

世界最小ソフト・プロセッサの設計

田中 雄一郎[†] 永塚 智之[‡] 吉瀬 謙二[‡]

東京工業大学 情報工学科[†] 東京工業大学 大学院情報理工学研究科[‡]

1 はじめに

近年、ソフト・プロセッサはField-Programmable Gate Array (FPGA) を使用するシステムにおいてより一般的なコンポーネントとなっており、制御及びデータ処理の幅広い機能を実装するために使用されている。携帯端末を始めとする小型デバイスにおいてハードウェア容量は限られており、その上に実装されるソフト・プロセッサは小さい方がより望ましい。そのため、最小のソフト・プロセッサを元に設計することは極めて有意義である。

本稿では、Supersmall Soft Processor[1] を元に世界最小ソフト・プロセッサの設計を行う。信号幅の変更と実行サイクルの見直しを行い、最小化に向け調整した Ultrasmall Soft Processor を提案する。

2 Supersmall Soft Processor

Supersmall Soft Processor (SSP) [1] とは、マルチクロックサイクルのスカラプロセッサである。ISA は 32 ビット MIPS 命令であるが、乗除算と非アライメントロード・ストアには対応していない。ALU の入出力信号を始めとする、各信号幅を 1 ビット化することで占有面積を減らしている。また、バレルシフトに演算結果を格納することで、シフト命令の実行も可能としている。ALU 使用時のデータフローを図 1 に、ロジカルなシフト命令処理時のデータフローを図 2 に示す。しかしこのような構造であるため、32 ビットの処理を行うには 32 サイクルを要するという欠点も持つ。

3 Ultrasmall Soft Processor

3.1 Ultrasmall Soft Processor の概要

FPGA の論理ブロックはルックアップテーブル (LUT)、全加算器、フロップフリップ (FF) などから構成された論理セルの集まりである。論理セルの構成はデバイスによって異なるため、各々の形状に適したハードウェア記述を行うことが望ましいと考えられる。

Ultrasmall Soft Processor (USP) がターゲットとするのは Spartan-3E XC3S500E, Spartan-6 XC6SLX16, Virtex-7 XC7VX485T である。Xilinx 社製 FPGA において論理セルはスライスと呼ばれ、このスライス数が最小となるように設計を行う。

Spartan-3E は 4 入力の LUT を持ち、Spartan-6 並びに Virtex-7 は 6 入力の LUT を持つ。入力数が異なるため、複雑なロジックを実装する上で必要となるスライス数は後者が少ないはずである。ロジックを複雑にすることで使用 FF を減らした場合、Spartan-3E 上では占有面積が大きくなるかもしれないが、残る 2 つでは却って小さくなる可能性がある。

USP のブロック図を図 3 に示す。USP では SSP に以下の手法を取り入れる。

- 信号幅の 2 ビット化
- 実行サイクルの見直し
- MUX による調整

Challenge to design the smallest soft processor
Yuichiroh TANAKA[†], and Tomoyuki NAGATSUKA[‡],
and Kenji KISE[†]

[†]Department of Computer Science,
Tokyo Institute of Technology

[‡]Graduate School of Information Science and Engineering,
Tokyo Institute of Technology

