ICF3 後継 LSI のモンゴメリ乗算器検討

2016/06/18 版 (初版)

平山　直紀

著者概略

1992 年 早稲田大学理工学部　電気工学科卒

1994 年 早稲田大学理工学研究科 計算機工学専攻卒

1994 年 日立製作所 中央研究所　超高速プロセッサ部　入社

1995 年 日立製作所 汎用コンピュータ事業部に転勤

2005 年 日立製作所　退職

2006 年 株式会社ｉＣａｎａｌ設立、代表取締役社長

このファイルを公開しているサイト URL

http://www.canal.mokuren.ne.jp/memo/icf4mont.html

ICF3 について

日立製作所　中型メインフレーム MP5600EX の暗号装置　(1999 年 発売)

LSI 開発コード名　ICF3

当時、世界一高速な RSA 暗号の演算器だった。

参考 URL

http://icanal.idletime.tokyo/ESD.html

注意事項

このメモは ICF3 の開発後、後継 LSI の検討中に作成したメモ。ICF3 の監督だった東大卒の人と数人の、ごく小規模な検討会の資料です。

RSA検討会 2./1

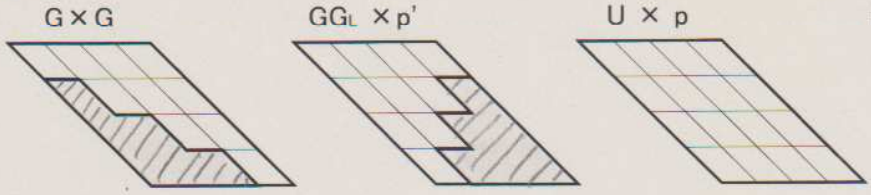
正方乗算器型の方式説明

1. 計算量の低減による高速化

$Y \times G\ R^{-1} \bmod p$

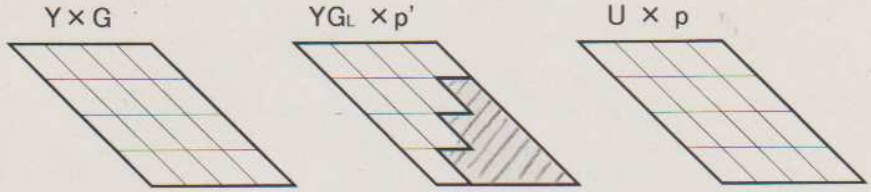
U = (Y × G × p') & 0xffff … ff;
A = (Y × G + U × p) >> (Rの長さ)
if(A >= p) A=A-p

演算不要



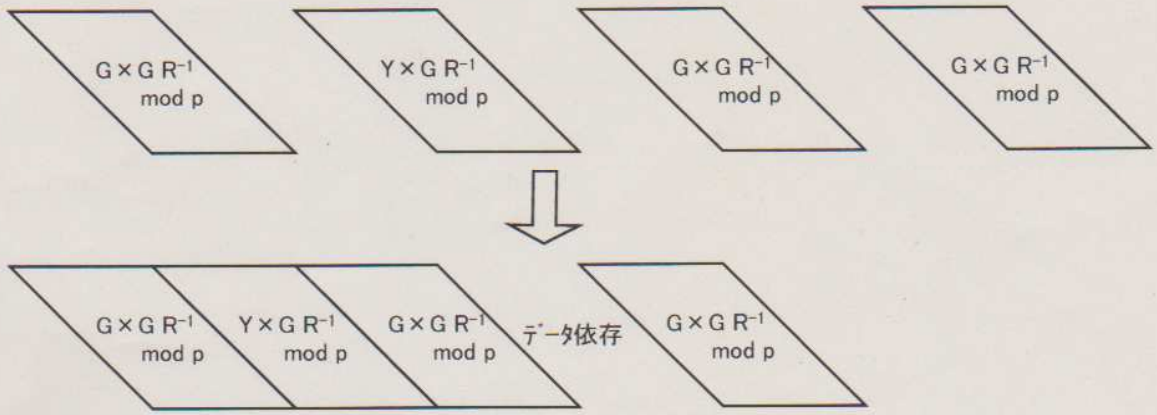
1024bit RSA演算での平均回数

1024回



512回

2. 演算のパイプラインによる高速化



3. 制御論理

ICF3の制御論理を拡張するだけで1.計算量の低減による高速化、2.演算パイプラインによる高速化を実現可能であり簡素な制御論理

00/07/25 Rev0

楕円曲線暗号用の高速演算機構の検討1

1. 目的

楕円曲線暗号演算で多用されるﾓﾝｺﾞﾒﾘ乗算は、bの値が異なっていても結果は同じになるが、演算の内容が異なる。ICF3ではb=2を選択しているが、$b=2^n$について詳細検討を行い、最も良い高速演算機構を考えていく。

