two numbers will match exactly as long as all of the incoming power falls within the passband of the FIR filter. The difference between the two powers can be used as a measure of the “filter loss” for a given pulse shape, i.e., the portion of signal that is lost outside of the filter’s passband.



Note: The “Total”, “Filtered”, and “MidSamp” values represent true RMS power (i.e., variance), and not merely a sum-of-squares. Thus, any DC offset present in the A/D converter will not affect these power levels.

5. Processing Algorithms (draft)

ⓘ **This draft chapter is based on the legacy RVP7 algorithms. The RVP8 will have some additional features and may not contain some of the legacy features.**

This chapter describes the real-time data processing algorithms implemented within the RVP8 signal processor. The discussion is confined to the mathematical description of these algorithms. Figure 5–1 shows the overall process by which the RVP8 converts the IF signal into corrected reflectivity, velocity, and width. Table 5–1 summarizes the quantities that are measured and computed by the RVP8. The type of the quantity (i.e., real or complex) is also given. Subscripts are sometimes used to denote successive samples in time from a given range bin. For example, s_n denotes the “I” and “Q” video sample from the n 'th pulse from a given range bin. In cases where it is obvious, the subscripts denoting the pulse (time) are dropped. The descriptions of all the data processing algorithms are phrased in terms of the operations performed on data from a single range bin — identical processing then being applied at all of the selected ranges. Thus, there is no need to include a range subscript in this data notation.

It is frequently convenient to combine two simultaneous samples of “I” and “Q” into a single complex number (called a phaser) of the form:

$$s = I + jQ$$

where “j” is the square root of -1 . Most of the algorithms presented in this chapter are defined in terms of the operations performed on the “s”s, rather than the “i”s and “q”s. The use of the complex terms leads to a more concise mathematical expression of the signal processing techniques being used. In actual operation, the complex arithmetic is simply broken down into its real-valued component parts in order to be computed by the RVP8 hardware. For example, the complex product:

$$s = W \times Y$$

is computed as

$$\begin{aligned} \text{Real}\{s\} &= \text{Real}\{W\} \text{Real}\{Y\} - \text{Imag}\{W\} \text{Imag}\{Y\} \\ \text{Imag}\{s\} &= \text{Real}\{W\} \text{Imag}\{Y\} + \text{Imag}\{W\} \text{Real}\{Y\} \end{aligned}$$

where “Real{ }” and “Imag{ }” represent the real and imaginary parts of their complex-valued argument. Note that all of the expanded computations are themselves real-valued.

In addition to the usual operations of addition, subtraction, division, and multiplication of complex numbers, we employ three additional unary operators: “|””, “Arg” and “*”. Given a number “s” in the complex plane, the magnitude (or modulus) of s is equal to the length of the vector joining the origin with “s”, i.e.

$$|s| = \left(\text{Real}\{s\}^2 + \text{Imag}\{s\}^2 \right)^{1/2}$$

The signed (CCW positive) angle made between the positive real axis and the above vector is:

$$\angle = \text{Arg}\{s\} = \arctan\left[\frac{\text{Imag}\{s\}}{\text{Real}\{s\}}\right]$$

where this angle lies between $-\pi$ and $+\pi$ and the signs of $\text{Real}\{s\}$ and $\text{Imag}\{s\}$ determine the proper quadrant. Note that this angle is real, and is uniquely defined as long as $|s|$ is non-zero. When $|s|$ is equal to zero, $\text{Arg}\{s\}$ is undefined. Finally, the “complex conjugate” of “ s ” is that value obtained by negating the imaginary part of the number, i.e.,

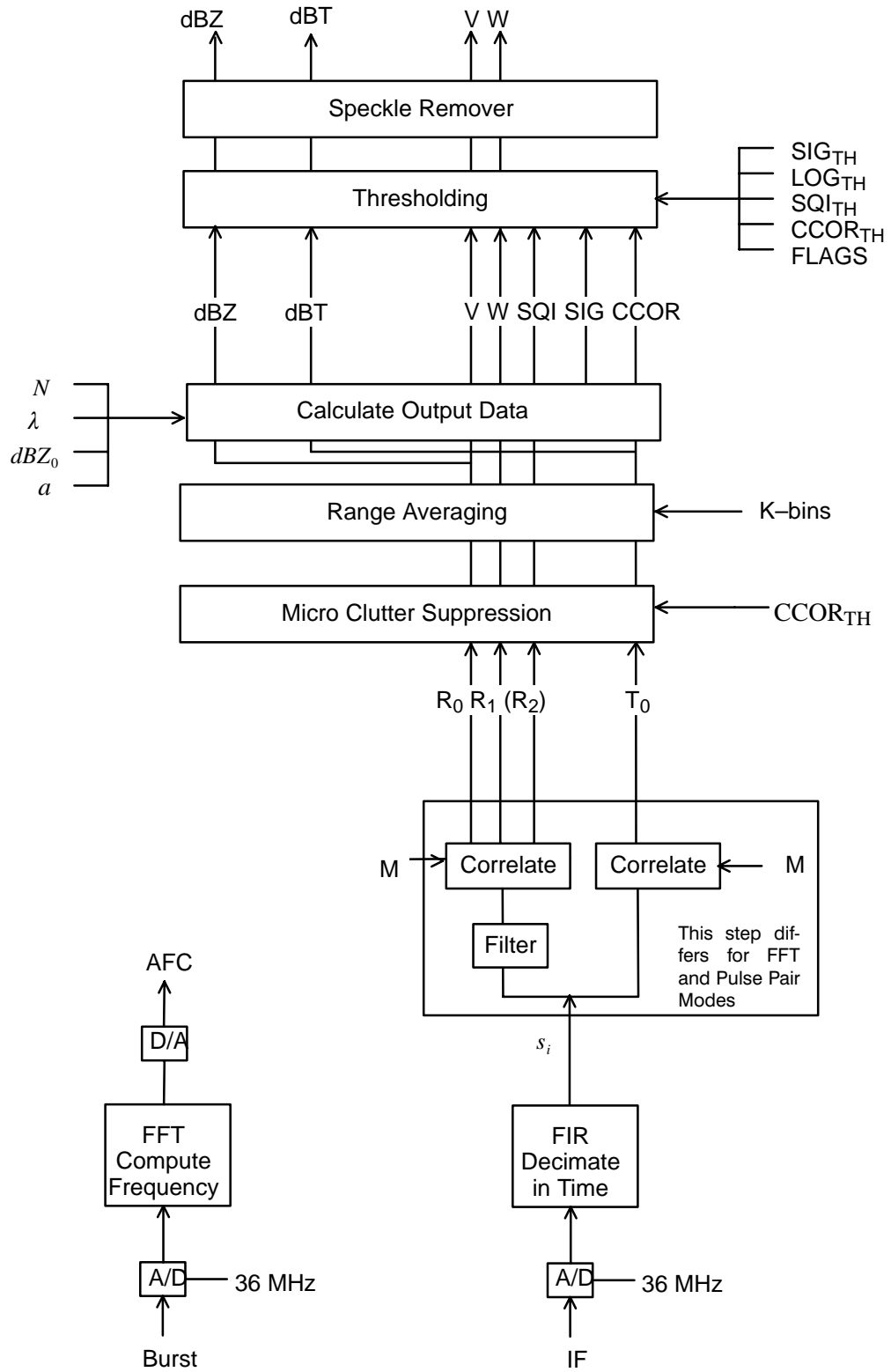
$$s^* = \text{Real}\{s\} - j \text{Imag}\{s\}.$$

Note that $\text{Arg}\{s^*\} = -\text{Arg}\{s\}$. The reader is referred to any introductory text on complex numbers for clarification of these points.

Table 5–1: Algebraic Quantities Within the RVP8 Processor

p	Instantaneous IF-receiver data sample	<i>Real</i>
b	Instantaneous Burst-pulse data sample	<i>Real</i>
I, Q	Instantaneous quadrature receiver components	<i>Real</i>
s	Instantaneous time series phaser value	<i>Complex</i>
s'	Time series after clutter filter	<i>Complex</i>
T_0	Zero th lag autocorrelation of A values	<i>Real</i>
R_0	Zero th lag autocorrelation of A' values	<i>Real</i>
R_1	First lag autocorrelation of A' values	<i>Complex</i>
R_2	Second lag autocorrelation of A' values	<i>Complex</i>
SQI	Signal Quality Index	<i>Real</i>
V	Mean velocity	<i>Real</i>
W	Spectrum Width	<i>Real</i>
CCOR	Clutter correction	<i>Real</i>
LOG	Signal to noise ratio for thresholding	<i>Real</i>
SIG	Signal power of weather	<i>Real</i>
C	Clutter power	<i>Real</i>
N	Noise power	<i>Real</i>
Z	Corrected Reflectivity factor	<i>Real</i>
T	UnCorrected Reflectivity factor	<i>Real</i>

Figure 5-1: Flow Diagram of RVP8 Processing



5.1 IF Signal Processing

The starting point for all computations within the RVP8 are the instantaneous IF-receiver samples p_n and, the instantaneous burst-pulse or COHO reference samples b_n . These data are available at a very high sampling rate (typically 36MHz), which makes possible the digital implementation of functions that are traditionally performed by discrete components in an analog receiver. The RVP8's all-digital approach replaces a great deal of analog hardware, avoids problems of aging and maintenance, and makes it easy to tune-up the receiver and alter its parameters.

This section describes these IF signal processing steps.

5.1.1 FIR (Matched) Filter

The RVP8 implements a digital version of the "matched" filter that is found in the traditional analog radar receiver. The equivalent Finite-Impulse-Response (FIR) filter is designed using an interactive graphical procedure described in Section 4.5. The filter length (number of taps), center frequency, and bandwidth are all adjustable. The design procedure computes two sets of filter coefficients f_n^i and f_n^q such that the instantaneous quadrature samples at a given bin are:

$$I = \sum_{n=0}^{N-1} f_n^i \times p_n, \quad Q = \sum_{n=0}^{N-1} f_n^q \times p_n$$

where N is the length of the filter. The input samples p_n are centered on the range bin to which the (I, Q) pair is assigned. Note that some of the p_n are likely to overlap among adjacent bins, i.e., the filter length may be chosen to be greater than the bin spacing. Such an overlap introduces a slight correlation between successive bins, but the longer length allows a better filter to be designed.

The convolution sums for I and Q are computed on the RVP8/Main board using dedicated FIR chips that can perform up to 576 million sums of products per second. The p_n are represented as 16-bit signed integers, and the f_n^i and f_n^q are represented as 10-bit signed integers. A numerical optimization procedure is used to quantize the ideal filter coefficients into their 10-bit hardware values. The overall spectral purity of the FIR filter will typically be greater than 66dBc.

The reference phase for each transmitted pulse is computed using the same two FIR sums, except with b_n substituted for the p_n . For a magnetron system the N b_n samples are centered on the transmitted burst; for a klystron system they are obtained from the CW COHO. If the klystron is phase modulated, then the samples should be from the modulated COHO.

The f_n^i coefficients are computed as:

$$f_n^i = l_n \times \sin \left[\frac{\pi}{4} + 2\pi \frac{f_{IF}}{f_{SAMP}} \left(n - \frac{N-1}{2} \right) \right], \quad n = 0 \dots N-1$$

where f_{IF} is the radar intermediate frequency, f_{SAMP} is the RVP8/IFD crystal sampling frequency, and l_n are the coefficients of an N-point symmetric low-pass FIR filter that is matched to the bandwidth of the transmitted pulse. The multiplication of the l_n terms by the $\sin()$ terms effectively converts the low-pass filter to a band-pass filter centered at the radar IF. The formula for the f_n^q coefficients is identical except that $\sin()$ is replaced with $\cos()$.

The phase of the sinusoid terms, and the symmetry of the l_n terms, has been carefully chosen to have a valuable overall symmetry property when n is replaced with $(N-1)-n$, i.e., the sequence is reversed:

$$f_{(N-1)-n}^i = l_{(N-1)-n} \times \sin\left[\frac{\pi}{4} + 2\pi \frac{f_{IF}}{f_{SAMP}} \left((N-1) - n - \frac{N-1}{2} \right)\right]$$

$$f_{(N-1)-n}^i = l_n \times \cos\left[\frac{\pi}{4} + 2\pi \frac{f_{IF}}{f_{SAMP}} \left(n - \frac{N-1}{2} \right)\right]$$

$$f_{(N-1)-n}^i = f_n^q$$

Thus, the coefficients needed to compute I are merely the reversal of the coefficients needed to compute Q ; if you know f_n^i , then you also know f_n^q . This is why it is sufficient to print only one set of FIR coefficients during the filter design process described in Section 4.5.

5.1.2 Automatic Frequency Control (AFC)

AFC is used on magnetron systems to tune the STALO to compensate for magnetron frequency drift. The STALO is typically tuned 30 MHz away from the magnetron frequency. The maximum tuning range of the AFC feedback is approximately 7MHz on each side of the center frequency. This assumes that the system's IF frequency is at least 4MHz away from any multiple of half the digital sampling frequency, i.e., 18, 36, 54, or 72MHz.

The RVP8 analyzes the burst pulse samples from each pulse, and produces a running estimate of the power-weighted center frequency of the transmitted waveform. This frequency estimate is the basis of the RVP8's AFC feedback loop, whose purpose is to maintain a fixed intermediate frequency from the radar receiver.

The instantaneous frequency estimate is computed using four autocorrelation lags from each set of N b_n samples. This estimate is valid over the entire Nyquist interval (e.g., 18MHz to 36MHz), but becomes noisy within 10% of each end. Since the span of the burst pulse samples is only approximately a microsecond, several hundred estimates must be averaged together to get an estimate that is accurate to several kiloHertz. Thus, the AFC feedback loop will typically have a time constant of several seconds or more.

Most of the burst pulse analysis routines, including the AFC feedback loop, are inhibited from running immediately after making a pulsewidth change. The center-of-mass calculations are held off according to the value of *Settling time (to 1%) of burst frequency estimator*, and the AFC loop is held off by the *Wait time before applying AFC*. This prevents introducing transients into the burst analysis algorithms each time the pulsewidth changes.

Additional information about using AFC can be found in Sections 2.2.10, 2.4, and 3.3.6.

5.1.3 Burst Pulse Tracking

The RVP8 has the ability to track the power-weighted center-of-mass of the burst pulse, and to automatically shift the trigger timing so that the pulse remains in the center of the burst analysis window of the **Pb** plot. This means that external sources of drift in the timing of the transmitted pulse (temperature, aging, etc.) will be tracked and nulled out during normal operation; so that fixed targets will remain fixed in range, and clean Tx phase measurements will always be available on every pulse.

The Burst Pulse Tracker feedback loop makes changes to the trigger timing in response to the measured position of the burst. Timing changes will generally be made only when the RVP8 is not actively acquiring data, in the same way that AFC feedback is held off for similar “quiet” times. However, if the center-of-mass has drifted more than 1/3 the width of the burst analysis window, then the timing adjustment will be made right away. Also, there will be an approximately 5ms interruption in the normal trigger sequence whenever any timing changes are made.

The Burst Pulse Tracker and AFC feedback loop are each fine-tuning servos that keep the burst pulse “centered” in time and frequency. These servos have been expanded to include a combined “Hunt Mode” that will track down a missing burst pulse when we are uncertain of both its time and frequency. This coarse-tuning mode is especially valuable for initializing the two fine-tuning servos in radar systems that drift significantly with time and temperature.

When the radar transmitter is *On* but the burst pulse is missing, it may be because either of the following have happened:

- It is misplaced in time, i.e., the Tx pulse is outside of the window displayed in the **Pb** plotting command. In this case, the trigger timing needs to be changed in order to bring the center of the pulse back to the center of the window.
- It is mistuned in frequency, i.e., the AFC feedback is incorrect and has caused the burst frequency to fall outside of the passband of the RVP8 anti-alias filters. In this case the AFC (or DAFC) needs to be changed so that proper tuning is restored.

The Hunt Mode performs a 2-dimensional search in time and frequency to locate the burst; searching across a $\pm 20\mu\text{sec}$ time window, and across the entire AFC span. If a valid Tx pulse (i.e., meeting the minimum power requirement) can be found anywhere within those intervals then the Burst Pulse Tracker and AFC loops will be initialized with the time and frequency values that were discovered. The fine servos then commence running with a good burst signal starting from those initial points.

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. The BPHUNT (Section 6.25) opcode allows the host computer to initiate Hunt Mode when it knows or can sense that a burst pulse should be present

5.1.4 Interference Filter

The interference filter is an optional processing step that can be applied to the raw (I, Q) samples that emerge from the FIR filter chips. The intention of the filter is to remove strong but sporadic interfering signals that are occasionally received from nearby man-made sources. The technique relies on the statistics of such interference being noticeably different from that of weather.

For each range bin at which (I, Q) data are available, the interference filter algorithm uses the received power (in deciBels) from the three most recent pulses:

$$P_{n-2}, P_{n-1}, \text{ and } P_n$$

where:

$$P_n = 10 \log_{10}(I_n^2 + Q_n^2).$$

If the three pulse powers have the property that:

$$|P_{n-1} - P_{n-2}| < C_1 \quad \text{and} \quad |P_n - P_{n-1}| > C_2 \quad (\text{Alg.1})$$

then (I_n, Q_n) is replaced by (I_{n-1}, Q_{n-1}) . Here C_1 and C_2 are constants that can be tuned by the user to match the type of interference that is anticipated, and the error rates that can be tolerated. For certain environments it may be the case that good results can be obtained with $C_1 = C_2$; but the RVP8 does not force that restriction.

This 3-pulse algorithm is only intended to remove interference that arrives on isolated pulses, and for which there are at least two clear pulses in between. Interference that tends to arrive in bursts will not be rejected.

Two variations on the fundamental algorithm are also defined. The CFGINTF command (Section 6.22) allows you to choose which of these algorithms to use, and to tune the two threshold constants. You may also do this directly from the **Mp** setup menu (Section 3.3.2).

$$|P_{n-1} - P_{n-2}| < C_1 \quad \text{and} \quad P_n - P_{n-1} > C_2 \quad (\text{Alg.2})$$

$$|P_{n-1} - P_{n-2}| < C_1 \quad \text{and} \quad P_n - \text{LinAvg}(P_{n-1}, P_{n-2}) > C_2 \quad (\text{Alg.3})$$

Where $\text{LinAvg}()$ denotes the deciBel value of the linear average of the two deciBel powers. The Alg.2 and Alg.3 algorithms also include the receiver noise level(s) as part of their decision criteria. Whenever power levels are intercompared in the algorithms, any power that is less than the noise level is first set equal to that noise level. This makes the filters much more robust and properly tunable, so that interference is more successfully rejected on top of blank receiver noise.

Optimum values for C_1 and C_2 will vary from site to site, but some guidance can be obtained using numerical simulations. The results shown below were obtained when the algorithms were applied to realistic weather time series having a spectrum width = 0.1 (Nyquist), SNR = +10dB, and an intermittent additive interference signal that was 16dB stronger than the weather. The interference arrived in isolated single pulses with a probability of 2%.

Performance of the three algorithms is summarized in the first three columns of Table 5–2, for which C_1 and C_2 have the common value shown. The fourth column also uses Algorithm #3, but with the value of C_1 raised by 2dB. The “Missed” rate is defined as the percentage of

interference points that manage to get through the filtering process without being removed. The “False” (false alarm) rate is the percentage of non-interference points that are incorrectly modified when they should have been left alone.

Table 5–2: Algorithm Results for +16dB Interference

C1, C2	Alg.1		Alg.2		Alg.3		Alg.3, C1+=2dB	
	Missed	False	Missed	False	Missed	False	Missed	False
6.0dB	17.8%	10.91%	17.8%	4.06%	17.8%	3.48%	10.3%	4.15%
8.0dB	10.5%	6.57%	10.5%	2.42%	10.4%	1.71%	6.1%	1.92%
9.0dB	8.5%	5.09%	8.5%	1.81%	8.3%	1.16%	5.4%	1.28%
10.0dB	7.3%	4.01%	7.3%	1.42%	7.5%	0.79%	5.4%	0.85%
11.0dB	8.9%	3.14%	8.9%	1.06%	8.3%	0.51%	6.5%	0.54%
12.0dB	11.6%	2.53%	11.6%	0.85%	11.3%	0.33%	9.9%	0.35%
13.0dB	17.0%	2.07%	17.0%	0.67%	16.3%	0.22%	15.3%	0.23%
14.0dB	23.5%	1.70%	23.5%	0.54%	22.4%	0.14%	21.6%	0.15%
16.0dB	39.2%	1.21%	39.2%	0.35%	39.6%	0.06%	38.9%	0.06%
20.0dB	67.3%	0.65%	67.3%	0.14%	72.5%	0.01%	72.4%	0.01%

It is important to minimize both types of errors. If too much interference is missed, then the filter is not doing an adequate job of cleaning up the received signal. If the false alarm rate is too high, then background damage is done at all times and the overall signal quality (especially sub-clutter visibility) may be compromised. We suggest that you try to keep the false alarm rate fairly low, perhaps below 1%; and then let the missed percentage fall where it may.

To summarize the numerical results in Table 5–2:

- The “Missed” rates of Alg.1 and Alg.2 are identical, but the “False” rate of Alg.1 is much higher. Alg.1 clearly does not perform as well for additive interference, but it is included in the suite for historical reasons.
- The “Missed” error rate for Alg.3 is nearly identical to that of Alg.2, but Alg.3 has a significantly lower false alarm rate. This is because of the somewhat improved statistics that result when the linear mean of P_{n-2} and P_{n-1} is used in the second comparison, rather than just P_{n-1} by itself. We recommend that Alg.3 generally be chosen in preference to the other two.
- Alg.3 can be further tuned by allowing the two constants to differ. For example, by raising C_1 slightly above C_2 (fourth column), we can trade off a decrease in the “Missed” rate for an increase in the “False” rate. Lowering C_1 would have the opposite effect.

Keep in mind that optimum tuning will depend on the type of interference you are trying to remove. In the previous example, where the interfering signal is only 16dB stronger than the weather, there was a close tradeoff between the “Missed” and “False” error rates. However, Table 5–3 shows the results that would be obtained if the interference dominates by 26db.

Table 5–3: Algorithm Results for +26dB Interference

C1, C2	Alg.1		Alg.2		Alg.3		Alg.3, C2+=5dB	
	Missed/False	Missed/False	Missed/False	Missed/False	Missed/False	Missed/False	Missed/False	Missed/False
6.0dB	17.8%	10.75%	17.8%	3.95%	17.8%	3.44%	17.8%	0.34%
8.0dB	9.9%	6.48%	9.9%	2.31%	9.9%	1.68%	9.9%	0.15%
9.0dB	7.4%	4.99%	7.4%	1.75%	7.4%	1.14%	7.4%	0.10%
10.0dB	5.9%	3.91%	5.9%	1.36%	5.9%	0.76%	5.9%	0.06%
11.0dB	4.8%	3.06%	4.8%	1.06%	4.8%	0.50%	4.8%	0.04%
12.0dB	3.2%	2.37%	3.2%	0.83%	3.2%	0.33%	3.2%	0.03%
13.0dB	2.6%	1.83%	2.6%	0.62%	2.6%	0.20%	2.8%	0.01%
14.0dB	1.9%	1.45%	1.9%	0.50%	1.9%	0.12%	2.6%	0.01%
16.0dB	1.3%	0.90%	1.3%	0.30%	1.3%	0.05%	5.8%	0.00%
20.0dB	3.1%	0.39%	3.1%	0.12%	2.0%	0.01%	31.5%	0.00%

Notice that we can now re-tune the constants and operate with $C_1 = 13dB$ and $C_2 = 18dB$ (fourth column); which yields a low 2.8% “Missed” rate, and an extremely low 0.01% false alarm rate. Since the false alarm rate is (approximately) independent of the interference power, these filter settings would leave all “clean” weather virtually untouched, i.e., we would have a very safe filter that is intended only to remove fairly strong interference. Such a filter could be left running at all times without too much worry about side effects.

5.1.5 Large-Signal Linearization

The RVP8 is able to recover the signal power of targets that saturate the IF-Input A/D converter by as much as 4–6 deciBels. This is possible because an overdriven IF waveform still spends some of its time in the valid range of the converter, and thus, it is still possible to deduce information about the signal.

Figure 5–2 shows actual signal generator test measurements with normal A/D saturation (lower line), and with the extrapolation algorithms turned on (upper line). The high-end linear range begins to roll off at approximately +10dBm versus +5dBm, and thus has been extended by 5dB.

5.1.6 Correction for Tx Power Fluctuations

The RVP8 can perform pulse-to-pulse amplitude correction of the digital (I,Q) data stream based on the amplitude of the Burst/COHO input. The technique computes a (real valued) correction factor at each pulse by dividing the mean amplitude of the burst by the instantaneous amplitude of the burst. The (I,Q) data for that pulse are then multiplied by this scale factor to obtain corrected time series. The amplitude correction is applied after the Linearized Saturation Headroom correction.

The mean burst amplitude is computed by an exponential average whose ($1/e$) time constant is selected as a number of pulses (See Section 3.3.2). A short time constant will settle faster, but will not be as thorough in removing amplitude variations (since the mean itself will be varying). Longer time constants do a better job, but will require a second or two before valid data is available when the transmitter is first turned on. The default value of 70 will give excellent results in almost all cases.

Whenever the RVP8 enters a new internal processing mode (time series, FFT, PPP, etc.), the burst power estimator is reinitialized from the level of the first pulse encountered, and an additional pipeline delay is introduced to allow the estimator to completely settle. Thus, valid

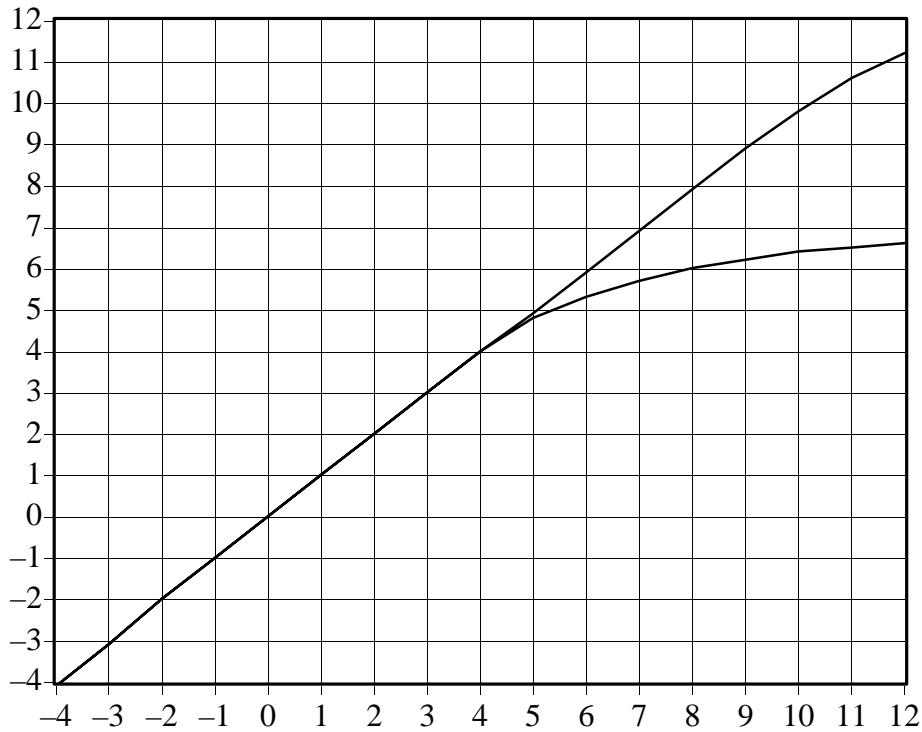


Figure 5-2: Linearization of Saturated Signals Above +4dBm

corrected data are produced even when the RVP8 is alternating rapidly between different data acquisition tasks, e.g., in a multi-function ASCOPE display. The additional pipeline delay will not affect the high-speed performance when the RVP8 runs continuously in any single mode.

For amplitude correction to be applied, the instantaneous Burst/COHO signal level must exceed the minimum valid burst power specified in the “Mb” setup section. If that level is not met, e.g., if the transmitter is turned off, then no correction is performed. Thus, the amplitude correction feature conveniently “gets out of the way” when receiver-only tests are being performed.

The maximum correction that will ever be applied is ± 5 dB. If the burst power in a given pulse is more than 5dB above the mean, or less than 5dB below it, then the correction is clamped at those limits. The power variation of a typical transmitter will easily be contained within this interval (it is typically less than 0.3dB).

Instantaneous amplitude correction is a unique feature of the RVP8 digital receiver. Bench tests with a signal generator reveal that an amplitude modulated waveform having 2.0dB of pulse-to-pulse variation is reduced to less than 0.02dB RMS of (I,Q) variation after applying the amplitude correction.

5.2 Video (“I” and “Q”) Signal Processing

This section describes the processing of the video (“I” and “Q”) data to obtain the reduced parameters: reflectivity, total power, velocity, width, signal quality index, clutter power correction, and sometimes ZDR. The RVP8 employs two methods (selectable) for processing the I and Q signals: pulse-pair and FFT. The methods are similar except in regard to the procedures for clutter filtering. The pulse pair methods are described below; the FFT clutter filtering algorithms are described in Section 5.8.

5.2.1 Time Series

Recall that the time series synthesized by the FIR filter consist of an array of complex numbers:

$$s_n = [I_n + jQ_n] \quad \text{for } n = 1, 2, 3, \dots, M$$

where “j” is $-1^{1/2}$. These data samples are analogous to the “I” and “Q” samples in a traditional analog receiver. They are sampled at a selectable resolution in the range 50–133 meters. The time series are the starting point for all calculations performed within the RVP8.

5.2.2 IIR Clutter Filter for PPP-Mode

The RVP8’s pulse-pair-processing mode employs a 4th order Infinite Impulse Response (IIR) digital high pass filter to remove low frequency signals due to ground clutter from the time series. Since the width of the clutter can change with the antenna rotation rate, eight different filters (seven high-pass, plus one all-pass) are provided. The filter stop-bandwidths vary from approximately 2% to 14% of the Nyquist interval, and stop band attenuation is at least 40 dB. A setup question allows selection of either 40 dB or 50 dB filters. The 50 dB filters are intended for Klystron systems. Any of the eight filters can be selected independently at each individual range bin. This permits range-dependent clutter removal. The filter algorithm is outlined below.

The input time series s_n is processed to form a filtered output time series s'_n as follows:

$$s'_n = B_0s_n + B_1s_{n-1} + B_2s_{n-2} + B_3s_{n-3} + B_4s_{n-4} \\ - C_1s'_{n-1} - C_2s'_{n-2} - C_3s'_{n-3} - C_4s'_{n-4}$$

where the B’s and C’s are the filter coefficients. Appendix C gives the magnitude response plots for the set of filters supplied with the RVP8.

5.2.3 Autocorrelations for PPP-Mode

The autocorrelations are computed during pulse-pair-processing mode according to the following algorithms (corresponding physical models are also given):

Parameter and Definition	Physical Model
$T_o = \frac{1}{M} \sum_{n=1}^M s_n^* s_n$	$g^r g^t (S + C) + N$
$R_o = \frac{1}{M} \sum_{n=1}^M s'_n{}^* s'_n$	$g^r g^t S + N$
$R_1 = \frac{1}{M-1} \sum_{n=1}^{M-1} s'_n{}^* s'_{n+1}$	$g^r g^t S e^{j \pi V' - \pi^2 W^2/2}$
$R_2 = \frac{1}{M-2} \sum_{n=1}^{M-2} s'_n{}^* s'_{n+2}$	$g^r g^t S e^{j 2\pi V' - 2\pi^2 W^2}$

where M is the number of pulses in the time average. Here, s' denotes the filtered time series, s denotes the original unfiltered time series and the $*$ denotes a complex conjugate. g^r and g^t represent the transmitter and receiver gains, i.e., their product represents the total system gain. Since the RVP8 is a linear receiver, there is a single gain number that relates the measured autocorrelation magnitude to the absolute received power. However, since many of the algorithms do not require absolute calibration of the power, the gain terms will be ignored in the discussion of these. T_o for the unfiltered time series is proportional to the sum of the meteorological signal S , the clutter power C and the noise power N . R_o is equal to the sum of the meteorological signal S and noise power N which is measured directly on the RVP8 by periodic noise sampling. T_o and R_o are used for calculating the dBZ values- the equivalent radar reflectivity factor which is a calibrated measurement. The physical models for R_o , R_1 and R_2 correspond to a Gaussian weather signal and white noise. W is the spectrum width and V' the mean velocity, both for the normalized Nyquist interval $[-1$ to $1]$.

The exact value of M that is used for each time average will generally be the "Sample Size" that is selected by the SOPRM command (See Section 6.3). However, when the RVP8 is in PPP mode and antenna angle synchronization is enabled, the actual number of pulses used may be limited by the number that fit within each ray's angular limits at the current antenna scan rate. The value of M will never be greater than the SOPRM Sample Size, but it may sometimes be less. For example, at 1KHz PRF, 20°/sec scan rate, 1° ray synchronization, and a Sample Size of 80, there will be 50 pulses used for each ray (not 80). Note, however, that the number of pulses used in the "batched" (non-PPP) modes will always be exactly equal to the Sample Size, since those modes are allowed to use overlapping pulses.

