FUJITSU THE POSSIBILITIES ARE INFINITE www.fujitsu.com

CHES2007
Vienna, Austria
2007/9/13

CAIRN2: Implementation of the Sieving Step in the Number Field Sieve Method

FUJITSU LTD.

Tetsuya Izu, Jun Kogure, *Takeshi Shimoyama

A part of this research is financially supported by a contract research with the National Institute of Information and Communications Technology (NICT), Japan

RSA and Integer Factoring Problem

FUJITSU

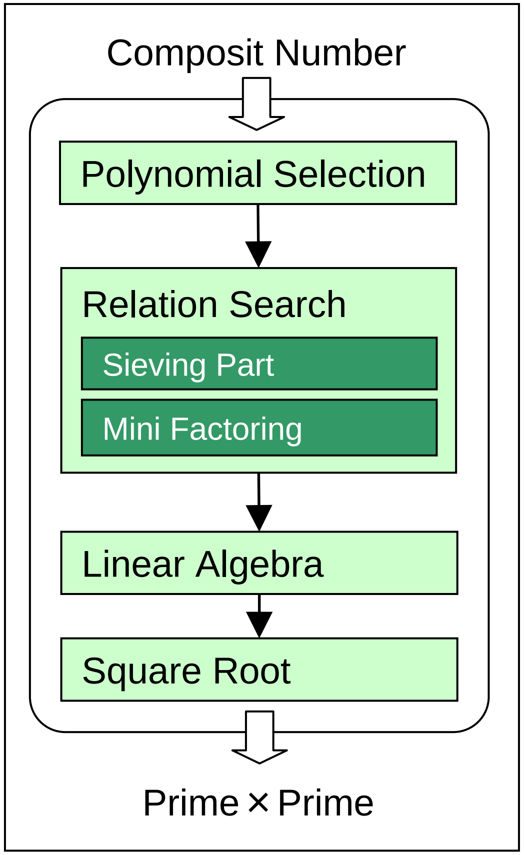
- Security Evaluation of RSA
  - RSA is one of the most important cipher for the current information security
    - Used world wide (Ex. SSL/TLS, SSH etc.)
    - International Standard of the public key cryptography
  - The security is depend on the Integer Factorization Problem
    - It's quite important to evaluate how large composit number can be factoring.

  It's believed that factoring the large integers are quite difficult, especially 1024-bit RSA keys.

