ASIC Benchmarking of Round 2 Candidates in the NIST Lightweight Cryptography Standardization Process: (Preliminary Results)

Mark D. Aagaard, and Nuša Zidarič
Department of Electrical and Computer Engineering
University of Waterloo, Ontario, Canada
{maagaard,nzidaric}@uwaterloo.ca

This draft report provides preliminary area and clock speed results for 53 NIST Lightweight Cryptography Round 2 candidates on an ASIC cell library.

For the preliminary results, logic synthesis was performed with Synopsys Design Compiler version P-2019.03 using the `compile_ultra` command *without* clock gating. Physical synthesis (place and route) was done with Cadence Encounter v14.13 using a density of 95% and target clock period of 20 ns to minimize area. The ASIC cell library used was ST Microelectronics 65 nm CORE65LPLVT 1.25V. We are currently running additional tests on multiple cell libraries with additional synthesis options and will report those as soon as they are ready.

Cipher instance	**Area (GE)**	**Clock period (ns)**
ACE-v1	9501	2.3
COMET_CI-v1	16920	3.3
COMET_CI-v2	13964	2.0
COMET_CI-v3	16050	3.3
DryGASCON-v1	38134	2.8
Elephant-v1	11740	2.2
Elephant-v2	16458	2.5
KNOT-v1x1	8464	2.1
KNOT-v1x1h	8694	1.9
KNOT-v1x2	9745	2.0
KNOT-v1x2h	10017	2.1
KNOT-v1x4	12321	2.6
KNOT-v1x4h	12692	2.7
KNOT-v2x1	12288	2.1
KNOT-v2x1h	12532	2.7
KNOT-v2x2	14220	2.3
KNOT-v2x2h	14496	2.6
KNOT-v2x4	18034	3.1
KNOT-v2x4h	18412	2.8
KNOT-v3	11249	2.0
KNOT-v3h	11615	2.3
KNOT-v4	14123	2.3
KNOT-v4h	14374	2.5
Oribatida-v1	14513	1.6
Oribatida-v2	12559	1.6
PHOTON-Beetle-v1	17640	3.7
Pyjamask-v1	50356	2.5
Pyjamask-v2	51061	3.1
Romulus-v1	6971	2.6
Romulus-v2	8094	2.6
Romulus-v3	10203	4.0
Romulus-v4	14430	7.7

This work was supported in part by the Canadian National Science and Engineering Research Council (NSERC) and the Canadian Microelectronics Corp (CMC)

Romulus-v5	5312	2.9
SCHWAEMM-v1	23995	5.6
SCHWAEMM-v2	27564	5.5
SpoC-v1	11577	1.9
Spook-v2	18537	2.7
Gimli_GT-v1	13339	3.5
Gimli_GT-v2	15155	3.5
Gimli_GT-v3	18297	4.9
Gimli_GT-v4	18751	8.3
Gimli_GT-v5	23767	5.4
Gimli_GT-v6	26659	8.2
Gimli_GT-v7	34553	10.0
Subterranean-v1	6954	2.1
TinyJAMBU_TJT-v1	3296	2.7
TinyJAMBU_TJT-v2	4241	1.6
TinyJAMBU_TJT-v3	5255	1.9
WAGE-v1	8891	2.1
Xoodyak_GMU-v1	12195	2.9
Xoodyak_GMU-v2	22890	2.7
Xoodyak_GMU2-v1	13510	5.0
mixFeed-v1	22805	2.6

The cipher instance source code and LWC Development Package [1] is identical to that used in the December 2020 version of the FPGA benchmarking report [2]. We are grateful to Kris Gaj and the rest of the Cryptograhic Engineering Research Group at George Mason University for integrating the implementer's source code and LWC Development Package.

Of the 92 instances included in the FPGA analysis, 53 are included here. For unique names and features please refer to Table 2 in FPGA benchmarking report [2]. Some of the submissions were unsynthesizable with Synopsys Design Compiler, and some instances so large that the length of the synthesis run time prevented the work from being included in this preliminary report. Also, the two-pass algorithms are not yet included in our analysis.

The most common reason that code was unsynthesizable was that Design Compiler has very restrictive rules for how signals may be used as array indices. In short, a signal may *not* be used in an array index expression on the right-hand-side of an assignment. On the left-hand-side of an assignment, signals may be used only in very simple range expressions, such as:

```
to_integer(signal) + constant downto to_integer(signal)
```

References

[1] J.-P. Kaps, W. Diehl, M. Tempelmeier, E. Homsirikamol, and K. Gaj. Hardware API for lightweight cryptography. Technical report, George Mason University, Oct. 2019.

[2] K. Mohajerani, R. Haeussler, R. Nagpal, F. Farahmand, A. Abdulgadir, J.-P. Kaps, and K. Gaj. FPGA benchmarking of round 2 candidates in the NIST lightweight cryptography standardization process: Methodology, metrics, tools, and results. Technical Report 2020-1207, IACR, Dec. 2020.