5.2.4 Range averaging and Clutter Microsuppression

The next step (optional) is to perform range averaging. Range averaging can be performed over 2, 3, ..., 16 bins. This is accomplished by simply averaging the T_0 , R_0 , R_1 and R_2 values. This reduces the number of bins in the final output to save processing both in the RVP8 and in the host computer.

At the user's option, the range averaged data can be restricted to include only those bins which have an estimated clutter-to-signal ratio that falls within the CCOR threshold interval. By excluding isolated point clutter targets from the range average the sub-clutter visibility of the averaged data is increased. Specifically, the Doppler test that is applied to each bin in order that it contribute to the overall sum is:

$$10 \log R_0 - 10 \log T_0 > CCOR_{thresh} .$$

5.2.5 Reflectivity

The corrected reflectivity Z is output using a log scale based on the following equation:

$$dBZ = 10 \log \left[\frac{T_0 - N}{N} \right] + dBZ_o + 20 \log r + ar + CCOR$$

This equation is simply a dB version of the familiar radar equation for distributed targets. The relationship between the measured autocorrelation function, the received signal and the noise can be expressed as:

$$T_o = g^t g^r S + N$$

where g^t and g^r represent the transmitter and receiver gains, S is the average backscattered power from the targets and N is the measured average noise power. Neglecting attenuation and the contribution of ground clutter (for the moment), the radar equation can be written as.

$$Z = CSr^2 = \left[\frac{Cr_o^2 N}{g^r g^t} \right] \left[\frac{r^2}{r_o^2} \right] \left[\frac{T_o - N}{N} \right]$$

where C is the radar constant and r_o is a reference range which we will later set to 1 km. This is identical to the first three terms of the dB version of the equation with the definition that:

$$Z_o = \frac{Cr_o^2 N}{g^r g^t} = Cr_o^2 I_o \quad \text{where} \quad I_o = \frac{N}{g^r g^t}$$

Z_o is called the calibration reflectivity factor. It is the equivalent radar reflectivity factor at the reference range when the return signal power is equal to the noise power (SNR=0 dB). It is sometimes called the minimum detectable dBZ at 1 km. The parameter I_o is the measured noise power at IF with appropriate calibration for the system gain. Calibration of the RVP8 involves defining the radar constant C and measuring the value of I_o . This is discussed in detail in Section 5.4.

Essentially, the measurement of I_o is based on the measurement of the system noise at the time of calibration. However, if the receiver gain were to change after calibration, the use of periodic noise sampling properly corrects for this. For example, if the receiver gain were to change by a factor k, then we would measure a noise value of kN and an autocorrelation value of kT_o , i.e.,

$$Z = CSr^2 = \left[\frac{Cr_0^2N}{g^r g^t} \right] \left[\frac{r^2}{r_0^2} \right] \left[\frac{k T_o - k N}{k N} \right]$$

Thus the k's cancel to give us the same result for Z. This makes the approach robust to system gain fluctuations. Another way of saying this is that as long as the system sensitivity (noise figure) does not change, then the system does not require re-calibration.

The individual terms in the dB form of the equation are summarized below.

1st Term : $10 \log \left[\frac{T_0 - N}{N} \right]$: Signal to Noise Ratio

The effect of this term is to subtract the measured noise. It is also used for LOG thresholding. If this number is above the user input value LOG_{thresh} the dBZ is passed.

2rd Term: dBZ_o : Calibration Reflectivity (see discussion above)

dBZ_o is the minimum detectable dBZ at a reference range $r_o=1$ km,

3th Term: $20 \log r$: Range Normalization

This term is the $\left[\frac{r}{r_o} \right]^2$ range normalization expressed in dB form.

4th Term: ar : Gaseous Attenuation Correction

This term accounts for gaseous attenuation. The constant a is set in the RVP8 EEROM since it is a function of wavelength. For a C-band system the default value is 0.016 dB per km (for two-way path attenuation).

5th Term: CCOR: Clutter Correction

This term corrects for the measured ground clutter. It's derivation is discussed in section 5.2.9.

5.2.6 Velocity

For a Doppler power spectrum that is symmetric about its mean velocity, the velocity is obtained directly from the argument of the autocorrelation at the first lag, i.e.,

$$V = \frac{\lambda}{4\pi\tau_s} \theta_1 \quad \text{where} \quad \theta_1 = \arg \{R_1\}.$$

λ is the radar wavelength, τ_s is the sampling time (1/PRF). θ_1 is constrained to be on the interval $[-\pi, \pi]$. When $\theta_1 = \pm \pi$, then $V = \pm V_u$ where the unambiguous velocity is ,

$$V_u = \frac{\lambda}{4\tau_s}.$$

If the absolute value of the true velocity of the scatterers is greater than V_u , then the velocity calculated by the RVP8 is folded into the interval $[-V_u, V_u]$, which is called the Nyquist interval. Folding is usually easily recognized on a color display by a discontinuous jump in velocities. For example, if the true velocity is $V_u + \Delta V$, then the velocity calculated by the RVP8 is $-V_u + \Delta V$, which is $2V_u$ away from the true mean velocity.

For 8-bit outputs, rather than calculating the absolute velocity in scientific units, the RVP8 calculates the mean velocity for the normalized Nyquist interval $[-1, 1]$, i.e., the output values are,

$$V' = \frac{\theta_1}{\pi}.$$

For example, an output value of -0.5 corresponds to a mean velocity of $-V_u/2$. The normalized velocity V' is more efficient use of the limited number of bits.

5.2.7 Spectrum Width Algorithms

The spectrum width is a measure of the combined effects of shear and turbulence. To a lesser extent, the antenna rotation rate can also effect the spectrum width. At high elevation angles, the fall speed dispersion of the scatterers also effects spectrum width.

There are two choices for the spectrum width algorithm used in the RVP8, depending on the speed and accuracy that are required for the application:

R_0, R_1 "fast" algorithm valid when $\text{SNR} \gg 10$ dB

R_0, R_1, R_2 "accurate" algorithm for $\text{SNR} \gg 0$ to 5 dB

The approach used is selected in the SOPRM command. The two approaches are described below:

R_0, R_1 Width Algorithm

Given samples of the Doppler autocorrelation function, numerous estimates of spectral variance can be computed (Passarelli & Siggia, 1983). The particular estimator used by the RVP8 employs the magnitudes of R_0 and R_1 and assumes that the Doppler spectrum is Gaussian (usually an acceptable assumption) and that the signal-to-noise ratio is large. Specifically we have (similar to Srivastava, et al 1979):

$$\text{Variance} = 2 \ln \left[\frac{R_0}{|R_1|} \right] = -2 \ln[SQI]$$

where “ln” represents the natural logarithm. This can be compared to the expression in the preceding section for SQI to illustrate that this expression for the variance is only valid when:

$$\frac{SNR}{SNR + 1} \approx 1$$

which occurs when the SNR is large.

This variance estimator is normalized to the Nyquist interval in units of $[-\pi, \pi]$. Thus, for example, a variance of $\pi^2/25$ would be obtained from a Gaussian spectrum having a standard deviation equal to one fifth of the total width of the plotted spectral distribution. For scientific purposes, the spectrum width (standard deviation) is more physically meaningful than the variance, since it scales linearly with the severity of wind shear and turbulence. For these reasons, the width W is output by the RVP8:

$$W = \frac{\sqrt{\text{Variance}}}{\pi}$$

Again, for efficient packing in 8-bits, width is normalized to the Nyquist interval $[-1, 1]$. For the example given above, the output width W would be $(1/5)$. To obtain the width in meters per second, one multiplies the output width by V_u .

R_0, R_1, R_2 Width Algorithm

The width algorithm in this case is similar except that the addition of R_2 extends the validity of the width estimates to weak signals. In this case the variance is:

$$\text{Variance} = \frac{2}{3} \ln \left[\frac{|R_1|}{|R_2|} \right]$$

The output width W is then defined as in the previous section.

5.2.8 Signal Quality Index (SQI threshold)

An important feature of the RVP8 is its ability to eliminate signals which are either too weak to be useful, or which have widths too large to justify further analysis. This is done via the signal quality index (SQI) which is defined as:

$$SQI = \frac{|R_1|}{R_0}$$

The SQI is the normalized magnitude of the autocorrelation at lag 1 and varies between 0 for an uncorrelated signal (white noise) to 1 for a noise-free zero-width signal (pure tone). Mean velocity estimates are degraded when the spectrum, width is large or when the signal-to-noise ratio is weak. The SQI is a good measure of the uncertainty in the velocity estimates and is a convenient screening parameter to compute. In terms of the Gaussian model, the SQI is :

$$SQI = \frac{SNR}{SNR + 1} e^{-\frac{\pi^2 W^2}{2}}$$

where the SNR is the signal-to-noise ratio. For very large SNR's the SQI is a function of the spectrum width only. For a zero-width pure tone ($W=0$), the SQI is a function of the SNR only (e.g., for $W=0$, an SNR of 1 corresponds to $SQI=0.5$). The SQI threshold is typically set to a value of 0.4 to 0.5.

5.2.9 Clutter Correction (CCOR threshold)

In addition to calculating the R_0 , R_1 and optional R_2 autocorrelation terms, which are based on filtered time series data, the RVP8 also computes T_0 which is the total unfiltered power. By comparing the total filtered and unfiltered powers at each range bin, a clutter power, and hence a clutter correction, for that bin can be derived. The clutter correction is defined as,

$$CCOR = 10 \log \frac{S}{C + S} = 10 \log \frac{1}{CSR + 1}$$

where S is the weather signal power, C is the clutter power and CSR is the clutter-to-signal ratio. The algorithm for calculating CCOR depends on whether the optional R_2 autocorrelation lag is computed as described below.

R_0, R_1 Clutter Correction

In this case CCOR is estimated from,

$$\begin{aligned} CCOR_{est} &= 10 \log \left[\frac{R_0}{T_0} \right] \\ &= 10 \log \left[\frac{S + N}{C + S + N} \right] = 10 \log \left[\frac{1 + \frac{1}{SNR}}{CSR + 1 + \frac{1}{SNR}} \right] \end{aligned}$$

Here, the expression is strictly valid only when the signal-to-noise ratio (SNR=S/N) is large. Thus when the 2-lag approach is used, the clutter corrections are not as accurate for weak weather signals. However, the error is typically less than 3 dB.

R_0, R_1, R_2 Clutter Correction

In this case there is enough information to compute the clutter signal and noise power independently. The algorithm for CCOR is:

$$CCOR_{est} = 10 \log \frac{S}{C + S} = 10 \log \frac{1}{CSR + 1}$$

The clutter power is computed from:

$$C = T_o - R_o = [C + S + N] - [S + N]$$

The signal power S is then computed from:

$$S = |R_1| \exp \frac{\pi^2 W^2}{2}$$

W is the width that has been previously calculated. This approach yields more accurate results for the clutter correction in the case of a low SNR.

5.2.10 Weather Signal Power (SIG threshold)

A parameter called SIG is also calculated to provide an estimate of the weather signal-to-noise ratio in dB for thresholding. The SIG calculation is different depending on whether the optional R_2 autocorrelation is computed.

R_0, R_1 Calculation

In this case the SIG is computed as follows:

$$SIG = 10 \log \left[\frac{T_0 - N}{N} \right] + CCOR$$

This term represents the SNR after the removal of clutter. The **CCOR** value is the one described for R_0, R_1 in the previous section.

R_0, R_1, R_2 Calculation

In this case the SIG is computed based on the SNR which is:

$$SIG = 10 \log \left[\frac{2\pi S}{R_0 - 2\pi S} \right]$$

where the signal power S is determined as described in the preceding section.

5.2.11 Signal to Noise Ratio (LOG threshold)

A parameter called LOG is also calculated to provide an estimate of the total signal-to-noise ratio in dB useful for reflectivity thresholding. The formula is below:

$$LOG = 10 \log \left[\frac{T_0 - N}{N} \right]$$

5.3 Thresholding

An important feature of the RVP8 is its ability to accept or reject incoming data based on derived properties of the signals themselves. Typically, “rejected” data are not displayed by the user’s software, thus making for very clean weather presentations.

5.3.1 Threshold Qualifiers

For data quality control, each RVP8 output parameter can be qualified, i.e., either accepted or rejected for output, based on four threshold criteria:

ID	Criterion Name	Pass Criterion
LOG	Signal-to-Noise Ratio	LOG > threshold
SQI	Signal Quality Index	SQI > threshold
CCOR	Clutter Correction	CCOR > threshold
SIG	Weather Signal Power	SIG > threshold

The calculation of the measured levels (e.g., SQI) for each of these qualifications has been described in previous sections of this chapter. All four qualification criteria can be switched on and off independently, and the threshold levels (e.g., SQI_{thresh}) can each be set independently. Further, each qualifier test can be AND’d and OR’d with any other. This allows very complex thresholds criteria to be constructed as required. The four threshold qualifiers are summarized below.

- LOG It is essentially a measure of the total power SNR. This is usually used for thresholding of the reflectivity data. The default LOG threshold value is 0.5 dB.
- SQI The SQI threshold is typically used for velocity and width thresholding since it is a measure of the coherency. It is a number between 0 and 1 (dimensionless) where 0 is perfect white noise and 1 is a pure tone (perfect Doppler signal). The default SQI threshold value is 0.5.
- CCOR The clutter correction threshold is typically used to reject measurements when the clutter in a range bin is very strong (i.e., when the calculated CCOR is a large negative number in dB). The appropriate value depends on the coherency of the radar system. The default threshold is set to -25 dB. Threshold values less than this (more negative) reject fewer clutter bins. Threshold values closer to zero reject more clutter bins.
- SIG This is typically used only for thresholding the spectrum width to assure that the signal power is strong enough for an accurate width measurement. The default threshold value is 10 dB. If R_2 processing is used, this can usually be reduced to 5 dB for width thresholding.

The following are the default threshold combinations for each of the parameters that can be selected for output from the RVP8:

Parameter	Description	Threshold
dBZ	Reflectivity with clutter correction	LOG and CCOR
dBT	Reflectivity without clutter correction	LOG
V	Mean velocity	SQI and CCOR
W	Spectrum width	SQI and CCOR and SIG
ZDR	Differential reflectivity	LOG

5.3.2 Adjusting Threshold Qualifiers

The effect of the various threshold qualifiers for each output parameter are discussed in this section. In optimizing thresholds for your application, it is recommended that you change only one parameter (level or criterion) at a time so that you can verify the effect. Some hints for optimizing the levels for the default criteria are provided below:

- LOG To optimize the LOG level, display dBT or dBZ and select the lowest value of the threshold that eliminates the display noise. If the LOG level is set too high you lose sensitivity. Note that if you average more pulses or ranges, then the threshold level can usually be reduced.
- SQI To optimize the SQI level, display velocity and select the lowest value of the threshold that eliminates the display noise. If the SQI level is set too high you lose sensitivity. In general, you should see a greater area covered by velocity than reflectivity since the velocity is more sensitive. If you do not, you should reduce your SQI threshold. Note that if you average more pulses or ranges, then the threshold level can usually be reduced.
- CCOR This is used to eliminate clutter targets that are very strong. It should not be set to eliminate all clutter targets on a clear day since this means that you are losing sensitivity. To optimize the CCOR threshold it is best to know your system coherency in terms of dB of clutter cancelation. Start at a value of 10 dB greater (closer to 0) than this. Now display a PPI of dBZ at an antenna elevation of ~1 degree. The display should be relatively clean of any clutter targets since most will be rejected. Now reduce the CCOR (more negative) to increase the number of clutter targets on the display until the number of clutter targets does not increase. The optimum value of the CCOR is approximately 5 dB more (closer to zero) than this point. For example, if the number of clutter targets is a maximum at -35 dB, then set the CCOR to ~-30 dB. Note that your clutter filter selection will effect the result.
- SIG This should be done last. To optimize the SIG level, display the width W and select the lowest value of the threshold that eliminates the display noise. If the SIG level is set too high you lose sensitivity. Note that if you average more pulses or ranges, then the threshold level can usually be reduced.

When thresholding dBZ and dBT reflectivity data with SQI, the comparison value for accepting those data is the secondary SQI threshold that is defined via a slope and offset from the primary user value (see **Mf** command). This secondary threshold is more permissive (lower valued), and

is traditionally used to qualify LOG data only in the Random Phase processing mode. But the secondary SQI threshold is applied uniformly in all processing modes whenever reflectivity data are specified as being thresholded by SQI.

This gives you more freedom in applying an SQI threshold to your LOG data, because the cutoff value for reflectivity can be chosen independently from the cutoff value for the other Doppler parameters. The full SQI test would not normally be applied to LOG data because of the so-called "black hole" problem, i.e., loss of LOG data within regions of high shear, even though the reflectivity itself was strong. You may experiment with applying a secondary SQI threshold to help cleanup the LOG data, but without introducing any significant black holes.

5.3.3 Speckle Filters

A speckle filter is a final pass over each output ray, wherein isolated bins are removed. There are two speckle removers in the RVP8.

- 1D single-ray speckle filter. This can be used for any output parameter.
- 2D 3x3 speckle filter. If enabled, this is applied only to T, V, Z and W.

The 1D speckle filter is the default technique. The 2D 3x3 filter is enabled by selection in the mp TTY setups:

2D Final Speckle/Unfold "User" or "Always"

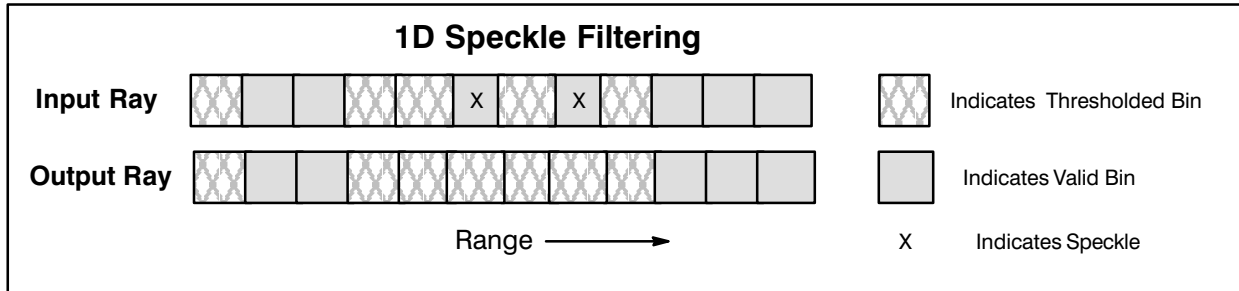
Both of these speckle filters remove isolated data points that are likely to be noise, interference, aircraft, birds or other point targets. Meteorological targets typically occupy multiple range bins so are not effected by the speckle filter. There are two primary benefits derived from using a speckle filter:

- Displays look "cleaner" to observers.
- Thresholds can be set slightly more sensitive without increasing the number of noise pixels.

The 2D 3x3 filter actually performs data filling of "missing speckles" as well as eliminating isolated speckle bins. The two algorithms are discussed below.

1D Speckle Filter

A ray is the basic azimuth unit of the RVP8 (e.g., 1 degree) over which the samples are averaged to obtain the output base data (T, Z, V, W). For this filter, a speckle is defined as any single, valid bin (not thresholded), having thresholded bins on either side of it in range. Any such isolated bin in a ray is set to “threshold”. The algorithm is shown schematically below.



Note that there are two independent 1D speckle removers– one for the reflectivity data (dBT, dBZ and ZDR) and one for the Doppler data (V and W). Each one should be switched on or off, depending on the specific nature of the targets being observed. For example, when making a clutter map of the area, one would certainly want to switch both speckle filters off.

2D 3x3 Speckle Filter

The 2D filter examines three adjacent range bins from three successive rays in order to assign a value to the center point. Thus, for each output point, its eight neighboring bins in range and time are available to the filter. Only the dBZ, dBT, Vel , and Width data are candidates for this filtering step; all other parameters are processed using the default 1D speckle filter.

The rules for the filter are as follows:

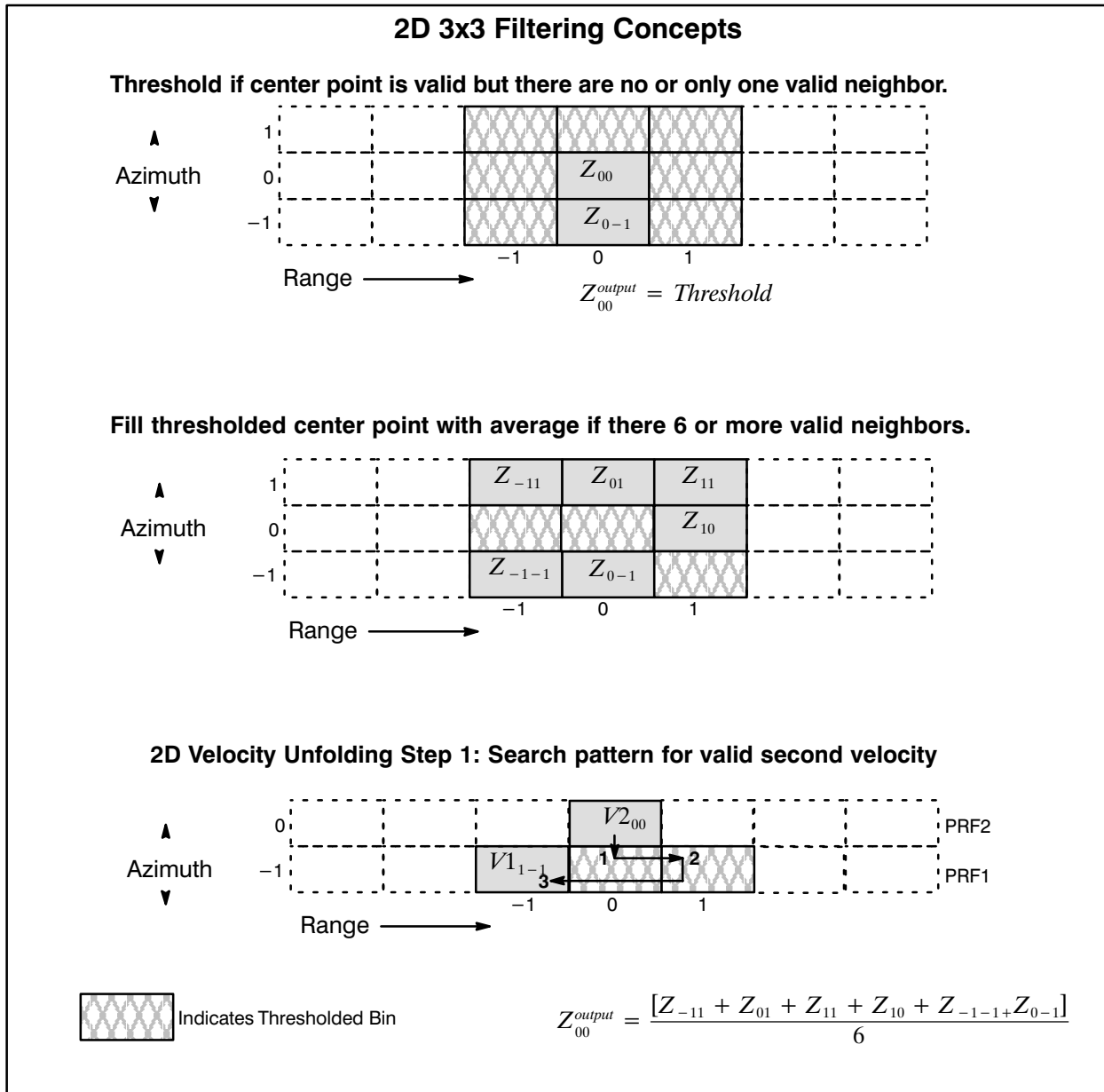
	Center Point Action	
	Assign Threshold	Else
Valid Center Point	If there are no or only one other valid point in the 3x3.	Do Nothing. Pass the center point value as-is.
Thresholded Center Point	If there are 5 or fewer valid neighbors in the 3x3.	If there are 6 or more valid neighbors in the 3x3, average to fill the center point.

Thus the 2D 3x3 filter performs 2 functions:

- Filling by interpolation.
- Thresholding of isolated noise bins.

Some examples are shown graphically in the figure below.

For dBZ, dBT, and Width, the interpolated value for filling is computed as the arithmetic average of all available neighbors. For Vel , it is not possible to define a meaningful average in a simple way; so the nearest valid neighbor is simply filled in.



The filter has some interesting properties when combined with other algorithms.

Dual PRF Unfolding

Dual-PRF velocity unfolding is computed within the 3x3 filter whenever both are enabled. There are two steps to the process:

- Step 1: The most recent and the previous ray are used. For every valid point in the most recent ray, the algorithm performs a search among the three nearest neighbors in the previous ray to find a valid velocity. The search pattern is shown at the bottom of the previous figure. This larger selection of alternate-PRF bins

makes it more likely that the algorithm will find the pairs of Low/High PRF data that are required for unfolding.

- Step 2: The unfolded velocities are then subjected to the standard 3x3 filtering.

Dual PRF, Random Phase Processing

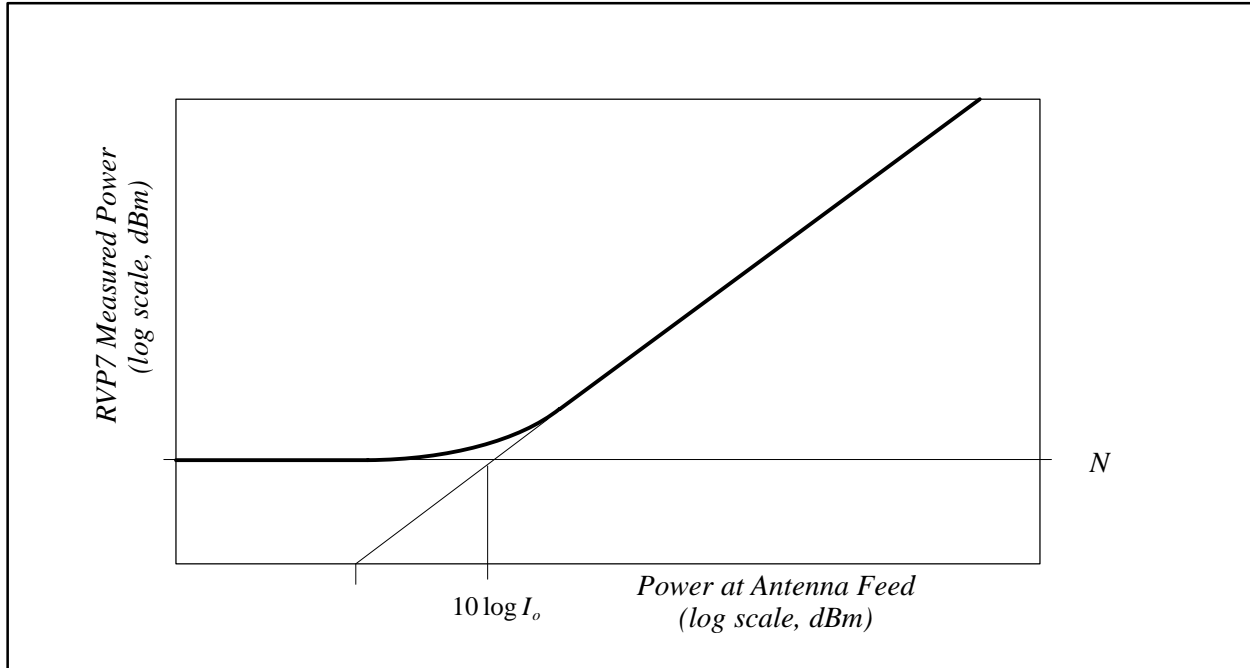
In random phase processing, the “seam” at the start of the second trip is always problematic since the transmitter main bang and nearby clutter will virtually always wipe-out the first few 2nd trip range bins. At a constant PRF the 2nd trip seam is always at the same range, but in dual PRF random phase mode, the seam is different each ray. Thus thresholded bins at the seam of the high PRF can be surrounded on either side by valid bins taken at the low PRF. The 3x3 filter has the effect of interpolating the reflectivity and width data over the bins at the 2nd trip seam. Velocity data will also be filled-in using the nearest neighbor. Thus the 2D filter mitigates much of the damage that is caused at the 2nd trip seam to make a nearly seamless display.

The maximum speed of the RVP8 is reduced to approximately 85000 bins/second when the 3x3 filter is ON— approximately 60% of its maximum throughput when the filter is OFF. This is still a rather large value, and should not affect most customers. For example, there would be no problem running a scan having 2048 bins at 1-degree resolution and a 40 deg/sec scan rate. However, if you really need to operate at the absolute upper limit of the RVP8 s throughput, then the 3x3 filter should be disabled in the Mp menu.

5.4 Reflectivity Calibration

The calculation of reflectivity described in section 5.2.5 required the calibration reflectivity dBZ_o . This section describes its derivation. Note that customers with the SIGMET IRIS system can use the **zauto** utility to perform the calibration. (See the *IRIS Utilities Manual*.)

Figure 5–3: Model Intensity Curve



Plot Method for Calibration of I_o

This approach generates the curve shown above to determine the value of I_o . The general procedure is to connect a calibrated signal generator to the radar receiver and inject known dBm power levels to generate a calibration plot of measured power vs the inserted power at the antenna feed, similar to that in Figure 5–3. The calibration reflectivity dBZ_o is computed from the radar constant and the value of I_o which is the intercept of the straight line fit with the Noise level. I_o is the signal level for 0 dB SNR, i.e., signal power equals noise power.

Typically a CW test signal is used for this. Follow the instructions provided by the radar manufacturer for injecting a test signal. During calibration, the radar should be fully operational, so that all sources of noise are present. Ideally the transmitter should be turned on during calibration.

⚠ **Important: Verify with the radar manufacturer that no damage will occur to the signal generator if the transmitter is running during the calibration.**

To perform the calibration, insert signals at steps of 5 or 10 dB over the entire range of the system. Draw the plot shown in figure 5–3. You can utilize fine resolution steps at the ends of the scale to observe the details of the roll off. Be sure to raise the antenna up a few degrees to

avoid ground thermal noise. Also tune the frequency of the signal generator using the setup command "pr", and displaying the received signal spectrum. Be sure to check the tuning at the end of the calibration to make sure the signal generator and IFD have not drifted apart.

Each time that a new signal level is injected, the measured power values are obtained by first invoking the SNOISE command and then reading-back the results using the GPARM command. The Log of Measured Noise Level (Word 6) from GPARM should be used. This procedure averages many samples together. For IRIS users, this is all handled by the **zauto** utility.

Finally turn it all the way down and make one more sample to measure the noise level N . I_o is obtained from the intercept of the horizontal line at N and the straight line fit to the linear portion of the curve. This value must be corrected for losses as discussed in the section below.

Single-Point Direct Method for Calibration of I_o

This calibration method requires no support software. The approach uses the TTY setups commands. Again the signal generator output must be calibrated in absolute dBm. Use a power meter to check the calibration.

- Turn the radiate off and connect the signal generator to the test signal injection point.
- Raise the antenna to at least 20 degrees, and set the azimuth to point away from any known RF sources including the sun.
- Select the pulse width using the mt command.
- Select the pr command and use the commands to set the following:

```
Plotting Received Power Spectrum...  
Rx: Pri, Zoom: x1-x8, Navg: 25, Start: 100.01 usec (14.99 km), Span: 50 usec
```

- Set the signal generator to the approximate radar RF frequency with a power level corresponding to a strong signal (30 dB above the noise). Use a DC test signal (not pulsed). This signal should be visible as a peak in the spectrum display. Adjust the signal RF frequency so that produces the precise IF frequency (e.g., IF frequency of 30 MHz).
- Turn the signal generator off and record the "Filtered" power level. Note that because of the large averaging it will require several seconds for the average to stabilize.
- Turn the signal generator on, verify that the peak is still at the IF frequency and adjust the power level to obtain precisely 3 dB more "Filtered" power than was observed with the noise only. Again, allow several seconds for the averaging to stabilize after you make each amplitude adjustment.

This is the value of I_o , i.e., the test signal signal power equals the noise power. The next step is to correct the value of I_o for losses as discussed in the section below.