2. $b=2^n$特徴

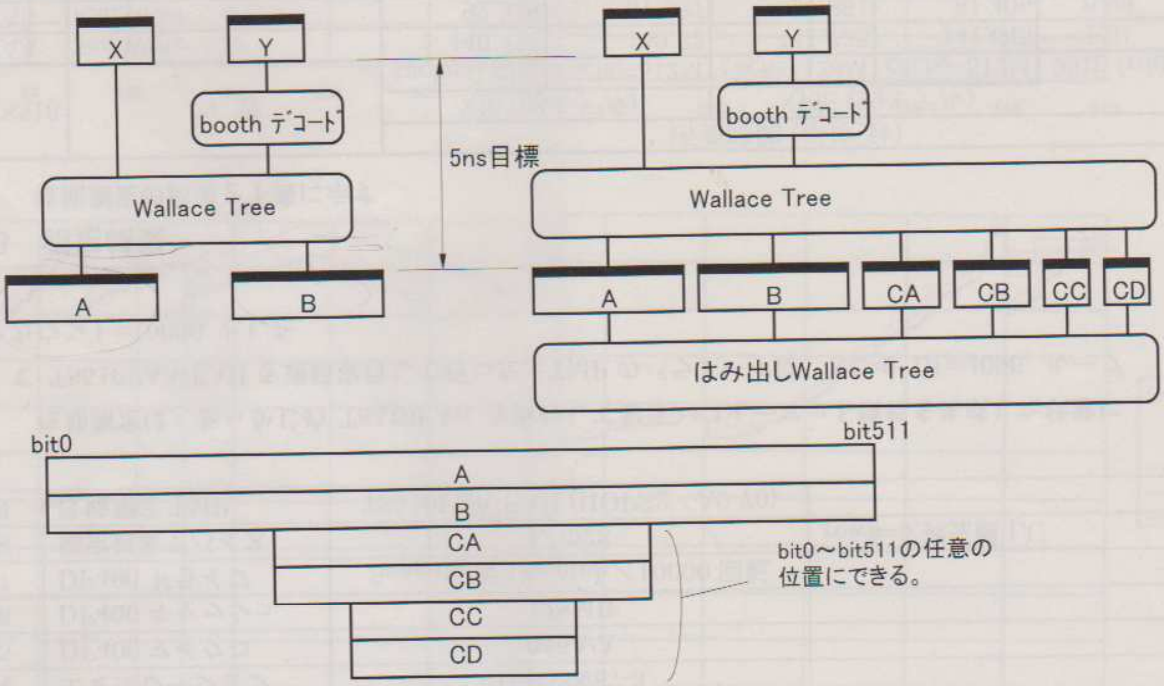
乗算器を高速化することでﾓﾝｺﾞﾒﾘ乗算の性能を上げることが可能だが、b=2では不要な逆数を必要とする。ただし、楕円暗号(ｽｶﾗ倍)では、最初に一度計算すればよく、ｽｶﾗ倍の演算に占める逆数演算の割合はわずかであるため問題はない。

3. 乗算器検討進捗

乗算器方式をあげ、ﾃﾞｨﾚｲと実装面積の評価を行うが、今回は、それぞれの方式におけるWallace Treeの簡易ﾃﾞｨﾚｲﾁｪｯｸ結果(打ち合わせまでに間に合ったもの)を示す。

