Vol. 24 No. 6 情報処理学会論文誌 Nov. 1983

RSA 公開鍵暗号の高速計算法と暗号 LSI の構成†

宮口 庄司††

RSA 公開鍵暗号法の高速計算法と，この計算法に基づく暗号 LSI の構成を提案する．新計算法においては，暗号計算単位である n を法とする乗算/除算が同一クロックで同時に並行して実行される．除算の商は，計算量減少のため近似され，近似商で算出された剰余は一定のルールによって簡単に補正される．新計算法を適用した暗号 LSI は，その大部分が規則的な回路と ROM により構成される．暗号計算速度は，現在の LSI 技術を前提として，50 kb/s が実現できる．これにより暗号計算速度の低速性が解決され，RSA 暗号の実用化が期待できる．

1. まえがき

データ通信の安全保護が最近ますます重要なものとなっている．この対策として一般に暗号が利用される．RSA 公開鍵暗号法[1]は，ディジタル署名が可能であること，計量的安全度が高いことなどにより最も有望な公開鍵暗号である．さらに，マスタ鍵の存在条件等の鍵管理法の研究も進んでいる[4]．しかし，RSA 公開鍵暗号法の高速計算法に関しては，まだ十分な検討が行われていない．Rivest の RSA 公開鍵暗号 LSI[2]は，暗号計算速度が 1,200 b/s である．この暗号計算速度は，端末コンピュータ間の通信にはよいが，コンピュータ間の通信には十分でない．RSA 法の計算に大型コンピュータを用いても，暗号計算速度は 1〜2 kb/s 程度である[7]．文献 3) では，暗号計算法は論じられているが，暗号 LSI の構成は論じられていない．

本稿は，RSA 公開鍵暗号の高速計算法と暗号 LSI 構成を提案する．新計算法は，文献 3) の計算法を改良し，性能/コストを向上させている．

本稿の計算法においては，RSA 法の計算単位である一対の乗算と除算は，同一クロックで同時に並行して実行される．計算は，商の近似と剰余の補正によって行われる．新計算法を適用した暗号 LSI は，その大部分が規則的論理と ROM により構成される．暗号計算速度は現在の LSI 技術を用いて 50 kb/s が実現できる．

本稿の構成は，次のとおりである．2 章で本稿の検討にのみ必要な RSA 暗号アルゴリズムを説明する．3 章で RSA 法の計算単位である一対の乗算と除算を同一タイミングで同時に実行する高速計算法を導出する．4 章で高速計算法を整理して示す．5 章で高速計算法を適用した RSA 暗号 LSI の構成法を検討する．6 章で，各 RSA 公開鍵暗号 LSI を，回路規模，暗号計算速度等の観点から評価する．

2. RSA 暗号アルゴリズム

本節では，本稿の検討にのみ必要な RSA 暗号アルゴリズムを説明する．

暗号化対象メッセージを M，暗号文を C，暗号鍵を e, d, n とする ($e \neq d$)．暗号化と復号化のアルゴリズムは，

$$C = M^e \pmod{n} \quad \text{(暗号化)} \qquad (1)$$

$$M = C^d \pmod{n} \quad \text{(復号化)} \qquad (2)$$

で表される．暗号鍵の大きさは $n \simeq 10^{200}$, $e \cdot d \simeq 10^{200}$ である．RSA 法は，暗号鍵の値が大きいため，暗号計算量は大きくなる．

RSA 法の暗号化計算は，以下の手続きにより実行できる[1]．

ここで，整数 e は 2 進数で $e_k e_{k-1} \cdots e_0$ と表す．

Step 1: $C \leftarrow 1$

Step 2: Step 2a と Step 2b を，$i = k, k-1, \cdots, 0$ と繰り返す．

Step 2a: $M_1 \leftarrow C$; $M_2 \leftarrow C$;
$R \leftarrow M_1 \times M_2 \pmod{n}$;
$C \leftarrow R$;

Step 2b: if $e_i = 1$ then
$M_1 \leftarrow C$; $M_2 \leftarrow M$;
$R \leftarrow M_1 \times M_2 \pmod{n}$;
$C \leftarrow R$;

Step 3: Halt

† A Fast Computing Scheme for RSA Public-Key Cryptosystem and Its VLSI Organization by SHOJI MIYAGUCHI (Yokosuka Electrical Communication Laboratory, N. T. T.).

