

Benchmarking of Round 2 CAESAR Candidates in Hardware: Methodology, Designs & Results



**Ekawat Homsirikamol,
Panasayya Yalla,
Ahmed Ferozpuri,
William Diehl, Farnoud Farahmand,
Michael X. Lyons,
and Kris Gaj
George Mason University
USA**

<http://cryptography.gmu.edu>
<https://cryptography.gmu.edu/athena>

Outline

- CAESAR Hardware API & the Compliant Code Development
- Overview of Submitted Designs
- Benchmarking Methodology
- Results
- ATHENa Database of Results

CAESAR

Hardware API

CAESAR Hardware API

Specifies:

- **Minimum compliance criteria**
- **Interface**
- **Communication protocol**
- **Timing characteristics**

Assures:

- **Compatibility**
- **Fairness**

Timeline:

- **Based on the GMU Hardware API presented at CryptArchi 2015, DIAC 2015, and ReConFig 2015**
- **Revised version posted on Feb. 15, 2016**
- **Officially approved by the CAESAR Committee on May 6, 2016**

GMU Support for Designers of VHDL/Verilog Code

Implementer's Guide

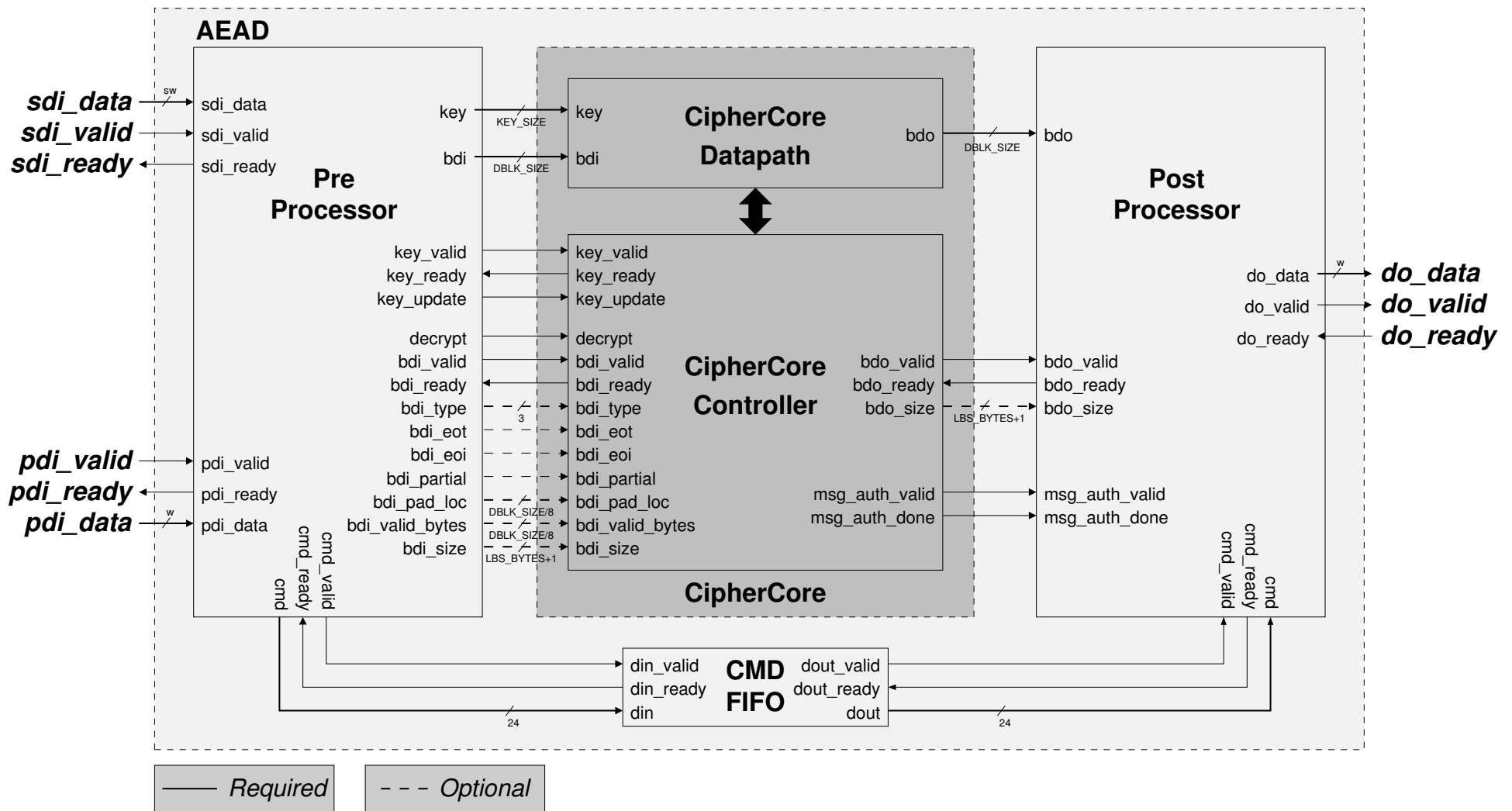
- v1.0 - May 12, 2016

Development Package

- a. VHDL code of generic pre-processing and post- processing units for high-speed implementations (src_rtl)
- b. Universal testbench (AEAD_TB)
- c. Python app used to automatically generate test vectors (aeadtvgen)
- d. Six reference high-speed implementations of Dummy authenticated ciphers

<https://cryptography.gmu.edu/athena/index.php?id=download>

Top-level block diagram of a High-Speed architecture



GMU Support for Designers of VHDL/Verilog Code

RTL VHDL Code

- AES (Enc/EncDec, 10/11 cycles per block, SubBytes in ROM/logic)
- Keccak Permutation F
- Ascon – example CAESAR candidate

Suggested List of Deliverables

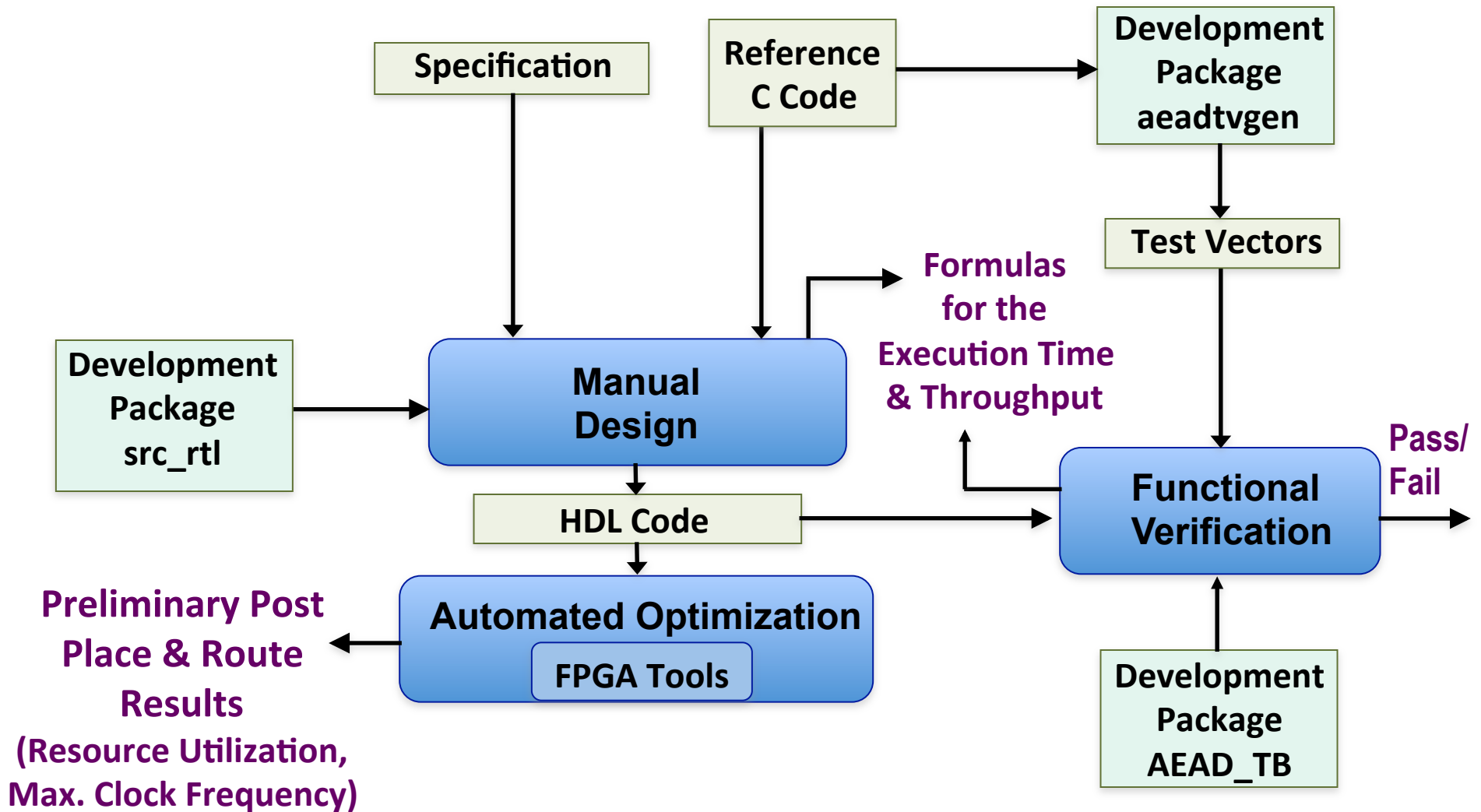
- a. VHDL/Verilog code (folder structure)
- b. Implemented variants (corresponding generics & constants)
- d. Non-standard assumptions
- e. Formulas for the execution time
- f. Verification method (test vectors)
- g. Block diagrams (optional)
- h. License (optional)
- i. Preliminary results (optional)

Known Limitations

- No support for intermediate tags

The API Compliant Code Development

The API Compliant Code Development



Overview of Submitted Designs

Summary of Submitted Designs (1)

Algorithms with:

- **2 Compliant designs + 1 Non-Compliant Design**
1: TriviA-ck
- **2 Compliant designs**
3: ASCON, CLOC, Minalpher
- **1 Compliant Design + 1 Non-Compliant Design**
8: Deoxys, ELmD, HS1-SIV, Joltik, NORX, Pi-Cipher, POET, SCREAM
- **1 Compliant Design**
17: ACORN, AEGIS, AES-COPA, AES-JAMBU, AES-OTR, AEZ, ICEPOLE, Ketje, Keyak, MORUS, OCB, OMD, PAEQ, PRIMATES-GIBBON, PRIMATES-HANUMAN, SHELL, SILC, STRIBOB
- **No Designs**
1: Tiaoxin

Summary of Submitted Designs (2)

Algorithm	Compliant Designs	Non-Compliant Designs	Primary Variants	Variants in Compliant Designs	Variants in Non-Compliant Designs
ACORN	1	-	1	1	-
AEGIS	1	-	1	1	-
AES-COPA	1	-	1	1	-
AES-JAMBU	1	-	1	1	-
AES-OTR	1	-	1	1	-
AEZ	1	-	1	1*	-
ASCON	2	-	2	2	-
CLOC	2	-	3	2	-
Deoxys	1	1	4	1	4

* Authenticator length = 16 bytes, Key length = 384 bits (default), Nonce length = 128 bits 13

Summary of Submitted Designs (3)

Algorithm	Compliant Designs	Non-Compliant Designs	Primary Variants	Variants in Compliant Designs	Variants in Non-Compliant Designs
ELmD	1	1*	4	1	1*
HS1-SIV	1	1	3	1	1
ICEPOLE	1	-	3	1	-
Joltik	1	1	8	1	4
Ketje	1	-	1	2	-
Keyak	1	-	1	2	-
Minalpher	2	-	1	1	-
MORUS	1	-	1	1	-
NORX	1	1	5	4	1

* A variant with intermediate tags

Summary of Submitted Designs (4)

Algorithm	Compliant Designs	Non-Compliant Designs	Primary Variants	Variants in Compliant Designs	Variants in Non-Compliant Designs
OCB	1	-	9	1	-
OMD	1	-	1	1	-
PAEQ	1	-	3	1	-
Pi-Cipher	1	1	8	3*	3*
POET	1	1	2**	1**	1**
PRIMATEs-GIBBON***	1	-	2	2	-
PRIMATEs-HANUMAN***	1	-	2	2	-

* Altogether, the compliant and non-compliant designs cover 4 variants with $|SMN|=0$

** Only a variant without intermediate tags implemented

*** Ciphers belonging to the same family. The 3rd member of this family, APE, not implemented

Summary of Submitted Designs (5)

Algorithm	Compliant Designs	Non-Compliant Designs	Primary Variants	Variants in Compliant Designs	Variants in Non-Compliant Designs
SCREAM	1	1	1	1	1
SHELL	1	-	8*	1	-
SILC	1	-	1	1	-
STRIBOB	1	-	1	1	-
TriviA-ck	2	1**	2	1	1**
AES-GCM	1	-	3	1***	-

* 4 values of d , 2 values of ℓ_{nonce}

** A variant with intermediate tags not supported by the CAESAR API

*** GCM based on AES-128

Effects of Known Limitations

- Current version of API does not support intermediate tags. The implementations of ELmD and TriviA-ck with intermediate tags follow the CAESAR API as much as possible, under this limitation.

Variant vs. Architecture

- Two different variants of the same algorithm produce different outputs for the same input
(e.g., they differ in terms of the key/nonce/tag size)
- Two different architectures of a specific variant produce the same output, but differ in terms of performance and/or resource utilization
(e.g., basic iterative and unrolled x2 architectures)

Architectures

- Majority of algorithms have designs based on
Basic Iterative Architecture (One Round per Clock Cycle)

Exceptions:

- | | |
|----------------------------------|--------------------------|
| ▪ ACORN: | 8bit & 32bit lightweight |
| ▪ ASCON: | Unrolled xN |
| ▪ HS1-SIV: | Folded /2h |
| ▪ Pi-Cipher (by Pi-Cipher Team): | Iterative |
| ▪ Pi-Cipher (by GMU Team): | Folded /4v |
| ▪ SCREAM: | Unrolled x2 |

Submissions with Multiple Architectures

Algorithm	Variants in Compliant Designs	Variants in Non-Compliant Designs	Variant-Architecture Pairs In Compliant Designs	Variant-Architecture Pairs In Non-Compliant Designs
ACORN	1	-	2	-
ASCON	2	-	9	-
SCREAM	1	1	2	2
STRIBOB	1	-	2	-

Architecture Types:

ACORN: 8bit and 32bit lightweight architectures

ASCON: basic iterative and unrolled xN architectures

SCREAM: basic iterative and unrolled x2 architectures

STRIBOB: with and without Miniboxes

Deviations from the CAESAR HW API

Affecting Fairness of Comparison

Deoxys & Joltik (by Axel York Poschmann & Marc Stöttinger)

- No decryption
- Full-block width interface similar to that of CipherCore
- Incomplete support for the CAESAR API Protocol
(no PreProcessor or PostProcessor)

[benchmarked, displayed under HW API: Full-Block width (custom)]

POET (by Amir Moradi)

- Full-block width custom interface
- No support for the CAESAR API Protocol

[benchmarked, displayed under HW API: Full-Block width (custom)]

SCREAM (by Lubos Gaspar & Stephanie Kerckhof)

- Full-block width custom interface
- No support for the CAESAR API Protocol

[benchmarked, displayed under HW API: Full-Block width (custom)]

API Deviations and Other Problems Affecting Benchmarking

Pi-Cipher (by Mohamed El-Haddedy)

- No full support for the CAESAR API Protocol
- No verification using a full set of test vectors
- Large number of clock cycles per block (1782)
[treated as compliant, but suboptimal results]

NORX (by Michael Muehlberghuber)

- Full-block width interface (2 x 768 bits) based on AXI4-Stream
- No support for the CAESAR API Protocol
- A custom wrapper required for implementation using Xilinx ISE and Altera Quartus Prime (not submitted) [not benchmarked]

HS1-SIV (by Sergei Volokitin & Gerben Geltink)

- Full-block width custom interface
- No support for the CAESAR API Protocol
- Code does not pass synthesis using Altera Quartus Prime, or implementation using Xilinx ISE [not benchmarked]

Minor Deviation from the API Compliance Criteria

Keyak (by the Ketje-Keyak Team)

- Compliance criteria:
 - supported maximum size for AD should be $2^{32}-1$ bytes
- Implementation:
 - supported maximum size for AD is 24 bytes

[treated as compliant in the database of results]

Designs with the Highest Potential for Improvement

- **SHELL** by the SHELL Team
 - Preliminary design
 - Throughput to area ratio 130-180x worse than for AES-GCM
- **OMD** by the GMU Team
 - Preliminary design
 - Known improvements possible in the Datapath & Controller
 - Require substantial amount of time to be incorporated
- **Deoxys & Joltik** by the GMU Team
 - Known improvement to the Throughput of Encryption & Authentication (**does not apply to Decryption**)
 - **Estimated improvements shown in the Results Section**

Other Factors Affecting Comparison

- Key sizes
- Security level
(lightweight vs. non lightweight,
single-pass vs. two-pass,
nonce misuse resistance, etc.)
- Nonce sizes
- Tag and/or authenticator sizes
- PDI & DO port width, w

Key sizes

- Majority of ciphers support 128-bit keys only

Exceptions:

- Joltik: 64 & 128
- PRIMATEs: 80 & 120
- AES-JAMBU, Ketje: 96
- Pi-Cipher: 96, 128, 256
- Deoxys, NORX: 128 & 256
- STRIBOB: 192
- AEZ: 384

Possible allowed key ranges:

$$|K| \geq 96$$

- covers all families
- excludes variants with 64 and 80-bit keys

$$|K| \geq 120$$

- covers all families except AES-JAMBU and Ketje
- covers stronger variants of PRIMATEs
- excludes lightweight variants

PDI & DO Ports Width, w

- The CAESAR API Minimum Compliance Criteria allow
 - High-speed: $32 \leq w \leq 256$
 - Lightweight: $w = 8, 16, 32$
- Majority of the API compliant implementations support $w=32$ or 64 only

Exceptions:

- ACORN: 8 & 32
- PRIMATES: 40
- HS1-SIV: 128
- NORX, Pi-Cipher: 128 & 256
- AEGIS, ICEPOLE, MORUS: 256

Ciphers vs. Variants

- Each cipher may have multiple variants, identified by
 - name, e.g., KetjeSr and KetjeJr
 - identifier, e.g., NR-128-64 and NMR-64-64, or
 - a set of parameters.
- PRIMATEs HANUMAN and PRIMATEs GIBBON are treated as separate ciphers, rather than variants (each has its own variants)
- CLOC and SILC are treated as separate ciphers, rather than variants
- In the database rankings, **each cipher is represented by only one variant** with the best value of a particular performance metrics used for ranking (e.g., Enc/Auth Throughput/LUTs, Auth-Only Throughput/Slices, Dec/Auth Throughput, LUTs, Slices, etc.)

Benchmarking Methodology

FPGA Families & Devices Used for Benchmarking

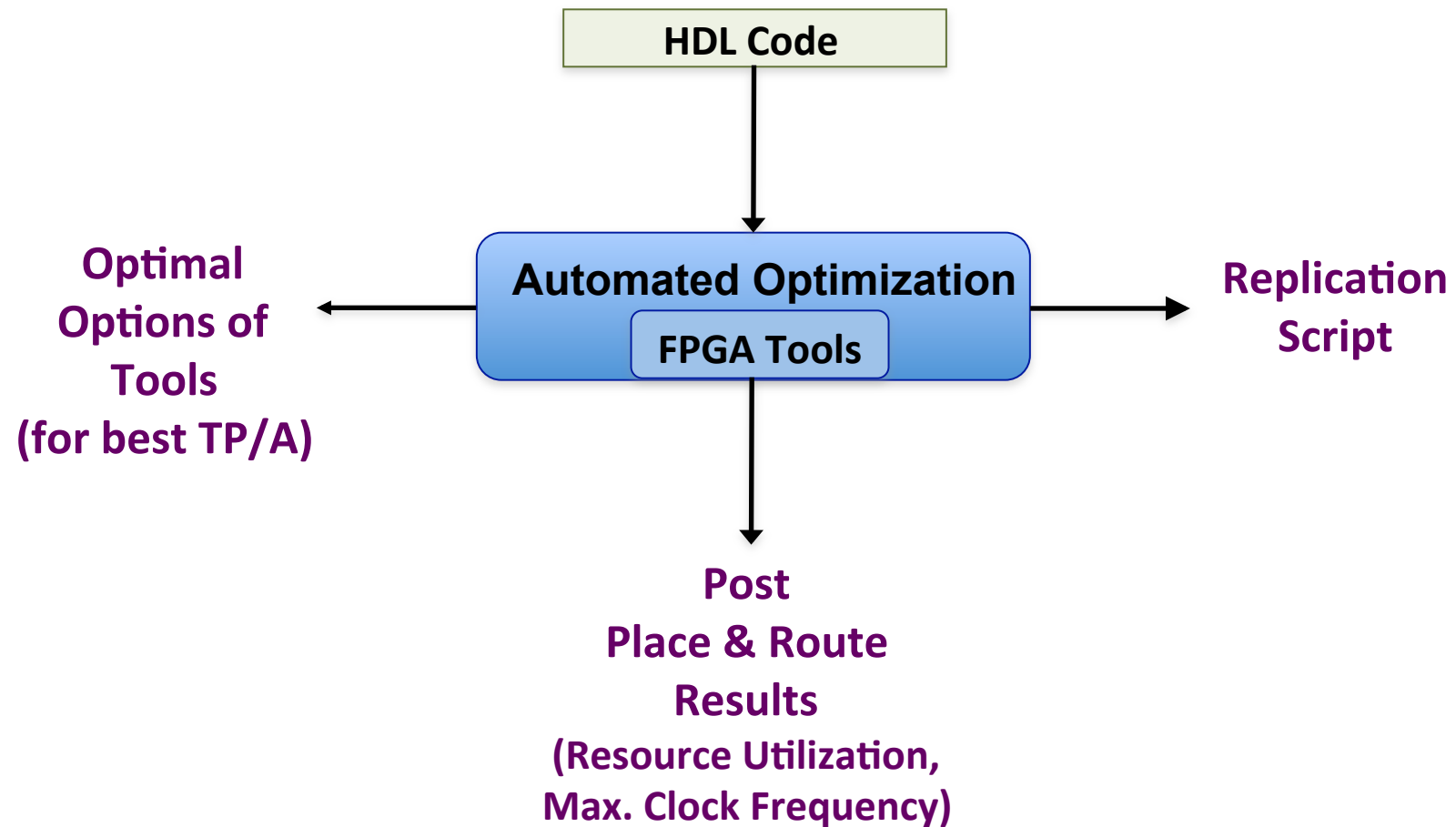
High-Performance FPGA Families used for benchmarking of All Round 2 Candidates & AES-GCM

- Xilinx Virtex-6: xc6vlx240tff1156-3
- Xilinx Virtex-7: xc7vx485tffg1761-3
- Altera Stratix IV: ep4se530h35c2
- Altera Stratix V: 5sgxea7k2f40c1

Low-Cost FPGA Families used for benchmarking of 10 Candidates with the Smallest Area in High-Performance Benchmarking:

- Xilinx Spartan-6: xc6slx16csg324-3
- Xilinx Artix-7: xc7a100tcsg324-3
- Altera Cyclone IV: EP4CE22F17C6
- Altera Cyclone V: 5CEBA4F23C7

RTL Benchmarking



FPGA Tools (1)

For Benchmarking Targeting Xilinx FPGAs (other than Virtex 7):

Target FPGAs:	Virtex-6, Spartan 6, Artix 7
Synthesis Tool:	Xilinx XST 14.7
Implementation Tool:	Xilinx ISE 14.7
Automated Optimization:	ATHENa

For Benchmarking Targeting Altera FPGAs:

Target FPGAs:	Stratix IV, Stratix V, Cyclone IV, Cyclone V
Synthesis Tool:	Quartus Prime 16.0.0
Implementation Tool:	Quartus Prime 16.0.0
Automated Optimization:	ATHENa

FPGA Tools (2)

For Benchmarking Targeting Xilinx Virtex 7 FPGAs:

Target FPGAs:	Virtex-7
Synthesis Tool:	Xilinx Vivado 2015.1
Implementation Tool:	Xilinx Vivado 2015.1
Automated Optimization:	25 Default Strategies of Vivado

Embedded Memories & DSP Units

- No embedded memories and no embedded DSP units allowed inside of
 - AEAD: for single-pass algorithms, and
 - AEAD-TP: for two-pass algorithms
- Their use eliminated using options of the respective tools (including, if necessary, the synthesis tool directives added to HDL code)
- **Without** this approach
 - Area = Resource Utilization Vector
e.g. Area = (1056 Slices, 4 BRAMs, 67 DSP units)
 - No known way of comparing FPGA Resource Utilization Vectors
 - No way of calculating Throughput/Area
- **Additional Benefit**
 - Good correlation of the obtained results with the corresponding ASIC results, as demonstrated during the SHA-3 Competition.
See <http://eprint.iacr.org/2012/368>, Section 9

Dealing with I/O Ports

- No wrappers used
- Ports of
 - AEAD: for single-pass algorithms, and
 - AEAD-TP: for two-pass algorithms, connected directly to I/O pins of a target FPGA
- In case of a number of I/O pins exceeded, a larger FPGA device of a given family used.
This step required only for
 - low-cost FPGA familiesAND
 - a few API compliant designs with the largest PDI/SDI/DO port widths, as well as
 - a few non-compliant designs with the full-block width interfaces.

