Field Guide to Byte Slips

07 jan 02; rvsd 21 jun 02

This document looks at the two types of byte slips, which arise in the DMM (delay memory module) or TRM (track recovery module). In general, the effect of a byte slip is to shift some or all of the "power" in a fringe from the original peak-lag to a different lag, some multiple of 8 lags away. Once a byte slip occurs in a subjob, it remains "in effect" until the end of the subjob or a gap long enough to cause the tapes to stop, but the following subjob (or scan after the gap) begins unaffected. In principle, we can see the effect of byte slips in a multiply-peaked amplitude-*vs*-lag spectrum, but the reality of too-dim sources, too-low weights, too-few lags *etc.* complicates diagnosis somewhat. Subjobs with byte slips should be recorrelated. In practice, the process of finding subjobs affected by byte slips and arranging for recorrelation is probably the single most time-consuming step in the current FITS-file preparation procedure. This guide collects sample diagnostic plots to aid detection, but much still depends on the design of individual experiments. It seems difficult to come up with a single magic bullet.

1. DMM Byte Slips

***** SYMPTOMS *****

- At some time t_1, the entire peak shifts by 8 lags in one SB/pol on all baselines to a specific station (hereafter *STA*). So far, DMM byte slips have always affected the 2$^{\text{nd}}$ channel as defined in the `VEX` file.
- There can be more than one DMM byte slip in a subjob.
- If *STA* occurs first in the baseline, the byte slip moves the fringe to a higher-numbered lag.
- DMM byte slips seem more prevalent when a correlation uses fewer SB/pol channels (*i.e.*, line rather continuum experiments).
- currently all(?) affected DMMs have already been moved to the upper half (*i.e.*, for the 2$^{\text{nd}}$ head), so we don't see these for the time being since we won't be installing the second head. However, when Mk V comes on-line and two-head recordings can be correlated in a single pass, byte slips originating in these boards may well recur.

***** PLOTS *****

Below are a series of illustrative DMM byte slip plots, taken from the experiment EO004A (line mode).

Figure 1 shows the amplitude-*vs*-lag spectra for all baselines in a given subjob.

You can see that all baselines to Wb in RCP (blue) have two sub-peaks below the main peak, suggesting that there were two byte slips in this subjob.

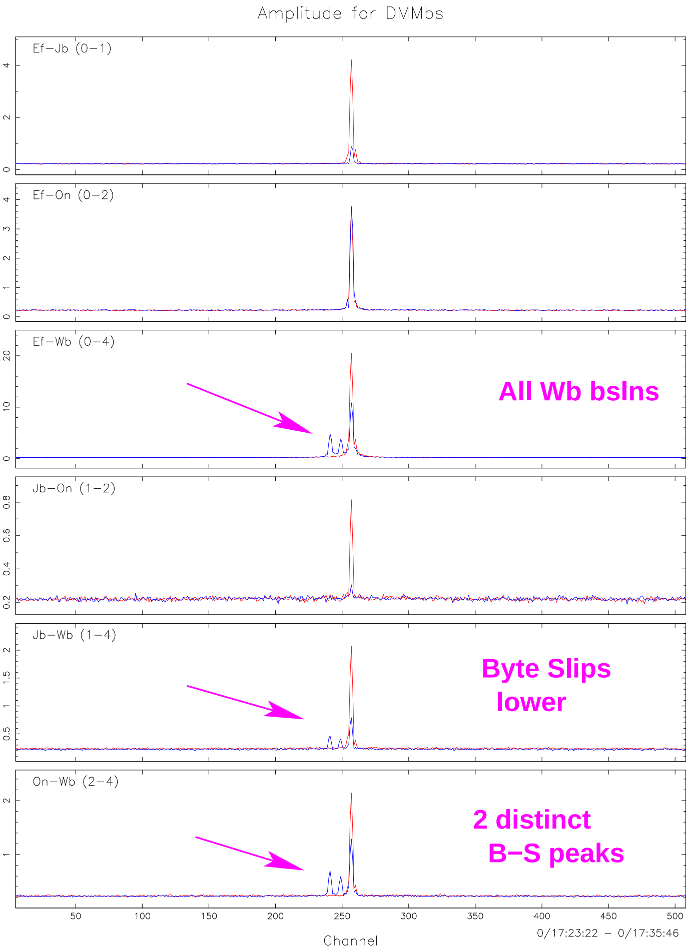

Fig. 1: Amplitude-*vs*-lag spectra for a DMM byte slip.

Figure 2 shows the amplitude-*vs*-lag spectra for all integrations on the baseline Ef–Wb (SB0, RCP) overplotted together. You can see the three peaks, each separated by 8 lags. The peak moves as a whole: the amplitude of the peaks for each integration stays the same, and there is an "out-of-peak" floor under each. The amplitudes of the sub-peaks in Figure 1 are different because it is an averaged plot and each of the "byte-slip states" lasts for a different duration, shown in Fig. 3.

Figure 3 plots the location of the peak lag *vs* time for both polarizations of Ef–Wb, SB0. The LCP channel, unaffected by byte slips, is constant throughout the subjob at about lag=257. Two distinct byte slips can be seen in the RCP channel, at $t_1 \simeq 17^{\mathrm{h}}31^{\mathrm{m}}$ and $t_2 \simeq 17^{\mathrm{h}}33^{\mathrm{m}}$. The longest "byte-slip state" is for $t < t_1$, and the shortest for $t_1 < t < t_2$; the relative amplitudes of the averaged amplitude-*vs*-lag spectra in Fig. 1 agree with this.

Turning to the effects of the DMM byte slip on the data (as typically considered by PIs), Figure 4 plots the vector-averaged amplitude (from the frequency M.S.) *vs* time for all baselines (same polarization/color scheme as in Figure 1). You can see

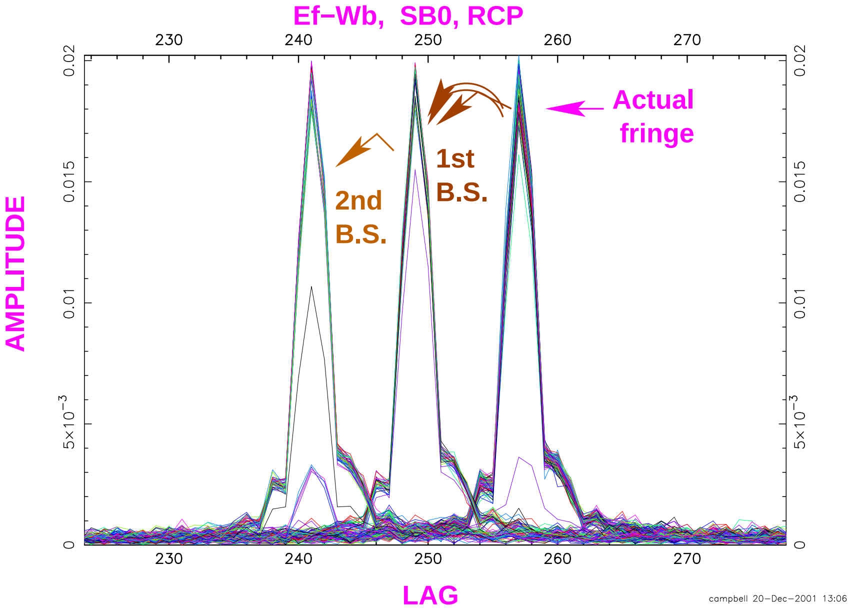

Fig. 2: Individual amplitude-*vs*-lag spectra for Ef–Wb (SB0, RCP).

