

# データ駆動型プロセッサのプログラムキャッシュ回路の基礎検討

1250292 大崎 綾斗 【コンピュータ構成学研究室】

## 1 はじめに

近年, IoT(Internet of Things) デバイスの普及が進み, 高性能かつ低消費電力なプロセッサの需要が高まっている. その一つとして, データ駆動型プロセッサ (DDP:Data-Driven Processor) が有望である. 多種多様な応用プログラムを DDP 上で実行するためには, 実行時にプログラムを入れ替える機能が必要になる. よって, 本研究では, DDP のプログラム記憶部 PS を実行時に更新する方式に関して検討した.

## 2 DDP 向きプログラムキャッシュ方式

DDP は, 複数のステージのそれぞれが前段と後段のステージとハンドシェイクを行う C 素子により自己タイミングでデータ処理を行うセルフタイム型パイプラインによって構成されたプロセッサである. これに対して, 一般的な外部メモリは同期回路であるため, そのインタフェース回路には, 同期-非同期のタイミング調整回路に加えて, PS を更新する仕組みが必要になる.

本研究では, 図 1 に示すように, DDP の入力パケットを用いて PS を更新する方式 (PacketCache), および, DDP の PS ステージに更新データを直接転送して PS を更新する方式を検討した. 図 1 では, DDP と外部の異なるクロックドメイン間における様々な差異を吸収するために DDP の入出力パケットを一時的に保持する Packet I/O[1] を, 両方式で活用している.

### 2.1 パケット更新型キャッシュ方式

パケット更新型キャッシュ方式は, Packet I/O Cache に一時的に保持されたパケットが入力され, PS ステージまで到達したときに入力パケットの上位 3bit から更新データであるかを判別する. 更新データであれば PS の書き込み許可信号を有効にして書き込みを行う. また, それ以降のステージではそのパケットは不要であるため, 後段のステージにパケット削除信号を出力する.

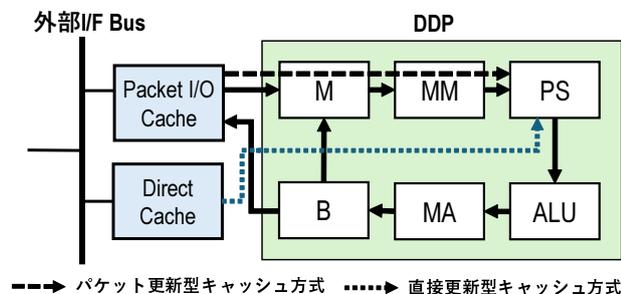


図 1 DDP 向きプログラムキャッシュ方式の概要

### 2.2 直接更新型キャッシュ方式

直接更新型キャッシュ方式は, 外部 I/F Bus からの更新データを Direct Cache に一時的に保持し, PS ステージへ直接データ転送を行う. この時, C 素子とのハンドシェイクにより PS に対する読み書きが同時に行われないうようにタイミングを図って更新を行う. そのために, Direct Cache から PS ステージに対する転送要求信号, および転送許可信号をもとに, Direct Cache の読み出し許可信号と PS の書き込み許可信号を制御する回路を設計して組み込むことで, Direct Cache と PS との間でデータ転送を行う. PS を更新している間は PS ステージの前段, および後段との C 素子間のハンドシェイクは停止し, 更新を開始すると Direct Cache が空になるまで DDP 外部のクロック信号に同期して PS を更新する.

## 3 評価

AMD 社の FPGA チップである Zynq-7010 上に, 2 つの提案回路を実装して, リソース使用量 (LUT:Look Up Table, Register, BRAM) と PS の更新時間  $T[\text{ns}]$  を評価した. 更新時間は, Packet I/O Cache または Direct Cache に最初のデータが書き込まれてから, PS に 64 エントリが書き込むまでの時間とする. この時, DDP 内部ではプログラムが動作していないものとする. 両方式の外部クロック信号は 50MHz とし, パケット更新型キャッシュ方式において DDP へのパケット入力から PS へ書き込むためのセルフタイムデータ処理信号が立ち上がるまでの遅延素子による遅延は約 38ns とした. これら条件下でのリソース使用量・更新時間を表 1 に示す.

リソース使用量は, リソース使用量の少ないパケット更新型を基準に, LUT: 約 1.7%, Reg.:2.4%, BRAM: 約 11% 増加している. LUT と Register の使用量では大きな差はなく, BRAM は, 直接更新型の Direct Cache が増加の原因であり, メモリサイズの縮小によりリソース使用量の差が縮まると期待される. 更新時間では, 直接更新型がパケット更新型より約 1.8 倍速いことが確認できる. そのため, 高速なバースト更新が必要な状況では直接更新型キャッシュ方式の利用が望ましいと考える.

## 参考文献

- [1] R. Uemoto, et al., "Reliable I/F circuit for embedded self-timed data-driven processors," ESCS'24, 2024.

表 1 リソース使用量・更新時間の結果

	LUT	Reg.	BRAM	T[ns]
パケット更新型	2313	1786	4.5	2,441
直接更新型	2354	1829	5	1,340