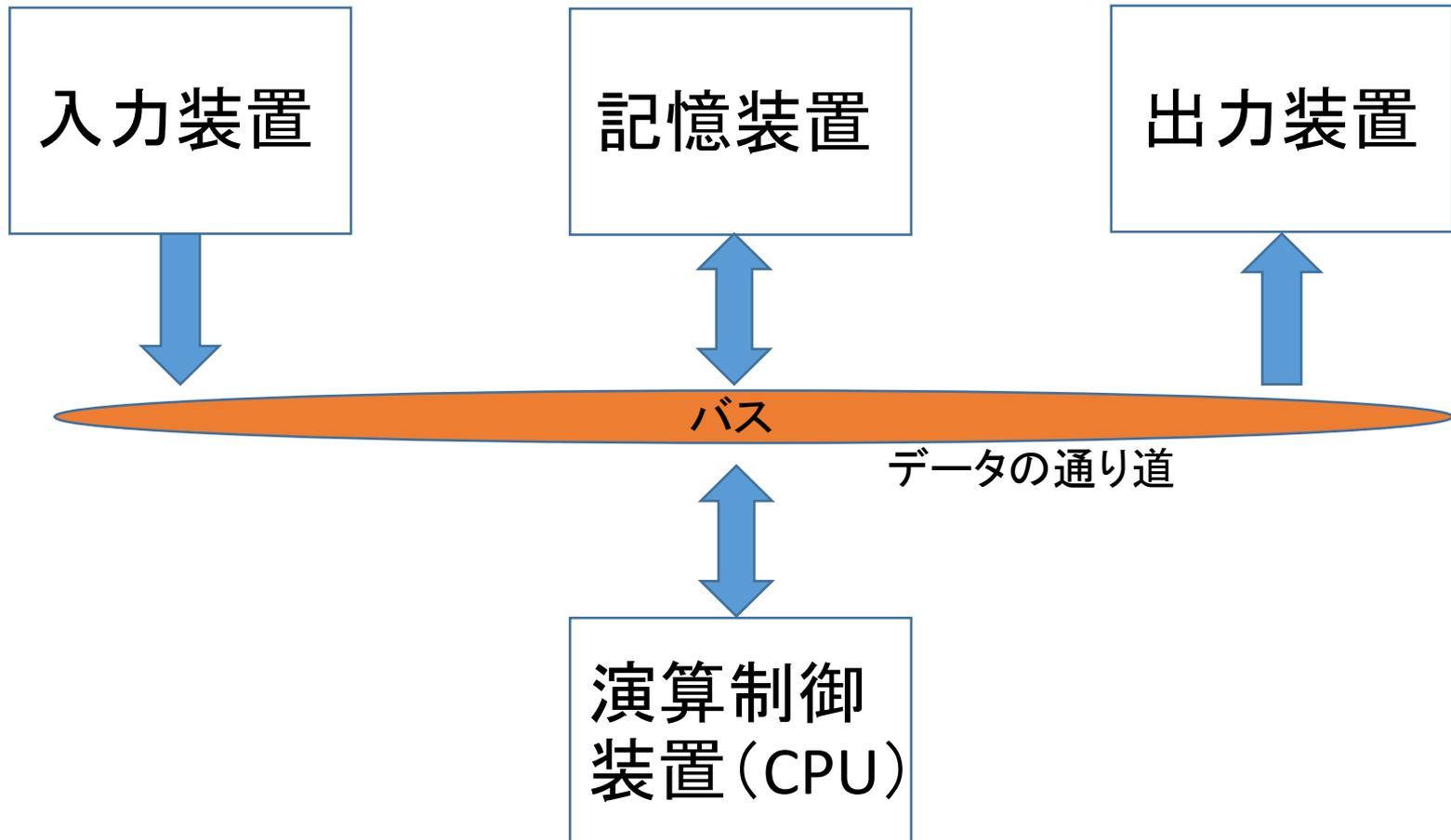


コンピュータの仕組み

コンピュータの基本構成



入出力装置

- プログラムやデータ(情報)を入力したり結果を出力したりする機器
- 入力装置の例
?
- 出力装置の例
?

記憶装置

- プログラムやデータを保存する装置
- 主記憶装置と補助記憶装置がある
- 主記憶装置
 - メモリとも呼ぶ
 - CPUと高速にやり取りするためにプログラムとデータを短期的に記憶するもの
 - 電源が消えるとデータが消える
 - 補助記憶装置や入出力装置ともデータのやり取りを行う
 - 記憶できるデータの量が少ない

記憶装置

- 補助記憶装置

- プログラムやデータを長期的に記憶するもの
- 電源がなくてもデータを保存できる
- 記憶できるデータの量のはるかに多い
- 既存のもの:

- HDD: Hard Disk Drive
 - SSD: Solid State Drive
 - メモリカード: USBメモリ、SDメモリカードなど
 - Blu-ray Disc/DVD/CD
- } ストレージと呼ばれる

Question

- 右はPCのWebカタログの一例。ここから主記憶装置と補助記憶装置の容量はどこから読み取れるか？

Winノート Latitude 7490



台数限定！14インチLatitude7000シリーズ

Core i5-8250U

メモリ: 8GB

SSD256GB M.2 PCIe NVMe

在庫あり

税込組価 **99,900円**

本体価格 90,819円

消費税 9,081.9円

注文番号 80579916

数量

カートへ入れる

価格は予告なく変更される場合がございますので、ご了承ください。

●台数限定

メーカー名	DELL
スペック・備考	●OS:Win10Pro64bit●CPU:Corei5-8250U(1.6GHz, 4コア)●メモリ:8GB(4GBx2/空0/32GB)●ストレージ:SSD256GB M.2 PCIe NVMe●ディスプレイ:14インチFHD(1920x1080)非タッチ●ドライブ:無し●駆動時間:約17時間●ネットワーク:802.11ac+Bluetooth4.1●重量:約1.4kg●保証内容:2年間翌営業日出張修理●その他:HDMIx1, USB3.1x3, USB Type-Cx1、Webカメラ

演算制御装置 (CPU)

- Central Processing Unit, 中央処理装置。コンピュータの中で「頭脳」と言われるほどもっとも重要な役割を担う
- CPU=プロセッサ=マイクロプロセッサ (ほぼ同義に使われる)
- 演算装置、制御装置などから構成される

演算制御装置 (CPU)

- 演算装置
 - ALU(Arithmetic and Logic Unit), 算術論理演算装置
 - 名の通り、四則演算や論理演算を行う

演算制御装置 (CPU)

