Hardware Architectures for Novel Low-Power Low Error-Floor LDPC Decoding

Shiva Planjery, Benedict Reynwar, David Declercq, and Bane Vasic

Codelucida, LLC

Contact: planjery@codelucida.com

Website: www.codelucida.com

Challenges in LDPC Design

- Typical Requirements: Rates 0.88-0.94, Lengths -1KB-4KB, Very low error-rates (UBER $\approx 10^{-15}$)

- Error Floor Problem

- High decoder complexity and power usage.

- Read latency issues associated with requirement of soft-information

LDPC: Message-Passing

- Messages which are (quantized) Log-likelihood ratios (LLRs) are passed between variable nodes and check nodes of the Tanner graph.

- Each variable node receives an LLR from the channel which we refer to as *channel value.*

- A variable node update function and a check node update function is used.

State-of-the-Art: Offset-Min-Sum

- Variable node update: Sum the incoming extrinsic messages and the channel value, and quantize (typically need 4-6 bits of precision in the message)

- Check node update:
  - Sign Operation: Product of signs of incoming extrinsic messages
  - Magnitude Operation: Find minimum and second minimum of incoming extrinsic messages and apply scaling factor

FAID: New Decoding Approach

- Finite alphabet iterative decoding (FAID): messages belong to a finite alphabet represented as 0, ±1, ±2, etc.

- Check node update function used is the same.

- Main difference is in the Variable node update (VNU) function.

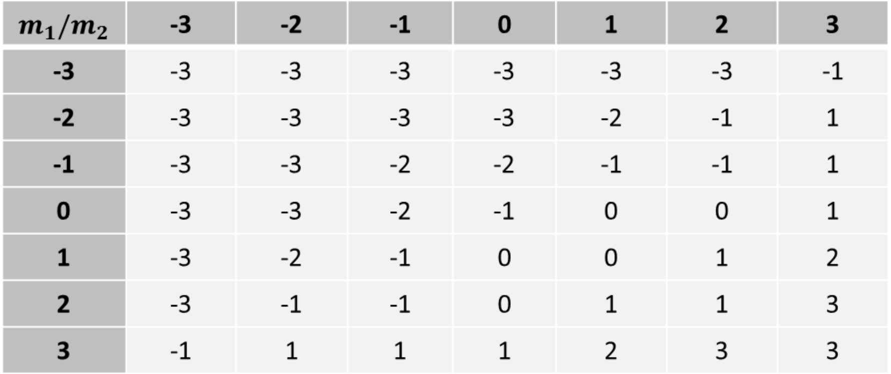
- VNU is a simple map designed to improve error floor performance with only 3-bit messages.

FAID: Hard-decision Decoding

- Y={-C,+C}, is the set of possible channel values.

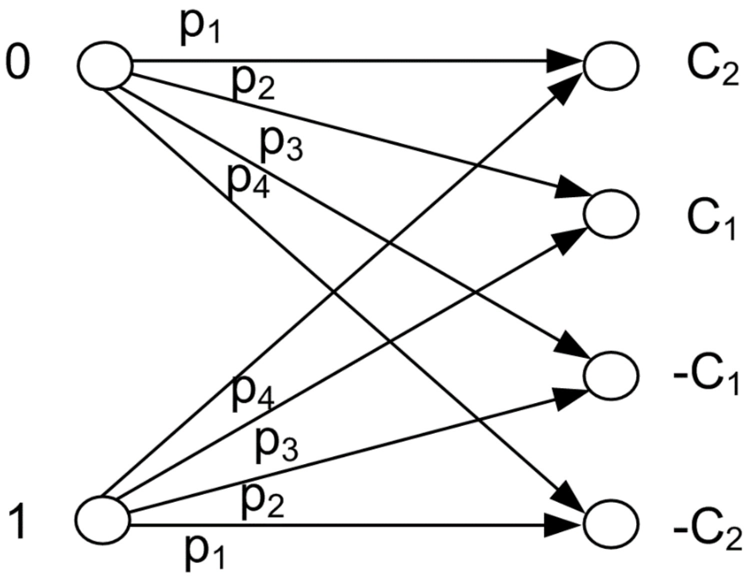
- The VNU is a (d_v-1)-dimensional map or look-up table (LUT), where d_v is the column-weight.

