

FPGAを用いたFDTD法による電磁界計算の高速化の検討*

Study on high speed calculation of electromagnetic simulation by FDTD method using FPGA

高橋 丈博 Takehiro TAKAHASHI**

Abstract

To understand the change of electromagnetic effect quickly due to parameter change, high speed calculation of FDTD method is studied. In this study the FPGA (field programmable gate array) which is a programmable hardware logic circuit is employed to calculate iterated calculation part of FDTD method. From the comparison of calculation speed between CPU and FPGA using 1st order FDTD, same performance is investigated by hardwareization of whole iteration part of FDTD calculation.

Keywords: electromagnetic simulation, numerical calculation, FDTD method, FPGA, high speed calculation

1. まえがき

電子機器の高速化、高機能化に伴い、機器内部の電気信号の高周波化が進み、電磁ノイズなどによる信号品質の低下や、放射ノイズの増加などが生じている。また、電磁的な現象も複雑になり、問題の解決に時間がかかっている。

電磁的な現象の解決方法の1つとして、計算機による電磁界シミュレーションが活用されている。電磁界シミュレーションを使って、電気信号の伝送の様子、周囲への干渉の様子、放射の様子などを計算することで、波形の歪や電磁干渉の程度などを見積もったり、波形を可視化したりすることができ、波形歪の改善や電磁干渉の低減などに役立つことができる。また、電磁界シミュレーションの結果を可視化することは、電磁界の現象を直観的に理解することにつながるため、教育的な効果も高いと考えられる。

ここで、電磁界シミュレーションを使って電子機器の物理的な構造と電磁的な振る舞いとの関連を把握する場合、配線や筐体形状などのパラメータを変化させて、電磁界の変化を調べることが多いが、計算に時間がかかると、パラメータの変化と現象の変化との関連を直観的につかみにくくなり、ノイズ低減の対策も進みにくくなるのではないかと思う。

そこで、今回電磁界シミュレーションの高速化を試すことにした。計算の高速化には様々な方法が考えられる。スーパーコンピュータや並列計算で、CPUによる計算を早くする方法、DSPやGPUなどの高速計算ボードを利用する方法などが考えられる。青木ら¹⁾はGPUを用いて3次元FDTD法を実装し、さらに高速表示を実現するためのモデルを提案し実現している。翁ら²⁾はFPGAに熱拡散シミュレーションやFDTDシミュレーションを実装し、メモリアクセス評価をし、データストリームの量を増やし本数を減らすことで計算性能の向上が見込めることを示している。さらに、平井ら³⁾は大規模科学計算のためのマルチFPGAシステムにおけるモジュール配置の最適化をSAで行うなどの試みを行っている。これらの知見はとても有益であるが、計算性能や効率向上が目的で、計算結果の利用を見据えたものではない。本研究では、基本的に計算結果を設計に反映させるために有効なシステムを目指すことにする。

今回は、FPGAを利用した計算の高速化を試した。本来は、いろいろな方法を使って比較すべきであるが、現在は、FPGAによる計算高速化の可能性を調べることや、進化の著しいFPGAの計算法を確かめるなどの目的もある。また、結果を表示する場合、ハードウェアから直接画像で結果を表示することができれば、計算から表示までのトータルの速度を高速化できる可能性もあり、今回はFPGAによる計算高速化を試した。

2. FDTD法の概要 [4-5]

電磁界の計算機シミュレーション手法は様々あるが、基本となるのは電磁界の振舞いを表すマクスウェル方程式で、積分領域の方程式を基にした周波数領域の解法と微分形式の方程式を基にした時間領域の解法との2つに大きく分けられる。前者に代表されるのがモーメント法で、計算は対象物を分割したセグメント同士となるので、ある程度範囲が限定されるが、プログラムには複雑な数値計算が含まれる。一方、後者に代表されるのがFDTD法で、後に述べるような単純な差分計算を解析空間内で繰り返す。単純計算を繰り返す方法はハードウェアにより高速化が期待できるため、本研究ではFDTD法を利用することにする。

