(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3726966号
(P3726966)
(45)発行日　平成17年12月14日(2005.12.14)　(24)登録日　平成17年10月7日(2005.10.7)

(51)Int.Cl.[7]　F I
G06F 7/52　G06F 7/52 310A
G09C 1/00　G09C 1/00 650A

請求項の数 13　(全 14 頁)

(21)出願番号　特願2003-15301(P2003-15301)
(22)出願日　平成15年1月23日(2003.1.23)
(65)公開番号　特開2004-227344(P2004-227344A)
(43)公開日　平成16年8月12日(2004.8.12)
審査請求日　平成15年7月31日(2003.7.31)

(73)特許権者　390009531
インターナショナル・ビジネス・マシーンズ・コーポレーション
ＩＮＴＥＲＮＡＴＩＯＮＡＬ　ＢＵＳＩＮＥＳＳ　ＭＡＳＣＨＩＮＥＳ　ＣＯＲＰＯＲＡＴＩＯＮ
アメリカ合衆国１０５０４　ニューヨーク州　アーモンク　ニュー　オーチャード　ロード
(74)代理人　100086243
弁理士　坂口　博
(74)代理人　100091568
弁理士　市位　嘉宏
(74)代理人　100108501
弁理士　上野　剛史

最終頁に続く

(54)【発明の名称】乗算器及び暗号回路

(57)【特許請求の範囲】
【請求項１】
乗算対象である２つの入力値に対して部分積を求め、当該部分積を冗長２進形式で加え合わせるワラスツリー部と、
前記ワラスツリー部から出力される冗長２進数を２の補数形式に変換する桁上げ加算器とを備え、
前記ワラスツリー部は、
前記部分積の値を桁ごとに加算する合計値計算部と、
前記合計値計算部による加算における桁上がり値を加算する桁上がり値計算部と
を備えることを特徴とする乗算器。
【請求項２】
前記合計値計算部の計算結果を２の拡大体に対する乗算結果として出力することを特徴とする請求項１に記載の乗算器。
【請求項３】
前記桁上げ加算器は、前記合計値計算部の計算結果と前記桁上がり値計算部の計算結果とを加算して整数に対する乗算結果として出力することを特徴とする請求項１に記載の乗算器。
【請求項４】
前記合計値計算部は、前記部分積と共に他の所定の値を対応する桁ごとに加算することによって積和演算を行うことを特徴とする請求項１に記載の乗算器。

【請求項５】
乗算対象である２つの入力値に対して部分積を求め、半加算器及び全加算器を用いて当該部分積を加算し、当該入力値の乗算を行う乗算器において、
前記部分積における合計値を各桁別に計算し、前記入力値が２の拡大体である場合に乗算結果として出力する乗算手段と、
前記乗算手段の計算において生じる桁上がり値を加算する桁上がり値加算手段と、
前記乗算手段の計算結果と前記桁上がり値加算手段の計算結果とを加算し、前記入力値が整数である場合に乗算結果として出力する加算手段と
を備えることを特徴とする乗算器。
【請求項６】
前記乗算手段は、前記半加算器及び前記全加算器から出力されるＸＯＲ（排他的論理和）演算による加算項のみを集めて外部へ出力することを特徴とする請求項５に記載の乗算器。
【請求項７】
前記桁上がり値加算手段は、前記乗算手段にて加算される加算項以外の全ての項を集めて半加算器及び全加算器により桁上がり項と加算項とを含めた加算を行うことを特徴とする請求項６に記載の乗算器。
【請求項８】
前記乗算手段における部分積の加算項に他の加算項を加えることによって積和演算を行うことを特徴とする請求項６に記載の乗算器。
【請求項９】
データの暗号化または復号化のための演算を行う演算手段と、
前記演算手段による演算を制御する制御手段とを備え、
前記演算手段は、
半加算器及び全加算器を用いた乗算器であって、
演算対象である２つの入力値に対して部分積を求め、当該部分積を冗長２進形式で加え合わせるワラスツリー部と、
前記ワラスツリー部から出力される冗長２進数を２の補数形式に変換する桁上げ加算器とを備え、
前記ワラスツリー部は、
前記部分積の値を桁ごとに加算する合計値計算部と、
前記合計値計算部による加算における桁上がり値を加算する桁上がり値計算部と
を備えることを特徴とする暗号回路。
【請求項１０】
前記演算手段は、有限体ＧＦ(２^{n})上の演算を行う場合に前記合計値計算部の計算結果を出力し、有限体ＧＦ(ｐ)上の演算を行う場合に前記桁上げ加算器の計算結果を出力することを特徴とする請求項９に記載の暗号回路。
【請求項１１】
前記合計値計算部は、前記半加算器及び前記全加算器から出力されるＸＯＲ（排他的論理和）演算による加算項のみを集めて前記演算手段の外部へ出力することを特徴とする請求項９に記載の暗号回路。
【請求項１２】
前記桁上がり値計算部は、前記乗算手段にて加算される加算項以外の全ての項を集めて半加算器及び全加算器により桁上がり項と加算項とを含めた加算を行うことを特徴とする請求項９に記載の暗号回路。
【請求項１３】
データの暗号化または復号化のための演算を行う演算手段と、
前記演算手段による演算を制御する制御手段とを備え、
前記演算手段は、
乗算対象である２つの入力値に対して部分積を求め、半加算器及び全加算器を用いて当該

部分積を加算し、当該入力値の乗算を行う乗算器であって、
前記部分積における合計値を各桁別に計算し、前記入力値が有限体ＧＦ(２n)である場合に乗算結果として出力する乗算手段と、
前記乗算手段の計算において生じる桁上がり値を加算する桁上がり値加算手段と、
前記乗算手段の計算結果と前記桁上がり値加算手段の計算結果とを加算し、前記入力値が整数である場合に乗算結果として出力する加算手段と
を備えることを特徴とする暗号回路。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】

本発明は、コンピュータのプロセッサなどに用いられる演算装置に関し、特に有限体演算を行う乗算装置に関する。

【０００２】

【従来の技術】

今日、公開鍵暗号の方式としては、べき乗剰余演算（有限体ＧＦ(ｐ)上の演算、なおＧＦ(ｐ)は整数を素数ｐで除算した余りの集合）を利用したＲＳＡ（Rivest Shamir Adleman）方式が主流となっている。公開鍵暗号では鍵長（ビット数）が長いほど安全性が向上するが、このＲＳＡ方式の暗号では実用上十分な安全性を得るために必要な鍵長が５１２ビットから、１０２４ビット、２０４８ビットと増加してきており、演算時間と回路実装におけるリソースの増大が問題となっている。これに対し、楕円暗号を用いた公開鍵暗号方式（以下、楕円暗号方式）では、１６０ビットや２２４ビットの演算で、１０２４ビットや２０４８ビットのＲＳＡ方式と同等の安全性を確保することが可能である。