前提条件：1cyc = 5ns
256bit × 256bit

3-1. Wallace Tree



		方式1	方式2	方式3	方式4	方式5	方式6	方式7
booth法		なし		2bit		2bit (注1)		4bit
はみ出し		なし	あり CA,CB: 96-384 CC,CD:160-320	なし	あり CA,CB:128-511 CC,CD:224-511	なし	あり CA,CB:448-511 CC,CD:448-551	なし
ﾃﾞｨﾚｲ		6558ps						5903ps
wallace Tree ﾃﾞｰﾀ	Basic ﾃﾞｨﾚｲ	1954ps	1626ps	1716ps	1478ps	3206ps	2921ps	1485ps
	ﾈｯﾄ数	195076	194176	167425	166083	121771	121392	83425
	CSA数	64770	64320	48768	48096	39228	39037	24192
	ﾊﾞｯﾌｧ数	0	0	20352	20352	3318	3318	10080
	はみ出しbit数	———	548	———	840	———	256	———

(注1) 符号処理を修正し、ﾈｯﾄ、CSA数を減らしたもの

Basicﾃﾞｨﾚｲ　Basic Delayの単純加算結果の最大値(booth decode分は入っていない)
CSA数　Wallace Treeを構成するCarry Save Adderの数、不完全なCSAも1としている
ﾈｯﾄ数　ﾏﾙﾁﾈｯﾄは、接続ﾎﾟｲﾝﾄ数-1でカウントされる。
ﾊﾞｯﾌｧ数　Wallace Treeで使用されるBufferの数

