US005101431A

United States Patent [19]

Even

[11] **Patent Number: 5,101,431**

[45] **Date of Patent: Mar. 31, 1992**

[54] **SYSTOLIC ARRAY FOR MODULAR MULTIPLICATION**

[75] Inventor: **Shimon Even,** Haifa, Israel

[73] Assignee: **Bell Communications Research, Inc.,** Livingston, N.J.

[21] Appl. No.: **628,570**

[22] Filed: **Dec. 14, 1990**

[51] **Int. Cl.**^{5} **H04L 9/30;** G06F 7/52; G06F 7/72

[52] **U.S. Cl.** **380/30;** 380/28; 364/746; 364/746.1

[58] **Field of Search** 364/200, 900, 224, 224.3, 364/260.81, 931.01, 931.02, 937.18, 739, 744, 746, 746.1

[56] **References Cited**

PUBLICATIONS

"A Survey of Hardware Implementations of RSA", E. F. Brickell, presented at CRYPTO '89, Santa Barbara, Calif., Aug. 1989.

"A Cryptographic Library for the Motorola DSP 56000", S. R. Dusse et al., EuroCrypt 90—Abstracts, May 21-24, 1990, Scantic on Arhus, Denmark, pp. 230-244.

"VICTOR, An Efficient RSA Hardware Implementation", H. Orup et al., EuroCrypt 90—Abstracts, May 21-24, 1990, Scanticon Arhus, Denmark, pp. 245-252.

"A One-Dimensional Real Time Iterative Multiplier", A. J. Atrubin, IEEE Trans. on Electronic Computers, vol. 14, pp. 394-399, 1965.

"Modular Multiplication Without Trial Division", P. L. Montgomery, Math. of Computation, vol. 44, pp. 519-521, 1985.

Primary Examiner—Bernarr E. Gregory
Attorney, Agent, or Firm—Leonard Charles Suchyta

[57] **ABSTRACT**

A systolic array (**10**) comprising a sequence of identical cells is utilized to perform modular reduction in linear time. Each cell receives a binary number at a first input and an n-bit modulus at a second input and performs binary addition to form a binary number output. The systolic array (**10**) for performing modular reduction may be combined with Atrubin's array (**60**) to perform modular multiplication.

