RDA 8.08.14 Release Notes (27 May 2005)

There were no RDA changes in this release.

RDA 8.08.13 Release Notes (23 May 2005)

These release notes cover changes made to the SIGMET Radar Data Acquisition platform, including primarily the RVP8 and RCP8 products. The last public release was RDA-8.08.11 dated 2 May 2005. If you are upgrading from an earlier version please also read the release notes that have been published since then.

New Features

1. The Point Clutter Filter is now applied first to the low-PRF reflectivity data in BATCH mode prior to using those data to unfold the high-PRF Doppler bins. This prevents point targets from being smeared during the range averaging step that reduces the variance of the low-PRF values. Note: the Point Clutter Filter is still applied to the final unfolded data ray just as it has been so far. Thus, the algorithm is applied twice in BATCH mode; first to the raw low-PRF reflectivities, and then a second time to the final output ray. The `DBFLG_POINTCLUT` bit will be set in the output parameter flags wherever point clutter was detected on either application.
2. In the **RCP8** changed the “site custom” configuration question “ Velocity multipler for Elevation” to read “Time lag for Azimuth”, and the limits are now in the range 0 to 2 seconds.

RDA 8.08.12 Release Notes (12 May 2005)

There were no RDA changes in this release.

RDA 8.08.11 Release Notes (2 May 2005)

These release notes cover changes made to the SIGMET Radar Data Acquisition platform, including primarily the RVP8 and RCP8 products. The last public release was RDA-8.08.10 dated 15 April 2005. If you are upgrading from an earlier version please also read the release notes that have been published since then.

Bug Repairs

1. The RCP8 TTY Setups “Site Host” command was ignoring the specified interface. Instead it was always using “eth0”.

RDA 8.08.10 Release Notes (15 April 2005)

These release notes cover changes made to the SIGMET Radar Data Acquisition platform, including primarily the RVP8 and RCP8 products. The last public release was RDA-8.08.9 dated 12 April 2005. If you are upgrading from an earlier version please also read the release notes that have been published since then.

New Features

1. The RVP8/Tx card can now synthesize pulsed waveforms on both of its output channels at the same time. Both transmit waveforms will have the same shape, but can be delayed by different amounts via the **Mt<n>** menu:

```
Tx Waveform
  Zero offset of Tx pulse – ChA: 0.00 usec, ChB: 0.00 usec
```

   Having two Tx output waveforms can be handy for testing, as the second one can serve as a delayed transmit pulse. This is especially useful for complex compressed waveforms that would be difficult to shift by tens of microseconds using conventional microwave delay devices.

   Note: The delayed Tx waveform is not intended for making transmitter phase stability measurements. Each channel is synthesized independently; hence one will not be an absolutely precisely delayed version of the other.

2. The **speed** utility now has a *–logRays* flag that causes it to print a full log of all rays that were acquired. Previously this was only done when the *–angsyn* flag was used.

3. The RVP8 can now generate parallel Tx phase control lines from the I/O-62 card as well as from the standard backpanel. Add the following stanza to your *softplane.conf* :

```
# Parallel Tx phase control outputs.  Options are:
#         I/O:4–11 : Eight IO62 RS–422 outputs #4–11.  These are
#                    wired to J3 on the standard backpanel.
#
splConfig.Io62[0].Opt.Cp.TxPhaseLines = ”I/O:4–11”
```

   This will result in eight RS-422 linepairs #4–11 being driven from the I/O-62 card as Tx phase bits #0–7. These same linepairs are available on the backpanel J3 connector, for which pins 1/14 are TxPhase-0, 2/15 are TxPhase-1, etc.

4. The maximum length of the WSR-88D Receiver Protect Trigger has been shortened so that it will never extend more than 1.8μsec (previously 7.1μsec) past the end of RF-Gate Trigger. This maximum time will be reached whenever the Cathode Pulse is missing.

Bug Repairs

1. The RVP8's algorithm for finding the next Coherent Processing Interval (CPI) would sometimes return retrograde CPIs, especially if the timeseries data stream was stopped during **tsarchive** playback.

RDA 8.08.9 Release Notes (12 Apr 2005)

These release notes cover changes made to the SIGMET Radar Data Acquisition platform, including primarily the RVP8 and RCP8 products. The last public release was RDA-8.08.8 dated 7 April 2005. If you are upgrading from an earlier version please also read the release notes that have been published since then.

New Features

1. The RVP8 now internally supports up to sixteen different pulsewidth definitions. The additional slots can be used by third party software for radars that require more than four different pulsewidths. This increase applies to the RVP8 only; IRIS and its supporting utilities still support just four.

   Changes were made to the SETPWF and PWINFO opcodes, as well as the **Mt** menu in which the available pulsewidth count can be changed by typing an additional number after setting the pulsewidth in the usual manner. All changes are backward compatible, and existing driver code will continue to run properly if you choose to ignore this new feature.