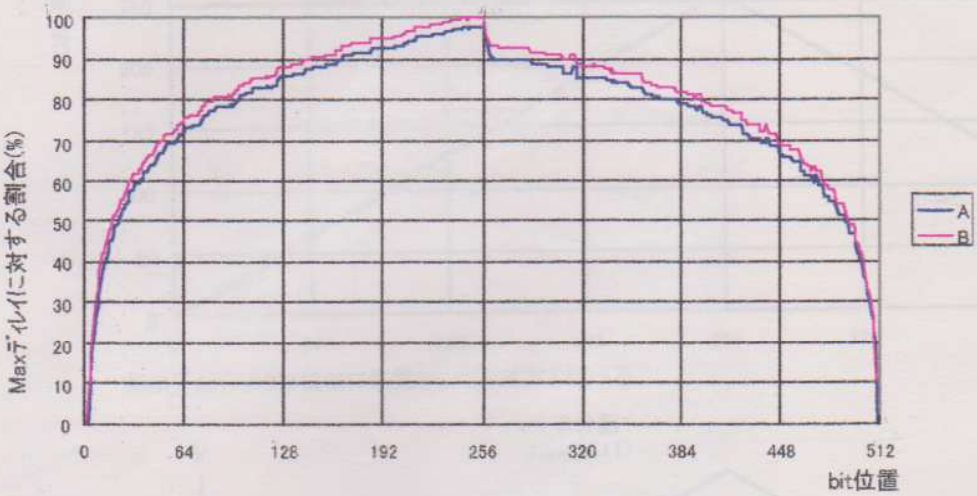
まとめ

結果は、符号配線処理のさせかたや、はみ出しWallaceのはみ出し位置や大きさ、により変わるため、今後、最良のものを選択していく。

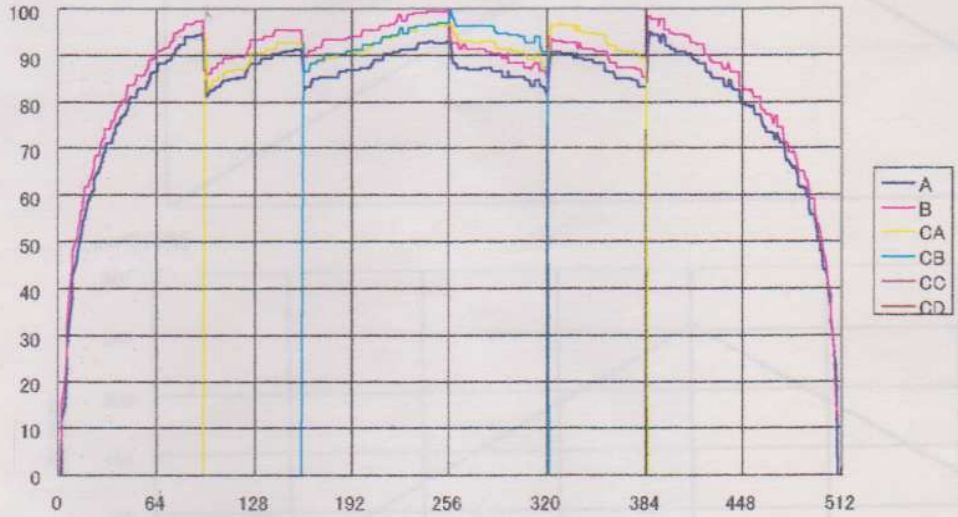
付録A

256bit×256bit乗算器 Wallace Tree のBasic Delayによるﾃﾞｨﾚｲ分布

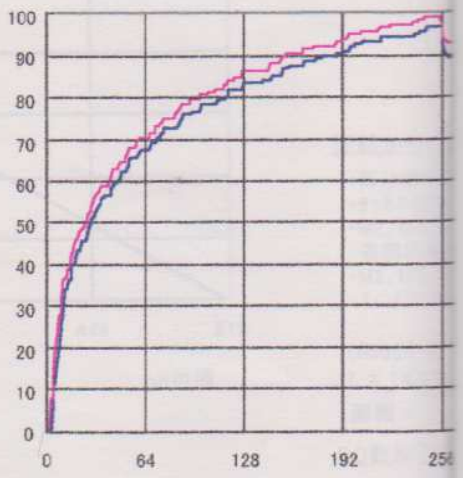
方式1



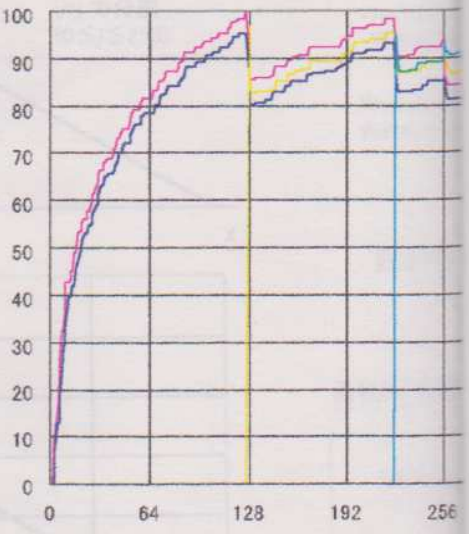
方式2



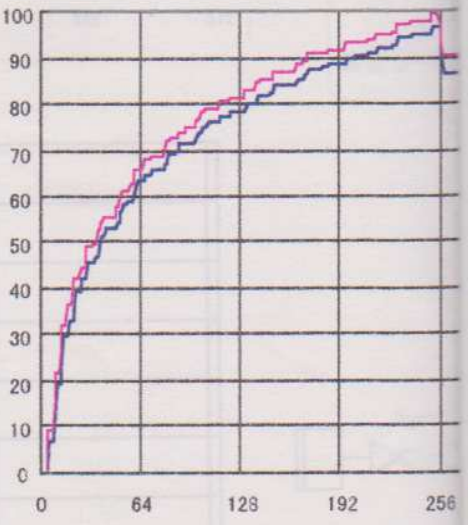
方式3

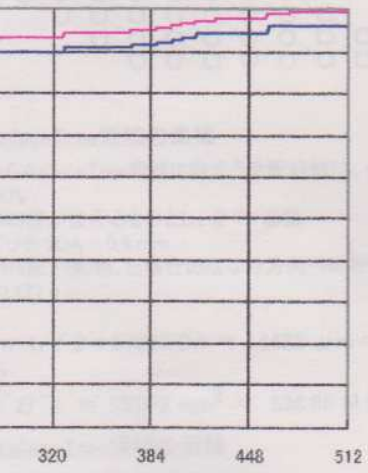


方式4

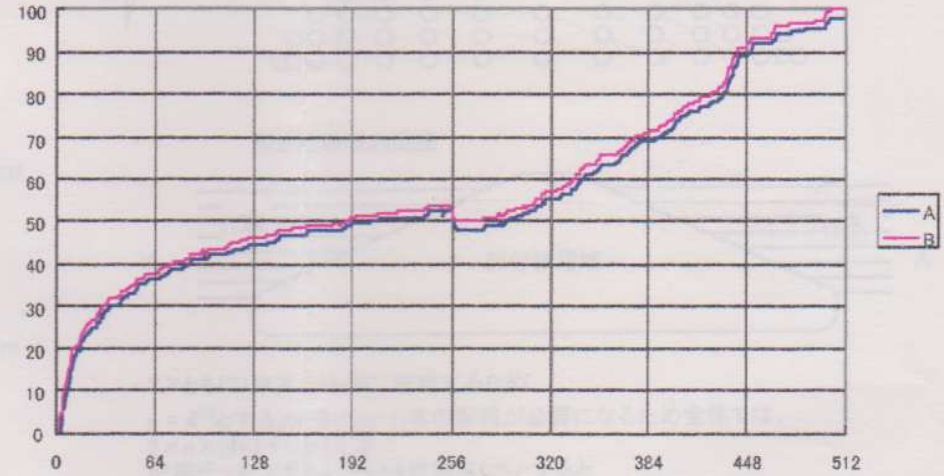


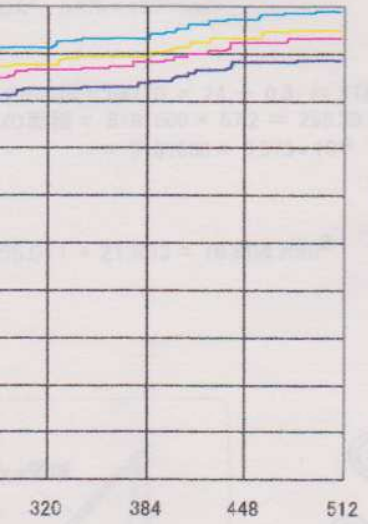