【０００３】

楕円暗号の基本演算は、ＲＳＡ暗号と同じＧＦ(ｐ)をベースにした演算と、２の拡大体ＧＦ(2ⁿ)による演算の２つに大別される。ＧＦ(2ⁿ)の演算は、ＸＯＲ（eXclusive OR排他的論理和）が基本であり、加算時に桁上がりを生じないため、整数の剰余演算によるＧＦ(ｐ)に対して高速である。しかしながら、ＧＦ(2ⁿ)上の楕円暗号方式においても、最も重要なアルゴリズムの１つである、楕円ＤＳＡ（デジタル署名アルゴリズム）による署名をサポートするためにはＧＦ(ｐ)上の乗剰余演算も必要とされる（例えば、非特許文献１参照）。

【０００４】

なお、乗剰余演算を高速に実行するアルゴリズムにモンゴメリ乗算がある（例えば、非特許文献２参照）。初期のアルゴリズムはＧＦ(ｐ)に対するものであり、加算器ベースのアルゴリズムであったが、今日では乗算器を使用したアルゴリズムやＧＦ(2ⁿ)上の演算への拡張がなされている。

【０００５】

【非特許文献１】

今井秀樹著、「符号理論」、電子情報通信学会、１９９０年

【非特許文献２】

Johann Groszschaedl, "A Bit-Serial United Multiplier Architecture for Finite Fields GF(p) and GF(2m)", C.K. Koc, D. Naccache, and C. Paar (Eds.): CHES 2001, LNCS 2162, p.202-219, 2001. Springer-Verlag Berlin Heidelberg 2001.

【非特許文献３】

I.F.Blake、N.P.Smart、G.Seroussi著、鈴木治郎訳、「楕円曲線暗号」、ピアソンエデュケーション、２００１年

【非特許文献４】

江藤良純、金子敏信　監修、「誤り訂正符号とその応用」、オーム社、１９９６年

【０００６】

【発明が解決しようとする課題】

上述したように、公開鍵暗号を楕円暗号にて実現する場合、ＧＦ(ｐ)による演算とＧＦ(

2^n)による演算とを行うことが必要となる。したがって、暗号回路においては、ＧＦ(ｐ)の演算を行う乗算器とＧＦ(2^n)の演算を行う乗算器とが必要となる。
ここで、８ビット程度の乗算器を用いる回路であれば、ＧＦ(ｐ)の演算を行う乗算器とＧＦ(2^n)の演算を行う乗算器との両方を載せて、セレクタで切り替える構成としても回路規模に与える影響は小さい。しかし、高速化のために３２ビットや６４ビットの乗算器を使用する場合、乗算器のゲート数が増加するビット数の自乗のオーダーで増加し、またセレクタや配線も増大するため、ＧＦ(ｐ)の演算を行う乗算器とＧＦ(2^n)の演算を行う乗算器とを両方搭載するとすれば、回路規模の増大が無視できない。
したがって、公開鍵暗号で扱われる１６０ビットや１０２４ビットといった数の演算を行う暗号回路の場合、暗号回路を搭載する機器や暗号回路自体の小型化を図るには、ＧＦ(ｐ)の演算を行う乗算器とＧＦ(2^n)の演算を行う乗算器とを個別に搭載することは好ましくない。
【０００７】
ところで、モンゴメリ乗算や楕円演算においては、ＧＦ(ｐ)上の演算とＧＦ(2^n)上の演算とのアルゴリズムはほとんど同じである。そのため、回路化した場合のデータパスも、乗算コア自体を除けばほとんどそのまま共有することが可能である。上記の非特許文献２には、ＧＦ(ｐ)上の演算とＧＦ(2^n)上の演算とに共用可能な乗算器が開示されている。
しかし、かかる従来の乗算器は、加算器をビット数分繰り返して使用し積を計算するシリアル乗算器であり、単にＧＦ(2^n)の演算時にキャリー（carry：桁上がり）をゲートして無視（Disable）するものである。シリアル乗算器で整数乗算を行うためには、毎サイクル毎に、キャリーを最下位ビット（ＬＳＢ：Least Significant Bit）から最上位ビット（ＭＳＢ：Most Significant Bit）まで伝播させるか、冗長２進数のまま途中結果をビット数の２倍のレジスタに保存する必要がある。そのため、演算の高速化が困難であり、高速化するためには回路規模を大幅に増大しなければならない。
【０００８】
また上述したように、公開鍵暗号では１６０ビットや１０２４ビットといった数が扱われ、その暗号回路では高速化のために非常に長い加算器が用いられる。そのため、データを高速に転送するためには、バス幅を加算器のビット長まで広げる必要があるため、物理的なチップサイズの増大を招来する。また、入力が１０２４ビット×２、出力が１０２４ビットのバスともなると、汎用のＡＳＩＣ用ライブラリによる自動論理合成および配置配線では実装することができず、人手によるカスタムレイアウトという繁雑な作業を要することもある。一方、バス幅を広げないとすれば、処理の実行時に演算対象のデータがそろうまで加算器を待たせる制御を行わなくてはならず、実行性能の低下を招く。
【０００９】
以上、公開鍵暗号における暗号回路に用いられる乗算回路について論じたが、暗号に限らず、ＧＦ(ｐ)上の演算とＧＦ(2^n)上の演算とが要求される、符号理論を応用した種々のアプリケーションに関しても同様のことが言える。符号理論を応用した他のアプリケーションの例としては、誤り訂正符号による誤り訂正回路がある（例えば、非特許文献３、４参照）。
【００１０】
そこで本発明は、回路規模を増大することなく、通常の整数乗算器およびＧＦ(2^n)上の乗算器として使用することが可能なパラレル乗算回路を提供することを目的とする。
また本発明は、ＧＦ(ｐ)及びＧＦ(2^n)の２つの演算を要する楕円暗号やＲＳＡ方式を初めとする乗剰余系の公開鍵暗号を１つのアクセラレータコアで実現する暗号回路を提供することを他の目的とする。
【００１１】
【課題を解決するための手段】
上記の目的を達成する本発明は、次のように構成された乗算器として実現される。すなわち、この乗算器は、乗算対象である２つの入力値に対して部分積を求め、この当該部分積を冗長２進形式で加え合わせるワラスツリー部と、このワラスツリー部から出力される冗

