

## **Appendices**

**Appendix A: AIS 31 type Report for the ZK-Crypt Noise Generator in Deterministic Mode**

**Appendix B: Interacting Blocks in the ZK-Crypt Dual Track FB Circuits – [1S]**

**Appendix C: Interacting Random Controller, Data Churn and Register Bank – [1C]**

**Appendix D: Basic Register Bank with Tiers & nLFSRs – Concept – [00CON]**

**Appendix E: Side by Sides or Concatenated – Double Trouble (for the Gangsters) [34TAND]**

**Appendix F: The Basic Functions- Three Blockbusters – [00OVRV]**

**Appendix G: Changed Message Bit 25, with Splash Rule Vector @ D→A Diffusion**

**Appendix H: Distinguishing Differentials & Observable Correlations- RW Results**

**Appendix J: Other Challenging Features**

**Appendix K: Critical Sub-Graphs**

Figures in appendices B to F are found, annotated and detailed in "The ZK-Crypt Dual Track FB Circuits & Concept Drawings", found on the FortressGB website.

Appendix A is a typical report (in this case annotated) demonstrating the dispersion of Repeated Nibble words generated by the "The ZK-Crypt Noise Generator Design Parameter Emulator", found on the FortressGB website.

Appendix G is a typical demonstration of the diffusive affect of the Data Churn, when Message Bit #25 is fraudulently modified. An extended study of permutations with data diffusing elements, and their potential effect appears in "An Expanded ZK-Crypt III Security Analysis" found on the FortressGB website.

## Appendix A: AIS 31 Report with ZK-Crypt Noise Generator in Deterministic Mode

AN ANNOTATED TYPICAL TEST REPORT GENERATED BY THE ZK-CRYPT DETERMINISTIC/RANDOM NOISE GENERATOR OPERATING IN SINGLE CLOCK MODE (NO RANDOM FM MODULATED OSCILLATOR)

Explanations & Circuit Diagrams in "The ZK-Crypt Noise Generator Design Parameter Emulator"

Statistical AIS-31 Analysis of ZK-Crypt Noise Source Date: 31.12.07;

This particular test is for typical cryptographic applications, where we know that the QTA signal is a "pretty" good pseudo-random, data dependent source of pseudo-randomness.

Test Parameters Sampled Outputs: 10,000,000 Samples; QTA In the S/W emulator we replace random Data with an external Randomization of the (P)Random Slip.

This is a normal test (takes our emulator about 30 seconds).

# of sampled '1's in test nodes: (see Drawings); (P)Random Clk 9218321 the missed pulse generator;  
NO RANDOM CLOCK → frClk 0;

All of the following sampled Results are excellent- PROBABILITY OF 0.5 "1"s  
das 4999256; 4'th Toggle 5000001; Juggle Splash 5000102;

Q8 4998231; Q7 5000576; Q6 4999784; Q4 5000252; Q5 4999784; Q3 4999325; Q2 4999325;  
Q1 4999498; Q0 4999498; A 4999657; C(3) 5000192; L(3) 5000358; L(4) 5000358; B 4999657;  
fff3 5000192; fff4 5000357; fff5 4999657; H 5005900; J 4996518;

These numbers on the right signify the total number of times the nibble occurred in 10M samples-all very close to the theoretical ideal of 156250

Nibble Frequencies of 4th Toggle. 9 tests from Test # 15625 e.g., on test 2 of 9 → 0010<sub>2</sub> = 2 appeared 4 times,  
[ 0] 3 6 5 6 2 2 5 4 5 157766 [ 8] 3 5 9 8 7 3 5 6 9 155590  
[ 1] 8 7 8 5 4 6 5 5 6 155668 [ 9] 3 7 3 4 5 3 2 5 6 152849  
[ 2] 7 4 6 8 4 1 6 4 2 158879 [ 10] 5 5 0 4 4 7 5 3 4 157403  
[ 3] 5 2 4 1 3 8 2 4 4 155184 [ 11] 9 7 6 3 9 8 5 5 3 157236  
[ 4] 7 4 5 4 7 4 9 7 7 156386 [ 12] 4 4 7 4 5 2 5 3 2 154931  
[ 5] 6 1 4 6 7 9 5 3 4 157514 [ 13] 5 4 1 3 5 8 8 6 3 158482  
[ 6] 3 4 6 5 6 8 5 7 1 152967 [ 14] 4 6 4 6 4 4 5 9 10 155506  
[ 7] 4 11 6 6 5 4 4 5 8 156319 [ 15] 4 3 6 7 3 3 4 4 6 157240  
Demerit Results = 10.8 16.8 16.4 10.8 10.0 21.2 9.2 8.4 20.4

A bad Demerit Result we say is 50 or more, a failed statistic is 65 or more. Test #8 was best with a Demerit Result = 8.4

Nibble Frequencies of das Slip Toggle. 9 tests from Test # 15625  
[ 0] 7 4 5 5 7 5 5 7 5 158545 [ 8] 9 4 5 3 4 8 2 2 8 157728  
[ 1] 5 5 5 4 5 5 6 4 7 156095 [ 9] 4 5 4 6 2 5 6 5 3 152628  
[ 2] 5 0 2 9 6 6 8 5 10 155636 [ 10] 7 3 3 4 0 3 4 3 8 157339  
[ 3] 4 6 6 2 8 5 5 4 7 155726 [ 11] 2 5 8 0 6 7 3 9 3 157147  
[ 4] 4 8 1 4 6 3 1 7 3 156879 [ 12] 5 4 6 7 3 3 5 7 6 155859  
[ 5] 7 4 6 4 1 4 8 4 2 157065 [ 13] 8 6 7 3 8 5 5 1 0 154887  
[ 6] 6 6 2 8 7 5 4 5 3 152744 [ 14] 3 5 10 7 4 5 4 7 3 155664  
[ 7] 0 7 5 7 5 4 9 5 5 157014 [ 15] 4 8 5 7 8 7 5 5 7 158964  
Demerit Results = 16.0 11.6 16.0 17.6 18.8 6.4 13.6 12.8 22.0

Nibble Frequencies of Juggle Splash Toggle. 9 tests from Test # 15625  
[ 0] 5 8 4 3 3 2 5 10 5 159381 [ 8] 5 6 4 7 2 4 5 3 1 156199  
[ 1] 5 7 6 8 8 9 5 4 5 155380 [ 9] 1 5 6 1 4 4 3 5 3 152335  
[ 2] 8 2 5 1 7 7 4 7 6 156925 [ 10] 4 4 5 4 4 9 7 5 7 157756  
[ 3] 6 8 4 6 6 7 5 2 6 155547 [ 11] 2 3 5 8 3 0 3 4 7 156467  
[ 4] 3 6 8 7 6 5 4 2 4 156516 [ 12] 6 3 5 4 5 2 9 7 6 155395  
[ 5] 7 6 9 6 2 1 5 4 6 157861 [ 13] 6 3 8 7 10 4 5 6 5 157014  
[ 6] 5 5 2 2 3 6 2 3 2 152315 [ 14] 5 7 2 4 6 6 7 8 4 155529  
[ 7] 4 2 2 8 3 8 5 5 5 155766 [ 15] 8 5 5 4 8 6 6 5 8 159534  
Demerit Results = 11.2 12.0 13.2 18.0 17.2 22.8 8.8 14.4 10.4

