

# Closed-Loop Analysis of Soft Decisions for Serial Links

Chatwin A. Lansdowne, *Member, IEEE*, Glen F. Steele, Joan P. Zucha, and Adam M. Schlesinger

**Abstract**—Modern receivers are providing soft decision symbol synchronization as radio links are challenged to push more data and more overhead through noisier channels, and software-defined radios use error-correction techniques that approach Shannon's theoretical limit of performance. The authors describe the benefit of closed-loop measurements for a receiver when paired with a counterpart transmitter and representative channel conditions. We also describe a real-time Soft Decision Analyzer (SDA) implementation for closed-loop measurements on single- or dual- (orthogonal) channel serial data communication links. The analyzer has been used to identify, quantify, and prioritize contributors to implementation loss in real-time during the development of software defined radios.

**Index Terms**— block codes, channel coding, codes, satellite communication, test equipment

## I. INTRODUCTION

SOFTWARE defined radios are answering today's challenge of transferring more data in less bandwidth with less signal and more interference and multipath, by using aggressive error correction codes, transmit pre-filters, tighter constraints on signal formation, and dynamically adaptive filter compensation. The corresponding decoders are usually premised on additive Gaussian white noise (AWGN).

But what is the performance of the receiver and what is the performance of the decoder in a systems environment? How can we separate the implementation losses of the receiver from those of the decoder for a specific transmitter implementation and a non-ideal but mission-representative channel? [1] To address these questions, we developed an instrument that can accurately measure the  $E_s/N_0$  and SER at the input to the decoder for an arbitrarily formatted stream.

Manuscript received October 30, 2012. This work was performed in NASA Johnson Space Center's Avionics Systems Division. Funding was originally provided by the Constellation Program.

C. A. Lansdowne is with the National Aeronautics and Space Administration, Houston, TX 77058 USA (phone: 281-483-1265; fax: 281-483-6297; e-mail: chatwin.lansdowne@nasa.gov).

G. F. Steele is with the National Aeronautics and Space Administration, Houston, TX 77058 USA (phone: 281-483-0191; fax: 281-483-6297; e-mail: glen.f.steele@nasa.gov).

J. P. Zucha is with ITT at the National Aeronautics and Space Administration, Houston, TX 77058 USA (phone: 281-483-2386; fax: 281-483-6297; e-mail: joan.zucha-1@nasa.gov).

A. M. Schlesinger is with the National Aeronautics and Space Administration, Houston, TX 77058 USA (phone: 281-483-0342; fax: 281-483-6297; e-mail: adam.m.schlesinger@nasa.gov).

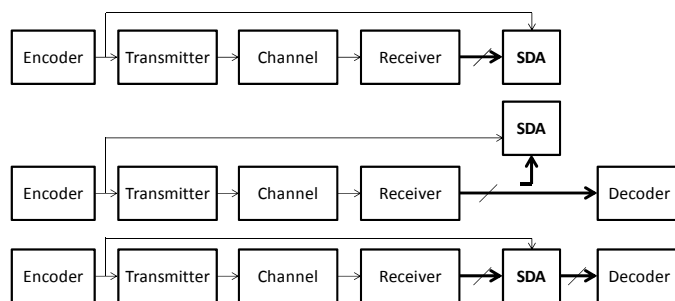


Figure 1 Closed Loop and Soft Decision Test Points

Closed-loop measurements allow us to know the performance of an actual combination of hardware and test conditions while the intended application is running. This is essential for measuring the performance of error-correction algorithms in their actual operating environment. The expedient of measuring link performance using a pseudorandom (PN) sequence, then measuring the performance of the decoder and application, is generally a very tedious and unsatisfactory exercise. Link performance changes when the link reacquires after the data source is reconfigured due to the lock phase and symbol timing, and drift can be very significant compared to the steepness of a decoder threshold. Using this technique, it is not necessary for the instrument to depend on the data format or even the data rate.