†† 電電公社横須賀電気通信研究所データ通信研究部

復号化計算は，暗号鍵 e の代りに，d を用いればよい．この手続きで計算時間を最も消費するのは，次の乗除算である．

$$R=M_1\times M_2 \pmod n \qquad (3)$$

3. 高速計算法の導出

本節では RSA 法の計算単位である一対の乗算と除算を，同一タイミングで同時に実行する高速計算法を導出する．

式(3)の乗除算は以下の手順で実行する．ここで，Q_j と R_j は，それぞれ，計算過程の商と剰余を示す．M_2 は μ ビットごとに l 等分する．$\lfloor x\rfloor$ は，x を越えない最大整数を示す．

Step 1: $R_{l+1}\leftarrow 0$

Step 2: 次のオペレーションを，$j=l, l-1, \cdots, 1$ と繰り返す．

$$Q_j\leftarrow \lfloor(2^\mu R_{j+1}+M_1\times M_{2,j})\div n\rfloor \qquad (4)$$

$$R_j\leftarrow (2^\mu R_{j+1}+M_1\times M_{2,j})-Q_j\times n \qquad (5)$$

Step 3: Halt

$(R=R_1=M_1\times M_2 \pmod n)$

ただし，$M_2=\sum_{j=1}^{l} M_{2,j}2^{(j-1)\mu}$

この手続きの証明は，付録に含まれる．

式(4)の分母，分子に 2^{-m} を掛ける．すなわち

$$Q_j=\lfloor\{(2^\mu R_{j+1})2^{-m}+(M_1\times M_{2,j})2^{-m}\}\div(n2^{-m})\rfloor$$

次に Q_j を次の q_j により近似する．

$$q_j=\lfloor X_j\div\lfloor n\,2^{-m}\rfloor\rfloor$$

ただし，$X_j=\lfloor(2^\mu R_{j+1})2^{-m}\rfloor+\lfloor(M_1\times M_{2,j})2^{-m}\rfloor+S$

$$q_j=Q_j+r_{1j} \qquad (6)$$

ここで，r_{1j} は整数であり近似誤差を示す．整定数 S は，近似誤差を抑制するため導入する．

次に $\lfloor n\,2^{-m}\rfloor$ の逆数に対応する変数 v を導入して q_j を q_j' で近似する．

$$q_j'=\lfloor X_j\cdot v\cdot 2^{-s}\rfloor+w_j$$

ただし，$v=\lfloor 2^s\div\lfloor n\,2^{-m}\rfloor\rfloor$

$$w_j=1: X_j\geq 0,\ \ 0: X_j<0$$

$$q_j'=q_j+r_{2j} \qquad (7)$$

ここで r_{2j} は整数であり近似誤差を示す．w_j は，$r_{2j}\geq 0$ を成立させるため導入するが，この詳細な理由は，付録“新計算法の証明”に示す．式(6)，(7)より，

$$q_j'=Q_j+r_j,\ \ r_j=r_{1j}+r_{2j}$$

Q_j の代りに q_j' を用いて，剰余 R_j' を求める．

$$R_j'\leftarrow (2^\mu R_{j+1}'+M_1\times M_{2,j})-q_j'\times n \qquad (5)'$$

誤差 r_j を最小化し，前記 Step 1〜Step 3 の繰返し計算を可能とする整定数 m, S, u を求める．この結果，$(M_1\times M_2)\div n$ の剰余 R は，次式により求められる．

$$R=R_1'+r_1\times n,\ r_1=0,1,2$$

なお，Q_j の近似は l 回行うが，R_j は $j=1$ のとき1回のみ補正すればよい．また，v の値は j に関係しない．いずれも，計算速度の向上に役立つ．

なお式(5)′の加算は，多入力・2出力のキャリア蓄積加算器 (CSA) により求める．このため，4章では R_j, R_{j+1} を以下により表現する．

$$R_j=R_{j,0}+R_{j,1},\ \ R_{j+1}=R_{j+1,0}+R_{j+1,1}$$

一方，乗算手法として，Booth の提案した乗算手法[5]が存在する．すなわち，

$$M_1\times M_2=\sum_{j=1}^{l} M_1\times(-\delta_{j\mu}\cdot 2^{\mu}+M_{2,j}+\delta_{(j-1)\mu})\times 2^{(j-1)\mu}-\delta_0\times M_1$$

ただし，

$$M_2=\sum_{i=0}^{l\mu-1}\delta_i\cdot 2^i$$

$$M_{2,j}=\sum_{i=0}^{\mu-1}\delta_{(j-1)\mu+i}\cdot 2^i$$

$$\delta_{l\mu}=0$$

Booth の手法については新計算法の乗算部分に導入する（オプション）．

4. 高速計算法

本節では3章で導出した高速計算法を整理して示す．q_j' は Q_j' と変えてあるが，この理由は，商の範囲を小さくし，暗号 LSI のハードウェア量を節約するためである．高速計算法の証明は付録に示す．

4.1 準　備

(1) 暗号計算条件：暗号計算条件を定めるため，L, μ, ω を定める．暗号鍵 n の有効長 L は，RSA 公開鍵暗号法の計量的安全度を考慮して定める．μ は，乗算と除算を一括して計算する単位である（$\mu=1,2,\cdots$）．Booth の手法は $\omega=1$ で選択し，$\omega=0$ で選択しない．ただし $\omega=1$ のとき μ は偶数に定める．