Nibble Frequencies of Concatenated String ...||Juggle||4'thToggle||das||Juggle||4'thToggle||das||...  
9 tests from Test # 46875  
[ 0] 10 6 5 5 5 7 6 2 7 470492 [ 8] 3 1 3 6 1 4 2 4 2 470647  
[ 1] 0 7 5 3 3 4 8 4 7 467549 [ 9] 6 5 5 7 9 8 2 2 11 465879  
[ 2] 3 6 5 4 5 7 4 8 4 470587 [ 10] 10 6 5 2 4 2 7 6 2 469551  
[ 3] 7 4 4 9 3 2 4 4 4 468537 [ 11] 2 3 7 3 6 9 5 7 7 467906  
[ 4] 6 4 7 4 14 5 2 6 9 467957 [ 12] 9 3 5 4 6 6 4 1 7 467876  
[ 5] 4 5 8 8 3 5 7 9 2 468758 [ 13] 4 7 5 4 3 3 4 5 2 469729  
[ 6] 2 7 1 9 2 1 6 3 4 466378 [ 14] 1 3 5 5 2 4 9 8 2 466639  
[ 7] 8 3 5 3 8 6 6 2 5 470190 [ 15] 5 10 5 4 6 7 4 9 5 471245  
Demerit Results = 30.0 14.8 7.6 14.4 32.0 16.0 13.6 21.2 23.2

At each clock, the three binary signals are concatenated- so that the ideal number is 156250 x 3 = 468750

This test typically amplifies correlations between sequential binary outputs.

## Appendix A: AIS 31 Report with ZK-Crypt Noise Generator in Deterministic Mode - Continued

The 3 binary signals tested out beautifully, with only one sample run above 50 (55.2) all demerit averages less than 15.1.

```
Demerit Distribution -
Juggle Nibble Test: # FM Warning Triggers (> 50.0) 0
                   # Failed Strings (>65) 0
Count ** 0-25= 29813 ** 26-35= 1350 ** 36-45= 86 ** 46-55= 0 ** 56-65= 0 max= 44.0 av= 14.9
4th Toggle Nibble Test: # FM Warning Triggers (> 50.0) 0
                       # Failed Strings (>65) 0
Count ** 0-25= 29826 ** 26-35= 1360 ** 36-45= 57 ** 46-55= 6 ** 56-65= 0 max= 46.4 av= 14.8
das Nibble Test: # FM Warning Triggers (> 50.0) 1 Groups # 22274
                 # Failed Strings (>65) 0
Count ** 0-25= 29773 ** 26-35= 1337 ** 36-45= 133 ** 46-55= 5 ** 56-65= 1 max= 55.2 av= 15.0
```

The concatenated tests, just about the same- (remember 3 times the number of sampled bits) the BAD FILE (NOT SHOWN HERE) records all of the warning signals.

Here we see that there were 8 occurrences of Demerit Results more than 50, where interval between occurrences was at least 117 test sequences.

```
3 signal Nibble Test: # FM Warning Triggers (> 50.0) 8 Groups # 5302 26213 40475 60445 60562 67894 74322 89896
                    # Failed Strings (>65) 0
Count ** 0-25= 83540 ** 26-35= 9130 ** 36-45= 992 ** 46-55= 84 ** 56-65= 3 max= 60.8 av= 16.9
```

This result shows that there is a slight correlation between the three binary concatenated symbols. Despite the problem that was noticed only

once in less than 1M samples the total average is still an enviable low 16.9, with the worst signal, which appeared once in 30M tests, of 60.8

```
Average Test Cumulative ( 61.6)/4 = 15.4 Max Test Demerit = 60.8
```

The un-weighted average of the four tests is 15.4 – excellent, and the worst test (of 600,000,000 separate statistical measures was 15.4,

the weighted average would be 15.9. The single worst test was 60.8.

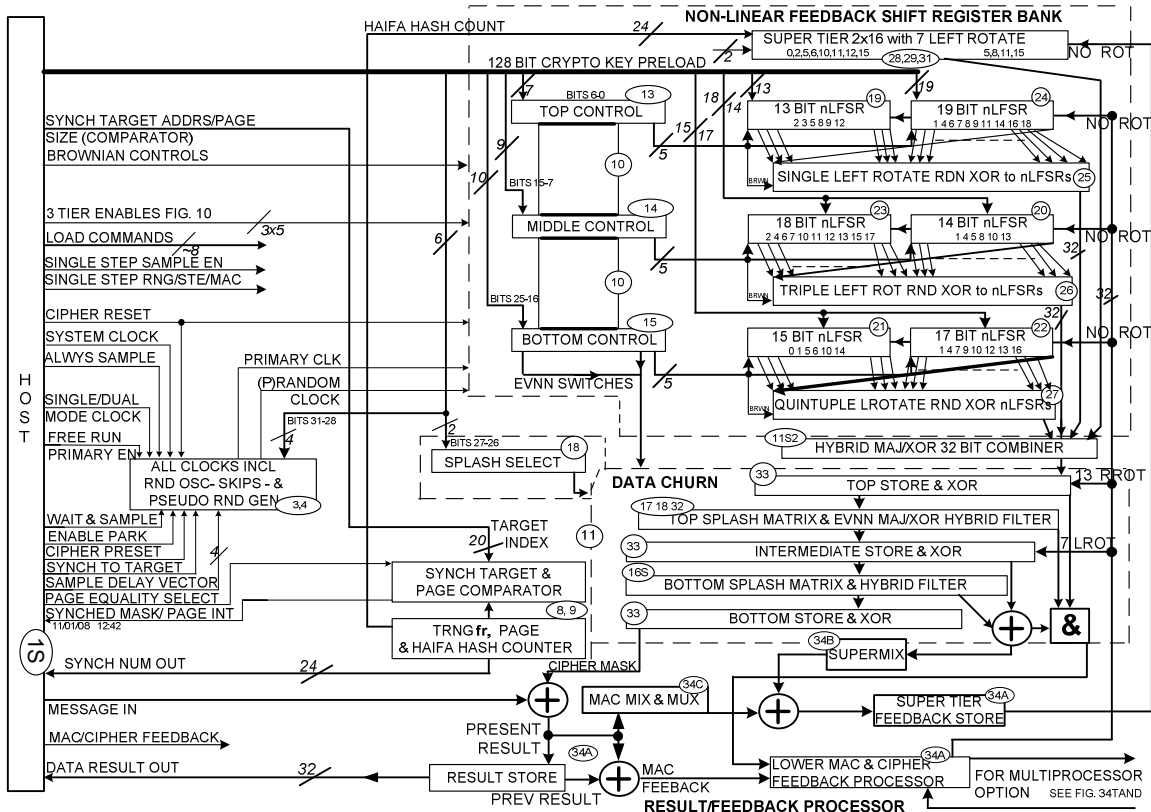
## Appendix B: Interacting Blocks in the ZK-Crypt Dual Track FB Circuits

The left hand side of the figure is the Random Controller which regulates the permutations and clocks of the (right hand top), Register Bank, and the underlying Data Churn.