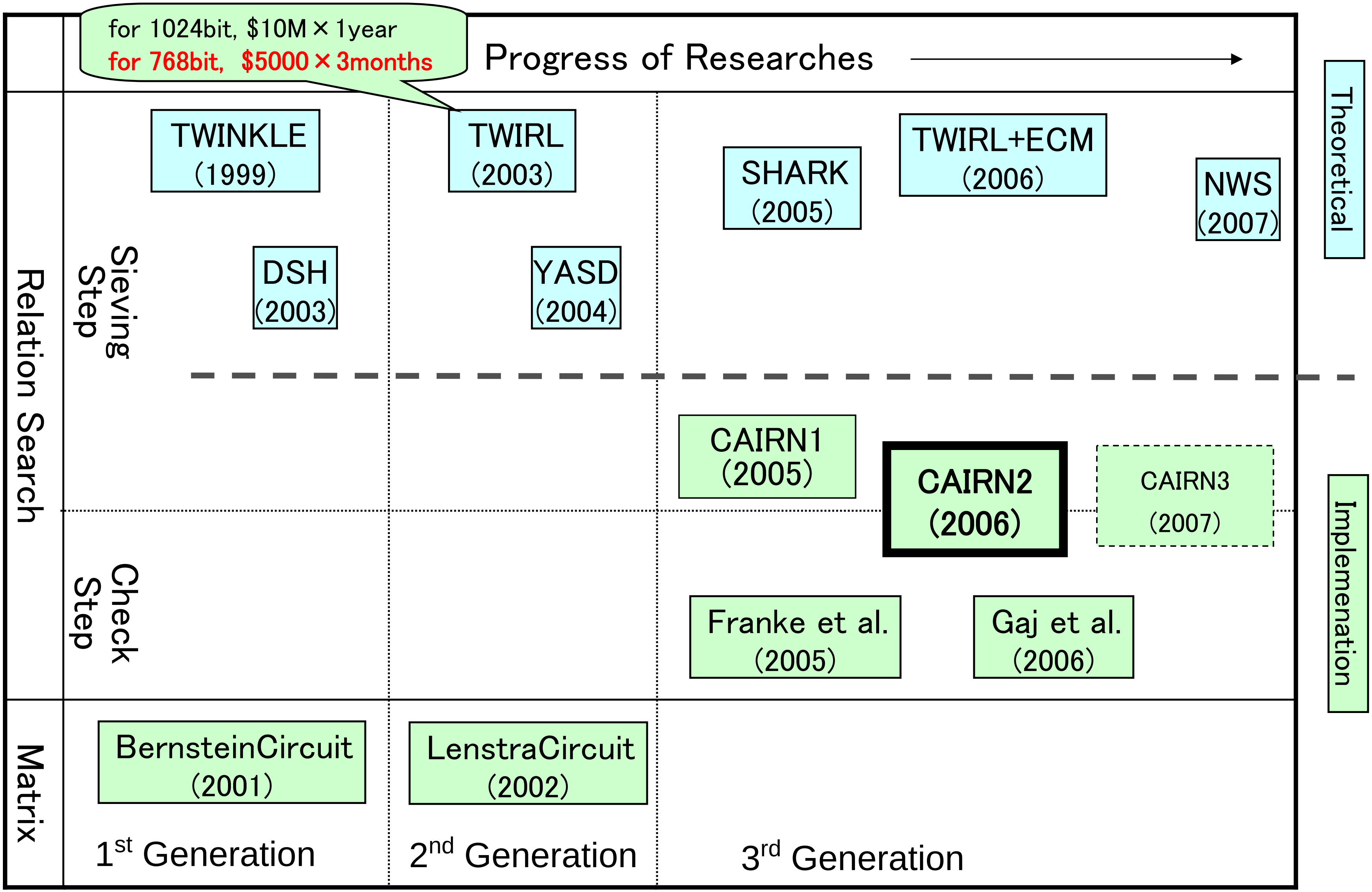
- Integer Factoring Problem (IFP)
  - GNFS is the most efficient method
    - It is consisted by 4 steps; Polynomial selection, Relation Finding, Linear Algebra, Square Root.
    - Most time consuming step is Relation Search and Linear Algebra.
  - Current World Records have done by Software on PCs
    - 663-bit in GNFS(2005/5)
    - 1017-bit in SNFS(2007/4)

  It's believed that 1024-bit RSA will be secure for a while.

GNFS procedure



Previous Works on the Integer Factoring HW



Motivation

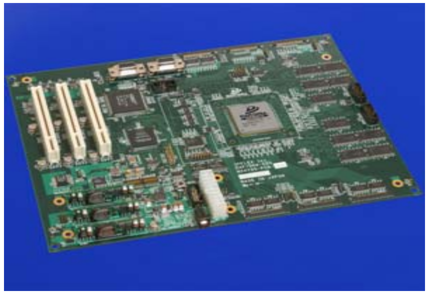
- There are many previous works in virtual world
- These Hardware devices of factorizations have not seen yet, in the real world.

⇩

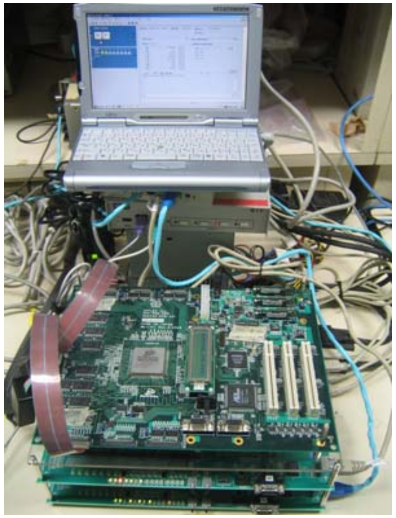
Let's Make it!