長２進数を２の補数形式に変換する桁上げ加算器とを備え、ワラスツリー部は、部分積の値を桁ごとに加算する合計値計算部と、この合計値計算部による加算における桁上がり値を加算する桁上がり値計算部とを備えることを特徴とする。
かかる乗算器において、２の拡大体（有限体ＧＦ(2^n)）に対する乗算を行う場合には、合計値計算部の計算結果のうち桁上がりを除く各桁の加算結果を当該乗算の結果として出力することができる。また、桁上げ加算器は、合計値計算部の計算結果と桁上がり値計算部の計算結果とを加算し、整数に対する乗算を行う場合には、この桁上げ加算器の計算結果を当該乗算の結果として出力することができる。

【００１２】

また、本発明は、乗算対象である２つの入力値に対して部分積を求め、半加算器及び全加算器を用いて部分積を加算し、入力値の乗算を行う、次のように構成された乗算器としても実現される。すなわち、この乗算器は、部分積における合計値を各桁別に計算し、入力値が２の拡大体である場合に乗算結果として出力する乗算手段と、この乗算手段の計算において生じる桁上がり値を加算する桁上がり値加算手段と、乗算手段の計算結果と桁上がり値加算手段の計算結果とを加算し、入力値が整数である場合に乗算結果として出力する加算手段とを備えることを特徴とする。
この乗算器において、乗算手段は、前記半加算器及び全加算器から出力されるＸＯＲ演算による加算項のみを集めて外部へ出力する。また桁上がり値加算手段は、乗算手段にて加算される加算項以外の全ての項を集めて半加算器及び全加算器により桁上がり項と加算項とを含めた加算を行う。

【００１３】

さらに本発明は、これらの乗算器を演算手段として搭載した種々の回路（アプリケーション）としても実現される。典型的な例としては、暗号回路における暗号化または復号化処理のための演算を行う演算器として搭載することができる。

【００１４】

【発明の実施の形態】

以下、添付図面に示す実施の形態に基づいて、この発明を詳細に説明する。
本実施の形態は、パラレル乗算器において、ＧＦ(ｐ)上の演算とＧＦ(2^n)上の演算とを実現する。ＧＦ(ｐ)上の乗算（整数乗算）を行うパラレル乗算器は、部分積（各桁の数ごとの積）をキャリーセーブ形式（冗長２進形式）で加え合わせていくワラスツリー（Ｗallace tree）部と、ワラスツリー出力の冗長２進数を２の補数形式に変換する桁上げ加算器とによって構成される。

【００１５】

乗算器へのｎビットの２入力をＡ＝（a_{n-1}，・・・，a_1，a_0）、Ｂ＝（b_{n-1}，・・・，b_1，b_0）とするとき、ワラスツリー部ではビットごとの部分積を半加算器（ＨＡ）と全加算器（ＦＡ）とによって加え合わせていく。
図１は共に８ビットの入力Ａ、Ｂを例とした乗算のイメージを示す図、図２は図１の乗算において部分積を計算するための回路構成を示す図、図３は図２の回路で計算された部分積を加え合わせるための回路構成を示す図である。
図２に示す回路によって、n^2個の部分積a_ib_j（ｉ，ｊ＝０，１，２，３，４，５，６，７）がそれぞれ算出され、図３に示す回路の対応する桁の全加算器または半加算器へ送られる。そして、図３に示す回路の出力（d_0，d_1，～，d_{15}）が、桁上げ加算器を経て乗算結果として出力される。

【００１６】

図４は半加算器の構成を示す図、図５は全加算器の構成を示す図である。
半加算器は、次の数１式のように２ビットの入力に対してｃａｒｒｙ（桁上がり値）とｓｕｍ（合計値）の２ビットを計算する。

【数１】

$$\begin{cases} \mathrm{sum}_{HA}(x_0, x_1) = x_0 \oplus x_1 \\ \mathrm{carry}_{HA}(x_0, x_1) = x_0 \cdot x_1 \end{cases}$$

また、全加算器は、数２式のように３入力に対してｃａｒｒｙとｓｕｍの２ビットを計算する。

【数２】

$$\begin{cases} \mathrm{sum}_{FA}(x_0, x_1, x_2) = x_0 \oplus x_1 \oplus x_2 \\ \mathrm{carry}_{FA}(x_0, x_1, x_2) = x_0 \cdot x_1 + x_1 \cdot x_2 + x_2 \cdot x_0 \end{cases}$$

上の各式において、「・」、「＋」は、それぞれＡＮＤ（論理積）、ＯＲ（論理和）を意味している。また「○」に「＋」の演算記号はＸＯＲ（排他的論理和）を意味している。

【００１７】

上記のような整数乗算回路に対して、ＧＦ(2^n)上の乗算を行うパラレル乗算器は、ｃａｒｒｙは発生せず、ワラスツリー部において、部分積を桁ごとに全てＸＯＲするだけである。すなわち、図１と同様の８ビットの入力Ａ、Ｂに対する演算を行う乗算回路において、ワラスツリー部は、部分積を計算する図２と同様の回路と、得られた部分積を桁ごとにＸＯＲ演算する回路とを備える。また、ＧＦ(2^n)上の乗算ではｃａｒｒｙが発生しないため、ワラスツリー部による部分積のＸＯＲ演算の結果が直ちに乗算結果であり、桁上げ加算器による変換を必要としない。

図６は、図２の回路で計算された部分積をＸＯＲ演算するための回路構成を示す図である。

【００１８】

以上の点に鑑み、本実施の形態による乗算器は、上の数１、２式における$\mathrm{sum}_{HA}()$及び$\mathrm{sum}_{FA}()$のＸＯＲを用いて、この部分にＧＦ(2^n)の乗算器を埋め込む構成とする。

