RUHR-UNIVERSITÄT BOCHUM

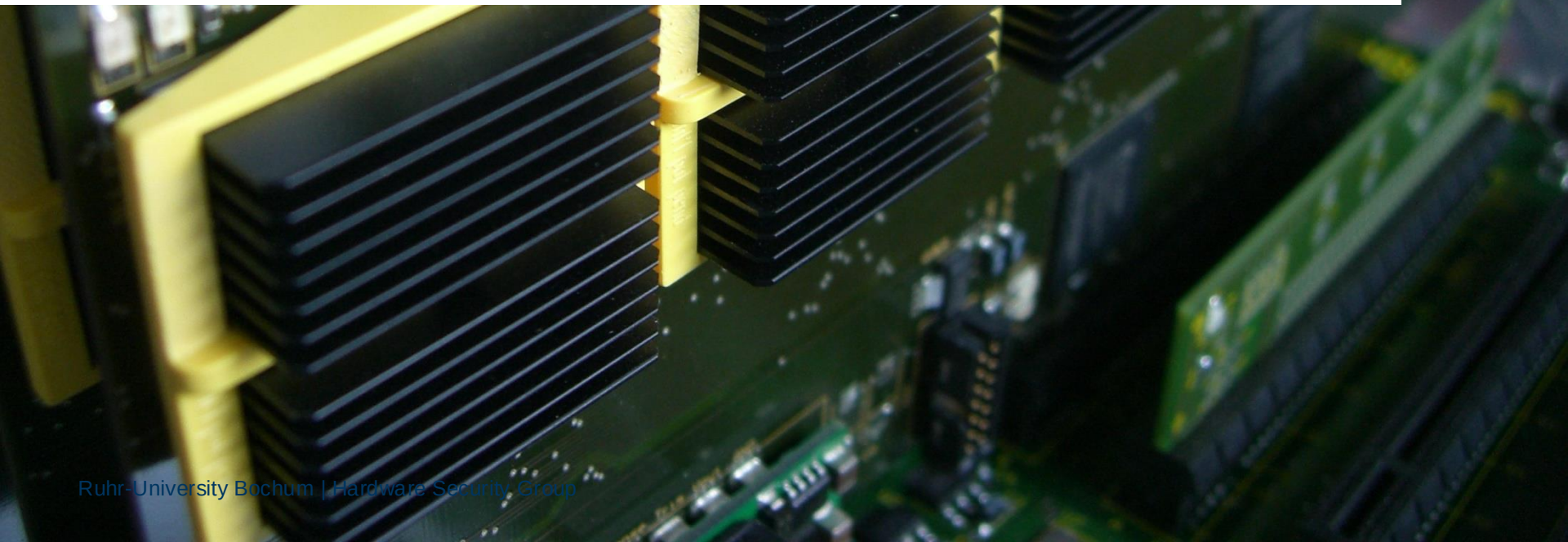
RUB

Efficient Elliptic-Curve Cryptography using Curve25519 on Reconfigurable Devices

Pascal Sasdrich, Tim Güneysu

6/27/2014

20. Kryptotag, Berlin, Deutschland



Efficient Elliptic-Curve Cryptography using Curve25519 on Reconfigurable Devices
Pascal Sasdrich, Tim Güneysu

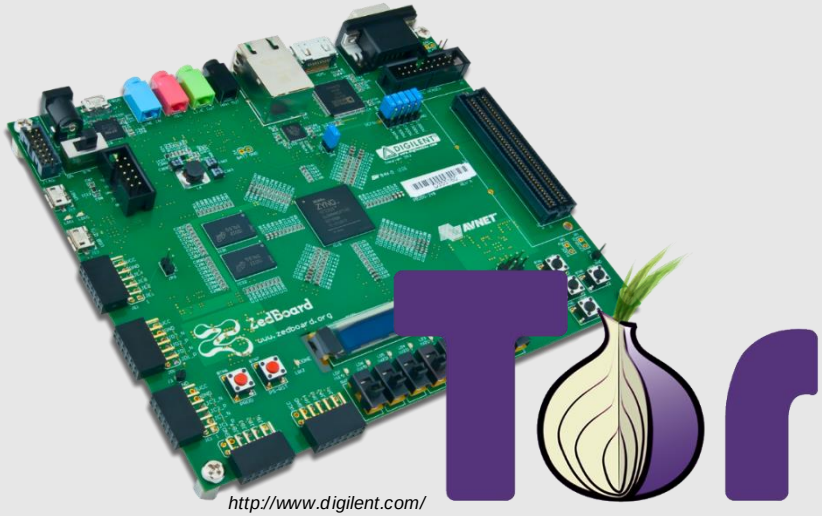
Content

Efficient Elliptic-Curve Cryptography using Curve25519 on Reconfigurable Devices
Pascal Sasdrich, Tim Güneysu