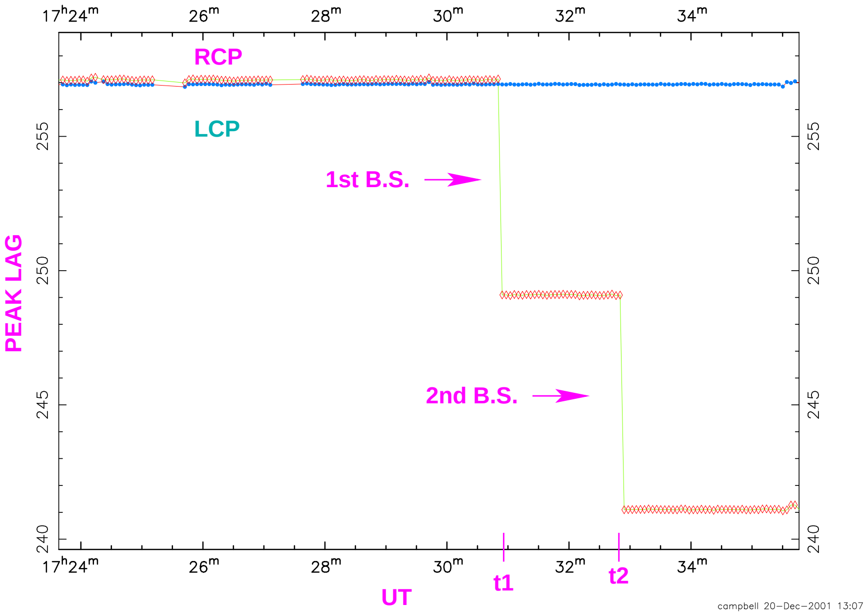

Fig. 3: Peak lag *vs* time for Ef–Wb, SB0.

an abrupt drop in RCP amplitude at t_1. This occurs because the byte slip introduces an offset of 8 lags in the location of the peak lag, which in turn causes much faster wrapping of phase as a function of frequency across the band (see Fig. 5). A scalar-averaged amplitude plot wouldn't care about this, and indeed doesn't show any RCP amplitude drop. Note that if the solution interval in fringing were set such that fringe-interval boundaries fell at t_1 and t_2, then there would be no problem in the post-fringed data. However, since this would in general be impractical, post-fringed intervals containing the byte slips would have corrupted delays/rates/phases.

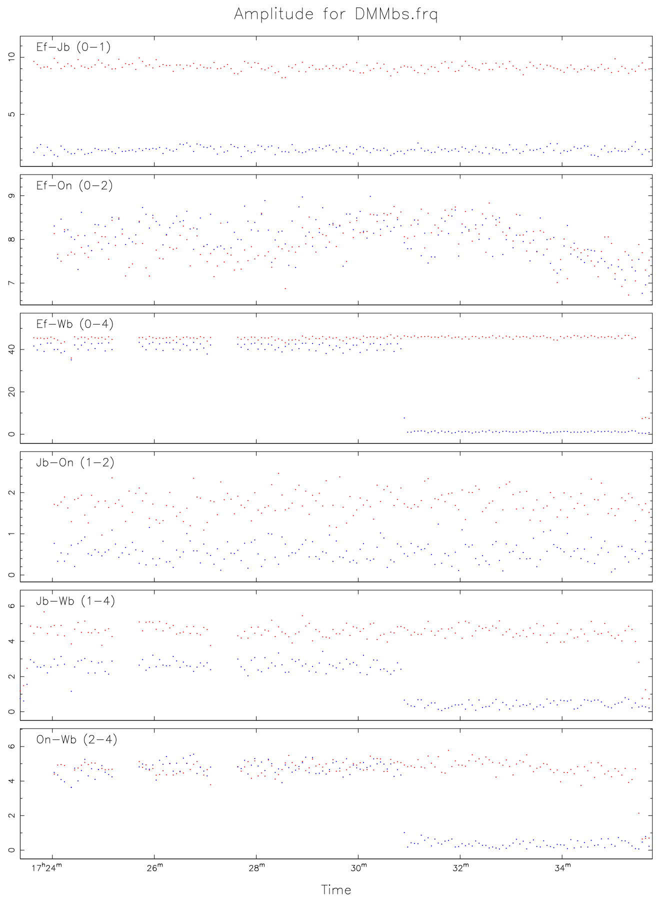

Fig. 4: Amplitude (vector-averaged) *vs* time for a DMM byte slip.

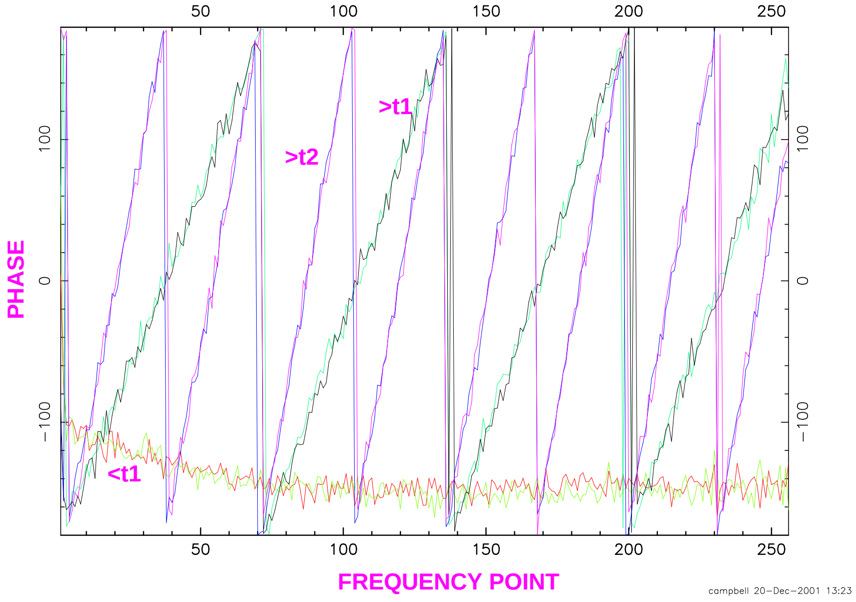

Fig. 5: Phase *vs* frequency across the band for Ef–Wb (SB0, RCP).

