P2S2

INTERNATIONAL CONFERENCE ON PARALLEL PROCESSING
ICPP/2020/EDMONTON/CANADA
AUGUST 17-20, 2020

acm

BSRNG:
A High Throughput Parallel BitSliced Approach for Random Number Generators

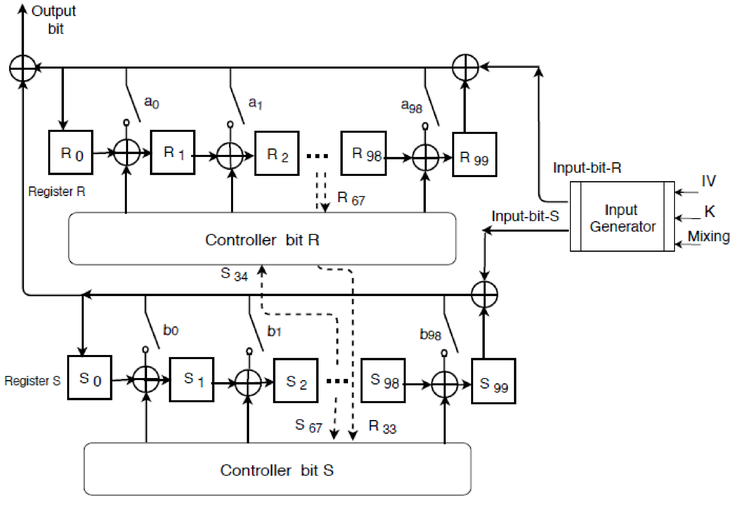
In this work, a high throughput method for generating high-quality Pseudo-Random Numbers using the bitslicing technique is proposed. In such a technique, instead of the conventional row-major data representation, column-major data representation is employed, which allows the bitslicing implementation to take full advantage of all the available datapath of the hardware platform. By employing this data representation as building blocks of algorithms, we showcase the capability and scalability of our proposed method in various PRNG methods in the category of block and stream ciphers. The LFSR-based (Linear Feedback Shift Register) nature of the PRNG in our implementation perfectly suits the GPU's many-core structure due to its register oriented architecture. The proposed implementation successfully passes the NIST test for statistical randomness and bit-wise correlation criteria. Our highest performance among all of the implemented CPRNGs with the proposed method is achieved by the MICKEY 2.0 algorithm, which shows 40% improvement over state of the art NVIDIA's proprietary high-performance PRNG, cuRAND library, achieving 2.72 Tb/s of throughput on the affordable NVIDIA GTX 2080 Ti.

Introduction and Background

PRNG (Pseudo-Random Number Generators), generates numbers that look random, but are actually deterministic, and can be reproduced if the state of the PRNG is known.

CSPRNG (Cryptographically Secure Pseudo Random Number Generators) are PRNGs with properties that make it suitable for use in cryptography.

Applications include Cryptography, High throughput Communications, Physics Simulations, and Optical communications.

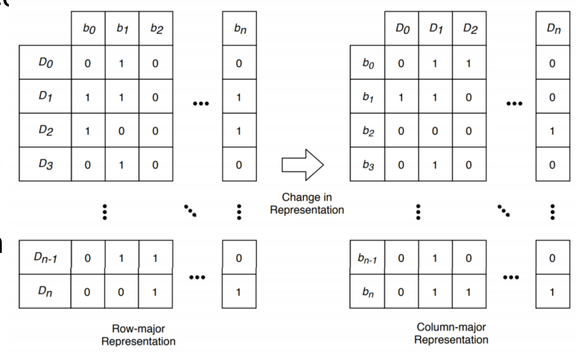


Proposal: Bitslicing For CSPRNG

Proposing **BitSlicing** parallelism within registers to increase the performance of **CSPRNG** algorithms

In the **BitSliced** column-major representation, each register stores the bits having an identical position from all the input data.

PRNG in software algorithm: **LFSR**, **MICKEY**, **AES**, and the **Grain** Stream Cipher



BITSLICING: Pros and Cons

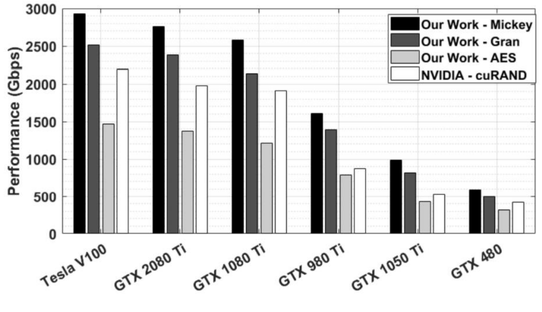
Good: **Logical** Operations (AND/ OR/ XOR/ NOR/ NOT), GF(2) and GF(2^m), Shift and Rotation operations.

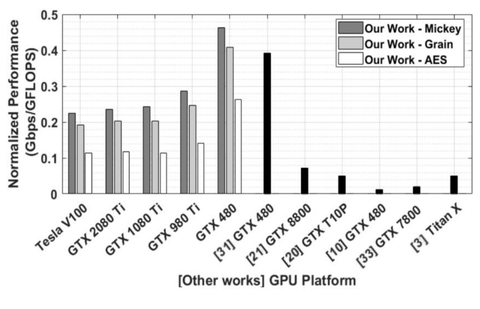
Bad: **Arithmetic** Operations (ADD/ Multiply) Complex Floating Operations

Good News: There is always a **Trade-Off** for different algorithms

Evaluation: Performance/Randomness

The BitSliced MICKEY and Grain Stream Ciphers outperform NVIDIA's cuRAND in terms of bps Throughput. the Proposed PRNG passes NIST Statistical Randomness Test.





Test	P-value	Proportion	Result
Frequency	0.251741	0.9982	Success
BlockFrequency	0.350485	0.9947	Success
CumulativeSums	0.4766135	0.9751	Success
Runs	0.534146	0.9781	Success
LongestRun	0.350485	0.9562	Success
Rank	0.213309	0.9950	Success
FFT	0.534146	0.9971	Success
NonOverlappingTemplate	0.4821360	0.9885	Success
OverlappingTemplate	0.739918	0.9912	Success
ApproximateEntropy	0.350485	0.9721	Success
Serial	0.7227795	0.99982	Success
LinearComplexity	0.739918	0.9840	Success