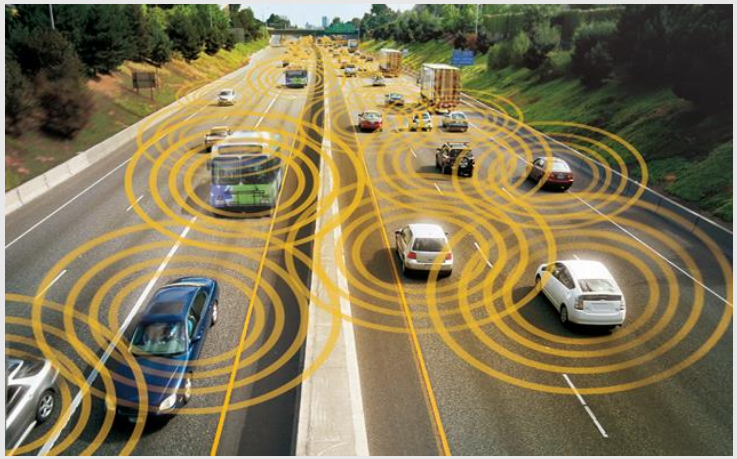
Motivation

Tor-Router

- Zynq FPGA on Zedboard
- router for Tor-network
- privacy protection
- defense against surveillance/analysis



http://www.digilent.com/



http://www.extremetech.com/

Car2Car Communication

- exchange of signed messages
- signature generation using ECC
- authenticity of messages
- protection against manipulation

Contribution

Curve25519 function was chosen for fast **software** implementations.

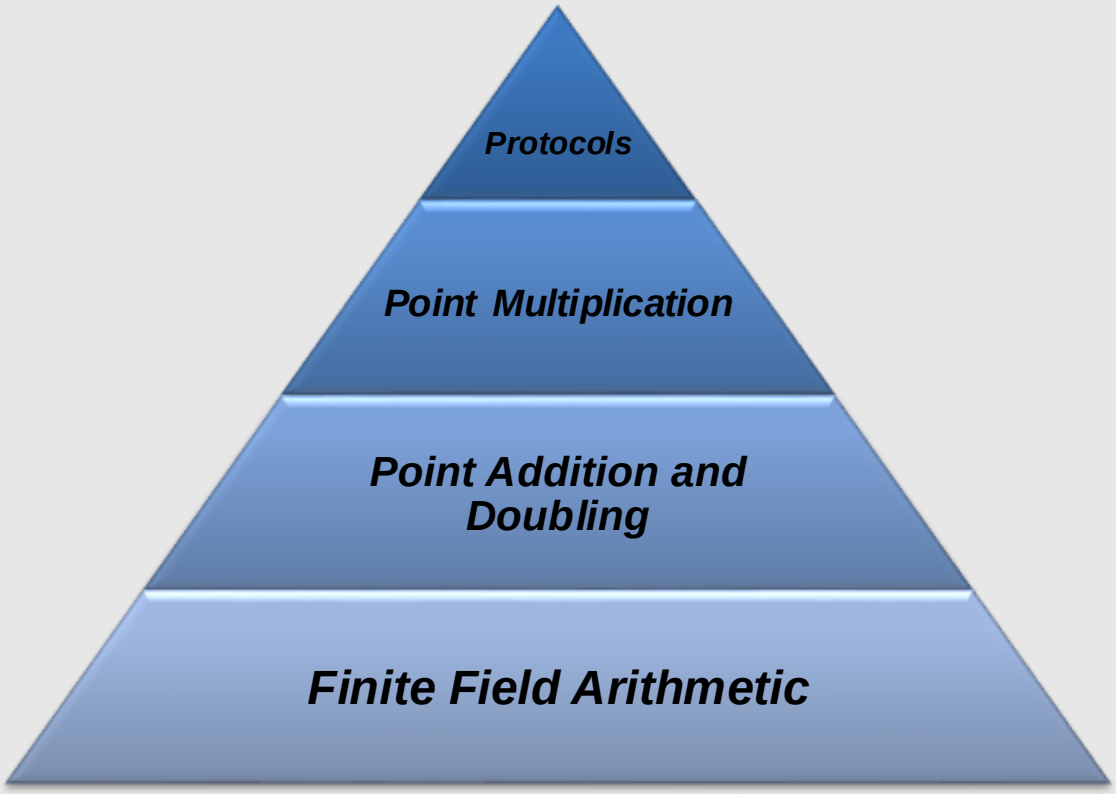
Our Contribution:

1. First efficient high-speed **hardware** implementation of Curve25519.
2. Improvement of implementation through parallelization techniques.
   More than 32000 point multiplications per second.

Content

Efficient Elliptic-Curve Cryptography using Curve25519 on Reconfigurable Devices
Pascal Sasdrich, Tim Güneysu

Elliptic-Curve Cryptography

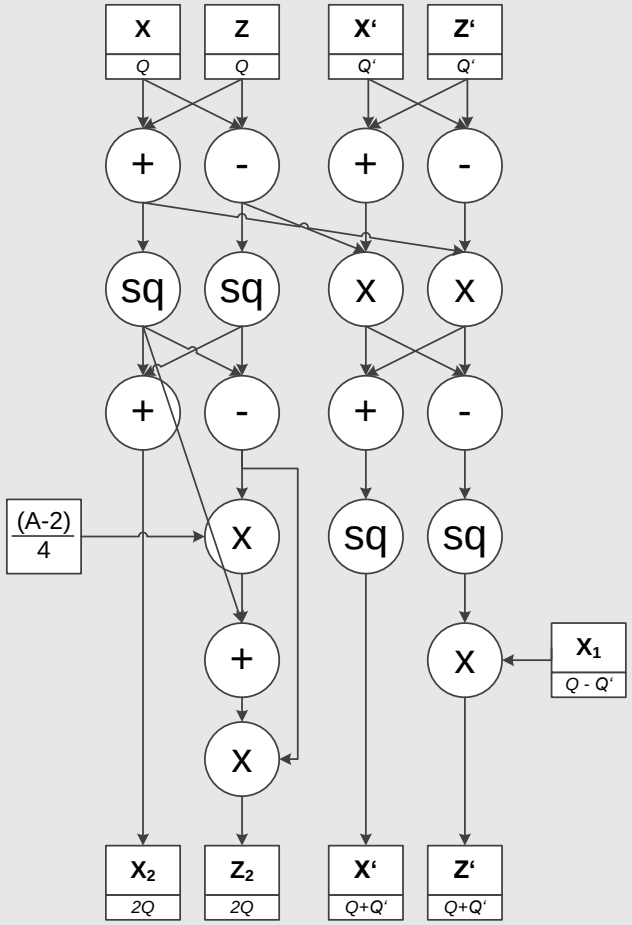


- Diffie-Hellman Key Exchange
- 255 calls of Double-And-Add function, timing constant
- Montgomery's Ladder, (always) Double-And-Add
- Modular reduction mod P, multiplication by 19, Pseudo Mersenne Prime

Efficient Elliptic-Curve Cryptography using Curve25519 on Reconfigurable Devices
Pascal Sasdrich, Tim Güneysu

Curve25519 Function

- **Inputs:**
  - k (secret key, 255-bit)
  - P (public point, affine coordinates x and y)
- **Computation:**
  - compute k x P in 255 steps
  - scan k bitwise and call double-and-add algorithm
  - swap inputs X/Z and X'/Z' depending on scanned bit
- **Output:**
  - requires conversion from projective to affine coordinates
  - inversion of Z with Fermat's Little Theorem ($Z^{-1} = X^{P-2}$)
  - return result $R = X \times Z^{-1}$