(2) 変数の範囲：次の条件を付与する．

$$2^{L-1}\leq n<2^L,\ 0\leq M_1,\ M_2<n \qquad (8)$$

(3) M_2 の表現：M_2 は2進数で表現し，各ビットを δ_i とする．さらに μ ビットごとに l 分割する．すなわち，

$$M_{2,j}=\sum_{i=0}^{\mu-1}\delta_{(j-1)\mu+i}\cdot 2^i \tag{9}$$

$$l=\lceil L\div\mu\rceil \tag{10}$$

次に，新計算法の表現を簡略化するため，変数 $Y_{j,i}$ と整定数 φ を定義する．

$$Y_{j,i}=\begin{cases}\delta_{(j-1)\mu+i}\cdot 2^i : \omega=0\\ -\delta_{(j-1)\mu+2i+2}\cdot 2^{2i+2}+\delta_{(j-1)\mu+2i+1}\cdot 2^{2i+1}\\ \quad +2\delta_{(j-1)\mu+2i}\cdot 2^{2i}-\xi\cdot\delta_{(j-1)\mu+2i}\cdot 2^{2i} : \omega=1\end{cases} \tag{11}$$

$$\xi=\begin{cases}0 : j\neq 1 \text{ or } i\neq 0\\ 1 : j=1 \text{ and } i=0\end{cases} \tag{12}$$

$$\varphi=\mu : \omega=0,\ \mu/2 : \omega=1 \tag{13}$$

（4） m, S, u の計算：L, μ, ω および φ の値をもとに，整定数 m, S, u の値を次式により定める．

$$\left.\begin{aligned}&m=L-\mu-4\\ &2^{\mu+1}+\varphi+2+\omega\cdot 2^{\mu}\leq S\\ &S\leq 2^{\mu+3}-2^{\mu+1}+1-\omega\\ &u=2\mu+5+\omega\end{aligned}\right\} \tag{14}$$

4.2 乗除算の実行

（1） 変数の入力：n を入力し，v を求める．ただし，

$$v=\lfloor 2^u\div\lfloor n\,2^{-m}\rfloor\rfloor \tag{15}$$

次に，M_1, M_2 を入力する．

（2） 繰返し計算（l 回）

Step 0：$j\leftarrow l,\ R_{l+1,0}'\leftarrow 0$

$$R_{l+1,1}'\leftarrow 0 \tag{16}$$

Step 1：$$X_j\leftarrow\sum_{i=0}^{1}\lfloor(2^{\mu}\cdot R_{j+1,i}')2^{-m}\rfloor + \sum_{i=0}^{\varphi-1}\lfloor(M_1\cdot Y_{j,i})2^{-m}\rfloor+S \tag{17}$$

Step 2：$$q_j'\leftarrow\lfloor X_j\cdot v\cdot 2^{-u}\rfloor+w_j \tag{18}$$

$$Q_j'=\begin{cases}2^{\mu+1}-1 : I_f\ q_j'\geq 2^{\mu+1}\\ q_j' \quad\ : I_f\ q_j'<2^{\mu+1}\end{cases} \tag{19}$$

ただし，

$$w_j=1 : X_j\geq 0,\ 0 : X_j<0 \tag{20}$$

Step 3：$$\sum_{i=0}^{1}R_{j,i}'\leftarrow\sum_{i=0}^{1}2^{\mu}R_{j+1,i}'+\sum_{i=0}^{\varphi-1}M_1\times Y_{j,i} - Q_j'\times n \tag{21}$$

Step 4：if $j=1$ then go to Step 5
else $j\leftarrow j-1$ and go back to Step 1 (22)

Step 5：Halt

（3） 補正計算（たかだか2回）

Step 6：$R\leftarrow R_{1,0}'++R_{1,1}'$ (23)

if $R\geq 0$ then go to step 8 (24)

Step 7：$R_{1,0}'+R_{1,1}'\leftarrow R_{1,0}'+R_{1,1}'+n$ and go back to Step 6 (25)

Step 8：Halt ($R=M_1\times M_2 \pmod n$)

Q_j' は符号ビットを含め，$\lambda+2+\omega$ ビットで表現できる．また $-2n\leq R_{j,0}'+R_{j,1}'<n$ が成立する．($R_{j,0}'+R_{j,1}'$) は符号ビットを含め $L+2$ ビットで表現できる．

v は $2^{\mu+1+\omega}<v\leq 2^{\mu+2+\omega}$ である．さらに n の条件を $2^{L-m-1}<\lfloor n\,2^{-m}\rfloor<2^{L-m}$ に限定すると，$2^{\mu+1+\omega}<v<2^{\mu+2+\omega}$ となる．

q_j' の代りに，Q_j' を用いる計算法は，文献3）の改良である．なお $\omega=1$ の場合は，$M_2'=M_2\times 2,\ n'=n\times 2$ として $(M_1\times M_2')\div n'$ の乗除算を行い，M_2' の最下位ビット $\delta_0'=0$ を補償する等，従来知られている工夫が必要なことはいうまでもない．