- For d_v = 3, it is a 2D LUT (below). For d_v = 4, it is a 3D LUT.



m_1/m_2	-3	-2	-1	0	1	2	3
-3	-3	-3	-3	-3	-3	-3	-1
-2	-3	-3	-3	-3	-2	-1	1
-1	-3	-3	-2	-2	-1	-1	1
0	-3	-3	-2	-1	0	0	1
1	-3	-2	-1	0	0	1	2
2	-3	-1	-1	0	1	1	3
3	-1	1	1	1	2	3	3

FAID:Soft-decision Decoding(2-bit)



- 2-bit quantized AWGN – thresholds chosen to maximize mutual information
- The set Y is now Y={-C_2,-C_1, C_1, C_2}.
- Messages are still 3-bit messages.
- For $d_v = 4$, we now need to design two 3D LUTs, one for ±C_1 and for ±C_2
- They are chosen to optimize for both waterfall and error floor performance.

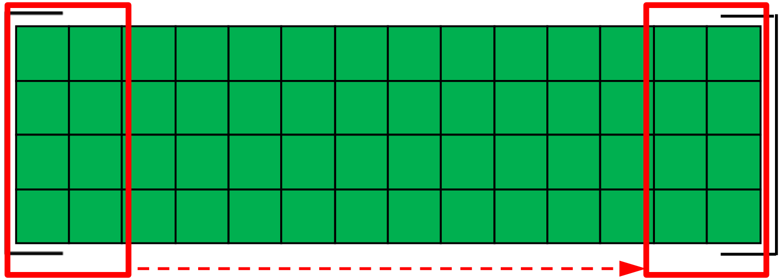
FPGA Implementation: Results

- Code Properties: Quasi-cyclic column-weight four codes. Parity check matrix is divided into blocks of circulants of size L=140.
  - N=1KB, R=0.94: 4 blocks of rows and 64 blocks of columns
  - N=1KB, R=0.91: 6 blocks of rows and 66 blocks of columns
  - N=2KB, R=0.883, 16 blocks of rows and 136 blocks of columns
- Decoding Architecture: Vertical Layered Decoding

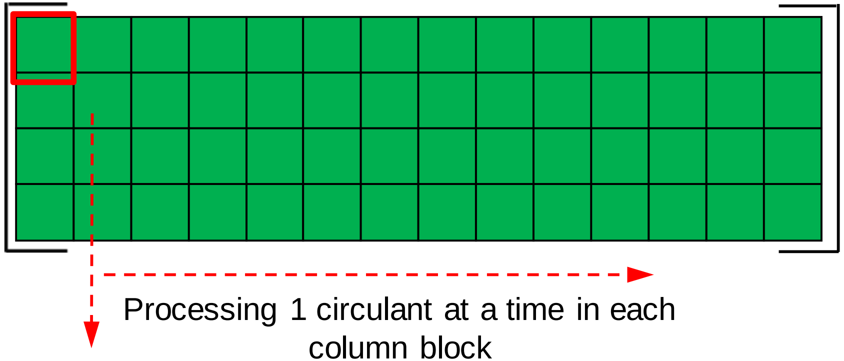
FPGA: Vertical-layered Decoding

- Sequentially updating messages across columns of the parity-check matrix.

- Processing 1 whole column block(s) vs 1 circulant per clock-cycle – impact on resource usage and throughput.