The Register Bank is composed four tiers; the Super Tier is the great randomizer, masking the combined output of the TMB, Top, Middle and Bottom Tiers.

Each Tier is composed of two unique nLFSRs, which produce a different rotated Image. The Super Tier's output is always the XOR sum of its nLFSRs output and the Image.

The TMB Tiers are randomly clocked, and each pair of nLFSRs' output is randomly XOR summed to its Image, thereby to produce the tier output.



If we fraudulently change one bit on the input Message, we randomly affect 8 or 9 and sometimes 10 output bits from the Register Bank.

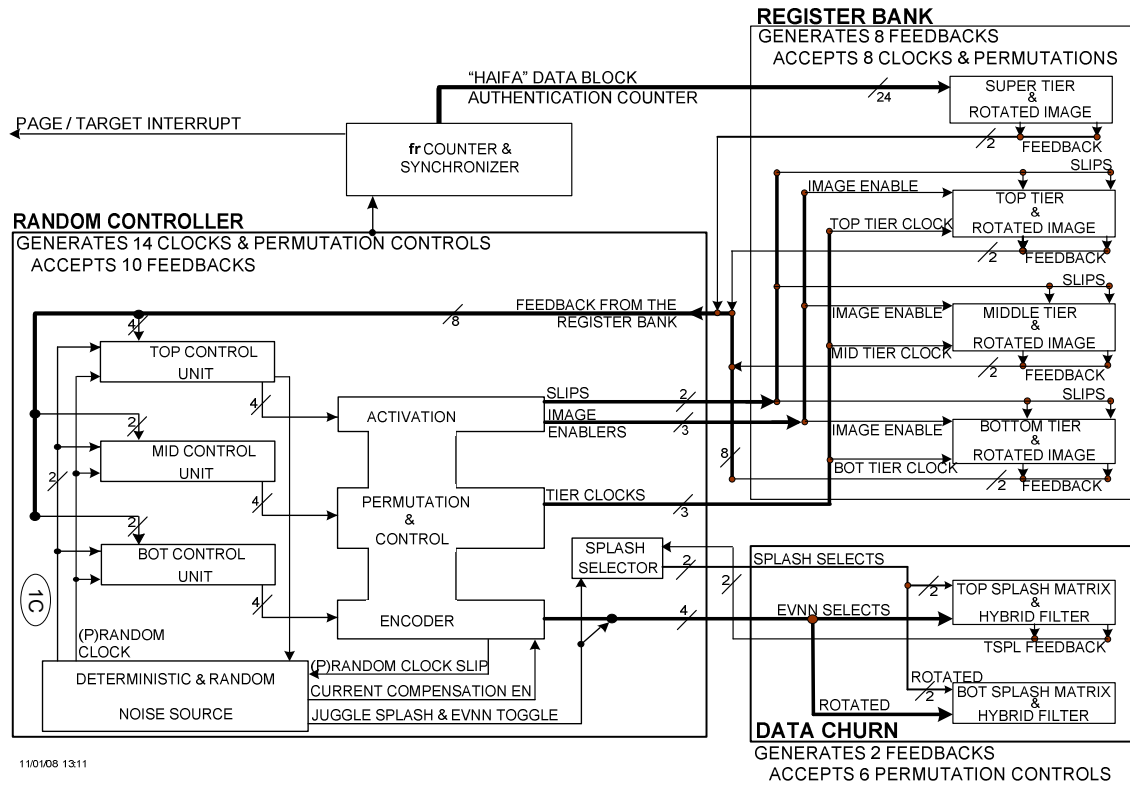
The Data Churn as explained previously, efficiently increases diffusion of the 8 or 9 bits for the Register Bank into typically all of the bit equations of the Cipher Mask.

Provision is made to concatenate any number of ZK-Crypt engines. Alternately, one engine can decrypt a file while a paired identical engine validates the same file.

Encircled numbers "point to" the relevant figures in the "ZK-Crypt Dual Track FB Circuits & Concept Drawings".

## Appendix C: Interacting Random Controller, Data Churn and Register Bank

The Random Controller on the left is "driven" by entropy from its own Noise Source, bottom left and random data signals emanating from the 8 nLFSRs in the four tiered Register Bank, and 2 bits from the output of the Top Splash Matrix.

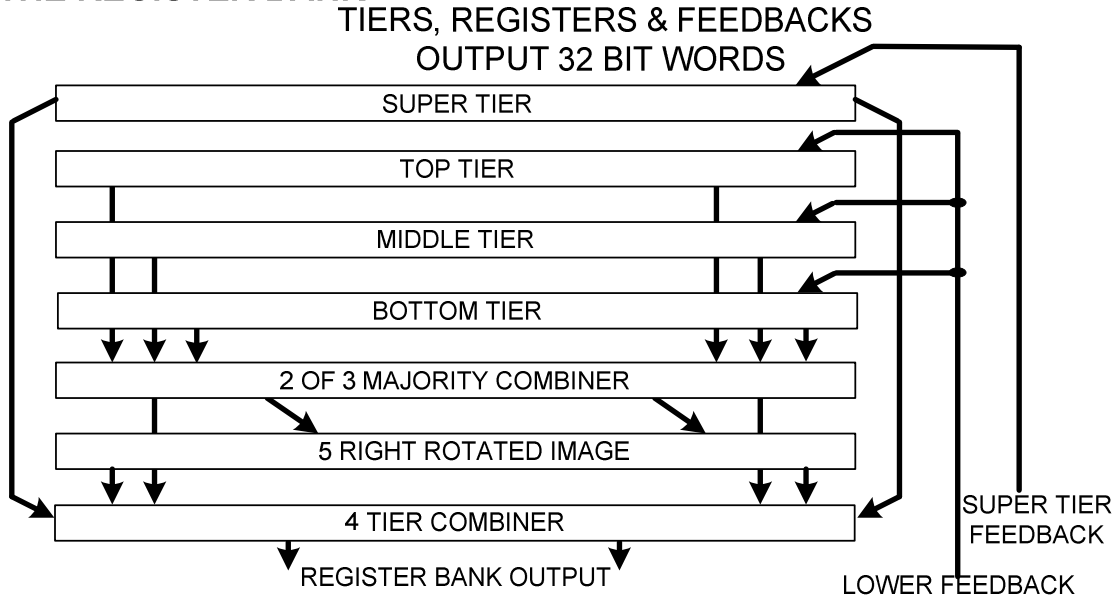


Each of the 10 random signals from the Register Bank affecting the Random Controller and the 14 random clocking and permutation signals from the Controller affecting the Register Bank the Data Churn diffuse into a multiplicity of monomial equations.