5. 新しい暗号 LSI の構成法

本章では高速計算法を適用した RSA 暗号 LSI の構成を検討する．

5.1 暗号計算ハードウェア

暗号計算ハードウェアは，2章で述べた暗号計算手続きを基本として，**図1**のとおり構成する．変数 e, n, M_1, M_2, C はそれぞれ専用のレジスタに格納する．変数 M_1 は，C レジスタの内容を参照して得られる．変数 M_2 は，M_1 か M のいずれかをセレクタによって選択して得られる．制御部 CTL-1 は，1ビット左シフトレジスタ e-$R_{e,g}$ の最上位ビット e_l から，セレクタの切替信号を生成する．乗除算 $R=M_1\times M_2$

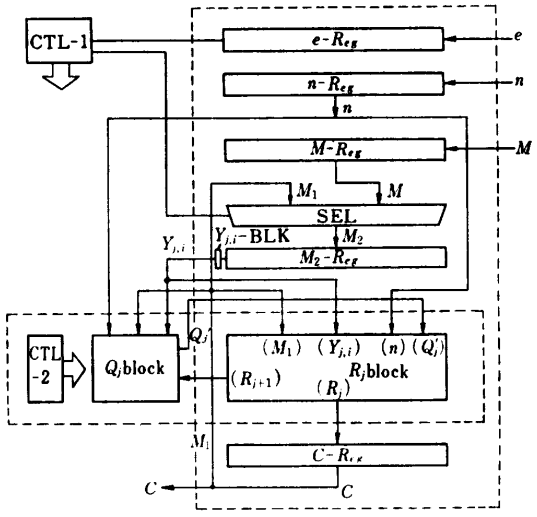

図 1 RSA 暗号計算ハードウェア
Fig. 1 RSA encryption hardware.

(mod n) は乗除算器によって求める．なお M_2 は式(11)の規則により，$Y_{j,i}$ に変えられる．この変換は $Y_{j,i}$-BLK が行う．$\omega=0$ のとき，$Y_{j,i}$-BLK は不要である．

乗除算器は，Q_j' を求める Q_j ブロック，R_j' の計算および剰余の補正計算を行う R_j ブロック，および制御ブロック (CTL-2) に分けて記述してある．

5.2 暗号 LSI の構成

(1) 回路の規則性：前述の暗号計算ハードウェアの規則性を各ブロックごとに検討する．

(a) レジスタとセレクタ：いずれも1ビットスライス可能であり回路の規則性はよい．

(b) R_j ブロック：R_j ブロックは式(21)，(23)，(25)の加算を受け持つ加算器である．加算を行う回路として R_j ブロックを**図2**により構成する．記号 $\times 2^{\mu}$ は μ ビット左シフト，すなわち 2^{μ} 倍する結線を示す．$M_1 \cdot Y_{j,i}$ は M_1 と $Y_{j,i}$ の論理積回路を示す．本回路は多数の AND 素子を規則的に並べて構成できる．$-Q_j \times n$ も同様である．COR は $-Q_j' \times n$ の2の補数加算のための補償回路であって，$Q_j' \geq 0$ のとき0を，$Q_j' < 0$ のとき $|Q_j'|$ を生成する回路である．回路規模はきわめて小さい．CSA は，1ビット全加算器を規則的に並べて構成する Wallace Tree である[5]．$R_{j,0}$ と $R_{j,1}$ は，それぞれ変数 $R_{j,0}'$，$R_{j,1}'$ を格納するレジスタであり，SEL は，2入力セレクタである．CPA は，2入力加算器である．CPA はキャリア伝播部を含めて，規則的に設計できるが，これについては，6章で述べる．

(c) Q_j ブロック：Q_j ブロックは，式(15)，および式(17)～(20)の計算を受け持つ．式(15)の計算は $\lfloor n 2^{-m} \rfloor$ をアドレスとして入力し，v の値を出力する ROM である．式(17)～(20)の計算回路は，規則的に設計できる加算器を含むが，全体として不規則回路である．

(d) 制御ブロック：制御ブロックの CTL-1，CTL-2 は，ともに不規則回路である．

(e) $Y_{j,i}$-B：不規則回路であるが，回路規模はきわめて小さい．

暗号 LSI のうち，ROM を除いた論理回路の規則性部分の比率は80%を越える ($L=512$, $\mu=8$ のとき)，Q_j ブロック，COR ブロック，$Y_{j,i}$-B ブロックを除いた部分は規則性がよい．規則性のよい回路は，そのハードウェア量に比し，設計コストが小さい．なお R_j ブロック内の CPA のキャリア伝播部を除けば，規則性部は μ ビットスライスできることに注意しよう．

(2) 暗号 LSI の分割構成：暗号計算ハードウェアを分割する方法について検討する．簡単のため，$L=512$, $\omega=0$ で，4分割の場合を説明する (**図3**)．Q_j チップと4個の CP-i チップ ($i=1\sim4$) は，暗号 LSI のチップである．ここで CP-i チップは，同一品種である．式(21)に現れる信号 $Y_{j,i}$ および Q_j' は各 CP-i チップに伝えられる．変数 e の最上位ビットは，CP-1 内の CTL-1 ブロックにより，SEL 切替制御信号となり，各 CP-i チップに伝えられる．CP-1～CP-4 内の各レジスタは，512ビットのデータを MSD 側から128ビットずつ格納する．これらレジスタは，μ ビット並列左シフトレジスタである．このため CP-i チップ間にデータシフト用の結線が現れる．セレクタの分割も，レジスタの分割と同様に行える．