RDA 8.08.8 Release Notes (7 Apr 2005)

These release notes cover changes made to the SIGMET Radar Data Acquisition platform, including primarily the RVP8 and RCP8 products. The last public release was RDA-8.08.7 dated 22 March 2005. If you are upgrading from an earlier version please also read the release notes that have been published since then.

New Features

1. Two additional selections are now supported within RVP8 ray headers: the timeseries playback version number, and the current TASKID information.

2. A new DSP driver entry point *dspw_LOGModifiers()* allows you to set the breakpoints and slopes that will modify the LOG threshold according to the Clutter-to-Noise ratio of the target. This makes the LOG threshold behave properly even as the noise floor becomes elevated due to very strong clutter targets. The default signal processor settings will be restored if a NULL arg is supplied.

3. We now have routines to find the Kth smallest element of an array using Niklaus Wirth's algorithm (for which *median* becomes merely a special case). You may enjoy taking a look at this elegant little algorithm in *libs/user/lowmath.c*; despite it's simplicity, it is one of the most efficient methods known.

4. Improvements have been made to the Wide Dynamic Range Mode (#4) to handle Lo/Hi channel phase differences more completely. There will always be some phase offset due to different components, different lengths of cable, etc., and the RVP8 now redesigns its FIR filters to include fractional tap delays that equalize the group delay of each channel. The nominal gain and phase offsets measured from the **Pr** plot should still be typed into the **Mp** setup menu exactly as before, and the output from *–showRxMode* is unchanged. What's new is that the fractional tap FIRs now permit the two channels to be spliced together properly even in regions of rapid target fluctuation.

5. The feature added in RDA-8.08.7 (strong saturating clutter being able to cause the LOG threshold to fail) is now configured via the OPPRM `iFlagsTS` word, rather than from the **Mf** menu. This allows the DSP driver to selectively enable the feature.

Bug Repairs

1. During RVP8 powerup, the Rx/IFD jitter diagnostic will now be skipped on fibre-style IFDs that are running with a VCXO (Voltage Controlled Crystal Oscillator). This is because the test is likely to fail with an open loop VCXO and cause a spurious RVP8 powerup failure to be reported. The jitter test is still run at all times when the tests are invoked via the **rdadiags** utility.

2. The noise levels reported by the GPARM opcode were always coming from pulsewidth zero. This bug was introduced in the 8.08.6 release.

3. An intermittent segmentation fault has (very likely) been repaired in both the RVP8 and RCP8, resulting from their use of the *mlockall()* system call to lock memory during real time operations. The symptoms of failure suggested some sort of memory management bug might be present, and a web search revealed several other users having `SIGSEGV` problems in other applications using locked pages. In an RDA system with plenty of physical memory there is no particular need to lock the RVP8 and RCP8 pages, so the solution for now is simply to leave them unlocked (normal user memory). We would like to thank the ORDA programming team for their help in tracking down this bug.

4. When an IO62 PCI card was powered up with active external signals attached to its first four digital I/O lines, there was a chance that a spurious IO62CP backpanel might have been reported attached to the card.

RDA 8.08.7 Release Notes (22 Mar 2005)

These release notes cover changes made to the SIGMET Radar Data Acquisition platform, including primarily the RVP8 and RCP8 products. The last public release was RDA-8.08.6 dated 12 March 2005. If you are upgrading from an earlier version please also read the release notes that have been published since then.

New Features

1. The LOG threshold test in the RVP8 can now be forced to fail whenever a strong clutter target has driven the IFD into saturation at a particular bin. This is generally the correct thing to do since the spectral noise induced by saturated clutter would dominate virtually any weather echo that might also happen to be there. The **Mf** menu question:

   ```
   Saturated clutter targets cause LOG threshold to fail: YES
   ```

   controls whether the LOG threshold is modified in this manner.

   i **Note:This setup question was removed in RDA-8.08.8 in favor of setting via the DSP driver.**

2. Added DspGetFullTime() to the dsp library. This allows an application to sample the time on the RVP8.

Bug Repairs

1. The *Fault Status Pin* selected in the RVP8 **Mb** menu was not being saved properly when the Active-Low option was chosen.
2. Starting in release 8.08.5 the *DspSetTaskId()* library call was forcing the RVP8 out of continuous output mode.

Setup Changes

1. In setup/RCP there is a new question in the control bits section. *Pulse Width Unchanged Control* lets you select the use of that bit in the control packet. If you set this to disabled, you can select whether you wish the bit high or low.

RDA 8.08.6 Release Notes (12 Mar 2005)

These release notes cover changes made to the SIGMET Radar Data Acquisition platform, including primarily the RVP8 and RCP8 products. The last public release was RDA-8.08.5 dated 8 March 2005. If you are upgrading from an earlier version please also read the release notes that have been published since then.

New Features

1. During timeseries (I,Q) data playback, the RVP8 can optionally get its noise levels and calibration reflectivity from the values that were originally recorded with those data. A new DSP driver entry point *dspw_tsOpts()* allows you to choose when this is done.