方式7



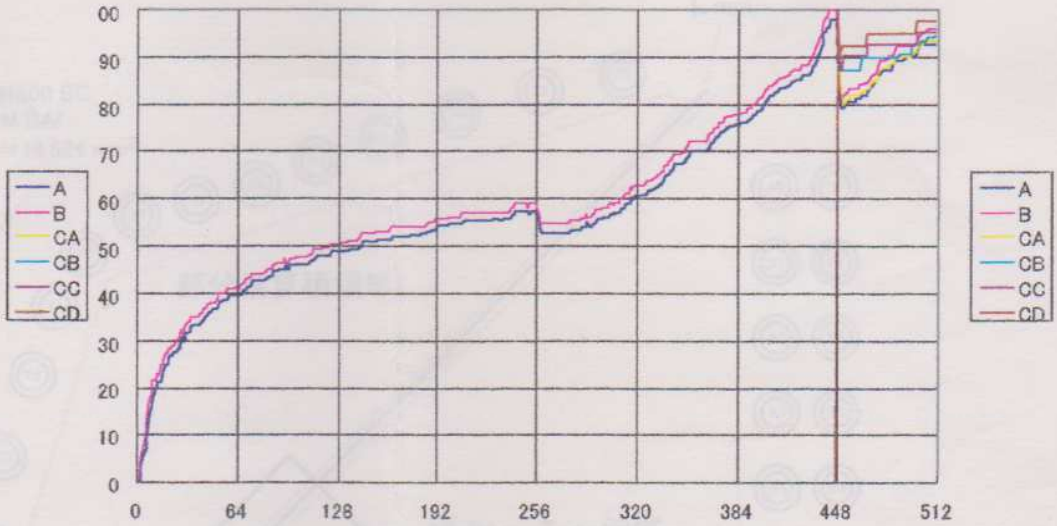

320
384
448
512
A
B

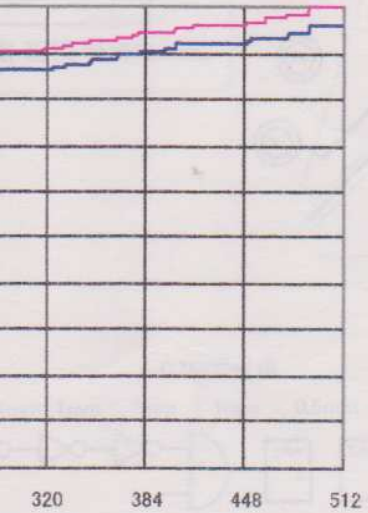
方式5


100
90
80
70
60
50
40
30
20
10
0
0
64
128
192
256
320
384
448
512
A
B


320
384
448
512
A
B
CA
CB
CC
CD

方式6

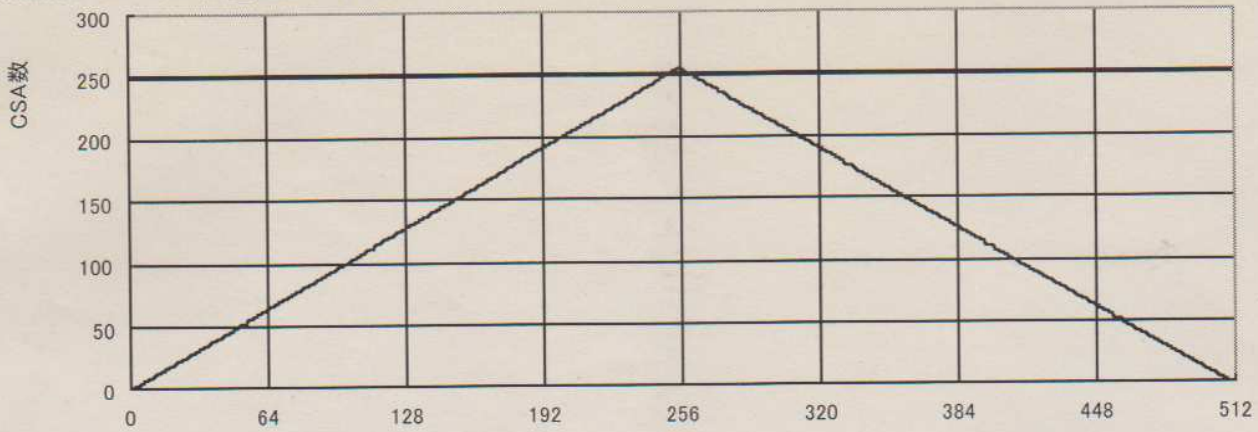

00
90
80
70
60
50
40
30
20
10
0
0
64
128
192
256
320
384
448
512
A
B
CA
CB
CC
CD


320
384
448
512
A
B

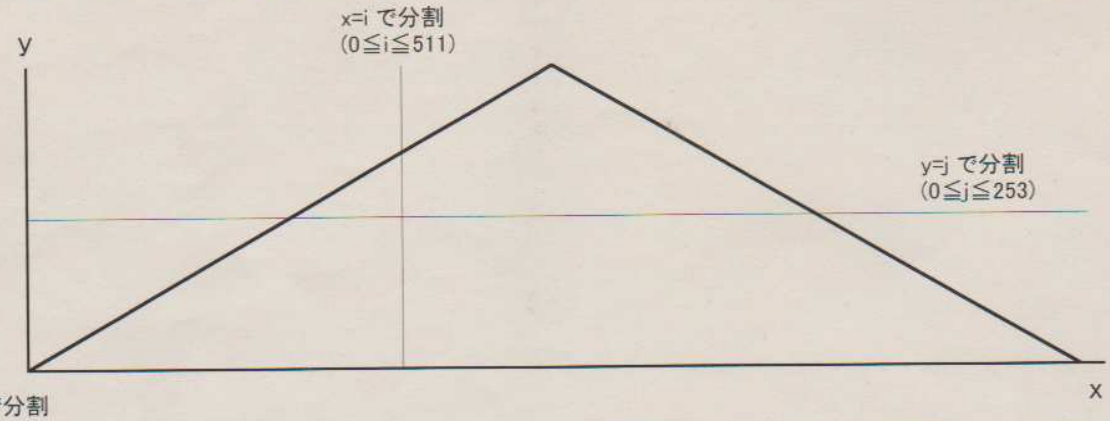
付録B

方式1

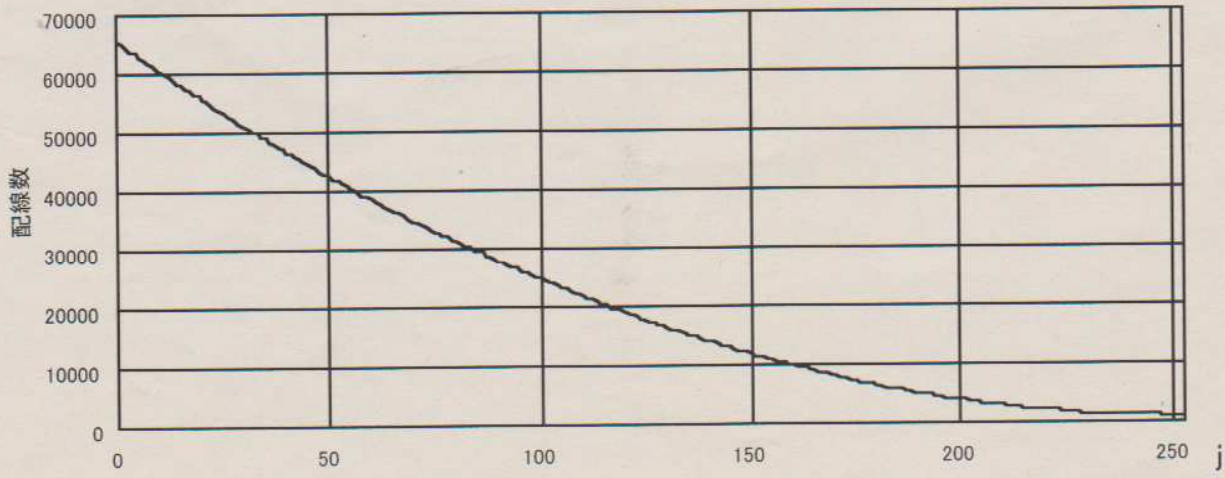