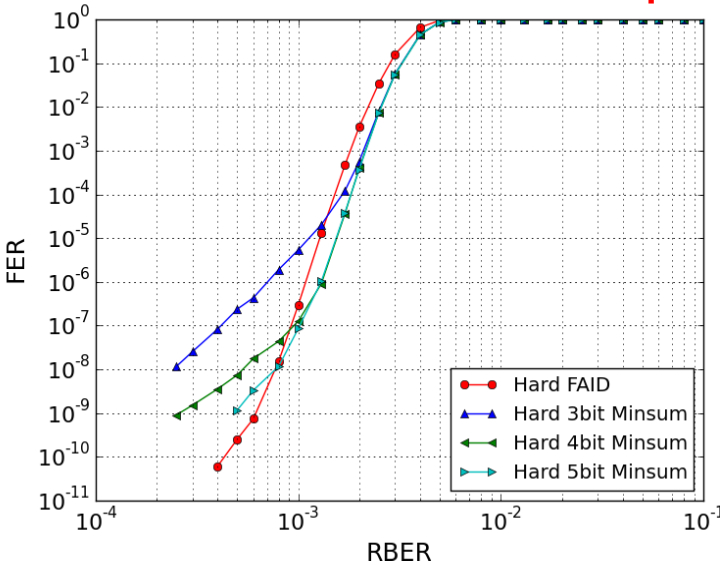


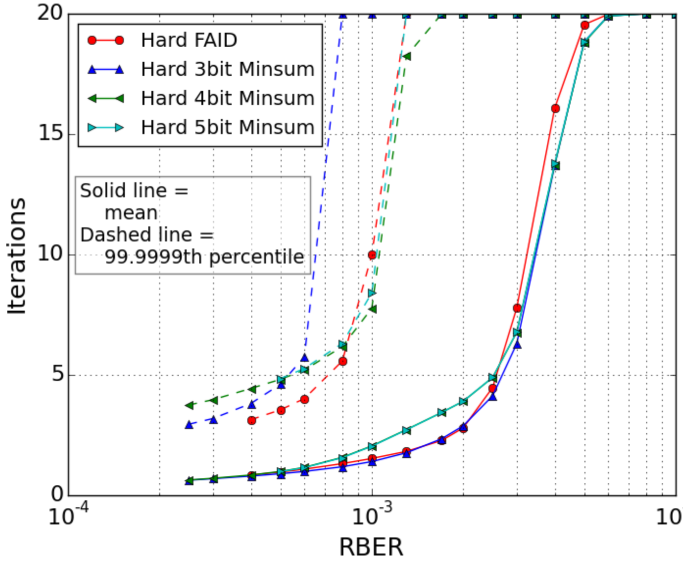
Processing P column blocks at a time (in this case P=2)



Processing 1 circulant at a time in each column block

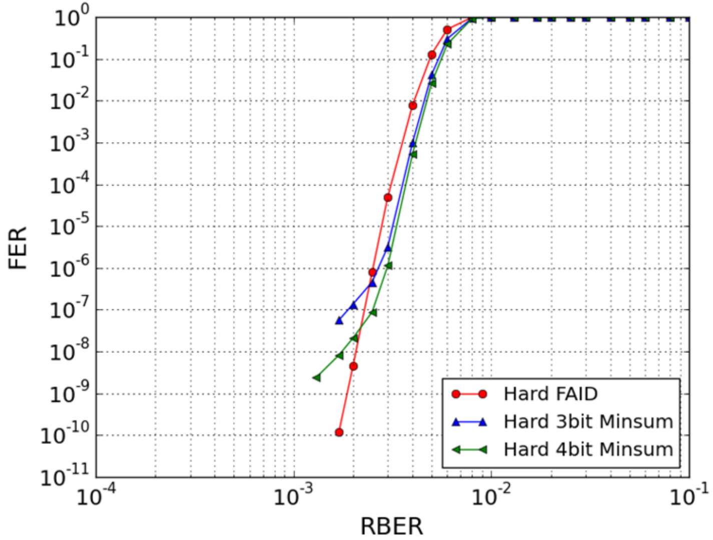
FPGA: Frame Error-rate and Latency vs RBER R=0.94, N=1KB quasi-cyclic code

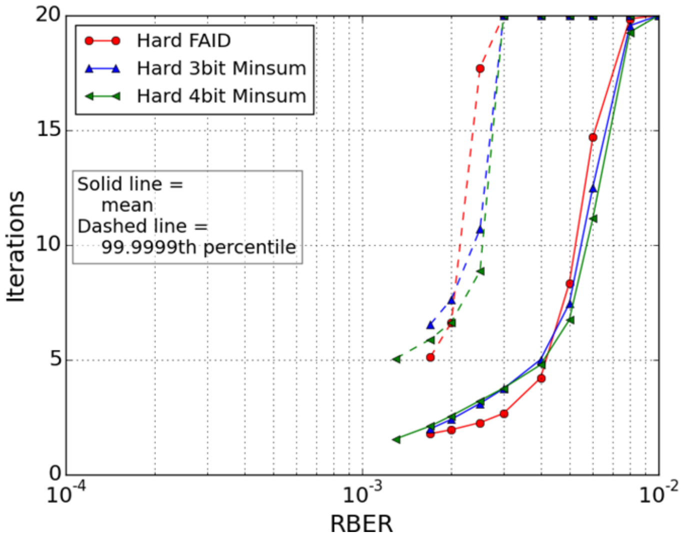




- "Hard FAID" denotes 3-bit FAID decoding on hard-decision channel
- "Hard 3bit Minsum" denotes 3-bit offset-min-sum decoding on hard-decision channel. "Hard 4bit Minsum" and "Hard 5bit Minsum" have similar denotations.
- "mean" latency indicates average n.o. iterations.
- "99.9999% latency" indicates the worst-case latency 99.9999% of the error patterns.