- 制御装置

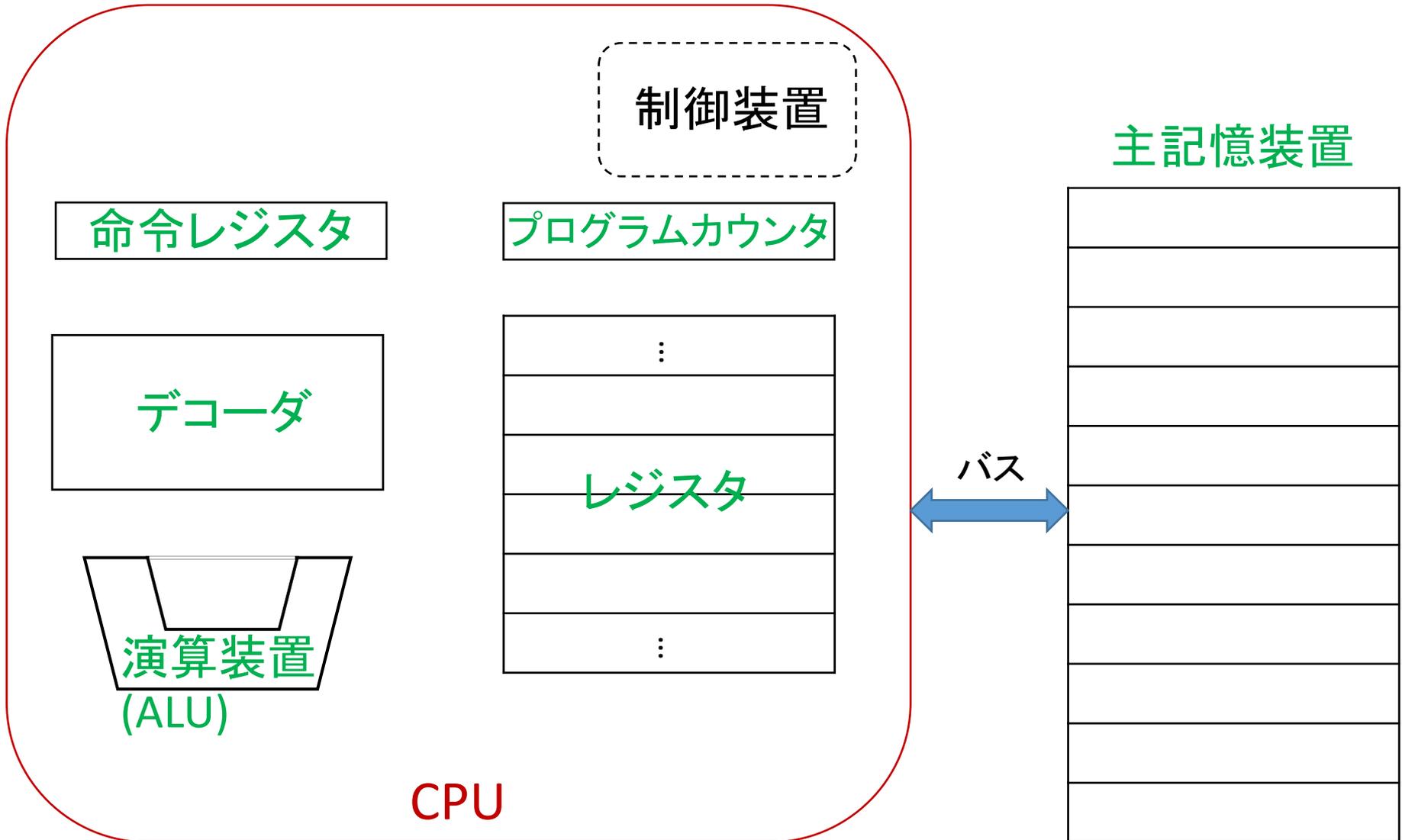
- CPUの動作は命令に従う。命令の実行ともいう。命令の実行は手順に従う。その手順を制御(管理)するのが制御装置の役割
- 制御(管理)は以下のようなもの:
 - 実行の順番 (PC更新、命令取り出し、命令解読...)
 - 実行のタイミング
 - 解読された命令に従っての各装置の制御 (後のアドレスデコーダ・命令デコーダの例を参照)

演算制御装置 (CPU)

- 制御装置

- 実際の制御は、決まったタイミングで、命令に従い、必要な信号を生成し、これら信号で各装置をオン・オフしているだけで、「制御装置」よりは「**制御信号群**」と見なした方がよいかもしれない
- 以降のCPUの図中には、制御装置は必ずしも**明示しない**

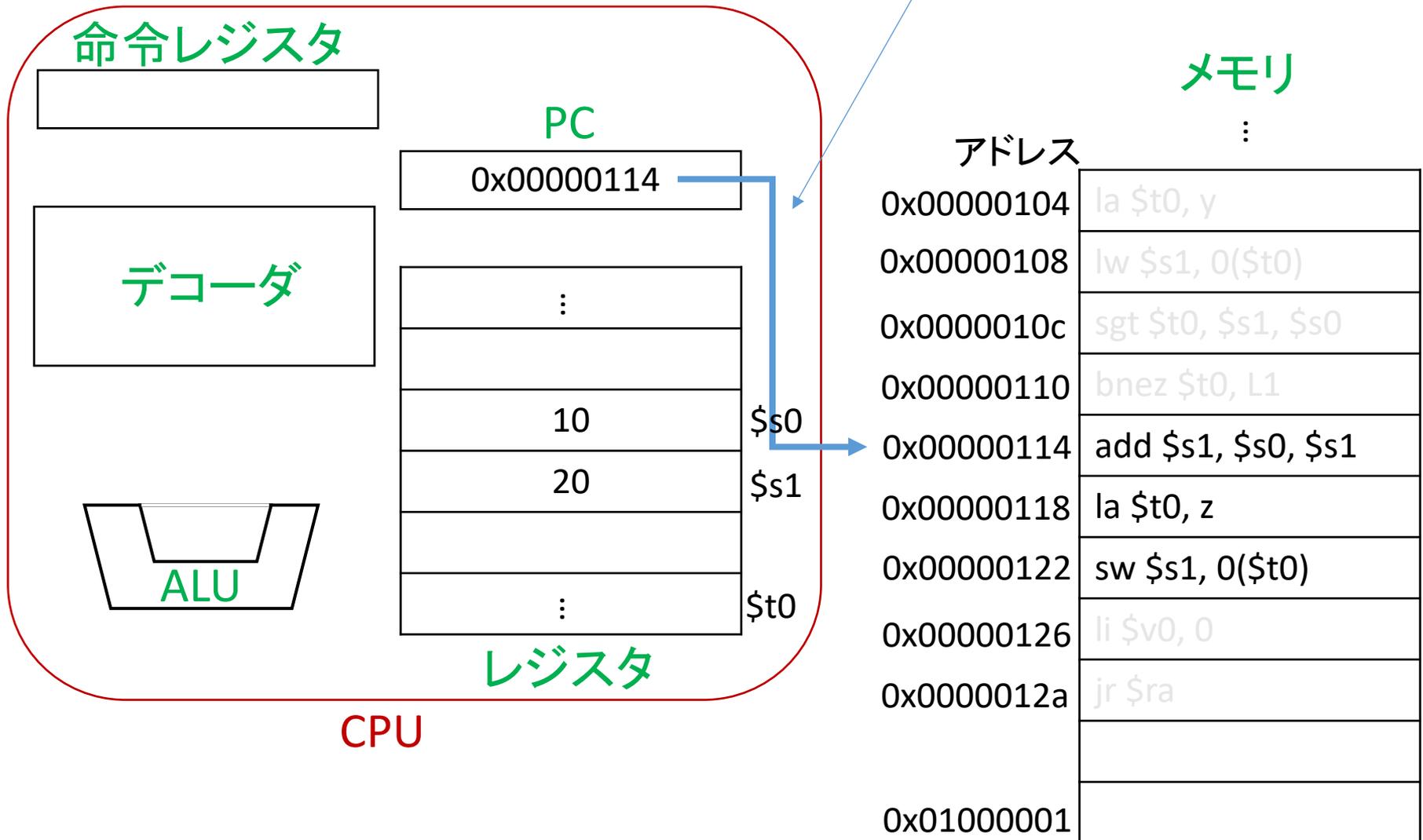
CPUの構成



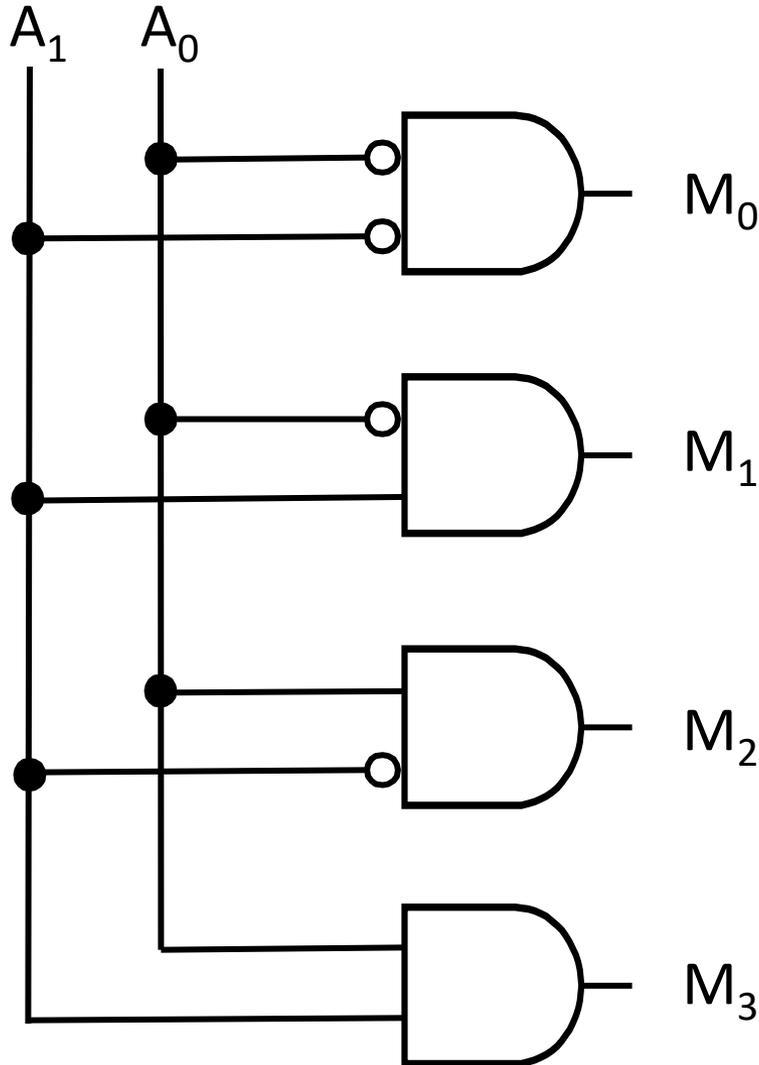
CPUの構成

- レジスタ
 - 一時的に情報を保持する記憶回路。もっとも高速だが、数個から数十個しかない
- 命令レジスタ
 - 命令を記憶するレジスタ
- プログラムカウンタ(PC)
 - 次に実行する命令のメモリアドレスを記憶するレジスタ
- デコーダ
 - 命令を解読する装置
- 演算装置(ALU)
- 制御装置