図７は、本実施の形態による乗算器の構成を示す図である。

図７を参照すると、本実施の形態による乗算器１００はワラスツリー部１１０と桁上げ加算器１２０とを備え、ワラスツリー部１１０は、入力値に対して桁ごとに合計値（ｓｕｍ）と桁上がり値（ｃａｒｒｙ）とを計算するｓｕｍ計算部１１１と、ｓｕｍ計算部１１１の計算による桁上がり値を加算するｃａｒｒｙ計算部１１２と、ｓｕｍ計算部１１１及びｃａｒｒｙ計算部１１２の計算結果を加算して桁上げ加算器１２０に渡すＨＡ／ＦＡアレイ１１３とを備える。ｓｕｍ計算部１１１は、部分積の加算において半加算器及び全加算器から出力されるＸＯＲ演算による加算項だけを集めて出力するものである。また、ｃａｒｒｙ計算部１１２は、ｓｕｍ計算部１１１にて加算される加算項以外の全ての項を集めて半加算器及び全加算器により桁上がり項と加算項とを含めた加算を行うものである。桁上げ加算器１２０は、ＨＡ／ＦＡアレイ１１３の出力を補数形式に変換して出力するもので、通常のパラレル乗算器における桁上げ加算器と同様である。

【００１９】

すなわち、本実施の形態の乗算器１００は、まず通常の整数乗算器と同様に、半加算器と全加算器とのアレイ（HA/FA array）からなるｓｕｍ計算部１１１によりｃａｒｒｙとｓｕｍとを生成する。そして、通常の整数乗算器のワラスツリー部では、同じ桁にあるｃａｒｒｙとｓｕｍとは区別せずに順次加算するのに対し、本実施の形態の乗算器１００におけるワラスツリー部１１０では、ｃａｒｒｙ計算部１１２とｓｕｍ計算部１１１とを用いて、ｃａｒｒｙとｓｕｍとを別のツリーで加算していく。

ｓｕｍ計算部１１１にてｓｕｍだけを加算（つまりＸＯＲ）していくことにより、２ｎ－１ビットのＧＦ(2^n)の乗算結果が得られる。最も深いパスでは、ｎビットの部分積を１ビットにＸＯＲ加算される。全加算器は３ビット入力に対して１ビットのｓｕｍを出力するので、ｓｕｍ計算部１１１全体の遅延は、ＦＡ換算で

【数３】

$$\lceil \log_3 n \rceil$$

段となる。

【００２０】

一方、ｃａｒｒｙ計算部１１２の回路構成は、様々な組み方が可能であるが、３ビット入力に対して全加算器は２ビットのｃａｒｒｙとｓｕｍとを出力するので、最大遅延はＦＡ換算で凡そ

【数４】

$$\lceil \log_{3/2} n \rceil$$

段となる。

【００２１】

図７を参照して明らかなように、本実施の形態による乗算器１００は、ＧＦ(ｐ)の乗算器の内部にＧＦ(2ⁿ)の乗算器に相当するｓｕｍ計算部１１１が埋め込まれた構成となっている。したがって、ＧＦ(2ⁿ)の乗算を行う場合は、図７に示すように、ｓｕｍ計算部１１１の計算結果をそのまま出力すれば良い。

上述したｓｕｍ計算部１１１の遅延とｃａｒｒｙ計算部１１２の遅延とから、ｓｕｍ計算部１１１の計算に要する時間の方が圧倒的に短い。そのため、図７に示すようにｓｕｍ計算部１１１（ＧＦ(2ⁿ)の乗算器に相当する部分）を分離したことによりｃａｒｒｙ計算部１１２が処理を待たされることはない。また、ｓｕｍ計算部１１１は、もともと整数乗算器に含まれていたＸＯＲ項を集めたものであるので、乗算器１００全体を構成する回路が増加することもない。

すなわち、本実施の形態による乗算器１００は、処理速度を低下させることなく、かつ回路規模を増大することなく、ＧＦ(ｐ)の乗算およびＧＦ(2ⁿ)の乗算が可能となっている。

【００２２】

図８は、図１に示した８ビット×８ビットの乗算を行う本実施の形態の乗算器１００におけるワラスツリー部１１０の構成例を示す図である。なお、図８では、半加算器を「Ｈ」と記述したブロック、全加算器を「Ｆ」と記述したブロックで表現しているが、これは図４、５にしたがっている。

図８に示すワラスツリー部１１０は、５つのステージからなる。また図８において、三角印（△）の入力は部分積、白丸印（○）の入出力はＧＦ(ｐ)及びＧＦ(2ⁿ)の演算における合計値（ｓｕｍ、以下、第１ｓｕｍと称す）、黒丸印（●）の入出力はＧＦ(ｐ)の演算のみにおける合計値（ｓｕｍ、以下、第２ｓｕｍと称す）、四角印（■）の入出力はＧＦ(ｐ)の演算における桁上がり値（ｃａｒｒｙ）を意味する。なお、図８に示すワラスツリー部１１０の構成は、説明の簡単のため、多少の無駄を含んで表現されている。

【００２３】

したがって、第１ステージでは各桁の部分積が入力され、ｃａｒｒｙと第１ｓｕｍとが出力される。

また、第２ステージでは第１ステージで得られたｃａｒｒｙが加算されて第２ｓｕｍが出力されると共に、第１ステージで得られた第１ｓｕｍが加算されてｃａｒｒｙと第１ｓｕｍとが出力される。ここで、第１ステージと第２ステージにおける第１ｓｕｍを加算する段とが、図７に示したワラスツリー部１１０のｓｕｍ計算部１１１に対応している。すなわち、この第２ステージで出力される第１ｓｕｍが、ＧＦ(2ⁿ)の乗算結果である。

【００２４】

また、第３ステージでは第２ステージで得られたｃａｒｒｙ及び第２ｓｕｍが加算されて、加算結果のｃａｒｒｙ及び第２ｓｕｍが出力される。そして、第２ステージで得られた第１ｓｕｍが通過している。この第３ステージにおけるｃａｒｒｙ及び第２ｓｕｍを加算する段が、図７に示したワラスツリー部１１０のｃａｒｒｙ計算部１１２に対応している。

第４ステージでは、第３ステージで得られたｃａｒｒｙ及び第２ｓｕｍと、第２ステージで得られ第３ステージを通過した第１ｓｕｍとが加算され、加算結果のｃａｒｒｙ及び第２ｓｕｍが出力される。また、下位のいくつかの桁では第１ｓｕｍあるいは第２ｓｕｍが加算されずに通過している。この第４ステージが、図７に示したワラスツリー部１１０のＨＡ／ＦＡアレイ１１３に対応している。