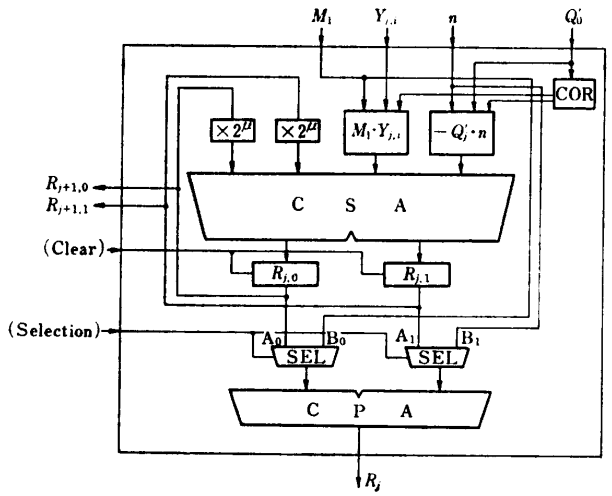

図2 剰余ブロック
Fig. 2 Remainder block.

なお，CP-i 内の C-r レジスタおよび R_j-r ブロックは，$i=1$ のとき130ビットデータを，$i\geq2$ のとき128ビットデータを扱う．この結果，全体として $L+2=514$ ビットのデータを扱う．また COR は CP-4 内でのみ動作させる．これらの詳細は，設計内容の説明になるので省略する．

Q_j ブロックの変数 v を生成する ROM を，LSI チップとして独立させ，Q_j ブロックから ROM を除いた回路を，各 CP-i チップに含める方法も考えられる．この利点は，論理 VLSI チップの品種が，1種類にできることである．

一方，CP-i チップは，N 個 ($N=1,2,\cdots$)，図3

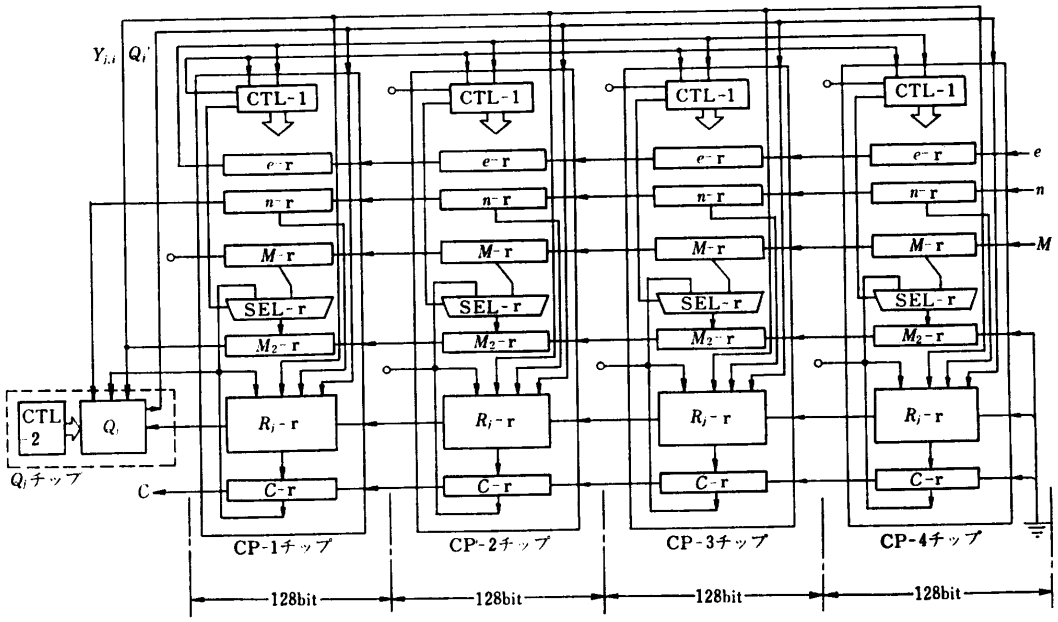

図 3 4分割した暗号計算ハードウェア
Fig. 3 Encryption hardware divided into four parts.

のように接続した場合も，暗号計算が行えることに注意しよう．この理由は，式(14)において，μ と ω が一定なら，$L-m$, S, u の値が変化しないことから理解される．

6. 評　価

本章では，各 RSA 公開鍵暗号 LSI を，回路規模，暗号計算速度等の観点から評価する．

Rivest の暗号 LSI と本稿による暗号 LSI との比較を**表1**に示す．暗号鍵 n は 512 ビット ($L=512$)，e, d は 256 ビットである．

本稿による暗号 LSI は，32 ビットプロセッサ[8]の製造実績のある CMOS を用いて試算している．CMOS のゲート長は 2 μm である．この値は，Mead と Conway のスケールファクタ λ[10]で表現すると 2λ =ゲート長 ($\lambda=1$) と近似できる．試算上のポイントは次のとおりである．①素子の Fan out は，最大3/素子（平均2）とする．これを越えるものは，Tr 数の多い高速素子を用いるか，または，Fan out を多数得るための分岐回路を用いる．②全加算器(FA)は，FA 内のすべての Tr を隣接して VLSI チップ内に配置する．これによって，FA 内の隣接した Tr の配線遅延時間を節約する．

表1は，$\mu=8$, $\omega=0$ で図3に示す4分割構成の試算値を示している．ただし，Q_j ブロック内の ROM

表 1 暗号 LSI の比較
Table 1 Comparison to encryption LSI chip.