FPGA: Frame Error-rate and Latency vs RBER R=0.91, N=1KB quasi-cyclic code





- “Hard FAID” denotes 3-bit FAID decoding on hard-decision channel
- “Hard 3bit Minsum” denotes 3-bit offset-min-sum decoding on hard-decision channel. “Hard 4bit Minsum”.
- “mean” latency indicates average n.o. iterations.
- “99.9999% latency” indicates the worst-case latency 99.9999% of the error patterns.

FPGA Synthesis Results: N=1KB, R=0.94

Table 1 Processing 1 block column (4 circulants) per clock cycle at clock frequency of 100MHz *

Decoder	Power	Logic LUTs	FFs
3-bit Offset Min-sum	1.014	137,117	104,498
4-bit Offset Min-sum	1.688	198,686	147,197
5-bit Offset Min-sum	1.987	309,795	221,186
3-bit FAID (Hard-decision)	1.09	143,984	104,498
3-bit FAID (2-bit Soft-decision)	1.371	148,855	104,497

*The results were synthesized using the Vivado design suite on a Xilinx Virtex-7 XC7VX690T evaluation board.

- 3-bit FAID (Hard-decision) provides a savings in LUTs of **27.5%** and 35.4% power savings compared to 4-bit offset Min-sum.
- 3-bit FAID (Hard-decision) requires only 5% more LUTs than the 3-bit offset Min-sum with improved FER performance.
- Only a 3% increase in LUTs for FAID from hard-decision to 2-bit soft-decision

FPGA Synthesis Results: N=1KB, R=0.94

Table 2 Processing 2 block columns (8 circulants) per clock cycle at clock frequency of 100MHz

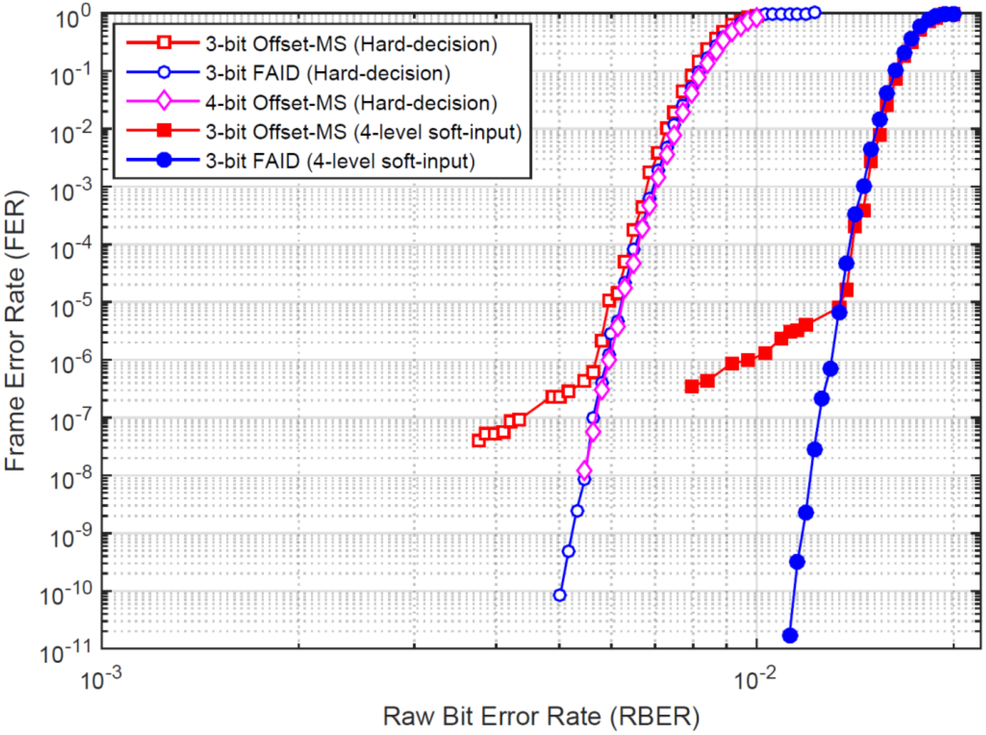
Decoder	Power	Logic LUTs	FFs
3-bit Offset Min-sum	1.014	137,117	104,498
4-bit Offset Min-sum	1.688	198,686	147,197
5-bit Offset Min-sum	1.987	309,795	221,186
3-bit FAID (Hard-decision)	1.09	143,984	104,498
3-bit FAID (2-bit Soft-decision)	1.371	148,855	104,497