処理の流れ



アドレスデコーダ (2ビットの例)

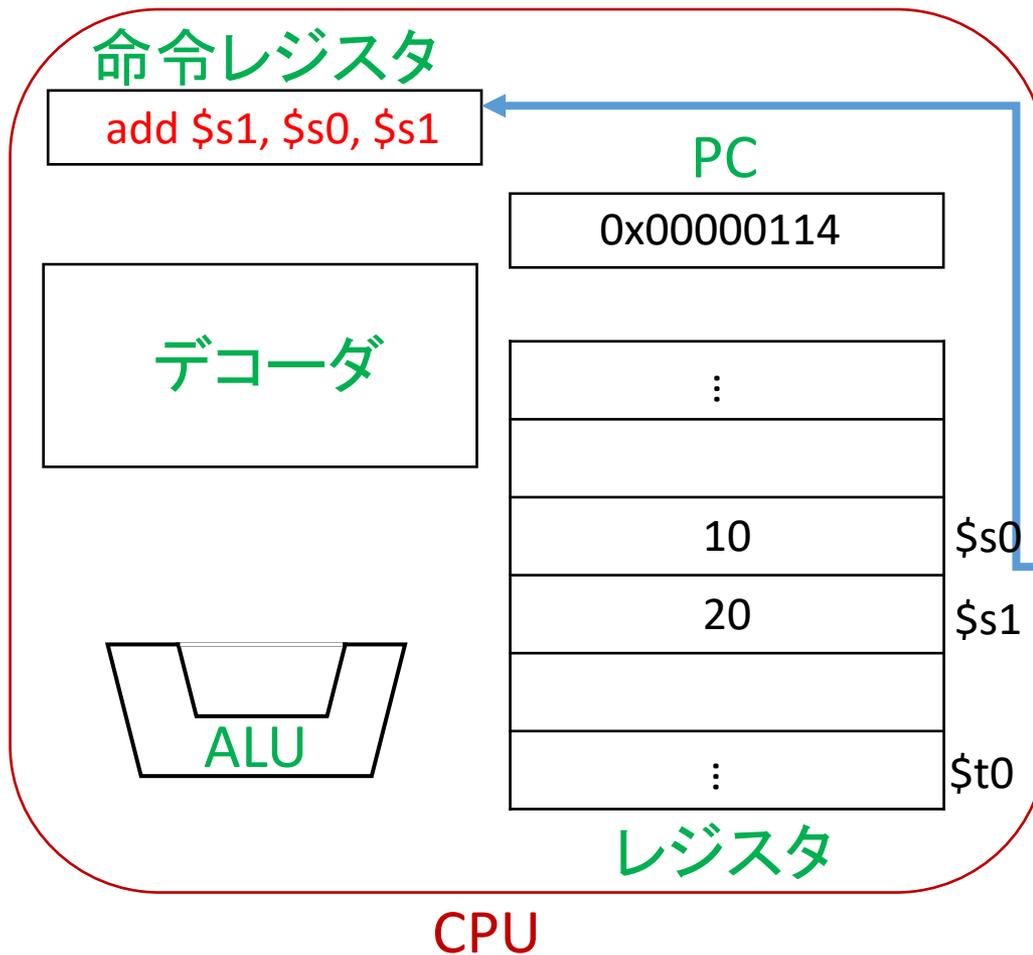


アドレス選択する制御信号



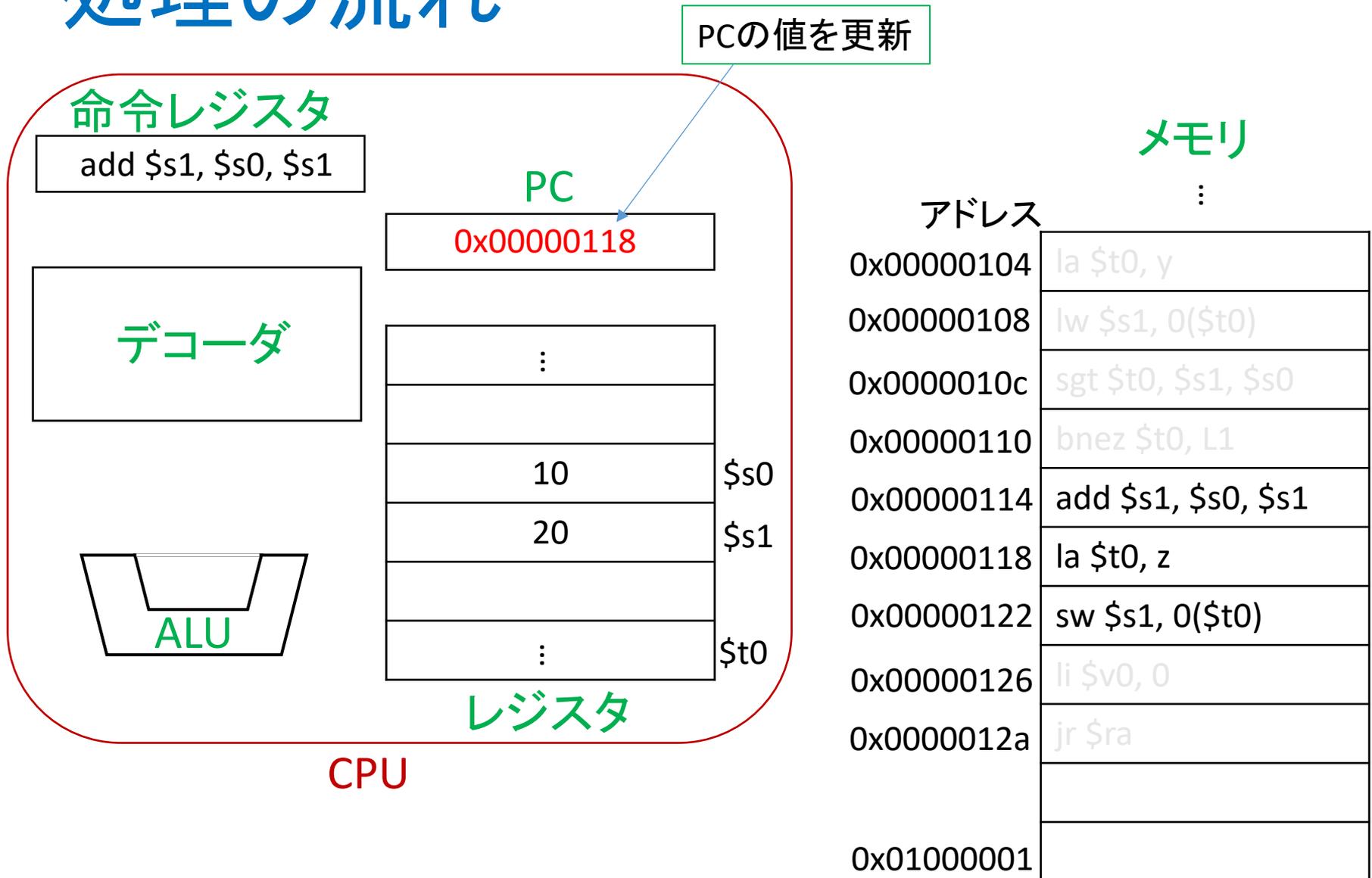
各メモリへ

処理の流れ

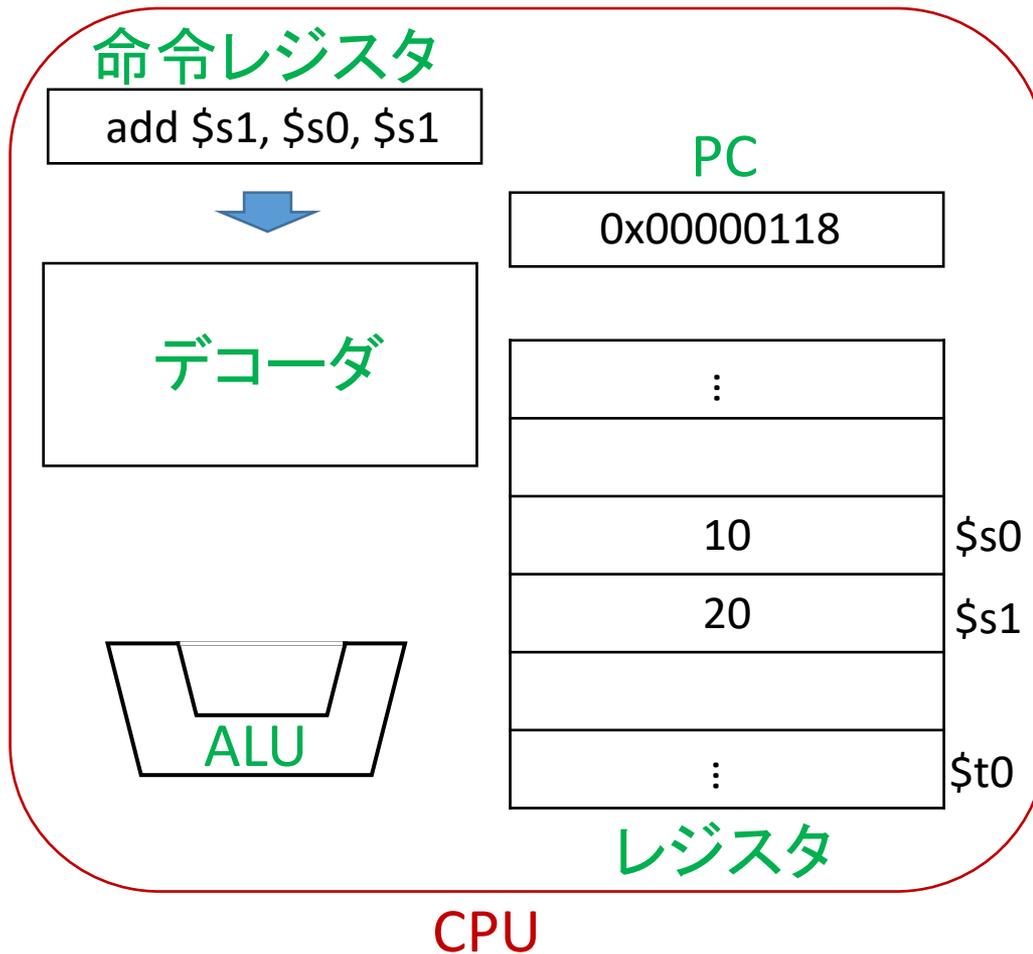


メモリ	
アドレス	
	⋮
0x00000104	la \$t0, y
0x00000108	lw \$s1, 0(\$t0)
0x0000010c	sgt \$t0, \$s1, \$s0
0x00000110	bnez \$t0, L1
0x00000114	add \$s1, \$s0, \$s1
0x00000118	la \$t0, z
0x00000122	sw \$s1, 0(\$t0)
0x00000126	li \$v0, 0
0x0000012a	jr \$ra
0x01000001	

処理の流れ



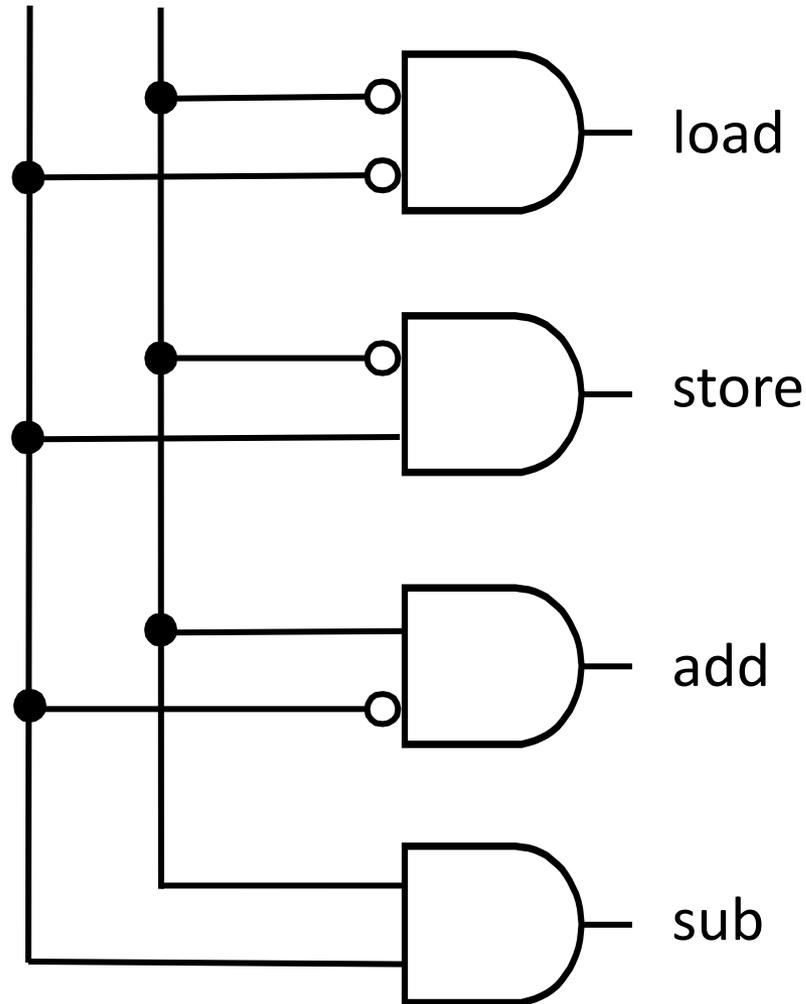
処理の流れ



アドレス	メモリ
0x00000104	la \$t0, y
0x00000108	lw \$s1, 0(\$t0)
0x0000010c	sgt \$t0, \$s1, \$s0
0x00000110	bnez \$t0, L1
0x00000114	add \$s1, \$s0, \$s1
0x00000118	la \$t0, z
0x00000122	sw \$s1, 0(\$t0)
0x00000126	li \$v0, 0
0x0000012a	jr \$ra
0x01000001	

命令デコーダ(例)