		本論文 ($\mu=8$, $\omega=0$)	Rivest の暗号 LSI
1	$R=M_1\times M_2 \pmod n$ の計算	乗算 $M_1\times M_2$ と除算 $(M_1\times M_2) \pmod n$ を同時に実行.	乗算 $W\leftarrow M_1\times M_2$ 実行後に 除算 $R\leftarrow W \pmod n$ を実行.
2	乗除算	$M_1\times\delta_1$, $-n\times\delta_2$, これらの同時加算, μ ビットシフトの繰返し. ただし $0\le\delta_1$, $\lvert\delta_2\rvert<2^{\mu+\omega}$	(1) 乗算: $M_1\times\delta$, 加算, 1ビットシフトの繰返し ($\delta=0,1$) (2) 除算 $n\times\delta$, 減算, 1ビットシフトの繰返し ($\delta=0,1$)
3	暗号計算速度	50 kb/s	1.2 kb/s
4	ハードウェア規模	145 kTr/chip (注)×4 chip.+$2^{11}\times9$ bit (ROM)	40 kTr/chip
5	LSI 技術	c MOS ($\lambda=1$ μm)	n MOS ($\lambda=2$ μm)

注) Q_j ブロックの論理部 24 kTr を含む (ROM を除く).
λ は Mead と Conway のスケールファクタ.

を除いた論理回路の部分は，各 CP-i チップに含めている．$\omega=1$ の場合は，Tr 総数が8％減少する．暗号計算速度はほとんど変わらない．この結果，Wallace の乗算手法は，新計算法でも有効であることがわかる．

R_j ブロック内の加算器 CPA の構成法を述べる．文献3)では CPA 回路の規則性を重視し，全加算器を並べて，キャリアを上位桁側に順次伝播する方式に

より，加算時間を試算している．本稿では，2進加算器の規則的設計手法[6]を採用し，試算する．ただしこの手法で定義される white processor，すなわち，データ伝播用素子を，CMOS のドライバ素子に置き換えた．これによって，CPA 全体の Tr 数の増加はあったが，CPA の回路の規則性を保って，加算時間を向上できた．この結果，暗号計算速度を 20% 以上向上させる見通しを得たので，μ の値を 12 から 8 に変え，暗号計算速度＝50 kb/s を保ったまま，ハードウェア量を減少させた．

式(19)の Q_j'' の計算法の改良は，CP-i チップの Tr 数を，約3％減少した．

暗号 LSI のクリティカルパスの内訳は，ROM を除いた Q_j ブロックが 50%，R_j ブロックが 35%，チップ間遅延が 15% である．

本稿の暗号 LSI の暗号計算速度は従来の暗号 LSI より約 40 倍速い．この理由は，概略すると次の相乗効果によると考えられる．

① LSI 技術の差，すなわち λ の差で2倍（表1，項番5）

② 乗算と除算の同時実行で2倍（表1，項番1）

③ 乗算と除算の一括実行で 10 倍（表1，項番2）

③の 10 倍は，μ=8 ビットの一括計算と μ=8 ビット並列シフト利用による実行時間の短縮により達成される．なお，③の場合は，速度向上にほぼ比例して Tr 数が増加している．しかし，回路の規則性がよいため，Tr 数の増加に比例して暗号 LSI の設計コストが増えない．

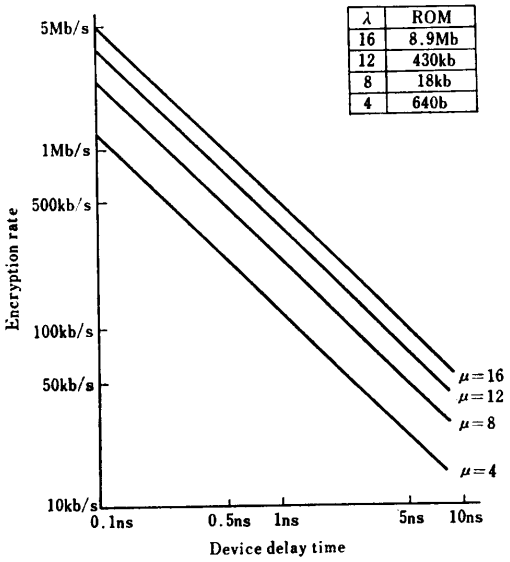

図 4 暗号計算速度と素子遅延時間
Fig. 4 Encryption rate and device delay time.