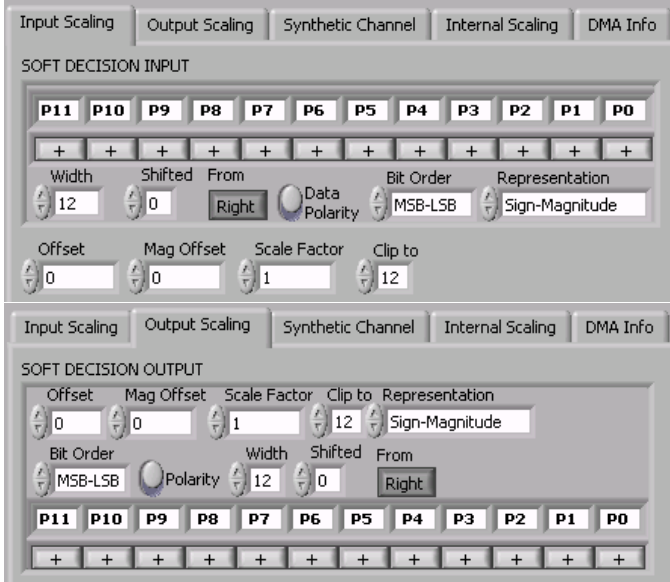
By comparing the soft-decision test point with the transmitted sequence and analyzing the result, we were able to construct a wide variety of helpful indicators and useful figures of merit that accelerate the work of communications system development and integration.

## II. THE SOFT DECISION INTERFACE

Every receiver we have examined has implemented the soft decision interface as a bus. Bus widths vary, from hard decision to 3, 5, 6, 7, 8, or even 12 bits wide. These bits describe the polarity and relative distance of the decision from the decision threshold, using sign-magnitude, one's complement, or two's complement representations. For quadrature links (quadrature phase shift keyed, QPSK, where the channels may be staggered, SQPSK, also called offset, OQPSK), two busses may be provided, or a single bus may be commutated with the channel source possibly identified using a channel flag signal.

### A. Calculating Translation Tables

The most simple and versatile interface implementation for interpreting the bus states is a lookup table. Managing such a table can be enormously complex. We found it was possible to create a relatively simple and concise user-interface for developing translation tables.



**Figure 2 User Interface for Lookup Table Definition**

As illustrated in Figure 2, the user can swap the pin order or set pins to 0 or 1, can invert polarity of individual pins, can select the bus width as a subset of the physical bus width, invert the bus width or bit order, and select from sign-magnitude or one's complement representations. Additionally, the decisions can be numerically offset (toward  $+\infty$  to compensate or create decision threshold imbalance, or away from 0 for translating to two's complement or adding weight to decisions) and be scaled and clipped. Our implementation translated to sign-magnitude format for internal use. The internal format could then be translated for an output bus, as a translation service that allows us to connect one vendor's receiver with another vendor's external frame synchronizer and decoder.

The versatility of this interface has been used for configuration checkout, to verify input and output signal connections line by line. It has also been used to remap the LSB as a copy of the sign bit (the hard data) to be used as a troubleshooting test point. A systems integration test is an expensive effort and it needs to identify and list most of the incompatibilities, not stop at the first one. Thus, the table generator has compensated for inverted differential signals and other wiring conditions that could have held up a test.

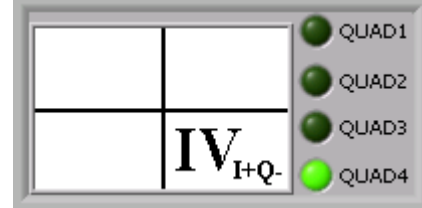
### B. QPSK symbol order

When the I channel transition timing is staggered mid-bit with the Q channel and data bits are alternated between the two channels, there is no sequencing ambiguity when reconstructing the data into a single stream. Yet we encountered a receiver implementation that would sometimes

reassemble the stream with temporally swapped bits, and there is also the case of QPSK that is not staggered, and so it was necessary to provide compensation for this condition of re-ordered symbol pairs in the reassembled stream. The condition is characterized by two correlators simultaneously detecting a match, causing data correlation to be indecisive while clock correlation is stable.