命令上位2ビット

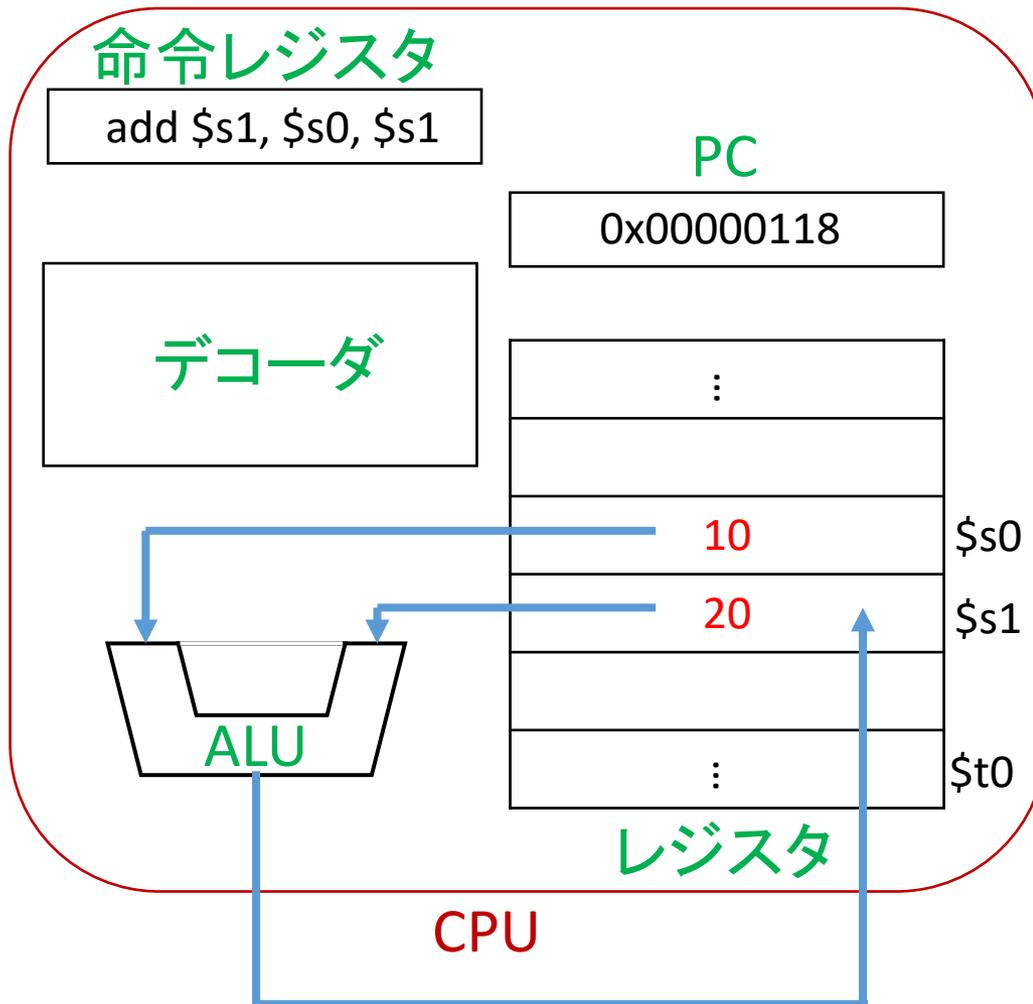


命令を指示する制御信号



ALUやレジスタへ

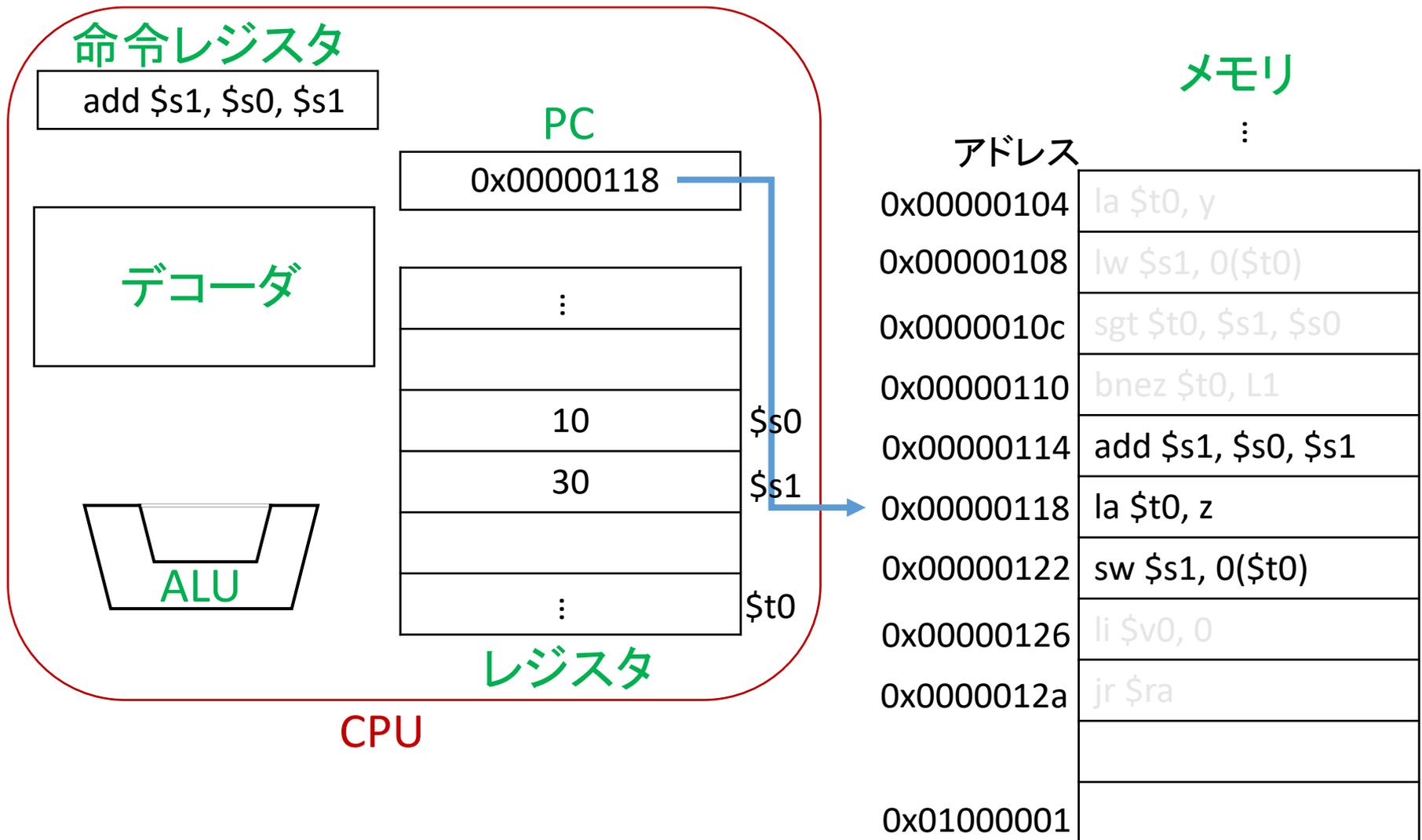
処理の流れ



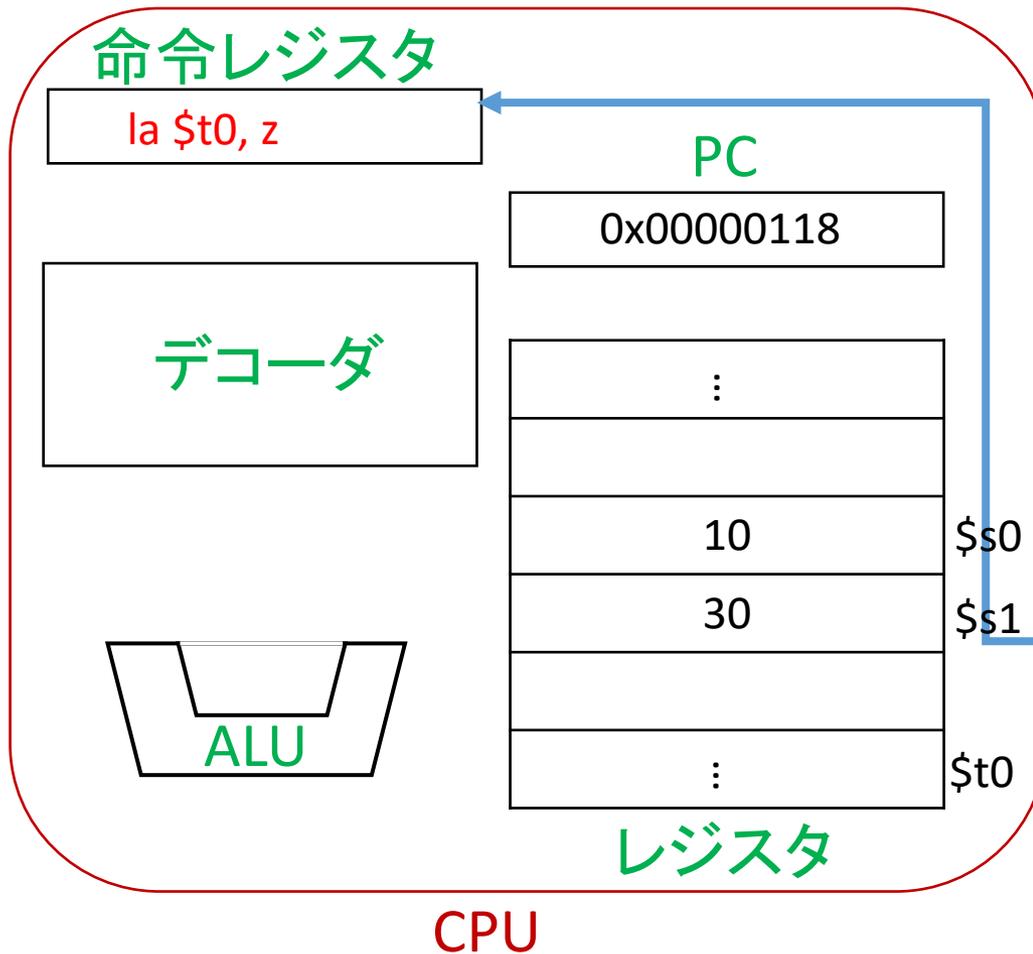
メモリ

アドレス	メモリ
0x00000104	la \$t0, y
0x00000108	lw \$s1, 0(\$t0)
0x0000010c	sgt \$t0, \$s1, \$s0
0x00000110	bnez \$t0, L1
0x00000114	add \$s1, \$s0, \$s1
0x00000118	la \$t0, z
0x00000122	sw \$s1, 0(\$t0)
0x00000126	li \$v0, 0
0x0000012a	jr \$ra
0x01000001	