Treatment of Losses in the Calibration

In the calibration of the dBm level of the test signal, be sure to account for any losses that may occur between the antenna feed and the injection point, and in the cable and coupler that is used to connect the signal generator to the injection point. Figure 5-4 illustrates the nomenclature of the various losses that are involved in the calibration. The relationship between the injected test signal and the value of the received power relative to the feed is:

$$dBm_{Feed} = dBm_{Injected} + dBL_{Feed:Coupler}$$

$$dBm_{Feed} = dBm_{Siggen} - dBL_{Coupler} - dBL_{Cable} + dBL_{Feed:Coupler}$$

For example, assume the following:

Loss between the feed and the coupler	$dBL_{Feed:Coupler}$	3 dB
Loss caused by the coupler	$dBL_{Coupler}$	30 dB
Loss in the cable from siggen to coupler	dBL_{Cable}	2 dB

Then if the test signal generator output is -50 dBm, the injected power is

$$dBm_{Injected} = -50 - [30 + 2] = -82 \text{ dBm.}$$

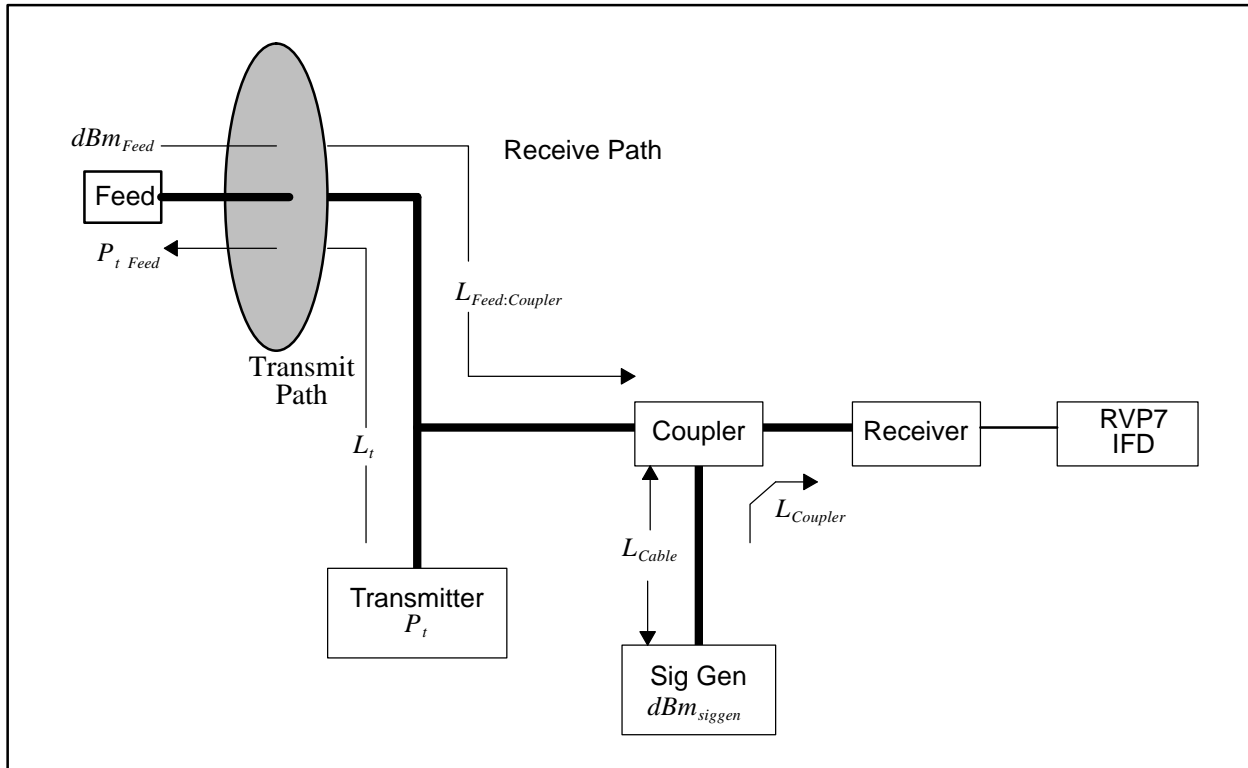
The equivalent power at the feed is then 3 dB more than this

$$dBm_{Feed} = -82 + 3 = -79 \text{ dBm.}$$

During the calibration, there are several ways to handle the losses using these equations. Two examples are:

- Each signal generator value can be corrected for losses so that the calibration plot shows IFD measured power vs received power at the feed. This is recommended for manual calibration.
- The signal generator values can be plotted directly and the intercept power I_o can be corrected for losses so that it is properly referenced to power at the feed. This is the approach used by the IRIS **zauto** utility.

Figure 5–4: Illustration of Losses that Affect LOG Calibration



Determination of dBZ_o

The calibration reflectivity is determined from the radar equation as follows:

$$dBZ_o = 10 \log [Cr_o^2 I_o]$$

where I_o is in mW (corrected for receive losses), the reference range r_o is 1 km, and the radar constant C is:

$$C = \frac{2.69 \times 10^{16} \lambda^2}{P_t \tau \theta \phi G^2} L_t$$

where,

- λ Radar wavelength in cm.
- P_t Transmitted peak power in kW.
- L_t Transmit loss (e.g., 3 dB corresponds to $L_t = 2$)
- τ Pulse width in microseconds.
- θ Horizontal half-power full beamwidth.
- ϕ Vertical half-power full beamwidth.
- G Antenna gain (dimensionless) on beam axis.

The radar constant is determined from the characteristics of your radar (check with the manufacturer if you are unsure of the values). Note that transmit losses are accounted for in the radar constant, while receiver loss is usually included in the calculation of I_o .

Finally, if the value of I_o calculated above was not based on loss-corrected dBm values, correct I_o as follows:

$$dBI_{o \text{ corrected}} = dBI_o - dBL_{\text{Coupler}} - dBL_{\text{Cable}} + dBL_{\text{Feed: Coupler}}$$

Example Calculation of dBZ_o :

This sample calculation is provided so that programmers can check their arithmetic. The radar parameters:

λ	Radar wavelength in cm.	5 cm.
P_t	Transmitted power in kW.	500 kW
L_t	Transmit Loss	2 (3 dB)
τ	Pulse width in microseconds	1 microsecond
θ	Horizontal half-power beamwidth in degrees	1 degree
ϕ	Vertical half-power beamwidth in degrees	1 degree
G	Antenna gain (dimensionless) on beam axis	19,953 (43.0 dB)

The radar constant for this example is,

$$C = \frac{2.69 \times 10^{16} \lambda^2}{P_t \tau \theta \phi G^2} L_t = \frac{(2.69 \times 10^{16})(5)^2}{(500)(1)(1)(19,953)^2} \quad (2.0)$$

$$= 6.76 \times 10^6 \left[mm^6 m^{-3} km^{-2} mW^{-1} \right]$$

Assume that I_o with loss correction is calculated to be -105 dBm (3.16×10^{-11} mW), then dBZ_o is,

$$dBZ_o = 10 \log [Cr_o^2 I_o] = 10 \log [(6.76 \times 10^6) (1)^2 (3.16 \times 10^{-11})]$$

$$= -36.7 dB (mm^6 m^{-3})$$

This value would be down-loaded to the signal processor using the SOPRM command.

5.5 Dual PRT Processing Mode

The RVP8 supports two major modes for Dual PRT processing, i.e., algorithms using triggers that consist of alternate short and long periods. Most of the Doppler parameters are available in each of these modes. You may also request time series data in both cases; the samples will be organized so that the first pulse of a short PRT pair always comes first.

5.5.1 DPRT-1 Mode

The DPRT-1 trigger consists of a very short PRT from which Doppler data are obtained, followed by a much longer PRT whose purpose is to limit the average duty cycle of the transmitter. No information is extracted from the long PRT pair, but Dual-PRF techniques can still be used by varying the short period from ray to ray. The “-1” suffix in the name for this mode is a reminder that Doppler parameters are computed from the short PRT only. The DPRT-1 mode is intended for millimeter wavelength radars that must run at a very high effective PRF (up to 20KHz) to get an acceptable unambiguous velocity, but which also have a much lower duty cycle constraint on the average number of pulses transmitted each second.

In DPRT-1 mode the requested PRF from the host computer will generally be quite large (up to 20KHz); and the reciprocal of this “effective instantaneous PRF” will determine the trigger’s short PRT interval. In this way, all subsequent physical calculations will be scaled correctly, e.g., unambiguous velocity, maximum first trip range, etc., are all supposed to be based on the short PRT interval. The host computer must therefore be configured so that it can ask for these very high trigger rates.

The duration of the long PRT interval is not specified directly by the host computer. Rather, the RVP8’s “Maximum number of Pulses/Second” setup parameter is used to compute how much delay to insert in order to insure that the transmitter’s duty cycle is not exceeded. This special treatment applies only in DPRT mode; all other modes that have uniform triggers continue to interpret the RVP8’s trigger bound as a simple “Maximum PRF”.

Since DPRT-1 mode uses only the short pairs of pulses, it is not possible to run the “R2” moment estimation algorithms. The RVP8 will return the GPARM “Invalid Processor Configuration” bit if “R2” is requested in DPRT mode. The error bit will also be returned if the number of pulses requested (sample size) is not even. All other error conditions are the same as FFT mode.



Warning: Since the RVP8’s “Maximum number of Pulses/Sec” is used to enforce the duty cycle limit, it is essential that it not be overwritten by the host computer’s upper PRF limit, which typically will be much higher. To insure this, you must make sure that the PWINFO command is disabled in the RVP8 “Mc” setup menu. You will have no duty cycle protection if you do not do this.



Note: You may still choose to run Dual-PRF velocity unfolding within the DPRT-1 mode. What will happen is that the short PRT will vary in the selected 3:2, 4:3, or 5:4 ratio, but the overall duty cycle will remain constant. The combination of Dual-PRF and DPRT-1 is tremendously effective in extending the radar’s unambiguous velocity interval.

5.5.2 DPRT-2 Mode

The trigger consists of alternating short and long period pulses, where the ratio of the periods is determined by the velocity unfolding ratio that has been selected. Doppler data are extracted from both the short and long pulse pairs (hence the “-2” suffix), and unfolded velocities are made available on each ray based on the combined PRT data from that ray alone. DPRT-2 mode is intended for rapidly scanning radars where the ray-to-ray spatial continuity assumptions of the traditional Dual-PRF algorithms do not apply.

The DPRT-2 velocity unfolding algorithm uses a modified version of the standard Dual-PRF algorithm. Both start by computing a simple velocity difference as a first approximation of the unfolded result. The standard algorithm uses that difference to unfold the velocity from the most recent ray, which yields a lower variance estimate than the difference itself. The DPRT-2 algorithm is similar, except that the folded velocity from both PRTs are unfolded independently and then averaged together.

In addition to the above, the RVP8 also computes the DC average of the (I,Q) data within each bin. This is used as a simple estimate of clutter power, so that corrected reflectivities are available in DPRT-2 mode whenever a non-zero clutter filter is selected. DPRT-1 mode is the same in this respect. However, the DPRT-2 widths use an improved algorithm based on the two different PRTs, and which avoids the SNR sensitivity of the DPRT-1 width estimator.

5.6 Dual PRF Velocity Unfolding

For a radar of wavelength λ operating at a fixed sampling period $\tau_s = 1/PRF$, the unambiguous velocity and range intervals are given by:

$$V_u = \frac{\lambda}{4\tau_s} \quad \text{and} \quad R_u = c \frac{\tau_s}{2}$$

where “c” is the speed of light. Often these intervals do not fully cover the span of velocity and range that one would like to measure. The problem is generally worse for short wavelength radars, since that unambiguous velocity span is directly proportional to λ for a given τ_s . If the unambiguous range interval is made sufficiently large by increasing τ_s , then the resulting velocity span may be unacceptably small.

The RVP8 provides a built-in mechanism for extending the unambiguous velocity span by a factor of two, three, or four beyond that given above. The technique, called Dual PRF velocity unfolding, uses two pulse periods rather than one, and relies on the extra information thus obtained to correct (i.e. unfold) the mean velocity measurement from each individual period. The Dual PRF trigger pattern consists of alternating (N+k)-pulse intervals where the period in each interval is either τ_l (for the low-PRF) or τ_h (for the high-PRF). Here “N” is the sample size, and “k” represents a delay that permits the clutter filter to equilibrate to the new PRF after each change. The clutter filter impulse response lengths vary according to which filter is selected.

The two trigger periods τ_l and τ_h must be chosen in either a 3:2, 4:3, or 5:4 ratio. These ratios give factors of two, three, and four times velocity expansion over the τ_h period alone. The unfolding algorithm makes use of the following results. Suppose that the radar observes a target with mean velocity V at each of the two trigger periods. The measured phase angles for the R_1 autocorrelations at the two PRFs are:

$$\theta_l = \frac{4\pi V \tau_l}{\lambda} \quad \text{and} \quad \theta_h = \frac{4\pi V \tau_h}{\lambda}$$

where angles outside the basic $[-\pi, \pi]$ interval are returned to that interval by appropriate additions of $\pm 2\pi$. These angles correspond to the ordinary single-PRF Doppler velocity measurements, and the $\pm 2\pi$ uncertainties reflects the fact that each measurement is folded into its own unambiguous interval:

$$V_{ul} = \frac{\lambda}{4\tau_l} \quad \text{and} \quad V_{uh} = \frac{\lambda}{4\tau_h}$$

If we define ϕ to be the difference between the two measured phases then:

$$\phi = \theta_l - \theta_h = \frac{4\pi}{\lambda} [\tau_l - \tau_h]$$

which can be interpreted as a phase angle within the unfolded interval:

$$V_{u \text{ unfold}} = \frac{\lambda}{4(\tau_l - \tau_h)}$$

Now if τ_l and τ_h are in a 3:2 ratio, then:

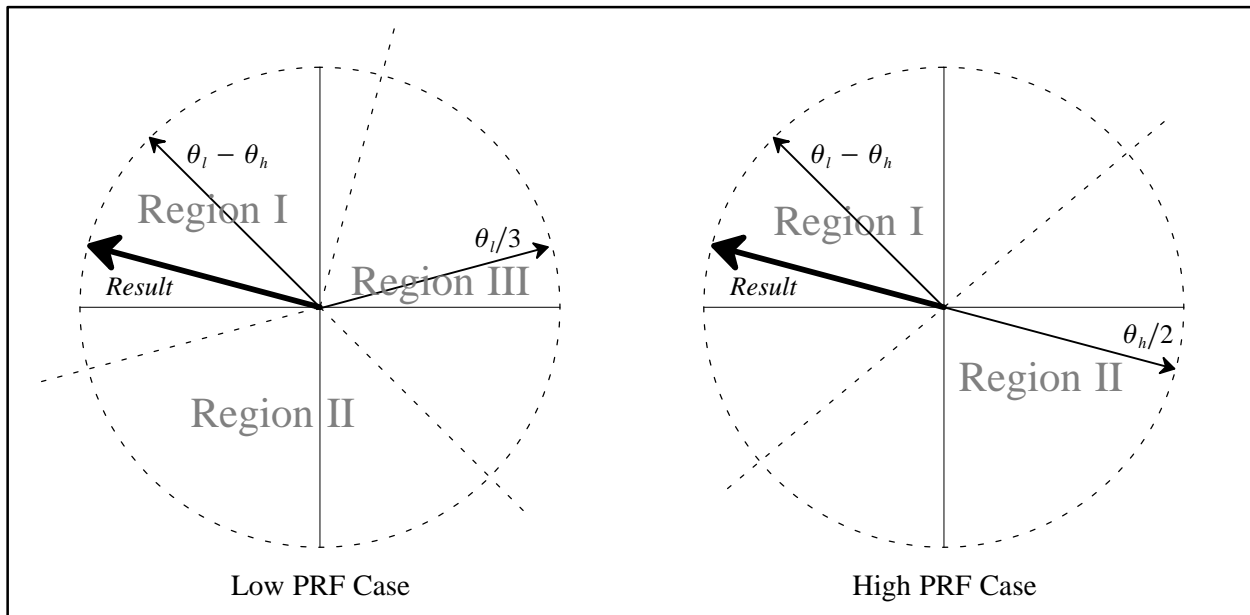
$$\tau_l - \tau_h = \frac{\tau_l}{3} = \frac{\tau_h}{2}$$

and thus $V_{u \text{ unfold}} = 3V_{ul} = 2V_{uh}$

The angle ϕ represents a velocity phase angle in $[-\pi, \pi]$, but with respect to an enlarged unambiguous interval. Thus, by simply differencing the folded angles from the high and low PRFs, we obtain an angle that is unfolded to a larger velocity span. Similar reasoning shows that the 4:3 ratio gives a factor of three improvement over V_{uh} .

In practice, the unfolded angle ϕ is not in itself a suitable velocity estimator. The reason is that the variance of ϕ is equal to the sum of the variances of each of its components, i.e., twice that of the individual measurements alone. If the target is at all noisy, then this increase in variance can be severe. Rather than use ϕ directly, the RVP8 uses it only as a rough estimate in determining how to unfold the individual velocity measured from each PRF.

Figure 5-5: Dual PRF Concepts



This technique is illustrated in Figure 5-5. The figure shows how the low-PRF and high-PRF angles are unfolded based on the difference angle. The diagrams show phase planes representing the large unfolded velocity interval, and the locations of various vectors on those planes. Referring first to the right figure, the difference angle is plotted, and the plane is divided into two equal size regions, one of which is centered on the difference vector. The high-PRF angle is then divided by two and plotted. The resultant unfolded velocity angle must either be this vector, or this vector plus π . Since adding π places the vector into acceptance Region 1 where it is

nearest the difference angle, we conclude that this is the correct unfolding. Likewise, on the left diagram we unfold the low-PRF angle by dividing the plane into thirds centered on the difference angle. The result angle is either

$$\frac{\theta_l}{3}, \quad \frac{\theta_l}{3} + \frac{2\pi}{3} \quad \text{or} \quad \frac{\theta_l}{3} + \frac{4\pi}{3}$$

depending on which one falls into the acceptance Region 1. Note that the resultant angle is the same in each case.

The RVP8 makes efficient use of the incoming data by unfolding velocities from both the low and the high-PRF data, making use each time of information in the previous ray. When low-PRF data are taken the derived velocities are unfolded by combining information from the previous high-PRF interval. Likewise, when high-PRF data are acquired the velocities are unfolded based on the previous low-PRF interval. Thus, when operating in the Dual PRF mode, the RVP8 outputs one data ray for each (N+k)-pulse interval. However, the velocity data in the Dual PRF rays are unfolded, so that the [-1,+1] interval now represents either two or three times the prior velocity range. Put another way, the data are still interpreted as described in the section on mean velocity estimation, except that V_u is now larger.

The width data are also modified somewhat during Dual PRF unfolding. Although valid widths are obtained independently on all rays, those measured at low-PRF are larger than those at high-PRF. This is simply because the dimensionless width units are with respect to a larger velocity interval in the latter case. To compensate for this, low-PRF widths are multiplied by either 2/3 or 3/4 before being output. This puts them in the same scale as the high-PRF values, and thus, the widths do not vary on alternate pulses. A useful consequence of this is that width data can be sent directly to a color display generator without having to plot every other ray in a different scale.

There are a few words of caution that should be kept in mind when using the RVP8 in the Dual PRF processing modes. The unfolding algorithms make the assumption that targets are more-or-less continuous from ray to ray. Otherwise, it would not make sense to use data from a previous ray to unfold velocities in the current ray. Users must therefore assure that their antenna scan rate and beamwidth are such that each target is illuminated, at least partially, over each full 2(N+k)-pulse interval. In practice, a certain amount of decorrelation from ray to ray is acceptable, since the previous rays are used only to decide into which unfolded interval the current ray should be placed. Small errors in the previous ray data, therefore, cause no error in the output. However, large previous-ray errors would lead to incorrect unfolding.

A more subtle side effect of Dual PRF processing arises from clutter filtering because clutter notches now appear at several locations in the unfolded velocity span, rather than just at zero velocity. These additional rejection points come about because the original velocity intervals are mapped some integer number of times to create the unfolded interval. Since each original interval has a clutter notch at DC, it follows that the final expanded velocity interval will have several such notches. For example, in the 3:2 case, in addition to removing DC the clutter filter removes velocities at $-2V_u/3$, $+2V_u/3$, and V_u .

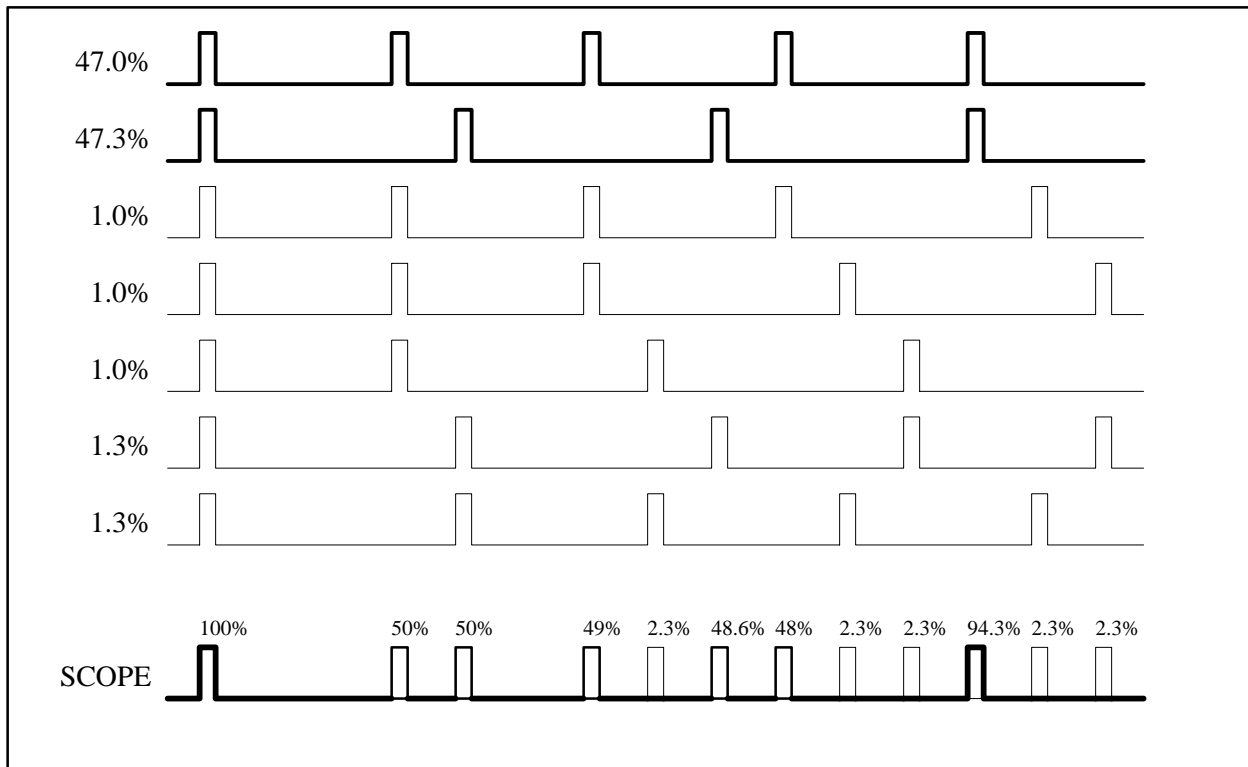
Unfortunately, these clutter filter "images" are a fundamental consequence of the Dual PRF processing technique and are not easily removed. They can cause trouble not only for the velocity unfolding itself, but because the computed clutter corrections to be wrong at the image

points. However, there is a useful work-around in the RVP8 to minimize their impact — turning the clutter filter off at far ranges where little clutter is expected and using a narrow clutter filter minimizes the effects of the clutter filter on weather targets.

The 4:3 PRF unfolding ratio is more susceptible to unfolding errors in cases where the spectrum width is large and/or the SNR is low. The user should experiment with the two ratios to determine which provides the best results for their particular application. Although the RVP8 trigger generator can produce any trigger frequency, only the 3:2 and 4:3 ratios can be used with the built-in unfolding algorithms. The RVP8 still permits other PRT ratios to be explored, but the unfolding technique must then be manually programmed on the user's host computer.

Oscilloscope observations of Dual PRF triggers can sometimes be confusing. Figure 5–6 shows seven possible scope traces (and their associated probabilities) for the RVP8 trigger during Dual PRF operation. The PRF ratio is 4:3, and the sample size is 50 pulses at the high PRF, and 37 pulses at the low PRF. The signal labelled “SCOPE” is the composite of these traces, and is what would actually be seen on an oscilloscope. Notice that there are a number of low probability pulses. The exact details of the sample sizes and the trigger hold off time can make the low probability pulses appear to come and go randomly. This is normal, and is no cause for alarm.

Figure 5–6: Example of Dual PRF Trigger Waveforms



5.7 Optional Dual Polarization- ZDR, PHIDP, KDP, LDR, ...

5.7.1 Overview of Dual Polarization

Polarization measurements can provide additional information that can be used to determine more accurate measurements of rainfall or, in some cases, infer particle type such as hail or graupel. The fundamental basis for polarization is that raindrops, particularly larger ones, are not spherical — they are oblate (flattened) such that the horizontal axis is longer than the vertical axis. This means that raindrops will respond differently, for example, to vertical and horizontal polarization of the electric field vector. Because of this, and for technical reasons, most polarization radars use horizontal and vertical polarization. For a review of polarization techniques and variables, please refer to Doviak and Zrníc (1993) section 8.5.

Fundamentally a polarization radar measures amplitude and phase in the same manner as a conventional radar. The new information is that the amplitude and phase can be measured at more than one polarization. The differences in amplitudes and phases measured at different polarizations contain information on the presence or absence of non-spherical scatterers such as large flattened drops. For convenience, some of the basic polarization variables are described below:

ZDR: Differential Reflectivity

In the case of amplitude (power) measurements, the larger horizontal axis of drops causes the power measured at horizontal polarization (of the electric field) to be larger than the power measured at vertical polarization. The ratio of the reflectivity factors Z_H/Z_V expressed in dB is given the name ZDR or differential reflectivity. It is generally positive in rain (i.e., >1) and is usually less than about 5 dB. When the rainfall rate is large, there are typically more large drops so that ZDR is larger. Low ZDR and high dBZ indicates the presence of hail which is perhaps tumbling with no preferred orientation. ZDR, because it is a ratio of powers, is not sensitive to the radar calibration as long as the overall gain of the H and V channels is the same (or calibrated).

PhiDP and KDP: Differential Phase and Specific Differential Phase

In the case of phase measurement, the speed of propagation is also affected by the asymmetry of the larger drops. Because of the longer dimension of the horizontal axis of drops, the medium is effectively more dense for horizontal than for vertical polarization so that the speed of light is reduced for horizontal polarization. This causes the horizontal wavelength to be slightly compressed (more phase cycles per unit distance) in comparison with the vertical wavelength which leads to a phase difference between horizontal and vertical. The phase difference $\Phi_H - \Phi_V$ is called Φ_{DP} differential phase shift. Φ_{DP} increases with range since the phase shifts faster (more frequency cycles per unit distance) for the compressed horizontal microwaves as compared to the faster vertical microwaves. The range derivative of the differential phase, i.e., the change of phase per unit distance, is called K_{DP} or the specific differential phase. K_{DP} is almost directly proportional to the rainfall rate so that it has the potential for improving precipitation rate measurements as compared to traditional Z-R relationship measurements which can be highly inaccurate.

LDR: Linear Depolarization Ratio

Some advanced polarization radars can transmit at one polarization and receive simultaneously in two channels, usually the co-polarized and cross-polarized components. For example, when transmitting horizontal, both horizontal (co-polarized) and vertical (cross-polarized) are received by two separate channels. In the case of vertical or horizontal, the ratio of the power $Z_{\text{cross}} / Z_{\text{co}}$ is called the linear depolarization ratio or LDR. The amount of incident radiation that is depolarized by a particle depends on the particle shape and orientation (e.g., canting angle with respect to horizontal). Perfectly spherical particles do not depolarize either horizontal or vertical polarization so that LDR is zero. Particles that are wet, tumbling and irregularly shaped will give larger LDR values. Therefore, LDR values in rain tend to be small, e.g., less than -25dB . Larger values of LDR can occur in the bright band or in the presence of hail.

A radar and antenna system must be optimized to measure LDR by assuring that the antenna, feed and supporting struts and radome are not themselves depolarizing the transmitted and received radiation. This is called "cross-pol isolation". The integrated cross-pol isolation of the antenna pattern must be better than about 30 dB for LDR measurement since -20 dB is a large LDR.

[RHOHV, PHIDP] [RHOH, PHIH] [RHOV, PHIV]: Correlation Variables

There are several correlation functions that can be calculated depending on the capabilities of the radar. These are generally complex having both an amplitude and phase. These are all normalized so that a perfect correlation magnitude is 1 and perfectly decorrelated is 0.

RHOHV and PHIDP are the magnitude and phase of the correlation between the horizontal and vertical co-polarized channels. These are available on H/V switching systems or on systems that transmit simultaneous H and V. As discussed in a preceding paragraph, PHIDP can be used to infer precipitation rate. RHOHV in rain is typically very close to 1 (0.98). RHOHV values can be reduced in the case of irregularly shaped, randomly oriented, wet tumbling particles. Thus RHOHV has information on the particle type.

RHOH and PHIH are the magnitude and phase of the correlation between the co-polarized and cross-polar channels for H transmission and simultaneous H and V reception. RHOV and PHIV denote the cross-channel correlation magnitude and phase for vertical transmission. These are available on dual-channel receiver with transmit either fixed or alternating. The information content of the cross-pol correlations is the topic of current research.

5.7.2 Radar System Considerations

A polarization radar is characterized by how it transmits and how it receives. For simplicity we will assume that the radar uses horizontal and/or vertical polarization. However, other polarization pairs could be used (e.g., right and left circular polarization).

Transmit Modes

- **Fixed** (horizontal or vertical)- this can be controlled by a switch or the radar can be simply fixed to transmit a single polarization. If a switch is used, it can be a simple slow waveguide switch rather than a fast switch (pulse-to-pulse).
- **Alternating** (horizontal and vertical)- in this case the radar alternates pulse-to-pulse between horizontal and vertical. A high-power fast switch is used to switch the polarization between the two channels.
- **Simultaneous** (horizontal and vertical)- horizontal and vertical are transmitted simultaneously.

Receive Modes

- **Single-channel receiver**- used only for alternating transmission. The receiver typically receives the co-polarized radiation (transmit H and receive H then transmit V and receive V).
- **Dual-channel receiver**- receives two channels (H and V) simultaneously.

The table below summarizes the various transmit and receive cases and the polarization variables that are available for each. Note that standard parameters are available for all cases (dBZ, V and W). The RVP8 supports all of these cases.

Receiver Type	Transmitter Type			
	Fixed H	Fixed V	Alternating H&V	Simultaneous H+V
Single-Channel	Conventional Radar	Conventional Radar	ZDR RHOHV PHIDP and KDP	Not applicable
Dual-Channel	LDRH RHOH PHIH	LDRV RHOV PHIV	LDRH LDRV RHOH RHOV PHIH PHIV ZDR RHOHV PHIDP and KDP	ZDR RHOHV PHIDP and KDP (STAR mode)

The fixed single -channel cases are conventional radars rather than polarization radars. The case of simultaneous H+V transmission and a single radar does not make physical sense. The other cases provide various polarization measurements. The fixed dual-channel cases allow the cross-polarization LDR and the co-pol/cross-pol correlation amplitude and phase to be measured

(e.g., RHOH and PHIH). The simultaneous H+V transmission and dual-channel reception is sometimes called the STAR mode (simultaneous transmit and receive). This allows the co-pol measurements to be made (ZDR, RHOHV, PHIDP and KDP). The alternating transmission dual-channel receiver allows both the co-pol and the cross-pol measurements to be made, i.e., it is the most complete.

Summary of Radar System Characteristics

The RVP8 supports all of these modes, but most polarization radar systems do not. As mentioned before, the measurement of cross-pol parameters such as LDR (fixed or alternating transmission and dual-channel reception) requires a radar system that has been optimized for cross-pol isolation, e.g., an offset feed antenna and no radome. By removing the feed, support struts and radome from the path of the radiation, the cross-pol isolation can be improved.

The single-channel alternating method has been used in several polarization radars for ZDR measurement. The advantage of this approach is that it is relatively easy to modify a conventional radar by simply adding a dual port feed and a high-power fast switch above the antenna rotary joints. The disadvantage is that the switch is costly and will eventually fail.

For these reasons, the STAR mode has come into recent use. No switch is required and the components are fairly reliable. The disadvantage of the approach (as it is usually implemented) is that a dual rotary joint and dual waveguides are required to duct both the H and the V through the antenna pedestal up to the antenna feed. In spite of this, the STAR mode offers perhaps the best approach for upgrading an existing radar or for factory installation on a new radar of conventional design.

