■関数型なHLS

Hardware Synthesis from a Recursive Functional Languageと，
参考文献を読むなど.

参考文献部分のメモ

- Hardware Synthesis from a Recursive Functional Language
    - https://dl.acm.org/citation.cfm?id=2830850
- Clash[1,2]
    - Structual Descriptions of Synchronous Hardware Using Haskell
        - https://dl.acm.org/citation.cfm?id=1901964
    - including Danvy's defunctionalization[4], inspired by Reynolds[24]
        - Defunctionalization at work
        - https://dl.acm.org/citation.cfm?id=773202
        - higher-order programming languages[24]
        - https://dl.acm.org/citation.cfm?id=805852
- SHard compiler[26]
    - http://scheme2006.cs.uchicago.edu/05-saint-mleux.pdf
    - Schemeから高位合成．再帰もサポート．CPS変換とlambda lifting
    - データタイプなどに制限が多い
- FLaSH compiler[22]
    - Hardware Synthesis Using SAFL and Application to Processor Design
    - https://www.cl.cam.ac.uk/~am21/papers/charme01-invite.ps.gz
        - 関数を1インスタンスとして実装，複数からの呼び出しをハンドリング
        - 再帰はサポートしてない
- Bluespec[14]
    - Synthesis of operation-centric hardware description
    - https://dl.acm.org/citation.cfm?id=603017
- Sheeran mufp (μ FP)[26],[27]
    - Hardware Design and Functional Programming: a Perfect Match
    - http://www.jucs.org/jucs_11_7/hardware_design_and_functional/jucs_11_7_1135_1158_sheeran.pdf
    - muFP, a language for VLSI design
    - https://dl.acm.org/citation.cfm?id=802026
    - ハードウェアストラクチャをfunctionalなアプローチで記述
    - Lavaへ
- Lava: hardware design in Haskell[3], Introducing Kansas Lava[10]
    - Lava - https://dl.acm.org/citation.cfm?id=289440
    - Kansas Lava - http://www.ittc.ku.edu/~andygill/papers/draft-kansas-lava-ifl09.pdf
    - circuit strctureの記述にフォーカス
- HML, a novel hardware description language and its translation to VHDL[18]
    - http://ieeexplore.ieee.org/document/820756/
    - Lavaとは違う，cicrcuit structureをfunctionalに記述するアプローチ
    - HML, an inovative hardware description language and its translation to VHDLというのもある
- Ghica et al, Geometry of synthesis iv: compiling affine recursion into static hardware[9]
    - https://dl.acm.org/citation.cfm?id=2034805
    - 再帰アルゴリズムをハードウェアに実装
    - Algolライクな言語
    - 再帰で使う変数を小さなメモリに格納
- Middendorf et al, Hardware synthesis of recursive functions through partial stream rewriting[21]
    - https://dl.acm.org/citation.cfm?id=2228583
    - C/C++がターゲット
- Stitt et al, Recursion flattening[30]

    - https://dl.acm.org/citation.cfm?id=1366143
    - a recursion flattening algorithm
    - 再帰の深さが読める場合
- Maruyama et al, Hardware Implementation Techniques for Recursive Calls and Loops
    - https://link.springer.com/chapter/10.1007/978-3-540-48302-1_52
    - PGA フレンドリな再帰関数の F マルチスレッド実行アーキテクチャ
    - 合成手法には言及してない
    - この合成手法は取り入れたい，らしい

この論文を引用している論文

- From functional programs to pipelined dataflow circuits
    - https://dl.acm.org/citation.cfm?id=3033027
- A Principled Approach to Secure Multi-core Processor Desing with ReWrite
    - https://dl.acm.org/citation.cfm?id=2967497

関連して

- Implementing Recursive Algorithms within Hardware using Finite State Machine
    - http://referaat.cs.utwente.nl/conference/15/paper/7284/implementing-recursive-algorithms-within-hardware-using-finite-state-machines.pdf
- Hardware Implemenation of Recursive Algorithms
    - http://sweet.ua.pt/iouliia/Papers/2010/058_8208.pdf
- Constraint-Based Hardware Synthesis
    - http://www.uni-ulm.de/fileadmin/website_uni_ulm/iui.inst.170/home/raiser/publications/Triossi2010.pdf