Results

Performance Metrics

High-Speed Designs

Primary:

- Throughput/Area
- Throughput

Secondary:

- Area

Lightweight & High-Speed/Lightweight Designs

Primary:

- Throughput/Area
- Area

Secondary:

- Throughput

Throughput Types

- Authenticated Encryption Throughput
- Authenticated Decryption Throughput
 - Different only for
 - **HS1-SIV**
 - **SHELL**
 - **Deoxys, Joltik (by GMU Team)**
- Authentication-Only Throughput
 - Different only for
 - **AEZ**
 - **CLOC, SILC (by CLOC-SILC Team)**
 - **Deoxys, Joltik (by Axel & Marc)**
 - **HS1-SIV**
 - **OMD**
 - **PAEQ**
 - **SHELL**

Area Units

For Xilinx FPGAs:

Target FPGAs: Virtex-6, Virtex 7, Spartan-6, Artix-7

Units of Area: LUTs (Look-up Tables)

Slices (1 Slice contains 4 LUTs, 8 registers & additional logic)

For Altera FPGAs (other than Cyclone IV):

Target FPGAs: Stratix IV, Stratix V, Cyclone V

Units of Area: ALUTs (Adaptive Look-up Tables)

ALM (Adaptive Logic Modules)

For Altera Cyclone IV:

Units of Area: Logic Elements (LE)

Included in High-Speed Rankings

Algorithms & Their Variants:

- AES-GCM
- PRIMATE_s-GIBBON, PRIMATE_s-HANUMAN
- CLOC, SILC
- 25 other Round 2 Candidates (all other except Tiaoxin & SHELL)
= **30 algorithms**
- **Key size ≥ 96 bits**

Designs:

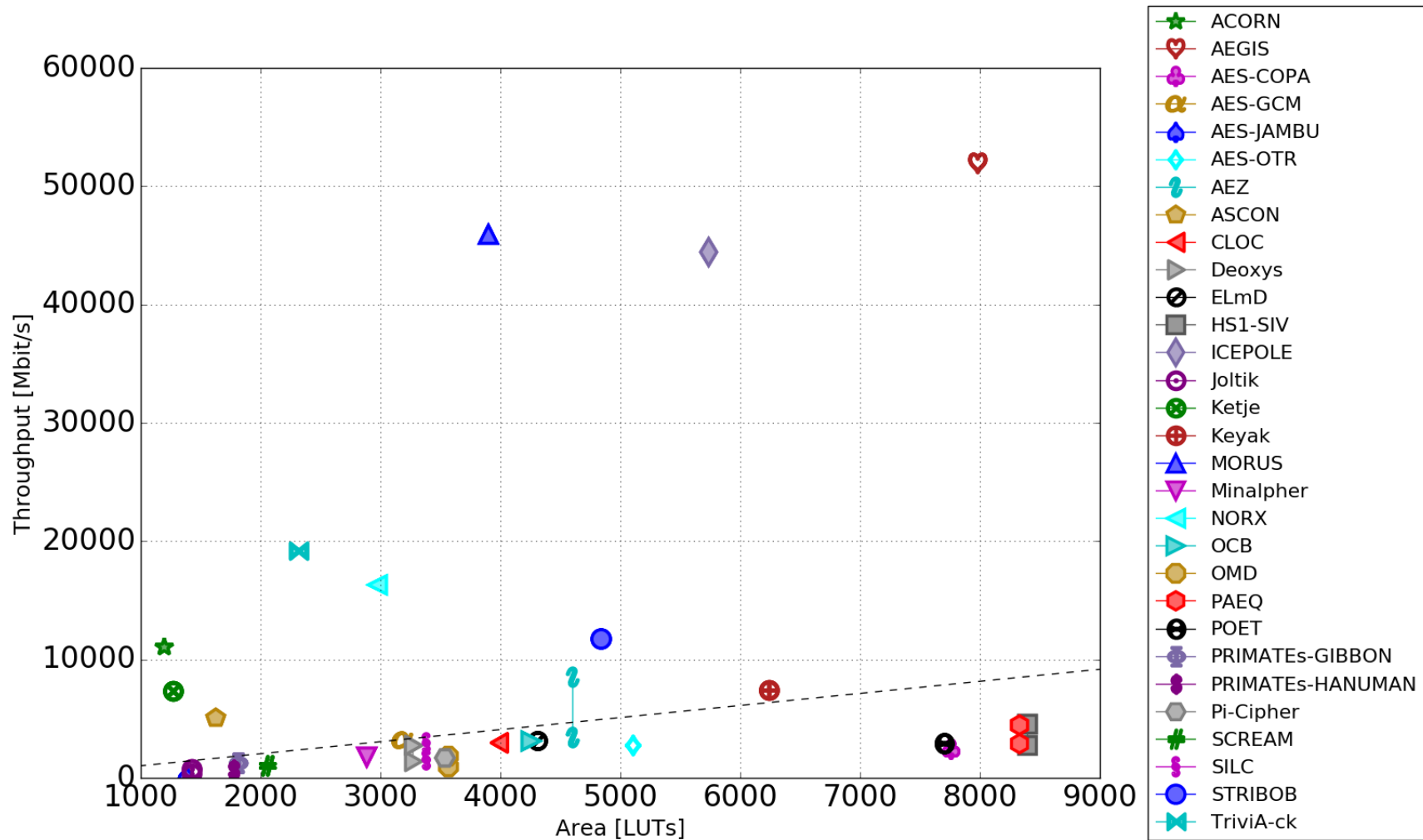
- Only Compliant with the CAESAR Hardware API
(including the design for Keyak with $|AD| \leq 24$ bytes)

Relative Results vs. [Absolute] Results

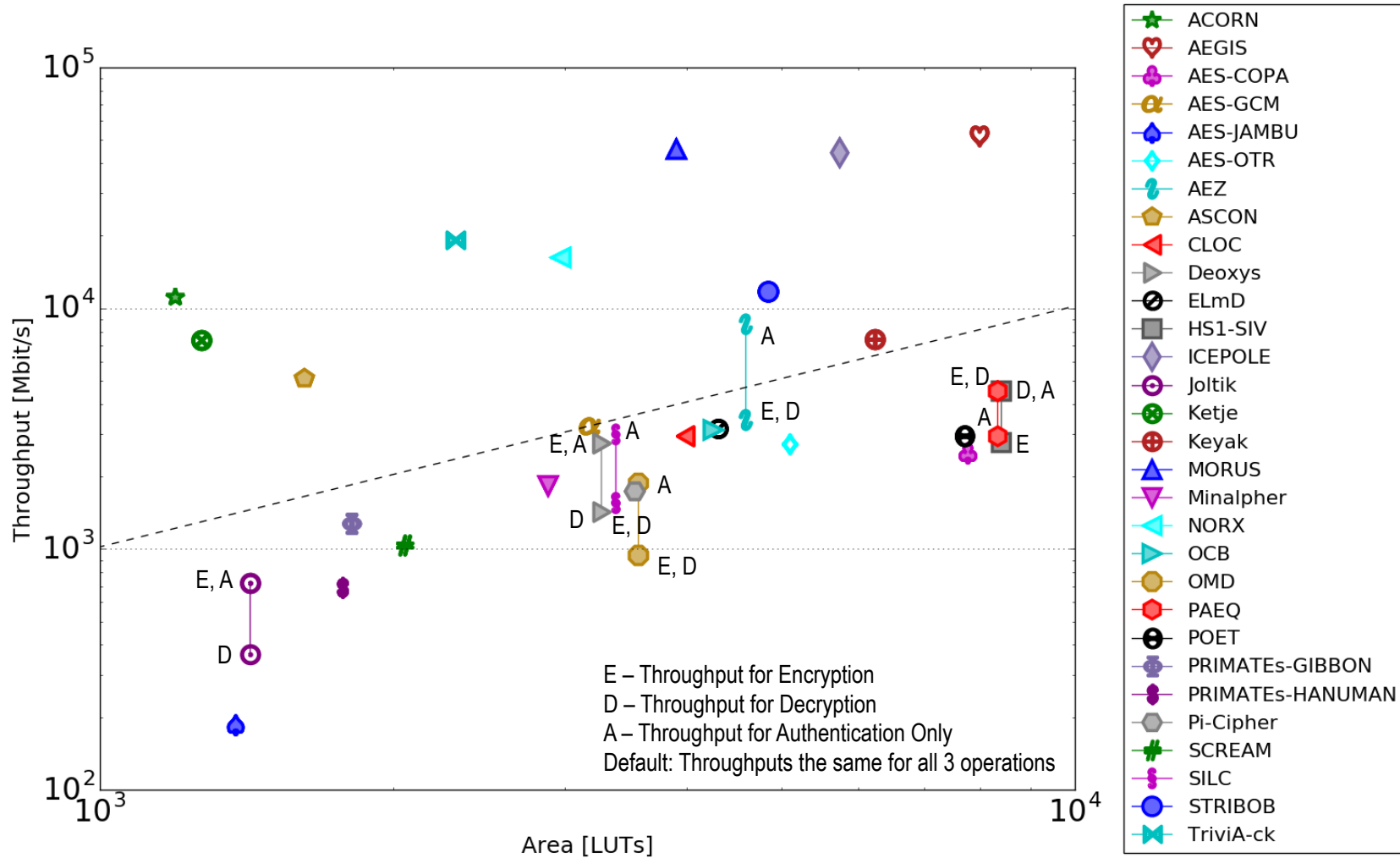
- **Relative Results**
 - Results divided by the corresponding results for AES-GCM, e.g.,
Relative Throughput of Candidate X = Throughput of Candidate X / Throughput of AES-GCM
 - Represent speed-up, area savings, efficiency improvement compared to AES-GCM
 - No units
 - 29 results reported (all results for AES-GCM by definition 1)
- **[Absolute] Results** (“Absolute” portion in the metric name optional)
 - “Regular” results for each candidate
 - Reported in the ATHENA Database of Results
 - Units appropriate to the given performance metric,
e.g., Mbit/s for Absolute Throughput

Virtex-6

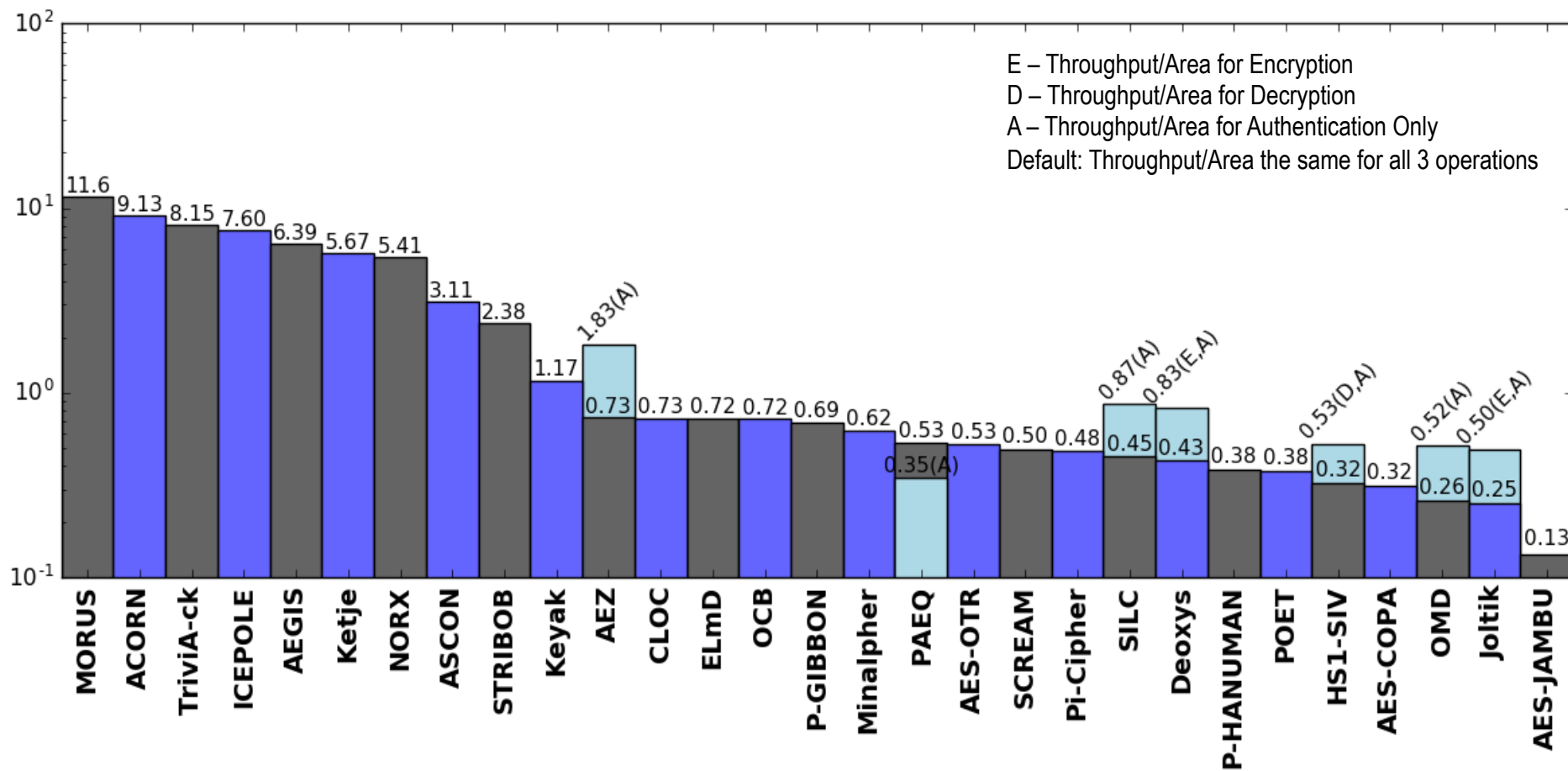
Results for Virtex 6 – Throughput vs. Area Linear Scale



Results for Virtex 6 – Throughput vs. Area Logarithmic Scale



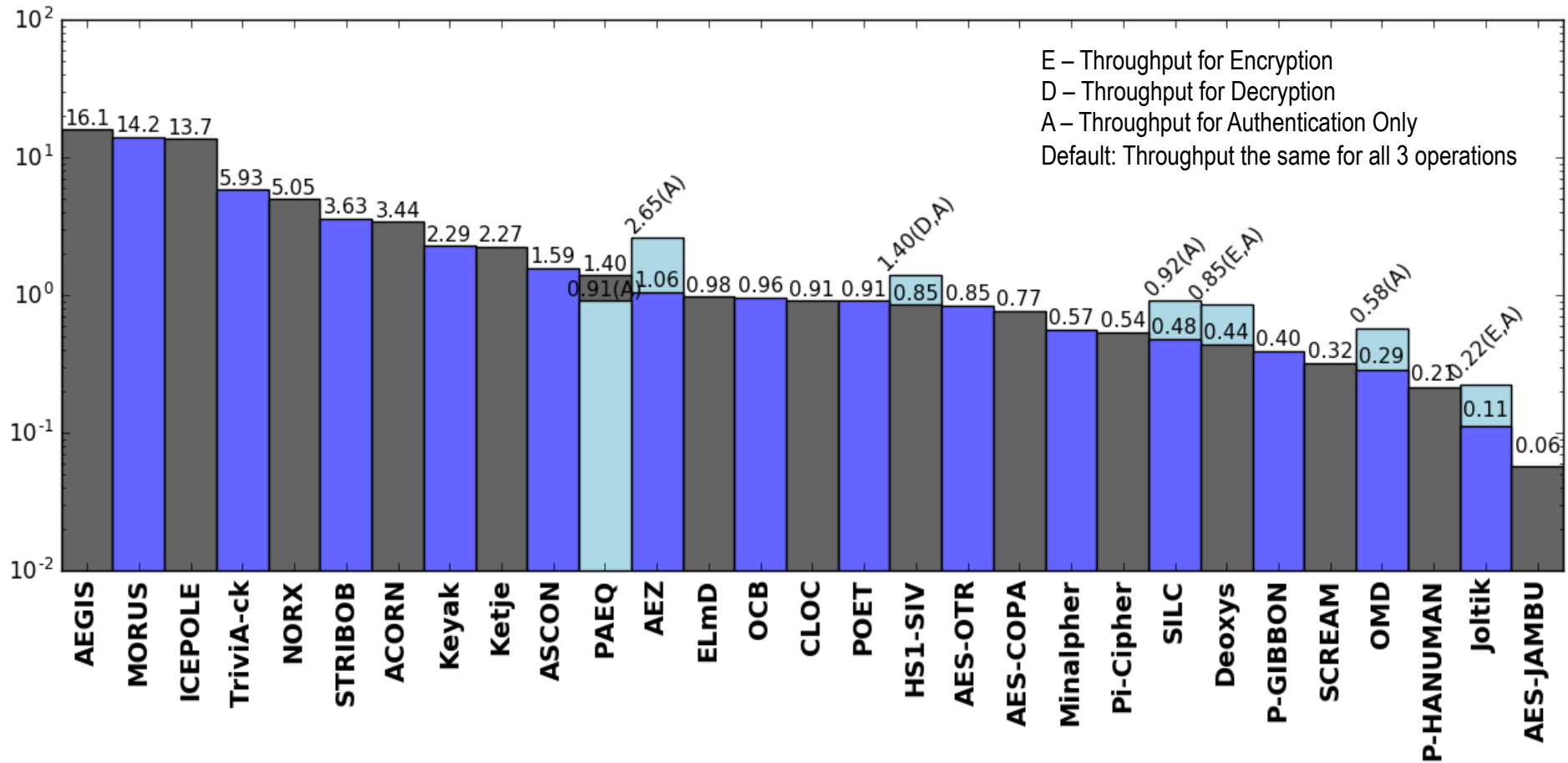
Relative Throughput/Area in Virtex 6 vs. AES-GCM



Throughput/Area of AES-GCM = 1.020 (Mbit/s)/LUTs

Relative Throughput in Virtex 6

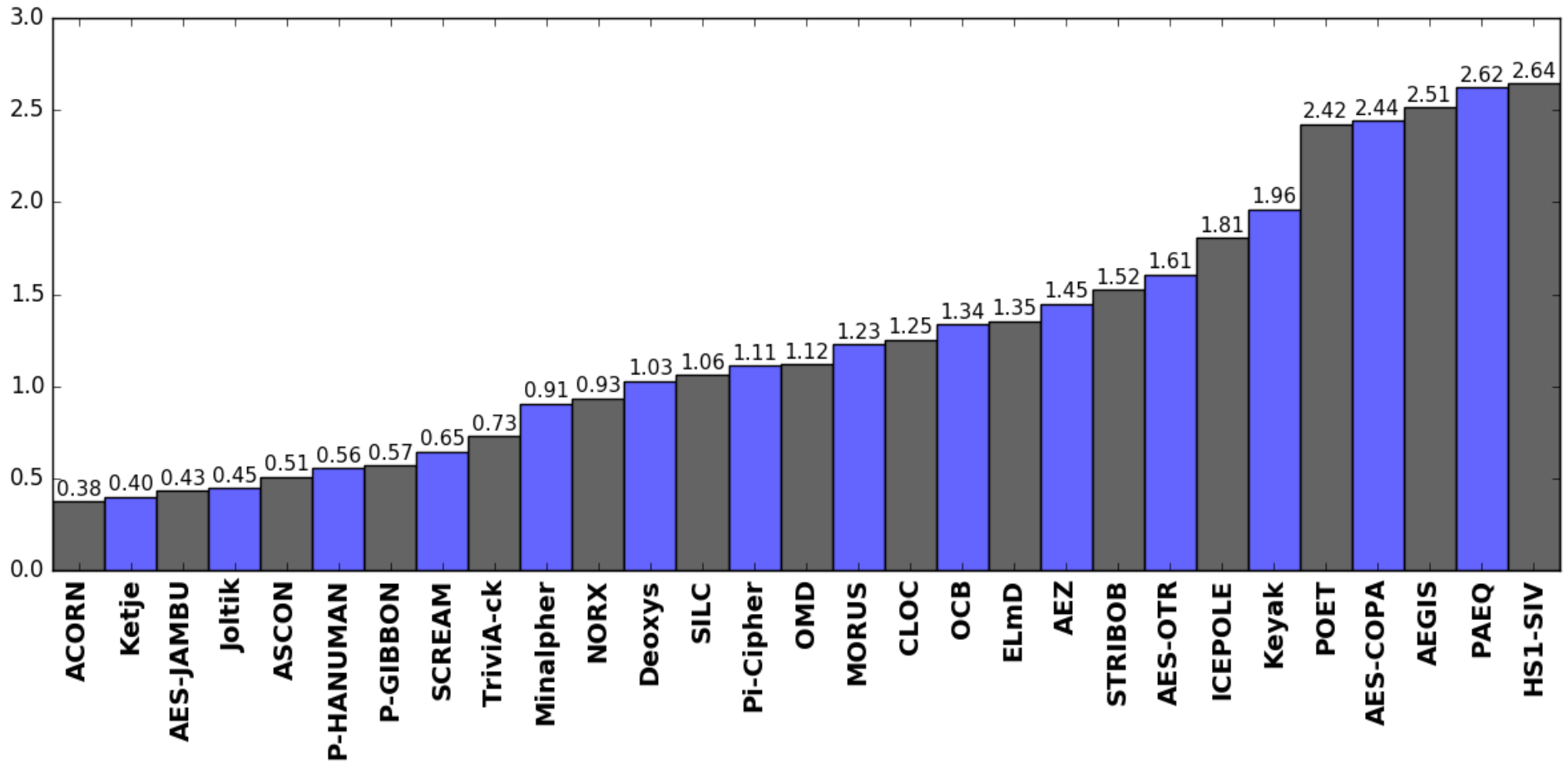
Ratio of a given Cipher Throughput/Throughput of AES-GCM