Figure 5 shows the effect on phase across the band in the three different "byte-slip states": two integrations drawn from $t < t_1$ (red, light green), two from $t_1 <$

$t < t_2$ (teal, black), and two from $t > t_2$ (dark blue, pink), still for the Ef–Wb baseline (SB0, RCP).

The extra phase-wrapping across the band will also affect the vector-averaged amplitude across the band. Figure 6 shows this for all baselines. The effect is clearly evident in Ef–Wb, fairly so in On–Wb, but only marginal in Jb–Wb (the RCP channel from Jb being rather weak). Similarly as for amplitude *vs* time, a scalar-averaged plot would show no problem. Note that because the fringe moves after the channel has be reconstituted (in the CRM), the autocorrelation bandpass for each station is fine; only the baselines are affected.

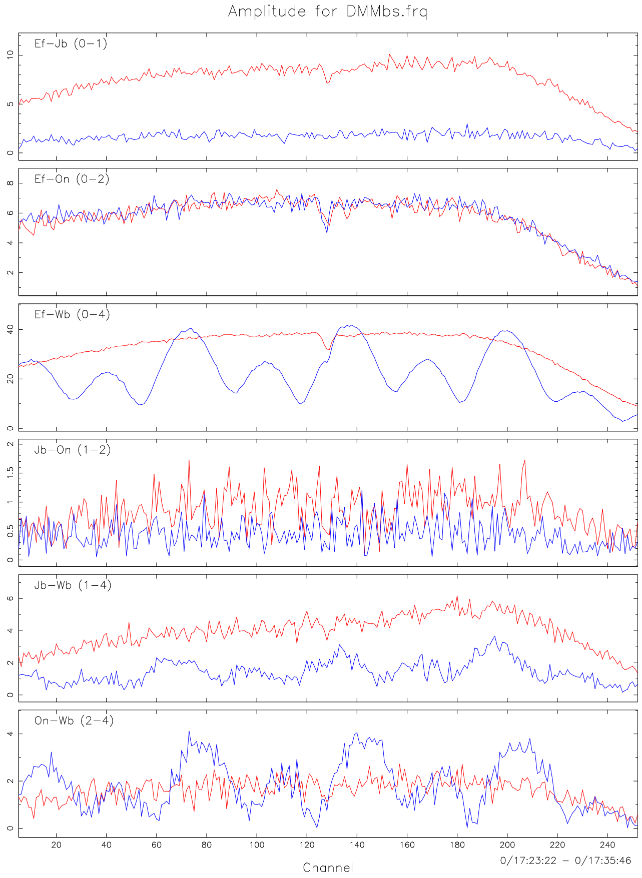

Fig. 6: Amplitude (vector-averaged) *vs* frequency across the band for a DMM byte slip.

Figure 7 shows the phase *vs* time for all baselines (vector average). You can see the breaks at t_1 and t_2 in Ef–Wb, and the loss of good detections in the weaker Wb baselines at t_1.

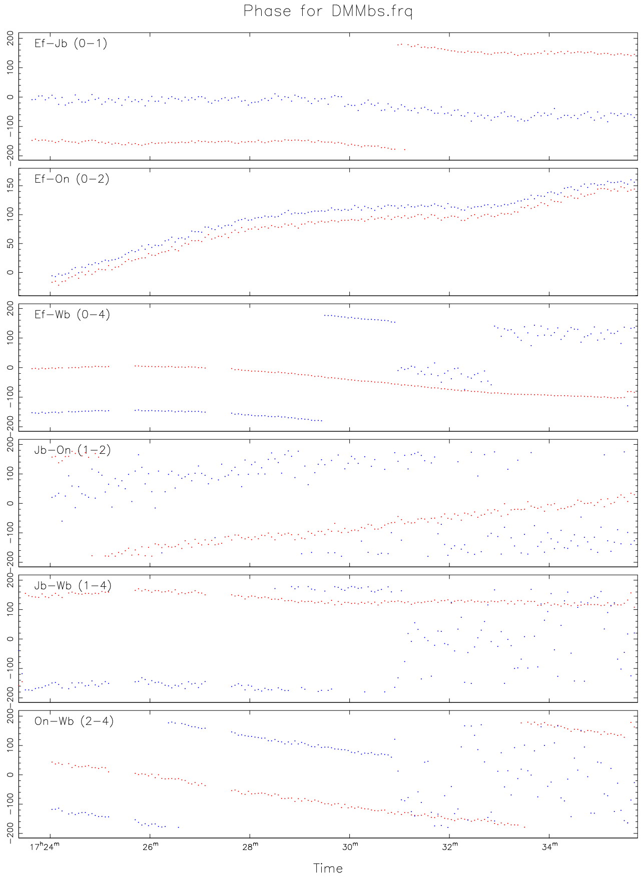

Fig. 7: Phase *vs* time for a DMM byte slip.

*** OPERATIONAL THOUGHTS ***

- As mentioned, all known DMMs that have ever caused byte slips have been moved to the upper half, so as long as only one head is used there shouldn't be any more DMM byte slips (barring a currently good DMM evolving into a byte-slipper). The DMMs with a history of byte slips should be labelled either on the board or in a logbook.
- The figures in this section come from a full-pass scan of a bright source, so the byte-slip effects are easy to see. In most cases, there will be complications in finding DMM byte slips, and hence in determining which subjobs should be recorrelated.
  - Byte slip effects as illustrated above won't be apparent for sources too faint to yield fringes in an integration. In a phase-referencing experiment, the effect of byte slips and their persistency throughout a subjob can be checked via the reference-source scans: if there's sign of a byte slip in the last reference-source scan, then a byte slip happened sometime earlier during the subjob even if its exact time can't be determined (but this is enough evidence to include this subjob for recorrelation). If the PI scheduled such that a subjob ends with a scan of a too-dim source, then we'll be insensitive

to finding byte slips that occur after the last scan of a bright-enough source.

- The greatest nuisance about byte-slip hunting is that the irrefutable smoking gun lies in lag-based measurement set (MS_ℓ). The frequency measurement set (MS_ν) obviously has to be made in the process of making the FITS file for the PI, but a MS_ℓ wouldn't really have to be if byte slips never happened. The behavior of the amplitude/phase *vs* time from the MS_ν (Figs. 4 and 7) could have other, non-byte slip causes, for which recorrelation would be unnecessary (*e.g.*, telescope goes off source [but not for Ef–Wb $\varphi(t)$, which doesn't mimic a non-detection]). Plots of amp/real/imag across the band (Fig. 6), especially in combination with good autocorrelation plots, seem stronger, more focused evidence. Perhaps individual subjob MS_ℓ could be made only when needed to investigate specific incidents.