Progress of the Development of the Sieving HW

CAIRN:
Circuit Aided Integrated Relation Navigator



CAIRN1



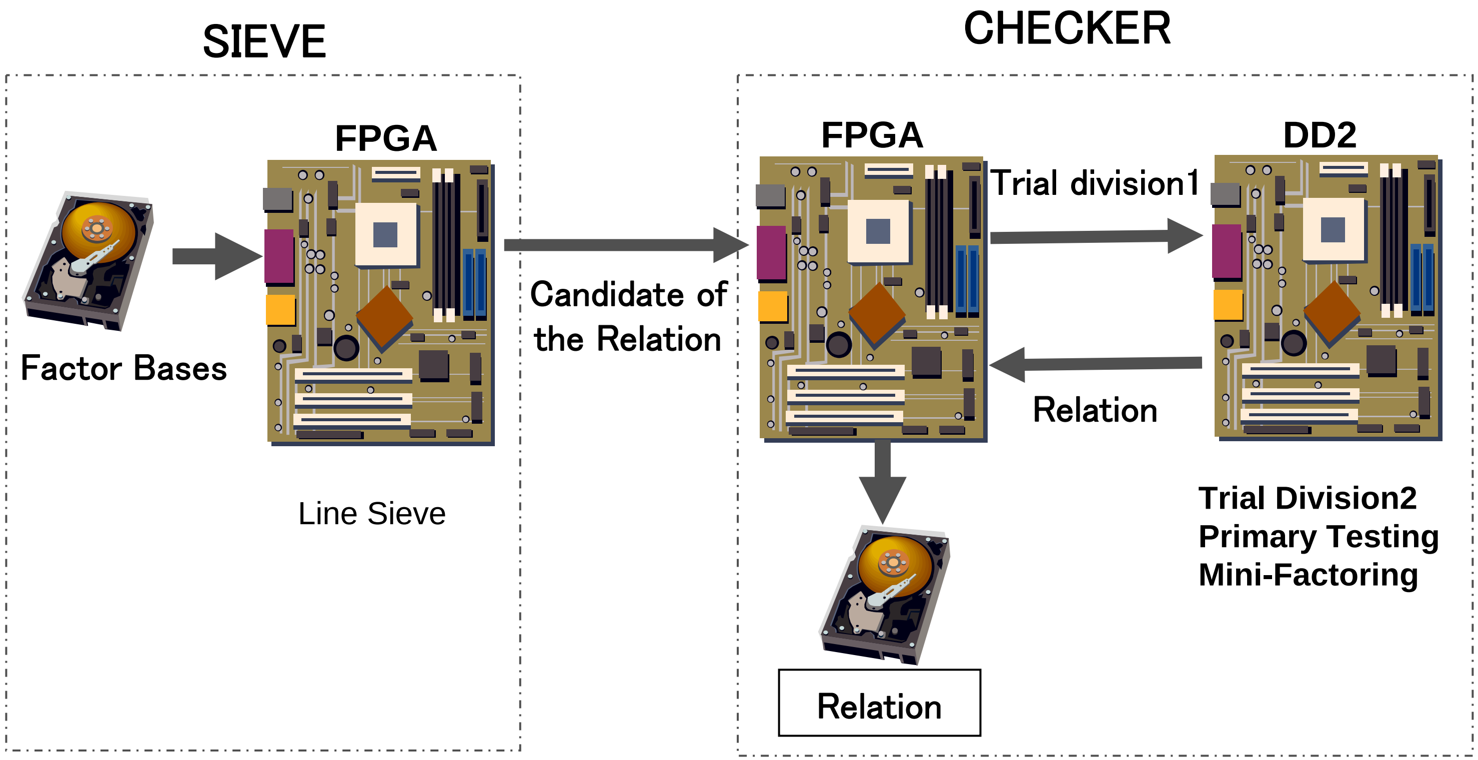
CAIRN2

- In 2005 (CAIRN1) [SHARCE2005]
  - Line Sieving implemented on DAPDNA2
  - Input : 100 digit number (RSA100)
- In 2006 (CAIRN2) [CHES2007]
  - Line Sieving and Relation checking
  - Implemented on FPGA×2 and DAPDNA2
- In 2007 (CAIRN3) [To be appeared(※)]
  - Lattice Sieveing and Relation Checking

(※) Extended abstract was reported at SHARCS2007.