本稿の高速計算法を適用した暗号 LSI について，暗号計算速度と素子遅延時間 (t_{pd}) との関係を**図4**に示す．本稿で示した CMOS は t_{pd}=5 ns である（ファンアウト遅延，配線遅延を含む平均値）．

以上述べた試算は，制御部を除き，論理設計言語 HSL[9]のコーディングにより確認してある（コーディング量：約 2,000 枚）．

7. むすび

本稿では，RSA 公開鍵暗号の新しい高速計算法を提案し，現在の LSI 技術を用いれば，暗号計算速度 50 kb/s の暗号 LSI が実現できることを示した．

RSA 公開鍵暗号は，ディジタル署名が可能であること，計量的安全度が高いことなどから，最も有望な公開鍵暗号である．本稿に示す RSA 法の高速計算法により，RSA 公開鍵暗号が実用になる見通しを得た．

今後は数百ビット長の四則演算を単独に計算できる機能を，本稿の暗号 LSI に付加し，汎用マイクロプロセッサと接続することにより，RSA 法の暗号鍵生成器を設計する方式の検討が課題である．

謝辞 日頃ご指導いただく，通信制御研究室 酒井室長を始め，有益なご討論をいただいた基礎第八研究室の小山補佐に深謝します．

参考文献

1) Rivest, R. L. et al.: A Method for Obtaining Digital Signatures and Public Key Cryptosystems, *Comm. ACM*, Vol. 21, No. 2, pp. 120–126(1978).
2) Rivest, R. L.: A Description of a Single-chip Implementation of the RSA Public-Key Cryptosystem, NTC Conference Record, Vol. 3 (1980).
3) Miyaguchi, S.: Fast Encryption Algorithm for RSA Cryptographic System, COMPCON 82 Fall, Proc. pp. 672-678 (1982).
4) 小山：RSA 公開鍵暗号法のマスタ鍵，信学誌 D, Vol. J 65-D, No. 2, pp. 163-170 (1982).
5) 宮田他：拡張 Booth の方法と，Wallace Tree を用いた高速乗算器，信学誌 D, Vol. J 65-D, No. 6, pp. 807-808 (1982).
6) Brent, B. P. and Kung, H. T.: A Regular Layout for Parallel Adders, *IEEE Trans. Comput.*, Vol. C-31, No. 3, pp. 260-264 (1982).
7) Michelman, E. H.: The Design and Operation of Public Cryptosystems, NCC *AFIPS*

Nat. Comput. Conf. Expo. Conf. Proc., Vol. 48, p. 310 (1979).
8) Horiguchi, S. et al.: An Automatically Designed 32 b CMOS VLSI Processor, ISSCC 82, Session VI (1982).
9) 階層仕様記述言語 HSL, 日本通信技術(株) (1981).
10) Mead, C. and Conway, L.: *An Introduction for VLSI Systems*, Addison-Wesley, Reading, Mass. (1980).

付録 新計算法の証明

(1) 準備

(i) 仮定

$$-2n \leq \sum_{i=0}^{1} R_{j+1,i}' < n \qquad (A1)$$

(ii) 定義

$$P_j = \sum_{i=0}^{1} (2^{\mu} R_{j+1,i}') 2^{-m} + \sum_{i=0}^{\varphi-1} (M_1 \cdot Y_{j,i}) 2^{-m} \qquad (A2)$$

$$X_j = \sum_{i=0}^{1} \lfloor (2^{\mu} R_{j+1,i}') 2^{-m} \rfloor + \sum_{i=0}^{\varphi-1} \lfloor (M_1 \cdot Y_{j,i}) 2^{-m} \rfloor + S \qquad (A3)$$

$$Q_{j0} = \lfloor \{ \sum_{i=0}^{1} 2^{\mu} R_{j+1,i}' + \sum_{i=0}^{\varphi-1} M_1 \cdot Y_{j,i} \} \div n \rfloor \qquad (A4)$$

$$U_j = X_j / \lfloor n 2^{-m} \rfloor - P_j / (n 2^{-m}) \qquad (A5)$$

(iii) P_j と X_j の関係 (証明は(7))

$$\left. \begin{aligned} -(2+\omega) 2^{\mu} \cdot n 2^{-m} &< P_j \\ P_j &< (2^{\mu+1} - 1 + \omega) \cdot n 2^{-m} \end{aligned} \right\} \qquad (A6)$$

$$X_i = P_j + S - \phi, \quad 0 \leq \phi < \varphi + 2 \qquad (A7)$$

$$-2^{\mu} < X_j < 2^{\mu} \qquad (A8)$$

$$0 < U_i < 1 \qquad (A9)$$

(2) 商の関係 (公式(8)・i 適用)

$$\begin{aligned} q_i - q_j' &= \lfloor X_j / \lfloor n 2^{-m} \rfloor \rfloor - \lfloor X_j \cdot v \cdot 2^{-\mu} \rfloor - w_j \\ &= \lfloor X_j \cdot 2^{-\mu} \cdot \{ 2^{\mu} / \lfloor n 2^{-m} \rfloor - 2^{\mu} / (n 2^{-m}) \} \rfloor + r_{1j} - w_j = r_{1j} - 1 \end{aligned}$$

$$\begin{aligned} q_j - Q_{j0} &= \lfloor X_j / \lfloor n 2^{-m} \rfloor \rfloor - Q_{j0} \\ &= \lfloor U_j \rfloor + r_{2j} = r_{2j} \end{aligned}$$

