

# コンテキスト指向プログラミングによる FPGA 動的部分再構成制御の検討

河村 まりや<sup>†</sup> 佐藤 未来子<sup>†</sup> 渡辺 晴美<sup>†</sup> 大川 猛<sup>††</sup>

**概要:** 近年, FPGA の動的部分再構成技術は様々なアプリケーションで用いられている. 一方で動作環境が多様なロボットでは, センサで取得した周辺環境データによって振る舞いを変化させる必要がある. そのようなロボットの開発にコンテキスト指向プログラミング(COP)によるレイヤ切り替えの技術が用いられている. 本技術は従来ソフトウェアコンポーネントの切り替えに用いられているが, 本研究ではこれを動的部分再構成へ応用する. COP で動的部分再構成を扱う場合, コンテキストとの結びつきをどのように行うか考える必要がある. 本研究ではコンテキストに応じた動的部分再構成を可能にするためのプラットフォームについて検討を行い, その構想について提案する.

## 1. はじめに

近年, FPGA (Field Programmable Gate Array) は画像処理や機械学習など様々なアプリケーションで用いられている[1][2]. FPGA は特定の機能に特化することで高性能な処理が可能である一方, 回路規模によっては全ての機能を搭載し切れない可能性がある. このような問題を解決する技術として動的部分再構成[3][4]というものがある. 動的部分再構成を用いることで, システムを停止することなく機能を入れ替え, 状況に応じて多様なハードウェア機能を提供することが可能となる. しかし, CPU 上で実行するソフトウェアと FPGA とを連動させながら活用する場合, CPU で動くソフトウェアモジュールと動的部分再構成を適用するモジュールを関連付けながらシステム全体を管理する必要がある. また, 動作環境が多様なロボットでは, センサで取得した周辺環境データによって振る舞いを変化させる必要があり, ソフトウェアのレイヤを切り替える際に用いられる技術としてコンテキスト指向プログラミング (以下 COP) [5] がある. 本研究では COP の技術を FPGA の動的部分再構成へ応用する. 本手法では, COP のレイヤにおいて, ソフトウェアモジュールの切り替えを指定するタイミングで, FPGA の構成を指定したものに変更することを可能とする. これにより, FPGA を含めたシステム構成をソフトウェアとハードウェアの両面で連携させて管理する. 本研究ではコンテキストに応じた FPGA の動的部分再構成を可能にするための FPGA 動的部分再構成制御方式について検討を行い, その構想について提案する.

## 2. 提案プラットフォーム

### 2.1 システムの概要

FPGA 動的部分再構成制御方法を担うプラットフォームの全体図を図 1 に示す. 各レイヤには FPGA で行う処理と CPU で行う処理があり, システムを取り巻く環境 (コンテ

キスト) に応じて Activate される. ソフトウェアコンポーネントの切り替えだけであれば, レイヤの Activate に応じて処理を切り替えるだけでよい. しかし, FPGA の処理切り替えを伴う場合は CPU 上の処理と連動させる必要がある. 本研究では, 動的部分再構成と各処理間の連携を管理するモジュールとして Reconfiguration Management を CPU 上に備え, ソフトウェアと連動した FPGA 動的部分再構成制御を担う. Reconfiguration Management はレイヤの Activate が行われると Partial Reconfiguration Controller (以下 PRC) に対してどのモジュールに切り替えるか指示を行う. PRC はその指示に従って動的部分再構成を行う.

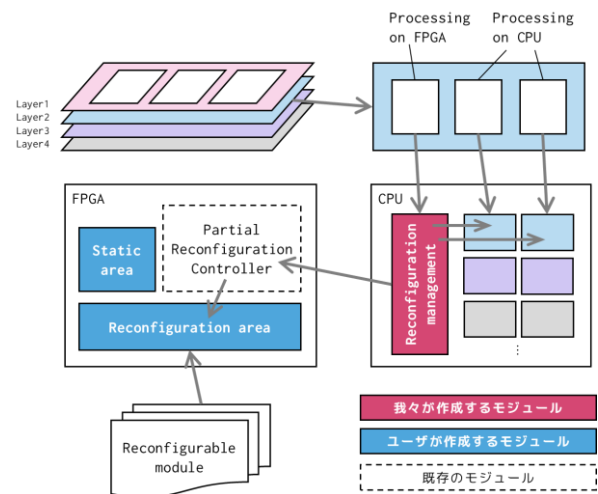


図 1 システムの全体図

### 2.2 Partial Reconfiguration Controller

FPGA 上の動的部分再構成管理には, Xilinx 社の Partial Reconfiguration Controller (以下 PRC) を利用する. PRC の構成と CPU 間の接続部分を図 2 に示す. PRC はソフトウェアまたはハードウェアからのトリガーを受け取ることで動的部分再構成を行う. Shutdown Handler によって再構成領域内の回路停止を要求し, 完了次第 Decoupler によって静的領域との分離が行われる. 分離が完了すると ICAP (Internal Configuration Access Port) によって新たなビットストリームが送信される. 最後に静的領域と再度接続して再構成領域内の回路の動作が開始される.