Content

Design Considerations

- **Target Platform:**

  latest Xilinx FPGA generation Zynq XC7Z020 on Digilent ZedBoad

- **FPGA provides many DSP slices:**

  dedicated processors for integer operations

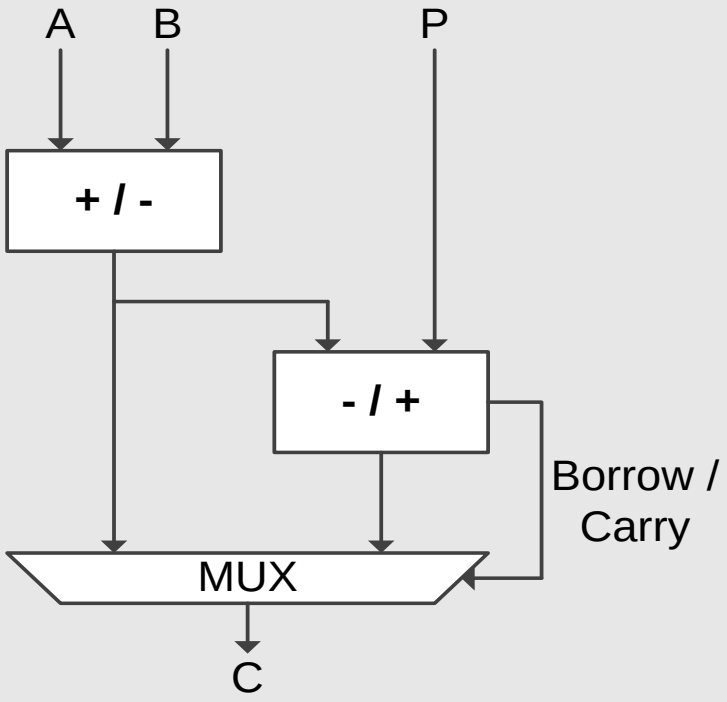
- **DSP addition and multiplication stages:**

  48-bit addition and 25x18-bit signed multiplication

- **Curve25519 operands:**

  255-bit length, can be divided into 15 x 17-bit chunks

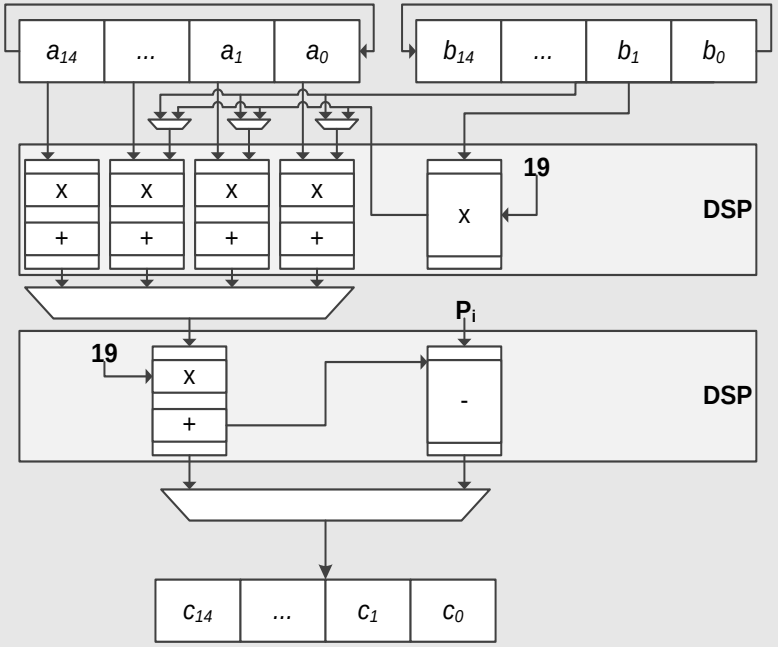
Finite Field Arithmetic



Modular Adder C = A + B *mod* P:

- always addition/subtraction and reduction
- output depends borrow/carry
- serial computation with 2 DSPs

Finite Field Arithmetic

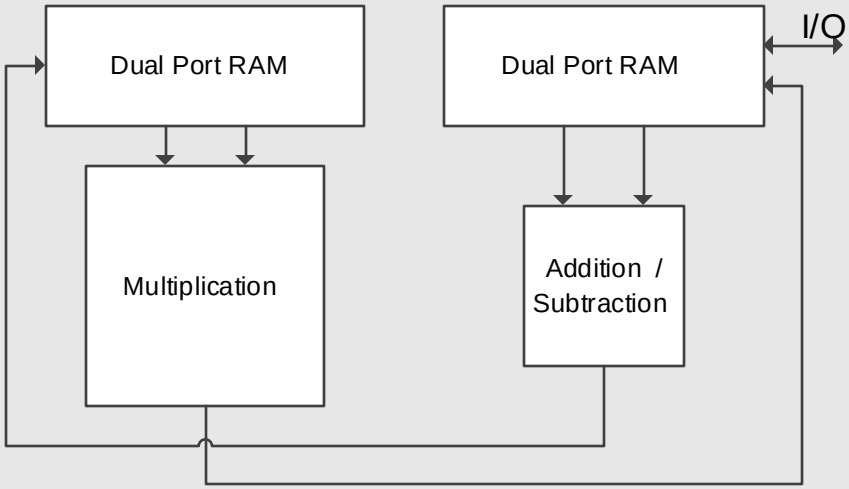


Modular Multiplier C = A x B *mod* P:

- pre-reduction of partial products
- Reduction by multiplication with 19
- accumulation of 17x17-bit products
- final accumulation and reduction (modified modular adder)
- DSPs for addition, multiplication and accumulation
- pipelining increases throughput

Efficient Elliptic-Curve Cryptography using Curve25519 on Reconfigurable Devices
Pascal Sasdrich, Tim Güneysu

Point Addition and Doubling

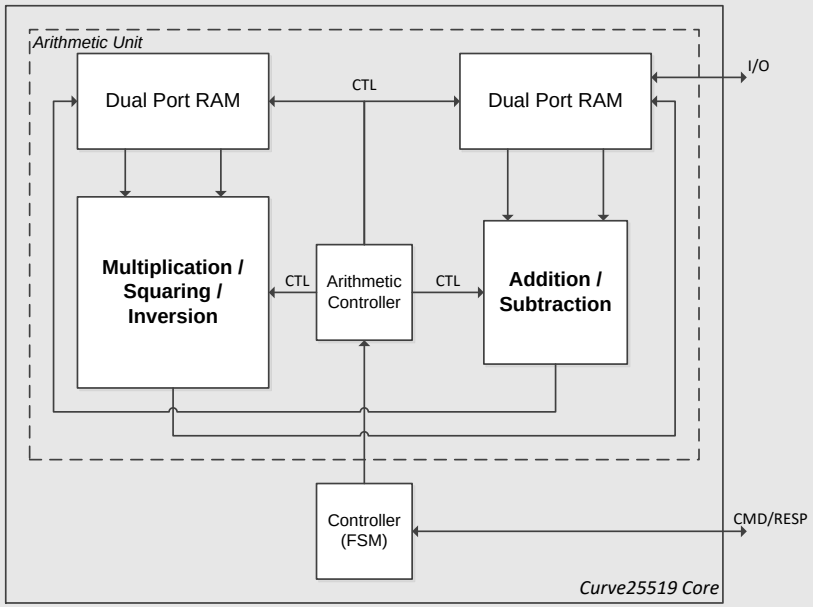


Double-And-Add:

- adder and multiplier can operate in parallel (2 BRAMs required)
- addition and multiplication operate in butterfly-mode
- point addition and doubling is computed according to Montgomery's ladder
- always compute point doubling and point addition in parallel

Efficient Elliptic-Curve Cryptography using Curve25519 on Reconfigurable Devices
Pascal Sasdrich, Tim Güneysu

Single Core Architecture



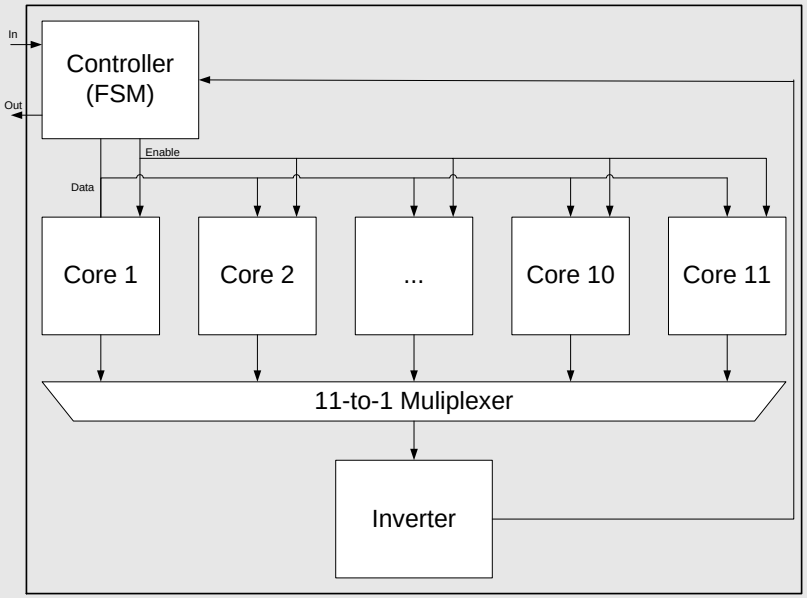
- **Parallel operation:**
  addition and multiplication can operate in parallel and independently
- **Inversion:**
  is realized by means of Little Fermat's Theorem using (many) multiplications
- **Controller FSM**
  manages the Diffie-Hellman computation by calling double-and-add function and final inversion