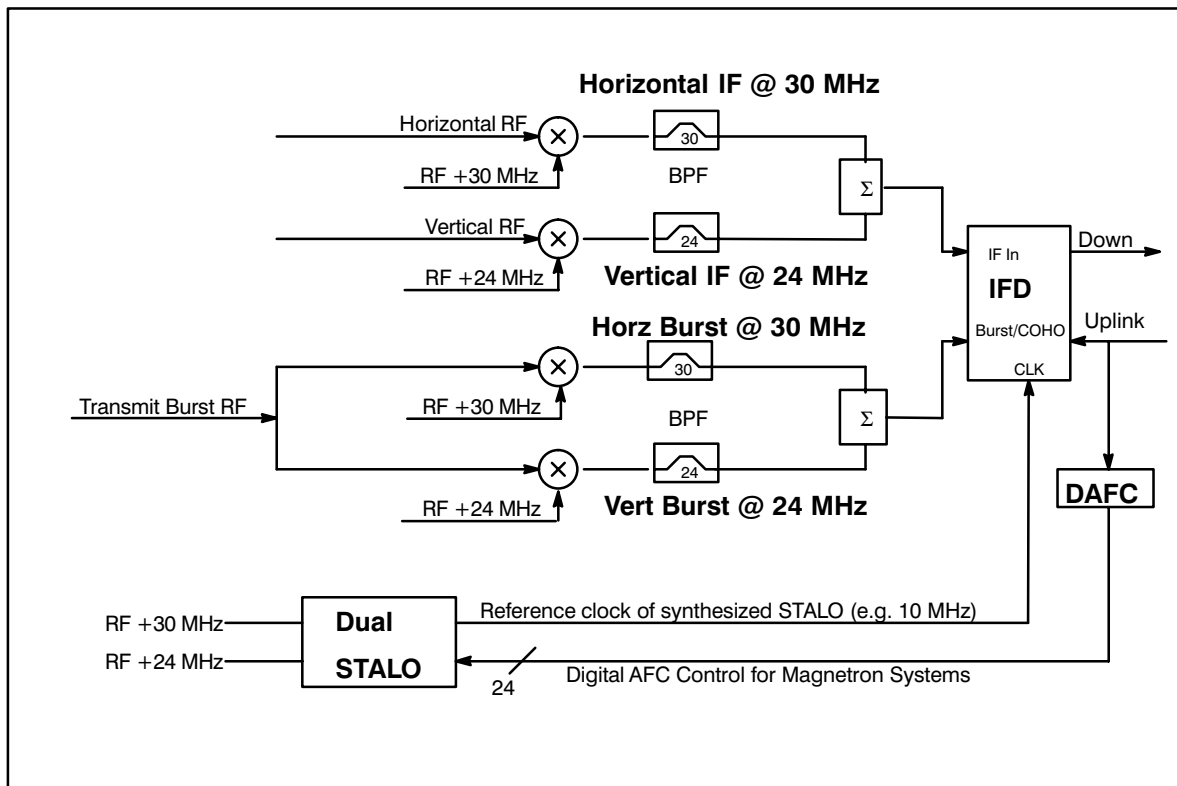
5.7.3 RVP8 Dual-Channel Receiver Approach

Dual-Channel Multiplexing for the IFD

The RVP8 uses an innovative technique for implementing the dual-channel receiver approach, i.e., dual-channel multiplexing. Just as a single wire can carry multiple telephone conversations, two polarization channels can be put on the same wire at different IF frequencies, digitized by the IF Digitizer and then separated by digital filtering. This means that the exact same hardware that is used for a single-channel digital receiver is used for the dual-channel application. The typical IF separation is 6 MHz and the channel isolation is about 50 dB which is more than adequate for even sensitive LDR measurements.

The figure below shows a block diagram of the approach for the magnetron case.

Figure 5–7: Dual Receiver Magnetron Case



Magnetron Approach

The approach for the magnetron case requires a dual output STALO to obtain the two IF's for horizontal and vertical (H and V). The H and V RF channels are mixed to obtain the H and V IF signals at 30 and 24 MHz in the example. In addition, the transmit sample (also know as the "burst pulse" is split and mixed with the two IF's so that there is a version at 30 and 24 MHz. This is important for the determination of the transmit phase corresponding to each IF. Both the H and V IF signals are combined and then digitized in the usual way by a standard RVP8 IFD. The same is done with the 24 and 30 MHz versions of the transmit burst sample. The

anti-aliasing filters usually installed on the IFD are removed and replaced by separate filters that are placed ahead of the point where the signals are combined. Note that these filters are centered at the appropriate IF frequency and are typically 6 MHz wide for a 6 MHz IF separation. The two composite signals are then digitized by the IFD identically to the case of a single-channel receiver and later separated during the digital band pass filtering/mixing step to obtain the I and Q of the burst sample and range bin values of I and Q values.

Klystron Approach

In the case of a Klystron system, the approach is the same, except that the COHO must generate two frequencies which are mixed with the STALO to provide the two reference frequencies. These are used in place of the STALO1 and STALO2 frequencies in the diagram. The same two COHO signals (e.g., at 24 and 30 MHz) are then treated identically to the transmit burst pulse in the magnetron case. In this case there is no burst pulse so the two burst pulse mixers are not required.

Reference Clock to IFD

In either case, it is critical that the oscillators (both STALO's in the case of a magnetron and the STALO and both COHO's in the case of a Klystron) be phase locked to a common reference clock. This clock, or a derivative frequency of the clock such as a COHO frequency, is input into the IFD to provide an absolute phase reference. Another alternative is to supply the difference frequency between the two IF's as the reference clock. To do this, the outputs of the two STALO's can be mixed by an additional mixer and filtered to obtain (for example) a 6 MHz reference frequency.

The RVP8 IFD phase locks its sampling crystal to the reference clock input. Trigger generation by the RVP8 will also be phase locked to the reference clock. The reference clock must be in the range 2 to 60 MHz at 0 to -10 dBm and stable in phase to 10^{-7} . The RVP8 IFD must be specially configured with a locking crystal to enable this feature. SIGMET will either factory install the modification or assist the customer in performing the modification and supply the necessary components.

5.7.4 Overview of Processing Algorithms

Polarization Modes and Outputs Supported by RVP8

The RVP8 supports four polarization modes summarized in the table below. For each case, the standard moments (T, Z, V and W) are calculated as well. The notation for the outputs used here is similar to that in standard usage (e.g., Doviak and Zrnice). However, for LDR we use the notation LDRH to indicate that this is the LDR for horizontal transmission. The notation RHOH and PHIH is used to indicate the magnitude and phase of the covariance between the co- and cross-polarized channels for H transmit.

Case	Transmit	Receive	Processing Mode	Polarization Outputs
1	Fixed Horizontal or Fixed Vertical	Dual-Channel	PPP only	LDRH RHOH PHIH or LDRV RHOV PHIV
2	Simultaneous H+V (STAR Mode)	Dual-Channel	PPP or ZDR for FFT, Random Phase and DPRT1&2	ZDR PHIDP KDP RHOHV
3	Alternating H/V	Single-Channel	PPP only	ZDR PHIDP KDP RHOHV
4	Alternating H/V	Dual-Channel	PPP only	LDRH RHOH PHIH LDRV RHOV PHIV ZDR PHIDP KDP RHOHV

Input Receiver Sample Notation

For the discussion of polarization, we will adopt the notation used by Doviak and Zrnice. The received signal for pulse n from a single range bin shall be denoted as:

s_{hh}^n	Receive h: Transmit h	Horizontal co-polar signal
s_{vh}^n	Receive v: Transmit h	Horizontal cross-polar signal
s_{vv}^n	Receive v: Transmit v	Vertical co-polar signal
s_{hv}^n	Receive h: Transmit v	Vertical cross-polar signal

The pulse index is now indicated by the superscript as opposed to the subscript. The first subscript indicates the received polarization while the second subscript indicates the transmit polarization. If the transmit is the same as the received polarization, then this is called the co-polarized signal. If the transmit and receive are different then this is called the cross-polarized signal.

These variables are complex and are the same as the “ s_n ” notation used earlier, for example we can write:

$$s_{hh}^n = I_{hh}^n + j Q_{hh}^n$$

to show the relationship to the received I and Q values. Either filtered and unfiltered versions of the samples can be selected for processing. However, for convenience we will drop the s' notation for filtered samples.

Notation and Model for Correlations

The pulse pair processing mode is used for all of the polarization calculations, except that ZDR-only processing for the STAR case can be done in either FFT or random phase as well as pulse pair. As with the standard moments, the autocorrelations form the basis for the processing of the polarization variables.

The autocorrelations are computed in a manner identical to the standard moments, e.g., in pulse pair mode, the autocorrelations for the horizontal transmit co-polar channel are:

$$\begin{aligned}
 T_{ohh} &= \frac{1}{M} \sum_{n=1}^M s_{hh}^n * s_{hh}^n \\
 R_{ohh} &= \frac{1}{M} \sum_{n=1}^M s_{hh}'^n * s_{hh}'^n \\
 R_{1hh} &= \frac{1}{M-1} \sum_{n=1}^{M-1} s_{hh}'^n * s_{hh}'^{n+1} \\
 R_{2hh} &= \frac{1}{M-2} \sum_{n=1}^{M-2} s_{hh}'^n * s_{hh}'^{n+2}
 \end{aligned}$$

What is different is that for polarization systems, this processing can be applied in up to four separate channels (s_{hh} , s_{vh} , s_{vv} and s_{hv}). The physical model for the channel powers is identical to the model used for the standard moment cases, i.e.,

Co-Channel Power

Cross-Channel Power

$$R_o^{hh} = g_h^r g_h^t S_{hh} + N_h$$

$$R_o^{vh} = g_v^r g_h^t S_{vh} + N_v$$

$$R_o^{vv} = g_v^r g_v^t S_{vv} + N_v$$

$$R_o^{hv} = g_h^r g_v^t S_{hv} + N_h$$

Here S is used to denote the actual backscatter average power to the radar which, when multiplied by the appropriate transmitter and receiver gains, yields the actual measured power. Sometimes in comparing powers in two channels (e.g., ZDR and LDR) we will need to know the relative gains of the two channels. However, in many calculations, the relative gains cancel-out and in these cases the algorithms are implemented assuming all the gains are equal to 1.

In the algorithm descriptions, we will often use the notation common in the literature that (for example):

$$R_{ohh} = \frac{1}{M} \sum_{n=1}^M s_{hh}'^n * s_{hh}'^n = \langle |s_{hh}'|^2 \rangle$$

Noise Bias of Channel Power and Optional Correction

The average noise powers N_v and N_h are assumed to be receiver noise only. These bias the autocorrelations at lag zero, i.e., the channel power measurements. Autocorrelations at lags 1 and 2 are not biased by noise. Cross channel correlations are also not biased by noise, assuming that the noise in the two channels is independent (a good assumption).

The channel noise values are measured directly by the RVP8 during noise sampling. Whether to use these measured values to correct for the noise power when computing a channel power is optionally configured in the TTY setups. The choice is made in the mp TTY setup question "Polarization Parameters NoiseCorrected:YES/NO". If enabled, every time that a channel power is calculated, the noise power is subtracted.

This has some interesting effects. With no noise correction, ZDR values in weak signal regions will be biased by noise toward 0 dB (equal power), while if noise correction is enabled the values will be unbiased but will show substantial deviation over the region. The choice is up to the user.

Clutter Filtering

Clutter filtering is available for all four cases. The use of clutter filters should be carefully considered since many polarization parameters such as ZDR and LDR require highly accurate bin-to-bin consistency. The clutter filters will attenuate some small amount of weather near zero velocity and the amount could be slightly different at the two polarizations. When using the clutter filters, users should verify that the functioning is acceptable for their application.

The standard moments (T, Z, V, W) are filtered in the usual manner by selecting a clutter filter (other than Filter #0 which is the All Pass filter). To enable clutter filtering of the polarization variables, the mp TTY setup question "Polarization Parameters Filtered" should be set to YES. The polarization algorithms will then be calculated with filtered time series. Note that filtering can be effectively disabled for the polarization variables by selecting Filter #0 even if the mp setup is set to enable filtering. This allows users to make easy comparison of the filtered vs unfiltered results using their application software without having to change the RVP8 setup.

The s' notation for filtered samples shall be dropped in the algorithm discussions. It is understood that the input samples in all cases may be either filtered or unfiltered according to the user's choice in the TTY setups.

5.7.5 Case 1: Fixed Transmit: Dual-Channel Receiver

Input Receiver Samples

In fixed mode the radar is configured (either permanently or by means of a switch) to transmit either vertical or horizontal polarization with dual-channel reception of both the co- and cross-channel polarizations, e.g., transmit horizontal and receive both horizontal (co) and vertical (cross) polarizations.

The received samples in the two transmit cases are:

Transmit Horizontal

or

Transmit Vertical

$$[s_{hh}^1 : s_{vh}^1] [s_{hh}^2 : s_{vh}^2] [s_{hh}^3 : s_{vh}^3] \dots [s_{hh}^M : s_{vh}^M]$$

$$[s_{vv}^1 : s_{hv}^1] [s_{vv}^2 : s_{hv}^2] [s_{vv}^3 : s_{hv}^3] \dots [s_{vv}^M : s_{hv}^M]$$

Calculation of the Polarization Measurands

The processing in this mode is done by pulse pair algorithm. The user may select a clutter filter, but in general this is not recommended for polarization studies since the clutter filter might interfere with the accuracy of sensitive parameters such as LDR.

The polarization measurands for the two transmit cases are as follows:

Transmit Horizontal

or

Transmit Vertical

$$\begin{aligned} LDRH &= 10 \text{ LOG} \left[\frac{S_{vh}}{S_{hh}} \right] & \text{or} & \quad LDRV = 10 \text{ LOG} \left[\frac{S_{hv}}{S_{vv}} \right] \\ &= 10 \text{ LOG} \left[\frac{\langle |s_{vh}|^2 \rangle - N_v}{\langle |s_{hh}|^2 \rangle - N_h} \right] - XDR & \text{or} & \quad = 10 \text{ LOG} \left[\frac{\langle |s_{hv}|^2 \rangle - N_h}{\langle |s_{vv}|^2 \rangle - N_v} \right] + XDR \\ RHOH &= |\rho_h| & \text{or} & \quad RHOV = |\rho_v| \\ PHIH &= \arg[\rho_h] & \text{or} & \quad PHIV = \arg[\rho_v] \end{aligned}$$

Here, the H and V average channel powers are computed as follows with optional noise correction, i.e.,

$$\begin{aligned} \text{Co-} \quad g_h^r g_h^t S_{hh} &= \langle |s_{hh}|^2 \rangle - N_h & \text{or} & \quad g_v^r g_v^t S_{vv} = \langle |s_{vv}|^2 \rangle - N_v \\ \text{Cross-} \quad g_h^r g_h^t S_{vh} &= \langle |s_{vh}|^2 \rangle - N_v & \text{or} & \quad g_h^r g_h^t S_{hv} = \langle |s_{hv}|^2 \rangle - N_h \end{aligned}$$

The complex covariance ρ (used above) is:

$$\text{for H transmit} \quad \rho_h = \frac{\langle s_{vh} s_{hh}^* \rangle}{\sqrt{S_{vh} S_{hh}}} \quad \text{or for V transmit} \quad \rho_v = \frac{\langle s_{hv} s_{vv}^* \rangle}{\sqrt{S_{hv} S_{vv}}}$$

Fortunately, the algorithms do not require us to know all of the individual gain terms. They cancel in the calculation of ρ so are taken as =1 in the implementation. However, the differential receiver gain XDR must be known from calibration to calculate LDR:

$$\text{dB Value is } XDR = 10 \text{ LOG } xdr \quad \text{where the linear value is } xdr = \frac{g_v^r}{g_h^r}$$

5.7.6 Case 2: Simultaneous Dual Transmit and Receive (STAR mode)

Input Receiver Samples

In this mode there is simultaneous transmit and receive of both vertical and horizontal polarization. For each pulse there is a measurement of the complex amplitude in each channel, i.e.,

$$[s_{hh}^1 : s_{vv}^1] [s_{hh}^2 : s_{vv}^2] [s_{hh}^3 : s_{vv}^3] \dots [s_{hh}^M : s_{vv}^M]$$

We will assume that M samples are collected for processing, i.e., Note that even though there is cross-polarized radiation received in each channel, this cross-polar contribution can be neglected since the co-polarized received signal is much stronger.

Calculation of the Polarization Measurands

The processing in this case is done by pulse pair mode. However both FFT and random phase processing can be performed if only ZDR and standard moments are requested for output. In any mode, the user may select a clutter filter, but in general this is not recommended for polarization measurements since the clutter filter might interfere with the accuracy of sensitive parameters such as ZDR.

The RVP8 calculates the following polarization parameters:

$$ZDR = 10 \text{ LOG} \left[\frac{S_{hh}}{S_{vv}} \right]$$

$$ZDR = 10 \text{ LOG} \left[\frac{\langle |s_{hh}|^2 \rangle - N_h}{\langle |s_{vv}|^2 \rangle - N_v} \right] + GDR$$

$$RHOHV = |\rho_{hv}(0)|$$

$$PHIDP = \arg[\rho_{hv}(0)]$$

KDP based on least squares fit to PHIDP (see Section 5.7.10).

where the following definitions are used:

$$g_h^r g_h^t S_{hh} = \langle |s_{hh}|^2 \rangle - N_h \quad g_v^r g_v^t S_{vv} = \langle |s_{vv}|^2 \rangle - N_v$$

The noise powers the two channels are denoted as N_h and N_v . The noise corrections to S_{hh} and S_{vv} are optionally configured in the TTY setups. GDR is the total (transmit and receive) differential channel gain. It must be calibrated for the system.

$$\text{dB Value is } GDR = 10 \text{ LOG } xdr \quad \text{where the linear value is } gdr = \frac{g_v^r g_v^t}{g_h^r g_h^t}$$

The correlation function is computed from:

$$\rho_{hv}(0) = \frac{\langle s_{vv} s_{hh}^* \rangle}{\sqrt{S_{hh} S_{vv}}}$$

The gain terms cancel in the calculation of ρ so in the implementation they are simply assumed to be =1.

5.7.7 Case 3: Alternating H/V Transmit: Single-Channel Receiver

Input Receiver Samples

This is the traditional ZDR radar with a high-power fast switch that alternates between horizontal and vertical on each pulse. The switch is made just prior to the transmit pulse so that the transmitter radiates and then receives at a single polarization for each pulse. Thus the samples are:

$$s_{hh}^1 \quad s_{vv}^2 \quad s_{hh}^3 \quad \dots \quad s_{vv}^{M+1}$$

For the discussion below we will assume that there are M+1 total samples with M/2 horizontal pulses indexed by (1, 2, 3...M-1) and M/2+1 vertical pulses indexed at (2, 4, 6, ...M). Note that the processor does not assume that the first pulse in a sequence is horizontal.

Calculation of the Polarization Measurands

The processing is done in pulse pair with optional clutter filter. Again, for accurate ZDR measurements, the clutter filter may interfere.

The RVP8 calculates the following:

$$ZDR = 10 \text{ LOG} \left[\frac{S_{hh}}{S_{vv}} \right]$$

$$ZDR = 10 \text{ LOG} \left[\frac{\langle |s_{hh}|^2 \rangle - N_h}{\langle |s_{vv}|^2 \rangle - N_v} \right] + GDR$$

$$PHIDP = \frac{1}{2} \arg [R_a R_b^*]$$

$$RHOHV = \frac{|\rho_{hv}(T_s)|}{[\rho_{hv}(2T_s)]^{0.25}}$$

KDP based on least squares fit to PHIDP (see Section 5.7.10).

where the following definitions are used:

$$|\rho_{hv}(T_s)| = \frac{|R_a| + |R_b|}{2\sqrt{S_{hh}S_{vv}}}$$

$$\rho(2T_s) = \frac{\left| \sum_{n=1}^{M/2-1} (s_{hh}^*[2n-1] s_{hh}[2n+1] + s_{vv}^*[2n] s_{vv}[2n+2]) \right|}{(M/2-1)(S_{hh} + S_{vv})}$$

$$R_a = \frac{1}{M/2} \sum_{n=1}^{M/2} s_{hh}^{2n-1} * s_{vv}^{2n} \quad \text{and} \quad R_b = \frac{1}{M/2} \sum_{n=1}^{M/2} s_{vv}^{2n} * s_{hh}^{2n+1}$$

The calculation of the channel powers ($\langle |s_{hh}|^2 \rangle$ and $\langle |s_{vv}|^2 \rangle$) is done using alternating pulses in this case. Note that in the calculation of R_b , the RVP8 uses the extra M+1 sample. The gain terms cancel in the calculation of ρ so in the implementation they are simply assumed to be =1.

5.7.8 Case 4: Alternating H/V Transmit: Dual-Channel Receiver

Input Receiver Samples

This is the most comprehensive case of polarization operation since it permits calculation of all of the polarization measurands. In this case the transmitter alternates pulse-to-pulse between horizontal and vertical polarization and the dual-channel receiver provides measurement of both the co- and the cross-polarized return, i.e.,

$$[s_{hh}^1 : s_{vh}^1] [s_{vv}^2 : s_{hv}^2] [s_{hh}^3 : s_{vh}^3] [s_{vv}^3 : s_{hv}^3] \dots [s_{vv}^{M+1} : s_{hv}^{M+1}]$$

We will assume that M+1 samples are collected for processing (an extra sample is required for the calculation Rb per section 5.7.7).

Calculation of the Polarization Measurands

The RVP8 calculates the following:

Co-polar channel measurements

ZDR, PHIDP, RHOHV

Identical to alternating case Section 5.7.7.

Cross-polar channel measurements

LDRH, LDRV, RHOH, RHOV, PHIH, PHIV

Identical to fixed case Section 5.7.9

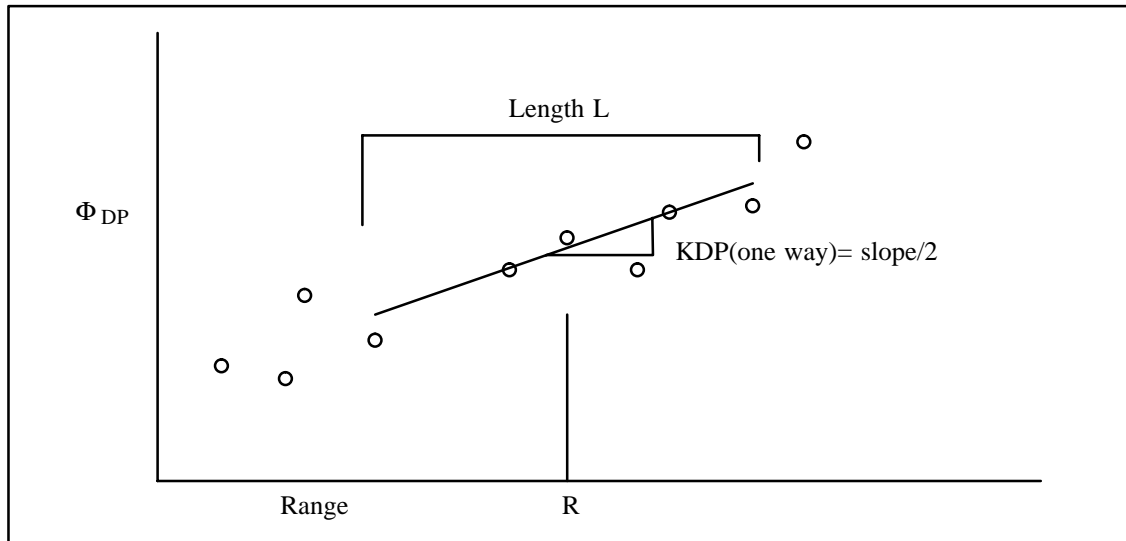
The co-polar channel measurements are exactly as they are for the alternating single-receiver case. The cross-polar measurements are calculated using fixed case algorithms except they are calculated for BOTH H and V polarizations.

5.7.10 KDP Calculation

In all modes that compute PHIDP, the signal processor can also be configured to compute KDP—the specific differential phase in units of degrees per km. This is the range derivative of PHIDP. There are two techniques that have been used to obtain this:

- The smoothed range derivative.
- The slope from a least squares fit.

The RVP8 uses the least squares approach which is shown schematically in the figure below.



The graph shows the thresholded differential phase vs range. This is the starting point for the algorithm. The length scale L is selectable by the user in the TTY setups (mp section, KDP Length in km, default 5.00 km). The KDP value for a bin at range R is computed from a least squares fit that includes points that are within $\pm L/2$ as indicated in the figure. PHIDP is output by the processor on the unambiguous interval of 0 to 180 degrees. Before fitting, the points are first unfolded to a common interval by starting at the left-most point and then moving right assuming that a difference of more than $1/2$ the unambiguous interval is the result of folding. Since it is the slope that is of interest, the absolute interval is not critical, as long as the points are in a common interval.

After fitting, the slope is obtained which corresponds to the 2-way KDP since it is based on the 2-way measurement of PHIDP. To be consistent with most values in the literature, the slope value is divided by 2 so that the final output is the one-way KDP in degrees per km (with a wavelength scaling in the data format).

This procedure is repeated for each bin. Thus if the bin spacing is 250 m, the output bin spacing of KDP will be 250 m. It is required that there be at least 50% of the possible number of bins present in the interval L to calculate a valid KDP, else the KDP is set to the threshold value. Since the input PHIDP values are already thresholded, the only additional threshold on KDP is this 50% rule.

5.7.11 Standard Moment Calculations (T, Z, V, W)

Overview

Standard moments are available for all four of the polarization cases. Since there can be up to four different channels of time series input, there are several choices for computing the standard moments. For example, in the STAR mode (Case 2), the standard moments can be computed from:

- s_{hh} samples
- s_{vv} samples
- Average of the results from the s_{hh} and s_{vv} samples

The third case is handled by averaging the individual channel correlations, and then using the average correlations in the standard moment processing. The averaging must take into account the differential gain of the channels.

The selection of which method to use is made in setup. There are four questions posed in the mp section:

```
T/Z/V/W computed from:  H-Xmt:YES  V-Xmt:YES
T/Z/V/W computed from:  Co-Rcv:YES  Cx-Rcv:NO
```

The first two questions are used to specify that *given a choice* between vertical and horizontal transmit, which transmit polarization to use. Thus for the fixed H or V case where there is only one transmit polarization, this question does not apply. The processor will simply use samples for the polarization is transmitted.

The second two questions are used to specify that *given a choice* between using the co- or cross-polar receivers which one shall be used. This question applies only to systems that can measure LDR, i.e., fixed or alternating transmit, dual-channel receiver systems).

The tables in the sections below summarize the standard moment calculations for each of the four modes and how to configure the four TTY setup responses. Note that these are the only supported modes. Some combinations of responses are unsupported. For example, it is not supported to answer both Co-Rcv: NO and Cx-Rcv: NO.

The top of each table identifies the transmitter/receiver case and what samples are available. The notation HH signifies that the s_{hh} samples are available. The tables use “—” to indicate that either a YES or NO response will cause the same result, i.e., the RVP does not care what response is made. In cases where averaging is performed, the type of weighting used is indicated (either GDR or XDR weighting).

Model for standard moment autocorrelations

The model for the moment autocorrelation calculations is as follows (using R_0 as an example):

$$\begin{aligned} R_0^{hh} &= g_h^r g_h^t S_{hh} + \bar{N}_h & R_0^{vh} &= g_v^r g_h^t S_{vh} + \bar{N}_v \\ R_0^{vv} &= g_v^r g_v^t S_{vv} + \bar{N}_v & R_0^{hv} &= g_h^r g_v^t S_{hv} + \bar{N}_h \end{aligned}$$

where:

$R_0^{hh}, R_0^{vh}, R_0^{vv}, R_0^{hv}$	Are the autocorrelations if the samples at lag zero.
$S_{hh}, S_{vh}, S_{vv}, S_{hv}$	The average power returned from the scatterers.
g_h^r, g_v^r	Receiver gains for horizontal and vertical receive.
g_h^t, g_v^t	Transmitter gains for horizontal and vertical transmit.
\bar{N}_h, \bar{N}_v	Measured noise power of the samples.

In other words, the power that is measured in a channel has two components:

- Backscattered power from the targets that is effected by the transmitter and receiver channel gains.
- Receiver noise which is measured by the RVP8 during noise sampling.

In the case of R1 and R2 autocorrelations, the model is similar except that there is no noise bias.

Calibration Parameters

For dBZ calculations, a calibration constant is required, i.e., the dBZ_0 value in Section 5.4. Depending on the polarization case and the technique selected for standard moment calculation, it may also be required to have GDR and XDR, i.e.,

- GDR- The ratio of the total gains (transmit/receive) of the two co-receive channels.
- XDR- The ratio of the receiver gains in a dual receiver system. This is not required for the Case 2: STAR or the Case 3: Alternating Single-Channel.

The RVP8 supports a single calibration reflectivity dBZ_0 . In all cases it is assumed that the dBZ_0 is for the horizontal co-receive (HH) channel. The only exception is for fixed vertical polarization, in which the algorithm assumes that the calibration is for the vertical co-receive (VV) channel. XDR and GDR are also downloaded and used to adjust the dBZ_0 as required depending on the user's selection for the standard moments. For example, in STAR mode, if the user selects dBZ to be computed from the VV channel, the dBZ_0 for the HH and a GDR adjustment are used to calculate the dBZ in the VV channel.

The remainder of this section discusses the standard moment calculation options for the each polarization case. For a discussion of how to calibrate XDR and GDR see Section 5.7.13.

Case 1H: Fixed Horizontal Transmit, Dual Channel Receive- (HH, VH)				
<i>dBZ_o from HH Channel</i>	TTY Setup Question Responses			
Calculate T, Z, V, W from:	H-Xmt	V-Xmt	Co-Rcv	Cx-Rcv
HH (co) (Recommended)	—	—	YES	NO
VH (xdr⁻¹ weighting)	—	—	NO	YES
HH+VH (xdr⁻¹ weighting)	—	—	YES	YES

HH Channel (co-pol)

This is the recommended channel for the case of linear polarization. The reason is that for linear polarization, the co-pol channel will have the strongest signal. Processing is identical to a conventional radar.

VH Channel (cross-pol)

This choice would be used for circular or elliptic transmit polarization. Since the algorithm assumes that dBZ_o is from the co-pol channel, xdr is used to adjust the autocorrelations as follows:

$$\begin{aligned}
 T_0 &= xdr^{-1} T_o^{vh} \\
 R_0 &= xdr^{-1} R_o^{vh} \\
 R_1 &= xdr^{-1} R_1^{vh} \\
 R_2 &= xdr^{-1} R_2^{vh} \\
 N &= xdr^{-1} N_v
 \end{aligned}$$

These adjusted autocorrelations are then used as per the standard moment processing for a conventional radar. To illustrate this, consider the example of reflectivity processing. The radar equation can be written as (see section 5.2.5):

$$\begin{aligned}
 Z^{vh} &= C S_{vh} r^2 = \left[\frac{Cr_0^2 N_v}{g_v^r g_h^t} \right] \left[\frac{r^2}{r_o^2} \right] \left[\frac{T_o^{vh} - N_v}{N_v} \right], \text{ where } T_o^{vh} = g_v^r g_h^t S_{vh} - N_v \\
 &= \left[\frac{Cr_0^2 N_h}{g_h^r g_h^t} \right] \left[\frac{r^2}{r_o^2} \right] \left[\frac{g_h^r}{g_v^r} \right] \left[\frac{T_o^{vh} - N_v}{N_h} \right]
 \end{aligned}$$

The third term is simply 1/XDR so that we can write:

$$Z^{vh} = \left[\frac{Cr_0^2 N_h}{g_h^r g_h^t} \right] \left[\frac{r^2}{r_o^2} \right] \left[\frac{xdr^{-1} T_o^{vh} - xdr^{-1} N_v}{N_h} \right]$$

In this case, the first term is the dBZ_o for the HH channel. Thus we can use the dBZ_o for the HH channel to calibrate the cross-channel, if we first adjust the cross-channel noise and power by 1/XDR and then normalize by N_h. The reflectivity calculation assumes that the calibrated XDR value compensates for any differences in the radar constant between the two channels, i.e., we do not need to have separate radar constants for the two channels.

HH+VH Channels

This choice would be used for elliptic transmit polarizations that give comparable return signal in both the co- and cross-channels. The approach is to obtain average autocorrelation functions as follows:

$$T_0 = \frac{T_o^{hh} + xdr^{-1} T_o^{vh}}{2}$$

$$R_0 = \frac{R_o^{hh} + xdr^{-1} R_o^{vh}}{2}$$

$$R_1 = \frac{R_1^{hh} + xdr^{-1} R_1^{vh}}{2}$$

$$R_2 = \frac{R_2^{hh} + xdr^{-1} R_2^{vh}}{2}$$

$$N = \frac{N_h + xdr^{-1} N_v}{2}$$

These adjusted autocorrelations are then used as per the standard moment processing for calibration with respect to the HH channel.

Case 1V: Fixed Vertical Transmit and Dual Channel Receive- (VV, HV)				
<i>dBZo from VV Channel</i>	TTY Setup Question Responses			
Calculate T, Z, V, W from:	H-Xmt	V-Xmt	Co-Rcv	Cx-Rcv
VV (co)	—	—	YES	NO
HV (xdr weighting)	—	—	NO	YES
VV+HV (xdr weighting)	—	—	YES	YES

This is the only case for which the calibration constant dBZ_o for the VV channel should be downloaded to the signal processor.

VV Channel (co-pol)

This is the recommended channel for the case of linear polarization. The reason is that for linear polarization, the co-polar channel will have the strongest signal. Processing is identical to a conventional radar.