10 Claims, 3 Drawing Sheets

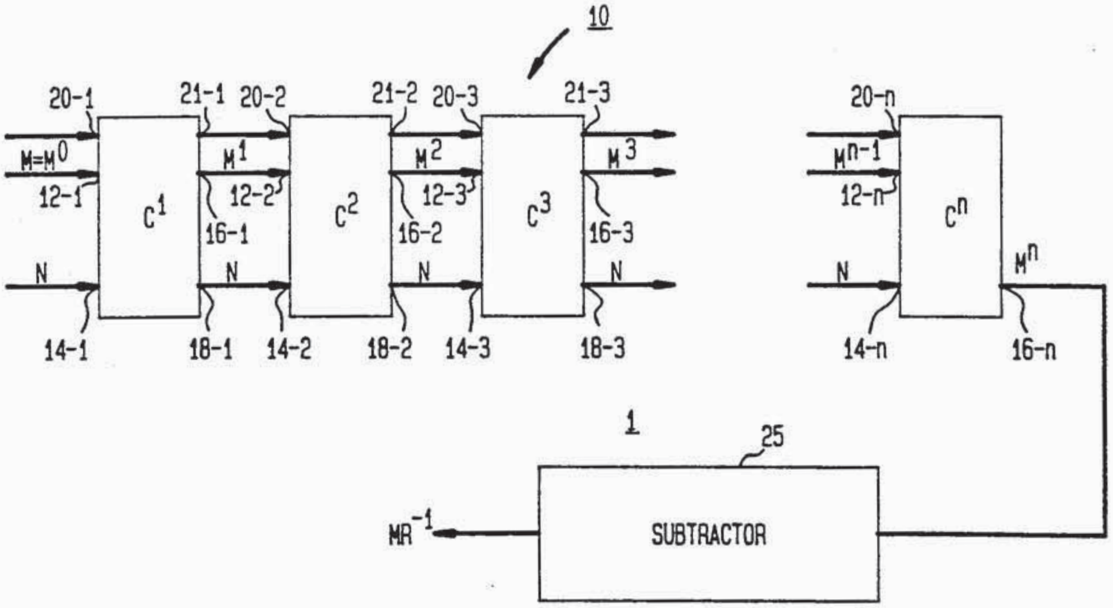


FIG. 1

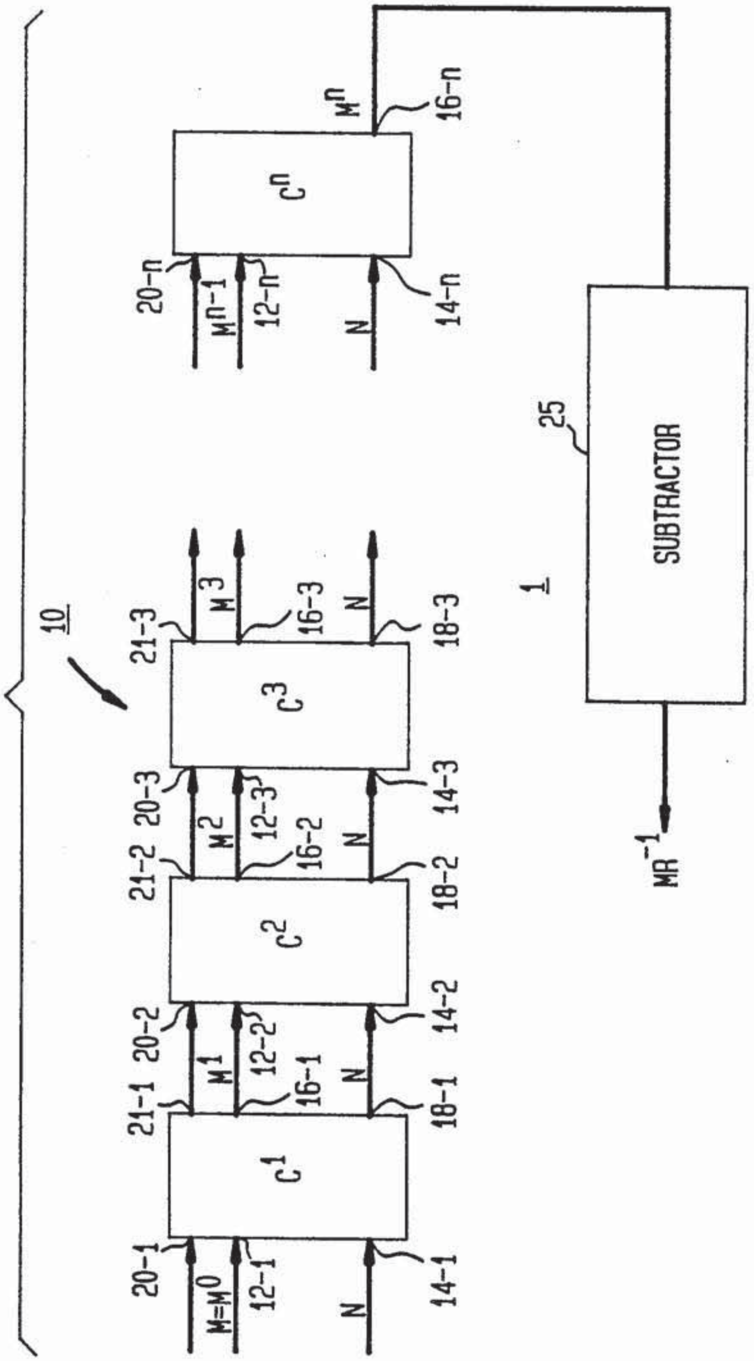
