

不定サイクル演算に対応した分散制御のための状態符号化の検討

State Encoding for Distributed Controllers Handling Variable Latency Operations

清水美帆
Miho Shimizu

石浦菜岐佐
Nagisa Ishiura

関西学院大学 理工学部 School of Science and Technology, Kwansei Gakuin University

1 はじめに

高位合成において、実行サイクル数が実行時に変動する演算器を含む回路を効率的に制御する方法として、分散制御方式が提案されている [1, 2, 3]. しかし、これらの文献で扱われているのは、単一のデータフローグラフ (以下, DFG) で表現される小規模な回路のみであり、回路規模が大きくなった際の制御機械の状態符号化については議論が行われていない。そこで本稿では、Del Barrio の分散制御方式 [1] を対象に、大規模な制御構造を扱うための状態符号化について検討する。

2 Del Barrio の分散制御方式

Del Barrio の分散制御方式は、1つの演算器に1つの状態機械を割り当てて演算の実行を制御するものである。例えば、図1のDFGの実行は図2に示す状態機械により制御される。DFG中の演算1, 3は演算器Aで、演算2, 4は演算器Mで実行される。FSM A, FSM MはそれぞれA, Mを制御する状態機械である。状態1は演算1の実行開始を待つ状態であり、演算1の開始条件 $s1$ が1になると演算1を実行する。演算器Aの完了信号 $Aend$ が0である間は状態1'で完了を待ち、 $Aend$ が1になれば次の状態3へ遷移する。演算の開始条件は、演算の依存関係から決まる。例えば演算3は演算2の実行完了後に実行できるので、 $s3 = M(4) + M(4') + M(F)$ となる (ただし、 $M(i)$ はFSM Mが状態 i にあることを表す)。

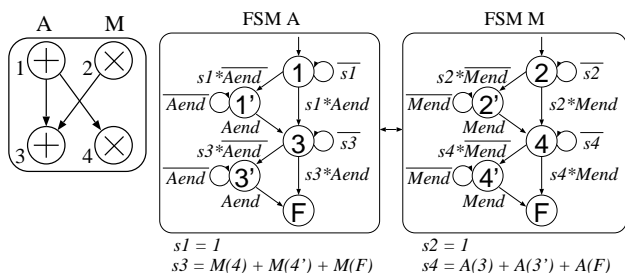


図1 DFG 図2 DFGを制御する状態機械

3 複数データフローグラフの制御と状態の符号化

与えられた動作仕様が複数のDFGで構成される場合には、DFG中の全演算が終了した後に次のDFGの演算を実行するように、各演算器のFSMを構成すれば良い。動作仕様中の総演算数を n 、DFG数を d 、演算器数を u とすると、制御に必要な状態数の総和は $2n + du$ となり、この状態をどう符号化するかが課題となる。各演算器のFSMの状態を2進符号やone hot codeで符号化する方法も考えられるが、one hot codeでは $2n + du$ 個のフリップフロップ (FF) が必要になる。一方、DFGのIDとDFG内の状態の符号を接続したものを状態符号として用いれば、状態遷移関数に影響を与えるFF数を削

表1 各FSMの符号化

		全状態の符号化	
1	bin	2進符号	
2	1h	one hot code	
		DFGのIDの符号化	DFG内状態の符号化
3	bin+bin	2進符号	2進符号
4	bin+1h	2進符号	one hot code
5	1h+1h	one hot code	one hot code

表2 実験結果

DFG数	演算数	符号化	Slices	FFs	LUTs	遅延 (ns)
10	159	bin	787	28	1486	11.250
		1h	543	358	1024	6.897
		bin+bin	703	32	1317	10.971
		bin+1h	674	64	1271	10.189
		1h+1h	768	88	1426	9.651
20	316	bin	1791	32	3408	13.231
		1h	1065	712	2013	7.899
		bin+bin	1595	36	3023	13.365
		bin+1h	1388	72	2624	11.694
		1h+1h	1445	132	2677	10.070
30	462	bin	2521	34	4834	14.677
		1h	1476	1044	2796	8.582
		bin+bin	2470	36	4715	13.925
		bin+1h	2154	72	4060	13.378
		1h+1h	2013	172	3737	10.303

減できると考えられる。

4 実験結果

表1に示す5通りの符号化について、制御回路を合成する実験を行った。実験にはランダムなDFG集合を用いた。各DFG中の演算数は平均15、演算器数は4、DFG数は10, 20, 30の3通りとした。制御回路は、演算器の開始信号の他に、入力側のマルチプレクサの選択信号も出力する。制御回路はVerilog HDLで記述し、Xilinx ISE (14.7)でFPGA (Spartan-3E)をターゲットに合成した。結果を表2に示す。FF数が問題でなければ、“1h”が最小の回路規模、遅延を与える。FF数を抑制したい場合には、“bin+1h”や“1h+1h”が有効と考えられる。

5 むすび

本稿では、不定サイクル演算に対応した分散制御のための状態符号化について検討した。実際の回路に対して評価実験を行うことが今後の課題である。

参考文献

[1] Alberto A. Del Barrio, et al.: “Using speculative functional units in high level synthesis,” in *Proc. DATE 2010*, pp. 1779–1784 (Mar. 2010).
 [2] Christian Pilato, et al.: “A runtime adaptive controller for supporting hardware components with variable latency,” in *Proc. AHS-2011*, pp. 153–160 (June 2011).
 [3] 山下, 石浦: “不定サイクル演算に対応した分散制御における動的演算バインディング,” 信学技報, VLD2013-128 (Jan. 2014).