Wallace Treeのbit位置によるCSA分布

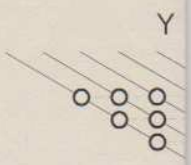


仮に下記のような領域に配置をして配線数を調べる。



x=iで分割





配線からくる

・部分積領域
・チャネル使用
・M2、M3、M
各層の配線
・M2、M3、M
1/ √2 DA

(65536÷0.
L = 14.842(

面積 = (L

BC数からく

・Wallace Tr
・CSA 1つ当
S029 + G
・1BCは、57
・実装率 30

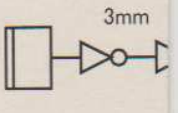
Wallace Tr
Wallace Tr

総面積

面積対策案

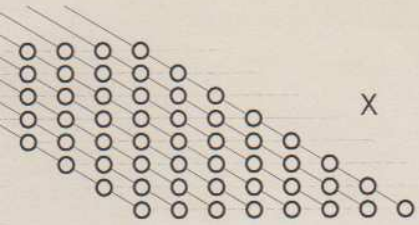
Wallace

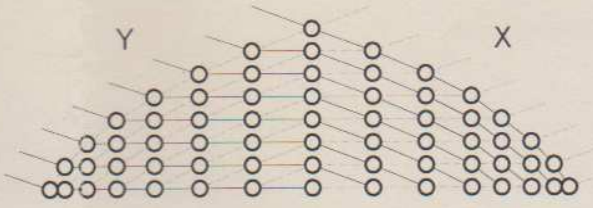
改善後の予



G982 G751 C
F.O. 4 F.O. 1 F

