5E-6

数生成アルゴリズム Mersenne Twister の実装

黒川恭一　　藤本繁伸　　野毛寛之
防衛大学校情報工学教室

1　はじめに

Mersenne Twister （以下MT）は、'96年に文献[1]にて発表された疑似乱数生成アルゴリズムである。その性能には、従来の乱数生成アルゴリズムにない長周期、高次元均等分布を持つという特徴を備えており、また、生成速度やメモリ効率においても優秀であることが認められている。[1] [2] また、MTは、k-distributionやspectral test等の統計的検定に対して十分満足の行く、良好な結果を収めている。現在、C言語など多くの言語によりプログラムが実装されており、いろいろな方面において利用されている。

一方、ＣＰＬＤやＦＰＧＡ等の再構成可能素子の発達により、対象に応じ、再構成することで最適な環境を提供するシステムが登場している。こうした技術を利用することで、パラメータの変更に対応可能なﾊｰﾄﾞｳｪｱも提案されている。[3]

本研究では、このMTによる乱数生成器を再構成可能素子やメモリを用いて作成した。同様に作成したM系列乱数発生器及びソフトウェアによる生成との比較結果も示す。

2　MTについて

2.1　MTの概要

wを計算機のワード長とし、$r(0 \leqq r < w)$, n, mを自然数定数とするとMTは、次のような漸化式を用いてｋ＝0,1, ‥‥として生成される。

$$X_{k+n} := X_{k+m} + (X^u_k \mid X^l_{k+1})A \quad (1)$$

さらに右から調律行列Tをかけて乱数列とする。MTとしては、そのパラメータの違いにより、MT11213A、MT11213B、MT19937、TT800などが提案されている。本研究では、MT19937をモデルとしてシステムを作成した。この乱数列は$2^{19937}-1$の長周期と,623 次元の高次元均等分布を持つ。[1][2]表１にそのパラメータをまとめて示す。

Table1　MT19937のパラメータ

各パラメータの値
$(w,n,m,r)=(32,624,397,31)$, u＝11
a＝9908B0DF, s＝7,t＝15,l＝18
b＝9D2C5680, c＝EFC60000

2.2　MT19937の流れ

MT19937による乱数生成の流れをFig.1に示す。まず、前処理としてMT19937の初期擬似乱数seedとなる624個の32ビットの整数乱数をX[0]～X[623]に格納する。この乱数は、線形合同法[4]によって生成する。次に、iを0とした後、(1)式の$(X^u_k \mid X^l_{k+1})$としてX[i]の上位１ビットとX[i+1]の下位31ビットを合成してyとする。yのLSBが0のときXOR (y>>1)を、さらに加えて、1のとき、行ベクトルaを乗じ、その結果を新たにX[i]として格納する。最後にSTEP4として調律行列Tを乗じ、最終的な乱数yを出力する。その後、iを更新してstep2へ戻る。

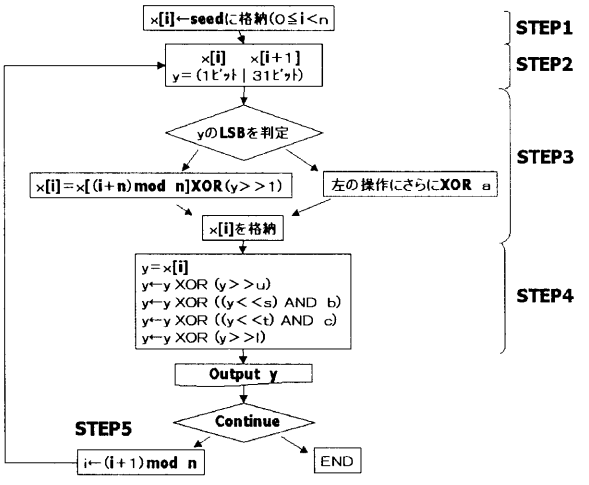

Fig.1　MT19937の生成法

3　システムの実装

MTはそのパラメータを換えることで、異なる乱数系列を生成することができる。これに対応する、本システムでは再構成可能素子を用いることによってMTのパラメータを換えることを可能としている。また、MTはその計算にビット演算しか用いないため、回路構成もそれほど膨大にはならない。さらに、メモリアクセスをパイプライン処理化可能であり、乱数生成効率の向上が期待できる。

3.1　システムの全体構成

本研究で使用した再構成可能素子は、Xilinx社のCPLD(XC95108)である。マクロセル数およびレジスタ数108、使用できるI/O数69、ゲート数は2400用意され、その特徴は高集積、高性能、および最大10000回の書き換えが可能であるFastFLASHセルの採用にある。これは、先に述べたように、MTの回路を書き