第５ステージでは、第４ステージで得られたｃａｒｒｙ及び第２ｓｕｍと、第４ステージを通過した第１ｓｕｍ及び第２ｓｕｍとが加算されて、加算結果である第２ｓｕｍが出力される。この第５ステージが、図７に示した桁上げ加算器１２０に対応している。すなわち、この第５ステージで出力される第２ｓｕｍが、ＧＦ(ｐ)の乗算結果である。

【００２５】

次の数５式は、任意のｎビット乗算において、本実施の形態の乗算器１００における第１ステージの部分を表現したものである。

【数５】

$$g_{k,j} = a_i \cdot b_j \quad (k = i + j, 0 \le i, j < n)$$

$$\{s_{k,0}, \cdots, s_{k,\lceil (n-|k-n|)/3 \rceil}\}$$

$$= \begin{cases} \{sum_{FA}(g_{k,0}, g_{k,1}, g_{k,2}), \cdots, g_k\} & (|n-k| \bmod 3 = 0) \\ \{sum_{FA}(g_{k,0}, g_{k,1}, g_{k,2}), \cdots, sum_{HA}(g_{k,k-1}, g_k)\} & (|n-k| \bmod 3 = 1) \\ \{sum_{FA}(g_{k,0}, g_{k,1}, g_{k,2}), \cdots, sum_{FA}(g_{k,k-2}, g_{k,k-1}, g_k)\} & (|n-k| \bmod 3 = 2) \end{cases}$$

$$\{c_{k,0}, \cdots, c_{k,\lceil (n-|k-n|)/3 \rceil}\}$$

$$= \begin{cases} \{carry_{FA}(g_{k,0}, g_{k,1}, g_{k,2}), \cdots, carry_{HA}(g_{k,k-1}, g_k)\} & (|n-k| \bmod 3 = 1) \\ \{carry_{FA}(g_{k,0}, g_{k,1}, g_{k,2}), \cdots, carry_{FA}(g_{k,k-2}, g_{k,k-1}, g_k)\} & (|n-k| \bmod 3 = 0 \text{ or } 2) \end{cases}$$

【００２６】

なお、上記のように構成された本実施の形態の乗算器１００は、演算対象である入力Ａ（a_{n-1}，・・・，a_1，a_0）、Ｂ（b_{n-1}，・・・，b_1，b_0）の部分積$a_i b_j$以外に別の加算項を１つあるいは複数組み込んで、ワラスツリー部１１０を構成することが容易である。したがって、例えば入力Ａ、Ｂの乗算に入力Ｃを加算する積和演算機能も簡単に実現することができる。

図９は、図１に示した８ビットの入力Ａ、Ｂの乗算に、同じく８ビットの入力Ｃを対応する桁ごとに加算する積和演算のイメージを示す図である。また図１０は、図９に示した８ビットの積和演算を行う本実施の形態の乗算器におけるワラスツリー部１１０の構成例を示す図である。図１０における半加算器および全加算器の表現は図８の場合と同様である。

図１０のワラスツリー部１１０と図８のワラスツリー部１１０とを比較すると、Ａ×Ｂにおける下位８桁の部分積に入力Ｃ＝（c_7，c_6，・・・，c_0）を加算するための加算器が追加されているものの、各ステージでの処理の内容は同様であることが分かる。

【００２７】

以上説明した本実施の形態の乗算器１００は、１つの乗算器１００でＧＦ(ｐ)及びＧＦ(2^n)の２つの演算を行うことができるため、乗算器１００を組み込む回路では、回路規模を増大することなく、これらの演算を実装することが可能である。

また本実施の形態の乗算器１００は、ｎビット×ｎビットのパラレル乗算器であるため、例えばｎ＝３２ビットバスとした場合、１０２４ビット加算器によるシリアル乗算器と同程度の処理を、より高速に行うことが可能である（３２×３２＝１０２４）。したがって、演算対象であるデータのサイズ（データビット長）の変化に対して柔軟性及び回路量当たりの処理能力が非常に高い。

【００２８】

さらに本実施の形態の乗算器１００は、ＧＦ(ｐ)及びＧＦ(2^{n})の２つの演算を１つの乗算器１００で行うことが可能であるため、符号理論を応用した種々のアプリケーションにて使用可能である。最も典型的な例としては、楕円暗号方式による公開鍵暗号を実現する暗号回路の演算器として用いられる。

図１１は、本実施の形態による乗算器１００を組み込んだ公開鍵暗号回路の構成例を示す図である。

図１１に示す暗号回路は、データの暗号化または復号化の処理（以下、暗号処理）における制御手段としてのデータ長カウンタ２１０、暗号制御回路２２０、鍵シフトレジスタ２２１、アドレスレジスタ２３０及びメモリ制御回路２４０と、処理対象であるデータを保持する保持手段としてのメモリ２５１、２５２と、暗号処理の実行手段である積和演算器２６０とを備える。

【００２９】

この暗号回路では、データ長カウンタ２１０にて指定されるデータサイズで、暗号制御回路２２０及び鍵シフトレジスタ２２１により、データの暗号化または復号化に用いられるパラメータ（鍵等）が設定される。そして、アドレスレジスタ２３０及びメモリ制御回路２４０の制御により、メモリ２５１、２５２に保持されている入力データ及び暗号制御回路２２０にて設定されたパラメータが積和演算器２６０に渡される。積和演算器２６０は、当該入力データの暗号化または復号化のための演算処理を行う。この積和演算器２６０として、本実施の形態の乗算器１００を用いることができる。

【００３０】

この暗号回路において、積和演算器２６０は、暗号処理において、ＧＦ(ｐ)上の演算とＧＦ(2^{n})上の演算とを実行するものとする。例えば、楕円暗号方式においてＤＳＡ署名をサポートする場合や、楕円暗号方式とＲＳＡ方式を併用する場合などである。

ＲＳＡ方式の暗号処理における演算や楕円暗号方式のＤＳＡ署名ではＧＦ(ｐ)上の演算を要するので、積和演算器２６０を構成する乗算器１００の桁上げ加算器１２０の出力が演算結果として出力される。

一方、楕円暗号方式の暗号処理における演算ではＧＦ(2^{n})上の演算を要するので、積和演算器２６０を構成する乗算器１００のｓｕｍ計算部１１１の出力が演算結果として出力される。