## Appendix D: Basic Register Bank with Tiers & nLFSRs - Concept

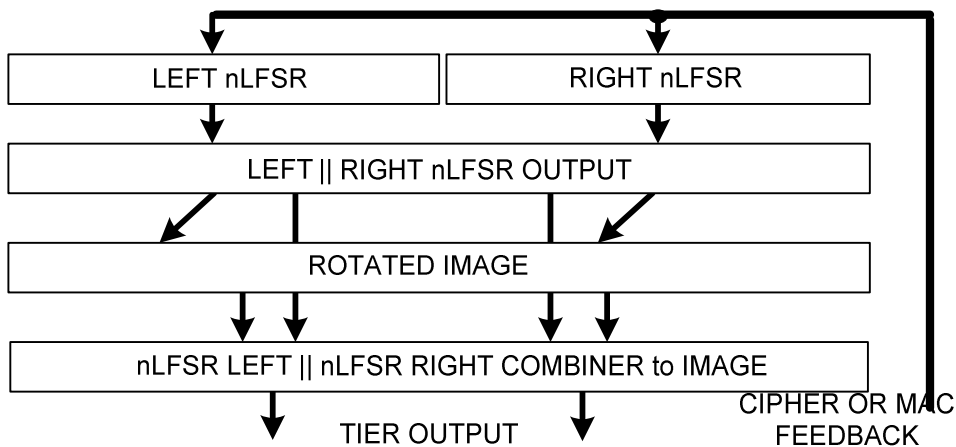
The Register Bank consists of the Top, Middle and Bottom (TMB) Tier unobservable Sanctum Sanctorum; the 3 Tier MAJ Combiner and its Image; and the Super Tier, which serves as a final randomizer of the TMB complex. Virtually no knowledge leaks into the TMB combiner, and whatever might have trickled down is masked by the output of the Super Tier. When randomly clocked, the TMB Tiers "captivate" the Lower Feedback words. (Either two or 3 tiers are clocked at each Host derived Primary Clock.) The Super Tier captivates the Super Tier input stream. In MAC mode the two Feedback streams are orthogonal.

### THE REGISTER BANK



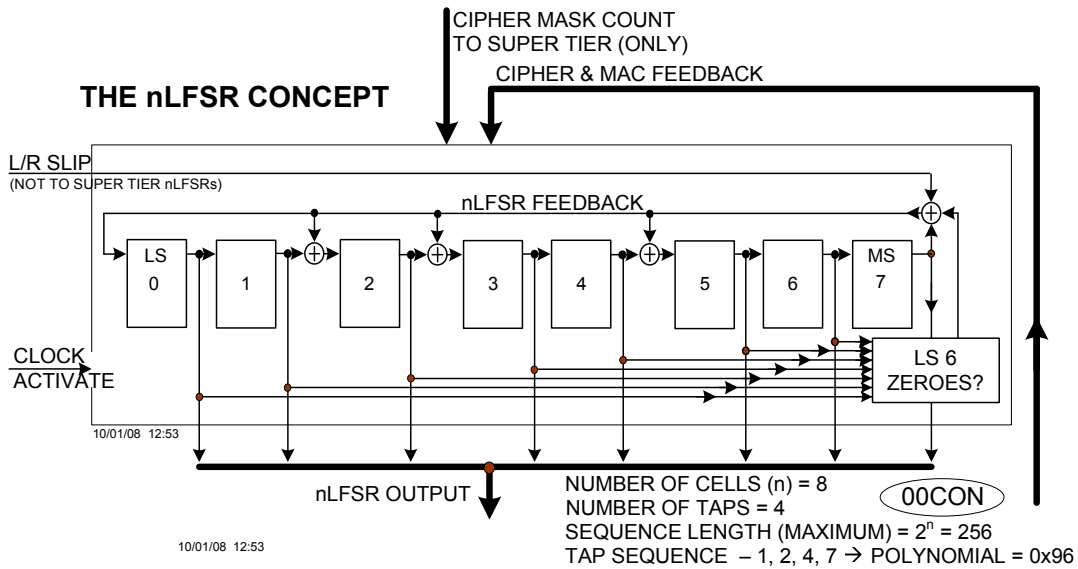
Each tier is composed of two unique nLFSRs. The Imaged output of the nLFSRs may be XOR summed with the nLFSRs output; therefore the TMB Tier output is randomly either the nLFSR pairs' output or the XOR sum of the pair's output and the Image.

### THE TIER CONCEPT



## Appendix D: Basic Register Bank with Tiers & nLFSRs Concept - Continued

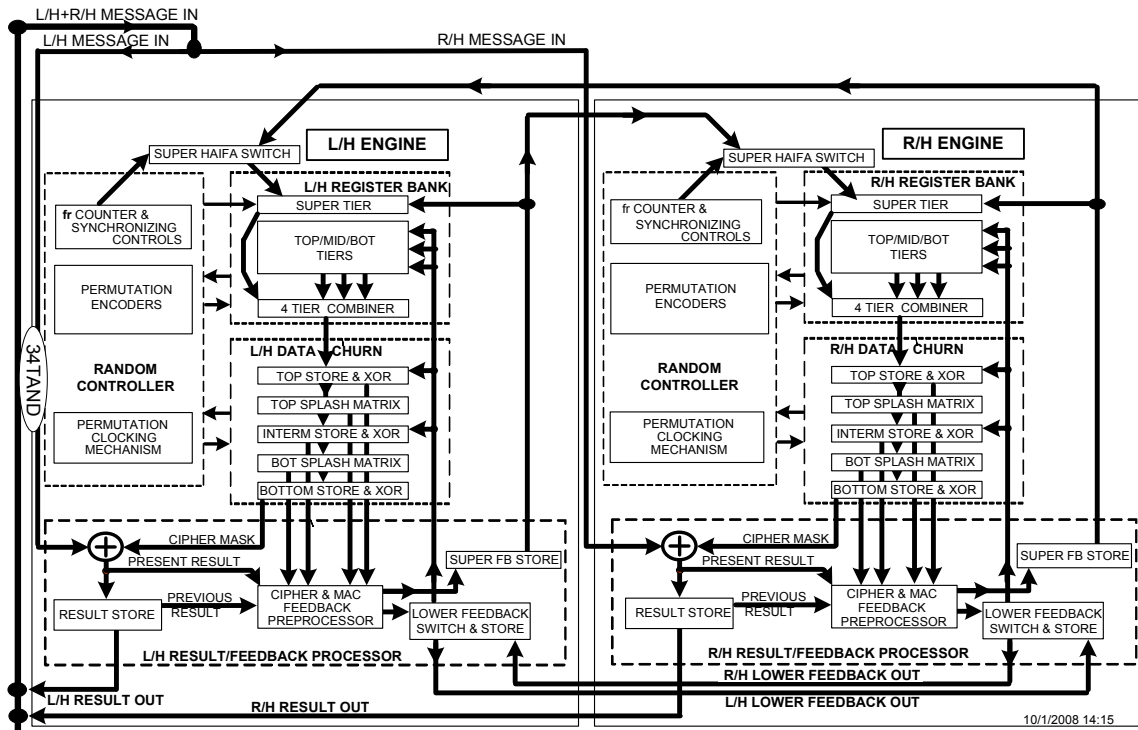
All nLFSRs are One to Many, which generate more local unpredictability than is generated in the oft used Many to One nLFSR configuration, where the shift register cell's outputs are not sampled. In normal Many to One nLFSR operation, a bit moving from "right to left" does not change polarity at each clock cycle, causing a greater degree of correlation between juxtaposed output bits. In these One to Many (Galois) configurations, inter-cell taps fed by random nLFSR Feedback change polarity of the "moving bits", causing additional local entropy. The nLFSR TMB Feedbacks are a function of the last MS R/H cell of the register; and a Random Controller Slip signal, and the NFix n-1 LS zero detector.