2. The residual clutter LOG noise margin is now computed from the Clutter-to-Noise ratio rather than from the Clutter Correction (CCOR). This more correctly captures the reason that this adjustment is needed, i.e., to prevent the elevated noise accompanying a strong clutter target from being misinterpreted as residual weather signal.

```
Residual clutter LOG noise margin:
  Baseline : 0.15 dB/dB for Clutter/Noise above 10dB
  HiSignal : 1.00 dB/dB for Clutter/Noise above 50dB
```

   The above setup questions in the **Mf** section have been reworded to reflect the change. The *Baseline* region compensates for worsened statistics of residual weather after any type of clutter filtering is applied, and *HiSignal* compensates for hard upper bounds on window & SNR performance. The *Baseline* correction slope applies from its starting point until the *HiSignal* starting point, after which the *HiSignal* slope is used. The LOG correction is continuous at that transition point, with only the slope changing. For the above settings, some example adjustments would be:

```
Clutter/Noise    Adjustment
-------------    ----------
     0 dB            0 dB
    10 dB            0 dB
    30 dB            3 dB
    50 dB            6 dB
    51 dB            7 dB
    52 dB            8 dB
             etc.
```

RDA 8.08.5 Release Notes (8 Mar 2005)

These release notes cover changes made to the SIGMET Radar Data Acquisition platform, including primarily the RVP8 and RCP8 products. The last public release was RDA-8.08.4 dated 22 February 2005. If you are upgrading from an earlier version please also read the release notes that have been published since then.

Setup Changes

1. In the **setup** RVP button, in the *Real Time Display* section there is a new option. Leave "System transmits RTD" to "No" or "From App" for legacy behavior. Setting it to "From RVP8" will cause the RVP8 to generate a real-time display stream locally. If using this feature be sure to also push the License button and set the RVP8's site name and site code. The site code will appear on the rtdisp display.

New Features

1. The RVP8 now invokes site-specific bootup and shutdown code. You may define the routines *rvp8MainSiteInit()* and *rvp8MainSiteExit()* for the main threads, along with *rvp8ProcSiteInit()* and *rvp8ProcSiteExit()* for the compute processes. The latter may need to check their slot/unit number *iProc_c* so that each of several processes can do exactly what's needed.

2. The RVP8 ***View*** command can now be invoked with a "p" suffix to request a special printout showing the current values of processing and thresholding parameters. So far the command lists the threshold settings for all data parameters. This can be handy for checking that your software is really loading up these values properly.

```
RVP8> Vp
Threshold Settings for All Data Parameters
           LOG      CSR      WSP      SQI      TCF (Equation)
         -------  -------  -------  -------  ------------------
    DBZ:  0.81dB  -25.0dB    8.0dB   0.398   0x8888 ( LOG & CSR )
    DBT:  1.00dB  -25.0dB    5.0dB   0.398   0xAAAA ( LOG )
    VEL:  0.81dB  -30.0dB    5.0dB   0.398   0xC0C0 ( SQI & CSR )
    WID:  0.81dB  -35.0dB    5.0dB   0.398   0xA000 ( SIG & SQI & LOG )
    ZDR:  0.81dB  -20.1dB    5.0dB   0.398   0xAAAA ( LOG )
    KDP:  0.81dB  -20.1dB    5.0dB   0.398   0xFFFF ( All Pass )
  PHIDP:  0.81dB  -20.1dB    5.0dB   0.398   0xFFFF ( All Pass )
  RHOHV:  0.81dB  -20.1dB    5.0dB   0.398   0xFFFF ( All Pass )
    SQI:  0.81dB  -20.1dB    5.0dB   0.398   0xFFFF ( All Pass )
   LDRH:  0.81dB  -20.1dB    5.0dB   0.398   0xFFFF ( All Pass )
   RHOH:  0.81dB  -20.1dB    5.0dB   0.398   0xFFFF ( All Pass )
   PHIH:  0.81dB  -20.1dB    5.0dB   0.398   0xFFFF ( All Pass )
   LDRV:  0.81dB  -20.1dB    5.0dB   0.398   0xFFFF ( All Pass )
   RHOV:  0.81dB  -20.1dB    5.0dB   0.398   0xFFFF ( All Pass )
   PHIV:  0.81dB  -20.1dB    5.0dB   0.398   0xFFFF ( All Pass )
```

3. The RVP8's TASKID opcode now includes the scan geometry (one of SCAN_xxx), as does the DSP library routine *DspWriteTaskId()* .

4. The One-Dimensional speckle remover is now implemented in the RVP8. This feature, which had not yet been ported from the RVP7, removes single bins that do not have any adjoining neighbors in range. The speckle remover is controlled by the OPF_DSR and OPF_LSR bits in the SOPRM flag word.

Bug Repairs

1. The elevation position limit shutdown had been disabled for non-encoder angle inputs since 19 Nov 2004.

2. The RVP8 correction for 2-way waveguide length (from the **Mt** menu) is now working. This feature had not yet been ported from the original RVP7 code.

3. The Boot-&-Go CDROM feature now will create the */usr/iris_data* directories with mode 775 instead of 755. This is needed on systems with more than one authorized operator account.