## III. ACHIEVING AND MAINTAINING CORRELATION

In order to perform closed-loop measurements, the transmit data stream and received data stream must be aligned with each other. Closed-loop measurements presume that not only are the received (test) clock and data available, but also the transmitted (reference) clock and data for comparison. Thus, this kind of measurement is usually confined to a laboratory setting and not normally conducted in the field on operational links. With the advent of modern channel simulators it is becoming standard practice to sound real channels or calculate time-varying channel impulse responses for critical scenarios (launch, stage separation, orbit, docking, re-entry) to use as repeatable test-cases.



**Figure 3 Quadrant Tracking and Reporting**

Knowing the receiver alignment is important because it sometimes correlates to performance. An algorithm may favor certain quadrants, the developer may have only tested with (assumed) a specific quadrant, or imbalances may favor certain transmitter-receiver alignments.

### A. Clock Correlation

If the two streams are truly phase-locked to each other, then the clocks should have the same average rate (barring data Doppler in the channel). In this scenario, the rate of clock slips is constrained by the loop bandwidth of the receiver's symbol synchronizer. By persistence-filtering the difference between the reference and test clocks, we could monitor for accumulating slips.

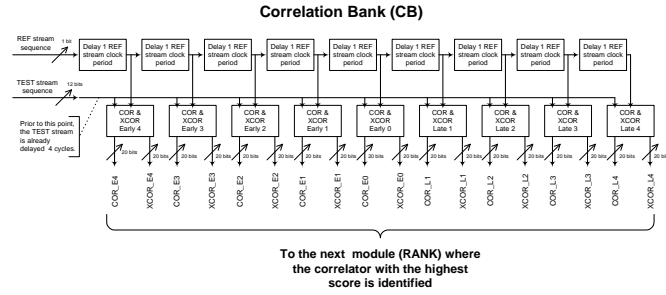
Correlating the reference and test clocks as well as the reference and test data provides two independent measures of slipping. During initial setup and during receiver acquisition testing, the clock correlators provide the sanity check that the clocks are in fact coherent. During testing, the clock correlators confirm that missing or extra symbols are real and not the symptom of a noise-induced correlation error.

### B. Data Correlation and Tracking with Slips and Rotations

The instrument needs to maintain alignment between the test and reference streams when slips and rotations occur, at lower SNR and with more agility than the receiver, decoder, or frame synchronizer under test. We implemented an entire matrix of variable-depth data correlators (Figure 4) to

continuously monitor for constellation rotations and symbol slips.

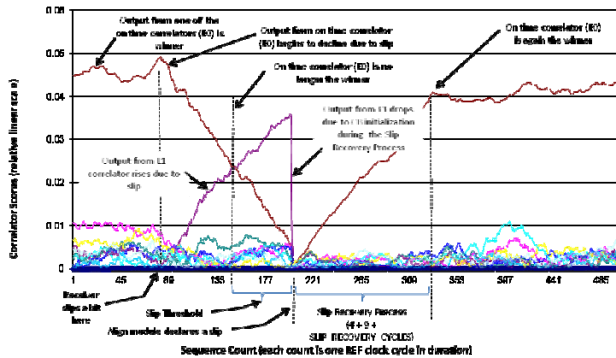
Correlation depths ranging from 16 bits to 512 bits provided different levels of agility. We evaluated two correlation techniques, simple power correlation and a modified algorithm adapted from Massey [3] by Ken Andrews. In this application where the data pattern is not fixed, we found that power correlation was simpler and more effective.