処理の流れ



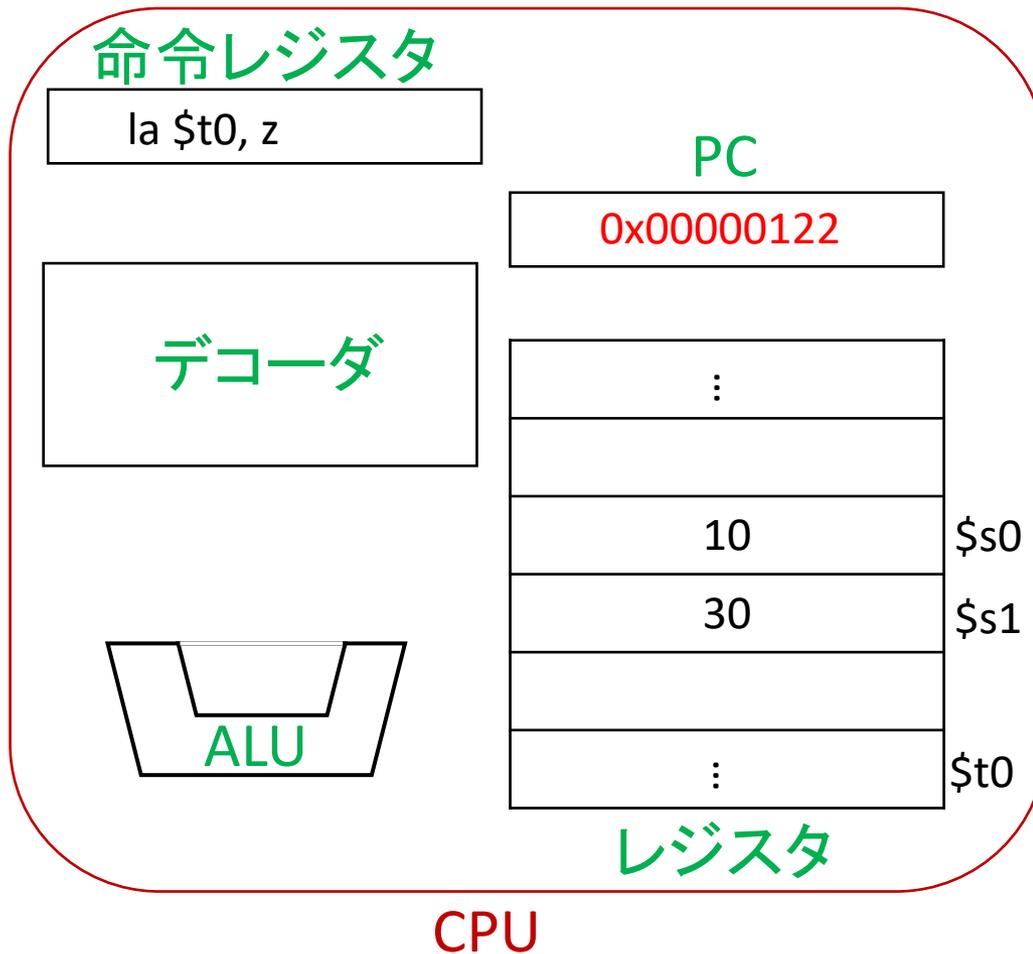
処理の流れ



メモリ

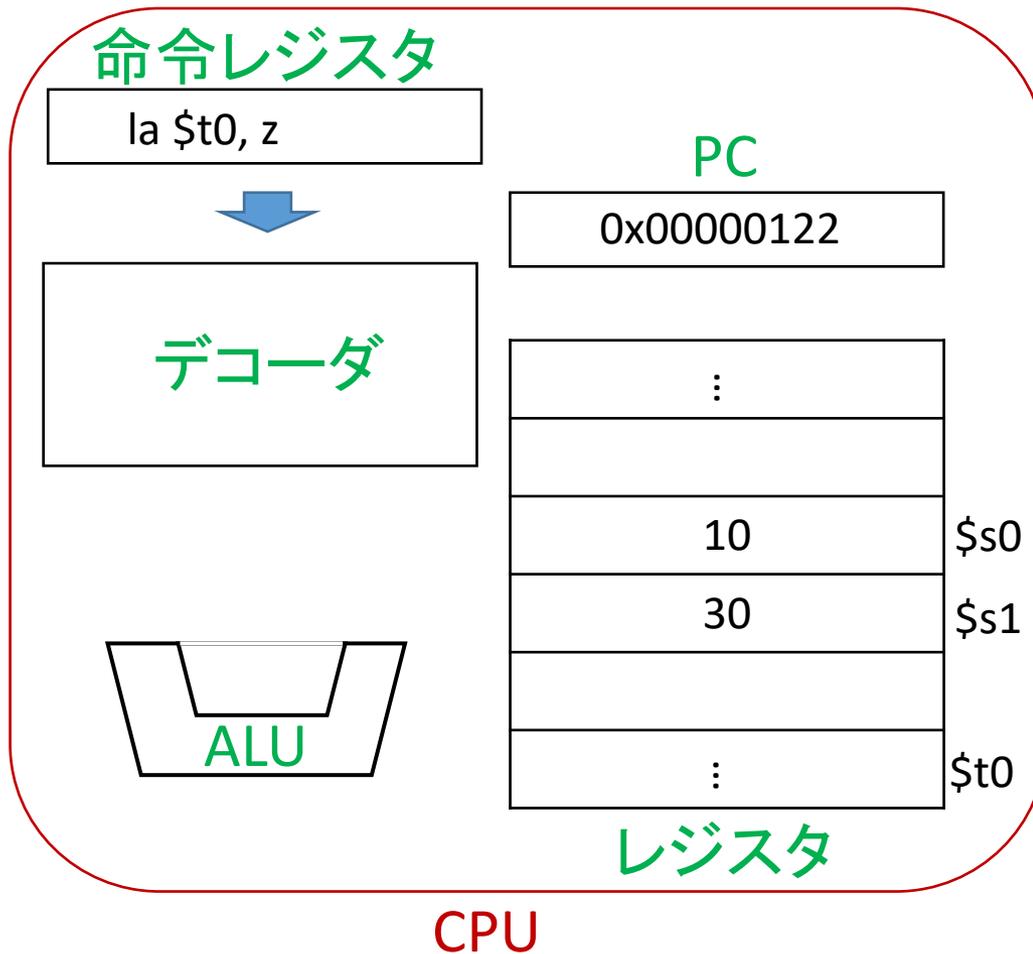
アドレス	メモリ
0x00000104	la \$t0, y
0x00000108	lw \$s1, 0(\$t0)
0x0000010c	sgt \$t0, \$s1, \$s0
0x00000110	bnez \$t0, L1
0x00000114	add \$s1, \$s0, \$s1
0x00000118	la \$t0, z
0x00000122	sw \$s1, 0(\$t0)
0x00000126	li \$v0, 0
0x0000012a	jr \$ra
0x01000001	