ただし, $r_{1j}=0, 1$, $r_{2j}=0, 1$

$$\left. \begin{aligned} &\therefore \quad q_j' = Q_{j0} + r_j' \\ &\text{ただし} \quad r_1' = 1 + r_{2j} - r_{1j} = 0, 1 \text{ or } 2 \end{aligned} \right\} \qquad (A10)$$

式(A2), (A4), (A6)から

$$-(2+\omega) 2^{\mu} < Q_{j0} < (2^{\mu+1} - 1 + \omega) \qquad (A11)$$

式(A10), (A11), 式(19)から

$$Q_j' = Q_{j0} + r_j, \quad r_j = 0, 1 \text{ or } 2 \qquad (A10)'$$

(3) v の範囲: 式(8), (14), (15)から v の範囲が求められる.

(4) 剰余の範囲: 式(A10)′を式(21)に代入し, r_j を左辺に移項する. このとき右辺は0と n の間にあるから次式が成立する.

$$0 \leq \sum_{i=0}^{1} R_{j,i}' + r_{j,i} < n \qquad (A12)$$

$$\therefore \quad -2n \leq \sum_{i=0}^{1} R_{j,i}' < n \qquad (A13)$$

(5) 繰返し計算: $j=l$ のとき, 式(16)から式(A1)が成立する. よって式(A13)が成立する. この結果 $j=l-1$ についても, 式(A1)と式(A12)が成立する. 同様にして, $j=l-2, \cdots, 1$ について, 式(A1)と式(A13)が成立する. 式(21)の両辺に $2^{(j-1)\mu}$ を乗じ, この結果に $\sum_{j=l}^{1}$ の演算を行う. 式(16)を考慮して次式をうる.

$$\sum_{i=0}^{1} R_{1,i}' = M_1 \times M_2 - n \times \sum_{j=1}^{l} Q_j' \cdot 2^{(j-1)\mu} \qquad (A14)$$

$(M_1 \times M_2) \div n$ の剰余を R とする. Q_j' は整数であるから, 式(A13), (A14)とから次式をうる.

$$R = \sum_{i=0}^{1} R_{1,i}' + r_1 \times n, \quad r_1 = 0, 1 \text{ or } 2 \qquad (A15)$$

式(A15)は, 式(23)以降の補正計算がたかだか2回であることを示す.

(6) 平均補正計算回数

式(A10)の導出において, q_j', q_j および Q_{j0} の小数部分が均等に分布すると近似し, 公式(8), ii を適用する. $r_{1j}=1$ なる期待値は $\exp\{r_{1j}=1\}=1/2$ となる. 同様にして $\exp\{r_{2j}=1\}=1/2$, $\exp\{r_j=0\}=1/4$, $\exp\{r_j=1\}=1/2$, $\exp\{r_j=2\}=1/4$ が成立する. したがって, 平均補正計算回数 $=0\times1/4+1\times1/2+2\times1/4=1$ が得られる.

なお, 式(A1)は, R_j ブロックの加算が, 符号ビットを含め, $L+2$ ビットであることを意味する.

(7) P_j と X_j の関係: 式(A1)から, $\sum_{i=0}^{1} 2^{\mu} \cdot R_{j+1,i}$ の範囲を求め, 式(8)～(13)から, $\sum_{i=0}^{\varphi-1} M_1 \cdot Y_{j,i}$ の範囲を求める. これらから, 式(A6)が導ける.

式(17)の右辺に, 公式(8), iii を適用し, 式(A2)

を代入する.

$$X_j=\lfloor P_j\rfloor+S-\Psi',\ \Psi'=0,1,\cdots,\varphi+1$$

次に $P_j=\lfloor P_j\rfloor+\varepsilon$, $0\leq\varepsilon<1$, $\Psi=\Psi'+\varepsilon$ とおき, 式(A 7)をうる.

P_j と S の最大値, Ψ の最小値と式(14)より $X_j<2^n$, $U_j<1$ をうる. P_j の S の最小値, Ψ の最大値より, $-2^n<X_j$, $0<U_j$ をうる.

(8) 公式:

i. $\lfloor x\rfloor-\lfloor y\rfloor=\lfloor x-y\rfloor+\delta$

ii. $\exp\{\delta=1\}=\exp\{\alpha_x<\alpha_y\}$

iii. $\sum_{i=1}^{\varphi}\lfloor x_i\rfloor=\lfloor\sum_{i=1}^{\varphi}x_i\rfloor-r_\varphi$

ただし, x, y, x_i は実数. $\delta=0,1$. $r_\varphi=0,1,\cdots,\varphi-1$. α_x, α_y は x, y の小数部分.

(昭和58年2月2日受付)
(昭和58年4月19日採録)