**Figure 4 Continuous-Tracking Data Correlator Bank**

Our FPGA implementation allows correlations to be performed continuously, symbol by symbol. Each interval, the reference (hard) and test (soft) symbols are compared in four quadrants, on-time as well as up to up to four positions slipped in either direction. Voting is used to decide the winning correlation, which is variably persistence-filtered (slip threshold). We found it was unnecessary to develop an acquisition algorithm, as disabling the persistence filter induces a very efficient search for alignment; however, systematic searching can be accelerated by stepping the entire nine-position window. Tightening the filter (“slip threshold”), we were able to track at up to 40% SER (simulated, as this is well below the tracking threshold of most receivers) while still maintaining agility of less than 700 symbols, which is a fraction of a frame and less than the frequency of slipping for an appropriately tight symbol synchronizer loop.

At a 50Mbps data rate, the correlators perform up to 1.4 billion multiplications and 1.8 billion additions per second. To achieve this, we adopted the sign-magnitude internal format so that a multiplication uses a single gate. These figures assume that intermediate product terms are not recalculated and so the complexity does not increase with the window size.



**Figure 5 Realignment Detection and Recovery**

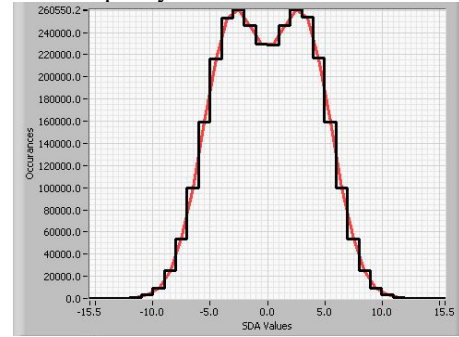
After the scores have been calculated, a voting algorithm

determines the winning correlation (Figure 5). We found that some adjustable filtering is necessary before responding to the vote. We also found that the tie-breaker algorithm was important, following the principle of “hold steady when blinded.” The most probable winner needs to be selected, which is the current alignment if it is a contender.

To accommodate catastrophic events larger than the slip window, a “watch-dog” was implemented to return the correlators to the proper alignment. And of course an instrument that can detect slips, rotations, and glitches, can also collect statistics on the distribution of these events in a given scenario. [6]

#### IV. SOFT DECISION HISTOGRAMS

A receiver has the task of estimating the ideal signal from a noisy and distorted copy of a non-ideal transmission. The receiver can estimate the signal-to-noise ratio using the error-vector-magnitude, the soft decision statistics, or other techniques such as split symbol moment.

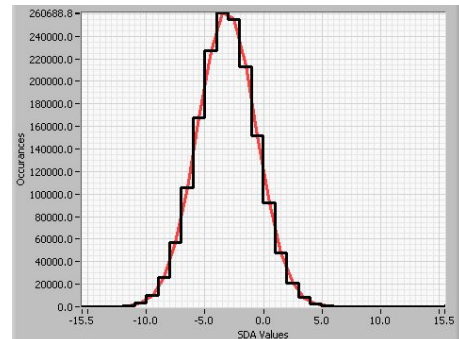


**Figure 6. Example of an Uncorrelated Histogram**

As Figure 6 illustrates, a receiver estimating SNR for very noisy (or very clean) signal conditions is confounded by quantization, clipping of large values, symbol errors, and deviations from the premise of Gaussian white noise.