- The ephemeral nature of the byte slips also makes searches that don't look at least at the last bright-enough source in every subjob liable to omissions. However, individual time-averaged plots (Fig. 2) over too many subjobs may also miss some byte slips, since their detectability would be roughly proportional to the ratio of time for which data are affected by a byte slip to that unaffected, weighted by the brightness of the sources. For instance, a fringe-finder pass plus 10 passes of a ~50 mJy source, once of which has a byte slip ~halfway through, all averaged together in one plot might mask the effect of the byte slip. An additional drawback to long averages is that once the effect of the byte slip is seen, the offending subjobs still need to be isolated from time-based plots. Finally, the longer the average the greater the chance there could be byte slips for multiple stations, with the potential for more confusing symptoms. There's probably some objective way of defining an optimal averaging time as a function of various parameters, but I've never bothered.

2. TRM Byte Slips

***** SYMPTOMS *****

- At some time t_1, in one SB/pol on all baselines to a specific station (*STA*), some power (almost always $< 50\%$) shifts from the main peak to another one $8 \times f$ away, where f is the fan-out factor. Since they are associated with specific TRMs, which could be positioned anywhere in the SU, these are not limited to the 2nd `VEX`-file channel.
- TRM byte slips occur only with Mk V-format data. There are significantly more prevalent ($\sim \times 10$) with 8 Mb/s/track recordings than with 4 Mb/s/track recordings. Some individual boards are significantly more prone to TRM byte slips (worst ones can give a byte slip every subjob).
- If *STA* occurs first in the baseline, the byte slip moves the (additional) fringe to a higher-numbered lag (same as for the DMM byte slips).
- If the correlation uses a small number of lags and a high fan-out factor, the additional fringe can be moved out of the window, and the only sign of the TRM byte slip would be an sudden loss of amplitude in the main peak. For instance, in a standard continuum ($N_\ell = 32$) experiment with fan-out of 4, a TRM byte slip would shift the additional peak ∼16 lags outside the window (even fan-out of 2 would push it right to the edge).
- Currently, most affected TRMs have already been moved to the upper half (*i.e.*, for the 2nd head), but we still encounter TRM byte slips on some SUs (the 8 *vs* 4 Mb/s/track dependence wasn't known from the beginning, so identification of "bad" TRMs has been biased against those boards that belonged to SUs that were in use in 8 Mb/s/track experiments). The concern about re-introducing known byte-slippy boards when Mk V, two-head recordings begin remains.

***** PLOTS *****

Below are a series of illustrative TRM byte slip plots, taken from the experiment S01C3B.

Figure 8 shows the amplitude-*vs*-lag spectra for all baselines done in a given subjob. You can see that all baselines to Mc (SB1, RCP) have a second sub-peak, in the direction of the "first=higher" rule.

Figure 9 shows the amplitude-*vs*-lag spectra for all integrations on the baseline Mc–Jb (SB1, RCP) overplotted together. Up to the time of the byte slip, there is just the main peak at $\ell = 33$. After the byte slip, the fringe splits into two separated by 16 lags (fan-out factor is 2). A distinguishing feature of the TRM byte slip is that part of the fringe remains at the original peak lag at all times, so that it doesn't have any of the "out-of-peak" floor under it. On the other hand, since the second peak usually (always?) has lower amp, the byte slip won't show up in plots of the location of the peak lag *vs* time as in Fig. 3. (It will show up in

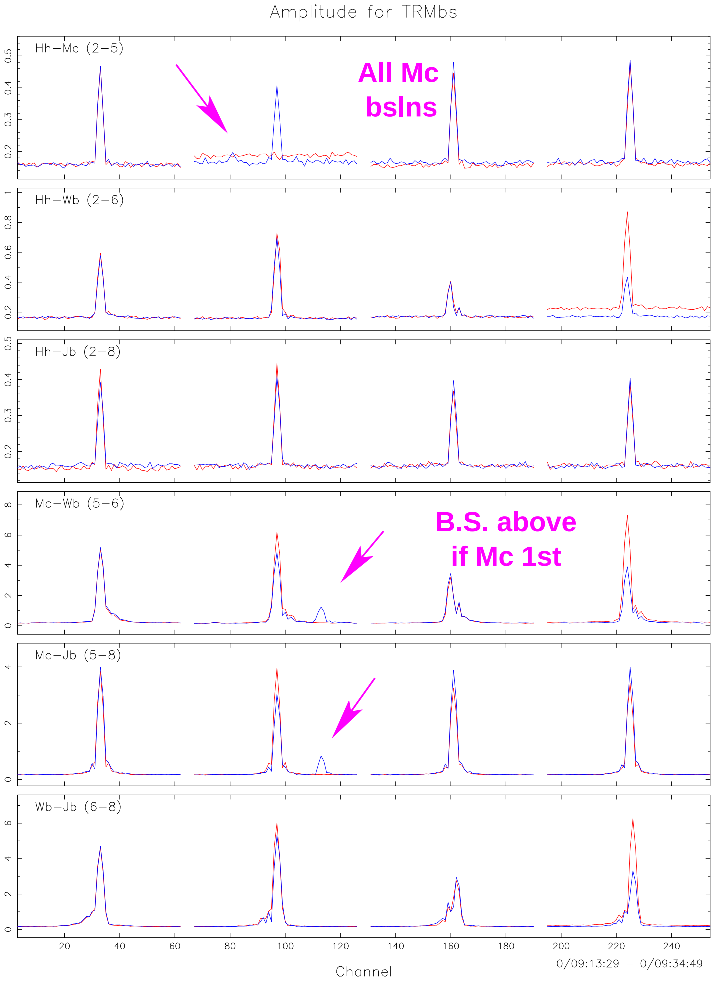

Fig. 8: Amplitude-*vs*-lag spectra for a TRM byte slip.