When (and only when) an nLFSR is clocked, it "captivates" (XOR) Lower or Super Tier Feedback into every cell.

## E: Side by Sides or Concatenated – Double Trouble (for the Gangsters) 34TAND

Two ZK-Crypt engines standing side by side may be concatenated, operable to increase single function application throughput, or in tandem, operative to simultaneously decipher and authenticate encrypted data.



Two concatenated engines can exponentially raise the complexity of a single engine, subsequently doubling potential unit throughput with double words, and without additional energy expended per processed bit. In such a configuration, The Left Hand Lower Feedback stream is "captivated by the Right Hand TMB tiers and the Top and Intermediate Store & XORs. The Right Hand Lower Feedback is similarly swapped with the Left Hand engine.

In Cipher mode, Right and Left Hand Super Tier feedback is both swapped and captivated locally.

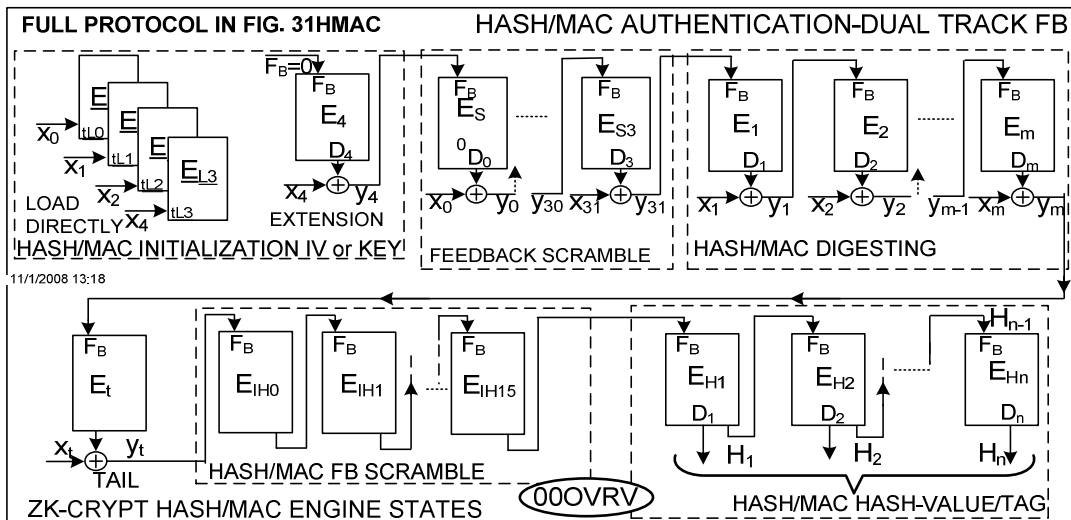
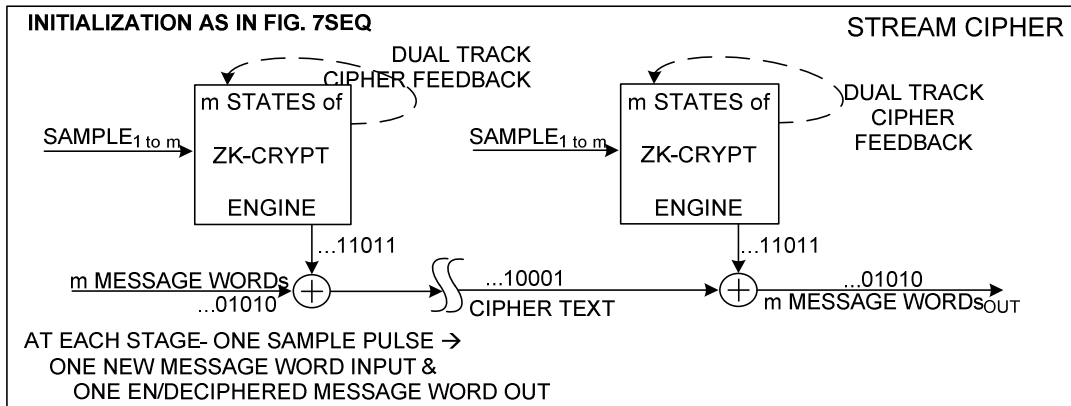
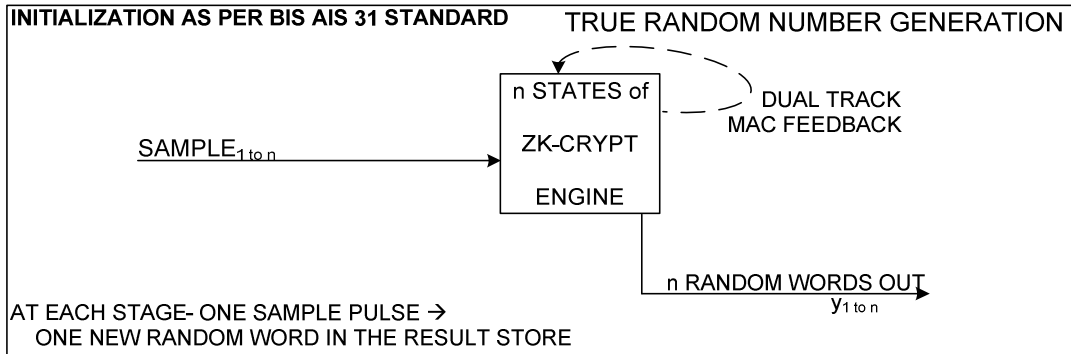
Typically, side by side units will be configured to simultaneously decrypt and authenticate the same data stream, where the left hand and right hand message stream will be identical, with only the decrypting stream outputting a result. In the first stage of initialization, it is anticipated that only deciphering engine will be initialized, as in this configuration we are only processing previously encrypted data.