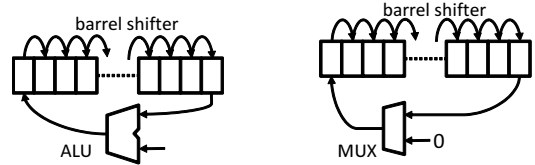


図 1: SSP で ALU を使用

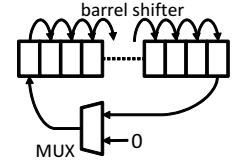


図 2: SSP でシフトを実行

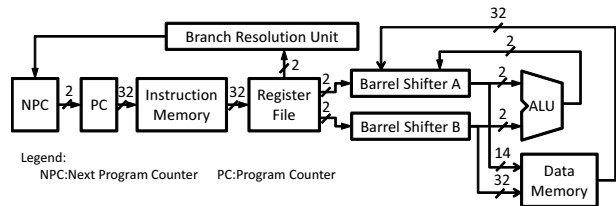


図 3: USP のブロック図

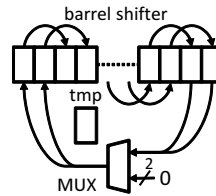


図 4: 2N ビットのシフト

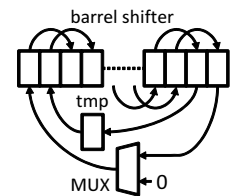


図 5: 1 ビットのシフト

3.2 信号幅の 2 ビット化

SSP において面積の削減に大きく貢献しているのは、信号幅を 1 ビットとしたことである。これにより、配線経路やレジスタを抑えることができる。しかし、信号幅を 2 ビットに変更しても変わるのは配線経路であり、必要とするレジスタの総数は殆ど変わらない。LUT の数だけが増加するため、複雑なロジックの実装に適した Spartan-6 並びに Virtex-7 上であれば SSP よりも小さくなる可能性がある。また、SSP に対して 1 サイクルで処理できるビット数が 2 倍になるために動作速度が大きく向上する。

一方で、奇数ビット分のシフトが困難になるという問題が生じる。図 2 に示した様に、SSP ではデータの保存先であるバレルシフタの特徴を活かしてシフトを行う。そのため、SSP と同様の設計では 2N ビットのシフトを行うことは可能であるが、奇数ビット分のシフトは不可能である。

そこで、USP ではシフトの処理状態を 2 つ用意することでこの問題を解決する。2N ビット分のシフト (図 4) と、1 ビット分のシフト (図 5) である。始めにシフト回数の LSB を無視して 2N ビット分のシフトを行い、LSB が 1 である場合のみ、追加で 1 ビット分のシフトを行う。このようにすることで、レジスタの総数を殆ど変化させることなくバレルシフタの特徴を活かすことが可能である。

3.3 実行サイクルの見直し

SSP は 32 ビットに対して処理を行うには 32 サイクルを必要とする。本来、32 サイクルで処理を終えて、次のサイクルでは別の処理を行うことが望ましい。しかし、

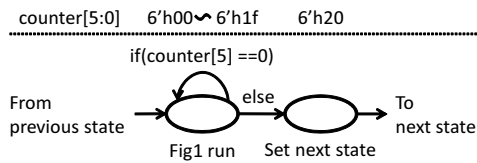


図 6: SSP で ALU を使用する際の状態遷移

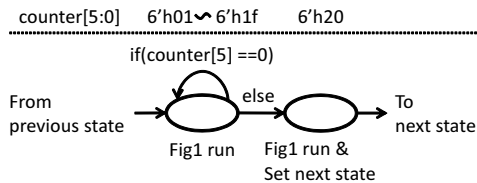


図 7: カウンタの初期値を 1 とした場合の状態遷移

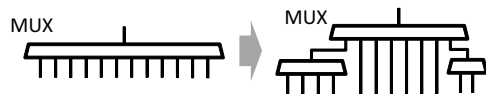


図 8: MUX による調整

SSP のカウンタは初期値が 0 であるため、32 サイクル目を検知するにはカウンタの下位 5 ビットに対して論理積を取る必要がある。これではロジックが複雑となってしまうため、SSP は 33 サイクル目では何も処理をせずに状態遷移だけ行う (図 6)。

これは徒に処理速度を低下させるだけでなく、配置配線を難化させる可能性がある。何故なら、33 サイクル目ではパレルシフタを作動させてはいけないため、カウンタを用いて動作の ON/OFF を切り替える必要があるからである。もし処理が 32 サイクルで終わるならば、切り替える必要がないためにカウンタからパレルシフタへの配線をなくすることができる。

そこで、カウンタの初期値を 1 とすることで 32 サイクル目で状態遷移を行えるようにする (図 7)。USP では信号幅が 2 ビットであるため、この手法により 1 状態の実行サイクルが 17 サイクルから 16 サイクルへと減ることになる。