HV Channel (cross-pol)

This choice would be used for circular or elliptic transmit polarization when most of the return is in the cross-pol channel. Since the algorithm assumes that dBZ_o is from the co-polar channel, xdr is used to adjust the autocorrelations as follows:

$$T_0 = xdr T_o^{hv}$$

$$R_0 = xdr R_o^{hv}$$

$$R_1 = xdr R_1^{hv}$$

$$R_2 = xdr R_2^{hv}$$

$$N = xdr N_h$$

These adjusted autocorrelations are then used as per the standard moment processing with dBZ_o calibrated with respect to the VV channel.

VV+HV Channels

This choice would be used for elliptic transmit polarizations that give comparable return signal in both the co- and cross-channels. The approach is to obtain average autocorrelation functions as follows:

$$T_0 = \frac{T_o^{vv} + xdr T_o^{hv}}{2}$$

$$R_0 = \frac{R_o^{vv} + xdr R_o^{hv}}{2}$$

$$R_1 = \frac{R_1^{vv} + xdr R_1^{hv}}{2}$$

$$R_2 = \frac{R_2^{vv} + xdr R_2^{hv}}{2}$$

$$N = \frac{N_v + xdr N_h}{2}$$

These adjusted autocorrelations are then used as input to the standard moment processing algorithms with dBZ_o calibrated with respect to the VV channel.

Case 2: Simultaneous Transmit and Receive- STAR (HH, VV) Case 3: Alternating Transmit Single-Channel Receive (HH, VV)				
<i>dBZo from HH Channel</i>	TTY Setup Question Responses			
Calculate T, Z, V, W from:	H-Xmt	V-Xmt	Co-Rcv	Cx-Rcv
HH	YES	NO	—	—
VV (gdr^{-1} weighting)	NO	YES	—	—
HH+VV (gdr^{-1} weighting)	YES	YES	—	—

A fundamental difference between these two cases is that for all standard moment processing choices, the STAR case has double the number of samples as compared to the single-channel alternating case. However, the processing is otherwise identical.

HH Channel

Since the HH channel is directly calibrated this is the recommended choice. Processing is identical to a conventional radar.

VV Channel

In this case, GDR is used to adjust the autocorrelations as follows:

$$T_0 = gdr^{-1} T_o^{vv}$$

$$R_0 = gdr^{-1} R_o^{vv}$$

$$R_1 = gdr^{-1} R_1^{vv}$$

$$R_2 = gdr^{-1} R_2^{vv}$$

$$N = gdr^{-1} N_v$$

These adjusted autocorrelations are then used as input to the standard moment processing algorithms with dBZo calibrated with respect to the HH channel.

HH+VV Channels

This approach gives the benefit of doubling the number of samples used for the reflectivity calculation.

$$T_0 = \frac{T_o^{hh} + gdr^{-1} T_o^{vv}}{2}$$

$$R_0 = \frac{R_o^{hh} + gdr^{-1} R_o^{vv}}{2}$$

$$R_1 = \frac{R_1^{hh} + gdr^{-1} R_1^{vv}}{2}$$

$$R_2 = \frac{R_2^{hh} + gdr^{-1} R_2^{vv}}{2}$$

$$N = \frac{N_h + gdr^{-1} N_v}{2}$$

These adjusted autocorrelations are then used as input to the standard moment processing algorithms with dBZo calibrated with respect to the HH channel.

Case 4: Alternating Dual-Channel (HH, VH, VV, HV)				
<i>dBZo from HH Channel</i>	TTY Setup Question Responses			
Calculate T, Z, V, W from:	H-Xmt	V-Xmt	Co-Rcv	Cx-Rcv
HH	YES	NO	YES	NO
VH (xdr⁻¹ weighting)	YES	NO	NO	YES
VV (gdr⁻¹ weighting)	NO	YES	YES	NO
HV (xdr/gdr weighting)	NO	YES	NO	YES
HH+VV (gdr⁻¹ weighting)	YES	YES	YES	NO
HV+VH (xdr & gdr weighting)	YES	YES	NO	YES

HH Channel

Since the HH channel is directly calibrated this is the recommended choice. Processing is identical to a conventional radar.

VH Channel

Processing is identical to Case 1H: Horizontal Transmit HV Processing.

VV Channel

Processing is identical to Cases 2&3:STAR and Single Channel Alternating VV Processing.

HV Channel

The weighting in this case uses both xdr and GDR.

$$T_0 = \frac{xdr}{gdr} T_o^{hv}$$

$$R_0 = \frac{xdr}{gdr} R_o^{hv}$$

$$R_1 = \frac{xdr}{gdr} R_1^{hv}$$

$$R_2 = \frac{xdr}{gdr} R_2^{hv}$$

$$N = \frac{xdr}{gdr} N_h$$

These adjusted autocorrelations are then used as input to the standard moment processing algorithms with dBZo calibrated with respect to the HH channel.

HH + VV Channels

Processing is identical to Cases 2&3: STAR and Single Channel Alternating HH+VV Processing.

HV + VH Processing

The weighting here has to correct for both transmitter and receiver effects in order to use the HH channel dBZ₀.

$$T_0 = \frac{\frac{xdr}{gdr} T_o^{hv} + xdr^{-1} T_o^{vh}}{2}$$

$$R_0 = \frac{\frac{xdr}{gdr} R_o^{hv} + xdr^{-1} R_o^{vh}}{2}$$

$$R_1 = \frac{\frac{xdr}{gdr} R_1^{hv} + xdr^{-1} R_1^{vh}}{2}$$

$$R_2 = \frac{\frac{xdr}{gdr} R_2^{hv} + xdr^{-1} R_2^{vh}}{2}$$

$$N = \frac{\frac{xdr}{gdr} N_h + xdr^{-1} N_v}{2}$$

These adjusted autocorrelations are then used as input to the standard moment processing algorithms with dBZ₀ calibrated with respect to the HH channel.

An example of how this weighted averaging works is given here. Suppose that we want to compute the average of the reflectivities for the VH and HV channels,

$$Z^{hv+vh} = Cr^2 \frac{S_{hv} + S_{vh}}{2}$$

$$= Cr^2 \frac{\frac{T_0^{hv} - N_h}{g_h^r g_v^t} + \frac{T_0^{vh} - N_v}{g_v^r g_h^t}}{2} = \frac{Cr^2}{g_h^r g_h^t} \frac{(T_0^{hv} - N_h) \frac{g_h^t}{g_v^t} + (T_0^{vh} - N_v) \frac{g_h^r}{g_v^r}}{2}$$

but since $xdr = \frac{g_v^r}{g_v^t}$ and $gdr = \frac{g_v^r g_v^t}{g_h^r g_h^t}$

$$Z^{vh+hv} = \frac{Cr^2}{g_h^r g_h^t} \left[\frac{\frac{xdr}{gdr} T_0^{hv} + xdr^{-1} T_0^{vh}}{2} - \frac{\frac{xdr}{gdr} N_h + xdr^{-1} N_v}{2} \right]$$

$$Z^{vh+hv} = \frac{Cr^2}{g_h^r g_h^t} [T_0 - N] = \left[\frac{Cr_o^2 N_h}{g_h^r g_h^t} \right] \left[\frac{r^2}{r_o^2} \right] \left[\frac{T_0 - N}{N_h} \right]$$

The first term in brackets is precisely dBZ₀ for the HH channel. Thus if we average the correlations using the appropriate GDR and xdr weighting as shown above, then the average reflectivity is obtained by using conventional processing with the HH channel dBZ₀.

5.7.12 Thresholding of Polarization Parameters

The thresholding of polarization parameters by the processor eliminates bins with weak or uncertain signals. Note that the thresholding can be disabled if it is desired to see all of the data regardless of the data quality.

All of the polarization parameters are based on power ratios. The RVP8 requires that each power term in a ratio pass a signal-to-noise test similar to the log power test. For example, there are up to four different powers that can be calculated (alternating dual-channel case) so the tests for each of these are:

$$\frac{\langle |s_{hh}|^2 \rangle}{N_h} > N_{thresh}$$

$$\frac{\langle |s_{hv}|^2 \rangle}{N_h} > N_{thresh}$$

$$\frac{\langle |s_{vv}|^2 \rangle}{N_v} > N_{thresh}$$

$$\frac{\langle |s_{vh}|^2 \rangle}{N_h} > N_{thresh}$$

where the linearized threshold that is input as the dB LOG threshold, i.e.,

$$N_{thresh} = 10^{LOGthresh/10}$$

For example, a valid LDRH requires both a valid S_{hh} and a valid S_{vh} . The parameters RHOH and PHIH have the same requirement since they are the magnitude and phase of the cross-correlation function which is based on S_{hh} and S_{vh} .

There are two exceptions:

ZDR

ZDR requires that both S_{hh} and S_{vv} pass the signal-to-noise tests noted above. However, ZDR can be additionally thresholded by any of the other threshold parameters (LOG, SIG, SQI, CSR) similar to a standard moment. See section 5.3 for a description of the standard moment thresholding.

PHIDP for single channel alternating case

PHIDP requires that both S_{hh} and S_{vv} pass the signal-to-noise tests noted above. In the single channel alternating case, PHIDP must also satisfy the additional test that the Doppler velocity at the range bin must be valid, i.e., not thresholded by its own criteria. This is because the algorithm for PHIDP in this case essentially subtracts the phase change due to the Doppler velocity. If the Doppler velocity is uncertain, the algorithm cannot produce reliable results.

5.7.13 Calibration Considerations

Polarization systems require additional calibration as compared to conventional systems. There are three aspects to the calibration:

- dBZ_0 measurement in both channels for dBZ and dBT calibration.
- GDR measurement for ZDR calibration.
- xdr measurement for LDR calibration.

These are discussed below.

dBZ₀ Calibration for dBZ

The RVP8 supports separate calibration of both polarization channels. Measurement of dBZ_0 for each channel of a dual polarization system is identical to the conventional radar case described in Section 5.4. Note that for a single-channel switching system, the only difference between the horizontal and vertical signal paths occurs after the high power switch, i.e., differential insertion loss of the switch itself and any differential insertion loss of the waveguides and feed after the switch. This means that for single-channel switching systems it may be sufficient to calibrate at one polarization and then adjust the calibration of the other channel by the differential gain GDR (see below).

GDR Calibration for ZDR

GDR is the relative between the co-polarized channels including both transmitter and receiver gain, i.e.,

$$GDR = 10 \text{ LOG } \frac{g_v^r g_v^t}{g_h^r g_h^t} \quad \text{and} \quad gdr = \frac{g_v^r g_v^t}{g_h^r g_h^t}$$

GDR is input into the processor as a dB value. However, for analyses in this chapter, the linear gdr value is sometimes more convenient.

In principle, if dBZ_0 could be calibrated perfectly in both channels, measurement of GDR would not be required. In practice, this is not possible because dBZ_0 cannot be calibrated to an absolute accuracy sufficient for ZDR, i.e., to 1/16th of a dB. Therefore, the RVP8 uses the GDR approach.

Since GDR includes both transmitter and receiver differential gains, accurate calibration requires that an actual target be observed. One way to do this is as follows:

- Set the GDR to be 0 dB using your application software (e.g., for SIGMET IRIS systems in the setup utility RVP section). Disable clutter filtering for ZDR in either your application software (by selecting filter 0) or explicitly in the RVP8 TTY setups mp section.
- Place the antenna at 90 degrees elevation (vertical incidence) during moderate to heavy rain. The melting layer should be at a height that is well above the recovery zone of the T/R and in the antenna “far zone”. A melting layer higher than 2 km is suggested, but the specific characteristics of the radar should be considered.

- Collect ZDR data at vertical incidence while the antenna is rotating in azimuth.
- Use a separate application program to average the ZDR values around a full 360 degrees at each range bin (height). Generate a plot of 360-average ZDR vs height.
- You should observe that the average ZDR values in regions of strong signal (>20 dB SNR) below the bright band are approximately constant with height. This is the value that should be used in your application software for GDR.
- Enter the value and repeat the calibration to verify that the average ZDR is now 0 dB.

The rationale for this approach is as follows. When viewed at vertical incidence, rain should have a ZDR of 0 dB since the drops will all appear circular. The reason for averaging over 360 degrees is to cancel-out effects from sidelobe contamination from nearby ground targets and other artifacts of the antenna/feed/radome system. For example the radome may have an obstruction light on the top. Some of these artifacts can be minimized by assuring the the weather targets are strong, i.e., heavy rain is preferred for this calibration.

XDR Calibration for LDR

XDR is the relative gain in dB between the co- and cross-receiver channels for LDR measurements. Analogous to GDR, it is defined as the dB value of the ratio of the vertical to horizontal receiver gains, i.e.,

$$XDR = 10 \text{ LOG } \frac{g_v^r}{g_h^r} \quad \text{and} \quad xdr = \frac{g_v^r}{g_h^r}$$

Three techniques for calibration of XDR are discussed. It is recommended for the transmitter to be off for all of these methods.

- **1. Solar method**
Use the sun to measure LDR. The measured value of LDR is then the XDR offset. LDR should be measured in fixed mode for both LDRH and LDRV. The values should be reciprocal (e.g., +1 dB and -1 dB). Use the average of the absolute value if they are not precisely reciprocal (e.g., for +1.4 and -1.2 use 1.3). Finally after inputting the XDR value, retest to verify that the sun has been properly corrected to have zero LDR.
- **2. Signal generator method with connection to waveguide**
Connect a signal generator with a splitter to both channels and measure XDR directly. This does not account for any effects that are before the coupler (e.g., waveguide, feed, radome, antenna gain).
- **3. Linear feed horn remote radiator method**
Use a calibrated linear feed horn with an RF source located several hundred meters from the radar. Maximize the H channel return and measure the response using the RVP8 pr command "Filtered" power in the "Primary Channel". Now rotate the feed horn to vertical and maximize the power in the "Secondary Channel". The difference in dB is XDR. Note that signal multi-path effects could bias the results from this technique.

In all cases it is recommended that for the calibration, XDR be set to 0 dB in the application user software and that the RVP8 TTY setups be configured as follows:

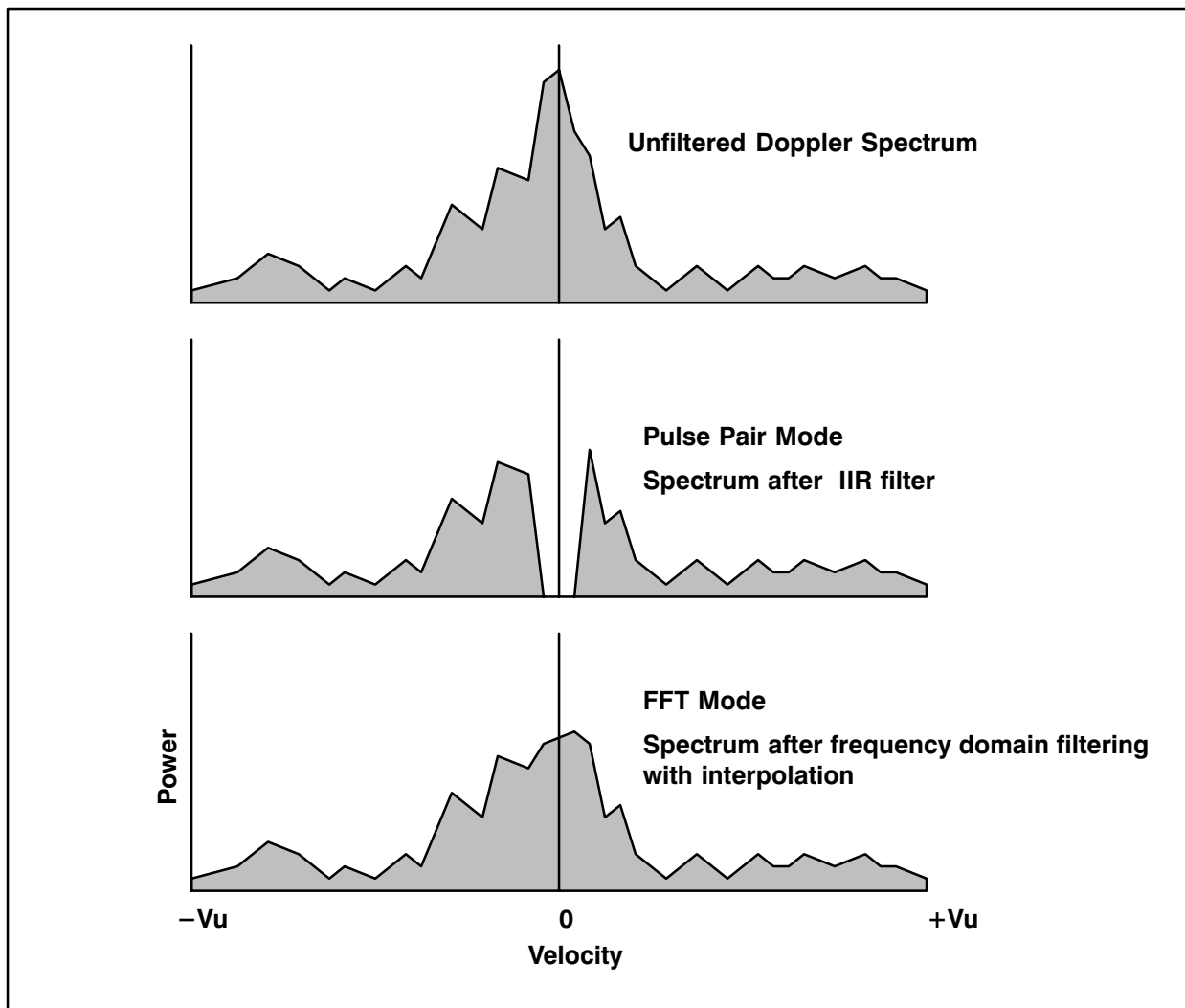
- Noise correction enabled for LDR and noise sample taken prior to the measurements (with care not to sample with a test signal turned-on or while looking at the sun).
- Clutter correction disabled for LDR.

5.8 FFT Mode

5.8.1 Overview

The RVP8 can perform FFT processing of the I and Q time series. This is indicated by the inset box in Figure 5-1. The major difference between FFT and pulse pair processing is the way in which clutter filtering is performed. The pulse pair mode uses a time domain IIR filter while the FFT mode uses a frequency domain filter. The advantage of the FFT approach is that it is less destructive to overlapped weather (near zero velocity) than the IIR filter since the clutter filter algorithm attempts to interpolate over the weather (see Figure 5-8). This results in more accurate estimates of velocity, width and clutter correction. Because the clutter correction is more accurate, the resulting reflectivity estimates are more accurate.

Figure 5-8: Comparison of Pulse Pair and FFT Clutter Filters



5.8.2 FFT Implementation

Figure 5–9 shows a data flow diagram for the FFT processing. In the example, 50 pulses of I and Q are processed. Recall that the I and Q samples refer to a single range bin with an I and Q sample generated for each pulse.

Sampling Division

The first step of processing is to split the I and Q values into two groups. The FFT algorithm requires the number of samples to be a power of two. This is not very convenient for radar sampling since it effectively quantizes the scan rates to be powers of two. The processing by the RVP8 allows an arbitrary number of pulses to be handled by splitting the samples into two groups — the first 2^N and last 2^N pulses. An FFT is performed on each group and then the results are averaged. This allows an arbitrary number of samples to be processed. The maximum number of samples is 255 which corresponds to performing FFT's on two groups of 128 pulses with 1 sample of overlap.

If the requested number of pulses is exactly a power of 2, then the samples are not split and only a single FFT is done.

Window and FFT

After splitting the samples, a weighting function, or window, is applied. The FFT algorithm is then applied and the magnitude squared of each component is calculated to yield the Doppler power spectrum. Figure 5–10 shows the response of the Doppler power spectrum to a ground clutter target for each of the three available FFT windows. The schematic Doppler spectra in the example do not show the details of the side lobe structure of the impulse response (see Oppenheim and Schaffer, 1975 p. 243–245).

- **Rectangular Window** — This is really the no-window case since it is equivalent to multiplying each sample by 1. This window has a sinc/x type impulse response, i.e., the impulse response is very narrow which is good, but the side lobes of the Doppler spectrum are rather high — ~ 20 dB near the peak. Thus it is not the best choice for high performance clutter cancelation since the high sidelobes will mask weak weather targets.
- **Hamming Window** — This common window is well matched to magnetron systems. The peak of the impulse response is broader than that for the rectangular window, but it has -40 dB peak-to-noise sidelobes. Weak weather targets above this level will not be obscured by these sidelobes. The phase noise of a magnetron system (~ 1 degree) will typically be slightly larger than the sidelobes caused by the window which means that the window is well matched to the magnetron performance.

Figure 5-9: FFT Processing — 50 pulse example

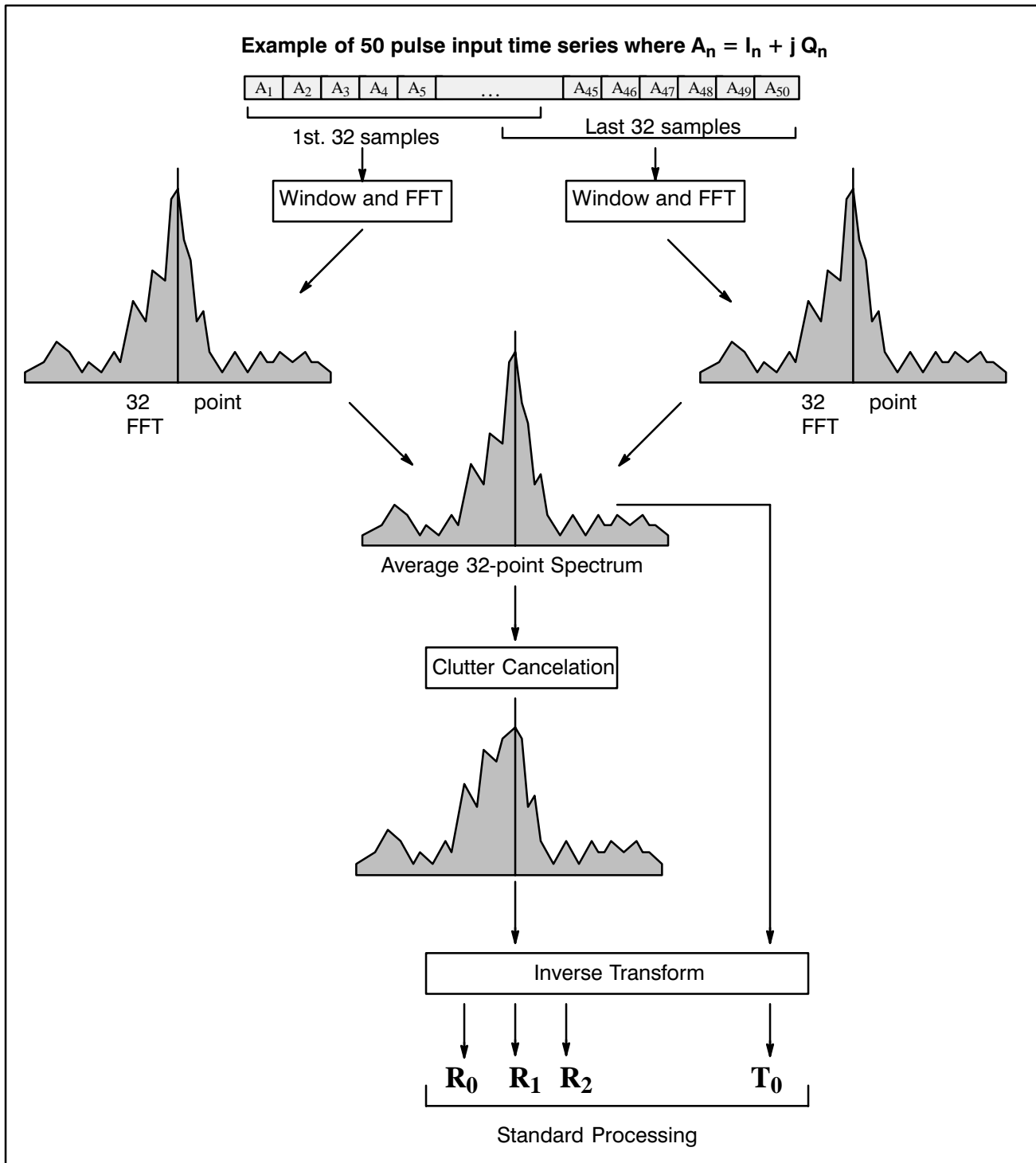
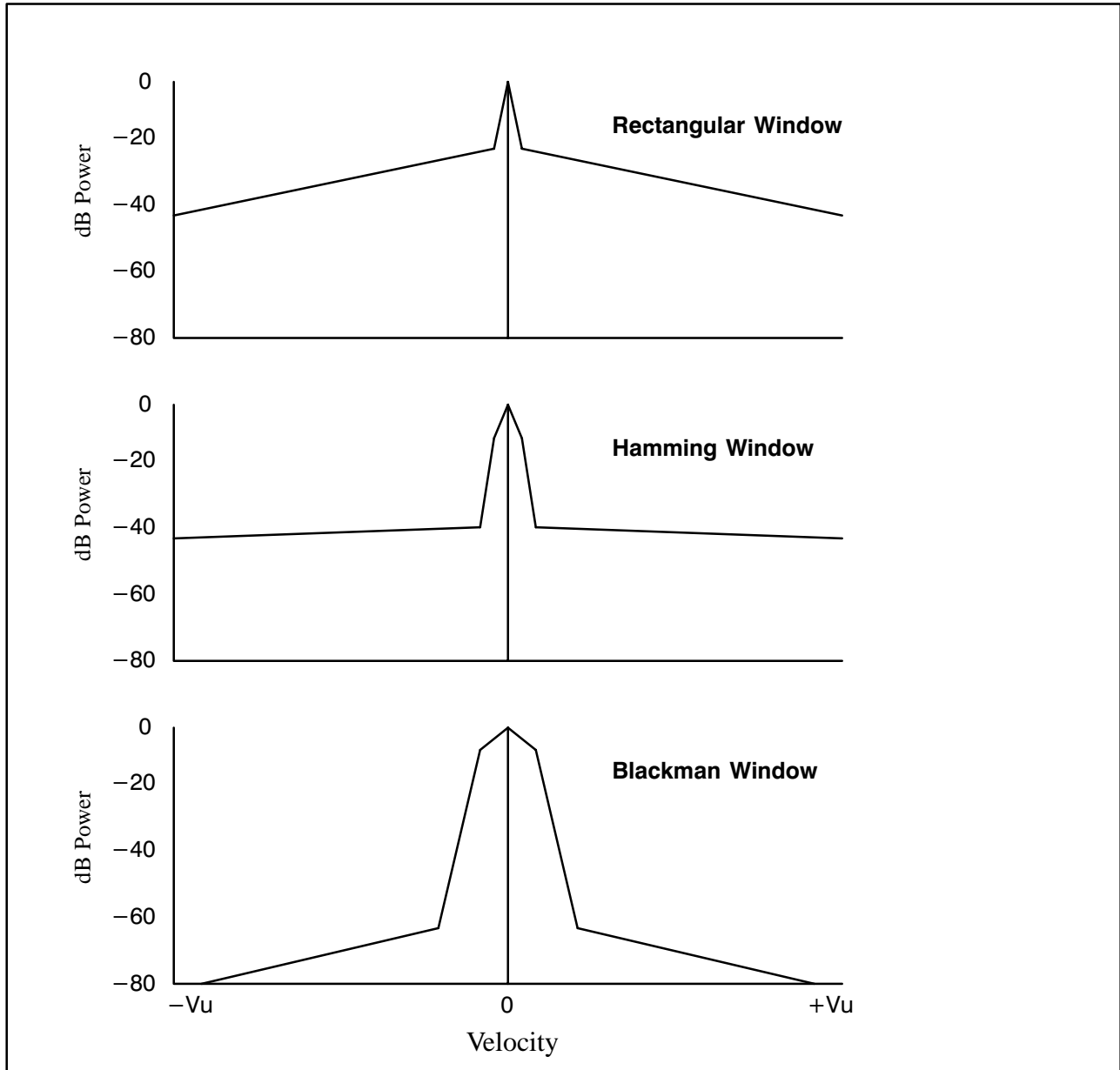


Figure 5–10: Effect of Windowing on FFT Response to Ground Clutter



- **Blackman Window** — This is the most aggressive window for clutter cancelation and is appropriate only for Klystron systems that can achieve very low phase noise (~ 0.1 degree). The peak is the broadest of all the windows, but the side lobes are the lowest. It is not recommended for magnetron systems except for performance testing.

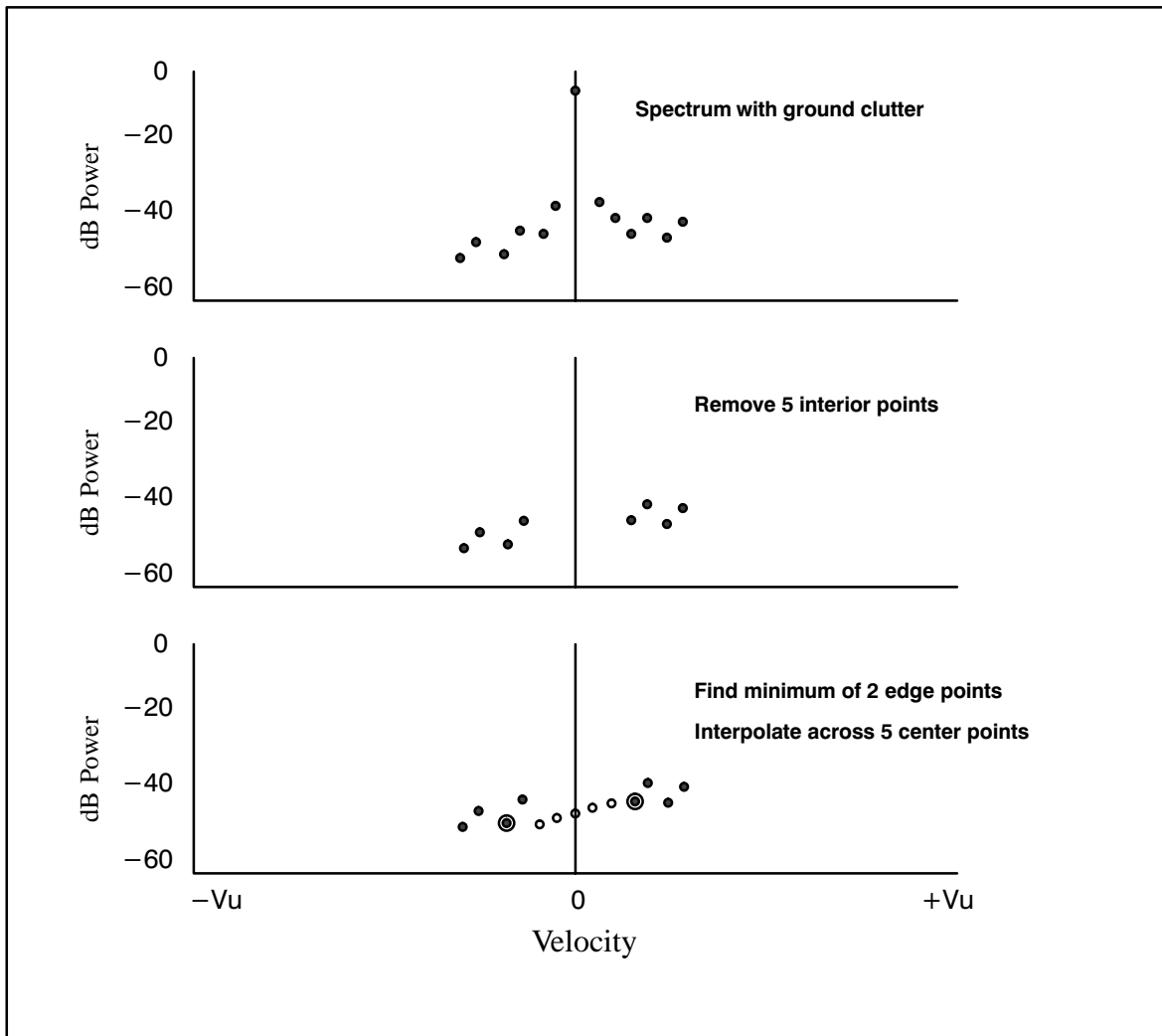
FFT Averaging

The power spectrum from the first group of samples is averaged with the power spectrum from the last group of samples. Note that if the total number of samples is exactly a power of two, then this step is skipped. Averaging the two power spectra from the overlapping sample groups effectively captures the information from all of the samples. The result is a smoother power spectrum than weather of the individual spectra.

Clutter Cancellation

The clutter cancelation is done by an interpolation technique shown in Figure 5–11. In general, the technique discards a selectable number of center points. The algorithm then takes the minimum value of a selectable number of edge points next to the discarded points. The minimum value on each side is used to interpolate across the points that were removed.

Figure 5–11: Example of FFT Clutter Filter in Frequency Domain



This procedure preserves the noise level and/or overlapped weather targets. The result is that more accurate estimates of dBZ are obtained. In extreme cases when the weather spectrum is very narrow, there can still be some attenuation of weather of a broad filter is selected.