RDA 8.08.4 Release Notes (22 Feb 2005)

These release notes cover changes made to the SIGMET Radar Data Acquisition platform, including primarily the RVP8 and RCP8 products. The last public release was RDA-8.08.3 dated 6 February 2005. If you are upgrading from an earlier version please also read the release notes that have been published since then.

New Features

1. The *LOG Threshold Level* in the ascope utility can now be set as large as 12dB, rather than its previous upper limit of 5dB.
2. The dsp_lib has a new function lDspHasHeaderCfg(). This reports which ray header features are available.

Bug Repairs

1. The RVP8 would sometimes produce the *Trigger Waveform was Altered* signal when the trigger quantization in the **Mt** menu was other than (1,0).
2. The RVP8 is now able to process single-bin timeseries data during *TSArchive* playback in which the original number of recorded bins was greater than one. Previously, a bin count mismatch was being reported. This special case allows the **ascope** utility to plot spectra from any type of recorded data.
3. **Ascope** now reads the actual data time from the RVP8, and displays that on the top line. This means that for time series playback the times recorded with the data are displayed. For RVP7 and RVP6 processors, the time label remains the current time on the application machine.

RDA 8.08.3 Release Notes (6 Feb 2005)

These release notes cover changes made to the SIGMET Radar Data Acquisition platform, including primarily the RVP8 and RCP8 products. The last public release was RDA-8.08.2 dated 24 January 2005. If you are upgrading from an earlier version please also read the release notes that have been published since then.

New Features

1. The RVP8 and RCP8 now keep track of the hardware patch level of the standard IO62CP backpanel board, the new 72MHz IFD, and all PCI cards. This type of tracking had been partially implemented already, but is now fully supported. Each board's etch version (Rev.A, Rev.B, etc.) now has an associated patch level telling which factory ECOs have been installed. Initially, an unmodified board is at patch level "1", after which the patch level will be incremented whenever an approved ECO is applied at SIGMET.

   The patch level of all boards is displayed along with its etch version separated with a colon. For example, the RVP8's startup board discovery text might show:

```
Attaching to SIGMET PCI hardware...
  Found PCI Card RVP8/Rx – Rev.C:1  Serial:2318  Code:6
   \––> Remote IFD Assy  – Rev.F:2  Serial:2490  Code:2
  Found PCI Card RVP8/Tx – Rev.D:1  Serial:2332  Code:13
  Found PCI Card I/O–62  – Rev.B:1  Serial:1841  Code:25
   \––> IO62CP Backpanel – Rev.B:1  Serial:1822  Code:3
```

   from which we learn that the IFD is a Rev.F board that has had a single ECO applied, while all other boards are electrically unaltered.

   Important: A new V4.3 kernel module is required to support this change. Please use "rdasys –stop" and "rdasys –start" as root to install the new version, or just reboot after the RDA installation. If you're running with a custom kernel, then please rebuild your kernel module manually from source.

2. An electrical change has been made to the 72MHz Rev.F IFD board bringing it up to Patch Level 2. The change remedies an interference problem on the secondary IF input port that was causing its noise level to be approximately 2dB higher than the primary input port. The only RVP8 receiver mode that is affected by this is #3 (Dual-Pol on a single IFD). The Wide Dynamic Range Mode (#4) is not affected because data from the secondary IF input (LoGain channel) are never used in that operating region. The RVP8 now prints a startup warning if one attempts to run Mode #3 on a Rev.F IFD without the patch applied.

3. The **ascope** utility can now activate point clutter detection and censoring. A new button labeled *PntClut* in the *Filters* menu is either blank (no filtering), or shows "1", "2", or "3" according to the separation of the two side bins from the center bin. A value of "1" means that the side bins are adjacent to (touching) the center bin; "2" means that the side bins are 2-bins away, i.e., one bin is skipped between the center and side bins, etc. The value in the button is one larger than the *iSideSkip* parameter that is passed to the DSP Driver's *dspw_pointClutter()* routine.

Whenever point clutter filtering is enabled, **ascope** provides the power threshold for the central point from the *CSR* value of the *Other* parameter in the *Thresholds* menu. This is a bit of a hack, but allows you to experiment with changing the point clutter threshold in a way that probably would not interfere with anything else.

4. The DSP Driver *dspw_pointClutter()* routine would sometimes accidently reset some of the SOPRM threshold parameters. This could cause unexpected behavior depending on the order in which other library routines happened to be called.

5. The comment text in *softplane.conf* documenting the hookup of relays from the PCI card was changed from:

   `# Relay2 – NO:J11,3 Arm:J11,4 NC:J11,2` *(incorrect)*

   to:

   `# Relay2 – NO:J11,3 Arm:J11,2 NC:J11,4` *(correct)*

6. A missing RVP8 feature is now implemented, in which the LOG threshold comparison level will automatically increase according to the amount of clutter removed from each bin. This feature had never been fully ported from the RVP7. The setup question in the **Mf** section:

   `Residual clutter LOG noise margin: 0.15 x (CCOR – 10dB)`

   defines how the LOG threshold will be modified at each bin, given the clutter correction *CCOR*. The offset permits no change to be made until *CCOR* exceeds a minimum value, and the multiplier then converts the excess *CCOR* (in dB) to an increase in the LOG test level (also in dB). Note that the LOG threshold test level can only be made larger (more strict) by this adjustment; negative dB values from the above equation are clipped at zero.