Both configurations lend themselves to parallelized multi-cored CPUs.

## F: The Basic Functions- Three Blockbusters

The ZK-Crypt's security is centered on the maximum security MAC mode, used in Cipher initialization and True Random Generation in addition to the intended secured Data Authentication.

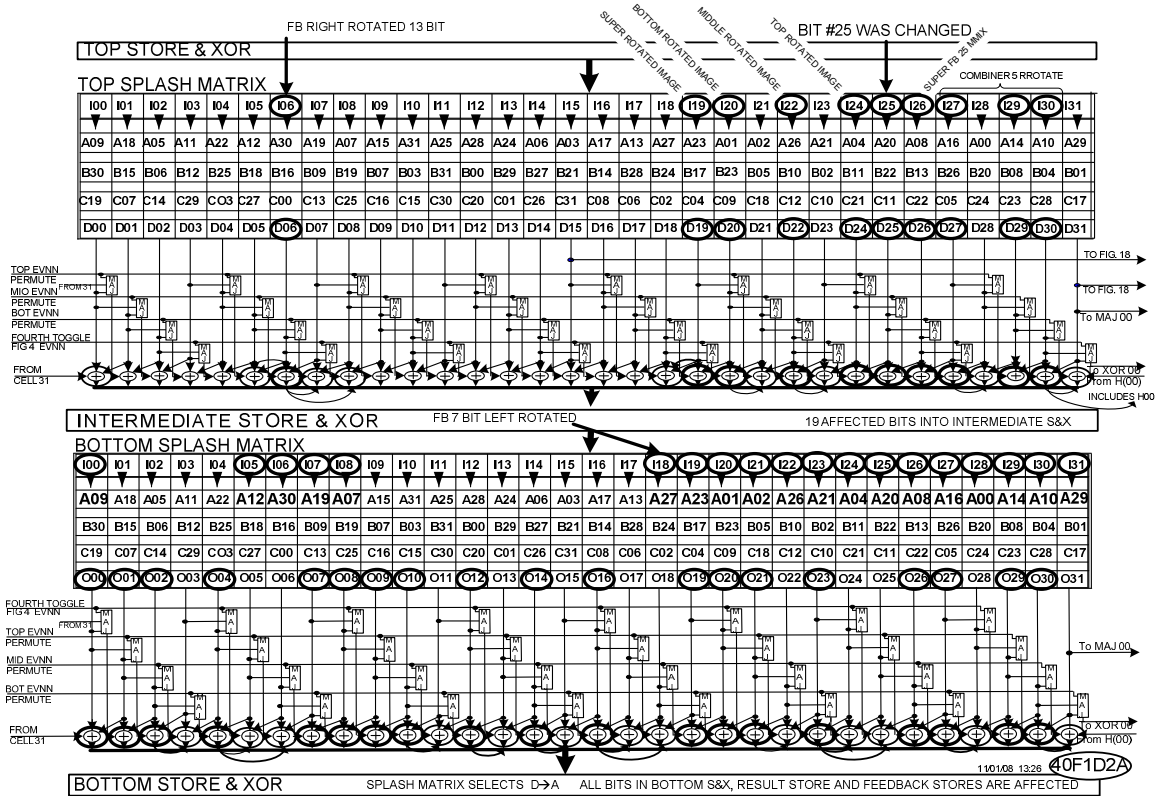
### THE ZK-CRYPT TRIPLE FUNCTION ENGINE



Data Authentication and the Stream Ciphering processes are differentiated by the inclusion of the Cipher Mask and the Message Word in the orthogonal MAC feedback. Neither the observable Cipher Mask nor Message Word may be included in the Ciphering process.

## Appendix G: Changed Message Bit 25, with Splash Rule Selector D→A Displacement

This illustrated case study shows the aberrations in the Data Churn with a Splash Rule Vector setting D→A (Top Splash on Rule D, Bottom Splash on Rule A) and a changed Message bit #25 in the ZK-Crypt. Note that the Register Bank rotated Images "amplified" this one bit change to affect 8 bits of input in to the Top Splash Matrix. A 9'th bit input, the I06 input, is affected by the Rotated Lower Feedback. Similar studies in "An Expanded ZK-Crypt III Security Analysis" include studies of Slip, the jittered TMB clocks, and the EVNN permutations on the Top and Bottom Hybrid Filters in the Data Churn.



As seen, the D vector is the least efficient amplifier. All 32 Bottom Store & XOR (Cipher Mask) outputs are affected, reflected in the Present Result, and, in MAC Mode, by the two orthogonal feedback streams.

## Appendix H: Distinguishing Differentials & Observable Correlations- RW Results

The design of the ZK-Crypt 32 bit Word Manipulator precludes differentials, as the 7 32 bit Word states in the Register Bank and Data Churn output are right and left rotating rings of binary variables; accepting inputs from other rings dependent on right and left rotations and pseudo random displacements. In the three 32 bit stores in the Result/Feedback Processor, two provably orthogonal feedback streams are generated [15], operative to supplement the disparate primitives in the Register Bank, the Data Churn and the Random Controller.

We show that there is no sensed bias (differential) in any binary state variable in the 32 bit Word Manipulator; proven by counting the number of '1's in hundreds of 10M samplings; averaging the results and calculating the standard deviation having counted the number of '1's in every state variable in the Register Bank, the Data Churn and the Result/Feedback Processor.

When necessary, we have created intermediary variables with strong unique correlation, to randomize and remove internal word correlation in juxtaposed intermediate results.

The State Variable RW results, with the exception of the Feedback Stores in MAC and Cipher Mode were virtually identical. In the following we annotate the Locked EVNN RW result with comparisons relating to normal MAC and Cipher Mode.

We now demonstrate the inherent resilience of the ZK-Crypt using the triple locked EVNN RW test syndrome used in the previous "distance" test. We show that the Hybrid Filters "smoothed out" exaggerated input aberrations.

As before, each of the four EVNN control signal affects every fourth output bit in the Hybrid Filter output of the Splash Matrix; there is a strong correlation between every fourth output bit; an average of almost 3/4 of the output bits are of like polarity, and here, with three EVNN signals locked at '1', we will expect an unusually high RW output of the MAJ filter.

The following results in Cipher Mode, where the Super Tier Cipher Feedback is the XORed sum of two apparently correlated words (as opposed to the Super Tier MAC feedback is an XOR sum of 3 possibly correlated words.). In Cipher mode, the Lower Feedback is non-linear and sparse, with an average of 4 '1's; a small pool of probable words generating a huge RW.