Inverse Transform

After clutter removal, an inverse DFT (Discrete Fourier Transform) is performed to obtain the autocorrelations R0, R1 and R2 (optional). The total power T0 in the unfiltered power spectrum is also computed by summing the spectrum components. Thus the final output of the FFT approach is identical to the pulse pair approach except that the clutter filtering is performed in the frequency domain.

5.9 Random Phase 2nd Trip Processing

5.9.1 Overview

Second trip echoes can be a serious problem for applications when the radar is operated at high PRF (e.g., >500 Hz). Second trip echoes are caused by the range aliasing of targets. They appear as false echoes on the display, usually elongated in the radial direction. On Klystron systems they will have valid Doppler velocities. On magnetron systems, the Doppler velocities are not valid, but the noise from the 2nd trip echoes can obscure valid first trip velocity information.

The RVP8 has optional random phase processing for the filtering and recovery of second trip echoes. Details of the technique are proprietary to SIGMET, Inc. However, the general principle is described here, along with a discussion of the various configuration options to optimize the algorithm performance.

The information that is used to separate the first and second trip echoes is the phase. For a magnetron radar, the phase of each pulse is different. This means that when 1st. and 2nd trip echoes are received simultaneously, the phase of the first trip return is different from the phase of the second trip return. For a magnetron radar, the RVP8 measures the phase of the transmitted pulse and the phase locking is done digitally as opposed to the traditionally locking COHO. For a Klystron radar, the phase is controlled by the RVP8 via a digital phase shifter that is precisely calibrated. Typically the Klystron COHO is phase shifted so that each transmit pulse has a different phase. The sequencing is controlled by the RVP8.

5.9.2 Algorithm

Figure 5–12 shows a schematic of the data processing for random phase. The figure shows the Doppler spectra for the 1st. and 2nd trip in the various processing stages. The vertical scale is in dB and the horizontal scale is velocity. In this example, the second trip echo is shown as being stronger than the first trip echo (usually the reverse is true).

Ideal 1st and 2nd Trip Echoes

The ideal 1st and 2nd trip echoes represent the echoes as they would appear individually. The ideal 1st trip echo is the echo that would be measured if there were no 2nd trip echo interference. The ideal 2nd trip echo represents what would be measured if there were no 1st trip echo interference. If there is no interference from the other trip, a standard Klystron system can measure the ideal spectra, but there is no way to know whether the echoes are in the 1st or 2nd trip.

Raw 1st and 2nd Trip Echoes

This figure shows how the echoes from the first trip and second trip interfere with each other. For the case of a standard magnetron system, the first trip echo is coherent, while the second trip echo is incoherent (white noise) since the phase of the second trip echo is random. This is because the receiver is phase locked only to the first trip.

Another way to implement a magnetron system is to let the COHO free-run (rather than phase locking to the transmit pulse), measure the phase of each transmit pulse and digitally correcting for the transmit phase. Using this digital phase locking technique, the RVP8 can phase lock or “cohere” to either the first or the second trip.

Using this technique alone, it is possible to distinguish between 1st and 2nd trip echoes for the case when the echoes are not overlapped. In other words, the echoes will appear as the idealized 1st and 2nd trip echoes. This range de-aliasing effectively doubles the range of the radar. The problem is that when echoes are overlapped, the noise contamination from the stronger echo will make it impossible to measure the weaker echo. This is illustrated in the figure. Thus if the first trip echo has a good signal-to-noise ratio of 10 dB, then the 2nd trip echo will have a signal-to-noise-ratio no better than -10 dB. This is the fundamental problem with using phase alone to separate the 1st and 2nd trip echoes.

Filtered 1st and 2nd Trip Echoes

Since the strong echo generates noise that obscures the weaker echo, the approach used in the RVP8 is to filter the echo from the other trip — the whitening filter. This is shown in the figure. The adaptive whitening filter removes both the clutter and the weather. All of the phase information for the other trip is then contained in the white noise portion of the spectrum. Note that the phase information under the coherent echo that is removed will be dominated by the coherent echo, i.e., the other trip phase information will be contaminated. For this reason, the filtering should effect as small a region of the spectrum as possible.

5.9.3 Tuning for Optimal Performance

The Random Phase algorithms are controlled by the same collection of setup and operational parameters that apply to all of the other processing modes, e.g., choice of sample size, clutter filter, angle sync, calibration, etc. However, a few parameters are special to Random Phase mode, and these are described below.

Secondary SQI Threshold

In standard Doppler processing, an SQI threshold is normally not applied to Reflectivity data because it would cause those data to be rejected in regions of high spectral width. In Random Phase mode we need to relax this convention because reflected power can only be assigned to a particular trip when it is coherent within that trip. Incoherent echoes, regardless of their strength, can not be placed into either trip.

Thus, an SQI threshold is required to qualify Reflectivity data in Random Phase mode. The RVP8 defines a secondary SQI threshold SQI_2 which is computed from the standard threshold value simply as:

$$SQI_2 = Offset + (Slope \times SQI)$$

Where *Slope* and *Offset* are the Random Phase SQI threshold parameters defined in the **Mf** setup section. The factory default values are (*Slope* = 0.50) and (*Offset* = - 0.05), i.e., the secondary threshold is a little less than half of the standard value. The Random Phase

algorithms check whether the SQI of each recovered trip is less than the secondary SQI threshold, and if so, the LOG portion of the data are rejected. This SQI test is necessary for a clean LOG picture, but we need to use a more permissive (lower) threshold value than would usually be applied to the Doppler data alone.

The *Slope* and *Offset* values should be adjusted so that the density of speckles in Random Phase LOG data is approximately the same as the density of speckles in FFT velocity data for a given primary SQI value. You may then adjust the primary SQI threshold to achieve the appropriate tradeoff of speckles vs. sensitivity for your system in all modes of operation. Even with proper adjustment, it is normal for Random Phase *dBZ* and *dBZ* data to show “holes” in regions of weather that have high turbulence or shear. These dropouts will usually match up with similar gaps in the velocity and width data, both of which are traditionally thresholded by SQI.

Maximum Power Ratio Between Trips

The adaptive filtering that is performed on the data for each trip greatly extends the visibility of a weak echo that is overlapped with a much stronger one. In practice, the filtering process is often able to remove 25-35dB of dominant power in order to reveal a much weaker echo in the other trip. The performance depends on many factors, primarily the spectral width of the dominant echo, and the overall stability of the radar system.

The difficulties of removing a dominant “other trip” echo from a weather signal are analogous to the challenge of removing a dominant clutter target from that same signal. In both cases we are trying to extract a weak weather signature using a filtering procedure that relies on the spectral confinement of the stronger signal. The RVP8 already has a parameter that can be adjusted to control sub-clutter visibility, i.e., the Clutter-to-Signal Ratio (CSR). Just as the CSR applies to the clutter filters, it can likewise be used to place similar limits on the depth of visibility of the adaptive filters.

As an example, suppose that the RVP8 is operating in Random Phase mode at a PRF of 1500Hz, and is observing widespread weather having uniform intensity in both the first 100km trip and the second 100km trip. If the CSR were set too conservatively at only 15dB, then the algorithm would generally be blind to second-trip weather in the range interval from 100km to 117.8km.

The explanation for this can be found in the $1/r^2$ geometric correction for weather echo intensity. At ranges less than 17.8km, the first trip weather would generally dominate the second trip weather by more than 15dB. Thus, the initial 17.8km ring of second trip data would be rejected by the CSR criteria. However, if the CSR were increased to 30dB, then the size of this missing ring would be reduced to only 3.2km.

If the CSR is set too low you will notice an abrupt ring of missing data in the beginning of the second trip. If set too high, there will be speckles and other spurious effects within this same interval. The optimum setting should strike a balance between these two effects.

R1 vs. R2 Algorithms

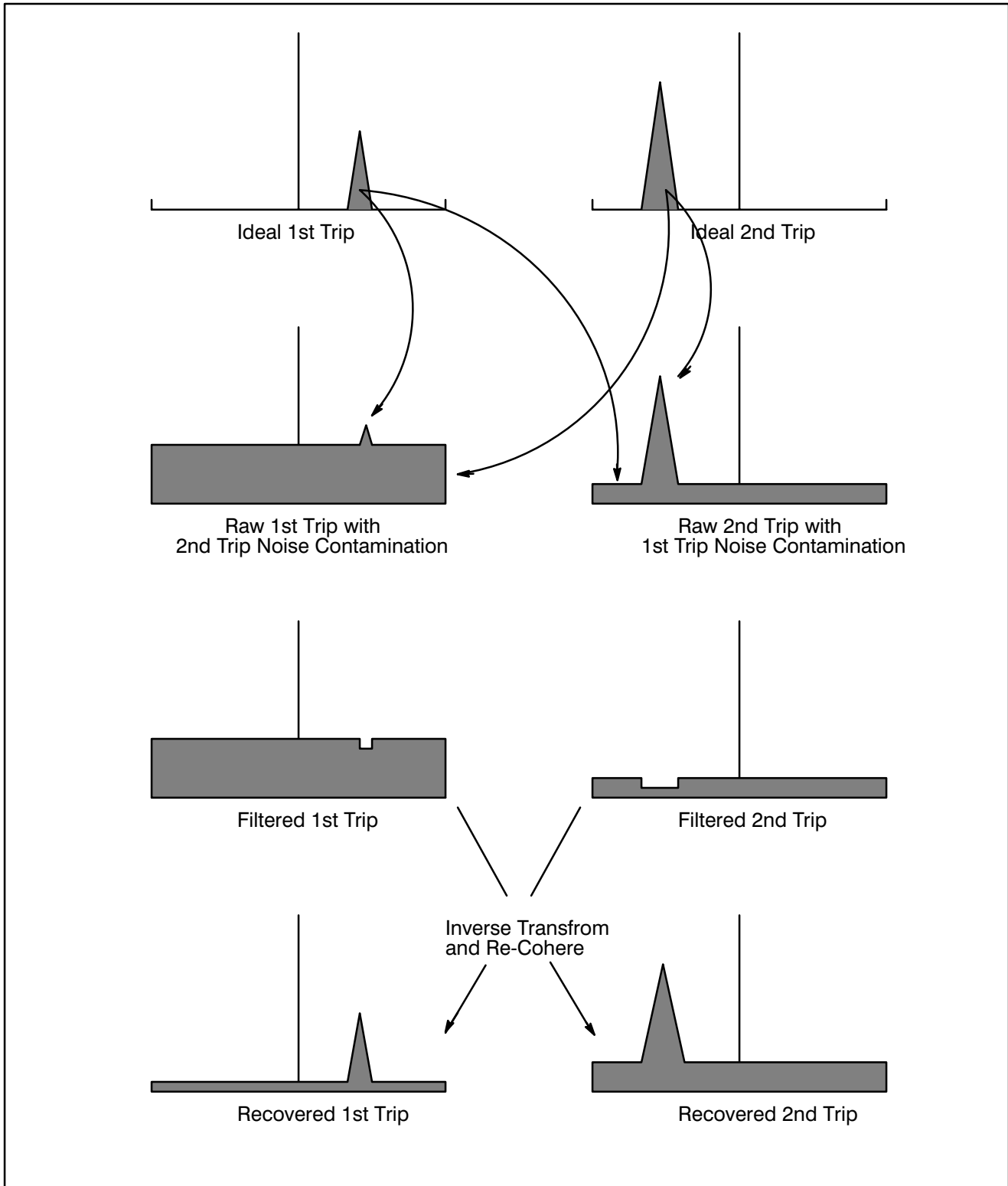
The Random Phase algorithms for adaptive filtering and separation of trips relies on having the best possible information about the weather's SNR and spectral width. Thus, the “R2” Doppler algorithms are always used, regardless of the setting of the R1/R2 flag in the user's operational parameters.

Random Phase and Dual PRF

The random phase processing works seamlessly with the dual PRF processing to provide advanced range and velocity ambiguity resolution. Both the first and 2nd trip echoes can be recovered and displayed to a maximum range of 2X the unambiguous range corresponding to the high PRF.

For optimum performance, the 2D 3x3 speckle filter should be used to smooth the 2nd trip seams that occur for each ray. In fact, this smoothing of the 2nd trip seam makes the dual PRF random phase mode work even better than the single PRF random phase.

Figure 5-12: Random Phase Processing Algorithm



5.10 Signal Generator Testing of the Algorithms

This section describes a variety of IF signal generator tests that can be used to verify correctness of the RVP8 processing algorithms. These tests are routinely performed at SIGMET whenever new algorithms and/or major modes are added to the processor. We have include a few of the test descriptions here so that they can be used by customers who need to debug their systems, or who want to better understand how they work. Additional tests for receiver sensitivity and dynamic range can be found in Appendix E.

5.10.1 Linear Ramp of Velocity with Range

Suppose that a continuous-wave IF waveform has an instantaneous frequency $f(t)$ in Hertz (cycles/sec). Consider a range bin located at time τ_{bin} within a set of pulses that are separated by $\tau_s = 1/PRF$. The phase measured at that bin on the n^{th} pulse will be the integral of the frequency within that pulse starting from range zero (since the RVP8 is phase locked to range zero):

$$\Phi_n = \int_{n\tau_s}^{n\tau_s + \tau_{bin}} f(t) dt$$

If we assume that the input frequency is a linear Frequency Modulation (FM) at the rate of M cycles/sec/sec on top of a base frequency T_o , then:

$$\Phi_{n+1} - \Phi_n = \int_{(n+1)\tau_s}^{(n+1)\tau_s + \tau_{bin}} (T_o + Mt) dt - \int_{n\tau_s}^{n\tau_s + \tau_{bin}} (T_o + Mt) dt = (M\tau_s)\tau_{bin}$$

which, remarkably, is independent of both T_o and n . Thus, a linear FM input signal produces a fixed (I,Q) phase difference from pulse-to-pulse at any given range. The magnitude of the phase difference is proportional to the range, and the slope is $(M\tau_s)$ cycles for each second of delay in range. For example, if the test signal generator is sweeping 100KHz every two seconds, then the velocity observed at a range of 300km at 250Hz PRF will be:

$$\Phi_{n+1} - \Phi_n = \left(\frac{100 \text{ KHz}}{2 \text{ sec}} \right) \times \left(\frac{1}{250} \text{ sec} \right) \times (300 \text{ km}) \times \left(\frac{6.6 \mu \text{ sec}}{1 \text{ km}} \right) = 0.40 \text{ cycles}$$

We would thus observe a velocity of $(0.8 \times V_u)$ at 300km, where V_u is the unambiguous Doppler velocity in meters/sec. Note that these phase difference calculations have made no assumptions about the RVP8 processing mode, and thus are valid in all major modes (PPP, FFT, DPRT, RPH), as well as in all Dual-PRF unfolding modes.

Interestingly, this simple FM signal generator will also produce valid second trip velocities that can be seen during Random Phase processing. This follows from the above analysis because we've never assumed that τ_{bin} was smaller than τ_s , i.e., it is fine for the range bin to be located in any higher-order trip.

5.10.2 Verifying PHIDP and KDP

The PHIDP and KDP processing algorithms can be tested using CW signal sources at IF. In the alternating-transmitter single-receiver case, a single FM signal generator is modulated with an RVP8 polarization select line so that slightly different frequencies are generated for the H and V pulses. A maximum FM depth of several kilohertz is all that is required. In the dual-receiver case, two (unmodulated) signal generators are used for each of the H and V intermediate frequencies, and one or the other is detuned slightly from its correct center frequency. In either case the frequency difference that produces a KDP value of 1.0 degree/km will be:

$$\left(1.0 \frac{\text{degree}}{\text{km}} \right) \times \left(\frac{1}{360} \frac{\text{cycles}}{\text{degree}} \right) \times \left(299792 \frac{\text{km}}{\text{second}} \right) = 833 \frac{\text{cycles}}{\text{second}}$$

5.10.3 Verifying RHOH, RHOV, and RHOHV

These three terms measure the normalized cross-channel covariance in a polarization radar. They all are computed in essentially the same way having the form:

$$RHOAB = \frac{\langle s_A^n s_B^{n*} \rangle}{\sqrt{\langle s_A^2 \rangle \langle s_B^2 \rangle}}$$

Where the s_A^n and s_B^n are complex (I,Q) vectors from two receiver channels A and B, and “ $\langle \rangle$ ” denotes expected value. This suggests that some form of amplitude modulation (AM) of the input signal might be helpful.

Suppose that the s_A^n and s_B^n samples are coming from two signal generators installed on a dual-receiver system, and that only the B-Channel is AM modulated so that:

$$|s_A^n| = \{ S_A, S_A, S_A, S_A, S_A \dots \} \quad , \quad |s_B^n| = \{ S_B, 0, S_B, 0, S_B \dots \}$$

Then the above estimator reduces to:

$$RHOAB = \frac{(\frac{1}{2}) S_A S_B}{\sqrt{S_A^2 \times (\frac{1}{2}) S_B^2}} = 0.707$$

A simple way to create these data is to set the A-Channel siggen for 95% AM depth, and use a sinusoidal modulation source of, perhaps, 400Hz. The reason for not choosing 100% depth is that we would loose the Burst phase reference when the amplitude became smallest. The 26dB reduction in S_B is a close enough approximation to zero in the above formula.

If we now observe the two receive channels with the RVP8 at a PRF of 800Hz, we will see the various RHOAB terms varying with range; reaching a high value of 1.00, and a low value of 0.707. The plots will be nearly stationary on the **scope** screen because the PRF is almost precisely twice the modulation rate (though they are free-running relative to each other).

Adjusting the amplitude of either signal generator will not affect the ρ terms, but it will have an interesting effect on SQI. If (T,Z,V,W) are being computed from both channels combined, then the SQI is:

$$SQI = \frac{S_A^2}{S_A^2 + (\frac{1}{2}) S_B^2}$$

If we solve this equation for $SQI=0.5$ we find that the individual S_A terms must have twice the power of the individual S_B terms. This can be checked by adjusting either signal generator until the minimum plotted SQI is 0.5, and then verifying that the average H and V powers are identical; or, equivalently, that ZDR , $LDRH$ and $LDRV$ are zero.

The linear FM ramp described in Section 5.10.1 can also be used as a test of $RHOAB$ in a dual-receiver system. With one siggen modulated and the other fixed, one receive channel will appear to be rotating relative to the other. If the FM modulation is such that $1/N$ of a full revolution occurs per pulse at a given range, then if the sample size is N pulses we will observe $RHOAB = 0$ at that range. In fact, the plot of $RHOAB$ will show a characteristic $\sin(x)/x$ behavior as a function of range.

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.

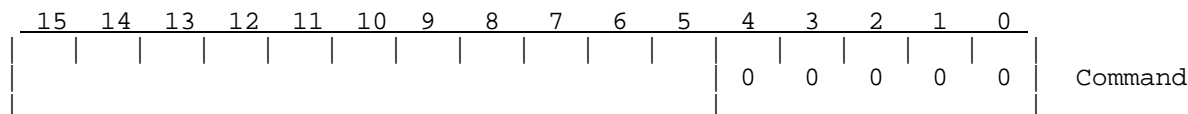
When the output FIFO is full and the RVP8 has the next word ready for output, there is another way that the idle wait loop can be exited, that is, if the processor detects that the user is performing a write I/O cycle. Since the user should have been reading data by now, the presence of a write cycle 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 x (N-1). The Range resolution is specified by a TTY setup question (see section 3.3.5), in the range 50 through 133 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. Because of memory constraints, the RVP8 uses only the first 5600 bits in the mask.

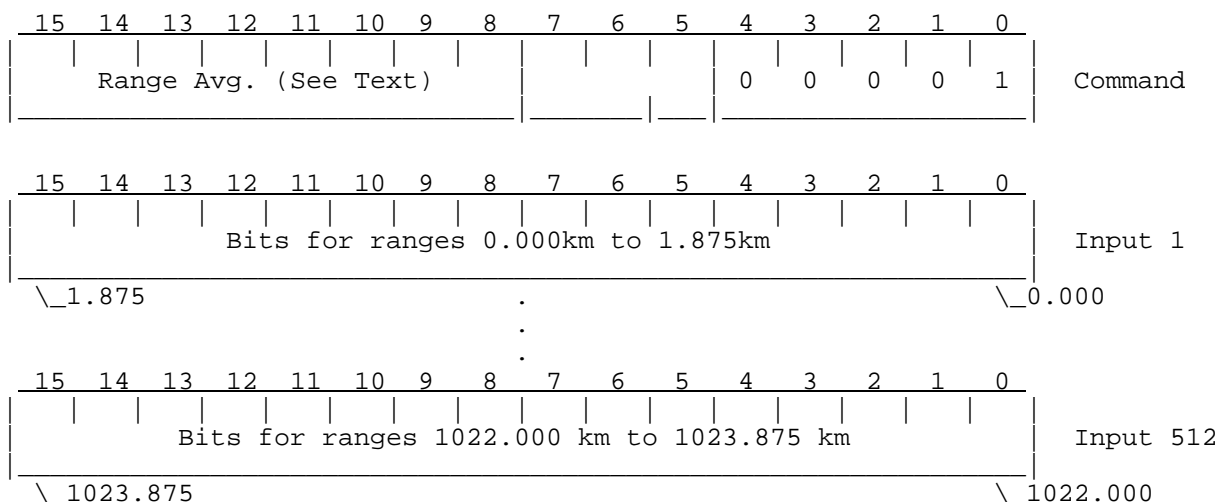
According to the range bins that are selected in the mask, the signal processor computes and stores internally a range normalization table which is later used to convert receiver intensity levels into reflectivity levels in dBZ. Note that the LRMSK command implicitly specifies the number of bins to be processed and output. The maximum bin count is 2048, 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 2048 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.

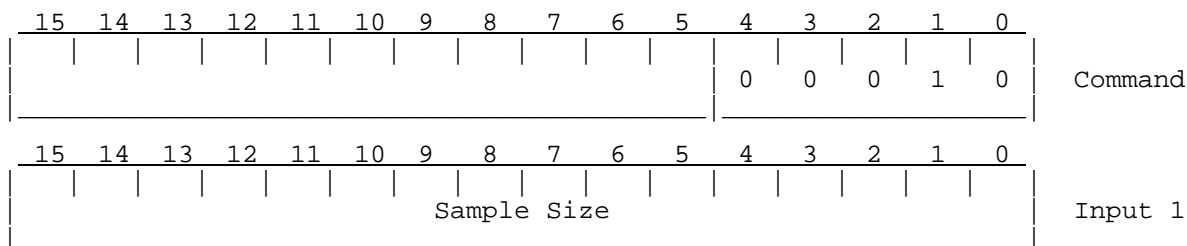
Incompletely averaged bins are discarded by the LRMSK command. In the above example, if the averaging were set to two so that triples of samples were summed, then only 33 bins would be output. This is because the 100-bit mask left a dangling 100th sample. In the extreme case where there are not enough 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.

When the RVP8 is operating in fast-switching dual-polarization mode (See SOPPRM Command, word #2), the maximum number of bins that can be processed is half the usual maximum. This is because duplicate internal memory is needed to hold the states of the clutter filters separately for each polarization. If polarization switching is on and a range mask is loaded having more than 1024 bins, then an error bit is set (GPARM Command, word 9) and only the first 1024 bins are used. Conversely, if a range mask has already been loaded having more than 1024 bins and the user attempts to enter the switching polarization mode, then that request is denied and the RVP8 continues to operate in its prior fixed polarization manner. Whenever the number of range bins is less than 1024, then none of the above interactions ever arises.

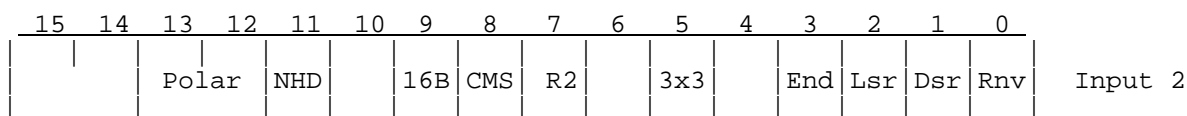


6.3 Setup Operating Parameters (SOPRM)

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.



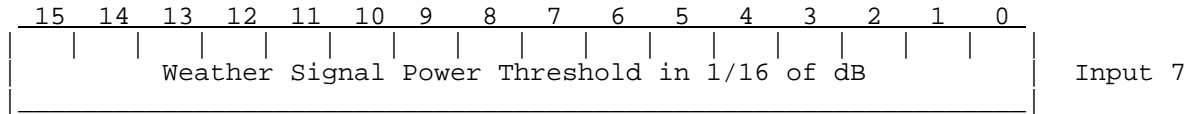
The sample size is continually adjustable from 1 to 256 pulses. However, during the alternating polarization mode, the sample size must be even. If an odd value is entered it is rounded up by one in that case.



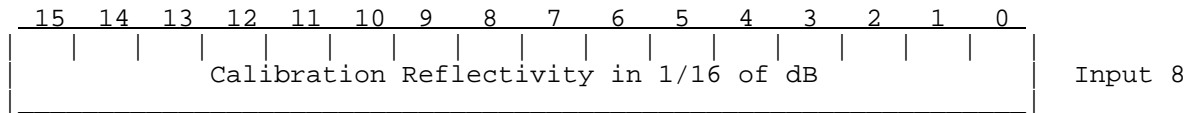
Each of the single-bit fields selects whether the given processing or threshold option is enabled (1) or disabled (0).

- 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).

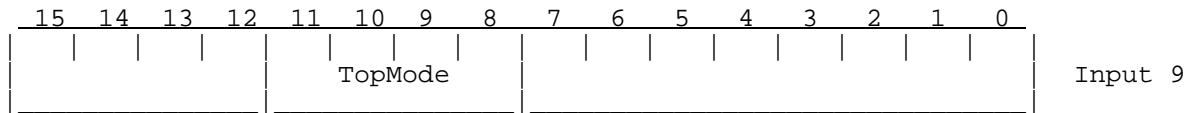
The Signal Quality Index (SQI) threshold is an unsigned binary fraction in the range 0 to 255/256. When the SQI for a range bin falls below the stated value it may result in thresholding of data.



Weather Signal Power (SIG) is an estimate of the SNR of the weather component of the received signal. When the SIG (see Section 5.2.10) falls below this comparison value it may result in thresholding of data.

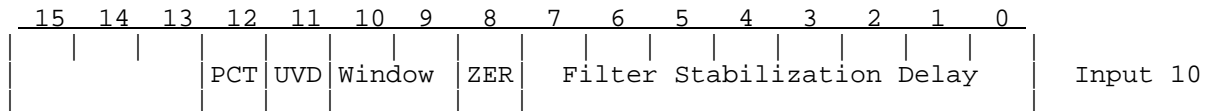


The calibration reflectivity is referenced to 1.0 kilometers.



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.

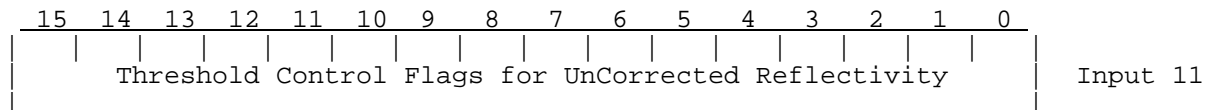
- 0000 Pulse Pair Processing Mode. Doppler clutter filters are 4th-order IIR high pass; data are processed one pulse at a time as each pulse arrives (see Section 5.2.3).
- 0001 FFT Processing Mode. Doppler clutter filters use nonlinear frequency-domain approach; data are processed in batches of pulses (see Section 5.8).
- 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 5.9).
- 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).



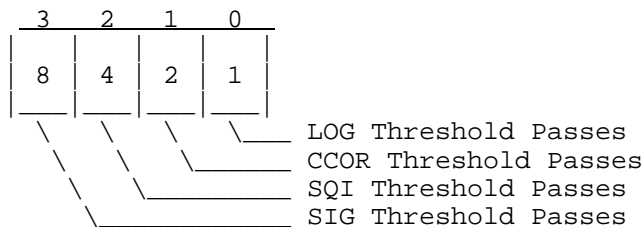
The RVP8 clutter filters are controlled by this word.

- 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.
- UVD** Unfold velocities using a simple ($V_{high} - V_{low}$) algorithm, rather than the standard algorithm described in Section 5.6.
- 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.



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:

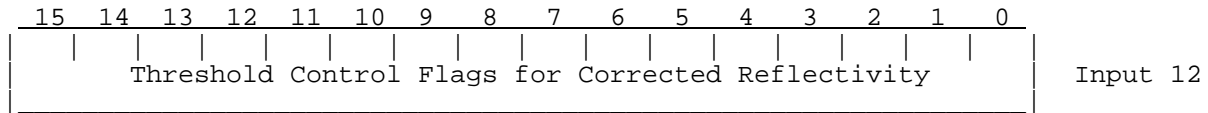
Value	Criteria
FFFF	All Pass (Thresholds disabled)
0000	All Fail (No data are passed)
AAAA	LOG
8888	LOG and CSR
A0A0	LOG and SQI
8080	LOG and CSR and SQI
F0F0	SQI
FAFA	SQI or LOG
C0C0	SQI and CSR
F000	SQI and SIG
C000	SQI and SIG and CSR
FFF0	SQI or SIG
CCC0	(SQI or SIG) and CSR

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.

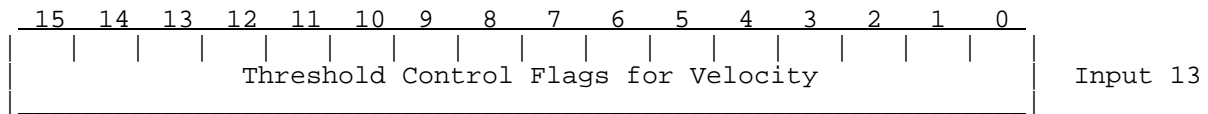
For example:

$$\begin{aligned}
 (\text{SQI or SIG}) \text{ and CSR} &= (\text{F0F0 or FF00}) \text{ and CCCC} \\
 &= (\text{FFF0}) \text{ and CCCC} \\
 &= \text{CCC0}
 \end{aligned}$$

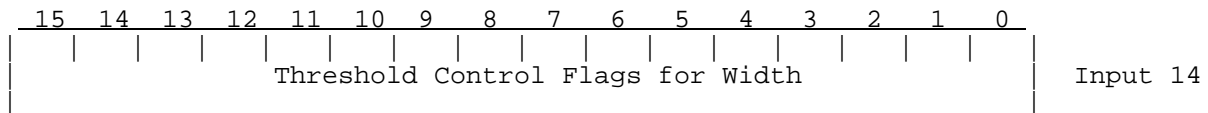
which corresponds with one of the examples given above.



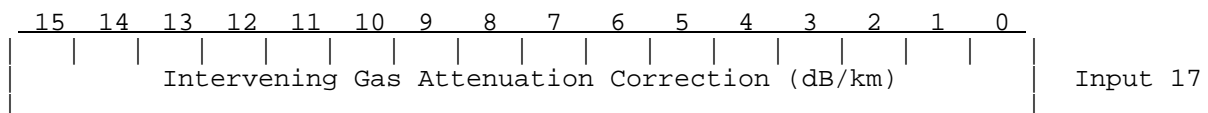
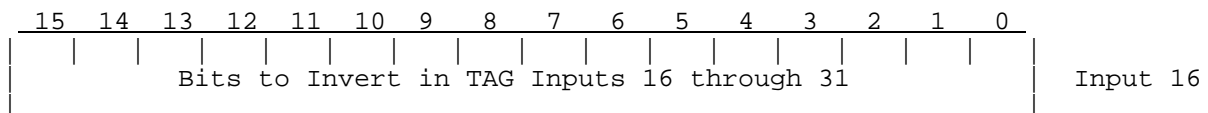
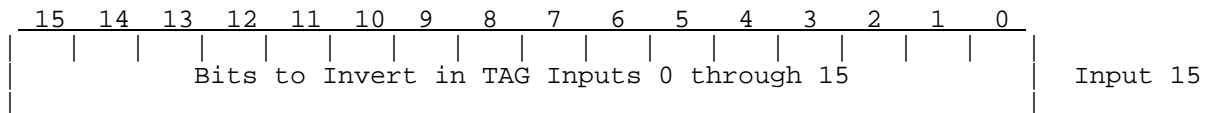
See Description for Input #11.



See Description for Input #11.



See Description for Input #11.