Bug Repairs

1. A race condition was repaired in the serial communications with the IO62-CP standard backpanel. This may have been responsible for intermittent panel lockups that could only be remedied by cycling power (rebooting was not sufficient).

[i] **Note: Half of this repair will be installed when the backpanel is *rdaflash*'ed to V4. However, the complete repair requires reprogramming the backpanel to bring it to Patch Level 2, which can only be done at SIGMET. The latter step should only be done if you continue having backpanel crashes after installing the new V4 code. Very few sites should need this.**

RDA 8.08.2 Release Notes (24 Jan 2005)

These release notes cover changes made to the SIGMET Radar Data Acquisition platform, including primarily the RVP8 and RCP8 products. The last public release was RDA-8.08.1 dated 16 January 2005. If you are upgrading from an earlier version please also read the release notes that have been published since then.

New Features

1. Several new sections have been added to the *RVP8 Developer's Notes* (Appendix E) of the *RVP8 User's Manual*. Details are now available for creating custom major modes from prior ones (E4), and the realtime callback mechanism is documented (E5) along with two detailed examples (E5.2 and E5.3). Improvements were also made to the introductory sections E1 and E2.
2. The demonstration code for realtime callback histogramming has been cleaned up and moved entirely to the *rvp8main/site* directory as *demohist.c* and *demohist.h*.
3. A new demonstration major mode that provides micro-staggered trigger pulses is now available in the *rvp8main/site* directory as *demostag.c* and *demostag.h*. It is largely a clone of the factory FFT mode, but with finely staggered PRTs that whiten the Doppler signature of multiple trip targets. Explainatory text is in Appendix E11.

RDA 8.08.1 Release Notes (16 Jan 2005)

These release notes cover changes made to the SIGMET Radar Data Acquisition platform, including primarily the RVP8 and RCP8 products. The last public release was RDA-8.08 dated 3 January 2005. If you are upgrading from an earlier version please also read the release notes that have been published since then.

Important Upgrade Notes

1. The dsp library was enhanced to serve out the contents of the **zcalib.conf** file. This means that on systems using **DspExport** to run applications on a different computer from the RVP8, the calibration file on the RVP8 is used. Previously it was the file on the application's computer (there might be several such computers). After upgrading in such an environment be sure to synchronize the calibration files by copying the **zcalib.conf** file from the application computer to the RVP8. In support of this, **zcal** was changed to not allow changing the calibration without I/O access to the DSP. Also the config library functions zcal_load() and zcal_save() were removed. Programmers who have written their own version of the dsp library should know that there are additional socket commands added to support reading and writing this info.

New Features

1. The *rvp8PulseInfo* structure contains the following new fields. Although these do not affect the timeseries directly, they are useful bits of adjunct information.

   - Calibration reflectivity dBZ_0, number of pulses per ray, and playback version number.
   - Flag indicating that angle sync'ing was enabled, and the mean angular spacing between rays.

   The **tsarchive** utility was enhanced to allow the user to set the playback version number on playback.

2. In support of the *rvp8PulseInfo* enhancements above, the **tsarchive** program now has a control in playback mode to specify the playback version number. The playback version number of data originating from the RVP8 is always set to zero. **Tsview** was enhanced to display all the new header information.

3. The RVP8's CFGHDR (Configure Ray Header) opcode can now request a 32-bit count of IFD system clock ticks at the start of each ray. This high speed counter (36MHz or 72MHz, depending on IFD version) is useful for obtaining very accurate ray timing.

4. The pulse levels specified in the **Mz** menu for the RVP8/Tx card can be expressed either in peak power or average power: *Output power level: 0.0 dBm, Peak:YES* . Choosing "Peak" will produce waveforms whose maximum amplitude is independent of amplitude shaping of the edges, whereas "Average" will attempt to preserve total energy within the

pulse by making the center portion stronger whenever the edges are tapered. The "Peak" choice is generally most appropriate for driving exciters and upconverters. Previously, the RVP8 always provided the "Average" method.

5. You can specify the number of decimal digits to show for AZ/EL angles on the RVP8's front panel display using the *Front panel display decimal digits – AZ:1 EL:2* question in the **Mc** section. Valid choices are 0-2. Note that the default is now to show one digit for azimuth rather than two, thereby matching the typical RCP8 angle display.

6. Several new elements have been added to the extended DSP parameter readback structure (*struct dspExParm*): Burst frequency (MHz), Burst power (dBm), Temperature of the IFD chassis and onboard FPGA chip, and Mean AZ/EL during the last noise sample.