## Appendix H: Distinguishing Differentials & Observable Correlations- RW Results (Continued)

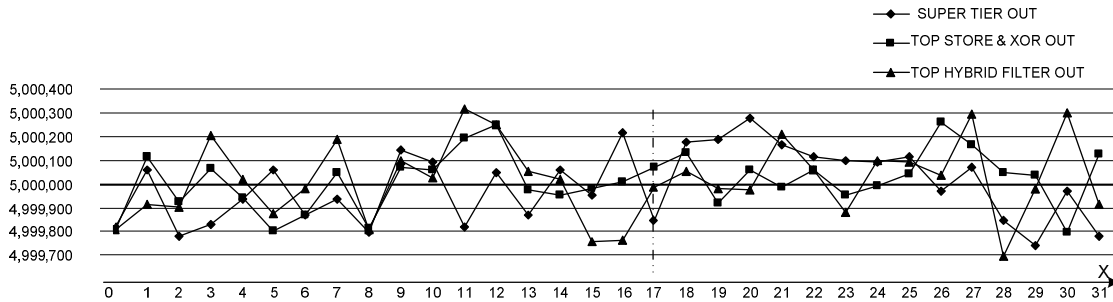
ZK-Crypt III (locked EVNNs) - the normal procedure Cipher Mask Result-	11,630 RWs,
ZK-Crypt III (locked EVNNs) Super Tier Cipher Feedback- the Super Tier Cipher feedback shows considerable trace correlation, In MAC mode calculations the Super Tier Feedback is a complex non-linear function of 3 Data Churn variables and show no detectable trace correlation	12,784 RWs, 11,639 RWs*
ZK-Crypt III (locked EVNNs) Lower Feedback (Sparse Aver 4 '1's) - almost all Lower Feedback words have less than 10 '1's; In MAC mode the Lower Feedback is the XOR sum of the two last Results	<b>8,207,387</b> RWs, 11, 640 RWs.*
ZK-Crypt III (locked EVNNs) Super Tier concatenated nLFSRs out proves to be a good and necessary primitive despite the feedback;	11,638 RWs,
ZK-Crypt III (locked EVNNs) Top Tier concatenated nLFSRs out in the TMB Sanctus Sanctorum. - the high score RW- when the Top Tier clock misses a beat- a word is repeated!!	143,784 RWs,
ZK-Crypt III (locked EVNNs) Middle Tier concatenated nLFSRs out in the TMB Sanctus Sanctorum; see previous comment	143,805 RWs,
ZK-Crypt III (locked EVNNs) Bottom Tier concatenated nLFSRs out in the TMB Sanctus Sanctorum; see Top Tier comment	143,873 RWs,
ZK-Crypt III (locked EVNNs) MAJ Filter output- with mostly strongly correlated '1's.	4,446,490 RWs,
ZK-Crypt III Normal (Unlocked EVNNs) MAJ Filter output- With strong EVNN correlated MAJ outputs..	1,847,870 RWs.*

The TMB Tiers are randomly clock. When a beat is missed, the nLFSRs of the tiers repeat their output- causing and immediate double RW.

\* Results from normal Cipher and MAC mode unlocked EVNN tests

The following graph shows the average number of ones in three significant internal output words of the 3 Locked EVNN test sequences.

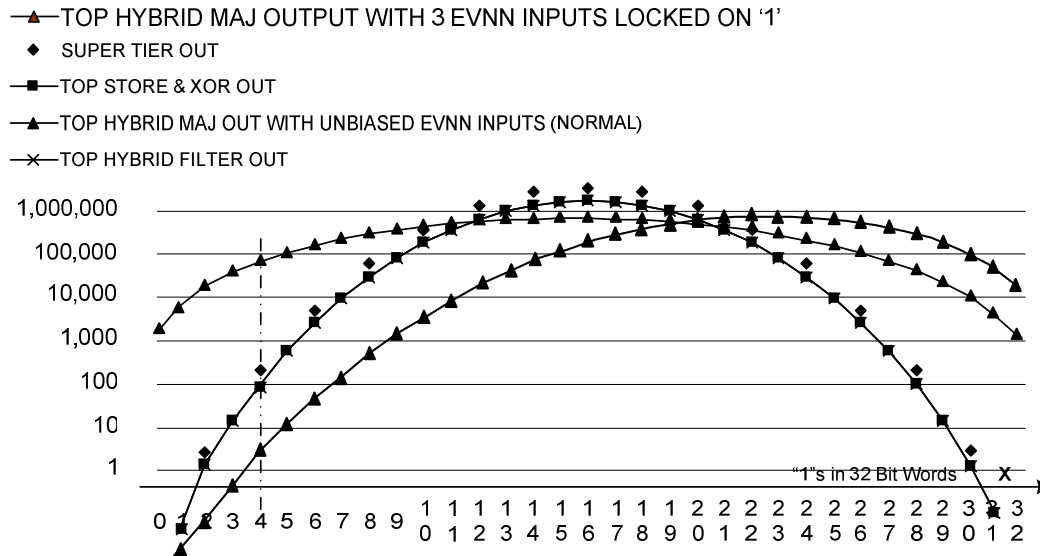
## Appendix H: Distinguishing Differentials & Observable Correlations- RW Results (Continued)



The Average Number of "1"s in Index Position X in 100 Samplings of 10M Rounds;  
 e.g., in Index Position 17, there were an Average of 4,999,847 "1"s in Output of the Super Tier;  
 in Index Position 17, there were an Average of 5,000,072 "1"s in Output of the Top Store & XOR; and,  
 in Index Position 17, there were an Average of 4,999,985 "1"s in Output of the Hybrid MAJ/3XOR Filter.

### No Sensed Differentials in Three Linear Combined Word Sequences

Visibly, we see that the suspect output of the Hybrid Filter leaves us with a string with no sensed differentials, despite the fact that the filter is an XORed sum of two operands each with strong internal correlations. We see that in all of the relevant 100 10M sampled tests, the number of ones is close to 5M. The maximum average deviate bit count value is 5,000,316; e.g.,  $316/(5 \times 10^6)$  which is less than  $10^{-4}$ . Remember, we are only analyzing an infinitesimal part of an assumed  $2^{379}$  binary sequence so that "things happen". The following graph shows the super imposed distribution of the average number of "1"s in significant words taken from the previous tests, with 3 EVNN inputs locked on '1', and the same test wherein all 4 EVNN inputs are uncorrelated and random. Note that a zero or close to zero fraction cannot be represented on this logarithmic graph.