Efficient Elliptic-Curve Cryptography using Curve25519 on Reconfigurable Devices
Pascal Sasdrich, Tim Güneysu

Content

Efficient Elliptic-Curve Cryptography using Curve25519 on Reconfigurable Devices
Pascal Sasdrich, Tim Güneysu

Multi Core Improvements



1. **Parallelization:**

   multiple instances of single core in parallel

2. **Dedicated inverter stage:**

   inversion is covered by binary EEA based inverter

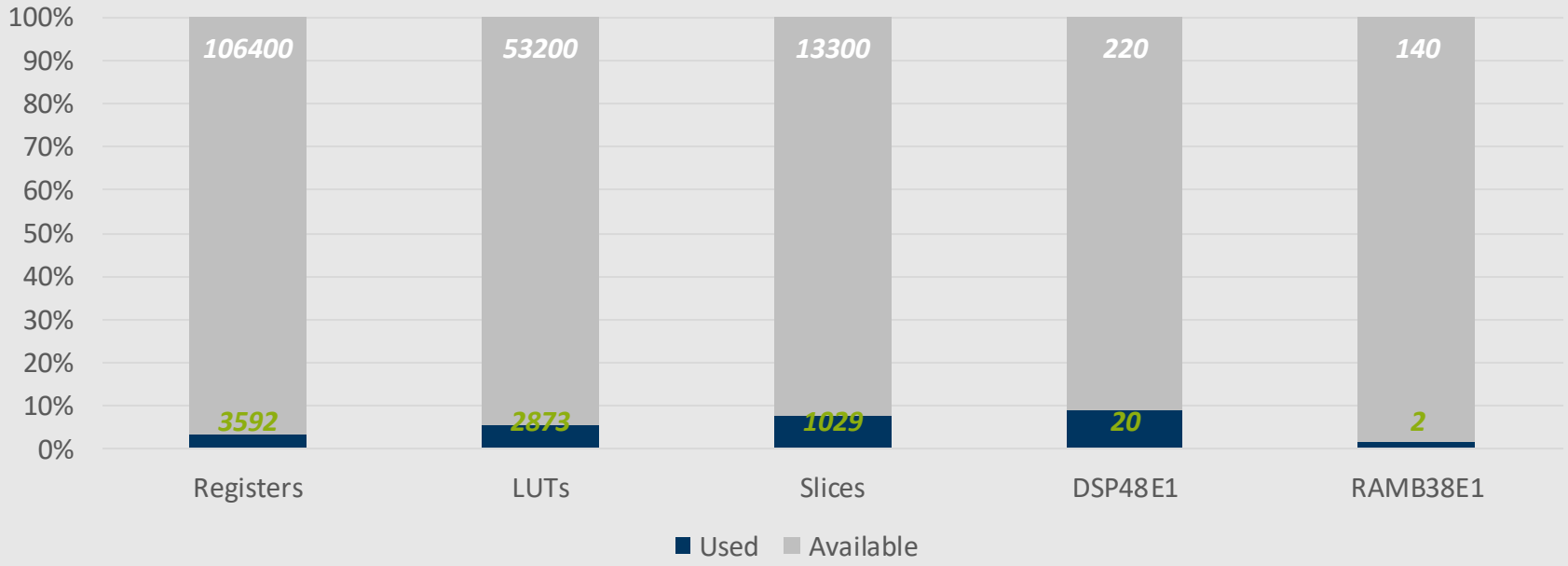
3. **Minimization:**

   cores now only support point doubling since inversion is outsourced

Content

Efficient Elliptic-Curve Cryptography using Curve25519 on Reconfigurable Devices
Pascal Sasdrich, Tim Güneysu