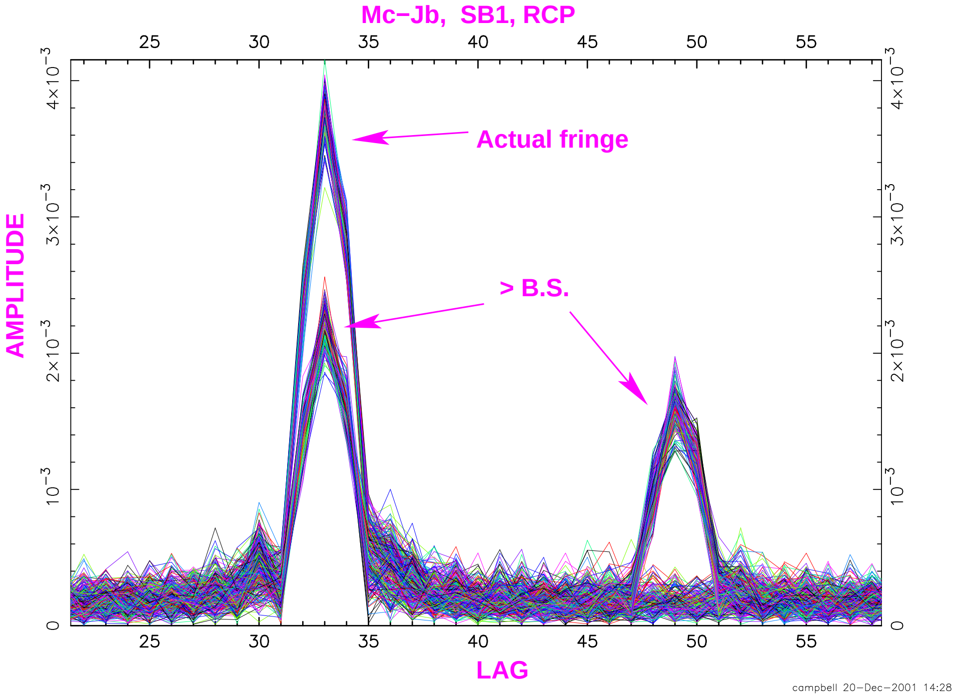

Fig. 9: Individual amplitude-*vs*-lag spectra for Mc–Jb (SB1, RCP).

plots of the amplitude of the peak lag *vs* time, but you'd have to know where the peak lag was first; such a plot is not shown here. Similarly, the $amp(t)$ of the lag to which the byte-slip jumps, if there enough lags for the jump to stay within the window, would show a corresponding increase.)

Turning to the effects of the TRM byte slip on the "PI data", Figure 10 plots the amplitude (from the frequency M.S.) *vs* time for all baselines. You can see an abrupt drop in SB1,RCP amplitude at t_1 in Mc baselines (at least the ones that are sensitive enough). In the case of TRM byte slips, the scalar/vector averaging doesn't matter to the existence of the amplitude drop. Also, because the fringe splits rather than moves, the fringing with precisely controllable fringe-interval boundaries will not salvage the post-fringed data for $t > t_1$ as in the case of DMM byte slips.

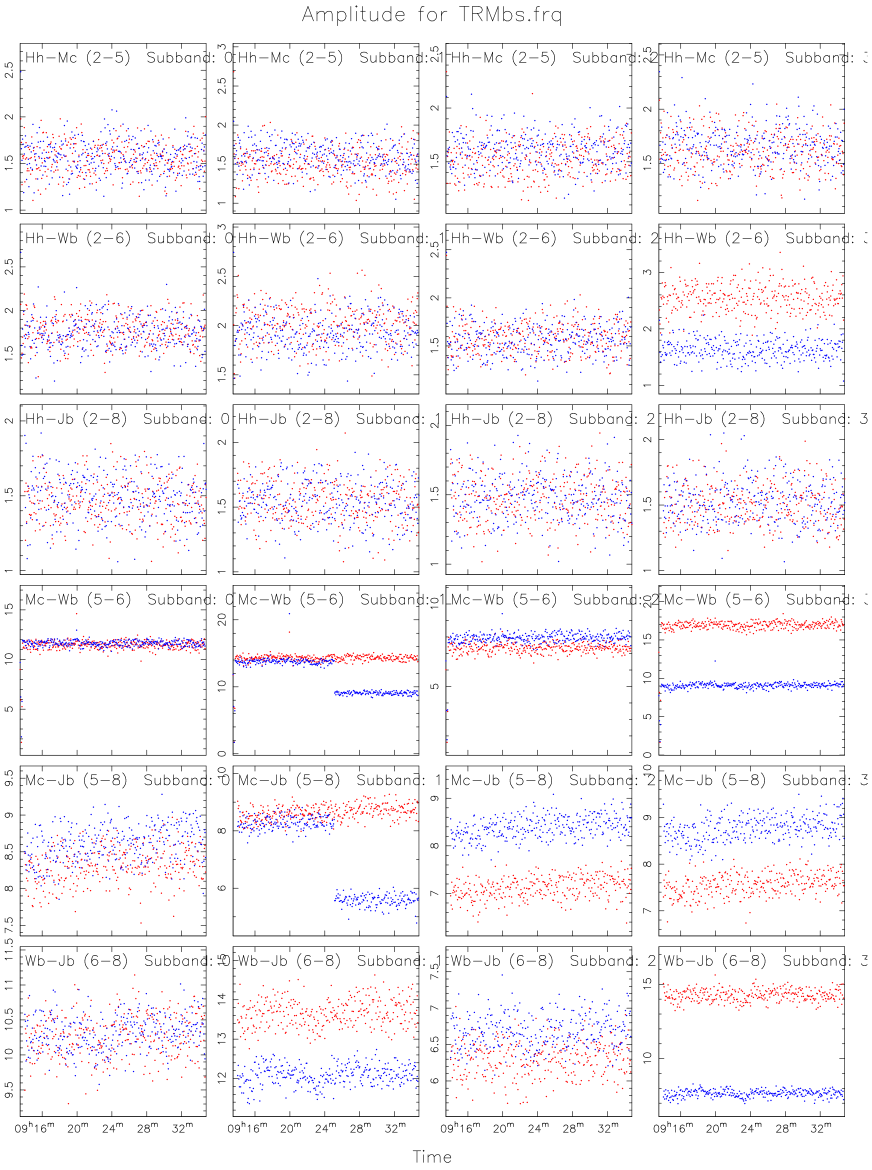

Fig. 10: Amplitude *vs* time for a TRM byte slip.

The period $t > t_1$, when there are two peaks in the lag spectrum, introduces a sinusoid behavior into the Fourier transform, which can be seen in Figure 11, the amplitude across the band. This ringing occurs for both scalar and vector averaging (the broader oscillations of the DMM byte slip occur only for vector averaging, as in Fig. 6).