Throughput of AES-GCM = 3239 Mbit/s

Relative Area (#LUTs) in Virtex 6

Ratio of a given Cipher Area/Area of AES-GCM

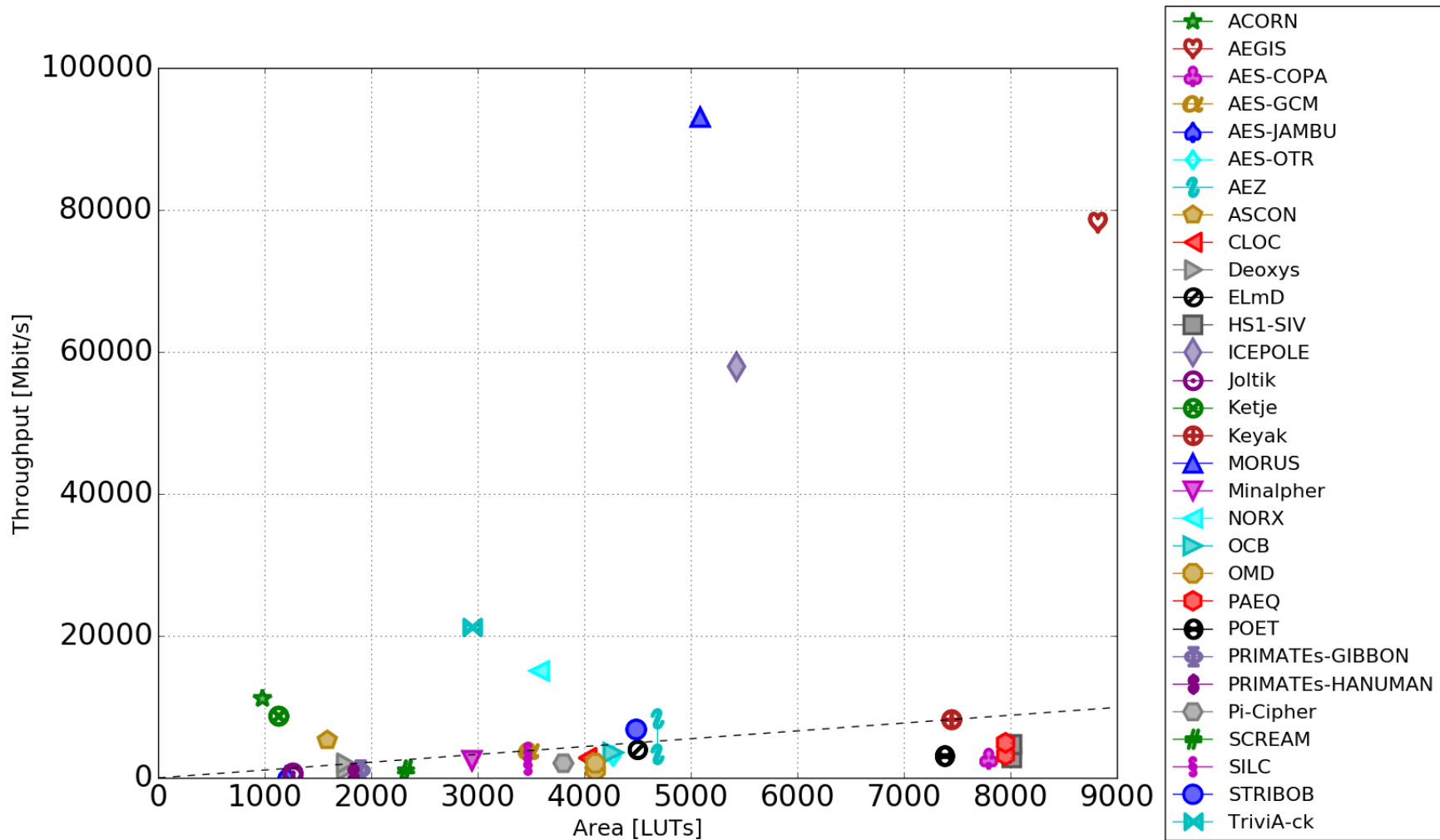


Area of AES-GCM = 3175 LUTs

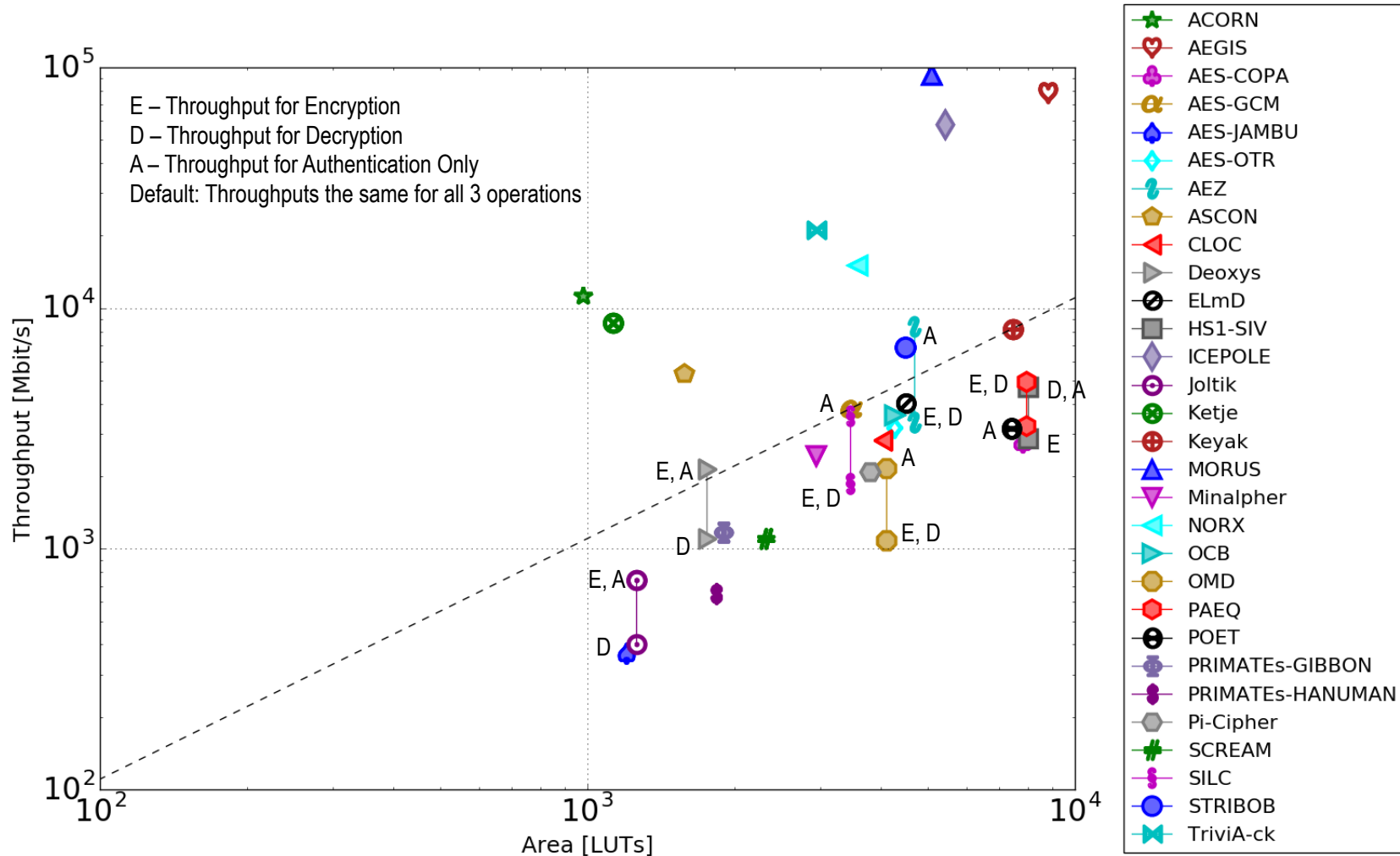
Virtex-7

Results for Virtex 7 – Throughput vs. Area

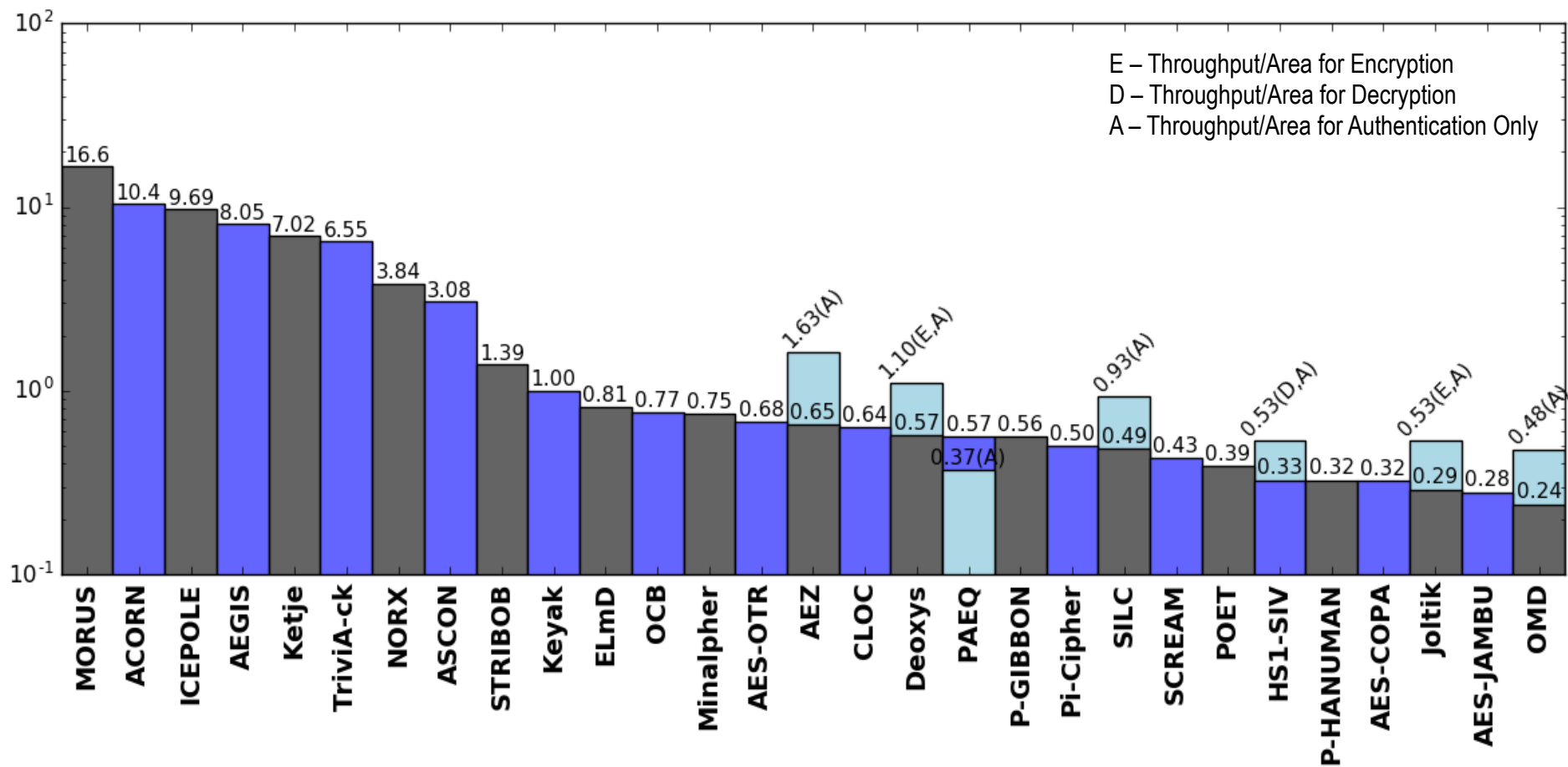
Linear Scale



Results for Virtex 7 – Throughput vs. Area Logarithmic Scale



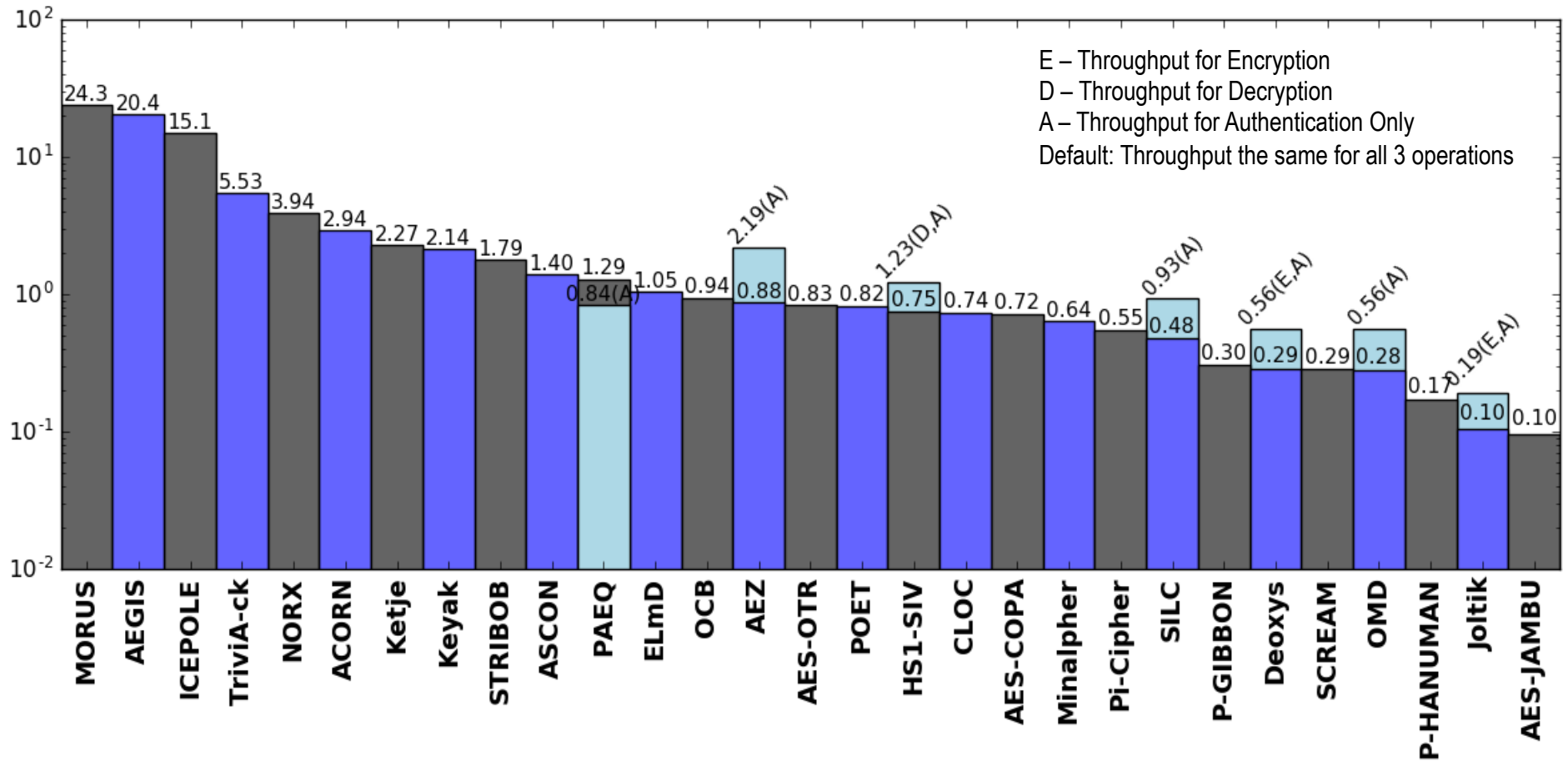
Relative Throughput/Area in Virtex 7 vs. AES-GCM



Throughput/Area of AES-GCM = 1.103 (Mbit/s)/LUTs

Relative Throughput in Virtex 7

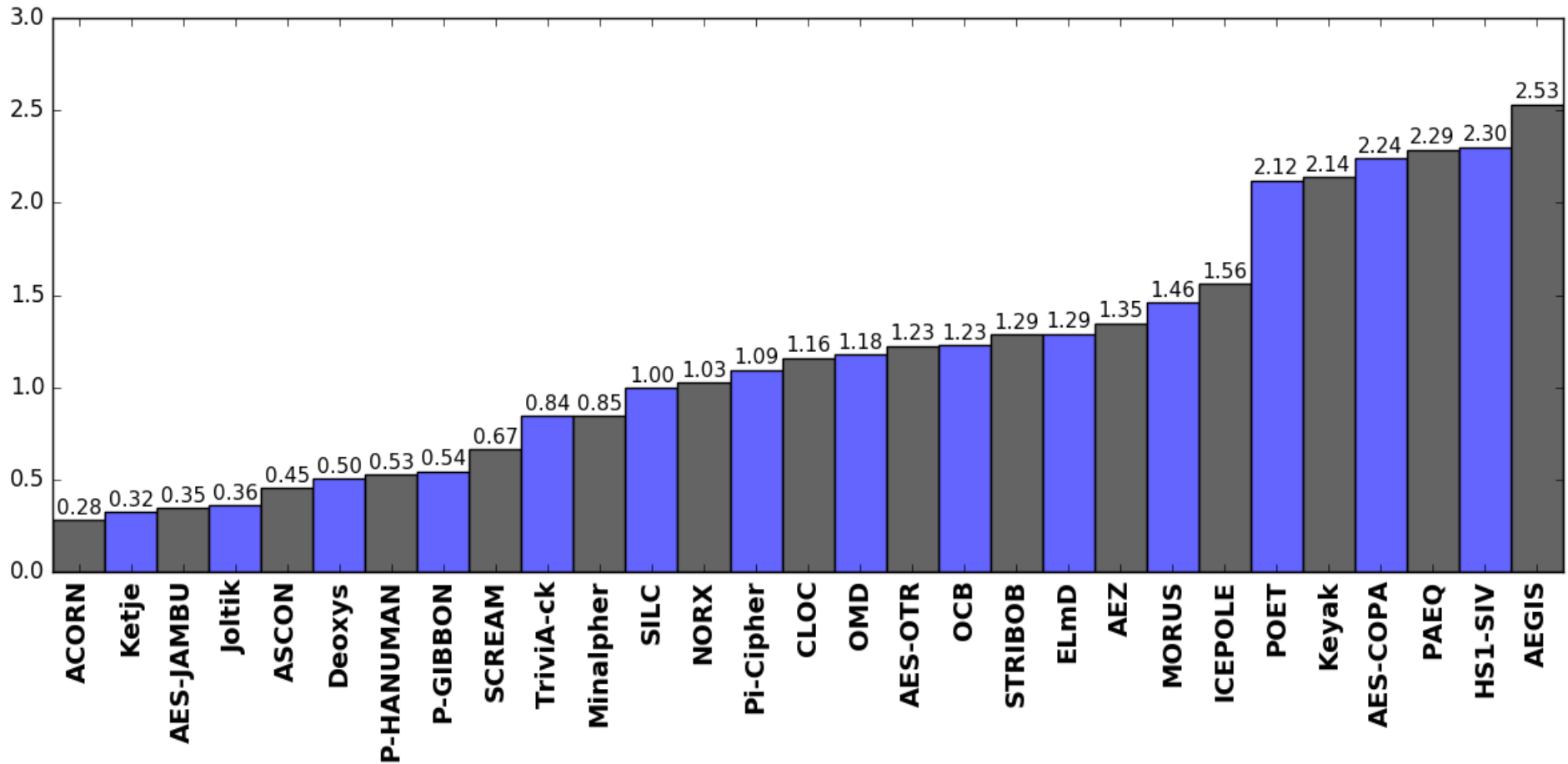
Ratio of a given Cipher Throughput/Throughput of AES-GCM



Throughput of AES-GCM = 3837 Mbit/s

Relative Area (#LUTs) in Virtex 7

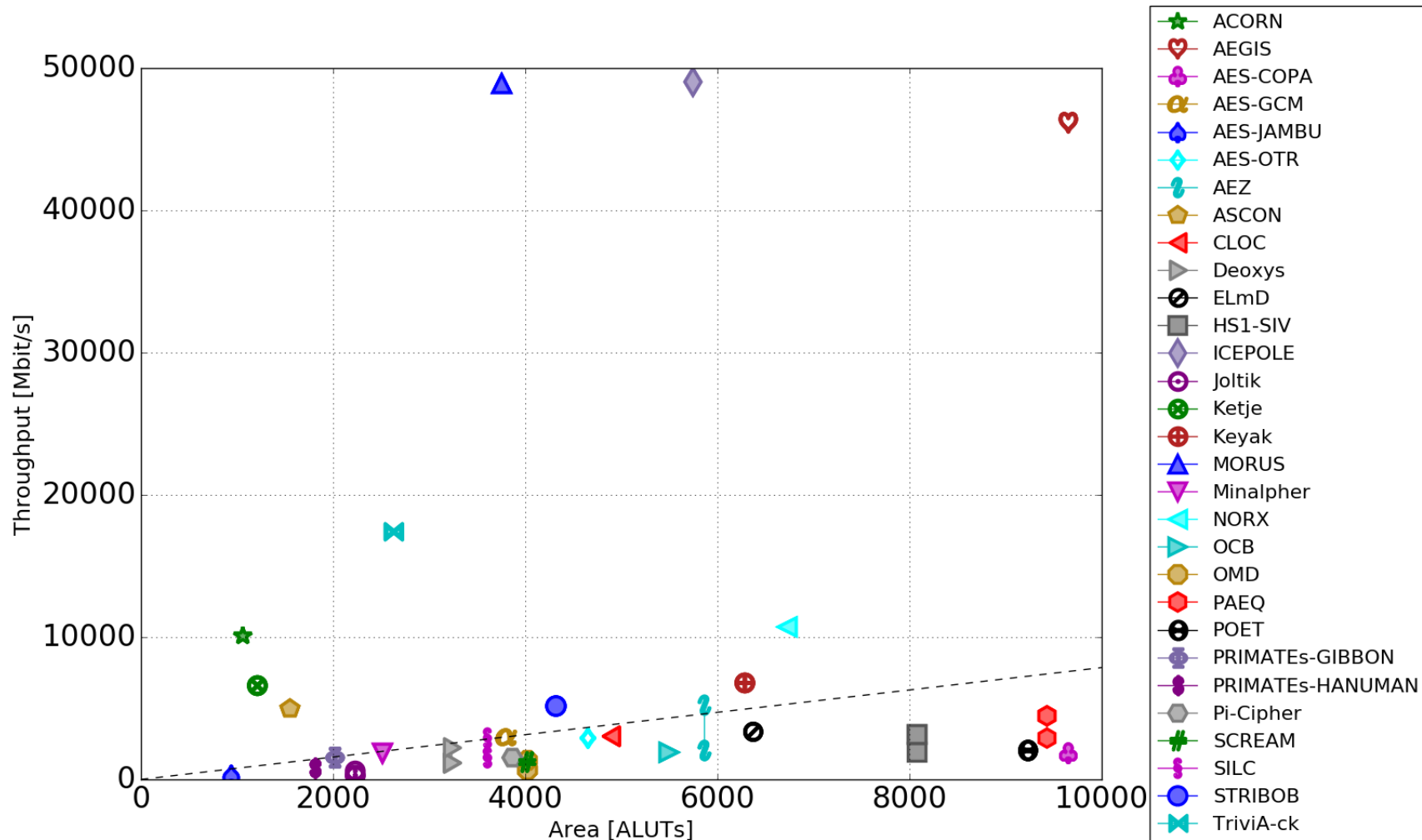
Ratio of a given Cipher Area/Area of AES-GCM