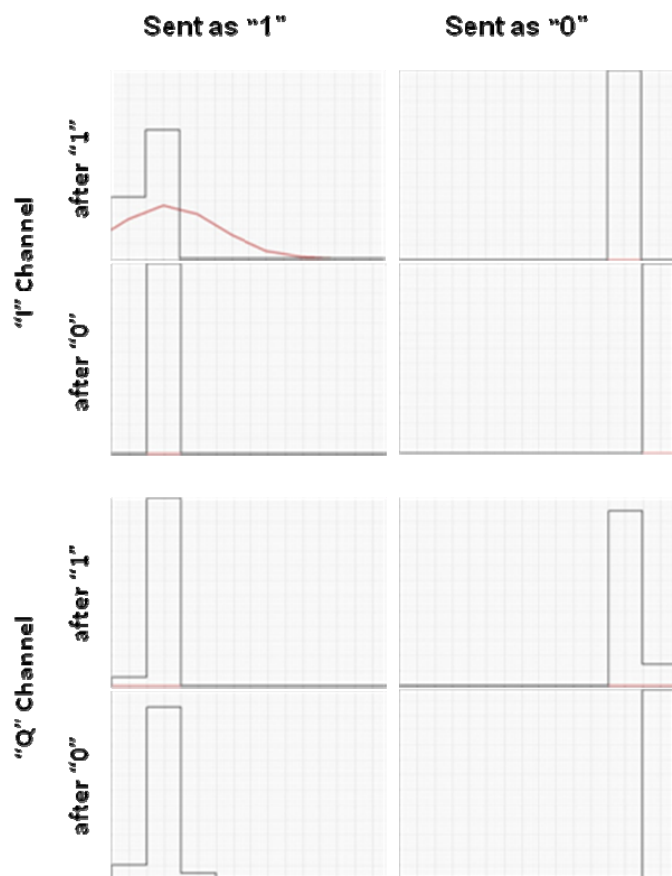
##### A. Correlated and Decomposed Histograms

Using a closed-loop technique produces correlated histograms, as in Figure 7. Since the instrument knows what the ideal signal is (the reference), it can measure the SNR at the output of the receiver, into the decoder, with absolute accuracy. In the figure, the black line shows actual measurements, and the red curve is a fitted Gaussian distribution. This measurement technique has been used to calibrate SNR estimators, and it has been observed that some decoders require an accurate estimate to achieve their performance claims.



**Figure 7. Example of a Correlated Histogram**

But knowing the data sequence allows the instrument to go deeper. We implemented a set of histograms (Figure 8), keeping separate statistics depending on the transmitted symbol polarity, the channel it was transmitted on, and the symbol that preceded it, or activity on the other channel.

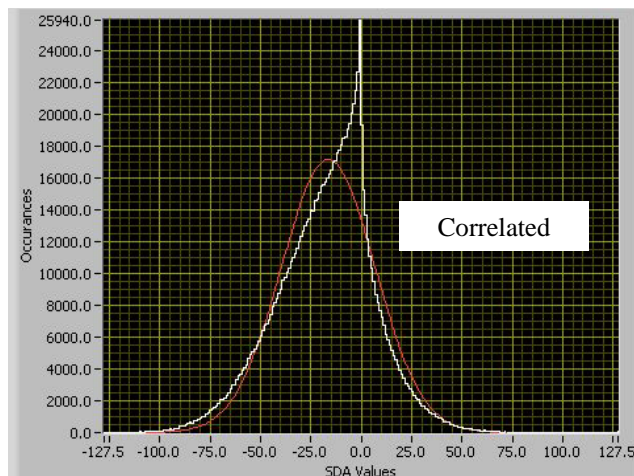
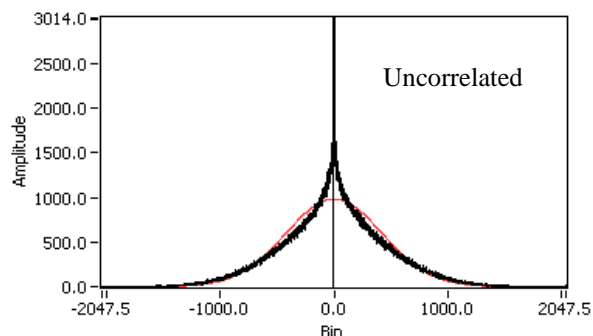


**Figure 8. Example of a Family of Histograms**

In Figure 8, taken from a real RF link, we see that symbols sent as “1” after “0” on the “Q” channel are a little weaker than the others: the zero-to-one transition causes inter-symbol interference (ISI) on the Q channel.