処理の流れ



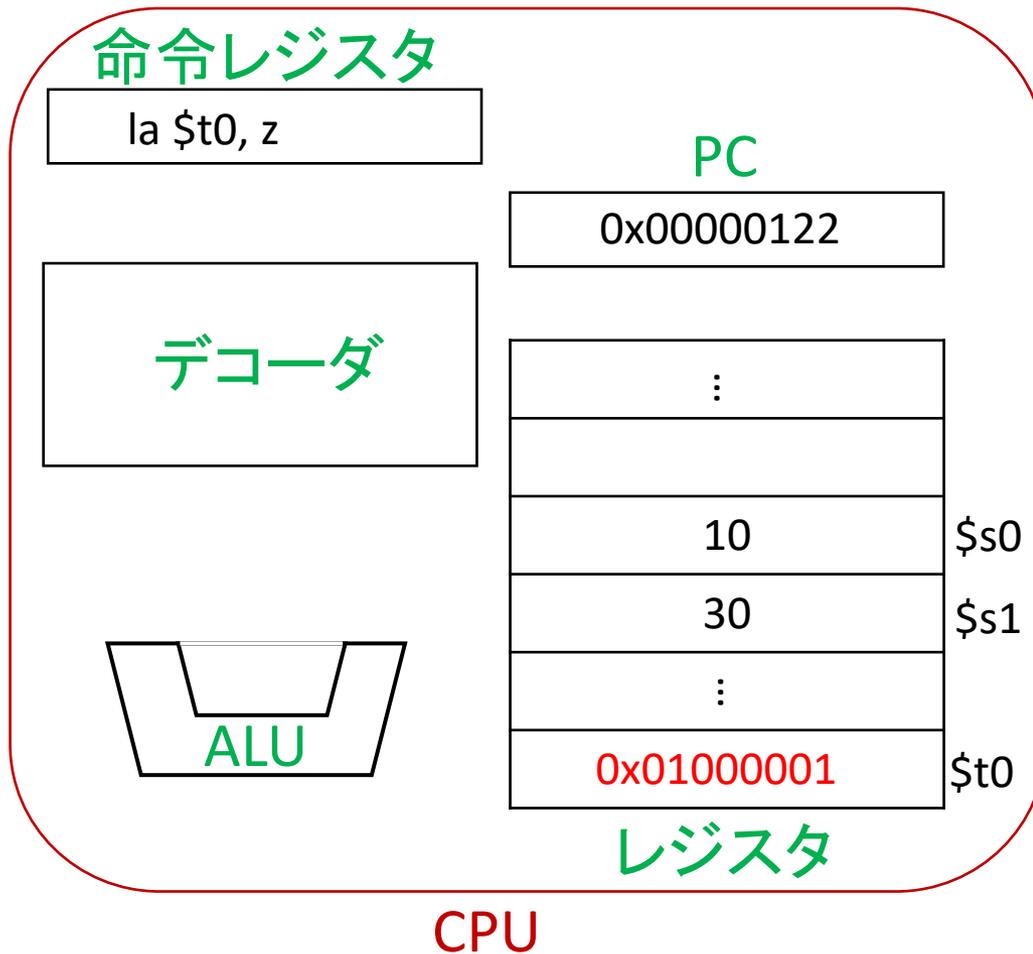
メモリ	
アドレス	:
0x00000104	la \$t0, y
0x00000108	lw \$s1, 0(\$t0)
0x0000010c	sgt \$t0, \$s1, \$s0
0x00000110	bnez \$t0, L1
0x00000114	add \$s1, \$s0, \$s1
0x00000118	la \$t0, z
0x00000122	sw \$s1, 0(\$t0)
0x00000126	li \$v0, 0
0x0000012a	jr \$ra
0x01000001	

処理の流れ



メモリ	
アドレス	
	⋮
0x00000104	la \$t0, y
0x00000108	lw \$s1, 0(\$t0)
0x0000010c	sgt \$t0, \$s1, \$s0
0x00000110	bnez \$t0, L1
0x00000114	add \$s1, \$s0, \$s1
0x00000118	la \$t0, z
0x00000122	sw \$s1, 0(\$t0)
0x00000126	li \$v0, 0
0x0000012a	jr \$ra
0x01000001	