S側優先でWalla





allaceTree領域の面積

らWallaceTree領域に向かう全配線数は、n × n = 65536

80%

M5層が使えるものとし、全て1倍幅

ピッチ 2DA = 0.5 μm

M5層を使用した場合のななめ方向への配線ピッチ

0.177 μm

× 1/√2 = 57926 DA ≒ 14482 μm = 14.482mm

m

√2) ^2 ≒ 55.072 mm^2 ≒ 838.86 M DA^2

WallaceTree領域の面積

領域中にあるCSAの数 64770

たりのBC数

2 × 3 + S003 = 12 + 2 × 3 + 6 = 24 BC

DA^2 = 3.575 × 10^{-6} mm^2

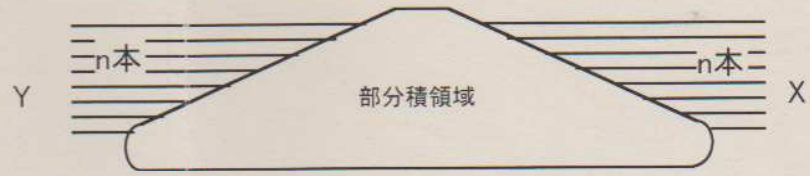
eのBC数 = 64770 × 24 ÷ 0.3 ≒ 5181600 BC

eの面積 = 5181600 × 57.2 ＝ 296.39 M DA^2