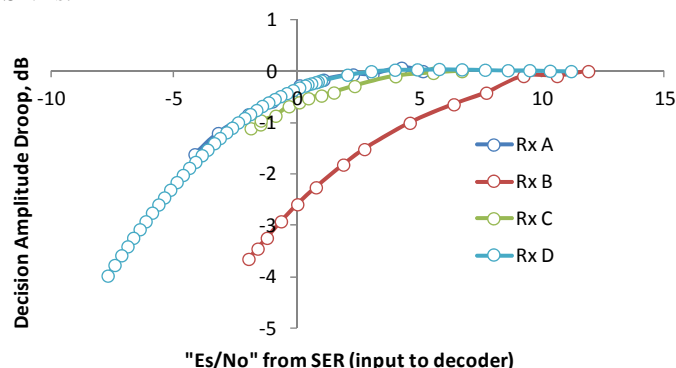
Another example of additional information in a correlated histogram is provided in Figure 9, which depicts a non-Gaussian behavior. In the correlated histogram, we can see that although most of the decisions have very low amplitude, most of them are also correct. This phenomenon was caused by saw-tooth oscillation in the receiver’s symbol scaler.

Decision PDF with Analysis



**Figure 9. Example of Correlated and Uncorrelated Views of an Anomalous Receiver Behavior**

We discovered that by publishing out information from the histograms while running SER performance curves, we had data that could be further analyzed to illustrate and quantify phenomenon such as average symbol amplitude droop at low SNRs.



**Figure 10. Decision Average Droop**

In Figure 10 for example, by further exploring data logged during SER test runs, we can see that different receiver designs use different algorithms to maintain their specified average symbol amplitude in noisy environments. Thus it becomes obvious that an algorithm for scaling symbols to a log-likelihood ratio, for example, cannot assume that the translation depends on the standard deviation alone.

### B. Analysis of Performance

This instrumentation separates the performance of the receiver from the performance of the symbol processor because it directly measures the  $E_s/N_0$  at the input to the symbol processor. This was valuable, as it decoupled the performance of the frame synchronizer, prescaler, and decoder from that of the preamplifier, down-converter, filters, and demodulator and equalizer.

The histograms can be analyzed to produce measurements of decision threshold imbalance, channel imbalance, and inter-symbol interference (ISI) or inter-channel interference (ICI). We produced a measurement in decibels by using the “Q” function to relate each SER to an  $E_s/N_0$  and then differencing. The instrument provides not only the measurement but also



interpretation of the confidence interval around it. Additionally, we used the comparative amplitude and standard deviation in the correlated soft decisions to produce a decibel difference between the sample sets.

This technique was also used for comparing to a baseline, for example to quickly sweep the degradation of receiver performance caused by symbol jitter introduced at the transmitter, for a range of signal-to-noise ratios, jitter rates, modulation indices, and loop parameters. [6] This data would not otherwise have been feasible to collect, and it lead to a new understanding of how to model the effects of jitter. [7]

The histograms can be analyzed in other ways as well. The computational cutoff rate in coding theory is a reflection of the shape of the distribution, and can be applied to identify contributors such as radio-frequency interference (RFI). [8] Also, the relative sample sizes can be used to calculate the data transition density (DTD) and average binary value for the source data.

Receiver performance depends on the quality of the transmitter and also the channel. Components such as phased arrays or switched antenna elements also contribute phase and amplitude dynamics. The ability to decompose situational receiver implementation loss into quantified contributors enables designers to systematically focus on specific countermeasures for specific distortions.

## V. EXTRAS

Some valuable support features can be provided by this platform with relatively little additional investment.