FDTD法は、解析空間をYee格子と呼ばれる格子状に区切り、交点に離散的な電界あるいは磁界を配置する。空間の電磁界の伝搬の様子は次のマクスウェル方程式の回転の式によって計算できる。

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad (1)$$

$$\nabla \times \mathbf{H} = \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t} \quad (2)$$

電界と磁界の格子を半セルずらすことによって、電界の回転の中心に磁界、磁界の回転の中心に電界を配置することができる。

3次元モデルにおけるYee格子の一部を図1に示す。図1(a)の格子は、格子の辺の中心に、辺の方向の電界成分を持っている。例えば、下奥の四角で囲んだ E_x 成分を計算するには、半セルずらした磁界の格子(図中の四角)の辺の中心に配置されている磁界成分 H_y, H_z を使い、次の式で計算する。

* 原稿受付 2020年10月27日

** 拓殖大学工学部

$$E_x^n \left(i + \frac{1}{2}, j, k \right) = \alpha_e E_x^{n-1} \left(i + \frac{1}{2}, j, k \right) + \beta_e \left[\frac{\Delta H_z^{n-\frac{1}{2}}}{\Delta y} - \frac{\Delta H_y^{n-\frac{1}{2}}}{\Delta z} \right] \quad (3)$$

但し

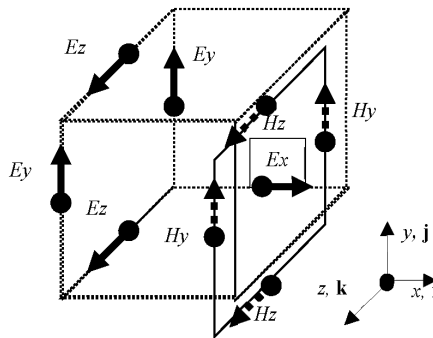
$$\Delta H_z^{n-\frac{1}{2}} = \Delta H_z^{n-\frac{1}{2}} \left(i + \frac{1}{2}, j + \frac{1}{2}, k \right) - \Delta H_z^{n-\frac{1}{2}} \left(i + \frac{1}{2}, j - \frac{1}{2}, k \right) \quad (4)$$

$$\Delta H_y^{n-\frac{1}{2}} = \Delta H_y^{n-\frac{1}{2}} \left(i + \frac{1}{2}, j, k + \frac{1}{2} \right) - \Delta H_y^{n-\frac{1}{2}} \left(i + \frac{1}{2}, j, k - \frac{1}{2} \right) \quad (5)$$

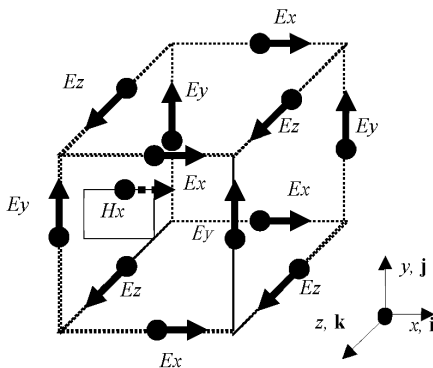
ここで、nは時刻で1つ前の時刻のExに、1/2前の時刻のHy, Hzの回転を差分を使って計算し、加えている。αe, βeは係数、Δy, Δzは格子の大きさを表す。Ey, Ezも同様に計算する。

同じく、図1 (b)の格子では左側の面の中央にある四角で囲んだHxを、それを囲むEy, Ezの回転から計算するための配置を表したものである。計算式は次の通り。

$$H_x^{n+\frac{1}{2}} \left(i, j + \frac{1}{2}, k + \frac{1}{2} \right) = \alpha_m H_x^{n-\frac{1}{2}} \left(i, j + \frac{1}{2}, k + \frac{1}{2} \right) - \beta_m \left[\frac{\Delta E_z^n}{\Delta y} - \frac{\Delta E_y^n}{\Delta z} \right] \quad (6)$$