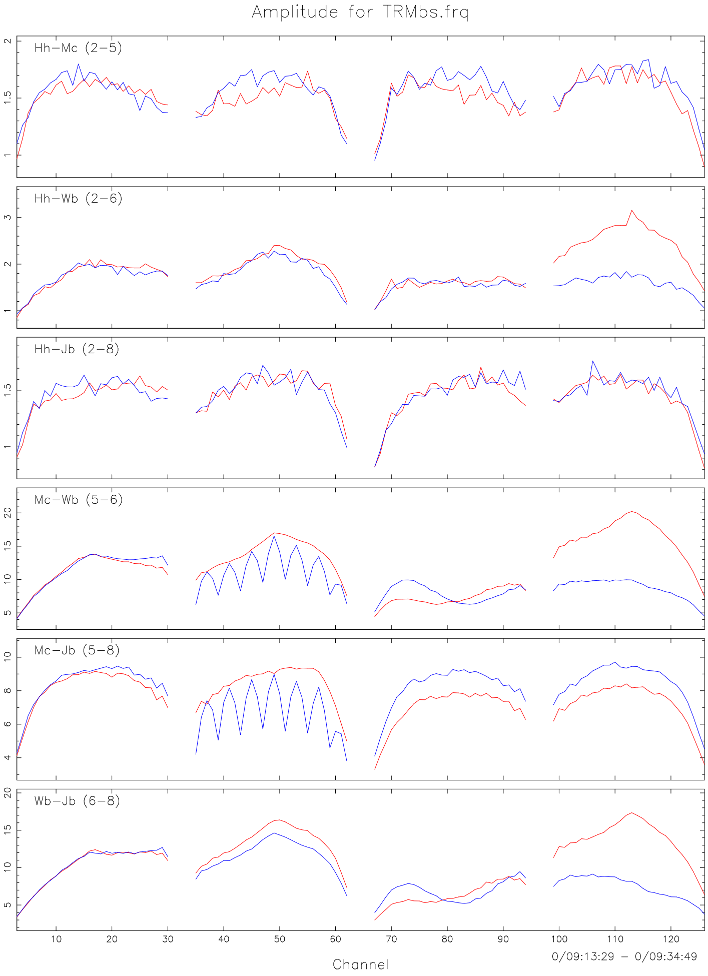

Fig. 11: Amplitude *vs* frequency across the band for a TRM byte slip (if there were enough lags to see the post-byte-slip secondary peak).

However, the effect of TRM byte slips is not so noticeable in phase *vs* time plots, as seen in Figure 12. There is a little more variance, but since the phase is only the direction of the vector average of the individual frequency-point phasors across the band, the ringing in $amp(\nu)$ essentially downweights the low-amplitude frequency points (*i.e.*, those affected by the existence of the 2nd peak). The magnitude of either sort of average will be reduced, hence the amplitude drop seen in Fig. 10.

Figure 13 follows the phase on frequency point 15 (a “low-amp” point) as a function of time for both polarizations of SB1. The increase in variance after t_1 for the RCP channel is evident.

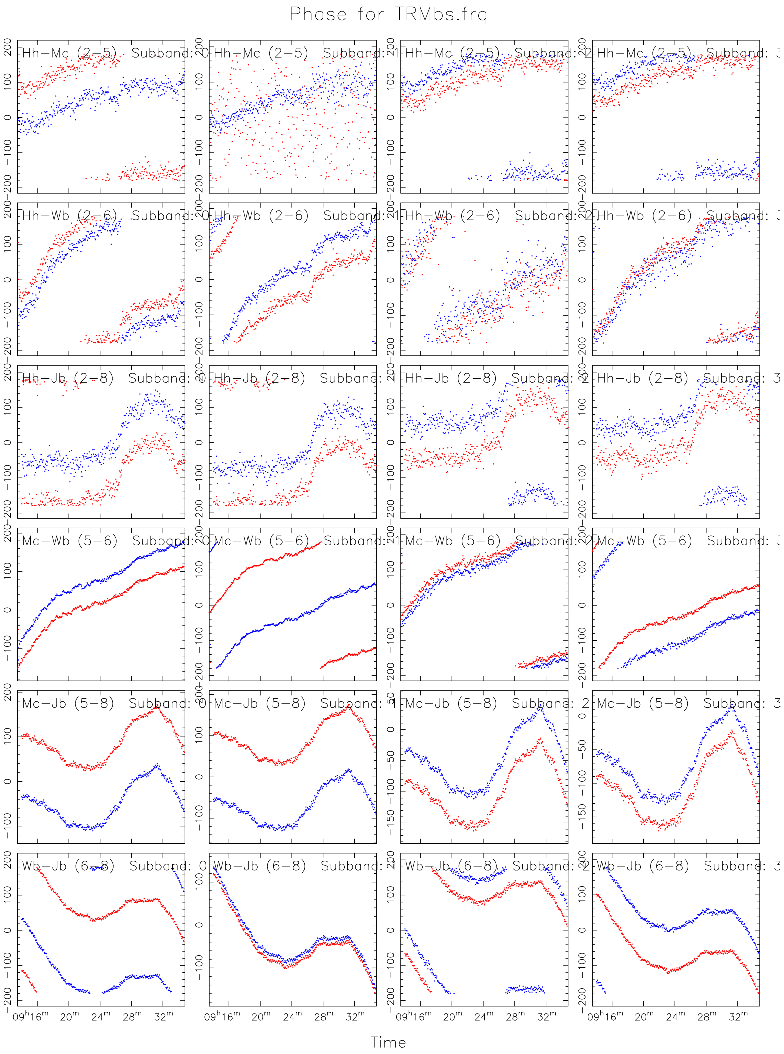

Fig. 12: Phase *vs* time for a TRM byte slip.

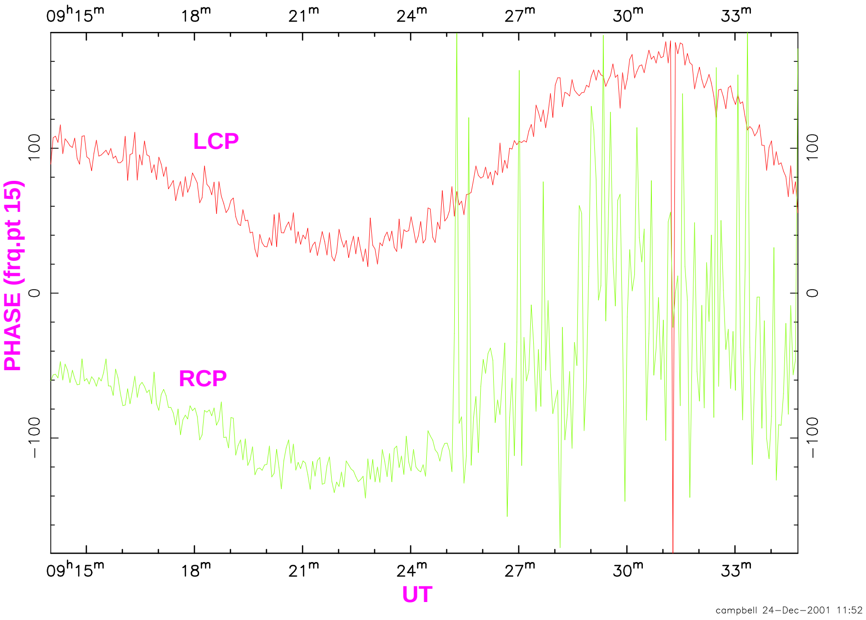

Fig. 13: Phase *vs* time for frequency point 15 Mc–Jb, SB1.