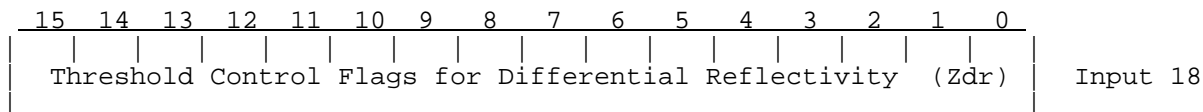


Gas attenuation correction attempts to compensate for overall (two-way) beam losses due to absorption by atmospheric gasses. The correction is linear with range, and is added to the data along with range normalization. Therefore, clearing the RNV bit in Word #2 above disables the correction. Of course,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

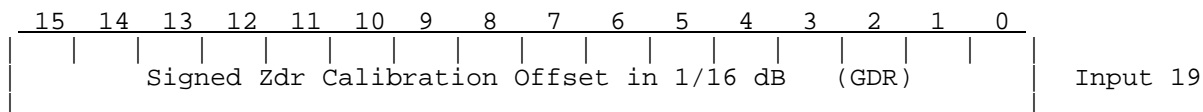
An attenuation of G db/km is encoded into the unsigned 16-bit word N as follows:

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

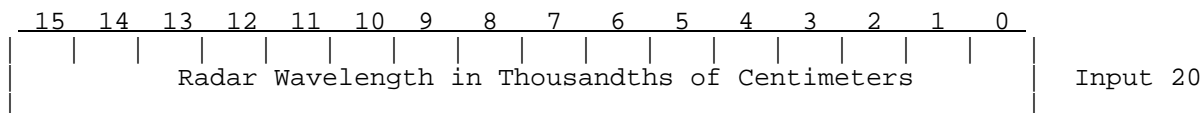
This format is backward compatible with the previous linear format for all values between 0.0 and 0.1dB/km; but it extends the upper range of values from 0.65535 up to 5.6535. These larger attenuation corrections are needed for very short wavelength radars.



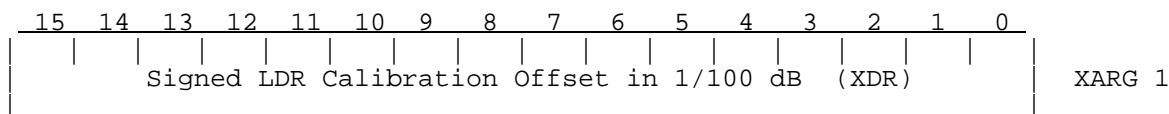
See Description for Input #11.



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.

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

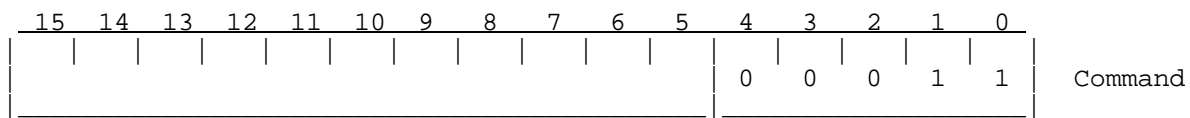
Parameter	Scientific Units	Input
Sample Size	25 pulses	25
Flag Word		0017 Hex
Log Slope	0.03 dB/LSB	1966
LOG Threshold	0.5 dB	8
CCOR Threshold	–25.0 dB	–400
Signal Quality Index Threshold	0.5 (dimensionless)	128
SIG Threshold	10.0 dB	160

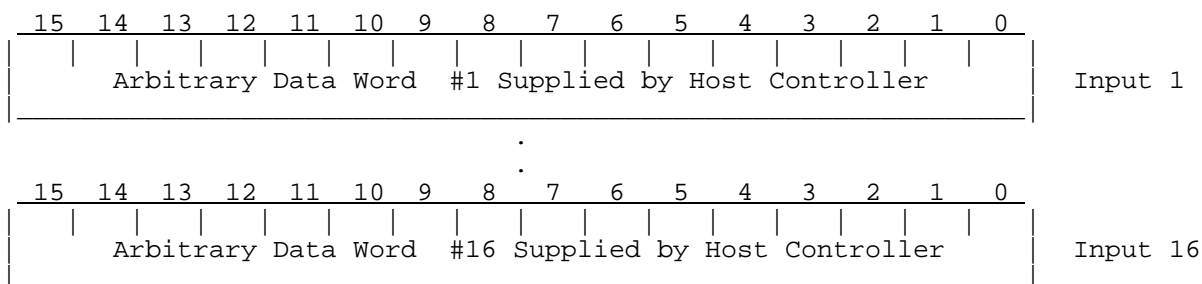
Table 6–1: Default Values For Operating Parameters (cont.)

Parameter	Scientific Units	Input
Calibration Reflectivity	–22.0 dBZ	–352
Gas Attenuation	0.016 dB/km	1600
Zdr Offset (GDR)	0.0 dB	0
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
TAG Bits to Invert	No Inversions	0000 Hex

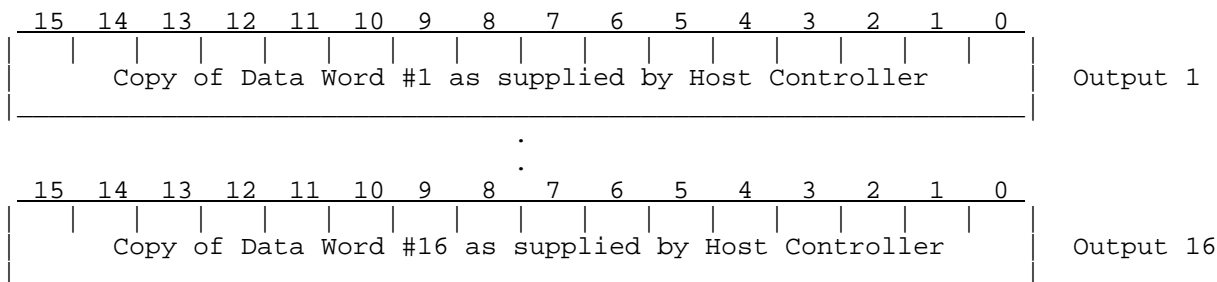
6.4 Interface Input/Output Test (IOTEST)

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.



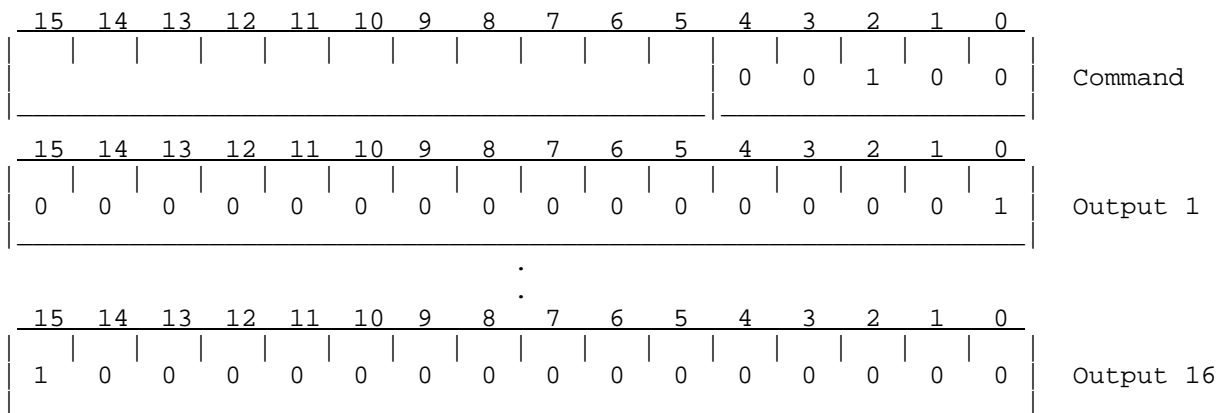


i **Note: The IOTEST command can also process and echo up to 128 additional XARGS data words (See Section 6.20).**



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 0.125 meters. The internal trigger generator is

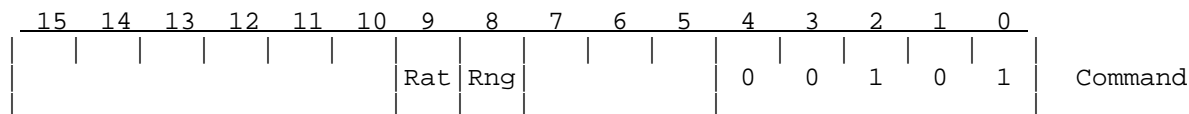
temporarily set to a special noise rate (usually much lower than the operating rate) during the process. It is ultimately the user's responsibility to insure that no returned power is present within the 32km sampling interval. In some cases it may be necessary to raise the antenna during the noise measurement 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.

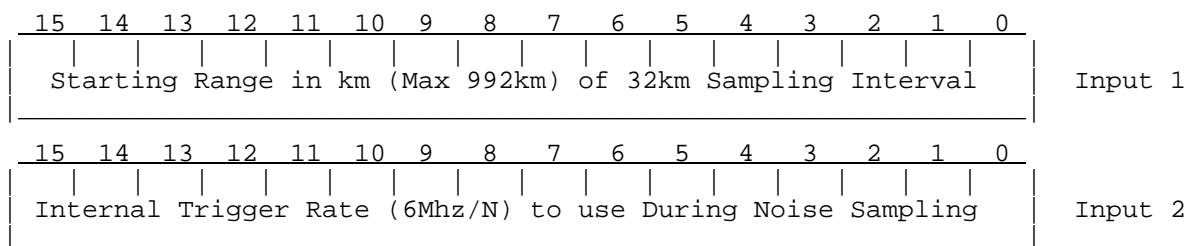
The SNOISE command bounds the requested starting range of the noise sampling interval. This is to insure that the noise samples will fit within the specified PRT, and within the range mask hardware RAM. The RVP8 sets an error bit when an improper range is requested. The noise sampling procedure also bounds the PRF to 1250Hz before making its measurements. This allows sufficient time for the algorithm to run properly on the 256 bins taken from 256 pulses. Note that the PRF bound is equally well imposed for external triggers too, i.e., external triggers will be ignored for 800µsec following each one that is actually used.

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. SNOISE must be executed at least once after power-up, before beginning to acquire and process data. 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.



- 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.



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 require 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

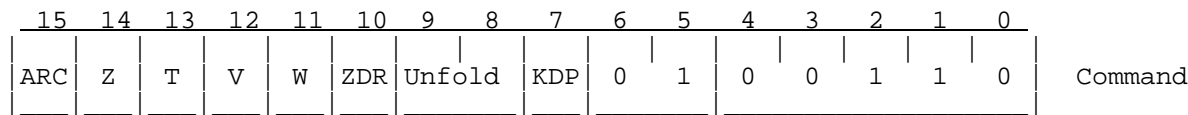
simultaneously, in which case the archive format is output first, followed by whichever individual display format values were also selected. The archive format is not recommended for use with new drivers because it can only handle four of the many possible output parameter types.

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).

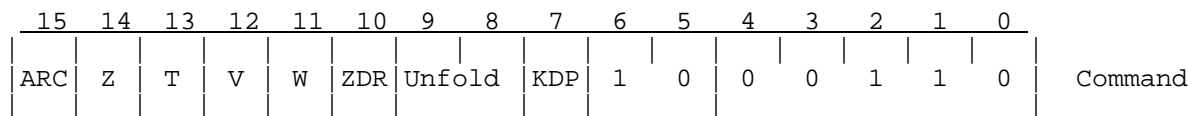
In addition to the above output data, the first words of each ray optionally contain additional information about the ray itself. These header words are configured by the CFGHDR opcode, and are included only if the NHD (No-Headers) bit in SOPRM Input #2 is clear. For example, if TAG angle headers are 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,

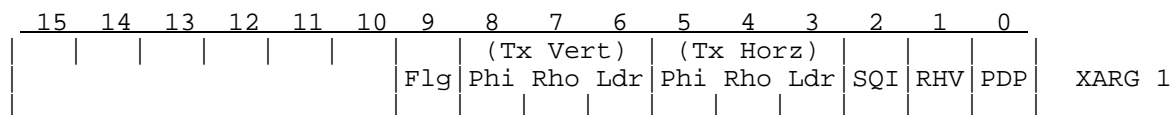
The Command word format for Synchronous Doppler Mode is:



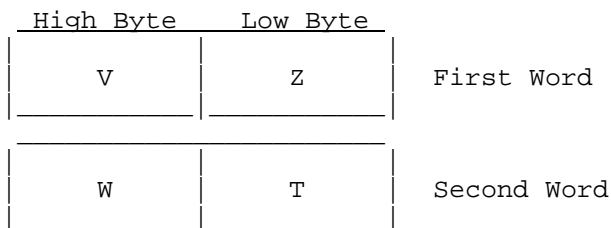
The Command word format for Free Running Doppler Mode is:



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



- 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:



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 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:

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

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:

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

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:

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

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 Φ_{DP} data.

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

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

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

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

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

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

RHV Selects dual polarization correlation coefficient ρ_{HV} data.

8-Bit ρ_{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 ρ_{HV} data available at this range
- 1 : 0.0000 (dimensionless)
- 2 : 0.0629
- 253 : 0.9980

- 254 : 1.0000
- 255 : Reserved Code

16-Bit ρ_{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 ρ_{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 (ρ_{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:

$$\text{dB} = -45.0 + (N-1) / 5$$

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

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

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

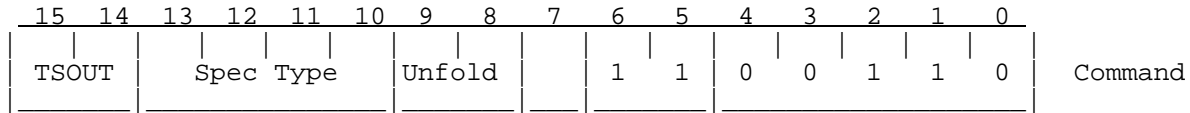
RHO Selects the cross channel correlation coefficient. This dimensionless parameter uses the same 8-bit and 16-bit data formats as RHV (ρ_{HV}).

PHI Selects the cross channel differential phase. This parameter uses the same 8-bit and 16-bit angular data formats as PDP (Φ_{DP}).

Flg Selects flag word output, bits defined as follows:

- 0 Range unfolding error
- 1 LOG threshold passed
- 2 CCOR threshold passed
- 3 SQI threshold passed
- 4 SIG threshold passed
- 5 Bin was speckle filtered

The Command word format for Time Series Mode is:



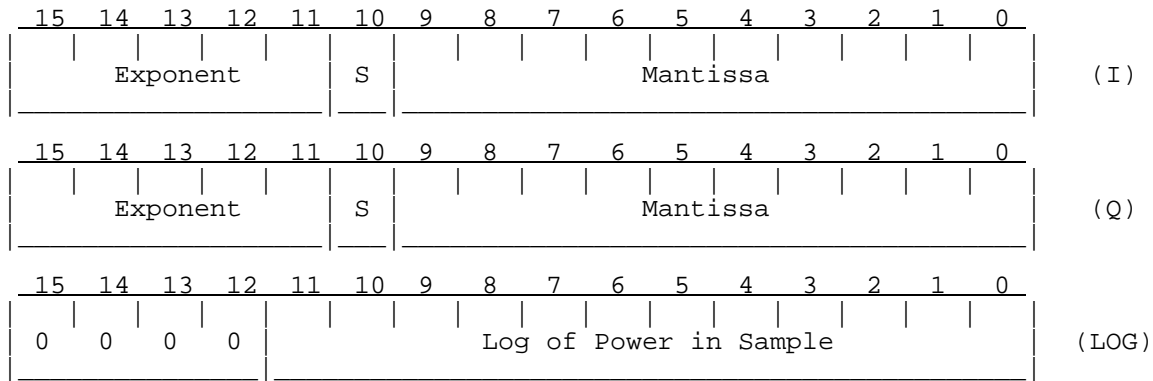
TSOUT Selects type of data to be output.
 00 : 8-bit Time Series 01 : Power Spectrum
 10 : 16-bit Time Series 11 : Unused

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

When time series output is selected the output data consist either of (3xBxN) or (2xBxN) 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 the floating I and Q samples to voltages: First create a 12-bit signed integer in which bits zero through 9 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^{*(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:

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

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.

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:

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

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

$$Slope = \text{“Log Power Slope” word 3 of SOPRM command. } 0.03 \text{ recommended.}$$

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 N2 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. 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.

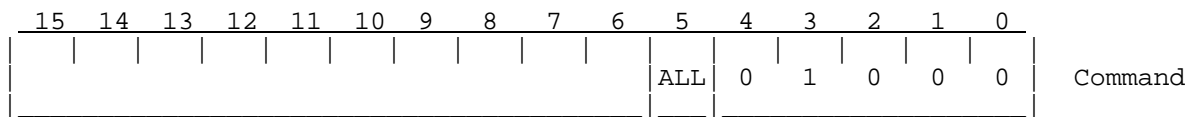
When the number of output words is large there is a possibility that the internal buffering within the RVP8 may overflow and data may be lost. Due to internal memory limitations, the product (BxN) must be less than 12000. A bit in the latched status word (See GPARM) indicates when time series overflows 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.

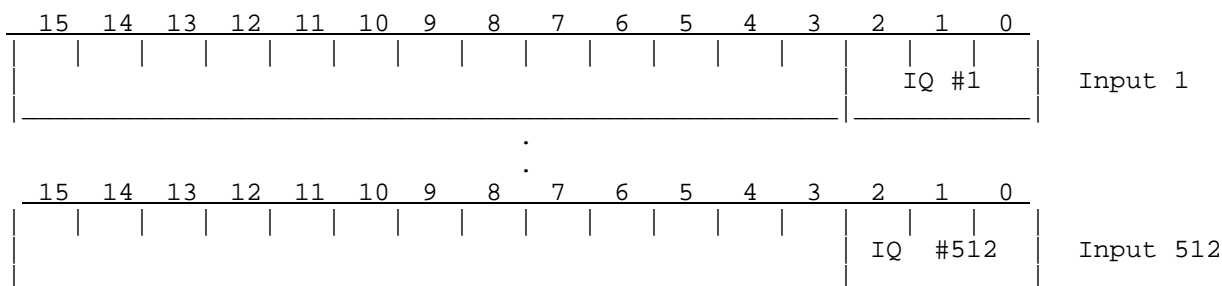
6.8 Load Clutter Filter Flags (LFILT)

A special feature of the RVP8 processor is that any of the eight 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.

The input words following the LFILT command specify the choice of filter to be applied at each of the (up to 2048) 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 be 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 appropriately 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. The correspondence between filter codes and filter characteristics is given in Appendix C. If the ALL bit is set in the Command, then 2048 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.





6.9 Get Processor Parameters (GPARM)

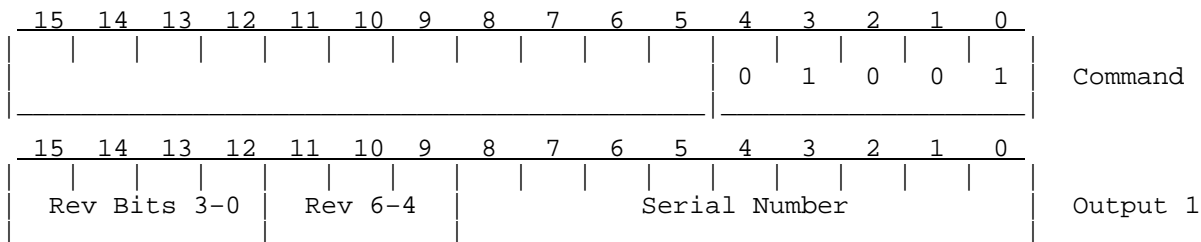
This command is used to access status information from the RVP8 processor. Sixty-four words are always transferred, some later words are reserved for future compatibility and are read as zeros. For convenience, a shorthand table of the output words is given in Table 6–2.

Table 6–2: RVP8 Status Output Words

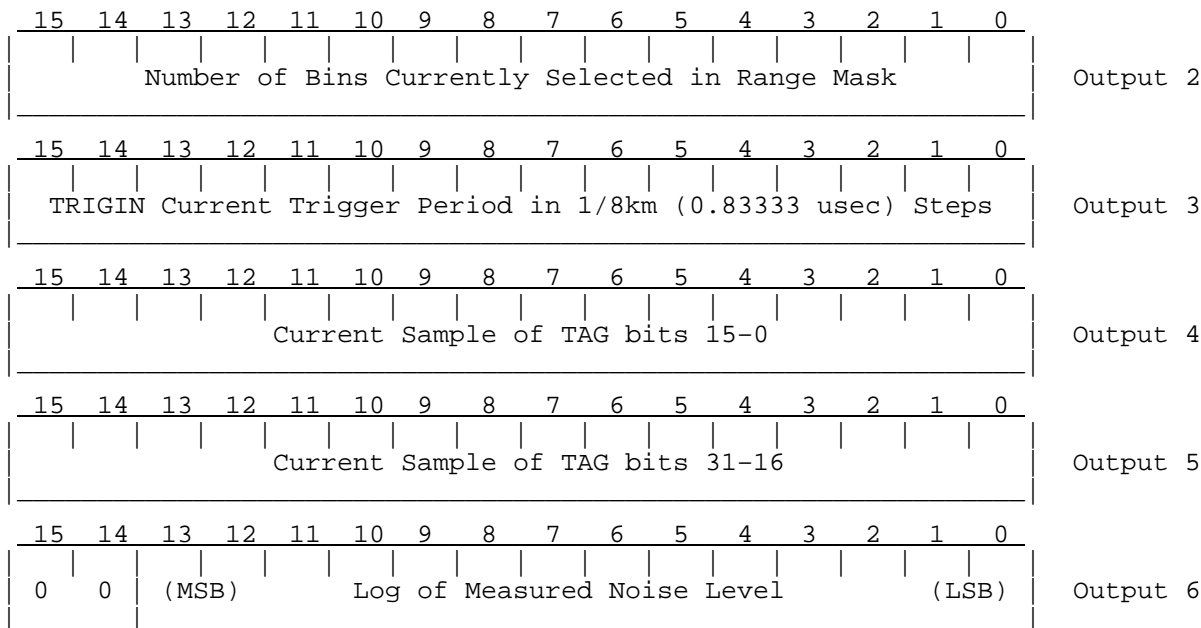
Word	Description	Word	Description
1	Revision / Serial number	33	LOG Threshold
2	Number of Range Bins	34	CCOR Threshold
3	Current trigger period	35	SQI threshold
4	Current TAG00 – TAG15	36	SIG Threshold for Width
5	Current TAG16 – TAG31	37	Calibration Reflectivity
6	Log of Measured Noise Level	38	— Reserved —
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-Exponential Noise Std. Dev.
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

Table 6–2: RVP8 Status Output Words (cont.)

Word	Description	Word	Description
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	— Reserved —
27	Current Trigger Gen. Period	59	— Reserved —
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 —

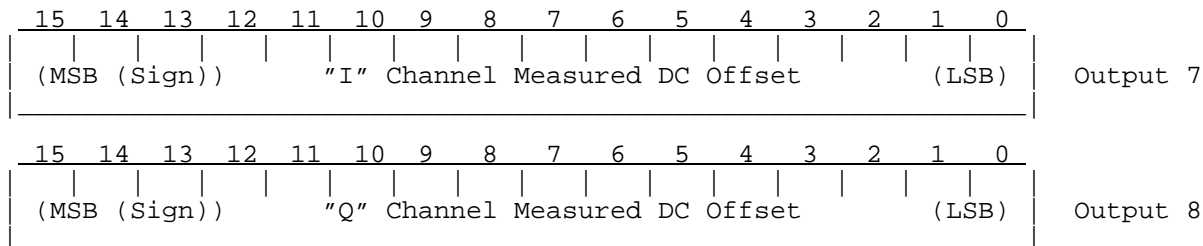


The revision and serial numbers of the particular RVP8 board are accessible here. This information is useful when computer software is being designed to handle a variety of signal processor revisions. The revision number is seven bits total; four of which are still in the high four bits of the word for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



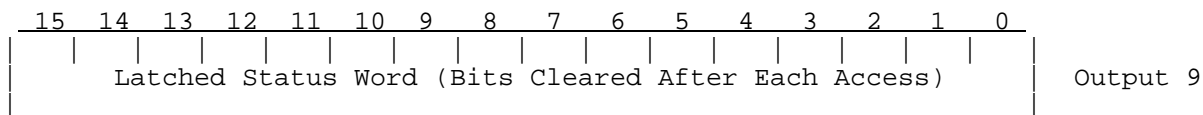
This value is scaled 4 times higher than the time series format. See the discussion in Section 6.7. To convert to dBm, use the equation:

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

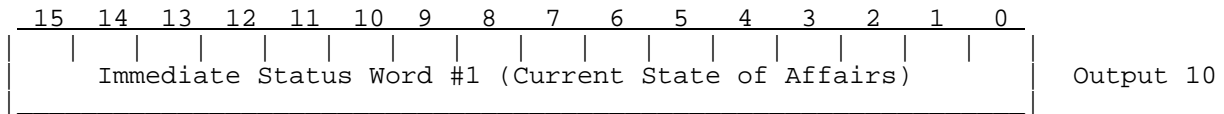


The "I" and "Q" values are stored in a linear format, similar to the short form time series. See the discussion in Section 6.7. These numbers should always be very close to zero, because the FIR filter that creates them is designed to reject DC. To convert to voltage, use the equation:

$$Voltage = V_{MAX} \times \left[\frac{Value}{32767} \right]$$

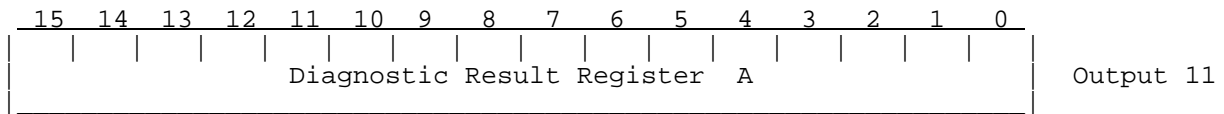


- Bit 0 No Trigger during noise measurement.
- Bit 1 Trigger too fast during noise measurement, i.e., some of the noise sample bins were positioned past the trigger range.
- Bit 2 No trigger during PROC command.
- Bit 3 PRF varied by more than 10 microseconds from the beginning to the end of a processing interval.
- 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



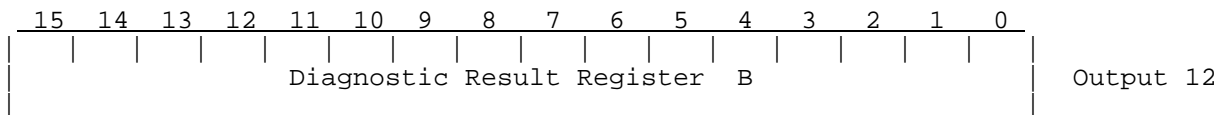
Output 10

- 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 AUX boards attached.
- Bit 14 DSP supports Power Spectrum output

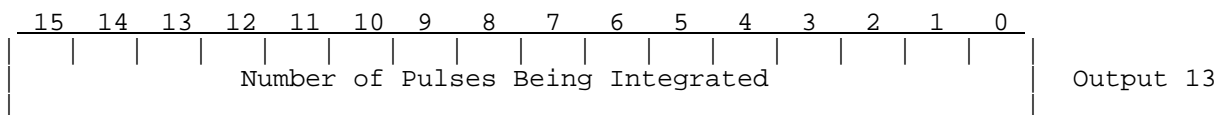


Output 11

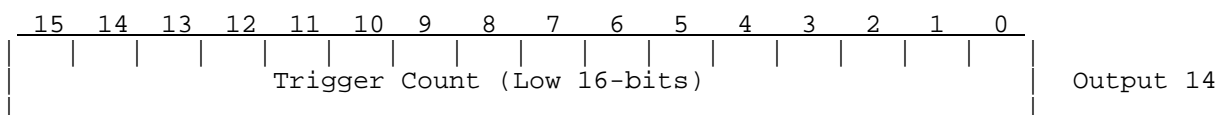
- Bit 0 RVP8/Rx card #1 failure
- Bit 1 RVP8/Rx card #2 failure
- Bit 2 RVP8/Tx card #1 failure
- Bit 3 RVP8/Tx card #2 failure
- Bit 4 IO62 card #1 failure
- Bit 5 IO62 card #2 failure
- Bit 6 Error loading config/setup files
- Bit 7 Error attaching to antenna library
- Bit 8 Problem when forking compute processes



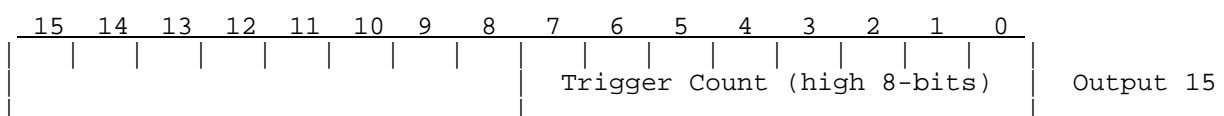
Output 12



Output 13

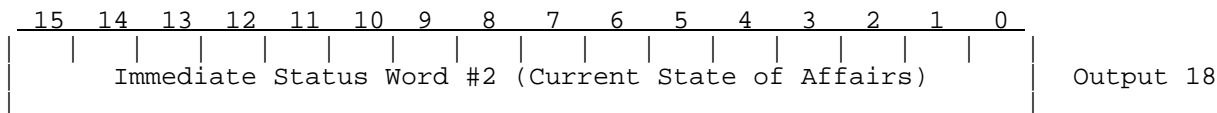
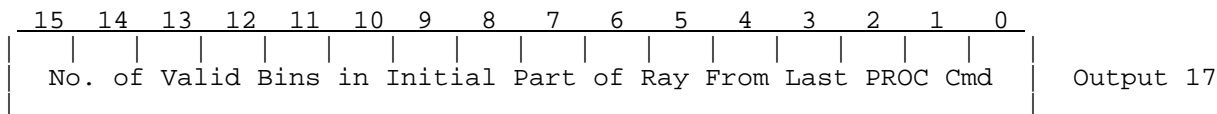
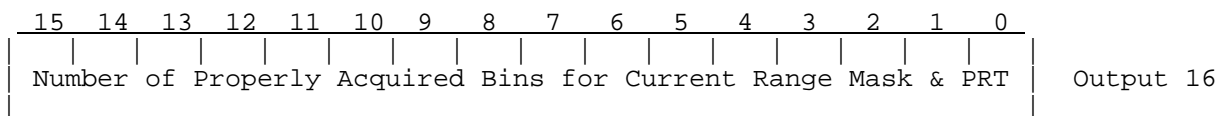


Output 14

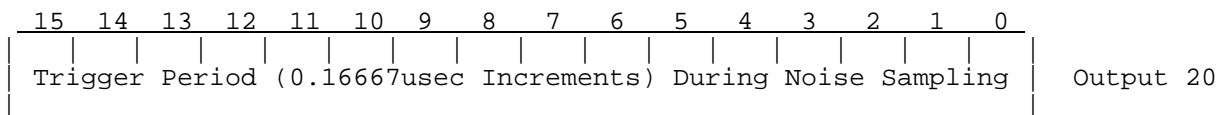
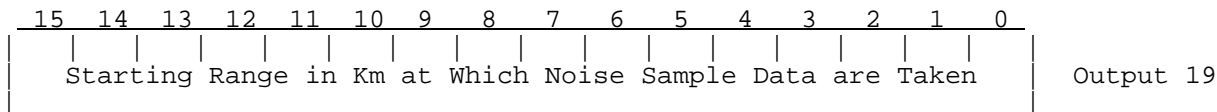


Output 15

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.



- Bit 0 Processor supports FFT algorithms
- Bit 1 Processor supports Random Phase algorithms
- Bit 3 Processor supports DPRT (Dual-PRT) algorithms
- Bit 4 Problem in UpLink COAX cable from RVP8/Main → RVP8/IFD
- Bit 5 Problem in DownLink fiber cable from RVP8/IFD → RVP8/Main
- 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 Burst Input
- Bit 14 Slave DSP count may be less than the number of available chips
- 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	Output 21
Min Trig Period (0.16667usec Increments) for Pulse Width 0																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 22
Min Trig Period (0.16667usec Increments) for Pulse Width 1																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 23
Min Trig Period (0.16667usec Increments) for Pulse Width 2																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 24
Min Trig Period (0.16667usec Increments) for Pulse Width 3																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 25
Four 4-bit Control Bit Patterns for Each Pulse Width																

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 26
						Window	TopMode						P.W.			

- P.W. Currently selected radar pulse width
- TopMode Major Mode (See SOPRM Input #9)
- Window Spectral Window Choice (See SOPRM Input #10)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 27
Current Trigger Generator Period (0.16667usec Increments)																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 28
Desired Trigger Generator Period (0.16667usec Increments)																

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.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 29
TRIGIN Period at Start of Last PROC Acquisition Time																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 30
TRIGIN Period at End of Last PROC Acquisition Time																

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.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 31
		Polar	NHD		16B	CMS	R2					End	Lsr	Dsr	Rnv	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 32
						Log Slope				65536 * dB / LSB						
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 33
						LOG Noise Threshold in 1/16 of dB										
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 34
						Clutter Correction (CCOR) Threshold in 1/16 of dB										
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 35
										SQI Threshold						
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 36
						SIG Threshold in 1/16 of dB										
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 37
						Calibration Reflectivity in 1/16 of dB										
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 38
						Reserved (Zero)										
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 39
						Reserved (Zero)										
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 40
										Range Avg (From LRMSK Command)						
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Output 41
						Reserved (Zero)										

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 Configuration of PROC data (Copy of CFGHDR Input #1)	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** ⁻¹⁶ LSB=2** ⁻³¹	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** ⁻¹⁵	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** ⁻¹⁶ LSB=2** ⁻³¹	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** ⁻¹⁵	Output 47

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

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

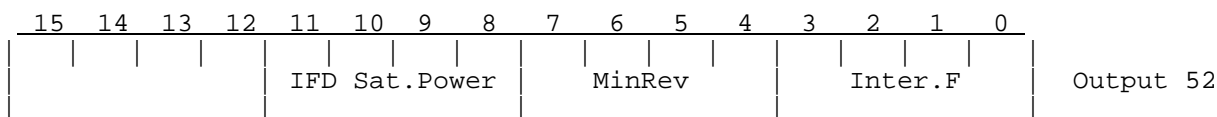
$$N_Q = (\text{Word 47}) \times 2^{-15} + (\text{Word 46}) \times 2^{-31}$$

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	Exponential-LOG Noise Standard Deviation (8000hex)	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	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



Inter.F Specifies which interference filter is running. Zero means “none”; see Section 5.1.4 for a description of the interference filter algorithms.