3.4 MUX による調整

USP に含まれるマルチプレクサ (MUX) には 12 入力 1 出力のものがあり、殆どの状態でその制御信号を指定する必要がある。しかし、中には特定の状態でしか選択されない入力が存在する。これらの入力を図 8 の様に下段に纏めることで 2 段構成の MUX を形成する。使用頻度の少ない入力を下段に配置することで、全体としての制御信号の指定数が減ることが考えられる。

4 評価・考察

SSP は Altera 社製の Stratix III FPGA をターゲットとしており、バイト、ハーフワード単位のロード・ストア命令に関して、そのメモリスシステムに依存したハードウェア記述がなされている。我々は Xilinx 社製の FPGA をターゲットとするため、双方からこれらの命令処理部を除いて比較を行う。また、簡単のため例外処理部も除く。ツールは Xilinx 社製 ISE (ver14.2) を使用し、Optimization Goal を Area に、Optimization Effort を High と設定して評価を行う。

表 1 に各手法を実装した場合と SSP との比較を示す。信号幅の 2 ビット化、実行サイクルの見直し、2 つの手法の融合、MUX による調整を加えたものがそれぞれ、2bit, cycle, fusion, USP である。

2bit は Virtex-7 以外のデバイスにおいて SSP よりもハードウェア量が大きくなる。ロジックが複雑になるため Spartan-3E では LUT 数が増加し、スライス数でも大きく劣る。Spartan-6 並びに Virtex-7 へ変更した際の

表 1: SSP と各手法との比較

Spartan-3E						
	Slices	Slice Reg	LUTs	LUTRAM	BRAM	%
SSP	205	164	300	8	10	0
2bit	211	144	302	4	10	0
cycle	205	162	296	5	10	0
fusion	207	141	291	4	10	0
USP	205	141	289	4	10	0

Spartan-6						
	Slices	Slice Reg	LUTs	LUTRAM	BRAM	%
SSP	140	152	205	4	10	31.7
2bit	142	144	201	2	10	32.7
cycle	131	149	205	1	10	33.7
fusion	142	142	203	2	10	31.4
USP	139	142	203	2	10	32.2

Virtex-7						
	Slices	Slice Reg	LUTs	LUTRAM	BRAM	%
SSP	163	152	200	4	6	20.5
2bit	153	144	202	2	6	27.5
cycle	154	149	194	1	6	24.9
fusion	158	142	202	2	6	23.7
USP	157	142	200	2	6	23.4

スライスの減少率は SSP よりも大きくなる。一方、予想に反して全体的にレジスタ数が減少するのは、あるパレルシフタが分散 RAM によって形成されるようになるためである。Spartan-6 ではスライス数以外の値では全て SSP より下回るため、配置配線も最小化において重要であることが分かる。

cycle では配置配線の易化と共に、無駄なサイクルの削減を図る。そのため、使用レジスタ数に変化はないが Spartan-6 並びに Virtex-7 上でスライス数が大きく減っており、この手法の効果が現れる。

以上 2 つの手法をそのまま融合したものが fusion である。2bit と同じように Virtex-7 を除く 2 つのデバイスで SSP より大きくなる。しかし、Spartan-3E 上においてもスライス数を除く全ての値で SSP を下回っており、これも fusion が配置配線に不向きな構成をしていたためだと考えられる。

USP は fusion に MUX による調整を行ったものである。全てのデバイス上で SSP と同等、もしくはそれ以上に小さいソフト・プロセッサとなった。我々が調べた限りで SSP が最小のソフト・プロセッサであるため、USP は世界最小であると言える。

5 まとめ

ソフト・プロセッサの最小化を目的として、占有面積の削減並びに速度向上を実現する手法を提案した。しかし、一部ではレジスタ、LUT 共に SSP よりも少ないにも関わらず、スライス数は大きくなるという事例が生じた。これは配置配線による差であると考えて USP を実装したが、これは ISE の出力結果からの判断によるものであり、実配線を元にした判断ではない。そのため、この配置配線の考察は不十分であると考えられる。今後の課題として、実配線を元にした最適な構成手法の提案が挙げられる。

参考文献

[1] Robinson, J. and Vafae, S. and Scobbie, J. and Ritche, M. and Rose, J. The supersmall soft processor Programmable Logic Conference (SPL), 2010 VI Southern.