There is a beneficial conspiracy of unrelated problems that provides an empirically easier way to detect TRM byte slips. Although stated rather declaratively below, actual cause/effect relationships are still hypotheses. It is also not entirely clear how final resolution of the underlying problems that allow these TRM byte-slip diagnostics (2-bit normalization) will affect their continued usefulness for TRM byte-slip diagnosis.

Because the TRM byte slips occur in the TRM and hence affect a specific track (or possibly set of tracks) prior to the physically observed channel being reconstituted in the CRM, the sampler statistics for the resulting channels can be affected. Because the correlator uses a truncated, unsigned, 2-bit multiplication table, non-Gaussianly distributed sampler statistics can affect the normalization of the output correlation coefficients, especially for the autocorrelations, which should have peak amplitudes of unity (this has been known since Summer 2001 as the "Sampler-Stats Effect"). Figure 14 shows the behavior of the autocorrelation peak amplitude as a function of the fraction of "high" bits (*i.e.*, both signs). The data are from ED017B, which had almost all ranges of $\Sigma(highs)$ sampled by one station or another, allowing the curve to be well traced (note that there are about 23,000 individual points in this plot).

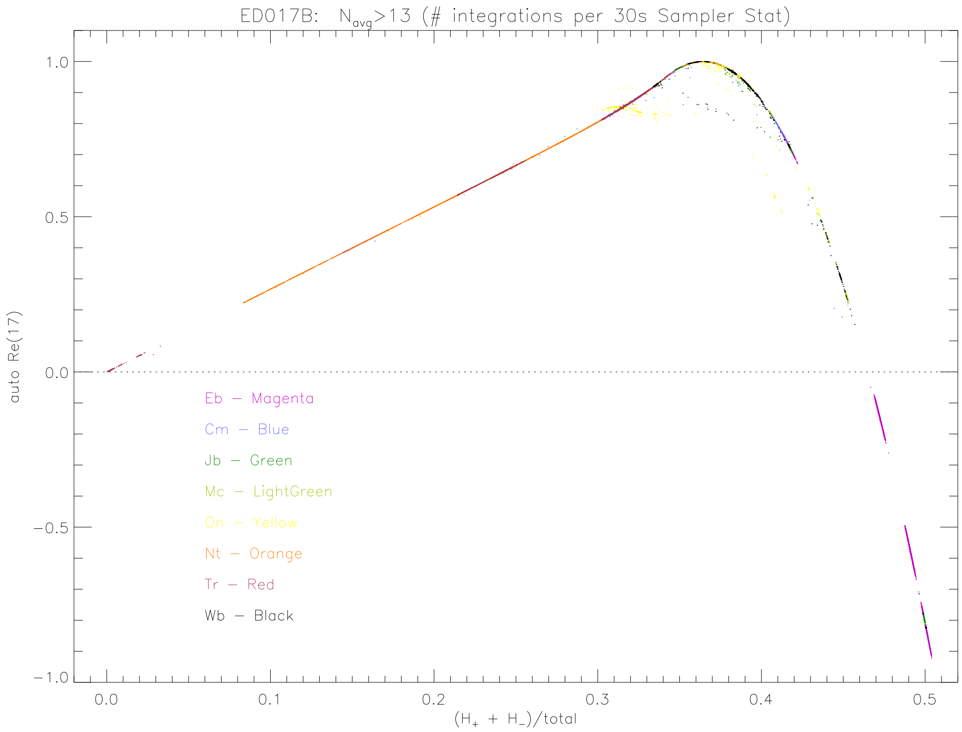

Fig. 14: Autocorrelation peak *vs* fraction of high bits.

We find empirically that a TRM byte slip will cause a $\Delta\Sigma(highs)$ of about 3% for fan-out 4, and about 6% for fan-out 2 (and assumedly correspondingly higher for fan-out 1). The resulting change in the autocorrelation peak real will depend on the initial value of $\Sigma(highs)$ for the station/SB/Pol having the byte slip, and on the direction of $\Delta\Sigma(highs)$. Figure 15 plots the real of the central lag of the autocorrelation of Mc for SB1 (the peak lag of autocorrelations is always $N_{\mathrm{lag}}/2+1$, so you don't need to worry about the location of the peak lag as for baselines). The drop in amplitude in RCP around $t = 9^{\mathrm{h}}25^{\mathrm{m}}$ is clear. Again, the actual change in the autocorrelation real could be different (even an increase) depending on the fan-out and the sampler-statistics behavior mentioned above.

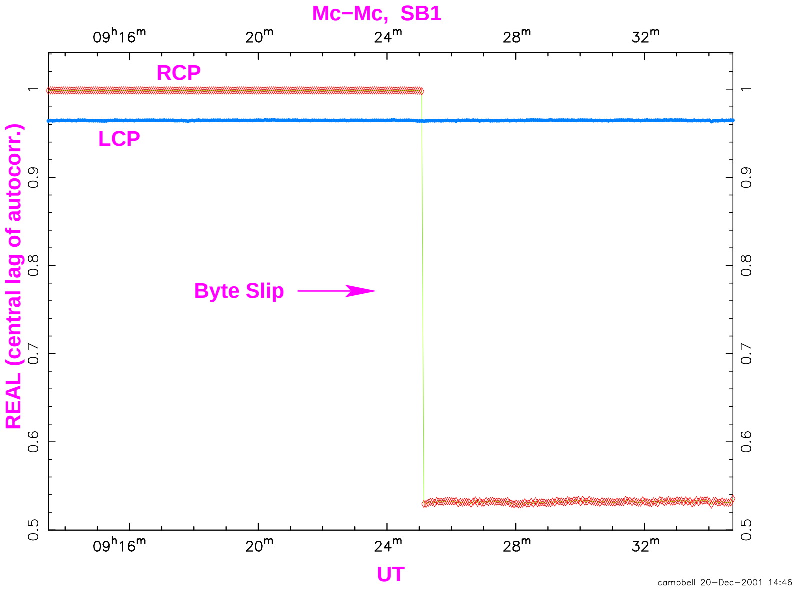

Fig. 15: Real *vs* time for lag 33 Mc–Mc, SB1.

Figure 16 shows the Mc (SB1, RCP) autocorrelation real-*vs*-lag spectra for all integrations overplotted together. These break up into two distinct groups: before the byte slip (peak ~1, little signal at ±16 lags) and after the byte slip (peak reduced, more signal at ±16 lags). The excess power at ±16 lags would of course be harder to see if there were only 32 lags in the correlation.