The Average Number of Words with X "1"s with EVNN Signals Locked and Unlocked in 10M Round Samplings  
 e.g., you'd find abt 6.8 Words with 4 "1"s in the output of the Top MAJ Gate with 3 EVNN signals locked on '1';  
 and about 83.2 Words with 4 "1"s in the FFs of the TOP STORE & XOR;  
 and 75,341 Words with 4 "1"s in the Output of the Top MAJ Gate in the Hybrid Filter;  
 and 166.8 words with 4 "1"s in the Output of the Super Tier (an EVNN Vector)

### The Dispersion of Words with '1's with EVNN Signals Locked and Unlocked

## Appendix J: Other Challenging Features

*Highly Irregular Clocking* The highly irregular clocking is a serious issue. (See the large number of RWs in the irregular clocked TMB nLFSR outputs in Appendix H.) The standard if-then-else construction is "if X then Y else Z". Algebraically, this is written

$$XY + Z + XZ$$

and you can verify that if  $X = 1$  then you get  $Y$  and if  $X = 0$  then you get  $Z$ . But the degree of the entire equation is the sum of the degrees of  $X$  and  $Y$ , or of  $X$  and  $Z$ , whichever is worse. The monomial count will be the product of the count of  $X$  and the sum of the counts of  $Y$  and  $Z$ . Thus if each of  $X$ ,  $Y$  and  $Z$ , is even remotely complex, the result formula will be huge.

*Synchronization (HAIFA) Counter XOR with Super Tier Feedback* One would imagine that since the increment function is so simple to conceptualize that it is simple for algebraic attacks to handle. In reality, this is not quite so. If one has a long bit string which being incremented, the carry bits have to look at each previous input, all the way to the least significant. Thus the 31st carry bit will include all 31 previous inputs. In fact, it will be a 31-input AND-gate, and thus a 31 degree polynomial.

*Large Internal State* In sensitive functions minimum internal state of the ZK-Crypt engine consists of a minimum of 384 binary state variables.

The 32 Bit Word Manipulator has 10 32 bit words – made up of:

- the 4 32 bit tiers of the Register Bank- each tier consists of 2 unique nLFSRs;
- the Data Churn – with 3 32 bit Store & XORs – separated by logic Splash Matrices and Filters; and,
- the Result/Feedback Processor with the Result Store, the Lower Feedback Store and the Super Tier Feedback Store, all 32 bit words.

The Word Manipulator feeds 10 signals to the Random Controller; and receives 14 permutation and clocking signals from the Random Controller.

The Random Controller with 3 unique Control Units, a Noise Source and the Random Controller outputs has 64 binary outputs.

*The Time-Memory Relationship* In block ciphers, a classical time-memory trade-off can be found in an attack on a one-way function  $f$ . This is derived from [2,19.] For example, let  $f_k(P) = C$  be the encryption of a plaintext  $P$  into a ciphertext  $C$  under a key  $k$ . In a pre-computation step,

$$f_k(n_0), f_k(f_k(n_0)), f_k(f_k(f_k(n_0))), f_k(f_k(f_k(f_k(n_0))))), \dots \text{is calculated,}$$

starting with a random  $\ell$ -bit string  $n_0$  for a length of  $2^{\ell^2}$ . Only the last value is stored – call these terminal values – along with the starting point  $n_0$  that lead to each terminal value. This is repeated  $2^{\ell^2}$  times, for a total of  $2^\ell$  computations. The total storage is  $2^{\ell^2+1}$ .

When an actual value of  $f_k(P) = C$  is found, then one starts at  $C$  and computes  $f_k(C)$ ,  $f_k(f_k(C))$ ,  $f_k(f_k(f_k(C)))$ ... and so forth for  $2^{\ell^2}$  steps. At any time, if one of the  $2^{\ell^2}$  "terminal values" is encountered, then start again at the initial value which led to that terminal value and keep repeatedly evaluating  $f_k$ , until one reaches the value  $C$ . The immediately preceding value is  $P$ . At worst, this requires  $2^{\ell^2+1}$  computations.

Thus by investing  $2^{\ell^2+1}$  memory, at  $2^\ell$  computation time as a pre-computation step, we can calculate any pre-image under  $f_k$  in  $2^{\ell^2+1}$  steps instead of  $2^\ell$  steps. This is a great speed-up.

However, in most cases  $2^\ell$  time is infeasible. (Note this process is highly amenable to parallel programming). However, using arguments based on the birthday paradox, one can only partially execute the above attack. If enough  $f_k(P) = C$  are available, eventually one of them will fall into a

"chain" whose initial and terminal value have indeed been stored, and the pre-image of that point can be found.

*Application to ZK-Crypt-III* Since the internal state of the function is 384 bits, having  $2^{(384-1)}$  memory is completely infeasible. Even using the birthday paradox to improve the attack results in requiring about  $2^{(384-2)}$  memory (depending on the method) and this, too, is infeasible. Note that we have not considered the 64 bits of dual track feedback which "squeezes" the 32 Bit Word Manipulator into a close knit difficult to divide block, and have not considered the 24 interacting bits between the Random Controller and the Word Manipulator.

The application to ZK-Crypt III is not to perform this attack on the entire ZK-Crypt III system, but rather to pick some important point in the circuit that forms a function easy to compute in one direction but hard to find pre-images (i.e., a one-way function). This particular "choke-point" then would become the  $f$  and the time could be invested to make the chains and store the initial and terminal values.

### **Appendix K: Critical Sub-Graphs**

Using the edge-cut argument of the graph relating to ZK-Crypt III on Page 5 of [25], we can argue that no sub-circuit or interesting portion of the ZK-Crypt III system could have internal smaller than a certain minimum value. The idea is to have each bit be a vertex and a directed edge of the directed graph should go from a vertex to another vertex if the latter is an immediate function of the former (i.e., inputs to logic gates would be typical edges). Cycles will pass through flip-flops, so the graph is acyclic for any given clock-tick (otherwise the circuit itself would be unstable.)

Grouping some of the circuit to form a one-way function that could be attacked by the time-memory trade-off technique above would partition the graph into two regions, one that is  $f$ , and the remainder. Edges will cross this boundary, because the graph is connected. If the size of  $f$  is small, indeed the attacker can run a time-memory tradeoff, but then the inverse or pre-image could also have been found by hand via pencil-and-paper analysis. In reality, the utility of doing this attack only comes about when  $f$  is a half, or a quarter, or some very significant fraction of the entire graph.

However, this is the balanced minimum edge-cut problem which is known to be NP-Complete [20]. That is to say, identifying a set of vertexes that will form  $f$  such that the vertex count of  $f$  is half that of the whole circuit, while minimizing the number of edges that cross the boundary, is an NP-hard problem. The reason the number of edges must be counted is that if there are  $e$  edges, the attacker must guess  $e$  bits at each clock-tick. In fact, this makes the probability of failure equal to  $2^{-e}$  and so  $e$  must be indeed be kept very small.

There is a theorem in graph theory [21] that says that the smallest such cut is  $|V|^{1/2}$  edges in expectation for a dense graph. One can see that this circuit is dense. Furthermore  $2^{(384-1)} \approx 2^{384}$  and a failure probability of  $2^{-383}$  for a two-clock tick attack is unacceptable.