CAIRN2 : Sieving Hardware

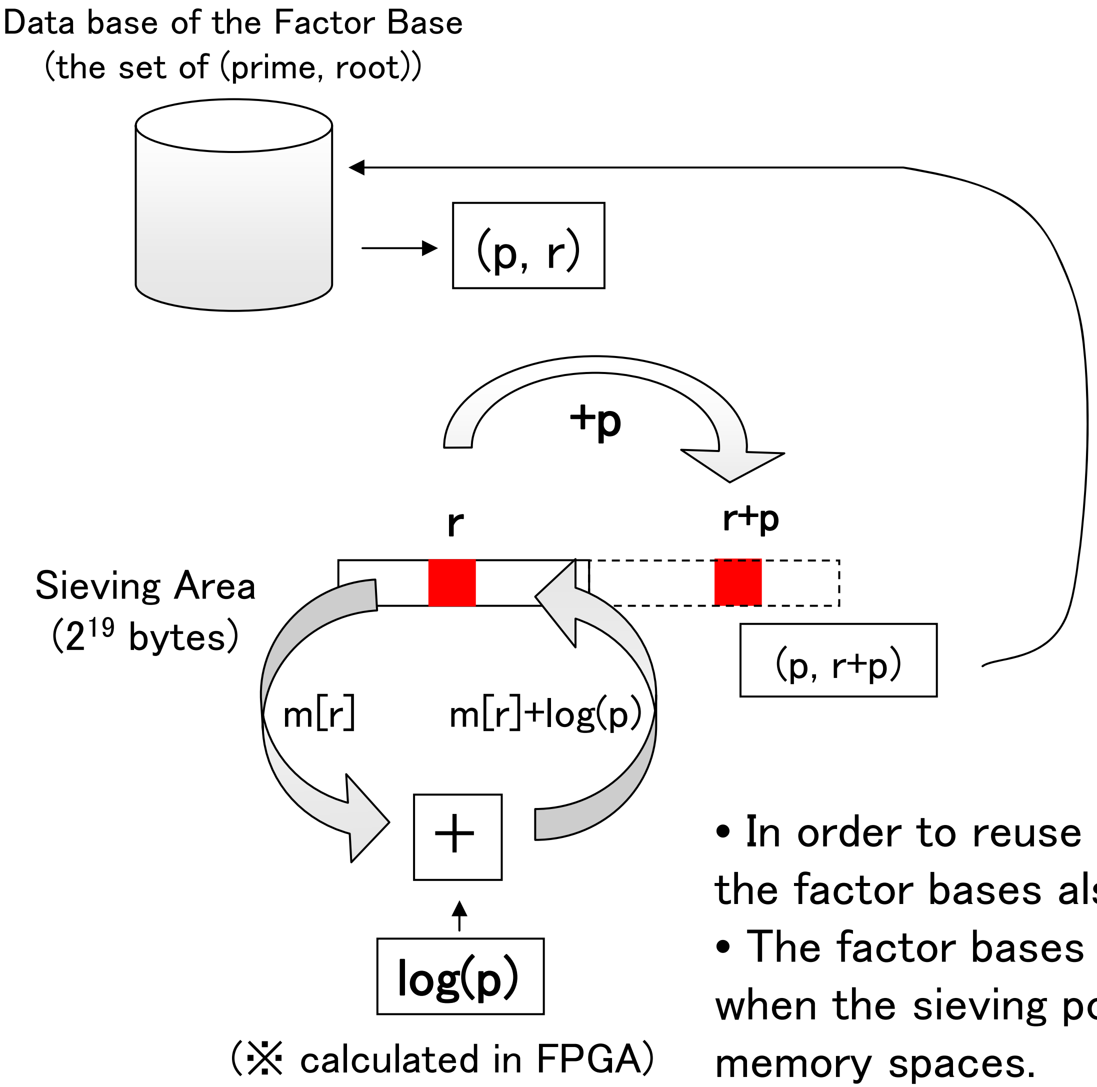
- Combination of the two kind of the devices
  1. Line Siever (SIEVE)
  2. Checker of the relation (CHECKER)
- Flow of the calculation



Summary of CAIRN2

1. Based on GNFS Method
   - Line Sieving
   - Maximum input is 768-bit composit number
2. Used FPGA Virtex-4 and Dynamic Reconfigurable Processor DAPDNA-2
   - FPGA
     - Logic Element 200448 Cell, Block RAM 336×18 Kbit(756KB),
   - Clock frequency
     - 133MHz (FPGA), 166MHz (DAPDNA2)
3. Used a lot of new techniques
   - Sieve
     - Pipelied Sieving for small primes
     - ➡ Partitioned Factor Bases for Large Primes
     - Updating Factor Bases
     - Buffer Estimation
     - Parallelized Buffers and Bucket Sorting
     - Computing Log Value in FPGA
   - Checker
     - ➡ Trial Division
     - Primarilty Test
     - Mini-Factoring

Sieving Algorithm



- In order to reuse the memory for the sieveing area, the factor bases also must be reused.
- The factor bases will be returned to the data base when the sieving point (r) of factor base go over the memory spaces.