7. The accuracy of RVP8 angle sync'ing to an accelerating antenna has been improved for the rays immediately following a startup PROC command. An angle smoothing window is used to minimize fluctuation of the number of pulses per ray, and that window had previously been fixed at 1.5 seconds. Now, a dynamic window ranging from 0.25 to 1.20 seconds is used, where the window expands only when the antenna does not appear to be accelerating.

8. The **speed** utility now accepts a *–setvel V* command to set the antenna scan speed to V deg/sec prior to beginning the run. This was added to test New Feature #7.

9. There is a new RCP8 setup question *Echo error signals to the chat interface: YES* in the **site misc** section. Error messages (such as bad angle inputs) are always sent to the error output. This allows them to be sent also to the chat interface. It defaults to the legacy behavior of "Yes". You may wish to turn this off if the messages interfere with use of the chat interface.

Bug Repairs

1. The RVP8 parallel compute processes (*rvp8proc* executable) were sometimes not being killed properly when the RVP8 exited. This would cause subsequent RVP8 startups to fail until those processes were manually removed.

RDA 8.08 Release Notes (3 Jan 2005)

These release notes cover changes made to the SIGMET Radar Data Acquisition platform, including primarily the RVP8 and RCP8 products. The last public release was RDA-8.07.3 dated 22 December 2004. If you are upgrading from an earlier version please also read the release notes that have been published since then.

Important Upgrade Notes

1. If you upgrade a dual–polarization radar be sure to read New Features #3. below and set your fundamental mode in the "Mc" section.

New Features

1. The **DspExport** program now has a heartbeat feature. This allows it to detect a sudden failure of the network link to one of its clients. During quiet times it will poll every 30 seconds. If there is no responding data it will disconnect. This feature is enabled by adding the **`–heartbeat`** command line option to the DspExport call. This has been added to our example startup script, but when upgrading you will need to turn it on explicitly. Note that this requires cooperation from the client program. That means the client IRIS must be at least version 8.07.5. Similarly, if you are running a non-IRIS application check with the manufacturer before turning it on. The source code for the dsp_lib support is in the file libs/dsp/SocketOpen.c.

2. The trigger pattern RAM allocation has been doubled on the Rev.C (72MHz) RVP8/Rx card, making it possible to store twice as much timing information (user triggers, phase control lines, PRF sectors, etc).

3. The RVP8 now supports six fundamental IFD and RVP8/Rx configurations which allow you to choose the best style of IF processing for your particular site. The following table summarizes the options where, ***BW*** is the net IF sampling rate (full 72MHz, or halfband filtered 36MHz), ***DynR*** is the dynamic range (normal single channel, or extra wide dual channel), ***Pol*** is the number of polarizations, ***Freq*** is the number of distinct intermediate frequencies, and ***IFD*** is the number of IFD's, along with their corresponding RVP8/Rx cards.

```
 #   BW   DynR  Filt  Pol  Freq  IFD      Description
 -  ----  ----  ----  ---  ----  ---  ----------------------
 0  Full  Norm  Norm   1     1    1   Standard single channel
 1  Full  Norm  Norm   2     2    1   Dual Pol on two frequencies
 2  Full  Norm  Norm   2     1    2   Dual Pol on separate IFDs
 3  Half  Norm  Norm   2     1    1   Dual Pol on single IFD
 4  Half  Wide  Norm   1     1    1   Extra wide dynamic range
 5  Half  Norm  Long   1     1    1   Extra long/fast FIR filters
```

The first three modes were already supported by the RVP8, but their new implementation has made many subtle improvements (better thermal management within the FPGAs, and closer attention to integer rounding noise). The last three modes are brand new and bring some exciting additional capabilities to the signal processor.

Important: The receiver mode is chosen in the "Mc" menu, but changes do not take effect until they are saved and the RVP8 is restarted.

Important: The factory default receiver mode "0" will be set after upgrading to this release. If you have a dual polarization radar, please be sure to select either "1" or "2" to continue operating as before.

The six receiver modes are summarized below. Please see the *Discussion of Halfband Filtering* later in these notes as it applies to Modes 3-5, and the *Discussion of Wide Dynamic Range* for additional details on using Mode-4.

Mode-0: Standard Single Channel This is the most common "vanilla" mode that is used by single-polarization CW-pulsed radars whose front-end LNA has a dynamic range less than ≈92dB. The (I,Q) data are computed from IF samples at their full acquisition rate (32MHz for Rev.D IFDs, and 72MHz for Rev.F), and the resulting dynamic range from 14-bit IFD samples is well matched to the RF components.

Mode-1: Dual-Pol On Two Frequencies This was the original dual-Pol configuration used by the RVP7 several years ago. A single IFD A/D converter receives the "H" and "V" channels using two distinct intermediate frequencies. Two different STALOs are required in this configuration, making the RF/IF components a bit more expensive, but only one IFD is required.

Mode-2: Dual-Pol On Separate IFDs This mode was introduced into the RVP8 in 2003, and provides dual polarization data using two IFDs connected to two RVP8/Rx cards in the same PCI chassis. A single intermediate frequency is used, hence only one STALO is required.