= 5181600 × 3.575 × 10^{-6} ≒18.524 mm^2

55.071 + 21.833 = 76.904 mm^2

部分積領域の面積



X,Yともに1本あたりn個に配線するため

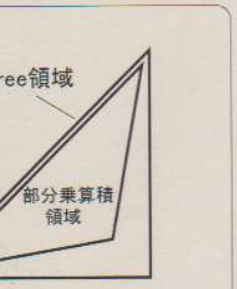
n = 4^{m}とすると4+3×(m−1)本の配線が必要になるため全体では、

2 × n × (4+3×(m−1)) 本

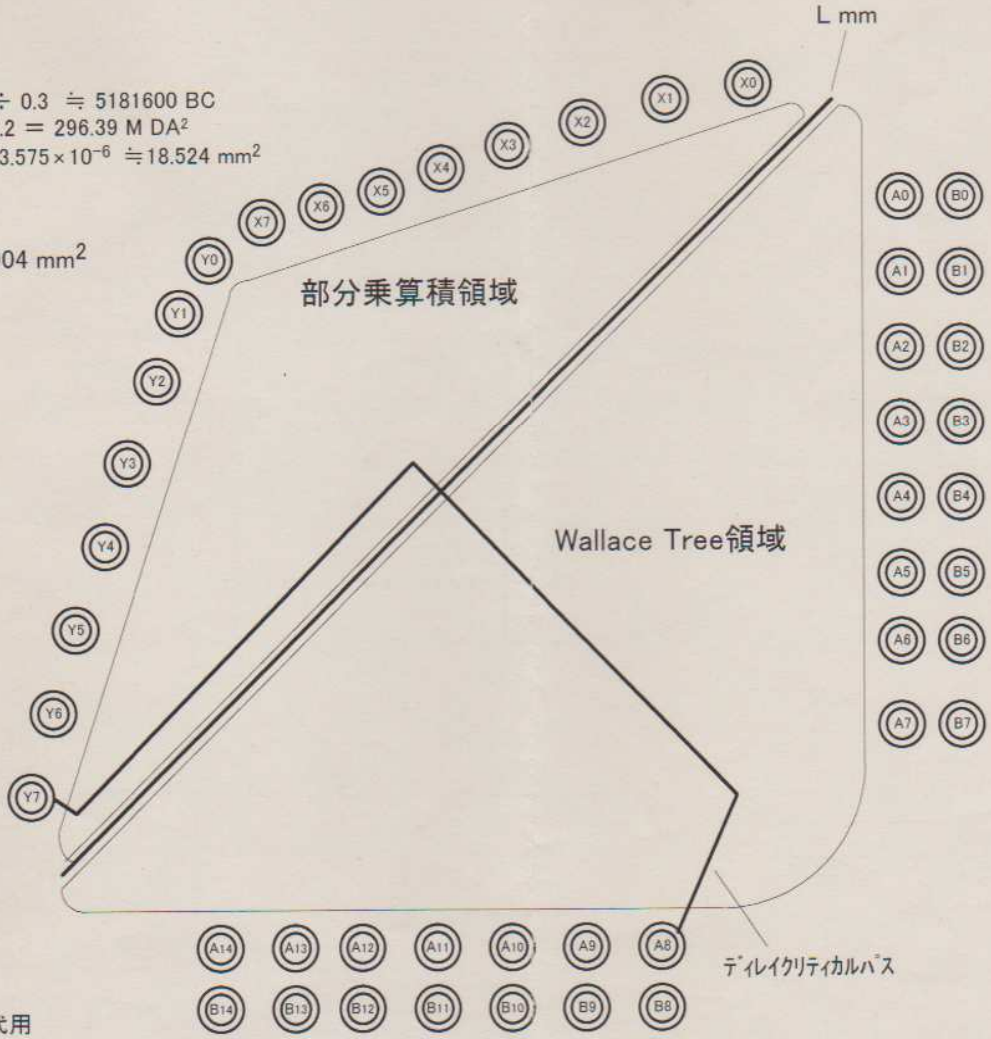
配線ピッチ √2 DA、ﾁｬﾈﾙ使用率80%とすると

2 × n × (4+3×(m−1)) ÷ 0.8 / √2 DA = 5884 DA = 1.471mm

面積 = L × 2.080 = 14.842 × 1.471 = 21.833 mm^2



側 面積 55.072 mm2



6558 ps

Treeを生成するとCSAの段数が1段減るがﾃﾞｨﾚｲではほとんど差がない。

配線をすべて0mmとした場合 4545 ps