【００３１】

また、本実施の形態の乗算器１００を使用可能な他のアプリケーションとして、誤り訂正回路がある。

図１２は、本実施の形態による乗算器１００を組み込んだ誤り訂正回路の構成例を示す図である。

図１２に示す誤り訂正回路は、シンドローム計算部３１０と、多項式生成部３２０と、誤り位置評価部３３０と、誤り値評価部３４０と、誤り訂正部３５０とを備える。ここで、シンドローム計算部３１０は、データの誤りパターンに応じたシンドロームを生成する。また、データの消失があった場合はその位置を決定する。多項式生成部３２０は、シンドロームとデータ消失位置の情報から、誤り位置を求める多項式（誤り位置多項式）と誤りのパターンを求めるための多項式（誤り値多項式）とを生成する。誤り位置評価部３３０は、多項式生成部３２０にて生成された誤り位置多項式に基づいて誤り位置を計算する。誤り値評価部３４０は、誤り位置評価部３３０で求めた誤り位置と多項式生成部３２０にて生成された誤り値多項式とに基づいて誤り値を計算する。誤り訂正部３５０は、誤り位置評価部３３０で求めた誤り位置に対応するデータに、誤り値評価部３４０で求めた誤り

値を排他的論理和することによって、誤りを訂正する。

【００３２】

この誤り訂正回路において、シンドローム計算部３１０、多項式生成部３２０、誤り位置評価部３３０及び誤り値評価部３４０を構成する積和演算回路３１１、３２１、３２２、３３１、３４１、３４２及び除算回路３４３として、本実施の形態の乗算器１００を用いることができる。そして、ＧＦ(2^n)上の演算結果が乗算器１００のｓｕｍ計算部１１１から出力される。

【００３３】

【発明の効果】

以上説明したように、本発明によれば、回路規模を増大することなく、通常の整数乗算器およびＧＦ(2^n)上の乗算器として使用することが可能な乗算回路を提供することができる。

【００３４】

また、本発明による乗算回路は、符号理論を応用した種々のアプリケーションにて使用可能であり、この乗算回路を実装することにより、例えば、ＧＦ(ｐ)及びＧＦ(2^n)の２つの演算を要する楕円暗号やＲＳＡ方式を初めとする乗剰余系の公開鍵暗号を１つのアクセラレータコアで実現する暗号回路を提供することができる。

【図面の簡単な説明】

【図１】　８ビットの入力を例とした乗算のイメージを示す図である。

【図２】　図１の乗算において部分積を計算するための回路構成を示す図である。

【図３】　図２の回路で計算された部分積を加え合わせるための回路構成を示す図である。

【図４】　半加算器の構成を示す図である。

【図５】　全加算器の構成を示す図である。

【図６】　図２の回路で計算された部分積をＸＯＲ演算するための回路構成を示す図である。

【図７】　本実施の形態による乗算器の構成を示す図である。

【図８】　図１に示した８ビット×８ビットの乗算を行う本実施の形態の乗算器におけるワラスツリー部の構成例を示す図である。

【図９】　図１に示した８ビットの乗算に、他の８ビットの入力を加算する積和演算のイメージを示す図である。

【図１０】　図９に示した８ビットの積和演算を行う本実施の形態の乗算器におけるワラスツリー部の構成例を示す図である。

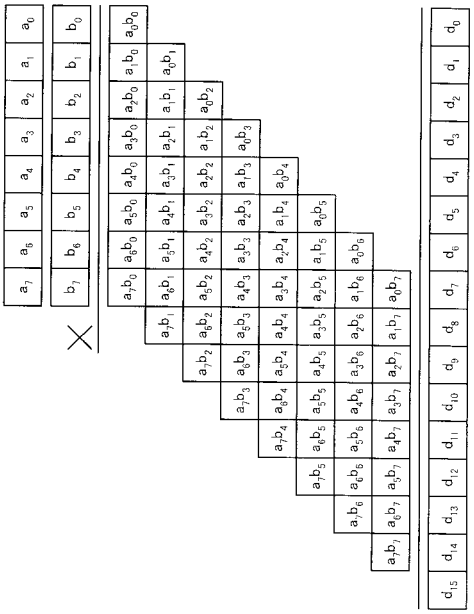
【図１１】　本実施の形態による乗算器を組み込んだ暗号回路の構成例を示す図である。