(a) 磁界Hy, Hz成分による電界Exの計算



(b) 電界Ey, Ez成分による磁界Hxの計算

図1 Yee格子と電界, 磁界成分の位置

但し

$$\Delta E_z^n = \Delta E_z^n \left(i, j + 1, k + \frac{1}{2} \right) - \Delta E_z^n \left(i, j, k + \frac{1}{2} \right) \quad (7)$$

$$\Delta E_y^n = \Delta E_y^n \left(i, j + \frac{1}{2}, k + 1 \right) - \Delta E_y^n \left(i, j + \frac{1}{2}, k \right) \quad (8)$$

同様に、Hy, Hzも計算できる。

このように、時刻を1/2ずつ進めながら、電界E, 磁界Hをそれぞれ計算してゆく。

3. FPGAについて

今回は、Xilinx社のFPGAであるZynq-7000を搭載した、Digilent社のZyboボードを使って検討を行った^{6,7)}。FPGAでは、以前よりFPGA上にCPUを構成し、さらに周辺回路を構成して接続することにより、CPUで動かすソフトウェアと構成するハードウェアとで協調させて動作させ、バランスのとれたシステムをつくることのできる点が特徴の1つだった。Zynqはそれをさらに進め、最初からCPUであるARM Cortex-A9を2個搭載しており、関連するコントローラやペリフェラルなどもあらかじめ用意されている。この部分はPS (Processing System) 部と呼ばれている。一方、自由に回路を構成できるPL (Programmable Logic) 部もあり、PS部とPL部は汎用のAXIポートによって接続することができる。

本ボードを利用すると、CPU上でプログラムだけを動かすこともできるし、プログラムの一部を回路で表現し、CPUのプログラムから関数のように計算回路に数値を渡し、計算結果を受け取るという処理ができる。そこで、回路化する部分を変えながら、どの程度の高速化が見込めるかを試すことができる。

問題は計算の回路化である。まず、回路による計算では、浮動小数点計算が苦手であるため、浮動小数点計算は避けて計算を行いたい。そこで、計算の桁を操作し、浮動小数点計算をしないで、FDTD法の計算をする工夫を行った。

次の問題として、計算を回路化する方法である。0から作成するのは困難であると考え、Xilinx社が提供する、高位合成ツールを利用することにより、Cプログラムで作成されている計算を回路に自動変換することにした。このツールを使うと、変換された回路はIPと呼ばれる1つの部品として登録され、CPUに簡単に接続することができ、プログラムからの簡単に呼び出すことができる。高位合成にはXilinx社のVivado HLS 2018.3. FPGAの全体の構成や動作は同じくVivado 2018.3を使って行った。

4. 1次元FDTD法による検討

4.1 プログラムの修正

文献⁵⁾に掲載されている1次元FDTDのサンプルプログラムを使い、FPGAで実行させた。

プログラムのフローを図2に示す。このフローは基本的に計算する次元数には依存しない。まず、初期設定で空間配列の確保とパラメータの準備を行う。メインルーチンでは、空間内の電界の更新を行い、時間を半ステップ進める。境界の処理を行い、磁界の更新を行って、時間を半ステップ進める。これを終了時間になるまで続ける。

文献⁹⁾にある解析モデルを図3に示す。長さ2mの1次元の空間を1000分割し、中心付近の500~550に比誘電率 $\epsilon_r=3$ の誘電体を配置し、その他は自由空間として設定している。100の位置に幅20のガウシアンパルスを加し、その時間変化を計算している。境界処理にはMurの1次元吸収境界を用いている。

サンプルプログラムはFortranで記述されていたため、まずC言語に書き換え、同様の結果が得られることを確かめた。これらのプログラムは浮動小数点演算で計算されている。これを、少なくともメインルーチンにおいて整数計算にするため、次の処理を行った。