Partitioned Factor Bases (1)

FUJITSU

Classify the set of the Factor Bases (p, r) as follows

Algebraic Sieve			
FB	$p/2^{19}$	#SB	size of SB
FB0	0	1	65536
FB1	1	2	32768
FB2	2 ～ 3	4	32768
FB3	4 ～ 7	8	32768
FB4	8 ～ 15	16	32768
FB5	16 ～ 31	32	32768
FB6	32 ～ 63	64	32768
FB7	64 ～ 127	128	32768
FB8	128 ～ 255	256	32768
FB9	256 ～ 511	512	32768
FB10	512 ～ 1023	1024	32768
FB11	1024 ～ 2047	2048	32768

Rational Sieve			
FB	$p/2^{19}$	#SB	size of SB
FB0	0	1	65536
FB1	1	2	32768
FB2	2 ～ 3	4	32768
FB3	4 ～ 7	8	32768
FB4	8 ～ 15	16	32768
FB5	16 ～ 31	32	32768
FB6	32 ～ 63	64	32768
FB7	64 ～ 127	128	32768
FB8	128 ～ 255	256	32768

Partitioned Factor Bases (2)

- $FB1 = \{ (p, r) \mid 2^{19} \leqq p < 2^{20} \}$

  Classify the Lattice base in FB1 by "r" in the following range

SB1_1 1st Sieve	SB1_2 2nd Sieve
$0 \leqq r < 2^{19}$	$2^{19} \leqq r < 2^{20}$

- $FB2 = \{ (p, r) \mid 2^{20} \leqq p < 2^{21} \}$

  Classify the Lattice base in FB2 by "r" in the following range

SB2_1 1st Sieve	SB2_2 2nd Sieve	SB2_3 3rd Sieve	SB2_4 4th Sieve
$0 \leqq r < 2^{19}$	$2^{19} \leqq r < 2*2^{20}$	$2*2^{19} \leqq r < 3*2^{19}$	$3*2^{19} \leqq r < 4*2^{19}$

- $FB3 = \{ (p, r) \mid 2^{21} \leqq p < 2^{22} \}$

  Classify the Lattice base in FB3 by "r" in the following range

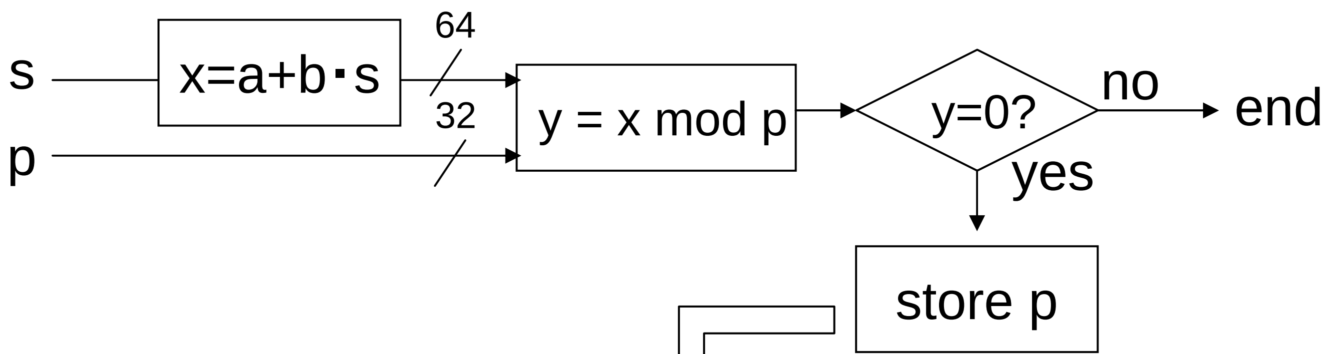
SB3_1 1st Sieve	SB3_2 2nd Sieve	SB3_3 3rd Sieve	SB3_4 4th Sieve	SB3_5 5th Sieve	SB3_6 6th Seive	SB3_7 7th Sieve	SB3_8 8th Sieve
$0 \leqq r < 2^{19}$	$2^{19} \leqq r < 2*2^{20}$	$2*2^{19} \leqq r < 3*2^{19}$	$3*2^{19} \leqq r < 4*2^{19}$	$4*2^{19} \leqq r < 5*2^{19}$	$5*2^{19} \leqq r < 6*2^{19}$	$6*2^{19} \leqq r < 7*2^{19}$	$7*2^{19} \leqq r < 8*2^{19}$