### A. Historical Record

Less frequent events such as symbol slips, constellation rotations, and larger glitches, can be time-stamped and recorded for correlation with causal phenomena observed by other instruments. Maintaining a log file helped the productivity of long runs; for example, to measure performance of a decoder at a discard rate expected to be near  $10^{-7}$ , we could leave instrumentation configured over a weekend.

In such a case, the SDA log files were used to identify a periodic glitch (once an hour-- to the second) that was impeding our ability to perform overnight runs. These glitches were so brief and so infrequent that they were impossible to capture and ascribe, except that the logs made the nature of the glitch completely obvious. The problem was then trapped and quickly traced to a queue overflow in the data generator and several good overnight runs followed.

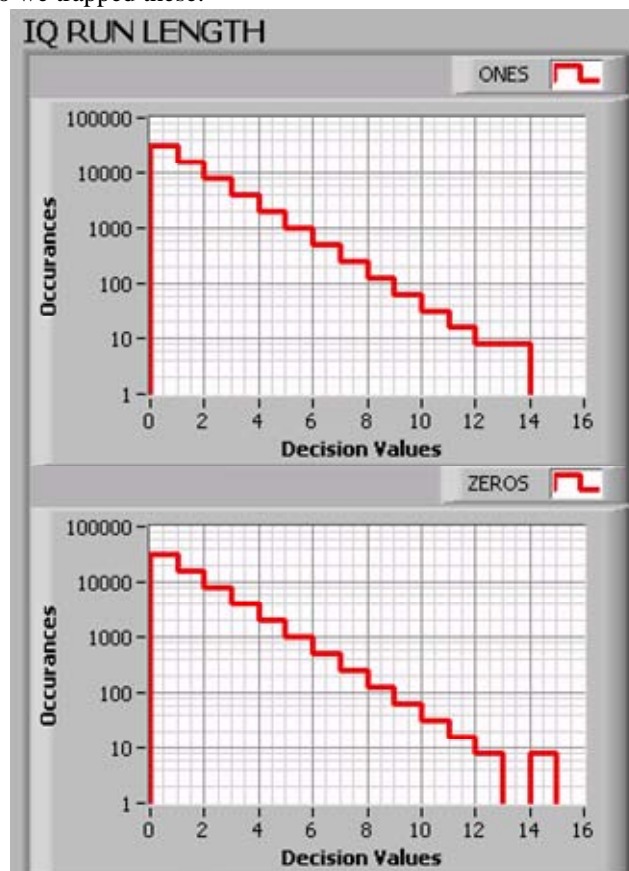
In another case, timestamps in the log file were compared to an oscilloscope to separate decoder losses caused by the operating  $E_s/N_0$  from those caused by unstable receiver lock. A modern oscilloscope was placed into a "fast-frame" mode and returned a file of time-stamps for trigger events caused by receiver loss of lock, which could then be ignored as sources of decoder problems.

### B. Receiver Acquisition Time

Receiver acquisition time requirements are tedious to verify, and good statistical significance is important to assure that durable false-locks do not occur. Independently measuring receiver acquisition times requires an external trigger signal "RF On" marking the time that the signal became available to the receiver. Then the analyzer can efficiently observe the time to symbol synchronizer phase lock, and data correlation, and report the elapsed time in reference clock cycles. This required an extra state machine to govern the otherwise pre-existing instrumentation.

### C. Run Lengths

Run lengths in the reference stream are easily instrumented in an FPGA. Short runs can be analyzed by histogram to determine the randomness of the source data, a quality affecting clock recovery at the receiver. Long runs are significant as they tend to correlate with link disruptions; and so we trapped these.



**Figure 11. Run Length Distribution Measurement**

In the example of Figure 11, the source data stream is a repeating pseudo-random pattern that contains two runs of 12, one run of 13 "1"s, and one run of 14 "0"s.

### D. Channel Simulation

We incorporated a channel simulator that over-writes the soft part of incoming soft decisions. It was invoked to provide sanity checks on symbol processor performance, to isolate performance factors including scaling and binning, ISI, receiver clock recovery, lack of repeatability, and SNR

estimation accuracy. The channel simulator we implemented was crude but served the needs of the developers.