- ・メインルーチンで使用する変数をint型にした
- ・ガウシアンパルス信号は電界の振幅が1V/mのため、境界の振幅も考慮し 10^6 倍とした

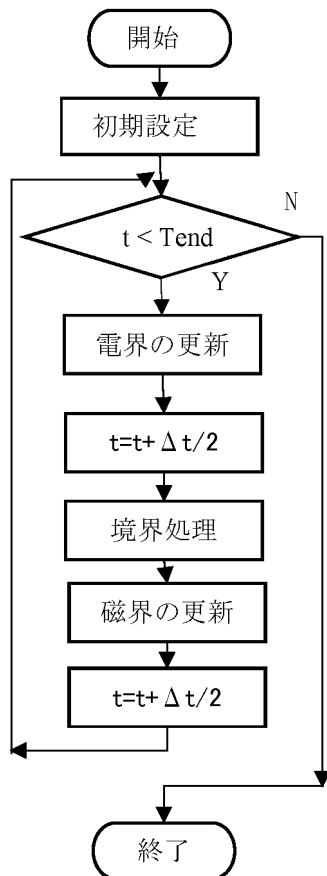


図2 FDTD法プログラムのフロー

- ・係数で1以下や、1以下の数が必要なものは、あらかじめ大きくしておき、係数をかけたところで、全体を小さくする処理をした

まだ十分ではないが、このような処理を行うことにより、整数計算でも、ある程度計算精度を保って計算できると考えられる。

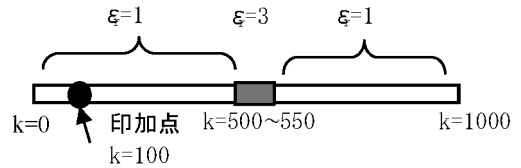


図3 文献⁹⁾の1次元解析モデル

4.2 プログラムの実行

プログラムの動作と、FPGAの動作を確認するため、ZYBOボードでCPUだけを使い、プログラムを実行した。FPGAを設計するツールVivadoを使い、CPUだけを配置して回路を生成すると、CPUだけが動作する回路が作られる。Vivado付属のSDKツールを起動し、CPUでプログラムを実行する。

C言語で書き直したサンプルプログラムと、それを整数計算処理したものについて、時刻100と700における、電界の分布を図4、5に示す。図5の整数計算の方は、印加信号の大きさを 10^6 倍しているため、大きな値となっているが、相対

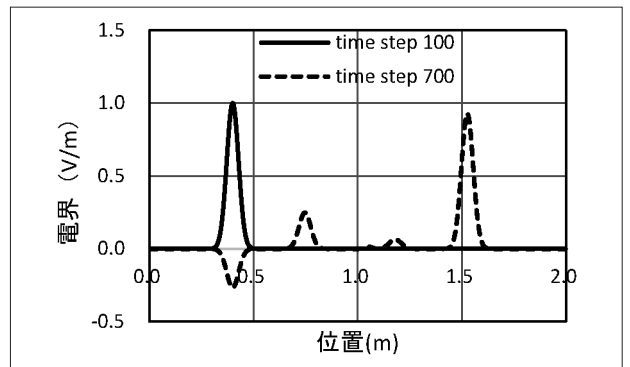


図4 浮動小数点計算によるサンプルプログラムの動作結果

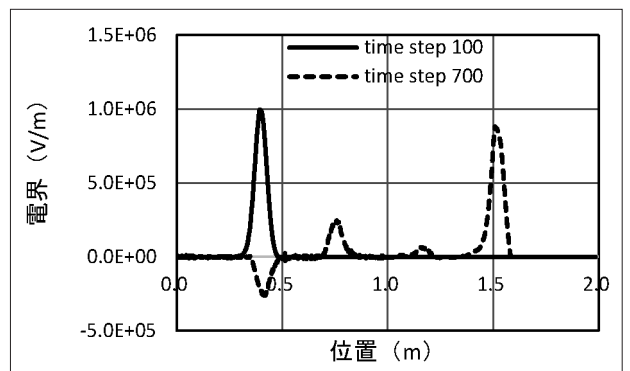


図5 整数計算によるサンプルプログラムの動作結果

的には両者ほぼ一致している。

文字の表示に時間がかかるため、printf文を除いて、両者の動作時間をプログラムに埋め込んだXTime()関数で計測したところ浮動小数点計算が295ms、整数計算が273msでほとんど差がなかった。

4.3 計算の回路化

図2の計算フローにおいて、電界や磁界の更新は、単純な加減算と乗算によるものである。例えば、電界の更新計算関数e_cal()は図6のようなプログラムである。ここで、exは電界のx成分の配列、hyは磁界のy成分の配列、ae、beは係数である。配列や係数は全て外部変数のため、引数や返り値はない。この計算を、高位合成によって回路化する⁷⁾。