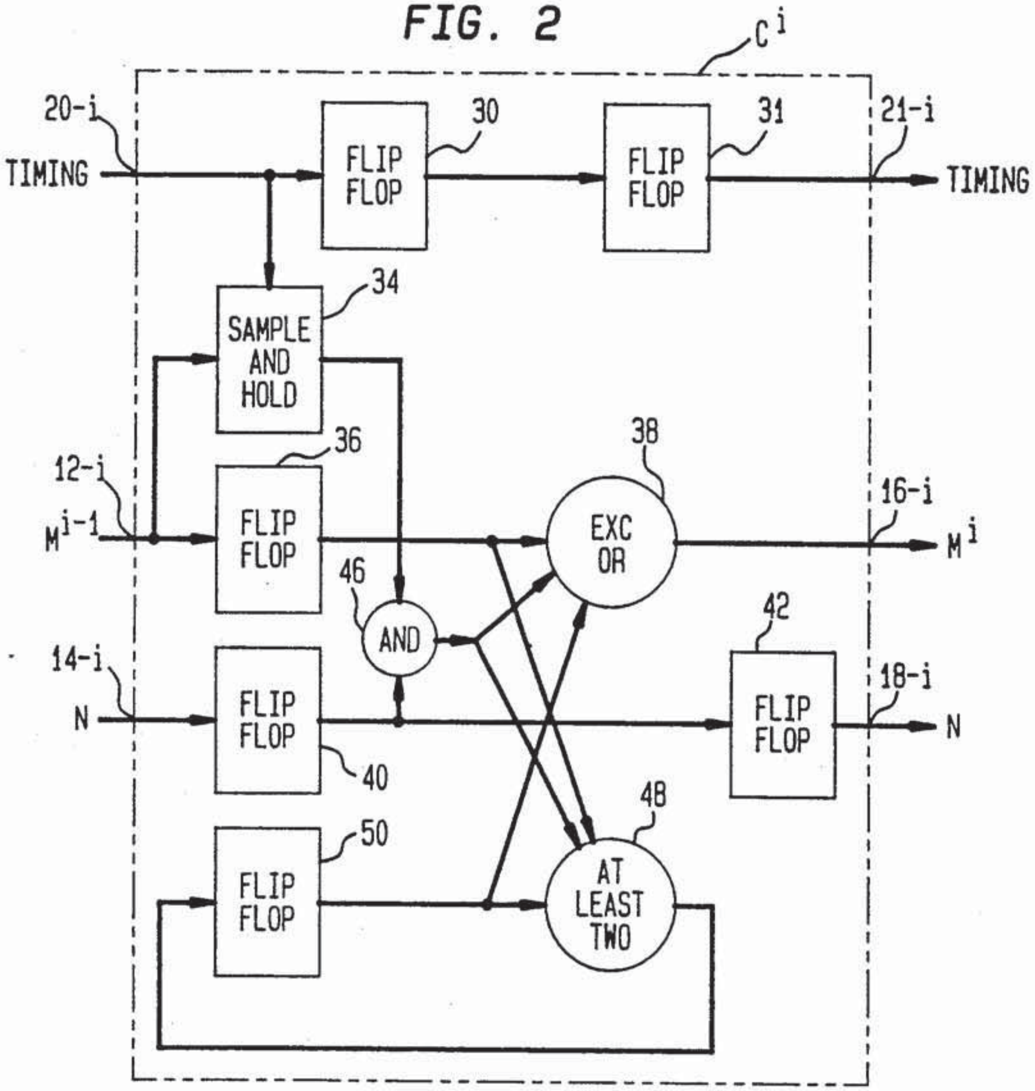
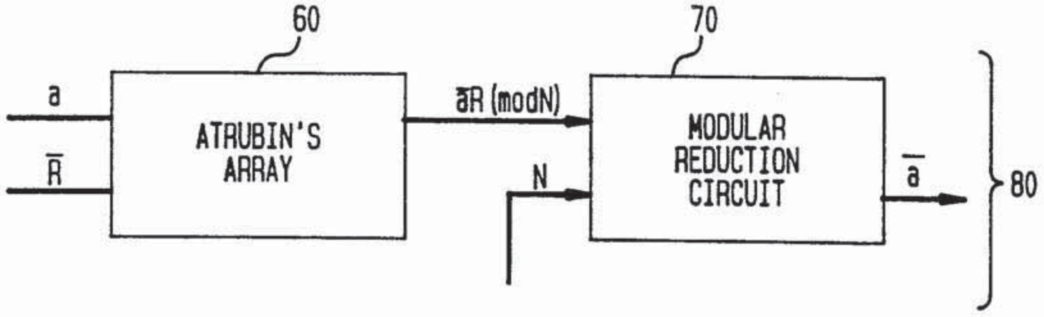