*The results were synthesized using the Vivado design suite on a Xilinx Virtex-7 XC7VX690T evaluation board.

- 3-bit FAID (Hard-decision) provides an LUT savings of 29.4% and power savings of 34% compared to 4-bit offset Min-sum.

- 3-bit FAIDs can provide double the throughput of 4-bit offset-min-sum with lesser LUT utilization.

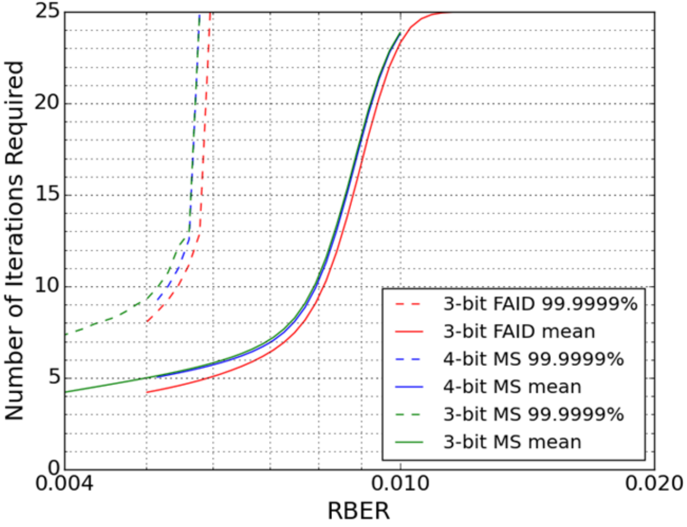
New FPGA Results: FER vs RBER R=0.883, N=2KB (hard-decision and soft-decision)



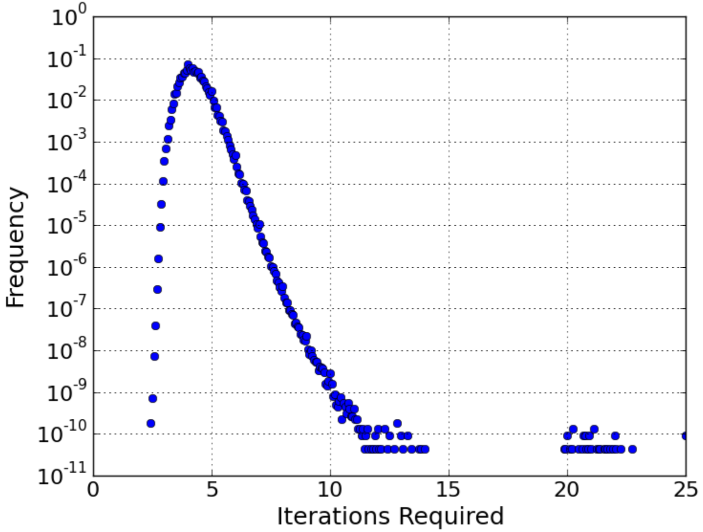
- FAID uses only 3-bit messages (leads to savings in chip area)
- Hard-decision decoding – Optimized for both waterfall and error floor performance.
- Soft-decision decoding – uses only three reads (2-bit precision which is 4-level soft) to reach RBER of 1.1e-2.
- Additional optimization possible to improve waterfall performance of soft-decision decoder.