【図１２】　本実施の形態による乗算器を組み込んだ誤り訂正回路の構成例を示す図である。

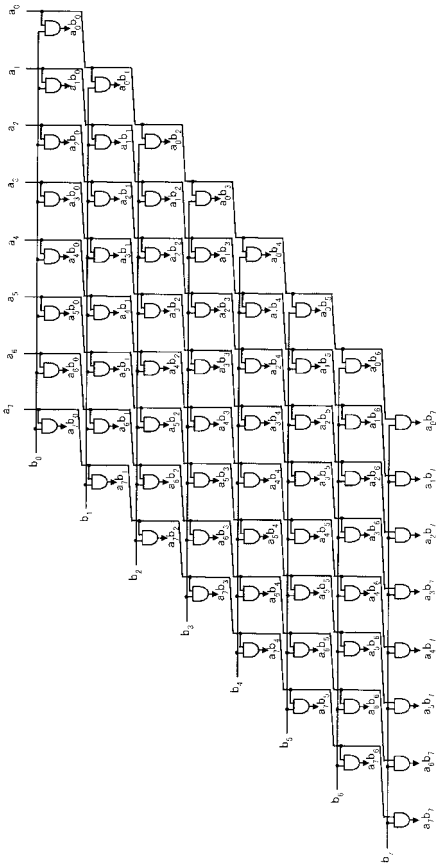
【符号の説明】

１００…乗算器、１１０…ワラスツリー部、１１１…ｓｕｍ計算部、１１２…ｃａｒｒｙ計算部、１１３…ＨＡ／ＦＡアレイ、１２０…桁上げ加算器

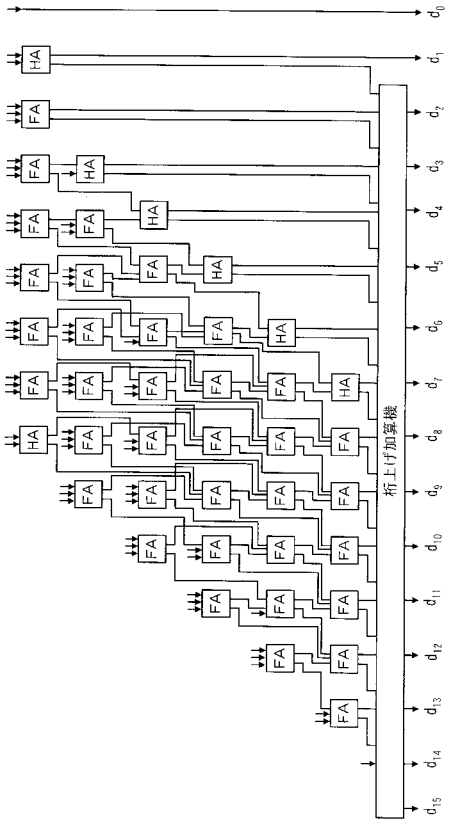
【図１】



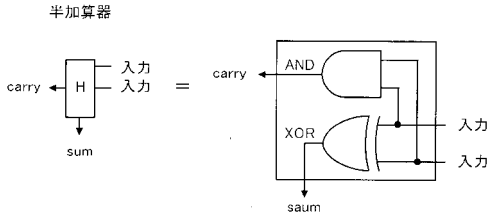
【図２】



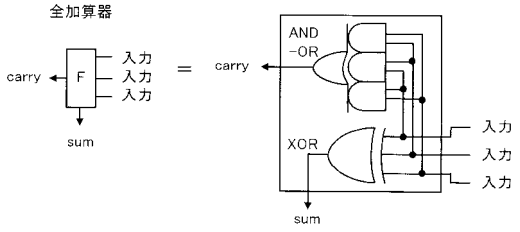
【図３】


HA
FA
桁上げ加算機

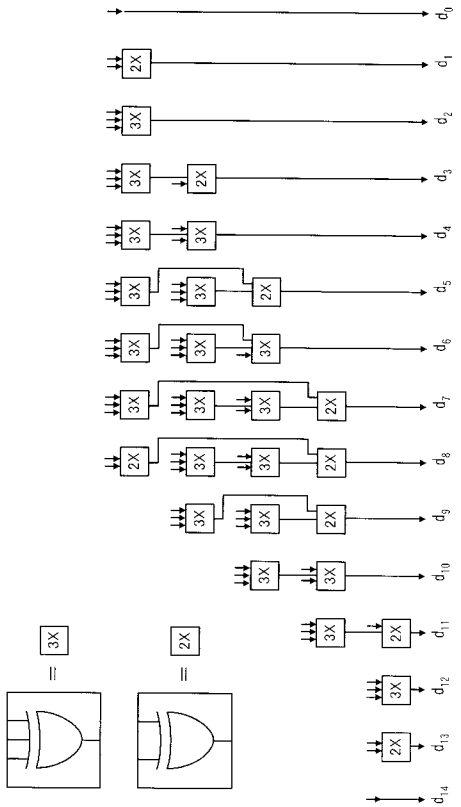
【図４】


半加算器
carry
H
入力
入力
sum
=
carry
AND
XOR
入力
入力
saum

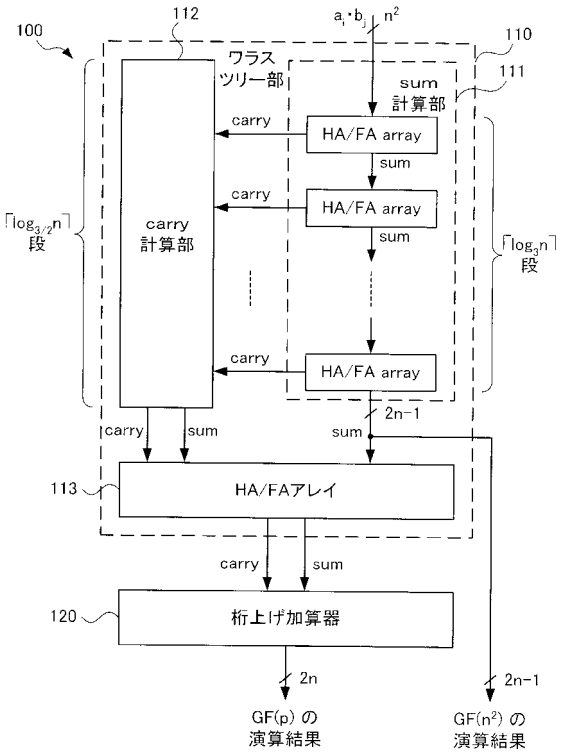
【図５】


全加算器
carry
F
入力
入力
入力
sum
=
carry
AND
-OR
XOR
入力
入力
入力
sum

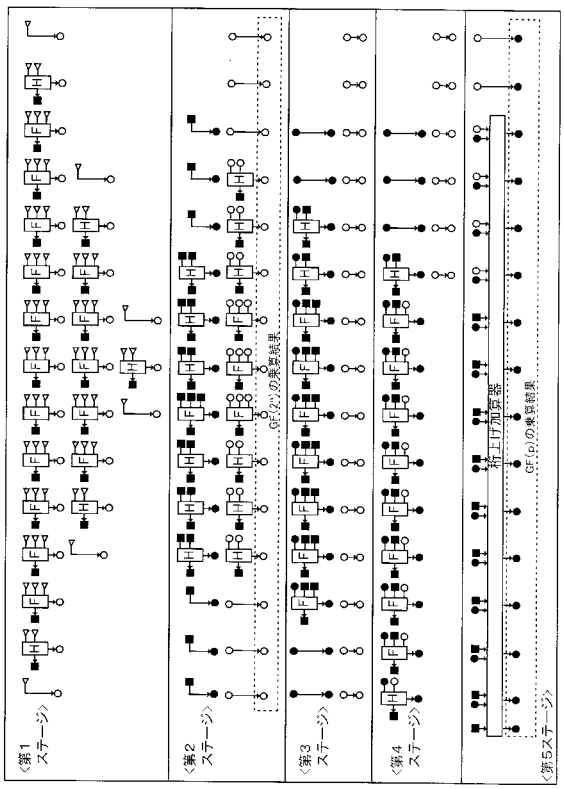
【図６】


3X
2X
d0
d1
d2
d3
d4
d5
d6
d7
d8
d9
d10
d11
d12
d13
d14

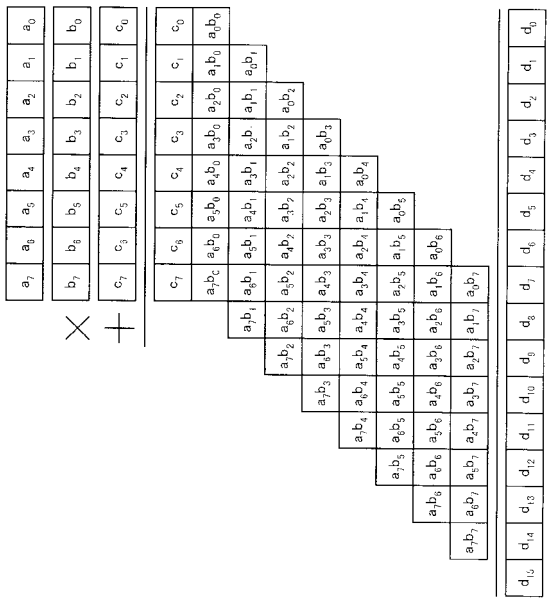
【図７】


100
112
a_i・b_j
n^2
110
111
ワラス
ツリー部
sum
計算部
carry
HA/FA array
sum
carry
HA/FA array
sum
carry
HA/FA array
⌈log3/2n⌉
段
carry
計算部
⌈log3n⌉
段
2n−1
carry
sum
sum
113
HA/FAアレイ
carry
sum
120
桁上げ加算器
2n
2n−1