処理の流れ

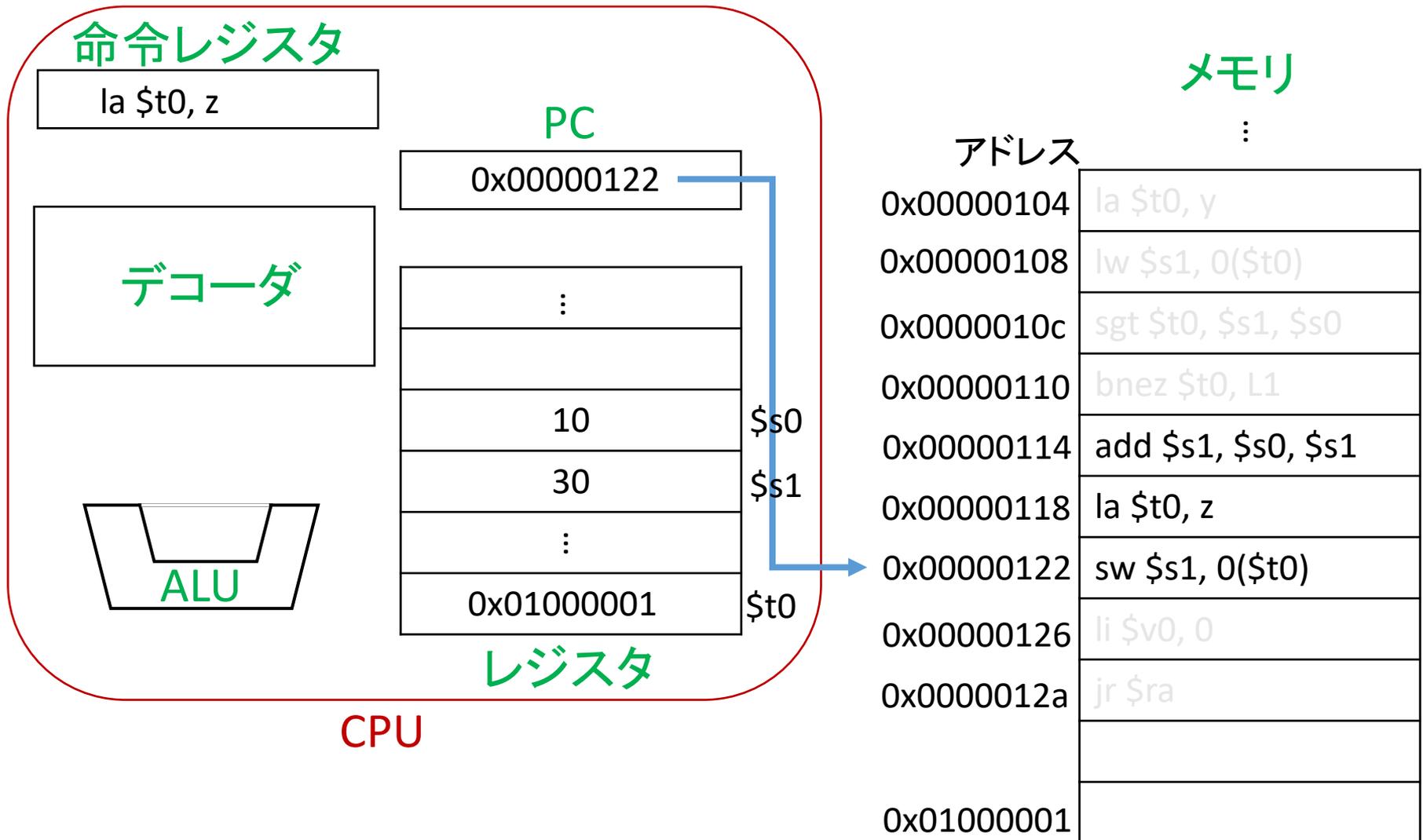


メモリ

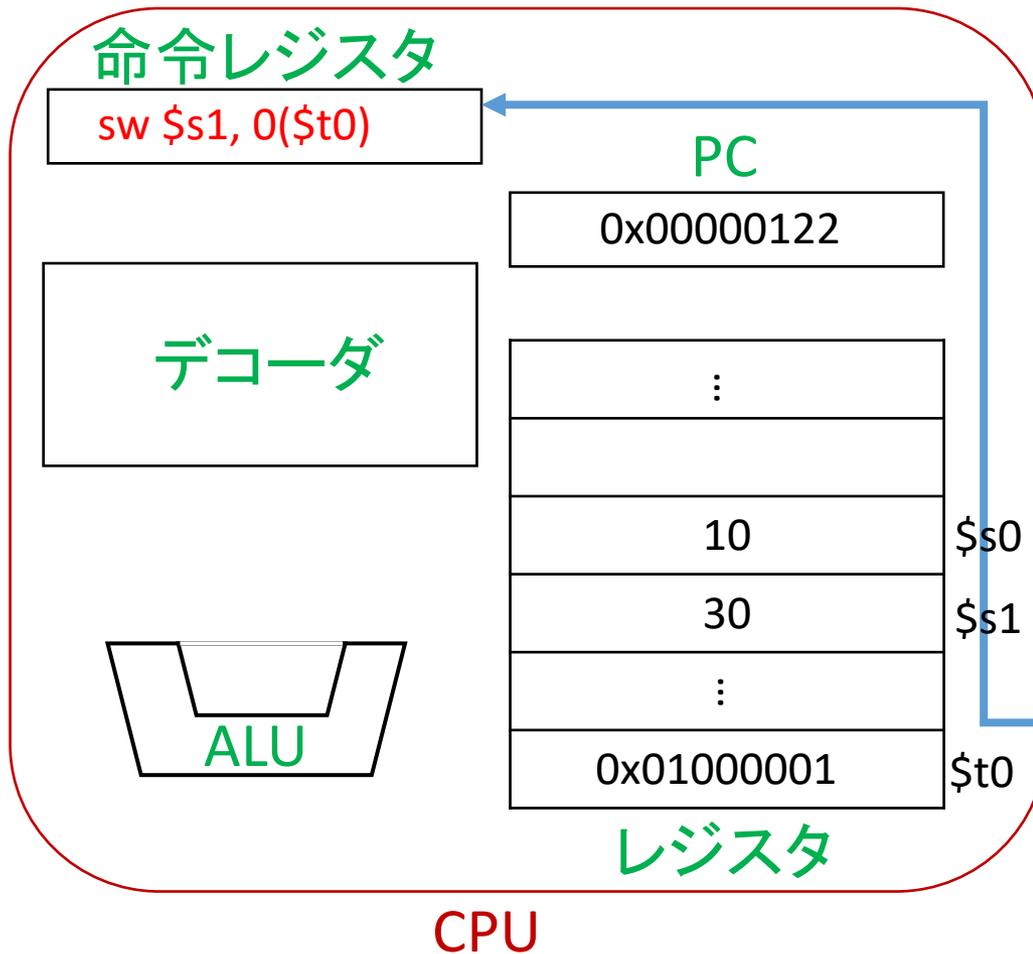
アドレス

0x00000104	la \$t0, y
0x00000108	lw \$s1, 0(\$t0)
0x0000010c	sgt \$t0, \$s1, \$s0
0x00000110	bnez \$t0, L1
0x00000114	add \$s1, \$s0, \$s1
0x00000118	la \$t0, z
0x00000122	sw \$s1, 0(\$t0)
0x00000126	li \$v0, 0
0x0000012a	jr \$ra
0x01000001	

処理の流れ