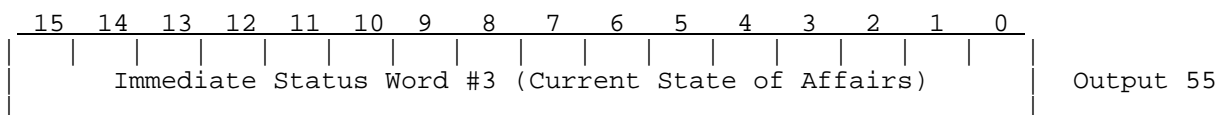
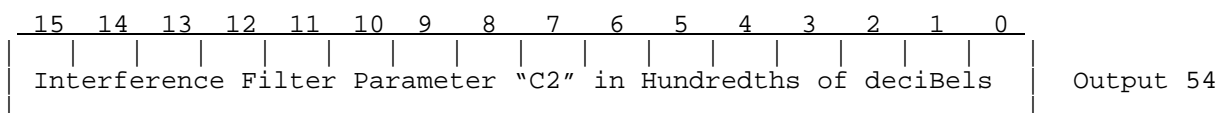
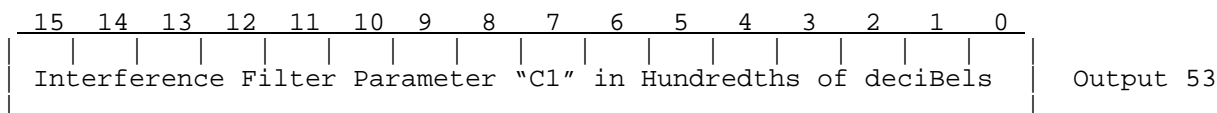
MinRev Minor revision level of the RVP8 code that is currently running.

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



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

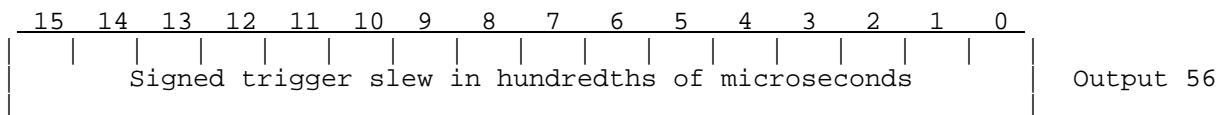
Bit 4 Last burst pulse hunt was unsuccessful

Bit 5 Dual PRT Type 2 is supported

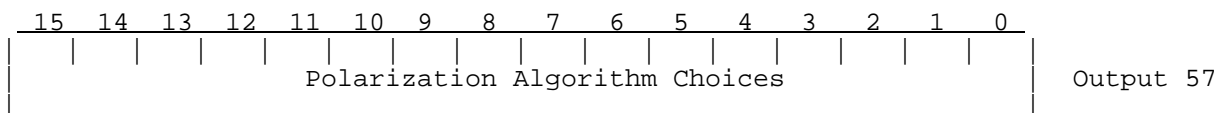
Bit 6 Could not generate the requested phase sequence

Bit 7 Problem with digital transmitter clock

Bits 8-11 User-defined Major Modes 1-4 are supported



This is the same format that is used by the SETSLEW command to set the current trigger slew (See Section 6.24)

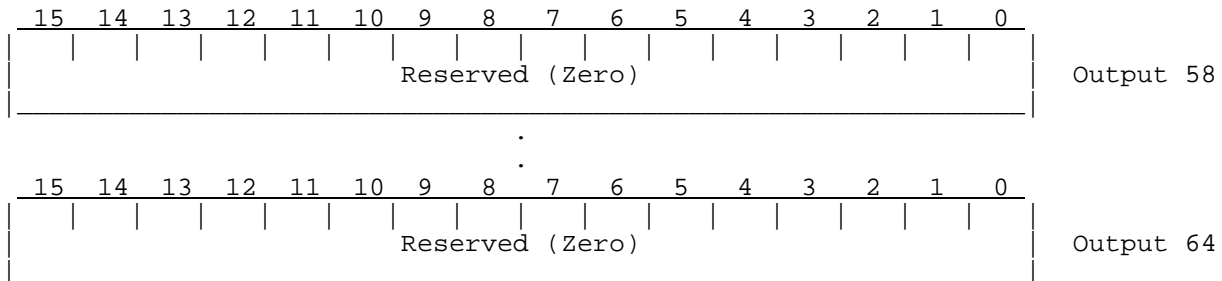


Bit 0 Use H transmissions for (T,Z,V,W)

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 params for noise
- Bit 5 Use filtered data for all polar params
- Bit 6 Sign convention for PHIdp

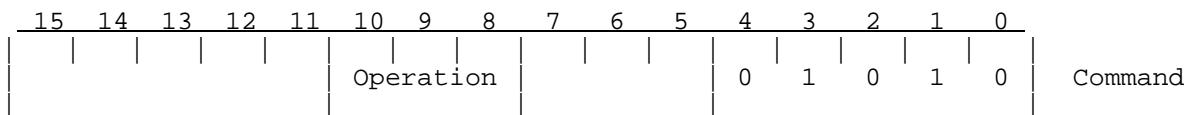


6.10 Load Simulated Time Series Data (LSIMUL)

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 ($N = \text{sample size}$) 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.

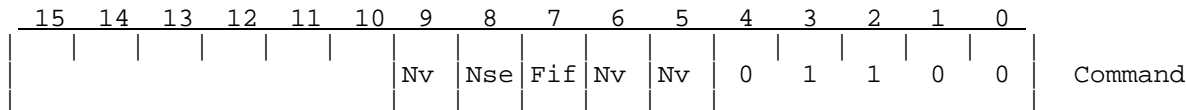


The available operations are:

- 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 ($N = \text{sample size}$) LSIMUL commands with Operation=2.

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 **Mt** setup questions (See Section 3.3.5).
- 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 *Introduction* to this chapter concerning actions taken by the RVP8 when the output FIFO becomes full.

6.12 Define Trigger Generator Output Waveforms (TRIGWF)

i

Note: This opcode is obsolete, and is included only for backward compatibility with the RVP6. The opcode is disabled by default (See Section 3.3.1), because the interactive trigger setup procedure described in Section 4.4 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.

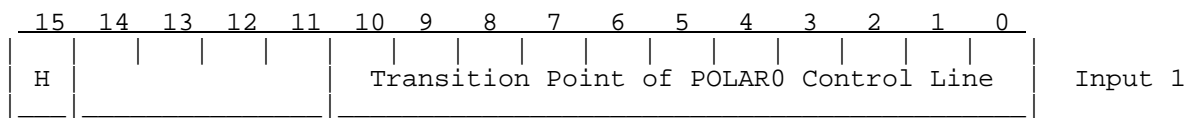
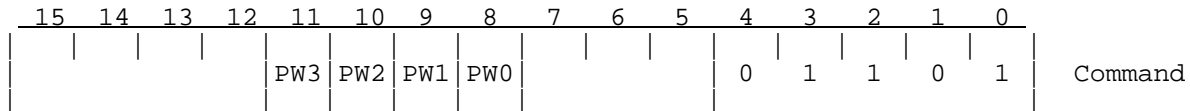
The RVP8 has a built-in trigger generator that can synthesize six independent digital output waveforms, each having arbitrary shape and being active anywhere in a window centered around zero-range. The six trigger outputs can be defined by a 2048-word by 6-bit table which is loaded from the user computer. 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.

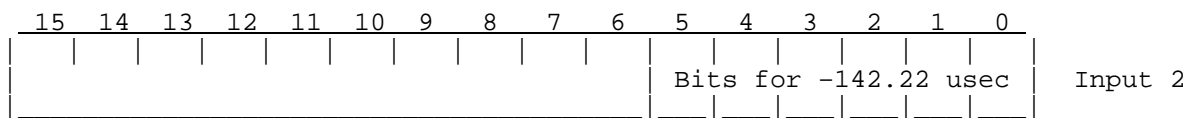
The first word following the TRIGWF command specifies the transition point of the POLAR0 polarization control signal. This control signal is either held low or high for the cases of fixed horizontal or vertical polarization, or it alternates from pulse to pulse for fast-switching polarization measurements 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.

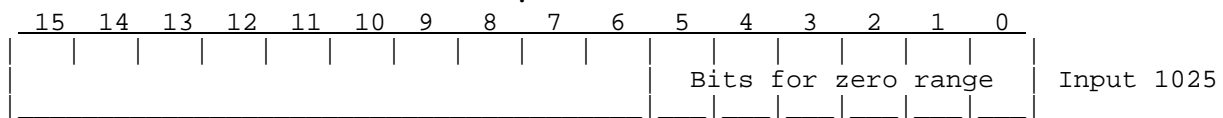
As an example, suppose we wish to make the TGEN0 output be a 0.42 microsecond pretrigger pulse, with a rising edge exactly five microseconds prior to range zero. This would be done by setting bit 0 in input words 988, 989, and 990 following the TRIGWF command, and leaving all other bit 0's clear. Further,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



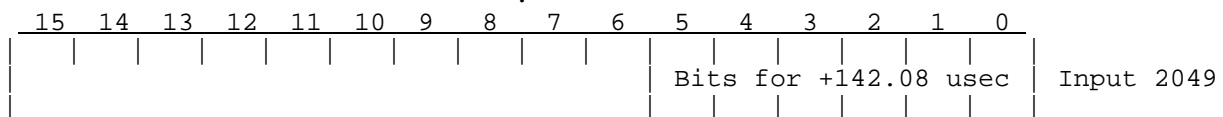
H This bit defines the sense of the control line when horizontal polarization is selected.



⋮



⋮



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

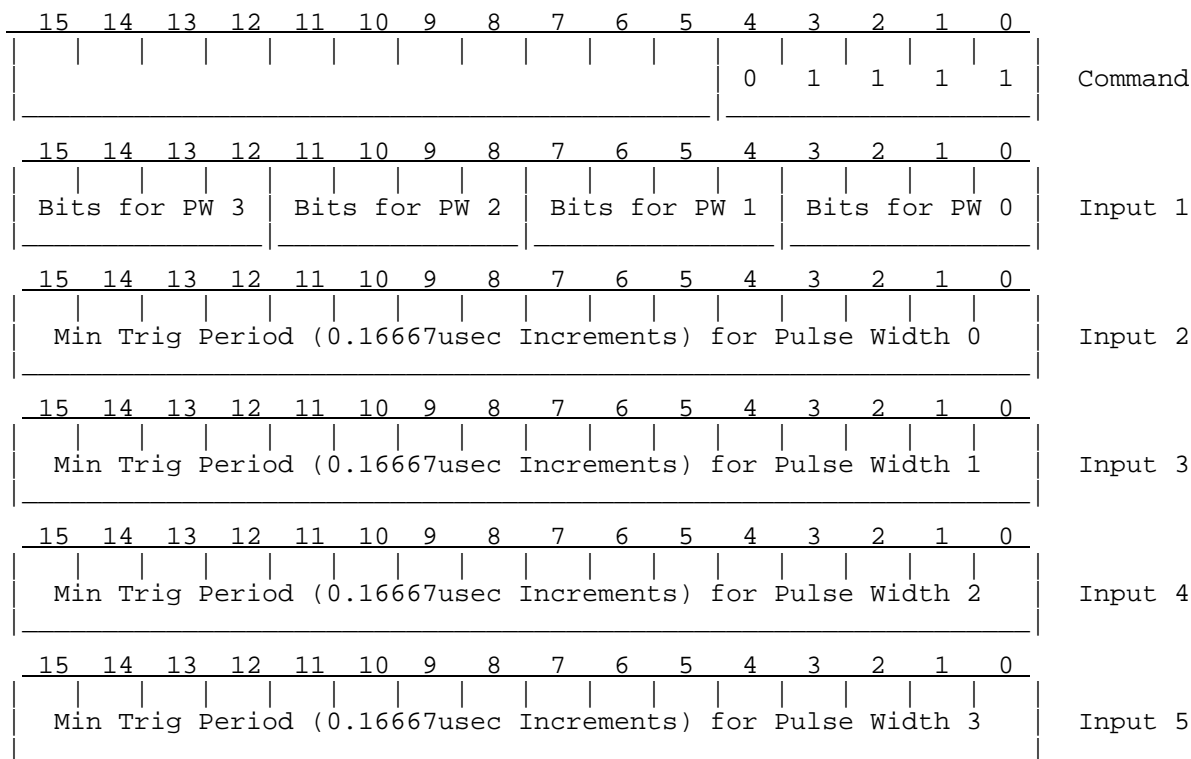
The RVP8 is equipped to control the radar transmitter's pulse width and corresponding receiver bandwidth. There are four pulse/bandwidth codes, numbered simply 0 through 3. The association between codes and pulse widths is completely determined by the needs and capabilities of the particular radar on 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.

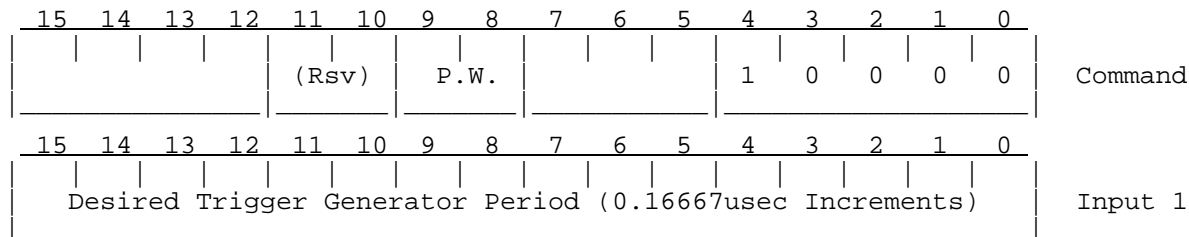
For each pulse width there is a corresponding minimum trigger PRT permitted. This bound is intended to limit the transmitter duty cycle to a safe value under all conditions. PWINFO sets up these minimum PRT's using words 2 through 5 following the command. The maximum frequency of the internal trigger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

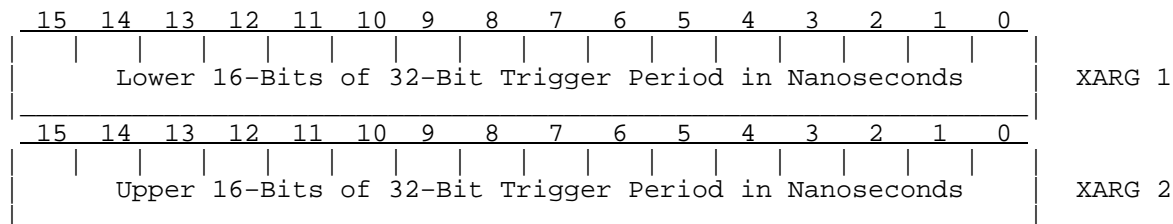


6.14 Set Pulse Width and PRF (SETPWF)

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.



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.

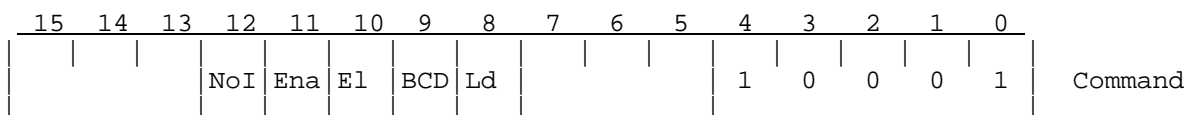
The synchronization algorithm works automatically with either clockwise or counterclockwise antenna rotation, and can tolerate any sequence of changes in direction, e.g., if the antenna itself is scanning a sector, or if it is turning erratically. Moreover, the trigger angles do not have to be hit exactly 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.



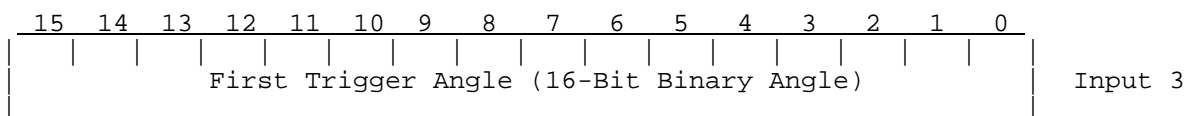
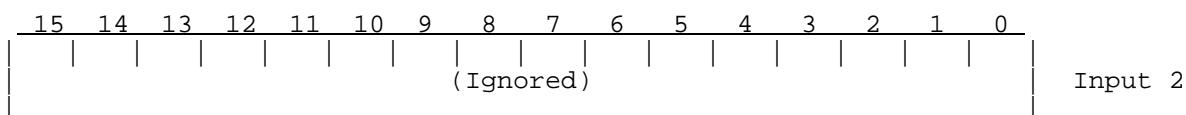
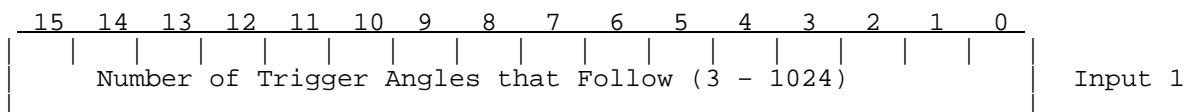
NoI Ordinarily, the potentially lengthy sync wait loop is terminated if the user writes additional words to the RVP8. Setting this bit prevents such interrupts. Beware that the processor loop can not be broken in this case except by moving the antenna across a trigger angle, or cycling the RESET_ line.

Ena Enables antenna synchronization.

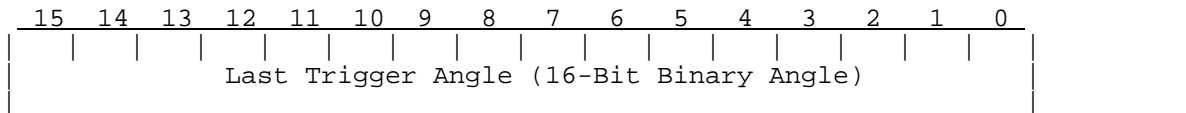
E1 Synchronization is based on TAG15–31 (Elevation) inputs, else TAG0–15 (Azimuth) is used.

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.

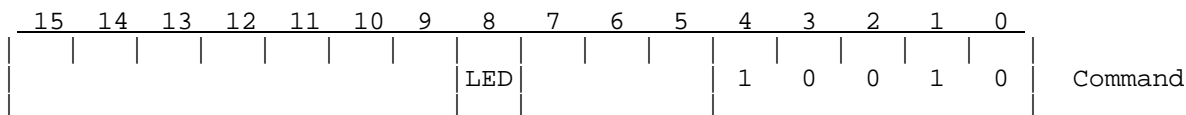


.



6.16 Set/Clear User LED (SLED)

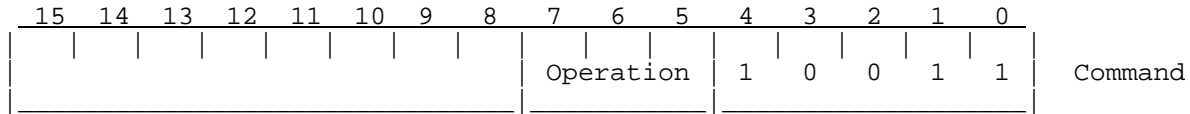
This command simply turns the red user LED on and off under program control. The LED is on during the initial running of internal diagnostics, and then remains off unless changed by this command. Note that the red LED can be configured to serve as an internal activity indicator (see TTY setups), in which 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.

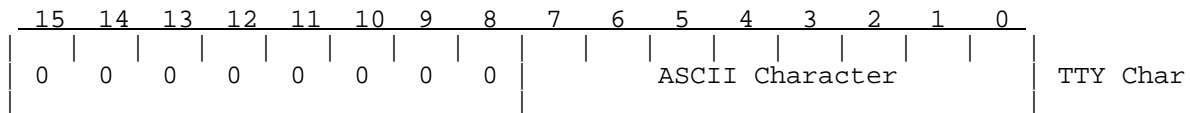


The operation codes are as follows:

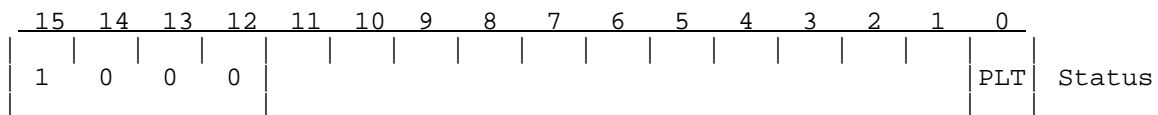
- 0 Sends the ASCII character in the upper byte of the word to the RVP8 as if it had been typed on the setup TTY's keyboard.
- 1 Allow scope plotting data to be output whenever a plot is being drawn. All relevant status and data words are output once upon each receipt of this command. Subsequently, status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



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



- 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.

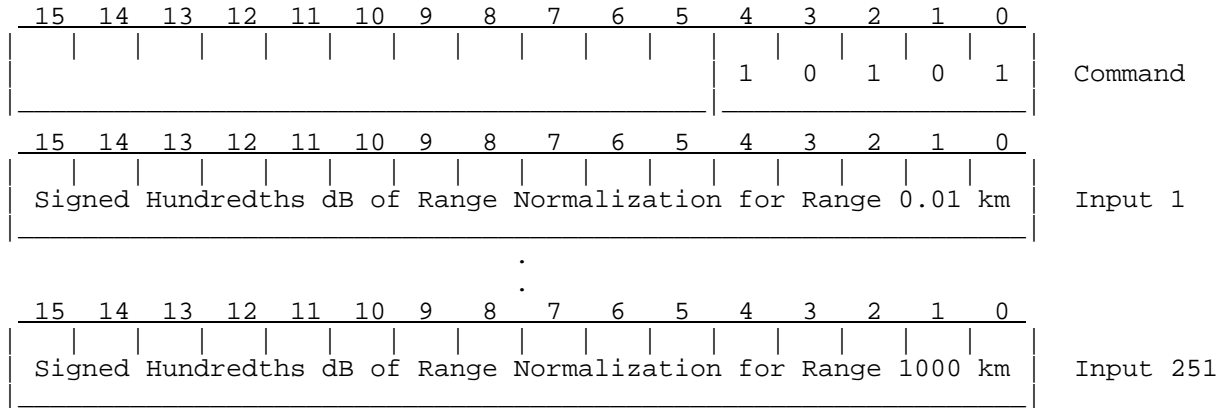
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	1	0											Stroke Number	Plot Data
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	1	1												Value to Plot (0 - 4095)
															Word #1
															.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	1	1												Value to Plot (0 - 4095)
															Word #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.

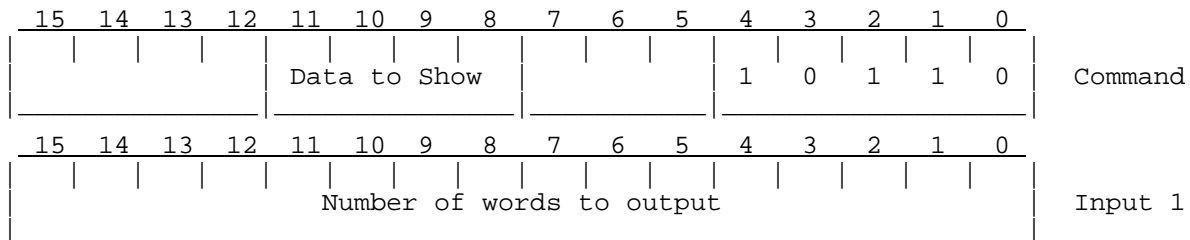
LDRNV loads a 251-word custom correction table holding values in hundredths of deciBels over five decades of $\log(\text{range})$ from 0.01km to 1000km. There are 50 table entries per decade of range. Thus, the range in kilometers corresponding to an input word #N is $10^{\lfloor \frac{N-1}{50} - 2 \rfloor}$, 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.



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.
- 4 Filter selection array from the last LFILT command.
- 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:

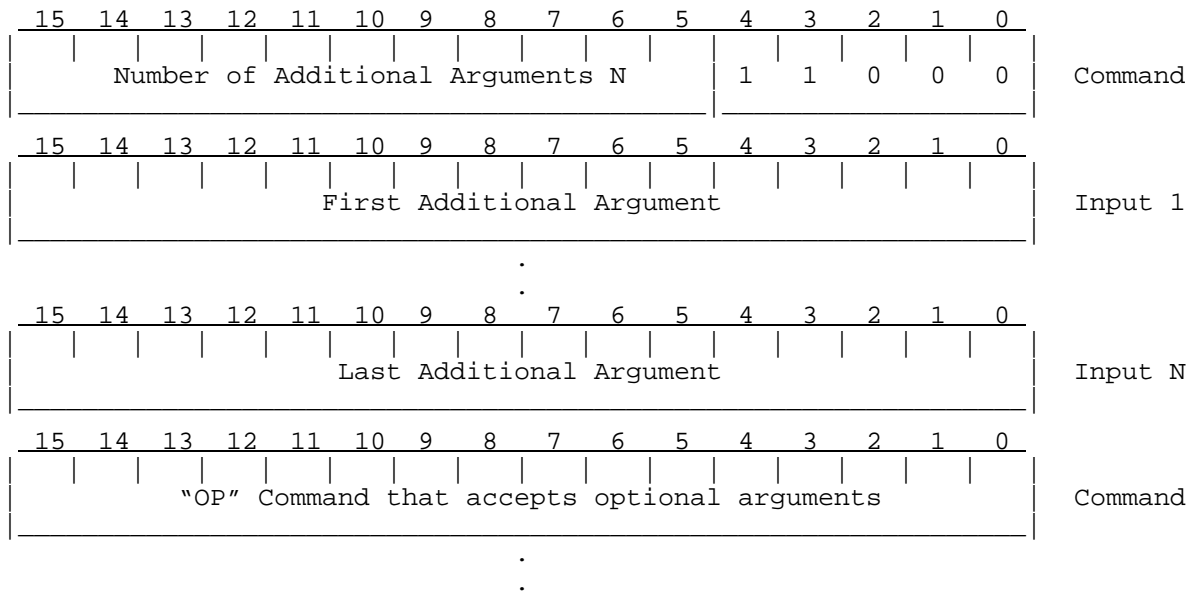
- a. Azimuth (TAG bits 0 – 15)
- b. Elevation (TAG bits 16 – 31)
- 8 Doppler clutter filter coefficients (Same format as for LFCOEFS command)
- 10 Range mask spacing in cm for each pulsewidth

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.

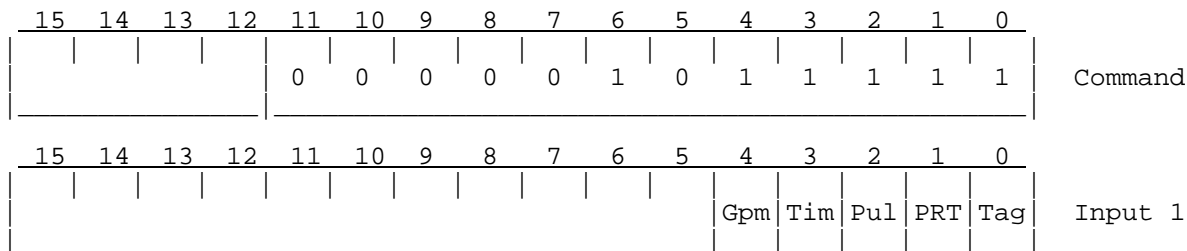
XARGS affects only the opcode that immediately follows it. The entire list of optional arguments is discarded after "OP" executes, even if "OP" did not use some or all of the list. However, if "OP" is yet another XARGS command, then the additional arguments that it supplies will be appended to the first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



6.21 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.



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

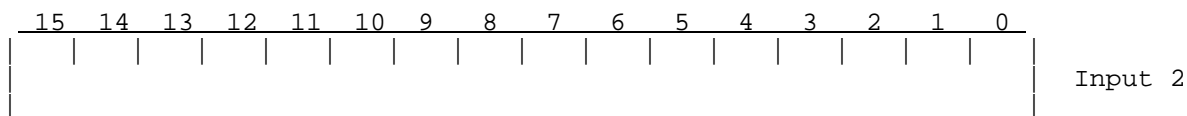
- When the RVP8 is operating in dual PRF mode, bit zero of the “start” TAG word is replaced with a flag indicating that the ray’s PRF was low (0) or high (1).
- When trigger blanking is enabled, bit zero of the “end” TAG word is replaced with a flag indicating that the trigger was blanked (0) or normal (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 Time stamp. Sixteen-bit counter incrementing at a rate of 1000 counts/sec, and sampled at the end of the ray.

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



6.22 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	1
																Command

Filter Chooses which interference algorithm should be run. See Section 5.1.4 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

6.23 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	
																Command
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
16-Bit AFC/MFC Value (-32768 through +32767)																Input 1

6.24 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 -327.68µsec to +327.67µsec. The actual span permitted by the RVP8 is ±20µsec. 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	Command
				0	0	0	0	1	1	0	1	1	1	1	1	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Input 1
Signed trigger slew in hundredths of microseconds																

6.25 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	Command
			Now	0	0	0	0	1	1	1	1	1	1	1	1	

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.

6.26 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 3.3.8.

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

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.

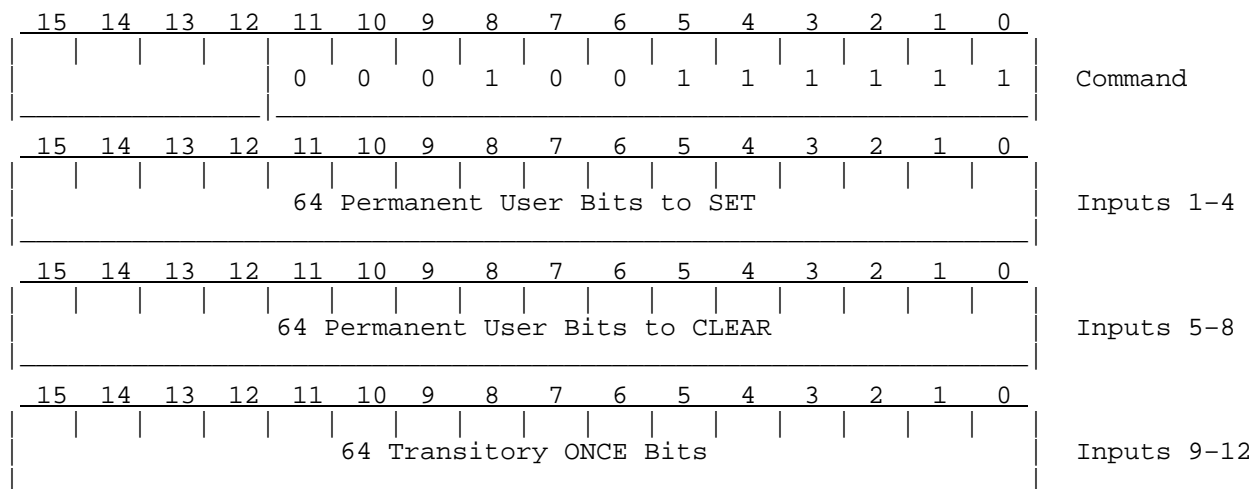
- PhSeq=2 Selects a *User Defined* sequence. If no XARGS have been supplied, then the RVP8 outputs the default idle phase that is defined in **Mz**. If XARGS are supplied, then they are interpreted as a sequence of 16-bit binary angles. The RVP8 will make the best match between each desired angle and 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.

6.27 Set User IQ Bits (UIQBITS)

Load user-specified bits that will be included with the pulse headers in the live (I,Q) data stream. The current permanent bits are stored right in the shared (I,Q) data segment, and a FIFO history is also 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.

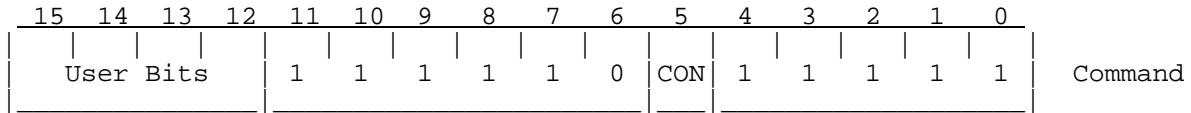
The permanent Set/Clr bits are updated in the signal processor and retain their value from the last time they were defined. The bits are repeated into all pulse headers. The ONCE bits, however, are transitory and will appear in only one pulse header each time they are set.

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.



6.28 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.