GF(p) の
演算結果
GF(n²) の
演算結果

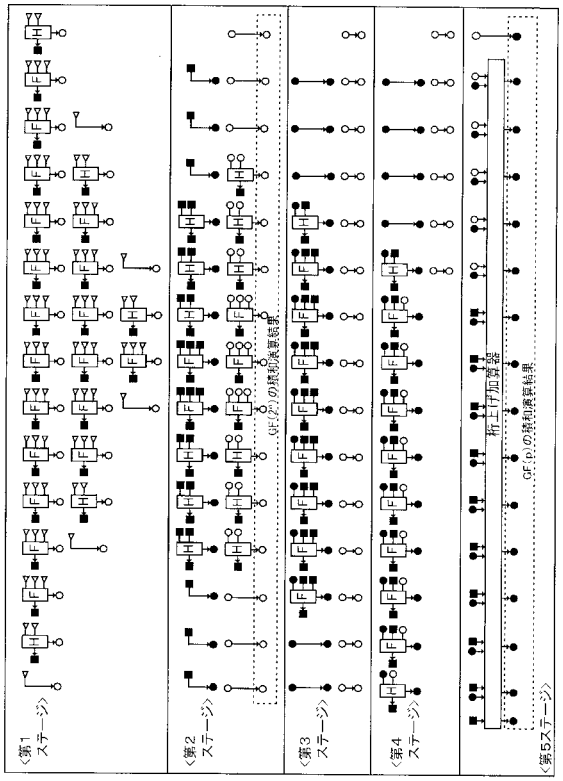
【図８】


〈第1
ステージ〉
〈第2
ステージ〉
〈第3
ステージ〉
〈第4
ステージ〉
〈第5ステージ〉
桁上げ加算器
GF(p)の演算結果

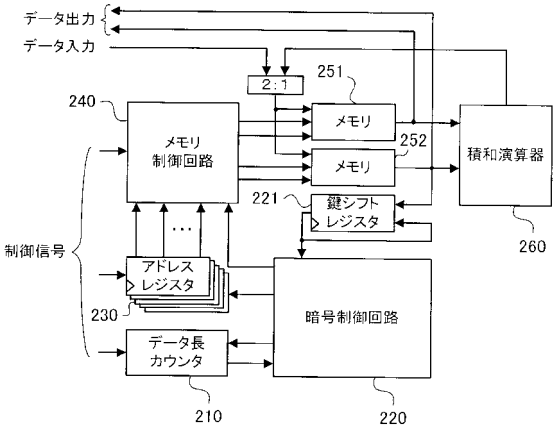
【図９】



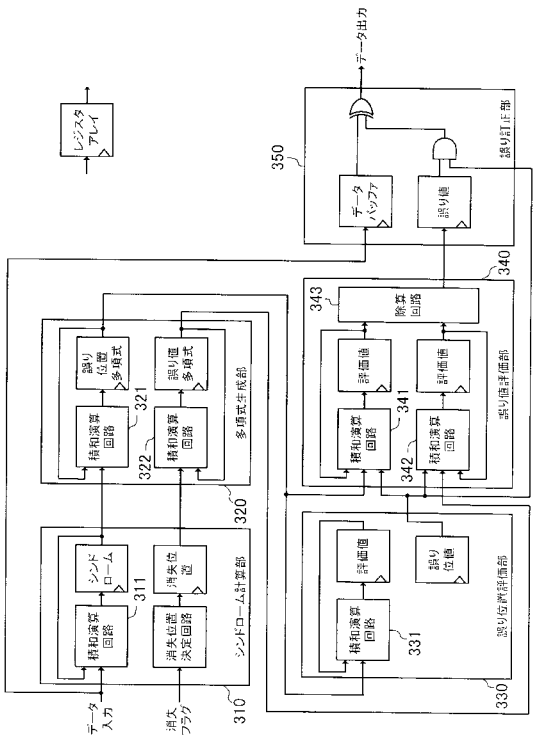
【図１０】


〈第1ステージ〉
〈第2ステージ〉
〈第3ステージ〉
〈第4ステージ〉
〈第5ステージ〉
GF(2^m)の積和演算結果
GF(2^m)の積和演算結果

【図１１】


データ出力
データ入力
2:1
251
240
メモリ制御回路
メモリ
メモリ
252
積和演算器
221
鍵シフトレジスタ
260
制御信号
アドレスレジスタ
230
暗号制御回路
データ長カウンタ
210
220

【図１２】


レジスタアレイ
データ出力
350
データバッファ
誤り値
誤り訂正部
340
343
除算回路
誤り値評価部
評価値
積和演算回路
341
342
誤り位置評価部
331
330
多項式生成部
誤り位置多項式
誤り値多項式
321
322
320
シンドローム計算部
シンドローム
消失位置
消失位置決定回路
311
310
データ入力
消失フラグ

フロントページの続き

(72)発明者　佐藤　証
神奈川県大和市下鶴間１６２３番地１４　日本アイ・ビー・エム株式会社　東京基礎研究所内
(72)発明者　高野　光司
神奈川県大和市下鶴間１６２３番地１４　日本アイ・ビー・エム株式会社　東京基礎研究所内

審査官　田中　友章

(56)参考文献　特開平１１－２４２５８６（ＪＰ，Ａ）
特開２００１－０３４１６７（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.[7]，ＤＢ名)
G06F　7/52　310
G09C　1/00　650