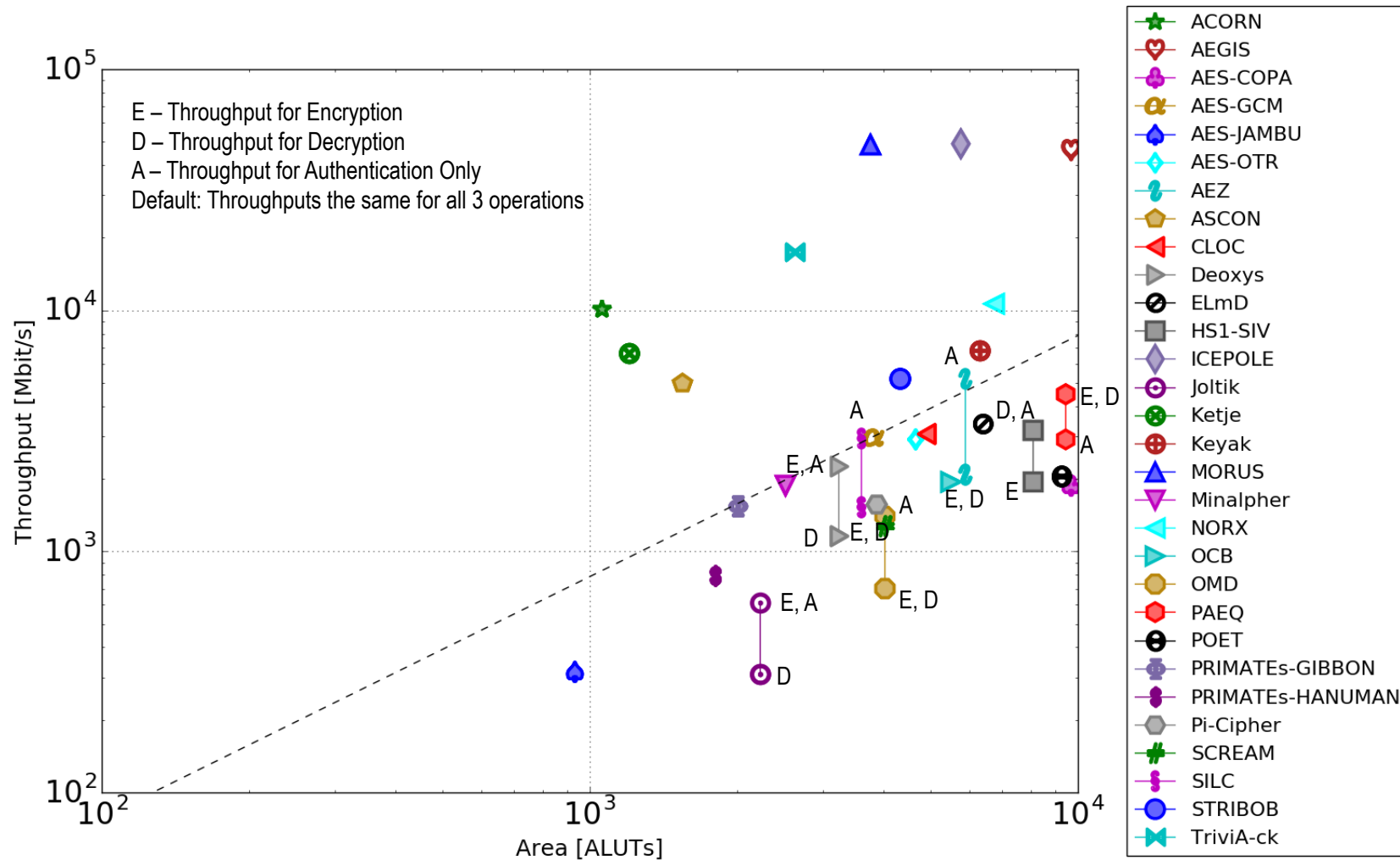
Area of AES-GCM = 3478 LUTs

Stratix IV

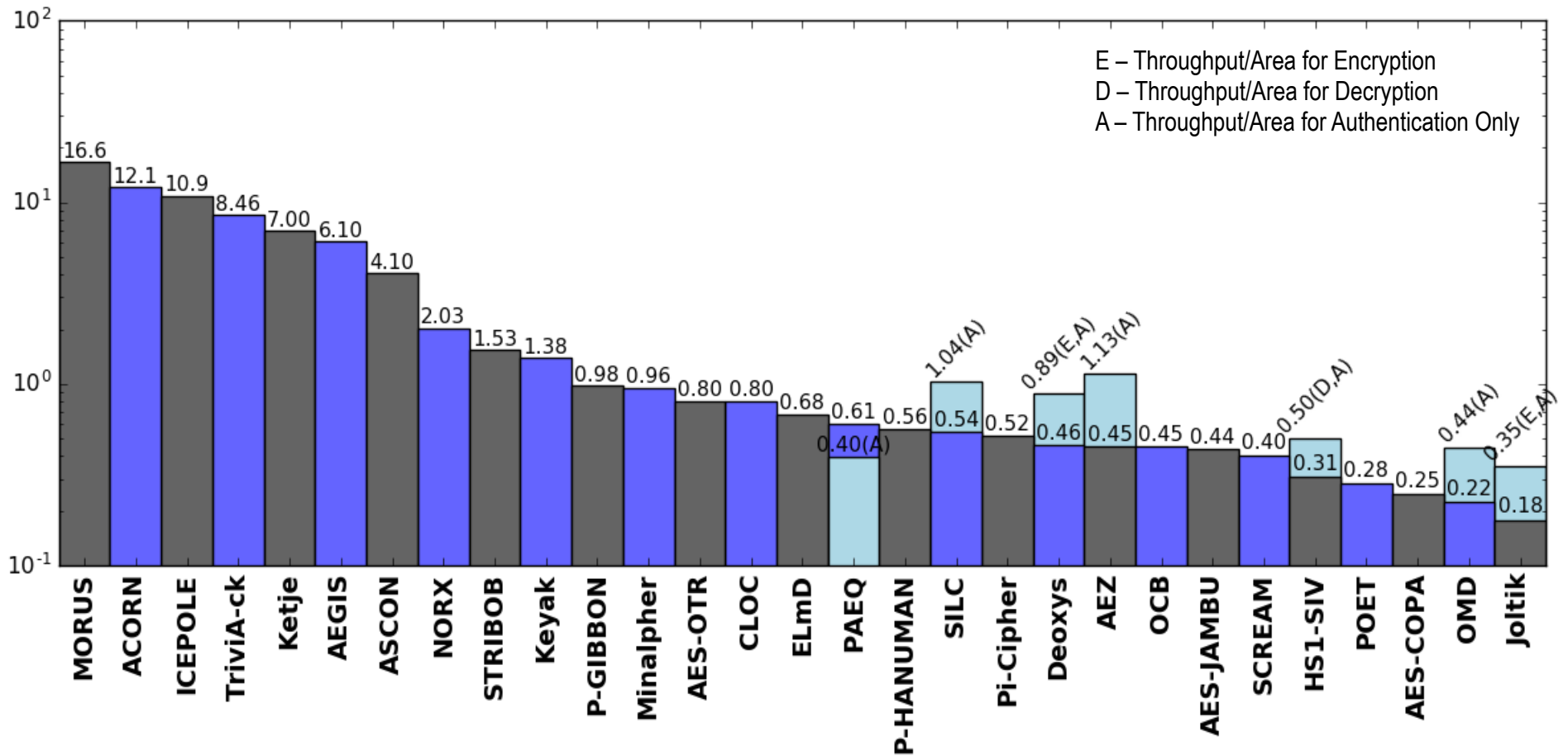
Results for Stratix IV – Throughput vs. Area Linear Scale



Results for Stratix IV – Throughput vs. Area Logarithmic Scale



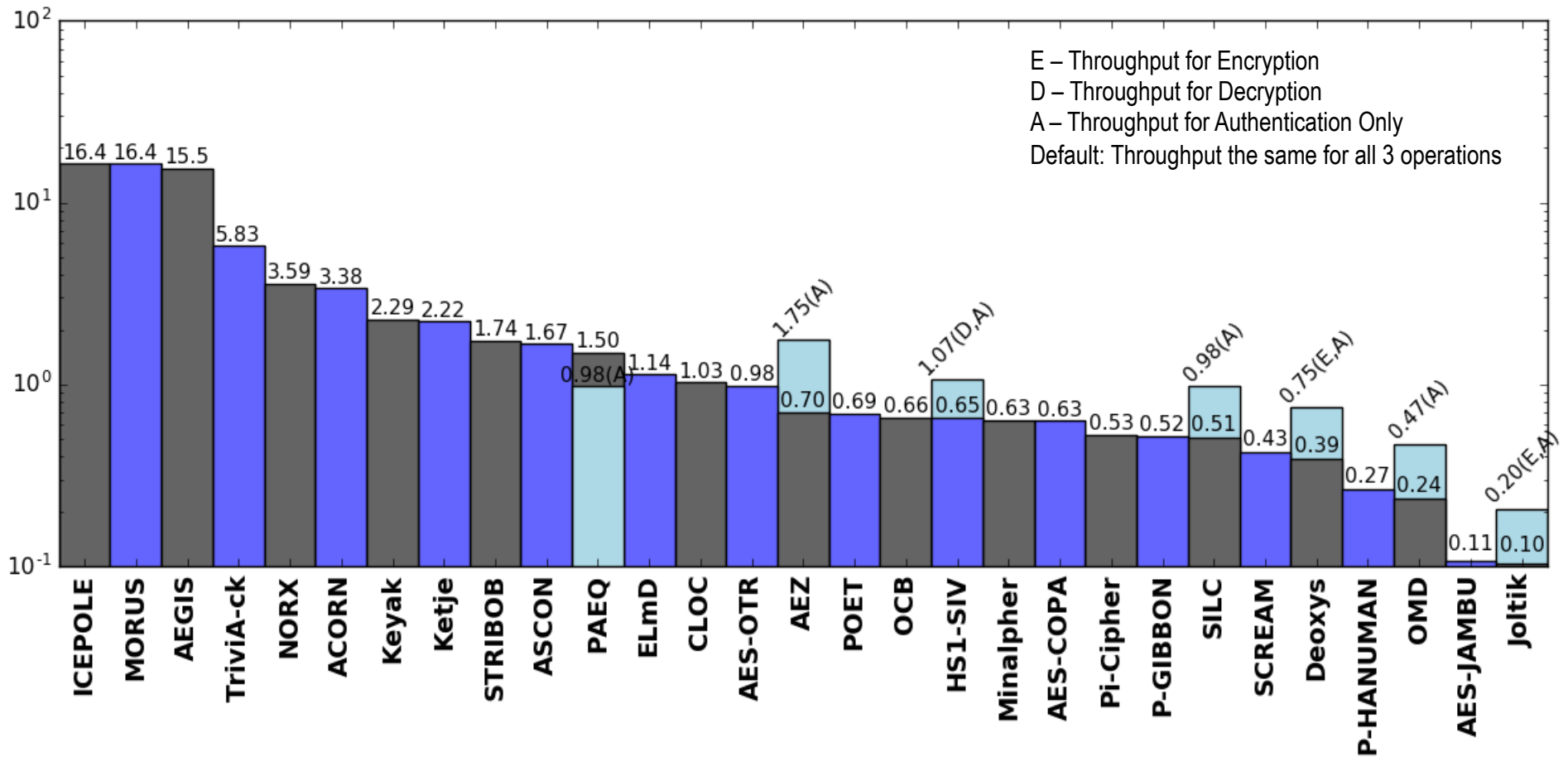
Relative Throughput/Area in Stratix IV vs. AES-GCM



Throughput/Area of AES-GCM = 0.786 (Mbit/s)/ALUTs

Relative Throughput in Stratix IV

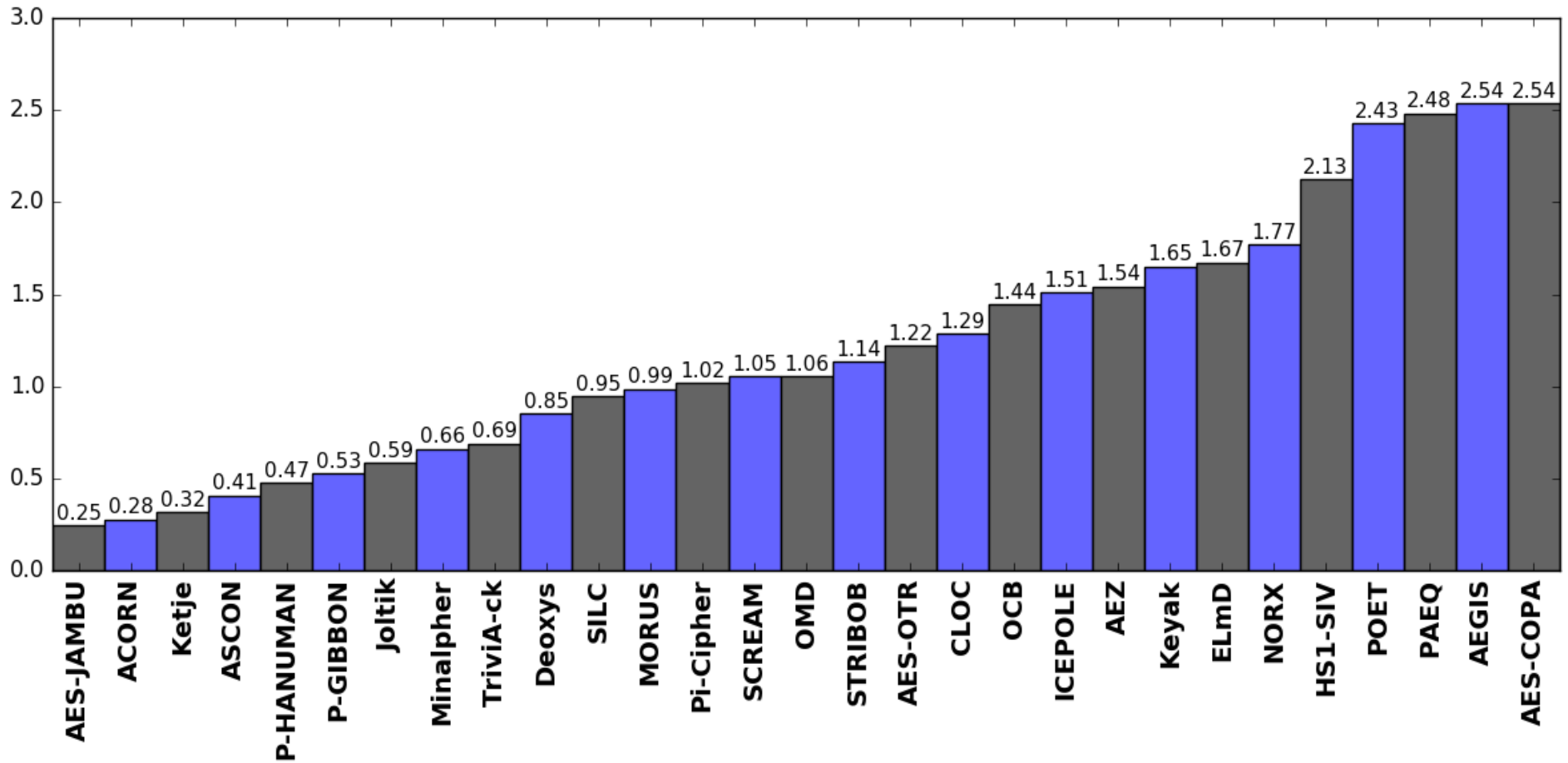
Ratio of a given Cipher Throughput/Throughput of AES-GCM