Mode-3: Dual-Pol On Single IFD This is the recommended dual polarization mode for all new RVP8 installations. The "H" and "V" channels are fed into the Primary and Secondary IFD inputs using a single intermediate frequency. System cost and complexity are both optimized in this design since only a single IFD, RVP8/Rx card, and STALO are required to process both polarization channels.

Mode-4: Extra Wide Dynamic Range Radars having very high performance front-end LNAs can preserve the full benefit of that investment by running two separate IF signals into the Primary (HiGain) and Secondary (LoGain) IFD inputs. A nominal channel separation of 25–30dB might be used to achieve an overall dynamic range of up to 110dB.

Mode-5: Extra Long/Fast FIR Filters This mode is intended for pulse compression systems that require unusually long filters (up to 80μsec), or finer range resolution in order to employ higher compressed bandwidths without the risk of missing echoes between bins. For example, a 30μsec pulse could be processed at an incoming range resolution of 50 meters and then range averaged down to 150meter output spacing.

4. The following new setup questions appear in the **Mp** menu during RxMode-4 (Extra Wide Dynamic Range):

```
IFD Wide Dynamic Range Parameters
  Channel separation: 24.70 dB,  63.5 deg
```

```
Maximum deviation :  0.50 dB,   5.0 deg
Overlap/Interpolate interval: 30.00 dB
```

The *Channel Separation* and *Overlap/Interpolate Interval* should be determined from the **Pr** printout described below. Sweep a SigGen across the shared power region of the two channels to determine a representative channel separation, along with the size of the overlap region at the top of the HiGain channel within which that separation remains steady and constant, i.e., unaffected by eventually approaching the noise floor of the LoGain channel.

The RVP8 continually measures and updates the complex channel separation during its normal course of operation. Ratios of echoes that fall within the overlap/interpolate interval are averaged over several minutes, thereby tracking gain and phase variations that occur with temperature changes and component aging. If the channel separation ever exceeds the specified maximum deviation, the `GI4S_IFDCHANERR` bit (11) will be set in GPARM Immediate Status Word #4.

5. When the **Pr** plot command is used in RxMode-4 (Extra Wide Dynamic Range), the TTY printout shows the complex magnitude and phase between the IF samples of the HiGain and LoGain channels. This is very handy for monitoring the channel separation across a wide range of CW signal generator levels, and for determining the span within which reliable channel ratios can be measured. Be sure that you use the “R” key to move the **Pr** starting range safely away from zero so that none of the IF samples are influenced by the burst waveform.

Bug Repairs

1. The DAFC Fault Pin was not being selected properly in the RVP8 **Mb** menu.
2. The RVP8 would sometimes crash during subsequent data processing when the trigger waveforms did not fully fit into the RVP8/Rx pattern RAM.
3. The RVP8 IF phase unwinding versus range was not working properly when the **Mb** setup question *IF increases for an approaching target* was set to “No”. The effects were minor and could only be observed in **ascope** plots of (I,Q) versus range.
4. A timing skew was repaired in the Rev.E/F IFD that may have caused occasional flurries of incorrect IF samples which might have been seen as brief fluctuations in dBZ.

Discussion of Halfband Filtering Modes 3-5

Traditionally, the IFD used by the RVP7 and RVP8 has sent raw 14-bit A/D samples from its Burst and IF inputs directly to either the RVP7/Main or RVP8/Rx cards for FIR filtering and conversion into complex (I,Q) values. The IFD would function simply as a waveform sampling device (hence the acronym **IF** **D**igitizer), and all of the front-end signal processing took place downstream of it.

This model has changed with the introduction of the Rev.F IFD which has the ability to carry out several billion multiply-accumulate cycles per second. This means that IF samples from multiple signals can be preprocessed entirely within the IFD and then encoded without loss onto the fixed bandwidth of its digital downlink. The new receiver modes 3 through 5 rely on this hardware capability and use a method known as “Halfband Filtering” to effectively double the downlink data rate.

Section 2.2.7 of the *RVP8 User’s Manual* contains a detailed account of how A/D quantization noise affects the dynamic range of the IFD. Briefly, for the Rev.F A/D converter which runs at 72MHz, the contribution of A/D quantization noise within any given 1MHz interval is 72 times smaller than the total noise of the converter itself. This is an important property of all wideband sampling systems: the noise floor after processing, and hence the dynamic range, are improved by increasing the fundamental A/D sampling rate.

Normally the IFD sends 72MHz A/D samples from a single input channel directly down to the RVP8/Rx PCI card. The samples are sent at full speed in order to realize maximum reduction of the final (I,Q) noise floor. But suppose we wanted to send two A/D waveforms down the same data link by interleaving the samples together. Each channel would have to be down-sampled to 36MHz in order to fit within this format, but that would cause its (I,Q) noise floor to increase by 3dB.