Trial Division

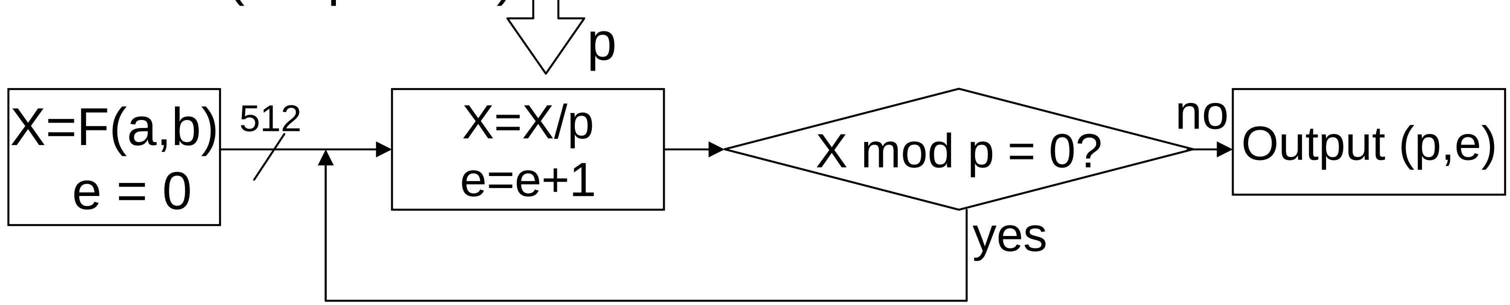
- Find largest "e" such that $p^e \mid F(a,b)$
  - Factor base (p,s) s.t. F(−s,1) = 0 mod p
  - F(a,b) = 0 mod p ⇔ a + b･s = 0 mod p

512bit **64bit**

- Trial Division Part 1 (Pipelining)



- Trial Division Part 2 (Sequential)



Results of the Implementation

- Maximum input : 768-bit composit number
- Size in FPGA (Virtex-4 XC4VLX200)

	SLICE (%)	RAM(%)	LUT(%)	Register(%)
SIEVE	24.0%	90.4%	15.9%	11.3%
CHECKER	78.0%	40.0%	45.0%	42.0%
Total	200,448	336	178,176	178,176

- Throughput
  - C128

Computation	Device	Throughput	Comment
Initial Setting	CPU	0.31msec	Sieving Area= 2^{19}
Sieving	FPGA	149.9msec	Sieving Area= 2^{19}
Extracting Relation	CPU	173.6msec	Sieving Area= 2^{19}
Sending Relation	EtherNet	19.9msec	

  - RSA768

Computation	Device	Throughput	Comment
Initial Setting	CPU	4.0msec	Sieving Area= 2^{19}
Sieving	FPGA	2381.3msec	Sieving Area= 2^{19}
Extracting Relation	CPU	2475.3msec	Sieving Area= 2^{19}
Sending Relation	EtherNet	131.2msec	

Factoring Test

FUJITSU

- Target of the composit number
  - c128 : A 423–bit (128digit) cofactor included in 7^{352}+1 which had not been factored yet. (One of the Cunningham number※)
- Execution period
  - About one month
- Factoring of c128
  - By processing the relations obtained from CAIRN2, we can find the factors of c128. (62 digit x 66 digit)

11002922872496853405938319182730880331313742514339168690475853 56
09065326627643139824106278480165493715571426969864417564889586 57

＝

4549363729281646485206701473657133979231541985978421807687 5841

×

2418563018313384375377878980960626923598195433036198640744103 82977

※ An integer formed as $b^c \pm 1$ (b=2,3,5,6,7,10,11,12, c : large)

Concluding Remarks & Future Work

Concluding Remarcs

- CAIRN2: Implementation of the Line sieving and Relation Check
- Factoring composit integer which has not been factored.
  - For test of running of CAIRN2
  - 128 digit = 62 digit x 66 digit in about 2 months

Future Works

- Improvement of the CAIRN2
  - Lattice Sieving [CAIRN3]
- Strictly Evaluation of the Security of RSA
  - Is 1024-bit RSA secure against the Special Purpose Hardware?

FUJITSU

THE POSSIBILITIES ARE INFINITE