<sup>†</sup> 東海大学  
 Tokai University  
<sup>††</sup> 宇都宮大学  
 Utsunomiya University

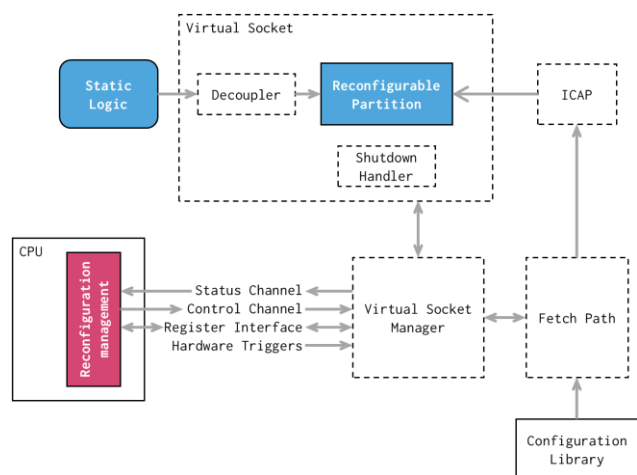


図2 Partial Reconfiguration Controller

### 2.3 Reconfiguration Management

Reconfiguration Management はレイヤの Activate が行われると PRC に対してどのモジュールに切り替えるか指示する役割を担っている CPU 上のモジュールである。COP と連携したシステムにおいて、Reconfiguration Management の要求仕様を以下に述べる。

- (1) COP と FPGA とが連携してレイヤを切り替えて動的にシステムを変えるためには、CPU から PRC に制御コマンドやソフトウェアトリガーを送ることとなる。その際、AXI4-Stream インターフェイスを用いた制御チャンネルや AXI4-Lite インターフェイスを用いたレジスタインターフェイスを介して通信を行う必要がある。Reconfiguration Management で本通信をサポートすることで、システム設計者が COP と PRC 間の連携のための通信方式を一から設計する必要がなくなる。
- (2) レイヤ切り替え時に、FPGA 上の動的部分再構成が完了しているかどうか知る必要がある。そのためには、Reconfiguration Management において PRC で把握される動的再構成の完了を CPU 上のレイヤ管理部へ通知する機能を設け、FPGA の再構成完了のタイミングに合わせてソフトウェア処理を再開するという同期処理を行うことが必要になる。

図3に Reconfiguration Management の構成例を示す。Reconfiguration Management は、COP のレイヤ管理からレイヤの Activate 通知を受け取ると、そのレイヤに対応したソフトウェアトリガーを Register Interface を通して Virtual Socket Manager に送信する。Virtual Socket Manager はソフトウェアまたはハードウェアからのトリガーを受け取り、Fetch Path に対して切り替えるモジュールのビットストリームのロードを要求する。動的部分再構成が完了すると、vsm\_<name>\_sw\_startup\_req という信号で再構成領域内の回路の動作許可を PRC から求められる。

また、Reconfiguration Management は動作許可を CPU 上の各処理に通知する。Reconfiguration Management は CPU 上の各処理から動作開始の準備が完了したという通知を受け取り次第、Virtual Socket Manager 上の CONTROL レジスタに Control Channel を通して Proceed コマンドを送信する。Virtual Socket Manager がコマンドを受信すると、再構成領域内の回路の動作が開始する。

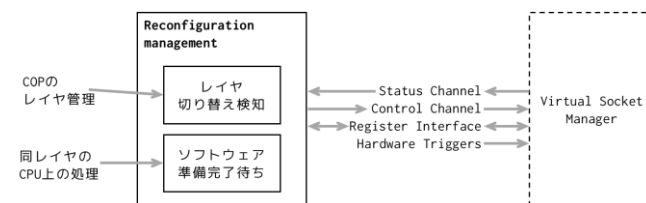


図3 Reconfiguration Management の構成例

### 3. おわりに

本研究では COP の技術を FPGA の動的部分再構成へ応用する方式について検討し、ソフトウェアレイヤーの切り替えと同時に、そのコンテキストに応じた FPGA の動的部分再構成を可能にするためのプラットフォームを構想し提案した。FPGA 上の動的部分再構成管理には、Xilinx 社の Partial Reconfiguration Controller (PRC) を利用し、COP によるレイヤの Activate が行われると PRC に対してどのモジュールに切り替えるか指示する Reconfiguration Manager が必要であり、その要求仕様および構成例を示した。

今後は、レイヤ切り替えからモジュールを切り替えるまでにかかる時間の評価を行う。また、FPGA 上の処理が複数ある、つまり再構成領域が複数ある場合の動作についても検討を行う。

### 参考文献

- [1] S. Pedre, T. Krajnik, E. Todorovich, P. Borensztein: Accelerating embedded image processing for real time: a case study, *Journal of Real-Time Image Processing*, Vol. 11, Issue 2, pp 349–374, 2016.
- [2] X. T. Nguyen, T. T. Hoang, H. T. Nguyen, K. Inoue, C. K. Pham, “An FPGA-Based Hardware Accelerator for Energy-Efficient Bitmap Index Creation”. *IEEE Access*. Vol. 6, pp. 16046-16059 (2018).
- [3] J. Khalifat, A. Ebrahim, A. Adetomi, T. Arslan, “A dynamic partial reconfiguration design for camera systems”. 2015 NASA/ESA Conference on Adaptive Hardware and Systems (2015).
- [4] H. M. Hussain, K. Benkrid, H. Seker, “Dynamic partial reconfiguration implementation of the SVM/KNN multi-classifier on FPGA for bioinformatics application”. 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (2015).
- [5] R. Hirschfeld, P. Costanza, O. Nierstarasz, “Context-Oriented Programming”. *Journal of Object Technology*. Vol.7, No.3, pp. 125-151 (2008).