10
1
20-1
M=M0
12-1
N
14-1
C1
21-1
M1
16-1
N
18-1
20-2
12-2
14-2
C2
21-2
M2
16-2
N
18-2
20-3
12-3
14-3
C3
21-3
M3
16-3
N
18-3
20-n
Mn-1
12-n
N
14-n
Cn
Mn
16-n
25
SUBTRACTOR
MR-1

FIG. 2



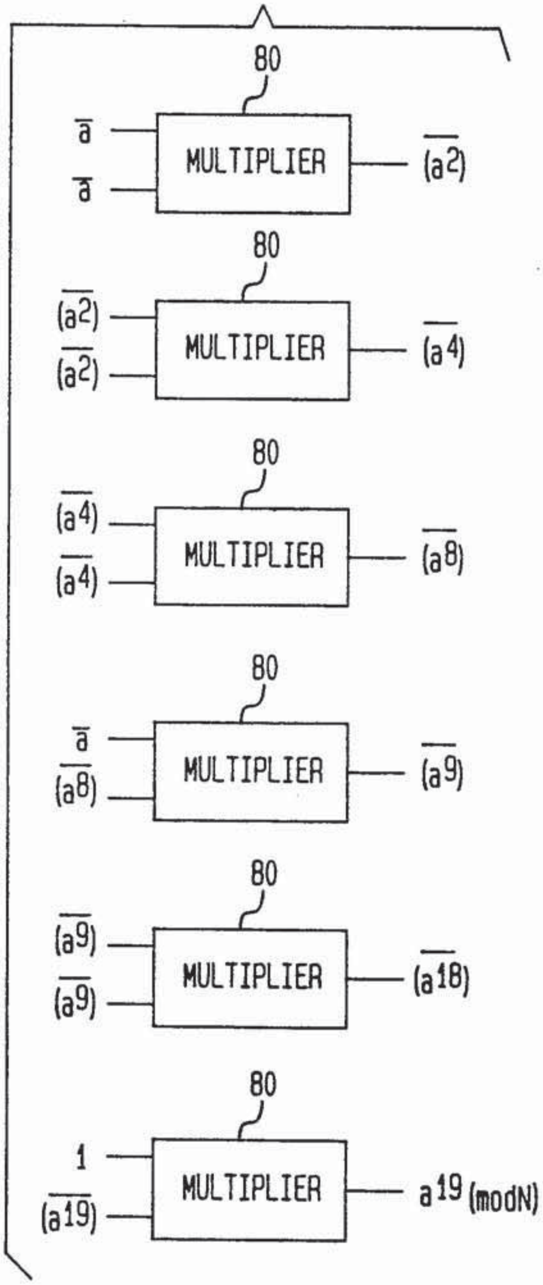
C^i
20-i
TIMING
30
FLIP FLOP
31
FLIP FLOP
21-i
TIMING
34
SAMPLE AND HOLD
36
12-i
M^i-1
FLIP FLOP
38
EXC OR
16-i
M^i
46
AND
42
14-i
N
FLIP FLOP
40
FLIP FLOP
18-i
N
48
50
FLIP FLOP
AT LEAST TWO

FIG. 3



60
70
a
R̄
ATRUBIN'S ARRAY
āR (modN)
MODULAR REDUCTION CIRCUIT
N
ā
80

FIG. 4


80
$\bar{a}$
$\bar{a}$
MULTIPLIER
$\overline{(a^2)}$
80
$\overline{(a^2)}$
$\overline{(a^2)}$
MULTIPLIER
$\overline{(a^4)}$
80
$\overline{(a^4)}$
$\overline{(a^4)}$
MULTIPLIER
$\overline{(a^8)}$
80
$\bar{a}$
$\overline{(a^8)}$
MULTIPLIER
$\overline{(a^9)}$
80
$\overline{(a^9)}$
$\overline{(a^9)}$
MULTIPLIER
$\overline{(a^{18})}$
80
1
$\overline{(a^{19})}$
MULTIPLIER
a^{19} (modN)

SYSTOLIC ARRAY FOR MODULAR MULTIPLICATION

FIELD OF THE INVENTION

The present invention relates to a systolic array for performing modular reduction. The inventive array is especially useful in cryptographic systems where repeated modular multiplication is utilized.

BACKGROUND OF THE INVENTION

A public key cryptography system is made up of a collection of users, each having his own encryption and decryption keys. In one such system, known as the RSA system, the encryption key comprises two integers N and e and the decryption key is single integer d. The integers N, d, and e are typically very large integers whose binary representations typically involve several hundred bits. Each user makes his encryption key available to the other users while keeping his decryption key secret.

In this system, N is an integer which is the product of two carefully selected large primes p and q and e is an integer such that its greatest common divisor with $(p-1)(q-1)$ is one [gcd $\{e,(p-1)(q-1)]=[$. Finally, d is calculated by solving the linear congruence

$$de=1\ (mod(p-1)(q-1))$$

If a user A wishes to communicate a numerically coded message M to another user B he calculates the ciphertext $K=M^e(\text{mod } N)(0\leqq K<N, 0\leqq M<N)$ where e and N form the encryption key for user B. To determine the plaintext M from the ciphertext K, B uses the decryption key d to calculate

$$M=K^d(mod\ N)\ (0\leqq M<N)$$

Thus, both the encryption and decryption operations involve repeated modular multiplications where the modulus N has a representation involving hundreds of bits.