高位合成ツールVivado HLSでは、プログラムの中の間数として記述した指定部分を回路化し、IP (Intellectual Property) と呼ばれる部品として使えるようになる。IP化するとPS部と内部のバスを通して接続でき、変数のやり取りも便利になる。

今回は、まずe_calの中の、exを計算する1行分を高位合成し、図6のfor文から呼び出すことで計算の動作を確かめた。CPUの動作するPS部とIPとの接続は、AXI-Liteバスで接続した。IPは次の手順で呼び出し、実行する。

- 1) あらかじめ、初期化をする。
- 2) 引数をセットする。
- 3) IPをスタートさせる。
- 4) 終了を待つ
- 5) 返り値を読みだす

必要な関数は全てVivadoによって用意される。

実行した結果、図5と同じ波形が出力され、正しく動作していることを確認した。動作時間は3737msとなり、CPUだけで計算した場合(273ms)に比べ、14倍の時間がかかってしまった。これは、引数のセットや読み出しに時間がかかってしまっていることが考えられる。

```
void e_cal() {
    for (int k=1; k<NZ-1; k++) {
        ex[k] = ae[k]*ex[k]
              - be[k]*(hy[k] - hy[k-1]);
    }
}
```

図6 電界更新関数e_cal()

次に、図2の電界の更新から磁界の更新及び時間を半ステップ進める部分、すなわち、ある時刻における線路上の全ての電界・磁界の更新を行う部分を回路化して比較を行った。線路は1000分割されているので、1000回のループが2回とな

る。回路化にあたり、そのまま回路を生成する場合と、ループをPIPELINE化する場合と、ループを全て個別の回路に展開する場合について比較した。このうち、ループを全て展開する場合は、素子数が不足し、回路構成ができなかった。動作を確かめられた前者2つの場合について動作時間を比較したところ、そのままの場合が2403ms、PIPELINE化した場合が224msとなり、ほぼ同程度となった。1回の計算を行う回路に比べ、1/3の時間となったが、まだCPUだけの計算時間の10倍である。

最後に、上記の計算を2000ステップ繰り返し計算する部分をそのまま回路化した。その結果、計算時間は241msとなり、CPUだけの計算と同等の計算時間となった。この回路を、素子が不足しない程度に並列化できれば、さらに計算時間を短くできると考えられる。

また、データのやり取りに外部メモリを使ったり、表示をハードウェアによって直接行うなどにより、高速な計算と表示を実現できると考えられる。

5. まとめ

本研究では、電磁界解析法の1つであるFDTD法のプログラムの高速化を検討するため、FPGAを使ったFDTDプログラムの計算のハードウェア化を行った。FDTDプログラムを整数で計算できるようにし、計算を行うプログラムをハードウェア化して、正しく動作することを確認した。また、繰り返し部分全体をハードウェア化することによって、CPUと同程度の計算速度を得ることができた。

今後は、変数をメモリに格納し、データ転送をなくすことや、2次元、3次元FDTDの高速計算、結果を表示する回路の作成などを行ってゆく。

参考文献

- 1) 青木他：“GPUを用いた3次元FDTD法による電磁界シミュレーションの高速な可視化処理の実装”，信学技報，CPSY2013-05，pp. 35-40 (2013)
- 2) 翁他：“高位合成ベースFPGAアクセラレータにおけるステンシル計算用メモリプロファイリングフレームワーク”，信学技報，RECONF2014-11，pp. 55-60 (2014)
- 3) 平井他：“マルチFPGAシステムにおける演算モジュールの配置手法の検討”，信学技報，VLD2013-118 (2014)
- 4) 山下：電磁波問題解析の実際，電子情報通信学会 (1993)
- 5) 宇野，何，有馬：数値電磁界解析のためのFDTD法—基礎と実践—，コロナ社 (2016)
- 6) 岩田，横溝：FPGAパソコンZYBOで作るLinux I/Oミニコンピュータ，CQ出版社 (2016)
- 7) 小林：FPGAプログラミング大全，秀和システム (2016)