Resources | Single Core Architecture

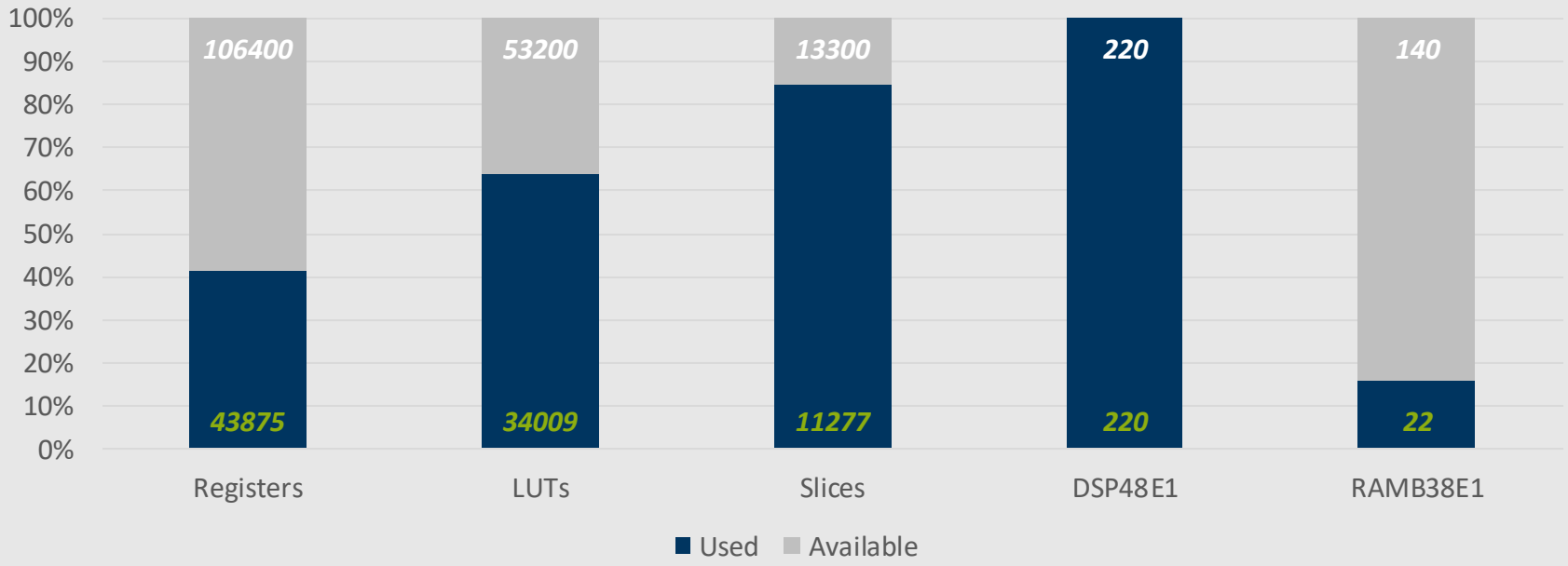


Single Core Architecture on Zynq XC7Z020:

- *200MHz clock frequency*
- *2518 point multiplications per second*

Efficient Elliptic-Curve Cryptography using Curve25519 on Reconfigurable Devices
Pascal Sasdrich, Tim Güneysu

Resources | Multi Core Architecture



Multi Core Architecture on Zynq XC7Z020:

- *100MHz clock frequency for inverter, 200 MHz for Curve25519 cores*
- *32303 point multiplications per second*

Results and Comparison

Device	Implementation	Logic	Clock	OP/s
XC7Z020	**255-bit (Curve25519)**	**1029 LS/20 DSP**	**200 MHz**	**2518**
XC7Z020	**255-bit (Curve25519)**	**11277 LS/220 DSP**	**100 MHz**	**32303**
XC4VFX12-12	256-bit (NIST)	1715 LS/32 DSP	490 MHz	2020
XC2VP125-7	256-bit (any)	15755 LS/256 MUL	39.5 MHz	260
Intel Core i3	255-bit (Curve25519)	64 bit	2.1 GHz	10810
AMD FX-4170	255-bit (Curve25519)	64 bit	4.2 GHz	14285

Content

Conclusion

Curve25519 was designed for fast software implementations but we showed how to implement the curve function efficiently in hardware.

Our design convinces with:

- moderate resource requirements (~1000 slices/20 DSP/2 BRAM per core)
- ~2500/~32000 OP/s on single/multi core architecture

Our design is cheap and easy deployable for many future security applications (e.g. for TOR-Routers and nTor).

RUHR-UNIVERSITÄT BOCHUM

RUB

Efficient Elliptic-Curve Cryptography using Curve25519 on Reconfigurable Devices

Pascal Sasdrich, Tim Güneysu

6/27/2014

20. Kryptotag, Berlin, Deutschland