換えることを可能にすることで、異なる乱数系列を生成させることができるようにするためである。またシステムを動作させるホストコンピュータにはPC98を用いた。

本システムは、XC95108のCPLDを4個、メモリHM6116P−3を4個使用している。Fig.2にシステム全体のブロック図を示し、以下に各部の説明を行う。

まず、乱数生成の前段階として624個の乗算合同法で作成された32ビットの初期擬似乱数seedをMemoryに格納する。ホストシステムに使用したコンピュータであるPC98は、I/Oバスのバンド幅が8ビットしかないため、それに合わせる必要がある。これらシステムの制御信号は,すべてI/Oアクセス信号をデコーダドすることによって作成した。

MT#1はX[i]、X[i+1]、X[i+397]に対応するｱﾄﾞﾚｽを保持する3つの10ﾋﾞｯﾄｶｳﾝﾀ、ホストコンピュータからのI/Oアクセス信号を処理するDecoder、モードﾚｼﾞｽﾀから成つ。MT#2では、STEP2とSTEP3を実行する。MT#3ではSTEP4をビットごとのXOR演算で実現する。MT#4では、生成された擬似乱数を8ビットづつ、4回のI／Oアクセスでホストコンピュータへ返す。

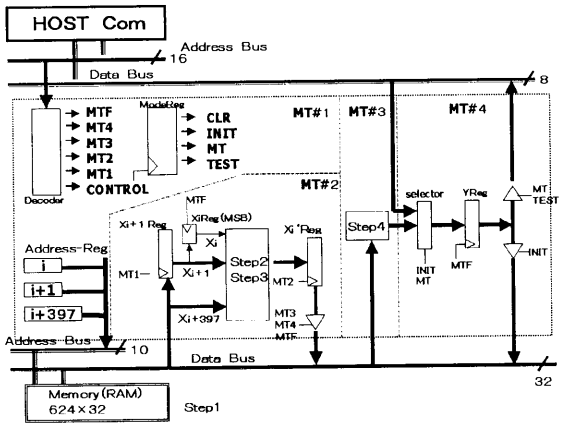

Fig.2　システム全体のブロック図

4　システムの検討

MT19937をC言語を用い実現したプログラムを発生させた場合と、本システムとを、乱数生成速度の面において比較検討した。その結果、同じコンピュータを用いた場合、本システムがおよそ23.5倍の生成速度を持つことがわかった。

また比較対象として、32ビットの整数乱数を発生させるM系列乱数生成器を開発した。両者の比較結果を表2にて示す。これと比較すると、MTに初期擬似乱数の設定が必要なことを除けば、実効命令数が同じこともあり、乱数を生成する速度は変わらない。M系列の周期が$2^{32}-1$であることと比べれば、MTの方が効率的であるといえる。しかし、MTでは4チップのCPLDを使用し、使用マクロセル数も約5倍であり、どうしても、回路規模が大きくなってしまった。

Table2　M系列との比較

	M系列	MT19937
生成速度	10.8μsec	10.8μsec
周期	$2^{32}-1$	$2^{19937}-1$
CPLD数	1	4
ﾏｸﾛｾﾙ数	68	335
ｸﾛｯｸ上限	83.3MHz	44.4MHz

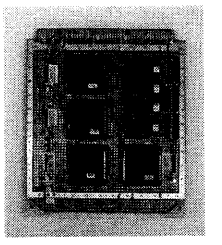

Fig.3 開発した写真

5　まとめ

本研究は、ソフトウェアによって行っている特定の処理を、CPLDなどを用いてハードウェアにおきかえることで、コンピュータ本体の負担を軽減し、且つ、より効率的なシステムを構築することを目的としている。その一例として、新しい乱数生成アルゴリズムMTを実装した。これによって、ソフトウェアとハードウェアによる協調処理により、効率的な処理を行うことができる。なお、今後の更なる効率化のための改善点としては、32ビットI/Oデータバスを持ったコンピュータへの本システムの実装、それに伴うコンピュータ本体からの実行命令の削減、CPLD内の回路の簡単化等が挙げられる。

参考文献

[1] M.Matumoto T.Nisimura : "Mersenne Twister: A 623-Dimentionary Equidistributed Uniform Pseudo-Random Number Generator", ACM Transactions on Modeling and Computer Simulation, pp1～pp30(Jan. 1998).

[2] 松本 眞："コイン投げで一儲けする方法" 情報処理学会誌, Vol.39, No.11, pp1166-1170 (1998-8).

[3] 末吉敏則："ﾘｺﾝﾌｨｷﾞｬﾗﾌﾞﾙ・ｺﾝﾋﾟｭｰﾃｨﾝｸﾞ" IPSJ Magazine Vol.40 No.8 Aug.1999,pp777～pp782

[4] Knuth, D.E.: "The Art of Computer Programming,Vol.2: Seminumerical Algorithms 2nd ed., " Addison-Wesley, Reading, Mass. (1981)