

FDL によるファジィ推論から System Verilog を生成する トランスレータ (FDLtoSV) の試作

紙名 佑一[†] 岩井 啓輔^{††}
† 防衛大学校 理工学研究科

松原 隆^{††} 黒川 恭一^{††}
†† 防衛大学校 情報工学科

1. はじめに

近年、組み込みシステムにおいて、高速化や低消費電力化を目的として、ASIC や FPGA によるハードウェア実装が盛んに行われている。その際、ハードウェア設計には多くの場合、ハードウェア記述言語(HDL)が使用されている。

一方、ファジィシステムにおいては、1996 年にファジィシステム記述言語(FDL)の言語仕様が策定され[1]、同時にその処理系である fdltoc 及び FDL ライブラリが開発された。その後も OS 依存性の解消や LUT 生成等の研究が行われ[2]、FDL で記述されたファジィ推論を C 言語に変換し、ソフトウェア実装する手法が確立されている。しかしながら、ハードウェア実装に関する開発環境は存在しないことから、ファジィ推論のハードウェア実装を行うためには HDL を自ら記述する必要があった。そこで、本研究では、FDL で記述されたファジィ推論から HDL の一種である System Verilog を生成するトランスレータ(FDLtoSV)を開発することにより、FDL によるハードウェア開発環境の構築を行い、ファジィ推論のハードウェア実装の容易化及び高性能化を目指す。

2. FDL の概要

FDL とは、C 言語をベースとした言語で、ファジィ推論に必要な型、演算子、ルールベース等を追加、拡張したものである。FDL は、大きく分けて 3 つの記述から成っている。1 つ目は、メンバーシップ関数を定義する記述(ファジィ型定義)。2 つ目は、入出力のファジィ型を指定してルールを列挙する記述(ルールベース)。3 つ目は、それら呼び出し使用するための記述(main 関数)である。main 関数内には、画面表示やファイルへの入出力等のソフトウェア的に必要な処理が C 言語で記述される。

3. FDLtoSV の実装

FDLtoSV を実装するにあたり、fdltoc とその FDL ライブラリは使用せず、FDL のファジィ型定義及びルールベースから System Verilog を生成するプログラムを新たに作成した。推論法については、FDL では各種推論法を指定できるのに対して、FDLtoSV では試作段階として、min-max-重心法のみ実装した。生成される System Verilog は、同期式順序回路として設計した。また、ファジィ推論における演算では、主に 0~1 の値をとるファジィ値を使用するため、浮動小数点方式をとるメリットが小さく、固定小数点方式を採用した。実装の流れを図 1、処理の流れを図 2 に示す。

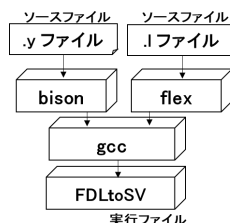


図 1. 実装の流れ

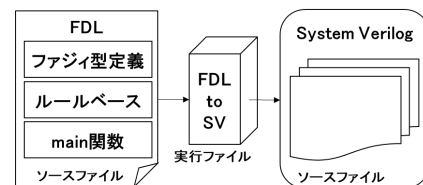


図 2. 処理の流れ

3. 性能評価

3.1 回路規模

FDL で記述した 2 入力 1 出力、分割数各 60 のファジィ推論について、小数点以下 7bit と設定した FDLtoSV で System Verilog を生成した。ターゲットデバイスを Xilinx Artix-7 FPGA(XC7A35T)として、Vivado 2019.1 を用いて論理合成を行った結果、表 1 に示す回路規模となった。

表 1. 回路規模

LUT	3,136
FF	581

3.2 速度・誤差比較

fdltoc で生成したソフトウェアを表 2 に示す実行環境で実行した場合と、3.1 で生成したハードウェアを 100MHz のクロックで動作させた場合の実行時間及び誤差の比較を行った。推論を 3600 回実行したときの実行時間は、ソフトウェアで 31.7ms に対して、ハードウェアは 3.1ms となり、約 10 倍高速であった。また、固定小数点方式を採用したことによる誤差は±4.1%以下であった。

表 2. 実行環境

OS	Windows10 pro 64bit
CPU	Intel Core i7-7700T(2.9GHz)
メモリ	8.0GB

4. まとめ

FDL から System Verilog を生成するトランスレータを試作し、ソフトウェア実装時と比べて高速であることを確認できた。FDLtoSV は試作段階であり、FDL の一部構文のみに対応しているため、FDL に準拠した形に拡張を行う。

参考文献

- [1] “ファジィシステム記述言語(FDL)標準仕様解説書,” 社団法人 日本電子機械 工業会(EIAJ)編, 1996.
- [2] 多和田 侑, 森 雄一郎, “FDL による組み込みソフトウェア開発ガイドライン,” Technical Reports on Information and Computer Science from Kochi, Vol.3, 2011.