As a result, great effort has been expended in the field of cryptography to find fast and inexpensive ways to do modular multiplication (see, e.g., E.F. Brickell "A Survey of Hardware Implementations of RSA" presented at CRYPTO'89, Santa Barbara, Calif., August 1989; S.R. Dusse et al, "A Cryptographic Library for the Motorola DSP 56000" EuroCrypt 90—Abstracts, May 21–24, 1990, Scanticon Arhus, Denmark, pp. 213–217; H. Orup et al, "VICTOR, An Efficient RSA Hardware Implementation" Eurocrypt 90—Abstracts, May 21–24 1990 Scanticon, Arhus, Denmark, pp. 219–227).

A systolic array of cells for performing ordinary (i.e. not modular) multiplication of large integers has been proposed by Atrubin (see, A.J. Atrubin, "A One-Dimensional Real Time Iterative Multiplier", IEEE Trans. on Electronic Computers, Vol. 14, 1965, pp. 394–399). Two positive integers to be multiplied are represented in binary. They are fed serially to the first cell of the array, least significant bit first. The product is supplied by the first cell, least significant bit first, without delay. The time required to obtain the product is linear with the length of the product. The structure of each cell in Atrubin's array is very simple and the array utilizes no long distance communications, i.e., each cell communicates only with its neighbors. Thus, a very high clock rate is possible.

It is an object of the present invention to provide a systolic array of cells which can be utilized in combination with Atrubin's array to perform repetitive modular multiplications of the type utilized in the above-described public key cryptographic systems.

As is shown below the inventive systolic array utilizes the modular reduction system proposed by Montgomery (see, P.L. Montgomery, "Modular Multiplication Without Trial Division", Math. of Computation, Vol. 44, 1985, pp. 519–521).

The modular reduction system of Montgomery may be understood as follows. Let the modulus N be an odd integer. Let n be the number of bits in the binary representation of N i.e. $2^{n-1}<N<2^n$. Let $R=2^n$. Let $\overline{x}$ be the image of the integer where $\overline{x}=xR(\text{mod } N)$. A binary representation of an image has at most n bits and it may exceed N, but it is nonnegative.

In the Montgomery modular multiplication system, modular operations are done with images. For example, suppose one wants to multiply two numbers x and y to obtain a product $z=xy(\text{mod } N)$. In the Montgomery system this is done by multiplying $\overline{x}$ and $\overline{y}$ to obtain $M=\overline{xy}=xyR^2(\text{mod } N)=zR^2\ (\text{Mod } N)=\overline{zR}(\text{Mod } N)$. Note that M has at most 2n bits and z is non-negative and has at most n bits.

In order to obtain $\overline{z}$ from M it is necessary to divide M by R and in order to obtain z from $\overline{z}$ it is necessary to divide again by R. This modular division by R is known as modular reduction.

Thus, it is a further object of the present invention to provide a systolic array of cells to perform the above-described modular reduction operation and to utilize the systolic array to perform repeated modular multiplications in a cryptographic system.

SUMMARY OF THE INVENTION

In accordance with the present invention, the computation MR^{-1} (mod N) is performed using the following algorithm.

```
begin
    $M^0 \leftarrow M$
    for $i=1$ to n do
    begin
        $M^i \leftarrow M^{i-1}+NM_0^{i-1}$
        Shift-right $M^i$, one bit
    end
    if $M^n 2^n$ then $M^n \leftarrow M^n-N$
end
```

where M^0-M is the number to be reduced, $M^n=M^0R^{-1}(\text{mod } N)$, and M_0^i is the least significant bit of the binary number M^i.

An array for carrying out this algorithm to perform a modular reduction comprises a sequence of cells C^i, $i=1, \ldots, n$. Each of the cells C^i except the first cell C^1 has a first input for serially receiving a binary number M^{i-1} from the immediately preceding cell C^{i-1} in the sequence. The first cell C^1 has a first input for serially receiving the binary number $M=M^0$ to be reduced from an external source. Each cell in the sequence also has a second input for serially receiving an n-bit modulus N. A serial adder in each cell serially performs the binary operation $M^i=M^{i-1}+NM_0^{i-1}$ and each cell includes means for affecting a one bit shift right operation on the binary number M^i. Each cell except the last cell in the sequence has an output for serially transmitting M^i to the first input of the next cell in the sequence. The last cell has an output for serially outputting a binary number M^n which is equal to M^0R^{-1} (mod N).

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.