To avoid this, we do not create the 36MHz streams merely by discarding every other A/D sample, but rather, by passing the original 72MHz data through a halfband digital filter and then discarding every other point of this filtered A/D stream. The difference is important. Since the halfband filter has removed all of the A/D quantization noise from half of the original Nyquist interval, there will be no increase in noise density within the passband of the (I,Q) filter when the halfband stream is downsampled to 36MHz. Thus, the A/D noise that would normally have folded into the (I,Q) data at 36MHz is first removed by the halfband filter so that we’re left with a 36MHz stream having *the same dynamic range* of the original 72MHz samples.

The IFD halfband filter is a 49-Tap equiripple FIR filter having 40dB of stopband rejection and 0.175dB of passband ripple. The passband extends either from 0–16.5MHz when configured as a lowpass filter, or 19.5–36MHz when configured for highpass. The RVP8 automatically selects the correct type of filter depending on the intermediate frequency specified in the **Mb** menu. The halfband filter has linear phase and is therefore non-dispersive. This means that it is totally suitable for handling compressed pulses and other wideband Tx/Rx waveforms.

Discussion of Wide Dynamic Range Mode-4

When a two channel IFD is used as an extended dynamic range receiver there are some important decisions to make with respect to setting up the RF/IF levels that drive the IFD.

The first of these is the amount of signal level separation between the high gain and the low gain IFD inputs. There is an absolute minimum and absolute maximum channel separation that still allows the IFD to capture the full dynamic range of the receiver. If a signal level separation is made that is outside of these absolute limits valuable receiver dynamic range will be lost.

> ***The absolute minimum separation*** of the channels is equal to the total dynamic range of the receiver minus the dynamic range of a single channel of the IFD. Generally, the total dynamic range of the receiver is set by the LNA. For example, if we are considering a 1μsec pulse (1MHz bandwidth), the dynamic range of the LNA may be about 105dB, and the dy-

namic range of a single channel of the IFD is about 84dB (from –78dBm to +6dBm). In this case, the minimum separation would be 21dB. At minimum separation, the overlap of the low gain channel and the high gain channel will be maximized, and that overlap is equal to the dynamic range of a signal channel of the IFD minus the separation. In this case, the overlap is (84dB – 21dB) = 63dB.

The absolute maximum separation of the channels is simply the dynamic range of a single channel of the IFD. In the above example this would be 84dB. At maximum separation, the overlap of the low gain channel and the high gain channel is zero -- we begin using one as soon as the other has begun to saturate.

We see that there can be a large difference between the absolute minimum and maximum signal level separations; thus additional criteria must be considered to choose an optimum value that is between these diverse limits.

Choosing a proper separation value is a tradeoff of several factors. If the separation value is too low, the IFDs may end up operating very close to their noise floors. And if the separation is too high, then the overlap between the two channels is reduced which makes it difficult for the IFD to make a smooth transition as it combines the data from both channels. Too high a separation may also result in receiver components that are not practical to build.

As a rule of thumb, channel separations in the 22–30dB range provide a good balance of the above criteria. In the case of a 1µsec pulse this results in an overlap interval of approximately 55-63dB, which is sufficient for good IFD transitions and also leads to receiver components that are practical to build.

Once a separation value has been chosen, one must consider how to build the receiver to achieve this. The basic receiver will take the form of an LNA and a mixer followed by a splitter resulting in a low gain channel and a high gain channel. We know the gain difference in the two channels (the separation value), but we must find the actual gain to use in each channel.

If we consider the total system dynamic range as generally set by the LNA (105dB in the above example), we can estimate the minimum detectable signal input to the LNA as well as the maximum usable linear level at the IFD. If the LNA has a noise figure of 1dB and we are using a 1µsec pulse, the minimum detectable signal at the LNA input is –113dBm, and thus the maximum signal is 105dB above this, or –8dBm. If we add to these number the gain of the LNA and the conversion loss of the mixer (and any other losses experienced through the power splitter for the low gain and high gain channels), we can use this information to determine the signal values of the components in these two channels.

For example, if the LNA has a gain of 17dB, the mixer has a conversion loss of 7dB, there is 1dB miscellaneous losses and 3dB loss in the power splitter, then the signal level at the output of the power splitter is (–113 + 18 – 7 – 1 – 3) = –106dBm for the minimum signal, and and –1dBm for the maximum signal. In the low gain channel, we need to bring the –1dBm up to the maximum input value of the IFD (+6dBm). To do this we need about 8dB of amplification (7dB plus one more deciBel to account for the anti–alias filter loss of the IFD). If we assume 25dB of channel separation, on the high gain channel we require about +33dB of amplification. Finally, this tells us that on the low gain channel, the minimum and maximum signals presented to the IFD are (–106 + 8) = –98dBm and (–1 + 8) = 7dBm. For the high gain channel, the signal levels are (–106 + 33) = –73dBm and (–1 + 33) = +32dBm. Note that as +32dBm is above the maximum input level tolerated by the IFD, the amplifier on the high gain channel must limit its output to less than +16dBm. Thus an amplifier with an output saturation value of between +10dBm and +15dBm should be used.