Figure 17 shows averaged real-*vs*-frequency spectra (*i.e.*, autocorrelation passbands) for all stations (type of averaging irrelevant for the reals). Ringing for Mc (SB1, RCP) is evident (again, given that there were enough lags in the correlation to see the post-byte-slip secondary peaks).

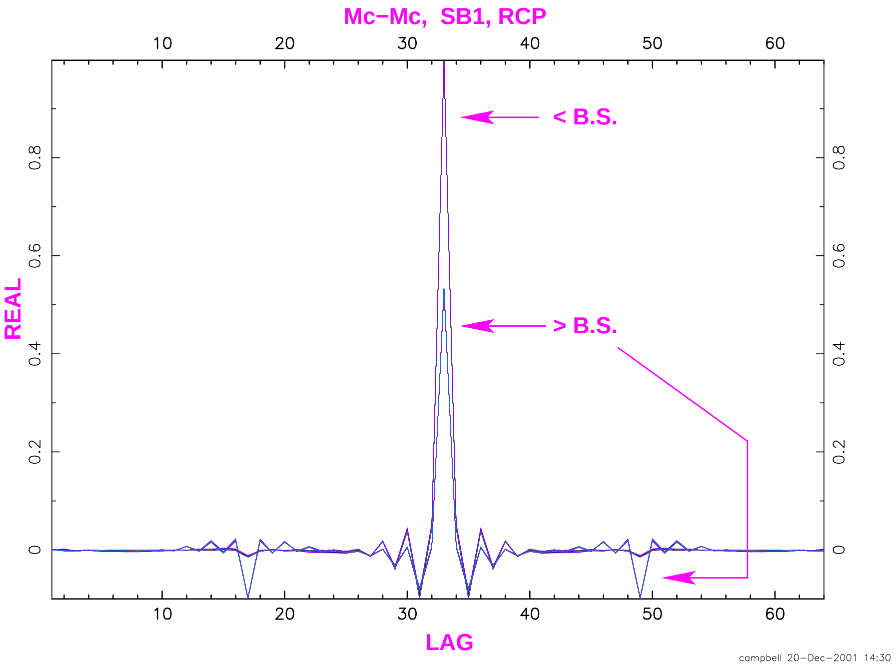

Fig. 16: Individual Real-*vs*-lag spectra for Mc–Mc (SB1, RCP).

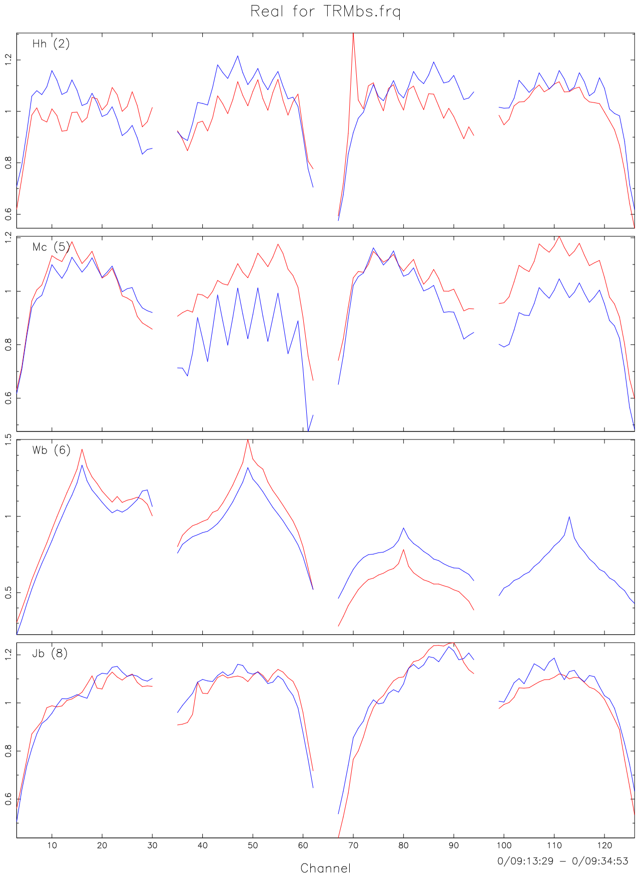

Fig. 17: Real *vs* frequency (autocorrelations) for a TRM byte slip.

*** OPERATIONAL THOUGHTS ***

- As mentioned, most TRMs that have caused byte slips have been moved to the upper half. However, although the DMM byte slips seem to be in hibernation, we keep coming across new TRM byte slips on specific SUs. Whether this means that TRMs can evolve into a byte-slipper more easily is unknown to me. Recognizing time-evolution of the susceptibility of individual TRMs has been complicated by the facts that:
  - ○ the dependence on the track bit-rate (8 Mb/s/track giving many more byte slips than 4 Mb/s/track) wasn't appreciated until after initial statistics were compiled, and
  - ○ there were preferentially more 4 Mb/s/track experiments done earlier, and conversely more 8 Mb/s/track experiments done nowadays.
- The great advantage of being able to use autocorrelations for diagnosis of TRM byte slips is that the process becomes largely independent of source brightness (weak sources not a problem as for detecting DMM byte slips). Further, we know the location of their peak lag *a priori*, we only have to use the reals rather than computing amplitudes, and only have to worry about N_{sta} rather than N_{bsln}. There is now an command-line option in `plotweight.pl` that allows the autocorrelations to be reviewed from the raw data once correlation is completed (*i.e.*, no measurement set has to be made), so operators can get immediate feedback and run recorrelations before the tapes are removed. Unfortunately, this autocorrelation trick doesn't work for DMM byte slips, nor does it work with 1-bit recordings, since the data normalization doesn't have this sort of sampler-statistic dependence. It is also not clear whether it will continue to work for TRM byte slips once the raw data are normalized properly in accordance with the observed station sampler stats.
- In principle, sudden changes in sampler stats could yield an autocorrelation real-*vs*-time signature on the peak lag similar to that in Fig. 15, but experience has so far shown this not to be the case. Secular sampler-stats change on specific BBCs tend to be more gradual (∼ hours). Stations seem to have different source- or source-change related sampler-stat variations (AGC decay times?) that can also feed into the autocorrelation reals; these could in principle be confused with a byte-slip, especially if there were only one source change in a subjob. Actual byte slips have no (known) relation to scan boundaries and never recover (until the next subjob or gap in the tape), which should allow discrimination of any source-change related autocorrelation variations in phase-reference observations.
- Steve & Sergei are working with Metrum to come up with a xilinx modification that will detect TRM byte slips and correct them (mark data as invalid & compensate for the time associated with the byte slip) by the next tape frame after the detected byte slip. Hopefully, this whole TRM byte slip section will soon be of historic interest only.