New FPGA Results: Latency vs RBER R=0.883, N=2KB (hard-decision)

Latency vs RBER



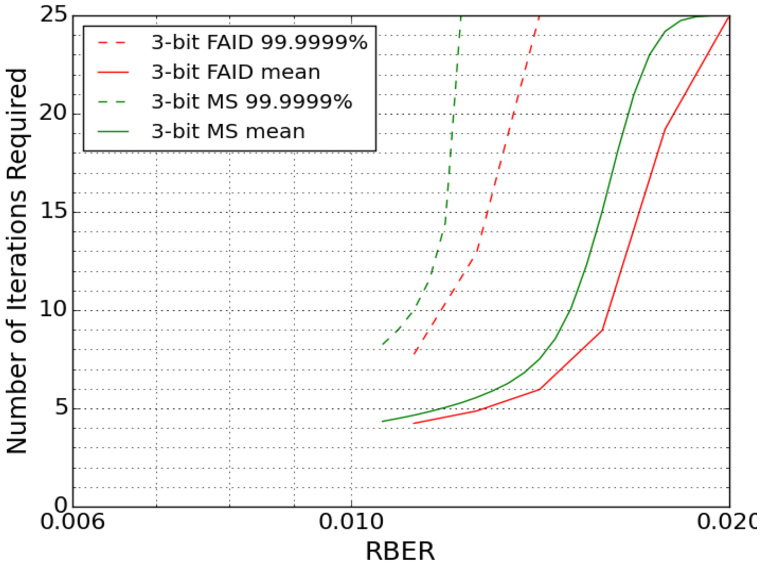
Iteration distribution at RBER=5e-3



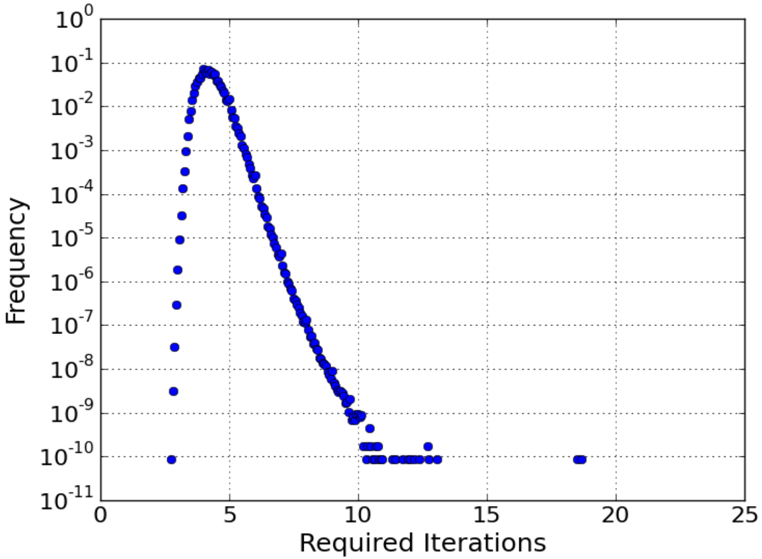
- Iteration distribution at FER=8.96e-11 and RBER=5e-3 shows what fraction of codewords require a certain number of iterations to get corrected.
- Both plots show that 99.9999% of error patterns need only 8 iterations or less at FER=8.96e-11.

New FPGA Results: Latency vs RBER R=0.883, N=2KB (2-bit soft-decision)

Latency vs RBER



Iteration distribution at RBER=1.12e-2



- Iteration distribution at RBER=1.12e-2 shows what fraction of codewords require a certain number of iterations to get corrected.
- Both plots show that 99.9999% of error patterns need only 8 iterations or less .

FPGA Synthesis Results: N=2KB, R=0.883

Table 3 Processing 1 circulant per clock cycle at clock frequency of 100MHz

Decoder	Logic LUTs	LUT RAMs	FFs	BRAMs
3-bit Offset Min-sum	25,715	2,098	18,169	184
4-bit Offset Min-sum	48,865	1,906	35,507	411
3-bit FAID (Hard-decision)	31,123	3,790	18,333	184
3-bit FAID (Soft-decision)	32,485	3,790	18,327	184

- FAID requires 36% less LUTs, nearly 50% less FFs and BRAMs than 4-bit Offset MS. This is also led to nearly 50% savings in power consumption.

- Processing 1 circulant per clock cycle mainly done to fit implementation on current FPGA board.

Conclusions and Future Development

Conclusions

- FAID can provides savings in power with no error floor for both hard-decision and soft-decision decoding.
- FAID achieves RBER=1.12e-2 with R=0.883, N=2KB, with only 3 reads (2-bit soft)

Future Development

- Comparisons with horizontal layered decoding
- Extension of soft-decision results for 1KB and 4KB
- Validation with data from Flash