Throughput of AES-GCM = 2987 Mbit/s

Relative Area (#ALUTs) in Stratix IV

Ratio of a given Cipher Area/Area of AES-GCM

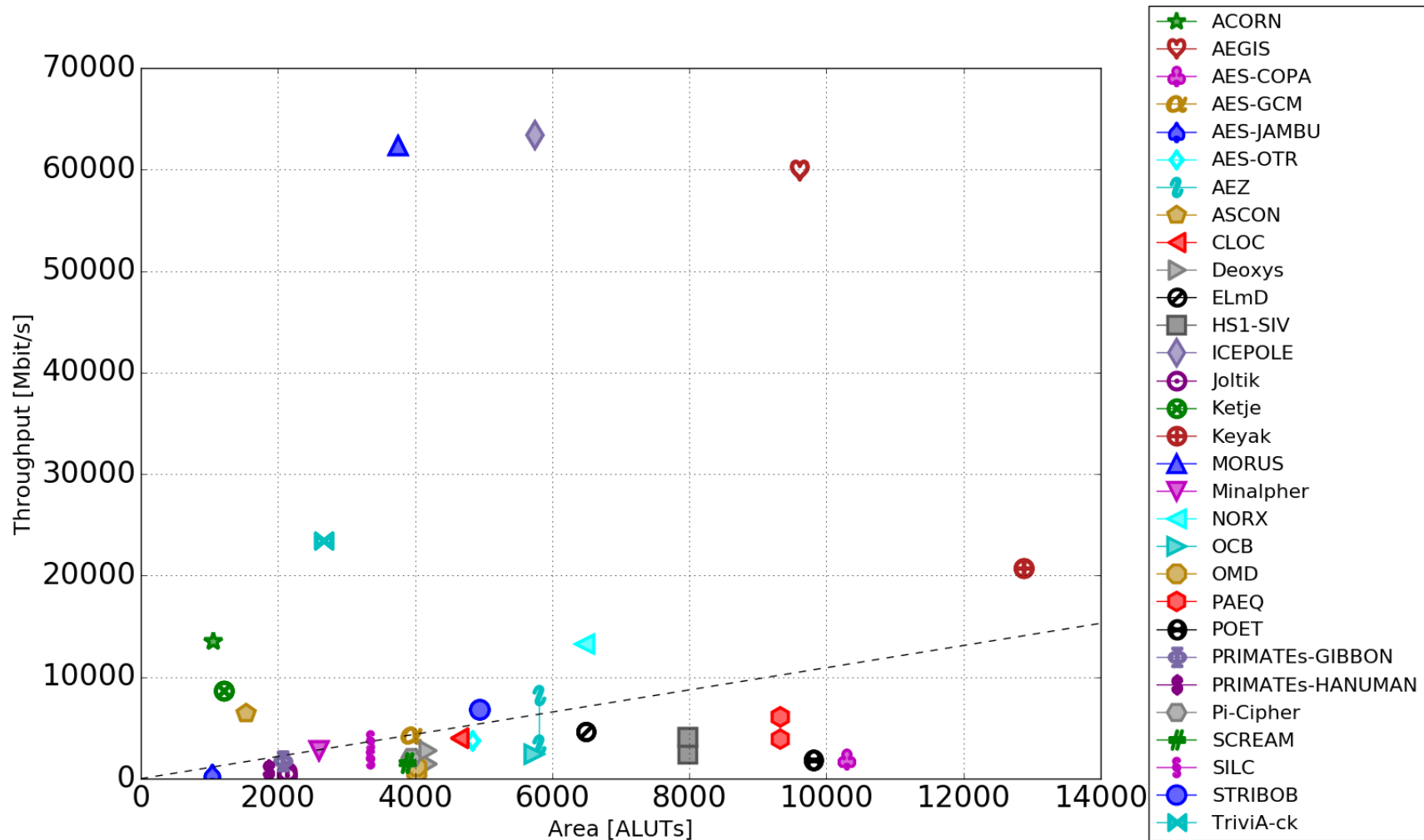


Area of AES-GCM = 3800 ALUTs

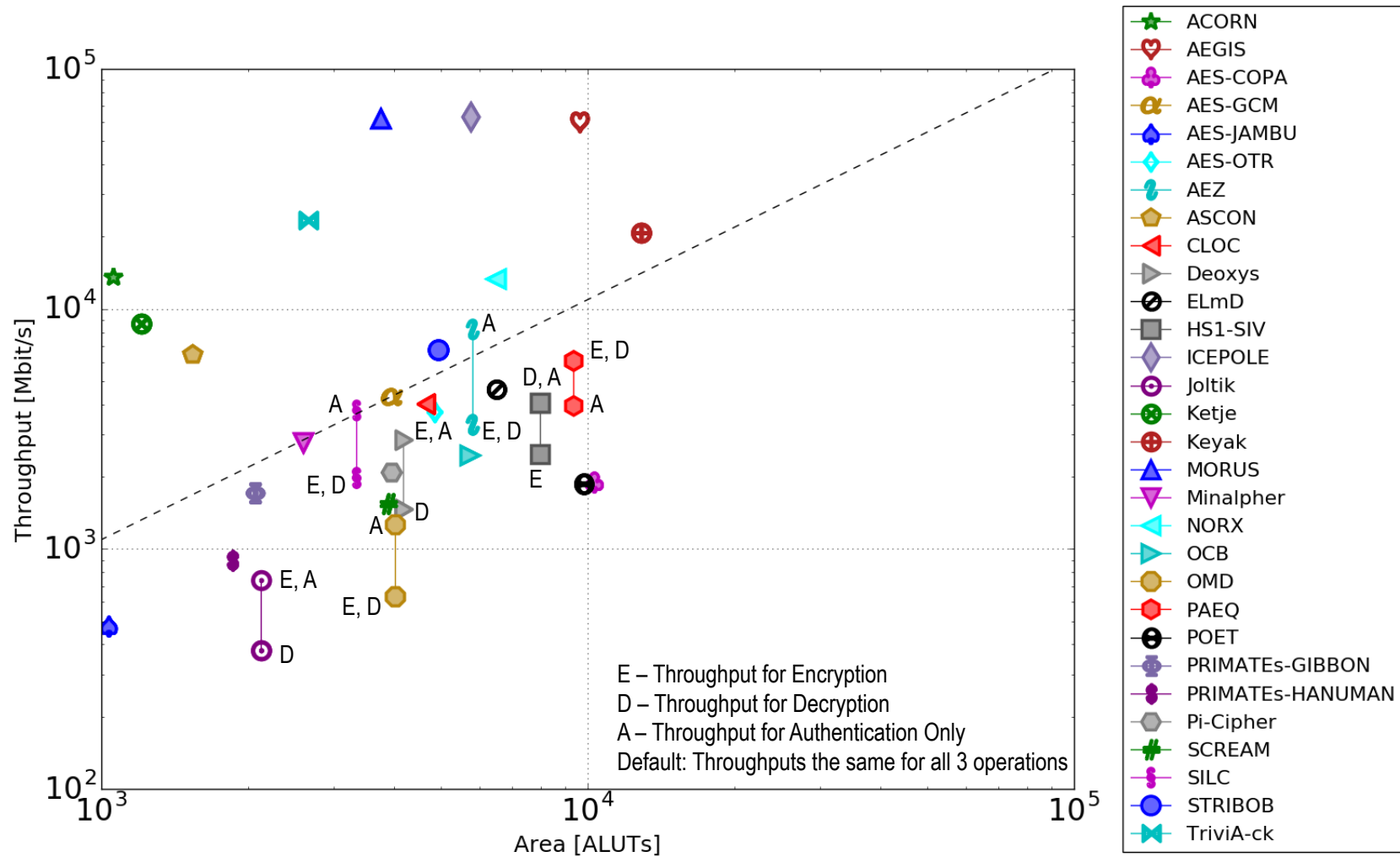
Stratix V

Results for Stratix V – Throughput vs. Area

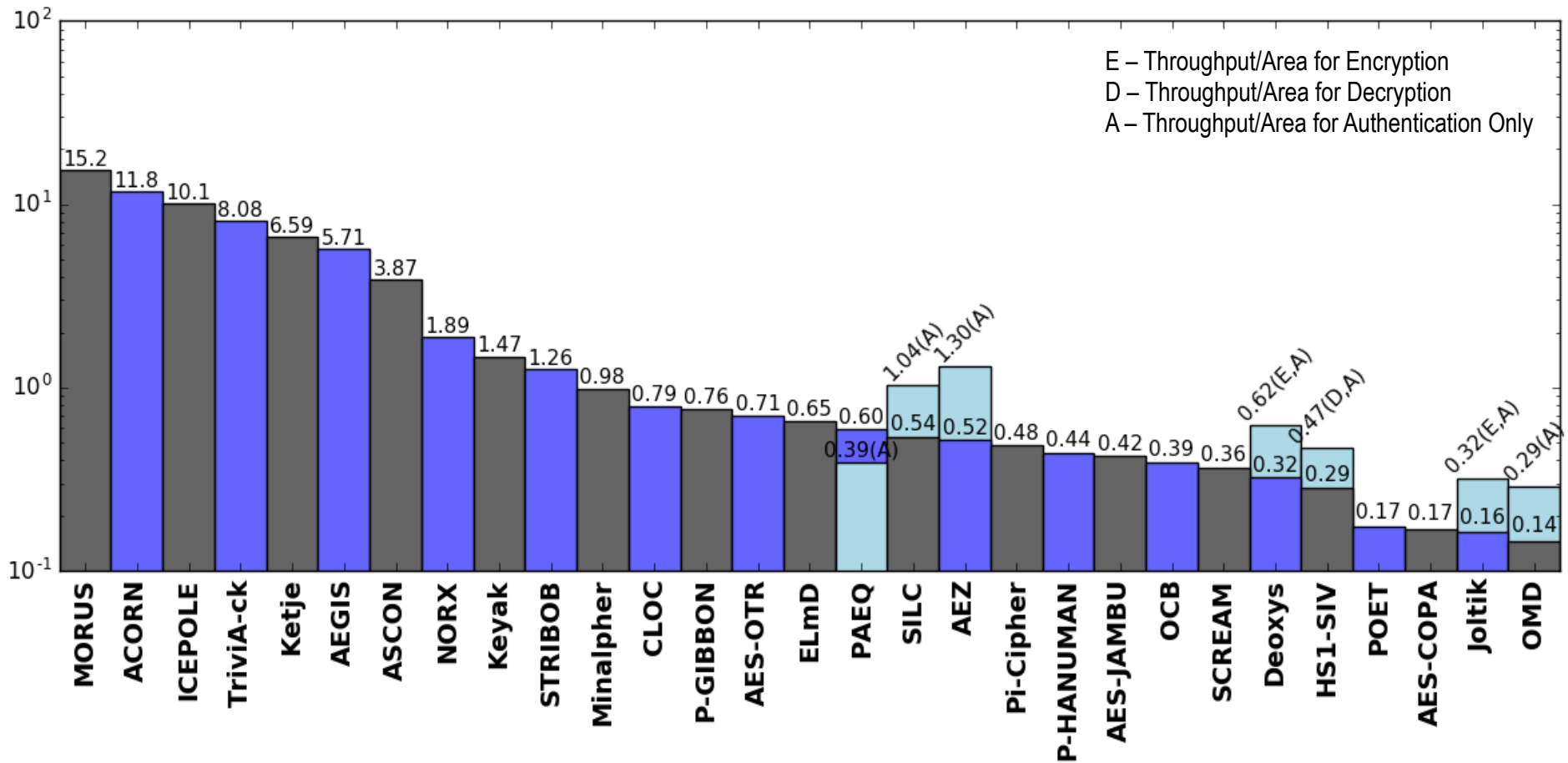
Linear Scale



Results for Stratix V – Throughput vs. Area Logarithmic Scale



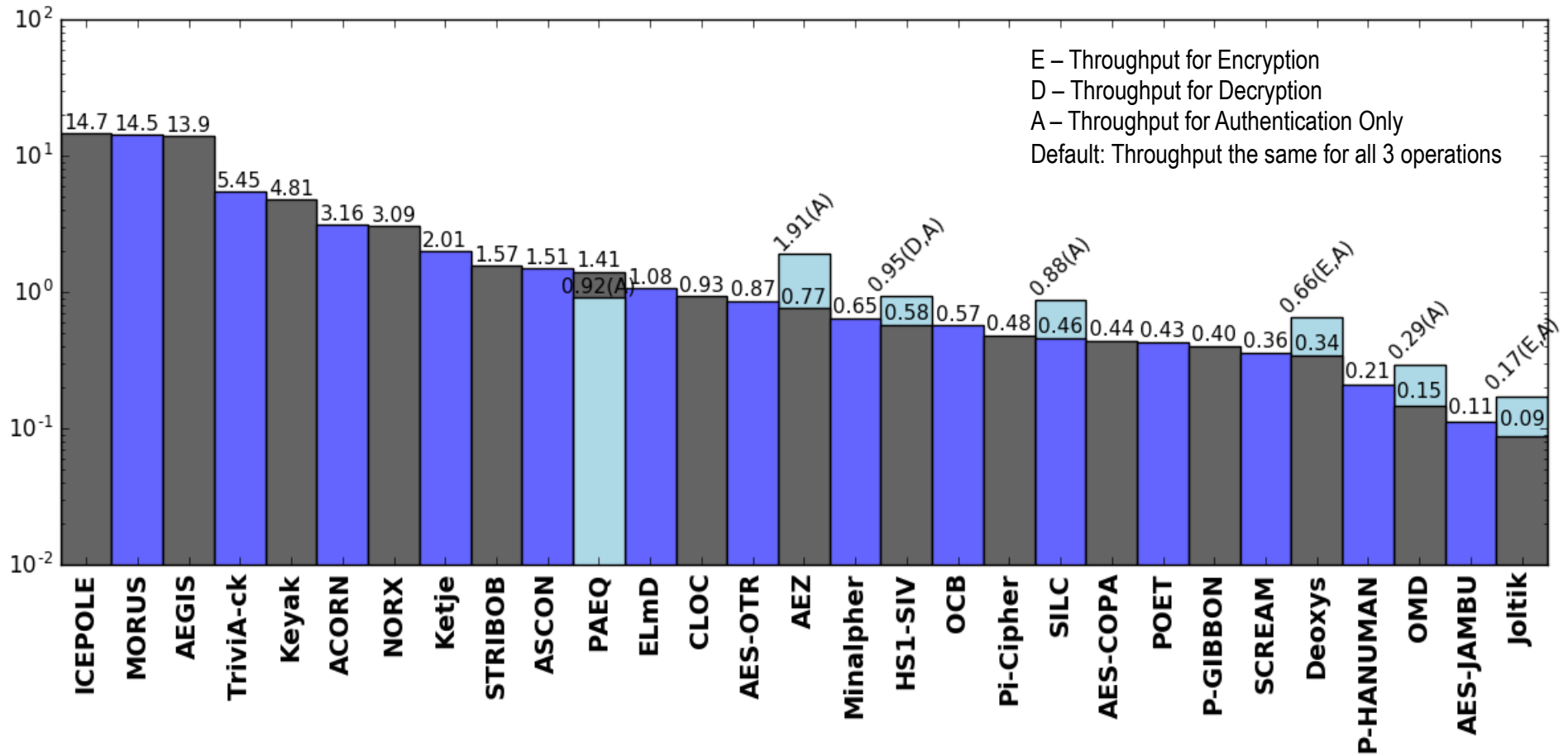
Relative Throughput/Area in Stratix V vs. AES-GCM



Throughput/Area of AES-GCM = 1.093 (Mbit/s)/ALUTs

Relative Throughput in Stratix V

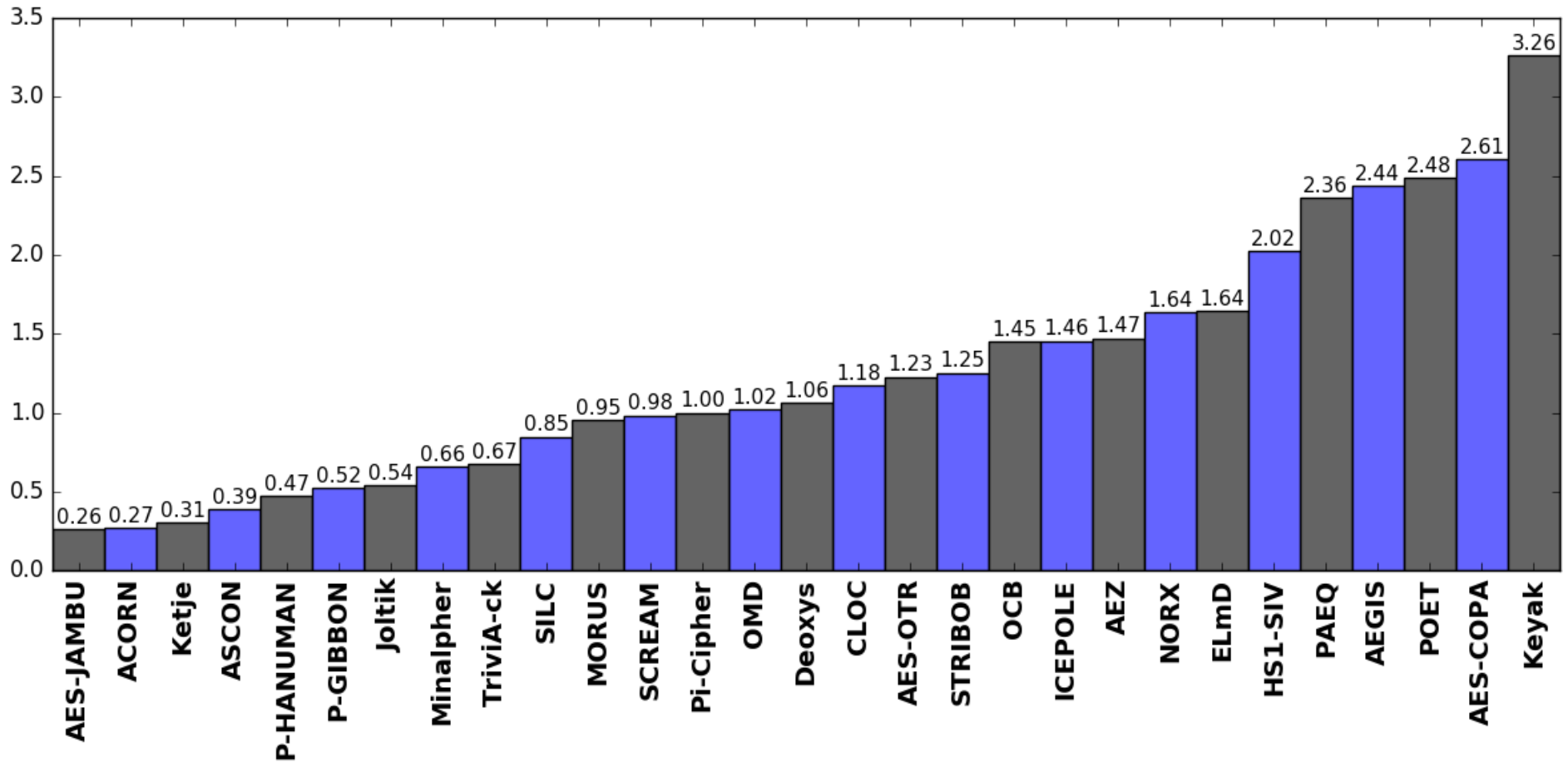
Ratio of a given Cipher Throughput/Throughput of AES-GCM



Throughput of AES-GCM = 4310 Mbit/s

Relative Area (#ALUTs) in Stratix V

Ratio of a given Cipher Area/Area of AES-GCM



Area of AES-GCM = 3943 ALUTs

Included in Preliminary Lightweight Rankings

Algorithms & Their Variants:

- Among 10 smallest in the majority of High-Speed rankings
- **Any key size**
- ACORN, AES-JAMBU, ASCON, Joltik, Ketje, Minalpher,
- PRIMATE_s-HANUMAN, PRIMATE_s-GIBBON, SCREAM, Trivium-ck

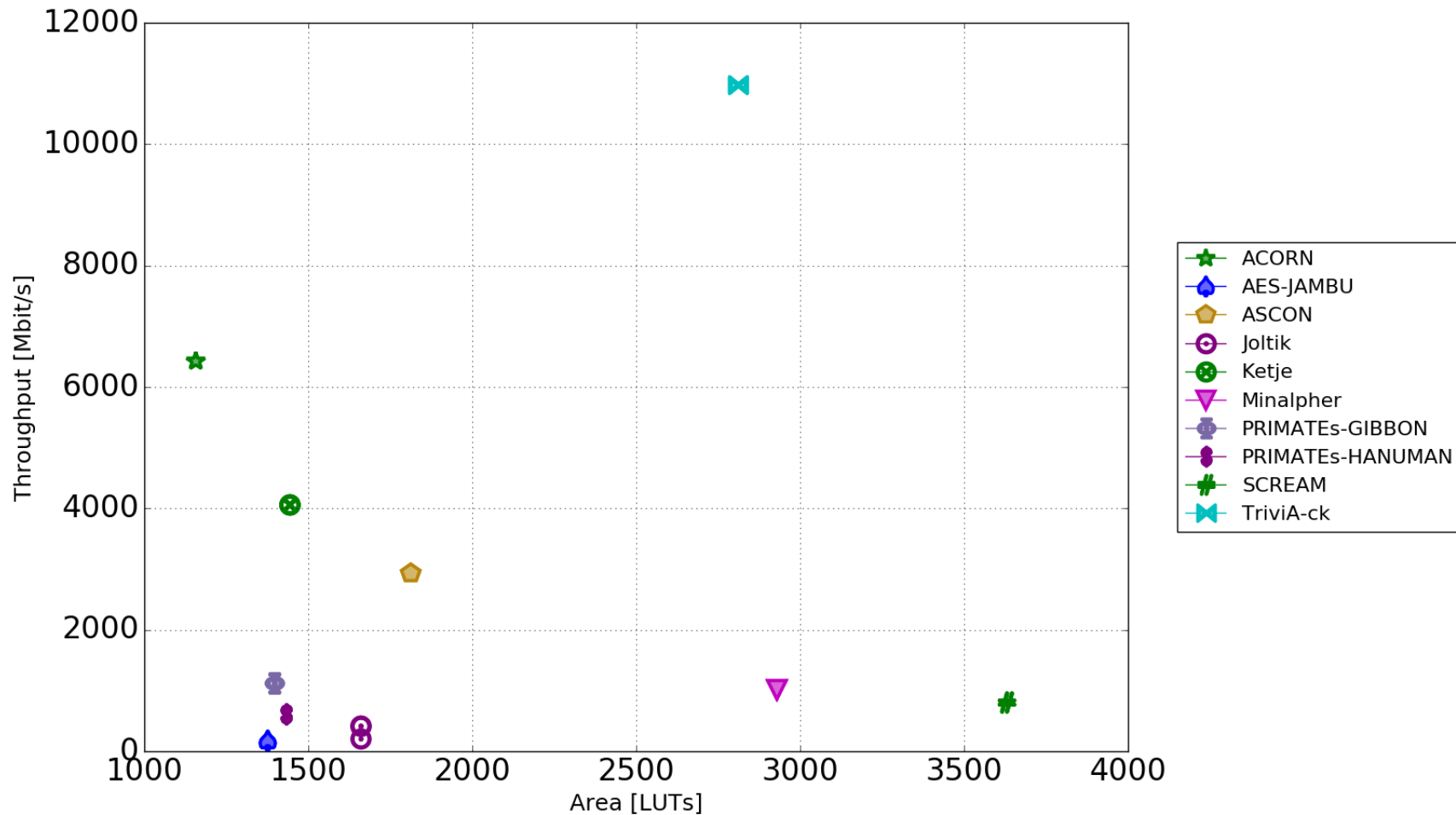
Designs:

- Only Compliant with the CAESAR Hardware API

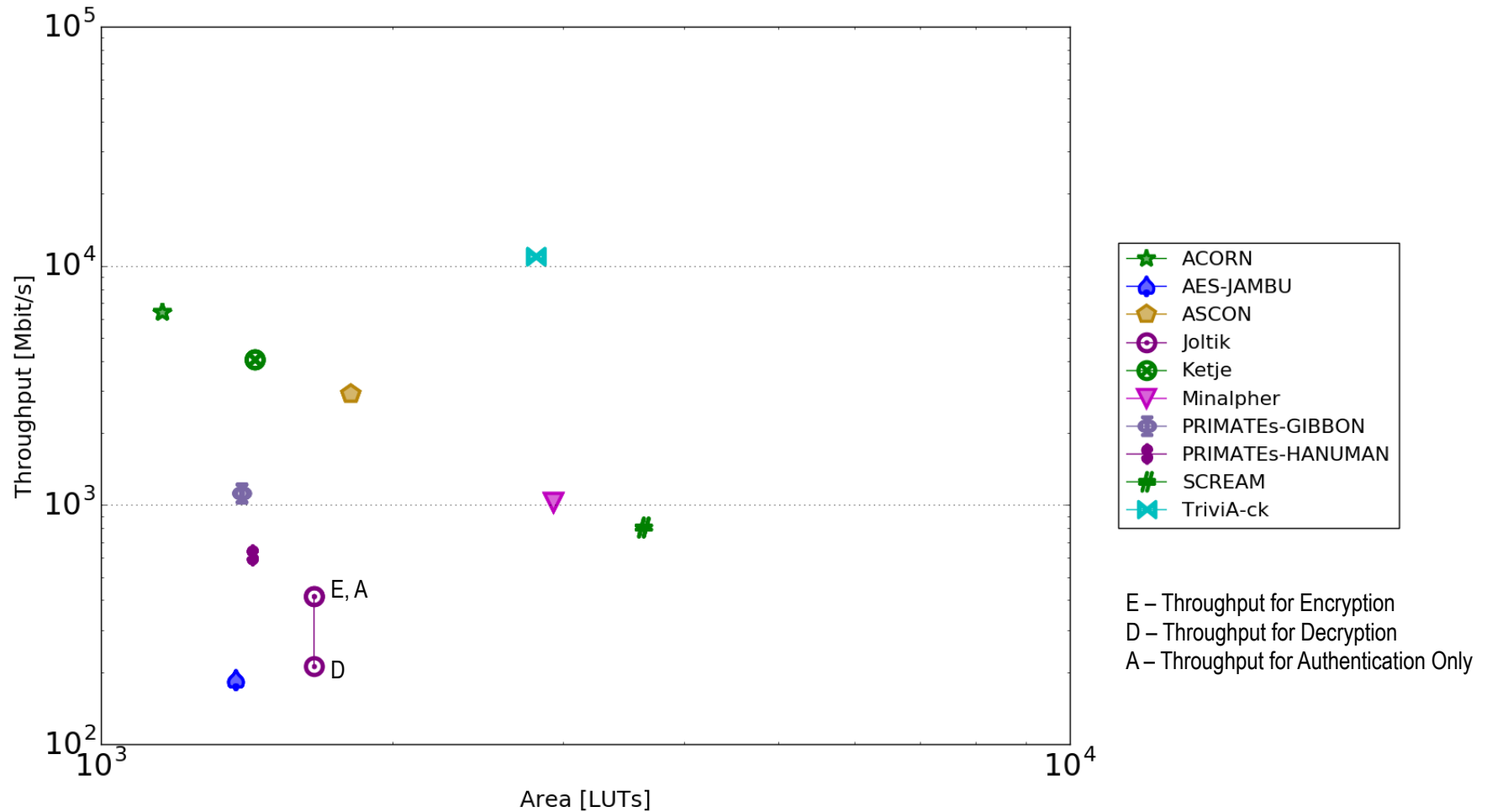
Spartan-6

Results for Spartan 6 – Throughput vs. Area

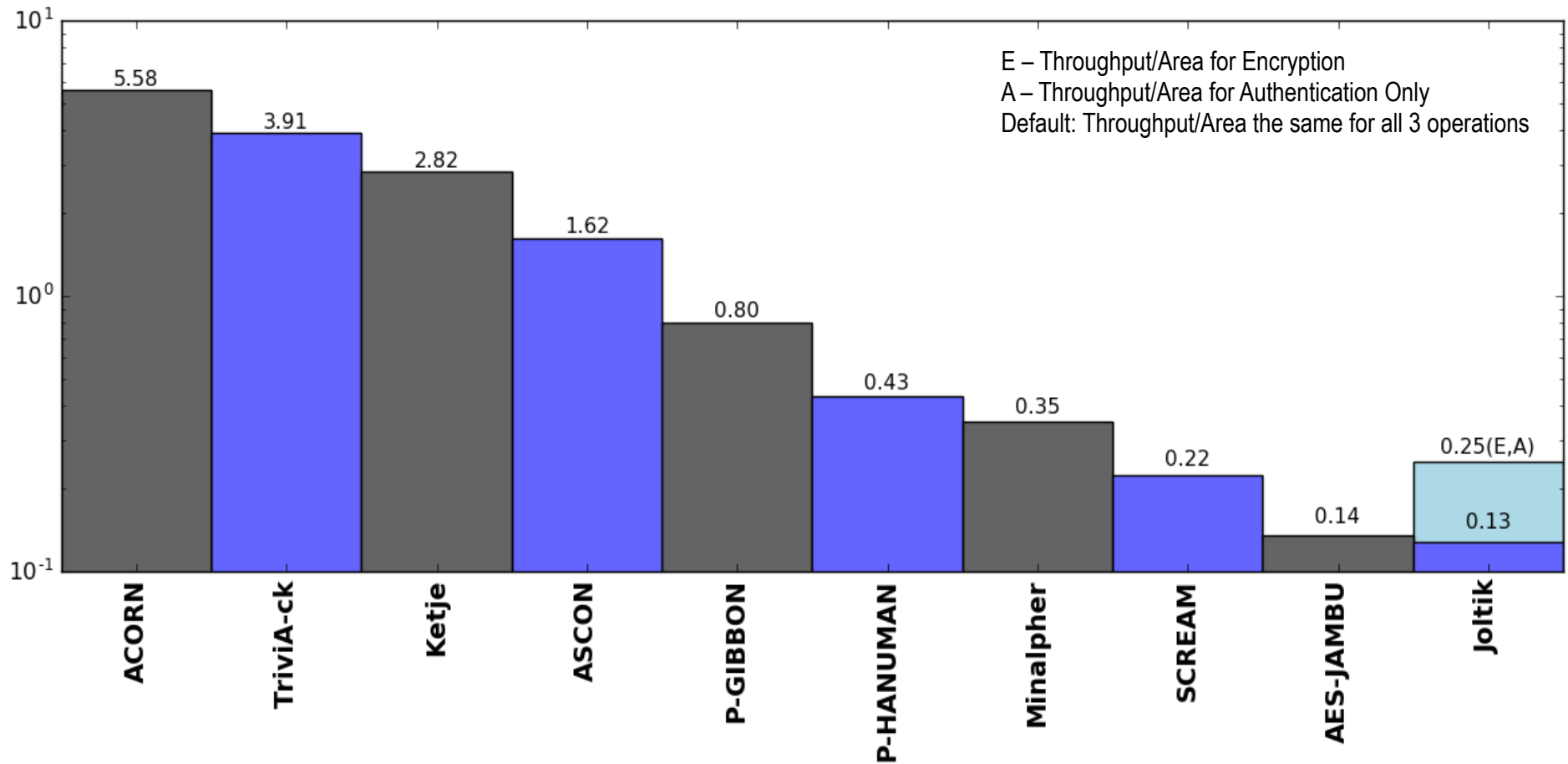
Linear Scale



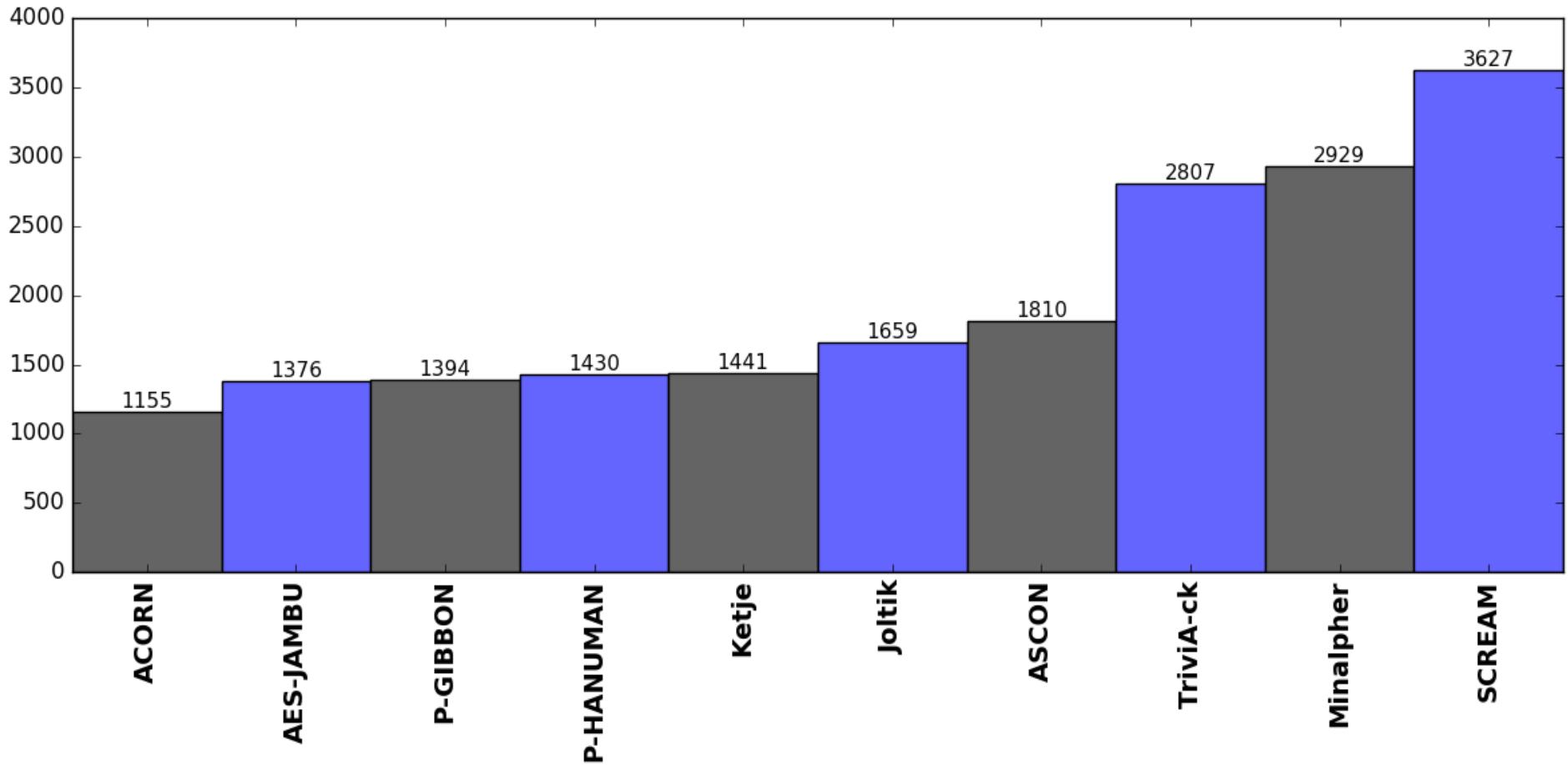
Results for Spartan 6 – Throughput vs. Area Logarithmic Scale



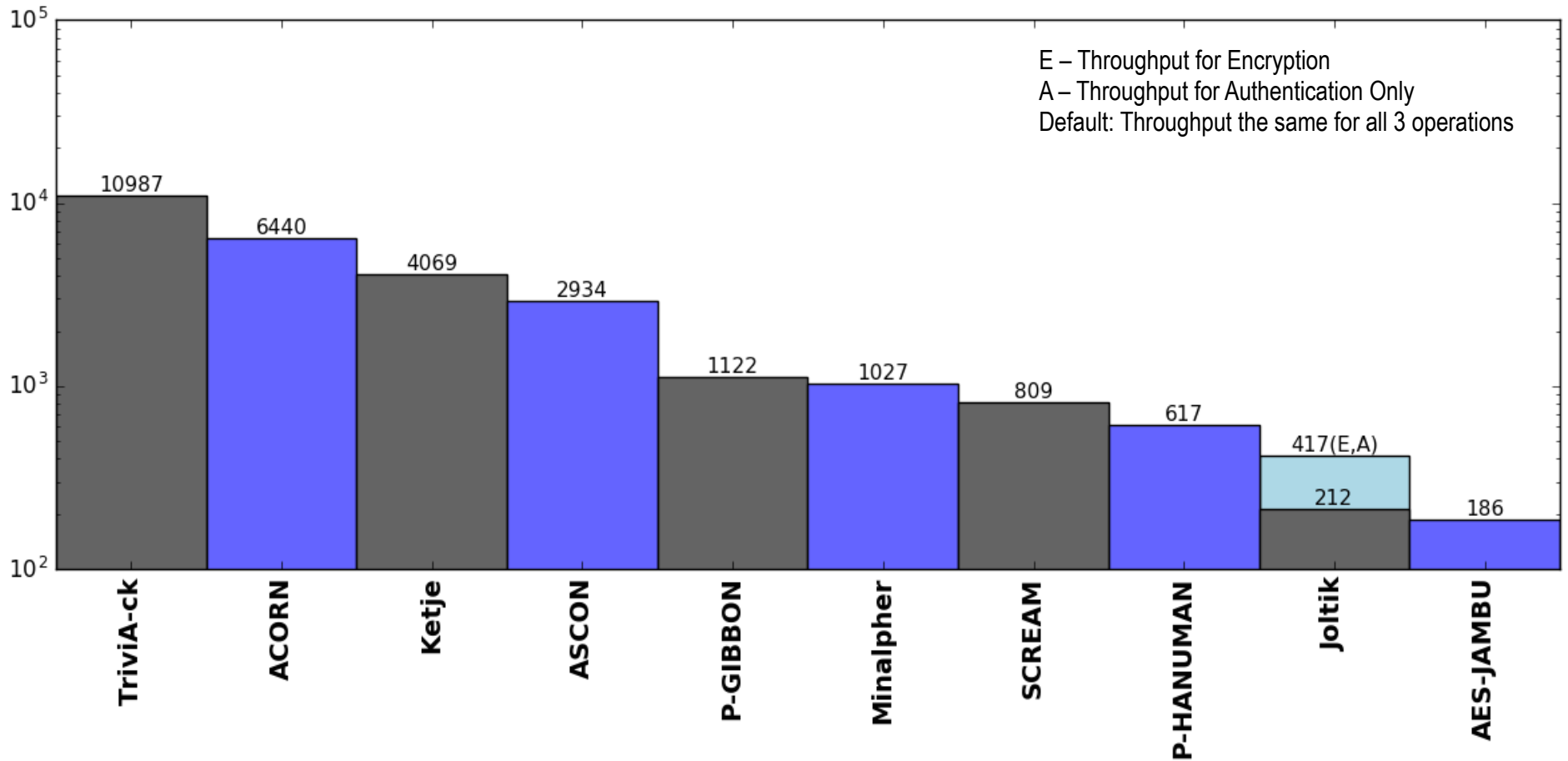
Absolute Throughput/Area [(Mbit/s)/LUT] in Spartan 6



Absolute Area [LUTs] in Spartan 6



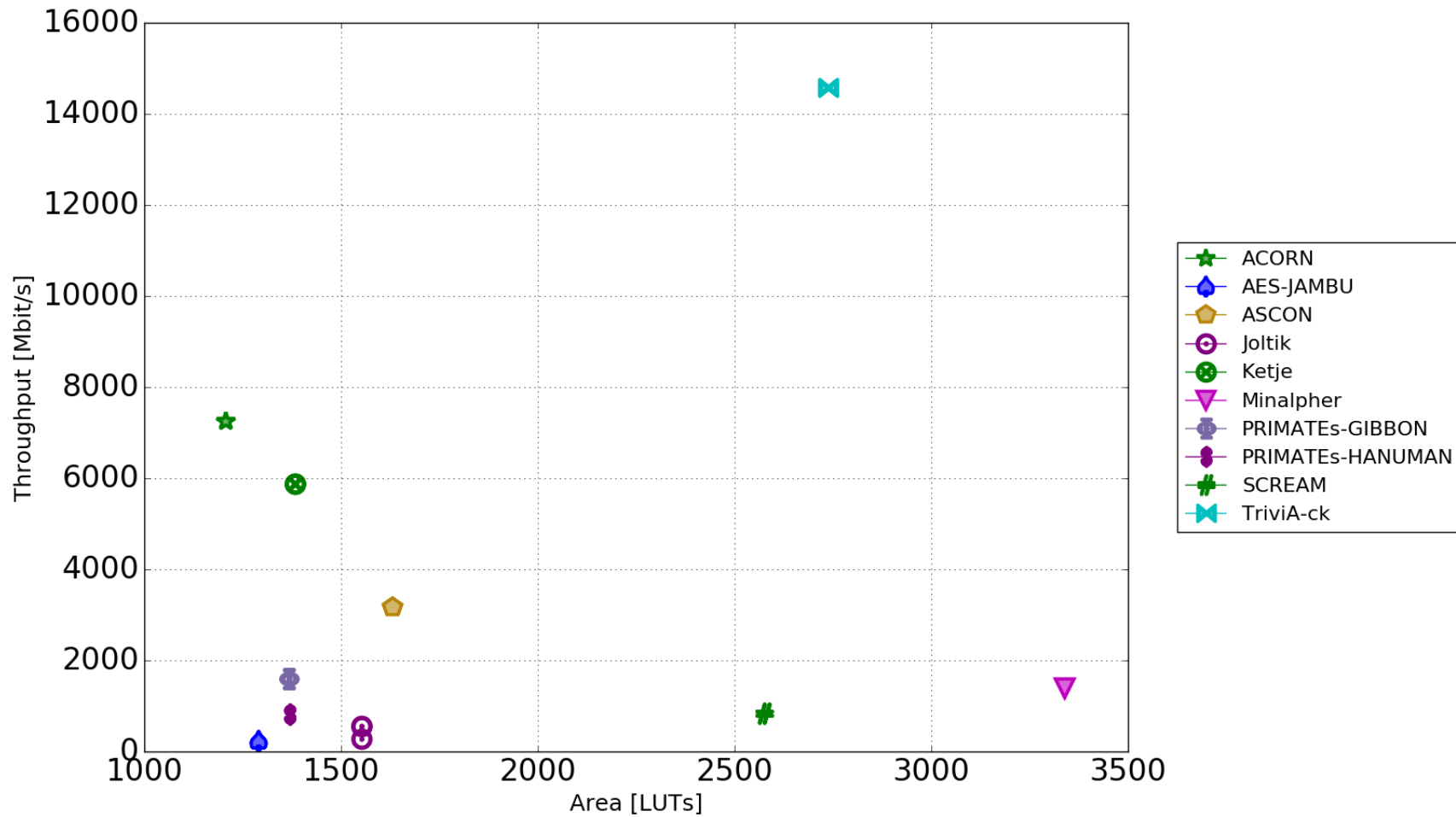
Absolute Throughput [Mbits/s] in Spartan 6



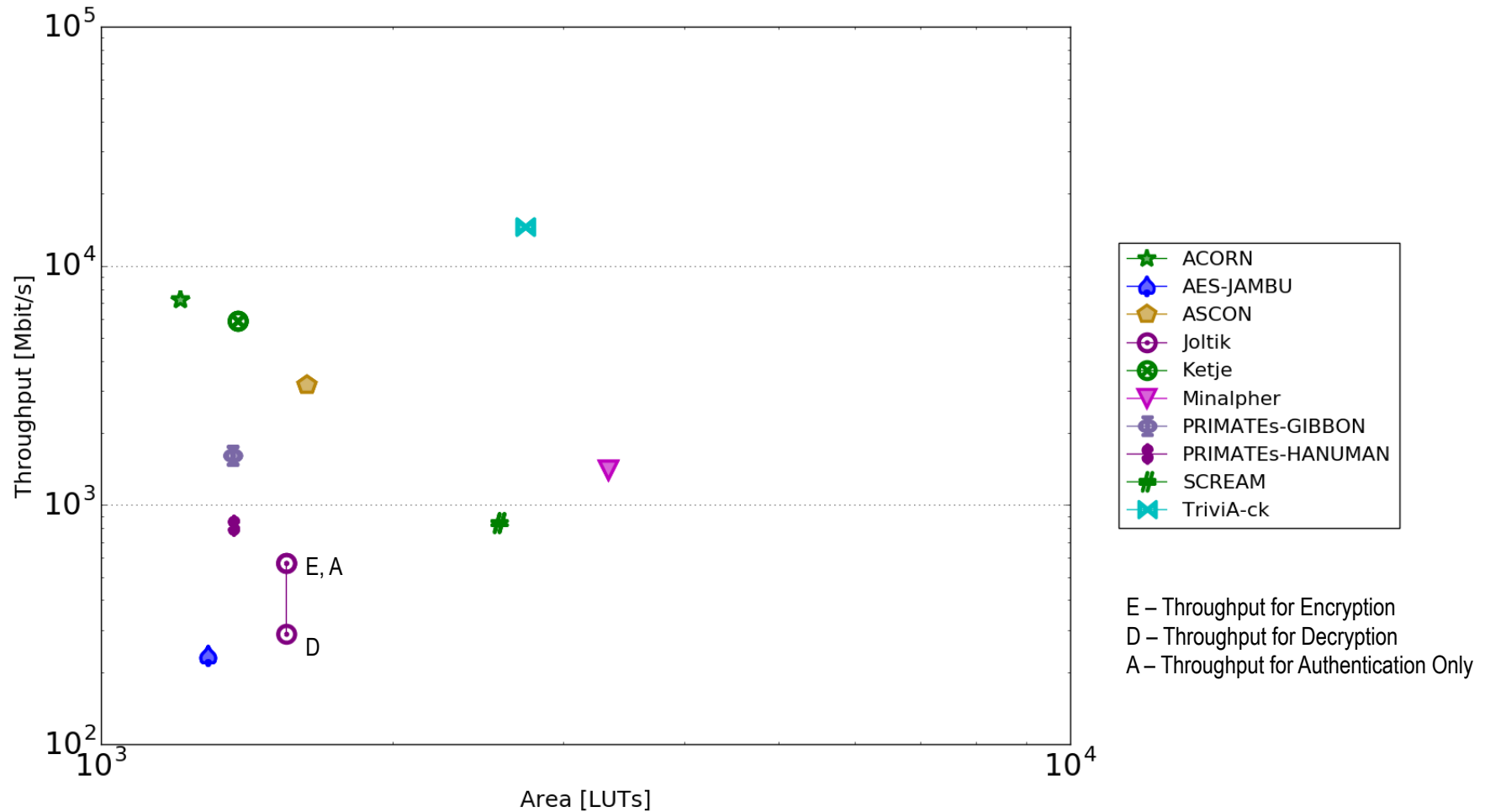
Artix-7

Results for Artix 7 – Throughput vs. Area

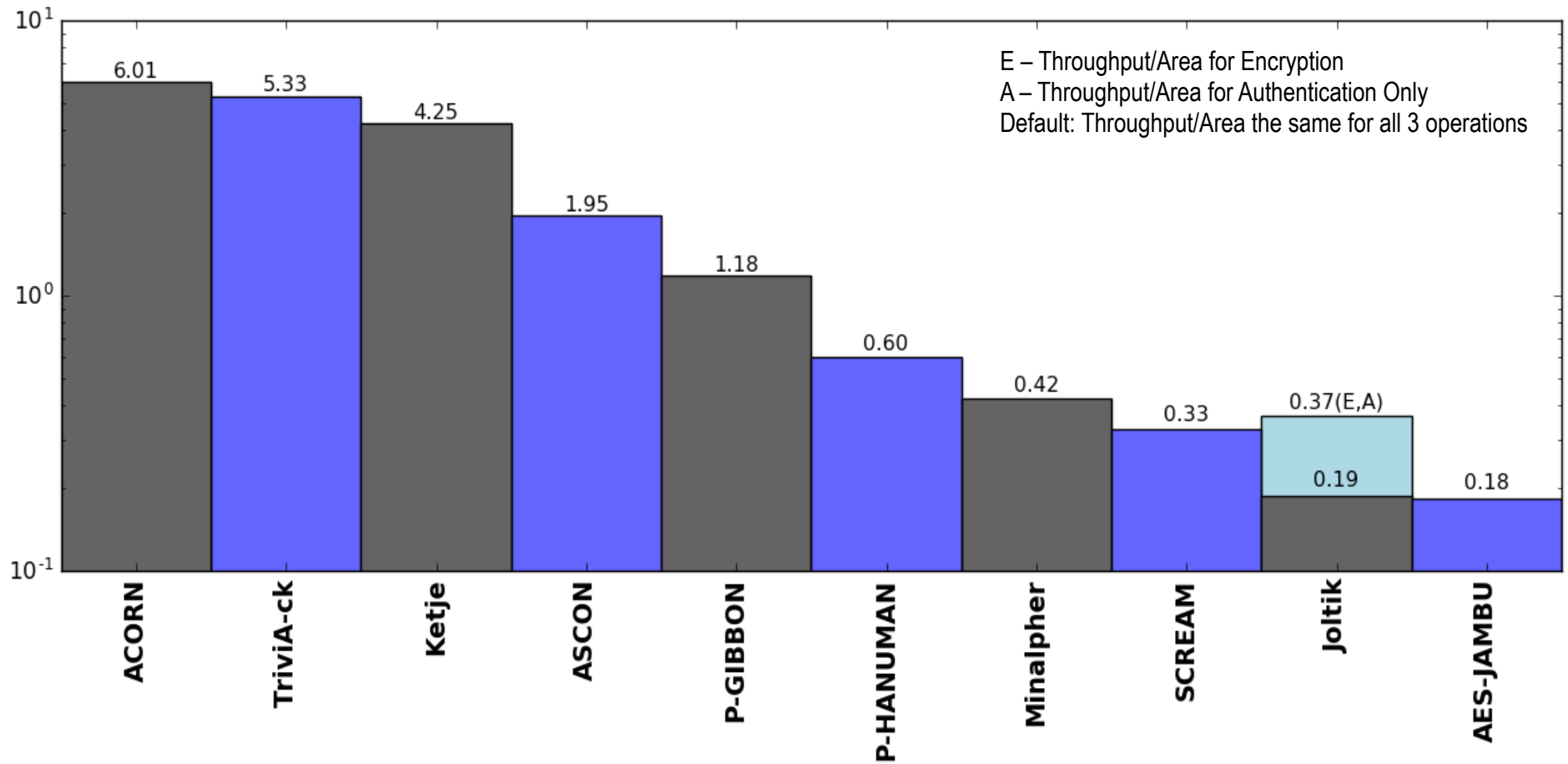
Linear Scale



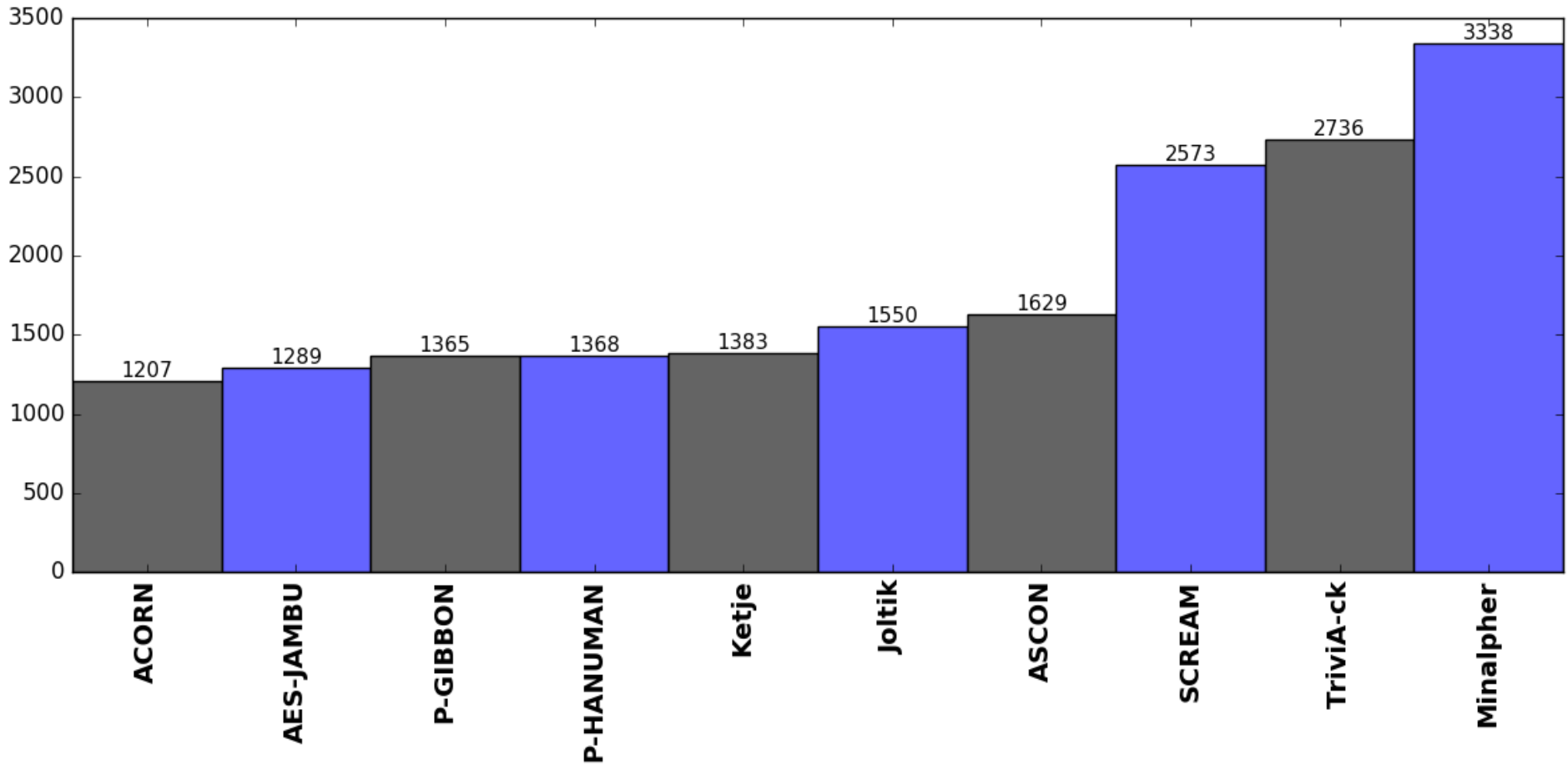
Results for Artix 7 – Throughput vs. Area Logarithmic Scale