Additionally, our device can rotate the quadrant of the outgoing decision stream. This was used in practice to provide controlled instantaneous quadrant changes for the symbol processor under test, as well as to cycle through and verify each quadrant, or select a specific quadrant when quadrant correction in the symbol processor was not yet implemented.

#### E. Discrete Programmable Test Points

Discrete signals are important for triggering external capture devices like logic analyzers and oscilloscopes. For example, when a slip is detected, an oscilloscope could be triggered and the capture analyzed by a jitter package to provide insight into the phenomenon.

There are many conditions one might use for triggering or otherwise correlating events, and we balanced sophistication with complexity by providing only four test points, but each of these could be used to probe any of a long list of signals: slips, rotations, slips or rotations, clock phase lock, correlation lock, watch-dog events, long runs. Additionally event pulses can be width-encoded to communicate the direction and magnitude of an event, and modern oscilloscope triggers can filter the characteristics of the event. This selectivity enables an engineer to quickly trace the specific causes of individual problems in an operating region where performance is stretched near breaking.

### VI. CONCLUSION

The instrument described here has been used extensively in the deployment of LDPC codes with CCSDS framing and Internet Protocol packets in a TDRSS environment. However, it could also support turbo codes or other data formats used on serial RF links, and could be applied for example to commercial communication satellites links.

#### ACKNOWLEDGMENT

The authors would like to acknowledge Dr. David Ni for his contribution of the computational cutoff rate. Ken Andrews at JPL also contributed many concepts that we considered or adopted.

The concepts for the Soft Decision Analyzer were developed and verified in the Electronic Systems Test Laboratory (ESTL) at NASA's Johnson Space Center, which extensively validated the instrument during actual tests performed for requirements development, technology development, and system prototype integration phases for the Orion Project capsule communication system and the User Service Subsystem Replacement ground station upgrade.

#### REFERENCES

- [1] A. Kaiser, S. Dolinar, M. K. Cheng, "Undetected Errors in Quasi-Cyclic LDPC Codes Caused by Receiver Symbol Slips," IEEE GLOBECOM 2009, Nov. 30, 2009.
- [2] K. S. Andrews, D. Divsalar, S. Dolinar, J. Hamkins, C. R. Jones, F. Pollara, "The Development of Turbo and LDPC Codes for Deep Space Applications," *Proceedings of the IEEE*, Vol. 95, No. 11, November 2007.
- [3] J. L. Massey, "Optimum Frame Synchronization," *IEEE Transactions On Communications*, Vol. COM-20, No. 2, April 1972
- [4] "Low Density Parity Check Codes For Use In Near-Earth and Deep Space Applications," CCSDS 131.1-O-2, Consultative Committee for Space Data Systems, September 2007
- [5] "TM Synchronization and Channel Coding—Summary of Concept and Rationale," CCSDS 130.1-G-1, Consultative Committee for Space Data Systems, June 2006
- [6] C. A. Lansdowne, A. M. Schlesinger, M. K. Cheng, D. K. Lee, "Jitter-Induced Symbol Slip Rates in Next-Generation Ground Segment Receivers," *AIAA SpaceOps 2010*, Huntsville AL USA, April 25-30, 2010.
- [7] M. Srinivasan, A. Tkachenko, M. Lyubarev, P. Estabrook, "Effects of Transmitter Symbol Clock Jitter Upon Ground Receiver Performance," *The Interplanetary Network Progress Report*, NASA Jet Propulsion Laboratory, vol. 42-181, pp. 1-22, May 15, 2010.
- [8] A. Weinberg, "The Impact of Pulsed RFI on the Coded BER Performance of the Nonlinear Satellite Communication Channel," *IEEE Transactions on Communications*, Vol. COM-29, No. 5 pp. 605-620, May 1981.