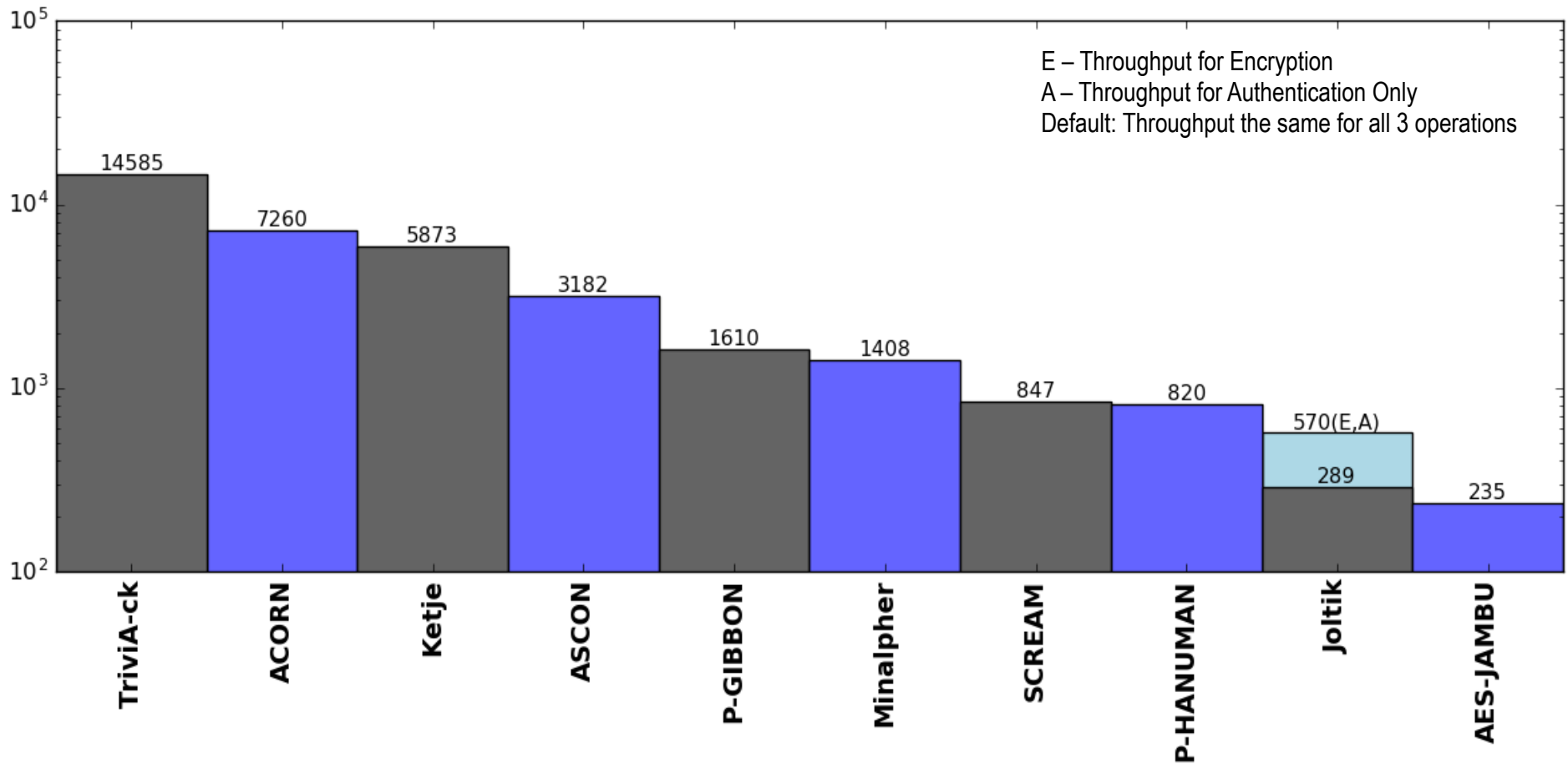
Absolute Throughput/Area [(Mbit/s)/LUT] in Artix 7



Absolute Area [LUTs] in Artix 7



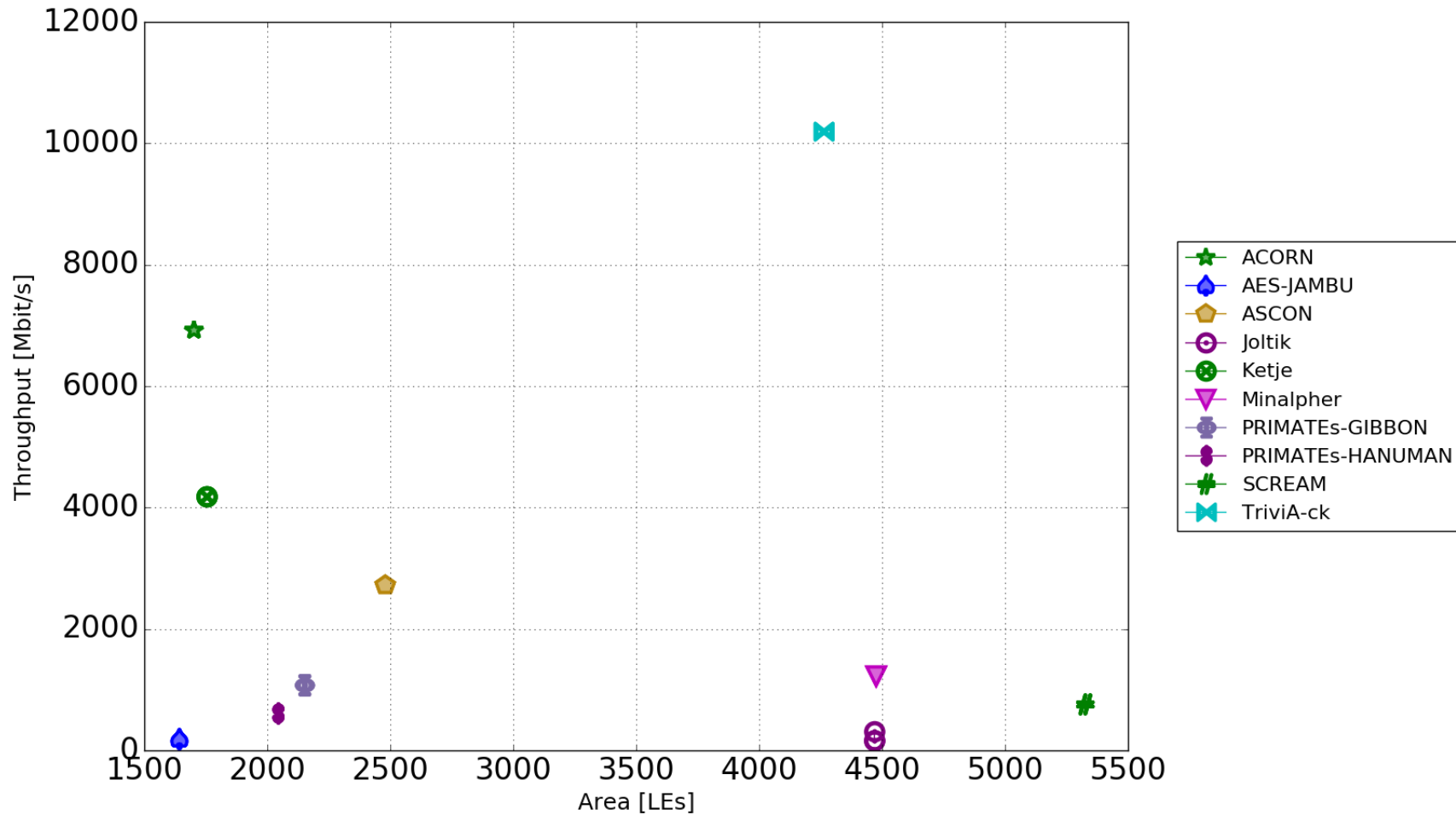
Absolute Throughput [Mbits/s] in Artix 7



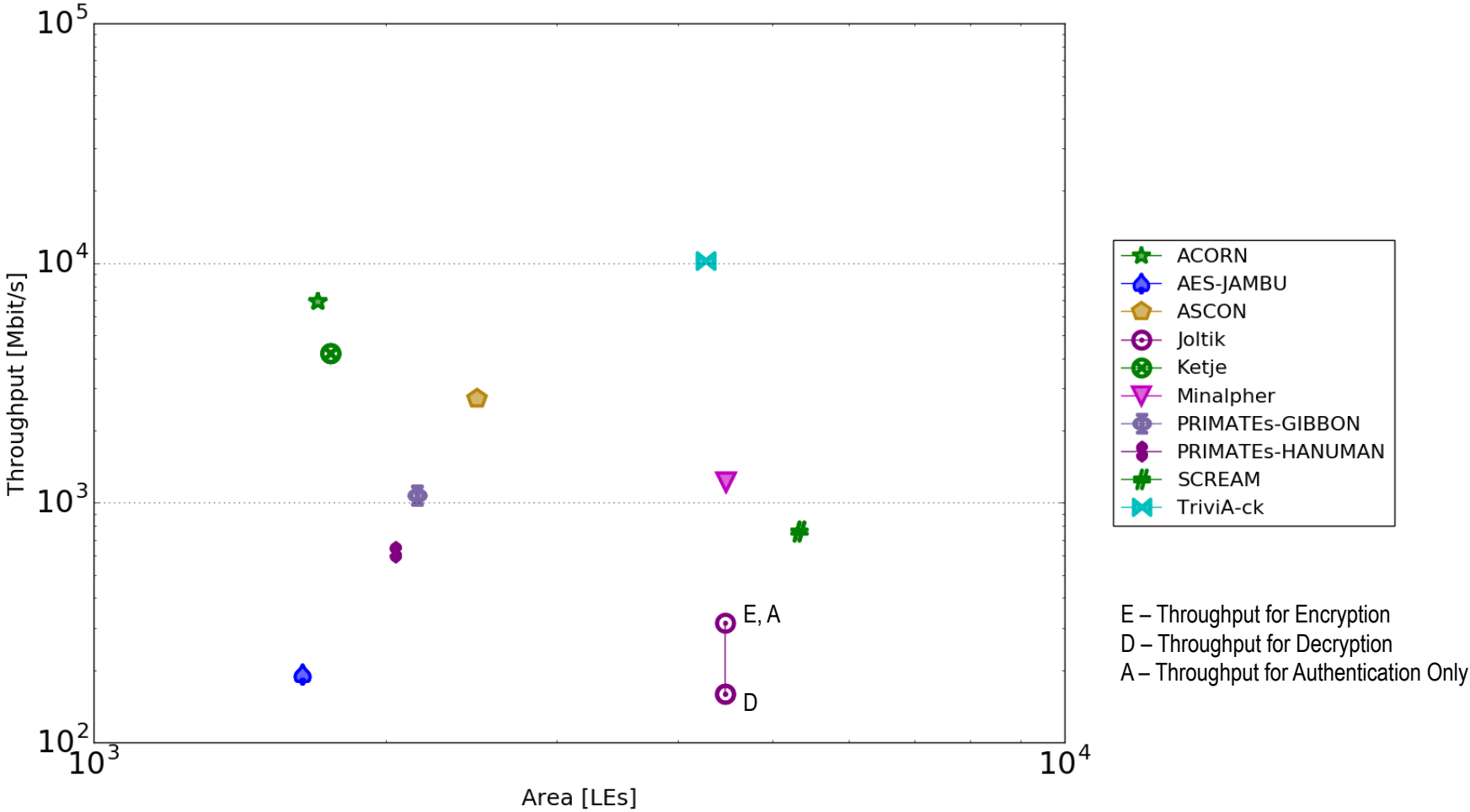
Cyclone IV

Results for Cyclone IV – Throughput vs. Area

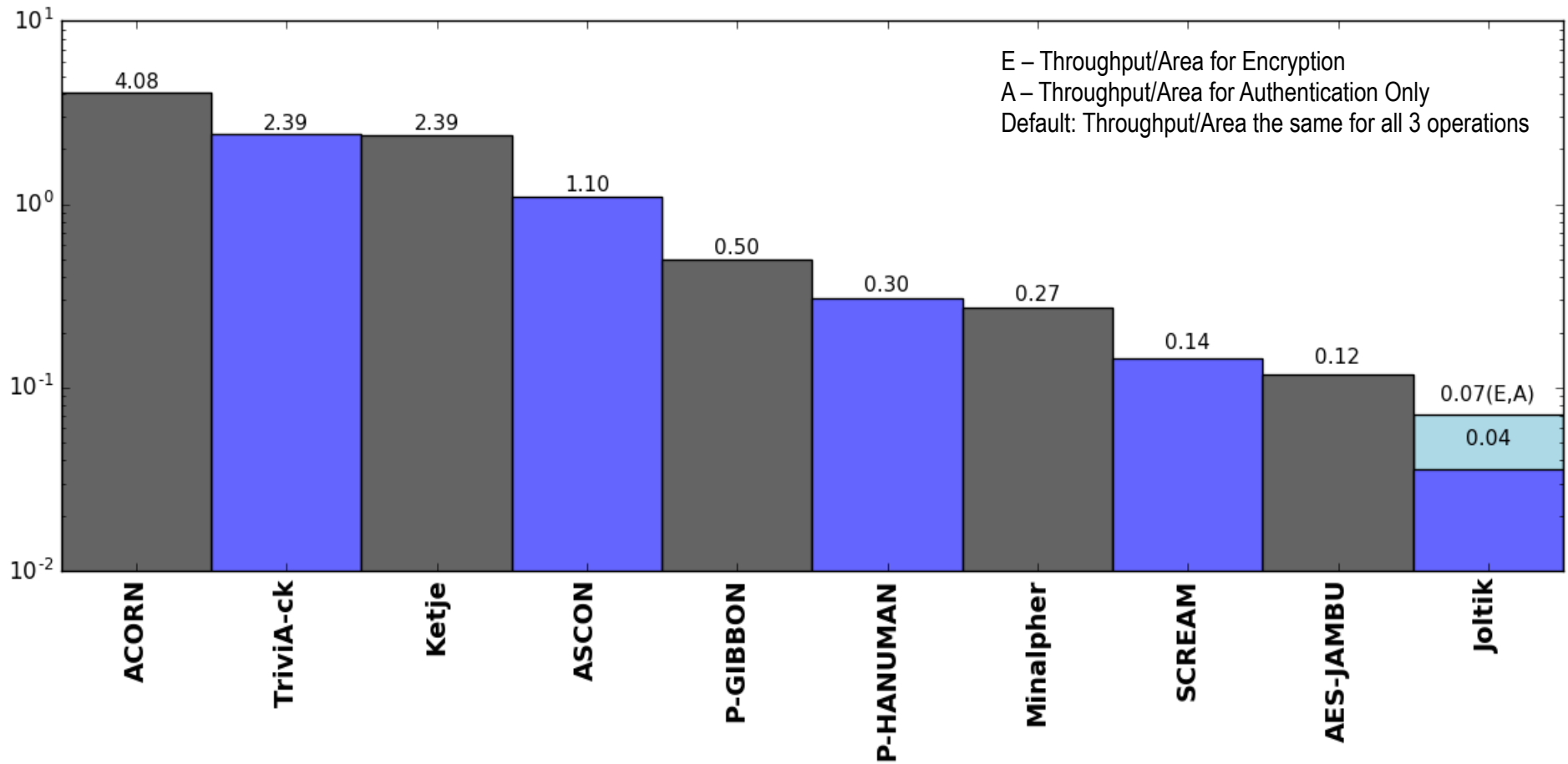
Linear Scale



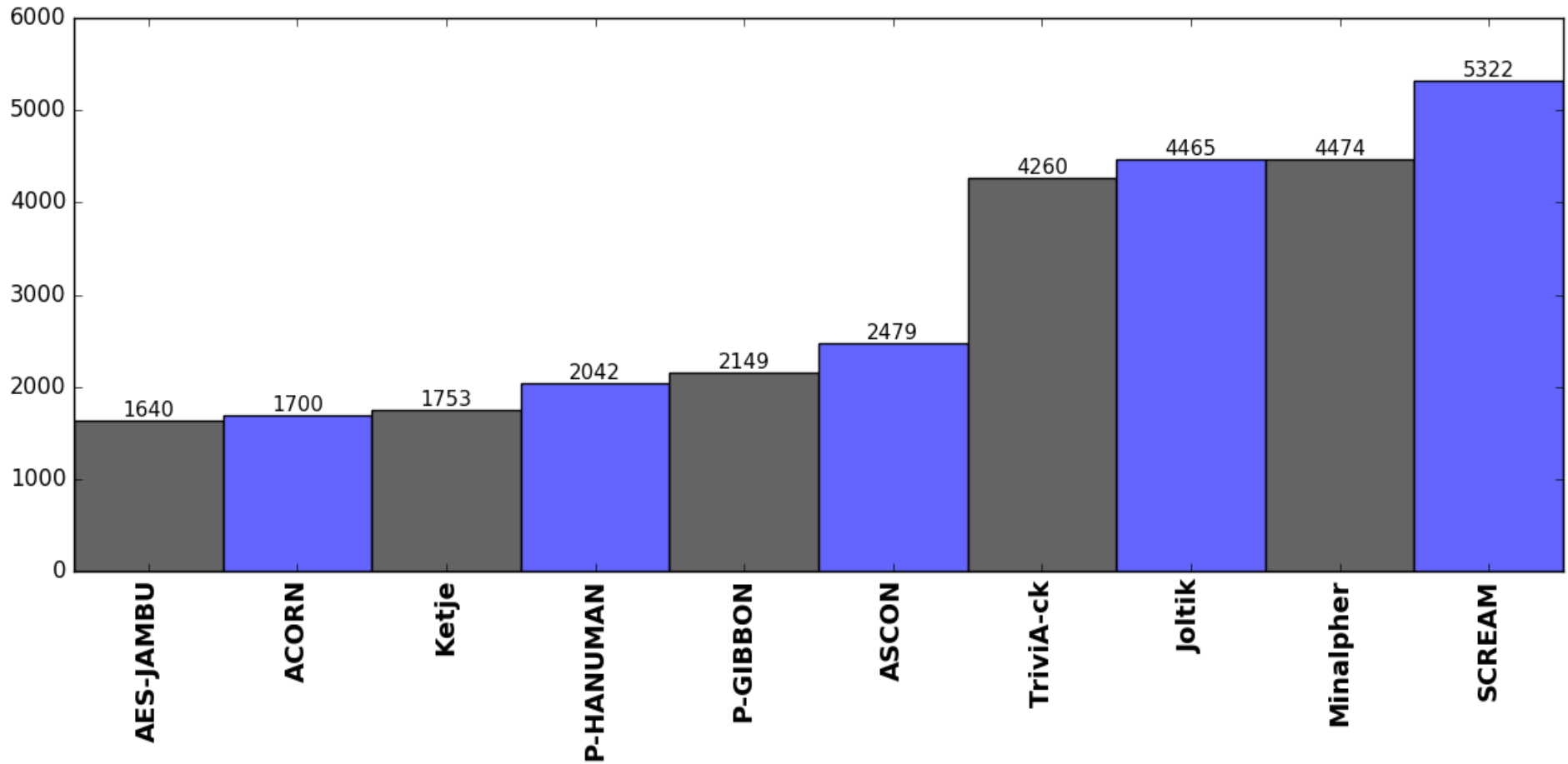
Results for Cyclone IV – Enc/Dec Throughput vs. Area Logarithmic Scale



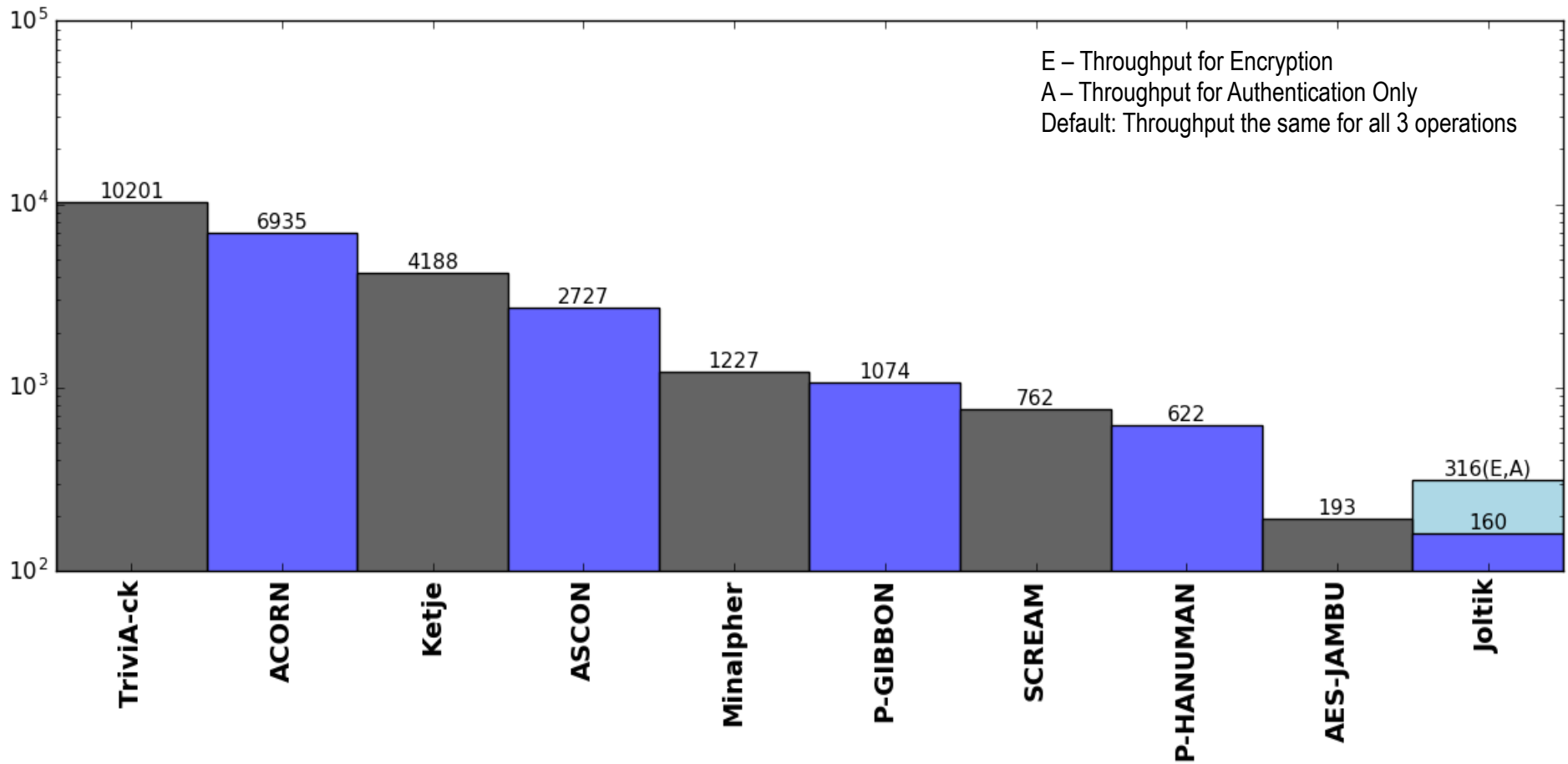
Absolute Throughput/Area [(Mbit/s)/LE] in Cyclone IV



Absolute Area [LEs] in Cyclone IV

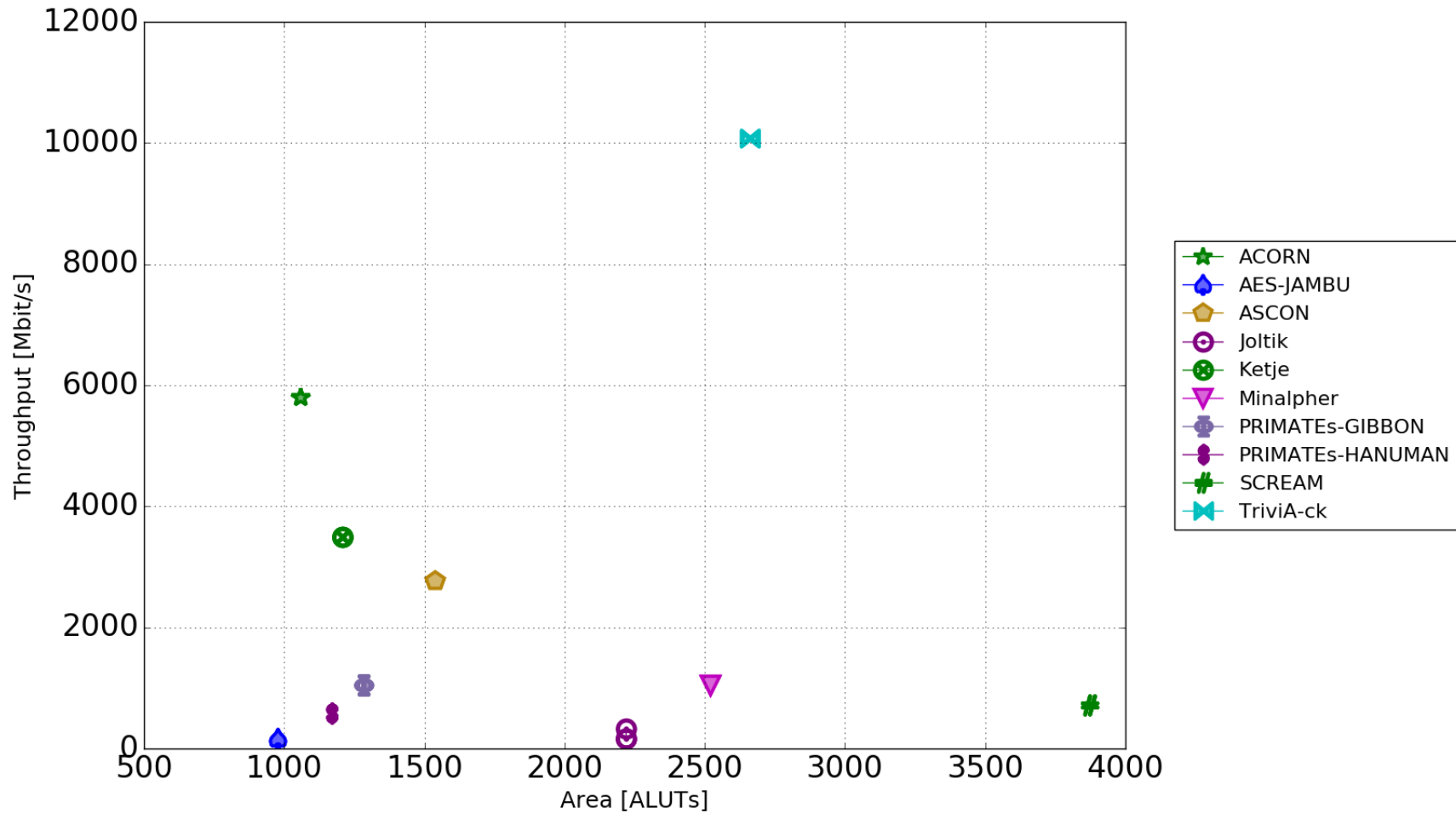


Absolute Throughput [Mbits/s] in Cyclone IV

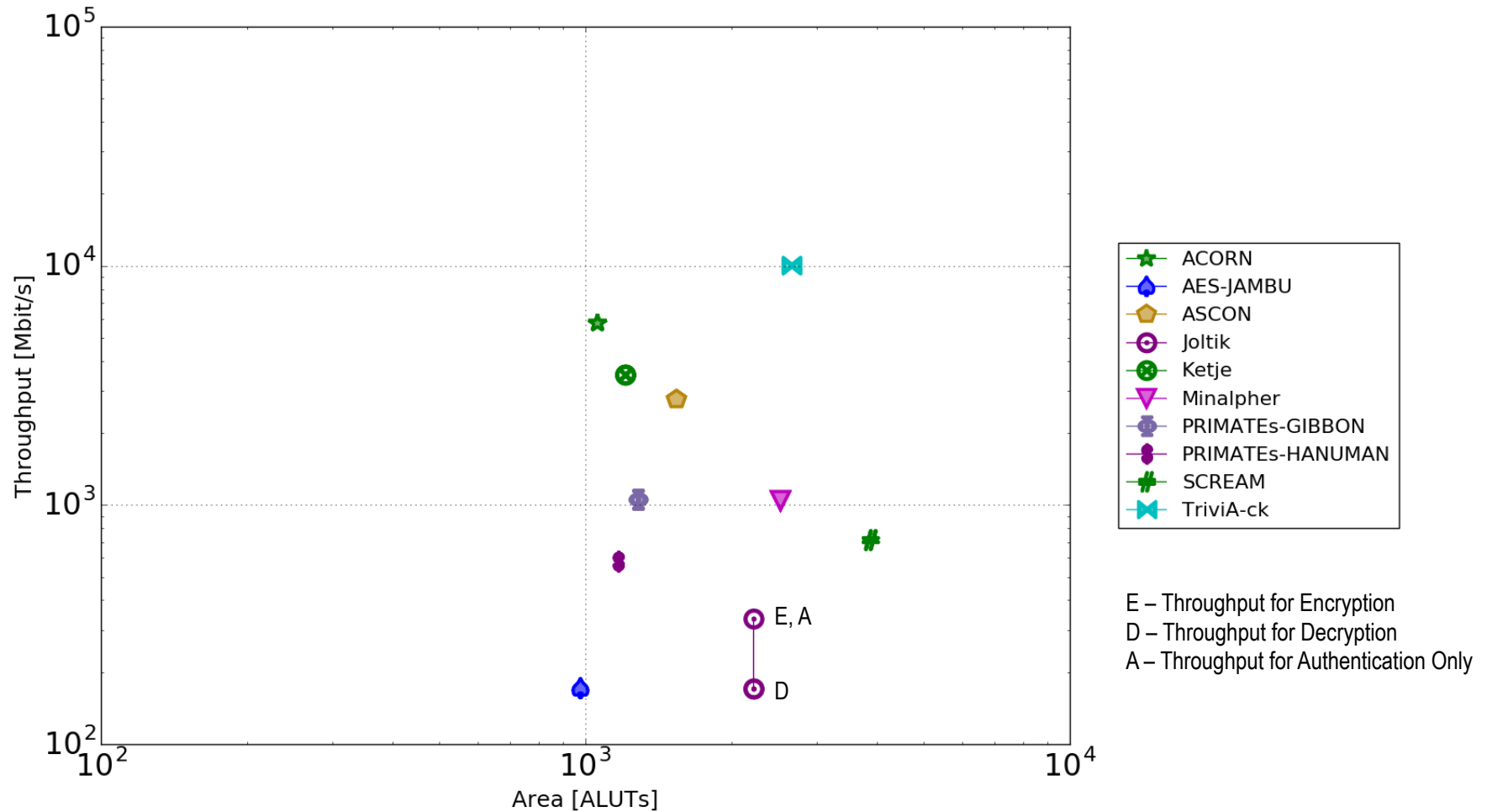


Cyclone V

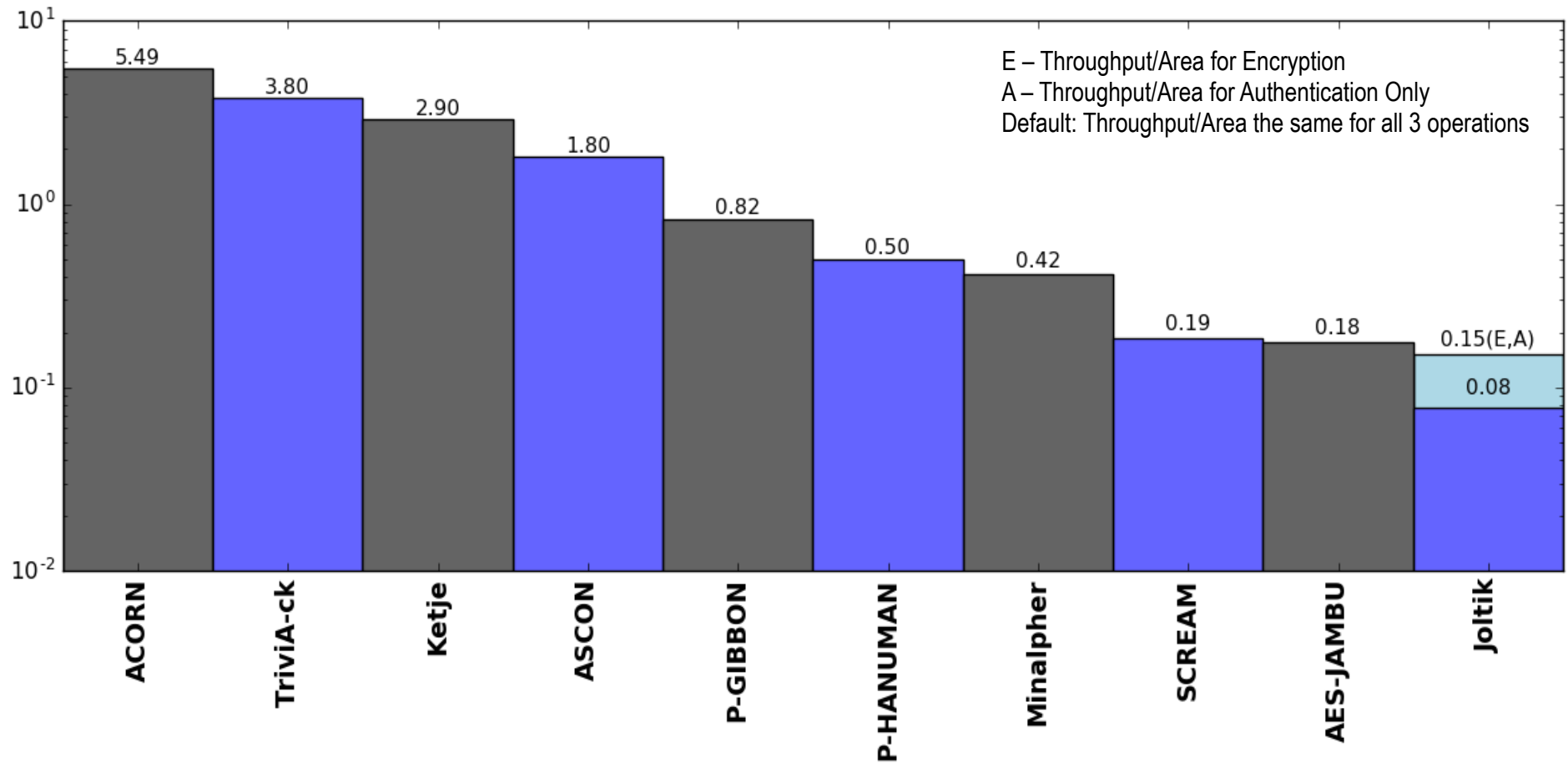
Results for Cyclone V – Throughput vs. Area Linear Scale



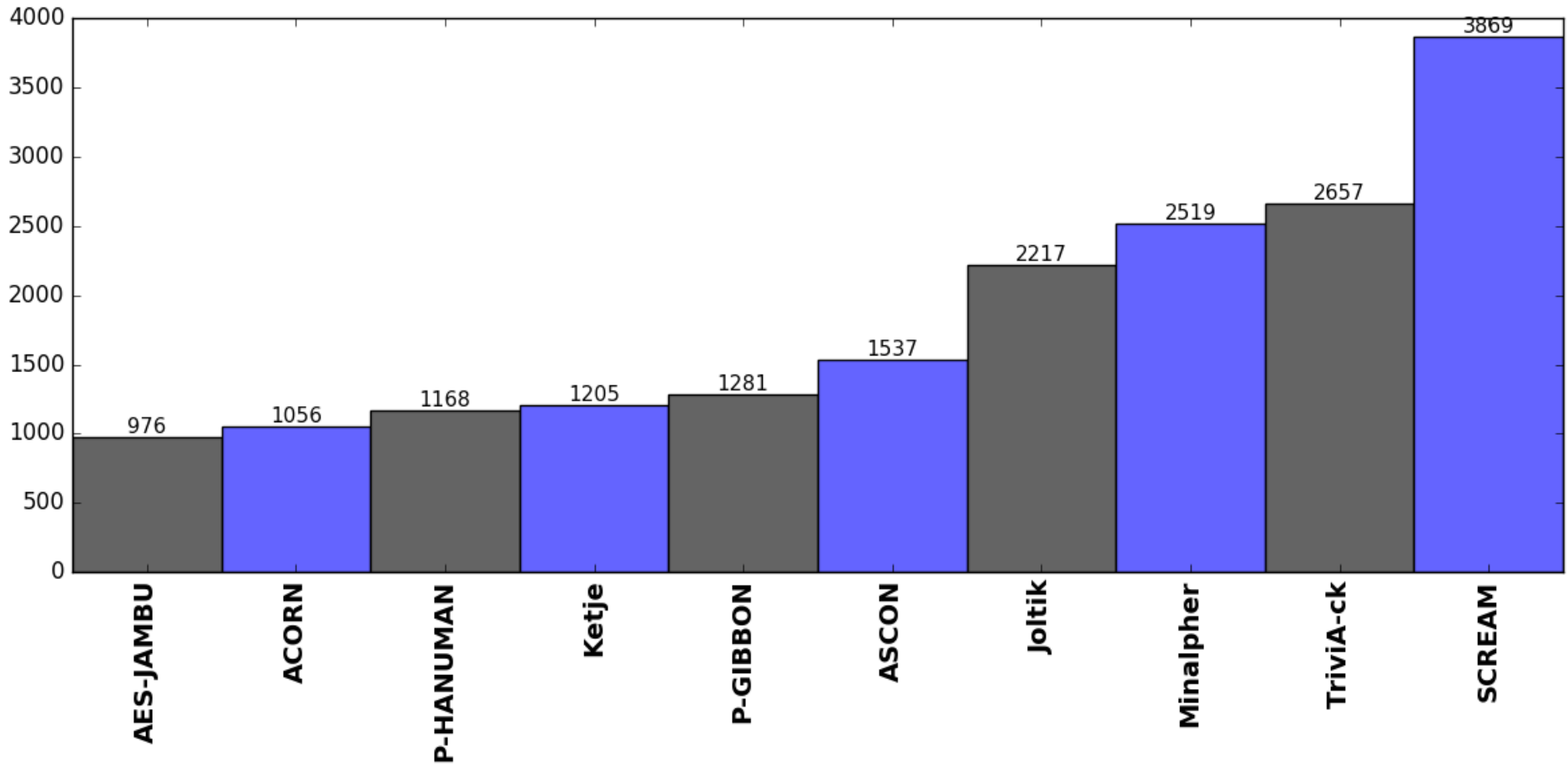
Results for Cyclone V – Throughput vs. Area Logarithmic Scale



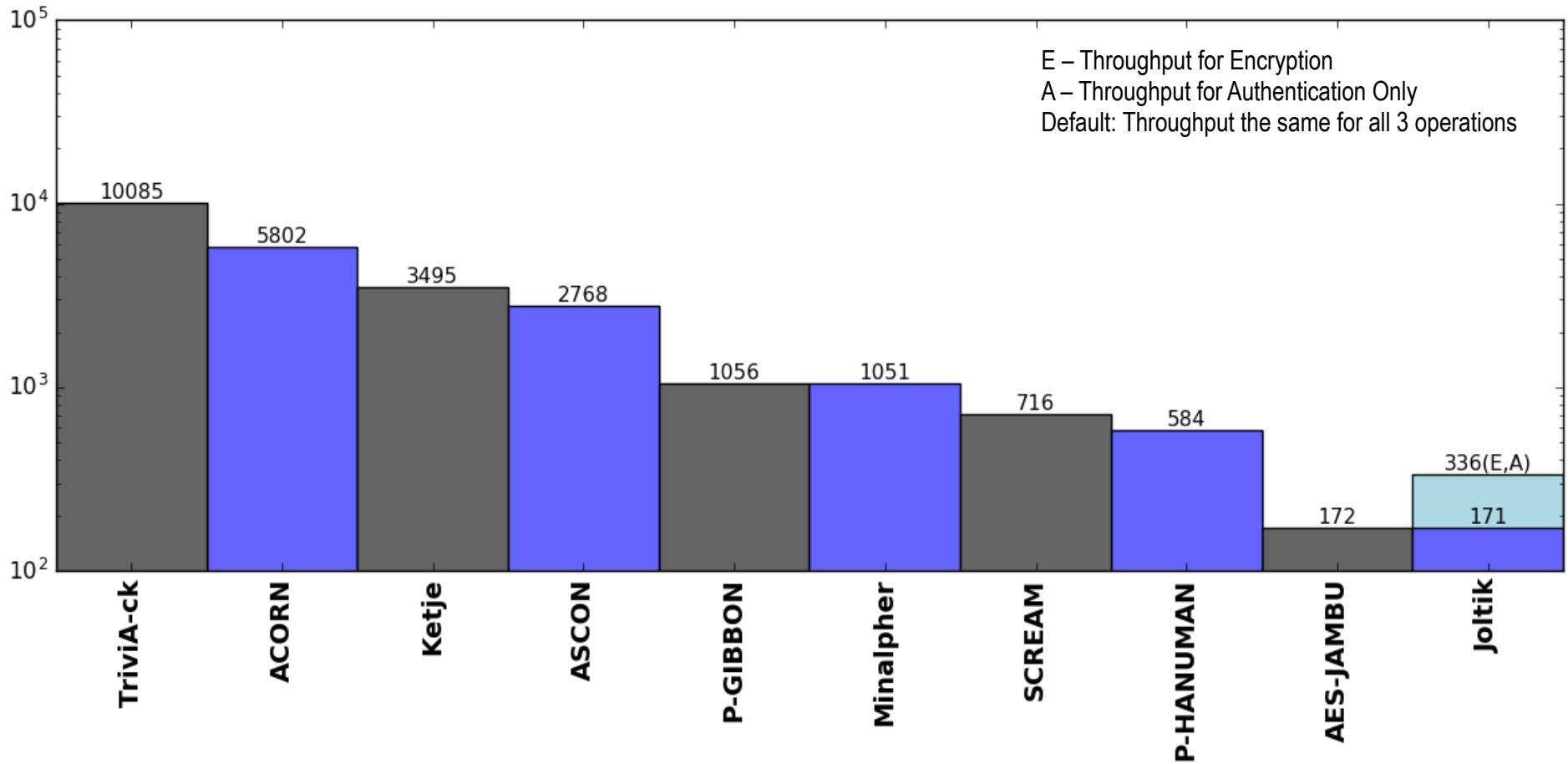
Absolute Throughput/Area [(Mbit/s)/ALUT] in Cyclone V



Absolute Area [ALUTs] in Cyclone V



Absolute Throughput [Mbits/s] in Cyclone V



ATHENa Database of Results

ATHENa Database of Results

- Available at <http://cryptography.gmu.edu/athena>
- Developed by **John Pham**, a Master's-level student of **Jens-Peter Kaps** as a part of the **SHA-3 Hardware Benchmarking project, 2010-2012**, (sponsored by NIST)
- In June 2015 extended to support **Authenticated Ciphers**

Two Views

- **Rankings View**
 - Easier to use
 - Provides Rankings
 - Only the best representative of each family/ the best variant shown (based on the ranking criteria)
- **Table View**
 - More comprehensive
 - Allows close investigation of all designs & comparative analysis
 - Geared toward more advanced users
 - On-line help

Hints on Using the Rankings View

- After each change of options, click on **Update**
- If you want to return to the default settings, please click on **FPGA Rankings**,
in the menu located on the left side of the page
- If you want to limit the key size to a particular range, please choose the option
Key size:
From <min> To: <max>
- You can further narrow down the search by using
Min Area:
Max Area:
Min Throughput:
Max Throughput:

Hints on Using the Rankings View

- For the results of High-Speed Benchmarking, choose **Family:**
 - **Virtex 6 (default)**
 - **Virtex 7**
 - **Stratix IV**
 - **Stratix V**
- For the very preliminary results of Lightweight Benchmarking, choose **Family:**
 - **Spartan 6**
 - **Artix 7**
 - **Cyclone IV**
 - **Cyclone V**

Hints on Using the Rankings View

- You can switch between ranking criteria by using the option:

Ranking:

Throughput/Area

Throughput

Area

- **Unit of Area:**

allows you to choose between two alternative units of area for each type of FPGA:

- for Xilinx Virtex 6, Virtex 7, Spartan 6, and Artix 7: **LUTs and Slices**
- for Altera Stratix IV, Stratix V, and Cyclone V: **ALUTs and ALMs.**

Please note that after each change a different variant may be used to represent a given family of authenticated ciphers.

The displayed variant is the best in terms of the current ranking criteria.

Hints on Using the Rankings View

- In order to include in the rankings any implementations that are non-compliant with the CAESAR Hardware API, please mark under

Hardware API:

Full-Block width (custom)

on top of

CAESAR Hardware API v1

Please keep in mind that making this change may lead to an unfair ranking, as the non-compliant designs may have an incomplete functionality, and typically do not support the CAESAR API Communication Protocol

One Stop Website

<https://cryptography.gmu.edu/athena/index.php?id=download>

OR

<https://cryptography.gmu.edu/athena>
and click on Download

- VHDL/Verilog Code of CAESAR Candidates: Summary I
- VHDL/Verilog Code of CAESAR Candidates: Summary II
- ATHENa Database of Results: Rankings View
- ATHENa Database of Results: Table View
- Benchmarking of Round 2 CAESAR Candidates in Hardware: Methodology, Designs & Results [[this presentation](#)]
- GMU Implementations of Authenticated Ciphers and Their Building Blocks
- CAESAR Hardware API v1.0

Thank you!

Comments?



Questions?

Suggestions?

ATHENa: <http://cryptography.gmu.edu/athena>

CERG: <http://cryptography.gmu.edu>