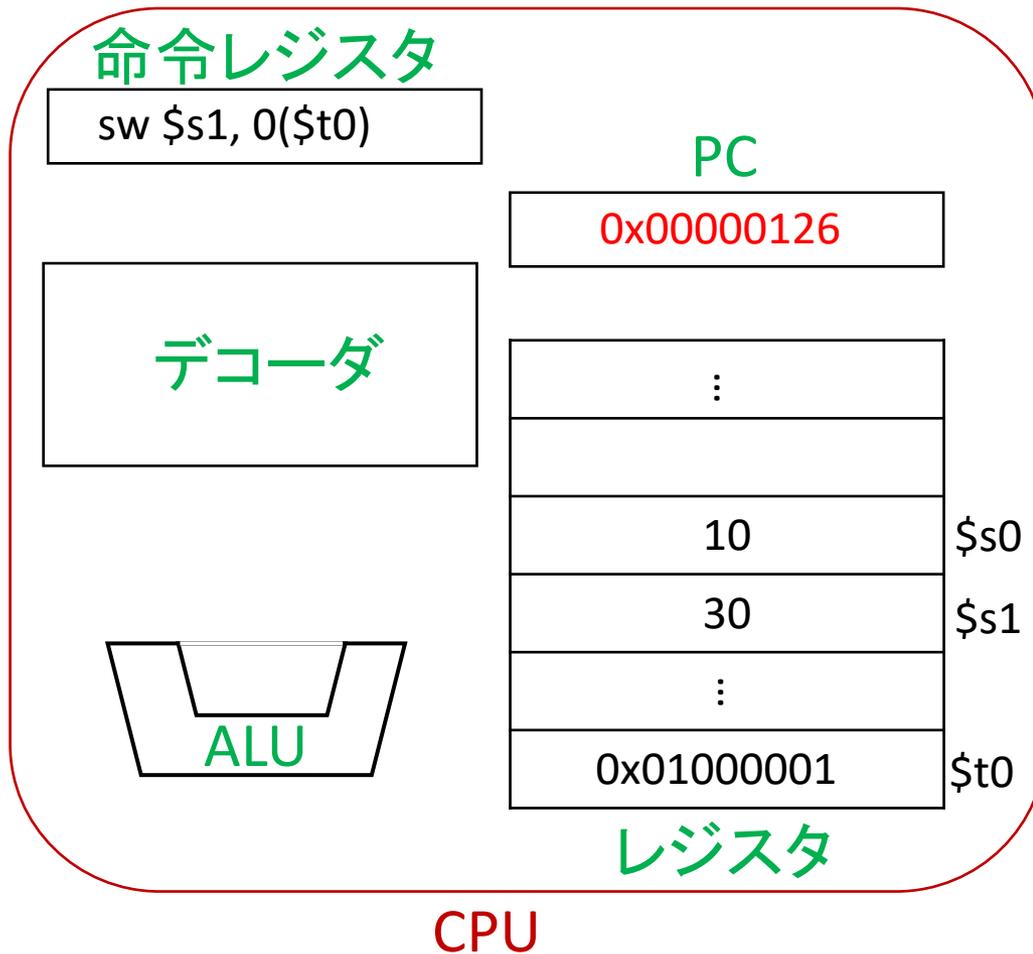
処理の流れ



メモリ

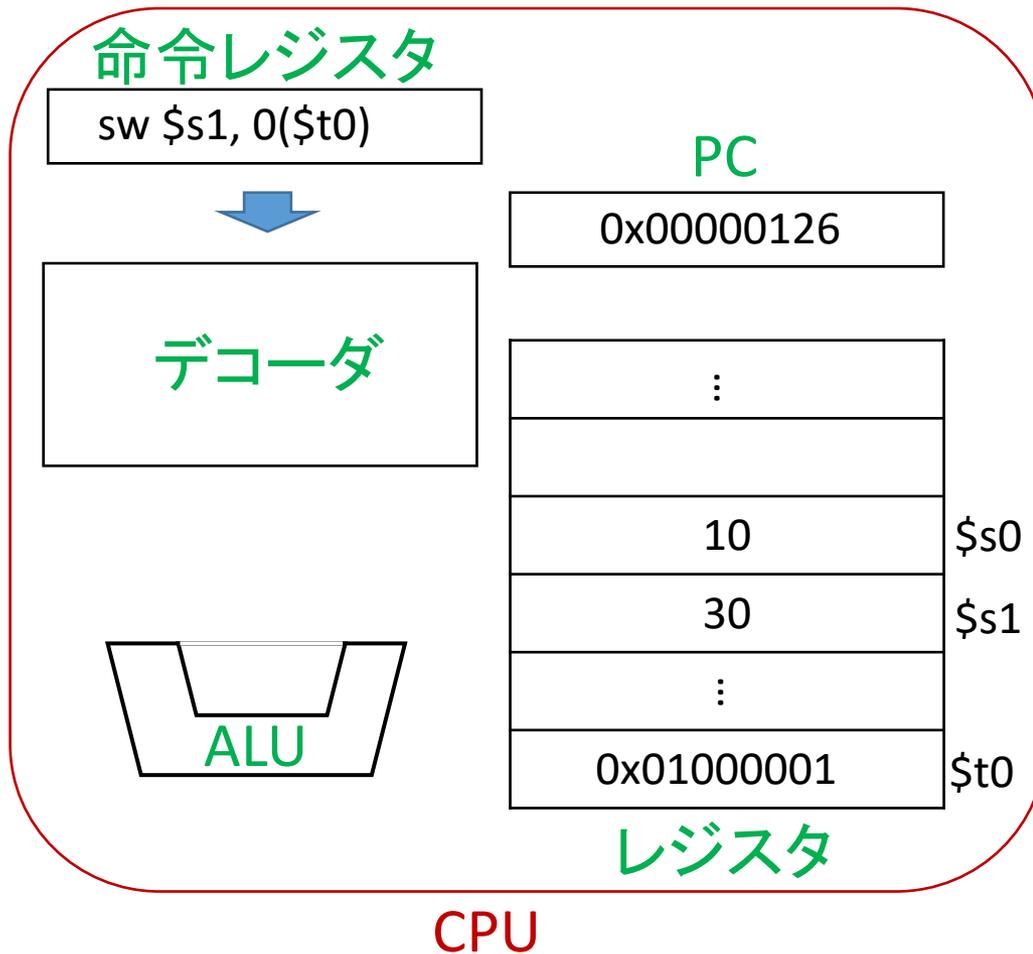
アドレス	メモリ
0x00000104	la \$t0, y
0x00000108	lw \$s1, 0(\$t0)
0x0000010c	sgt \$t0, \$s1, \$s0
0x00000110	bnez \$t0, L1
0x00000114	add \$s1, \$s0, \$s1
0x00000118	la \$t0, z
0x00000122	sw \$s1, 0(\$t0)
0x00000126	li \$v0, 0
0x0000012a	jr \$ra
0x01000001	

処理の流れ



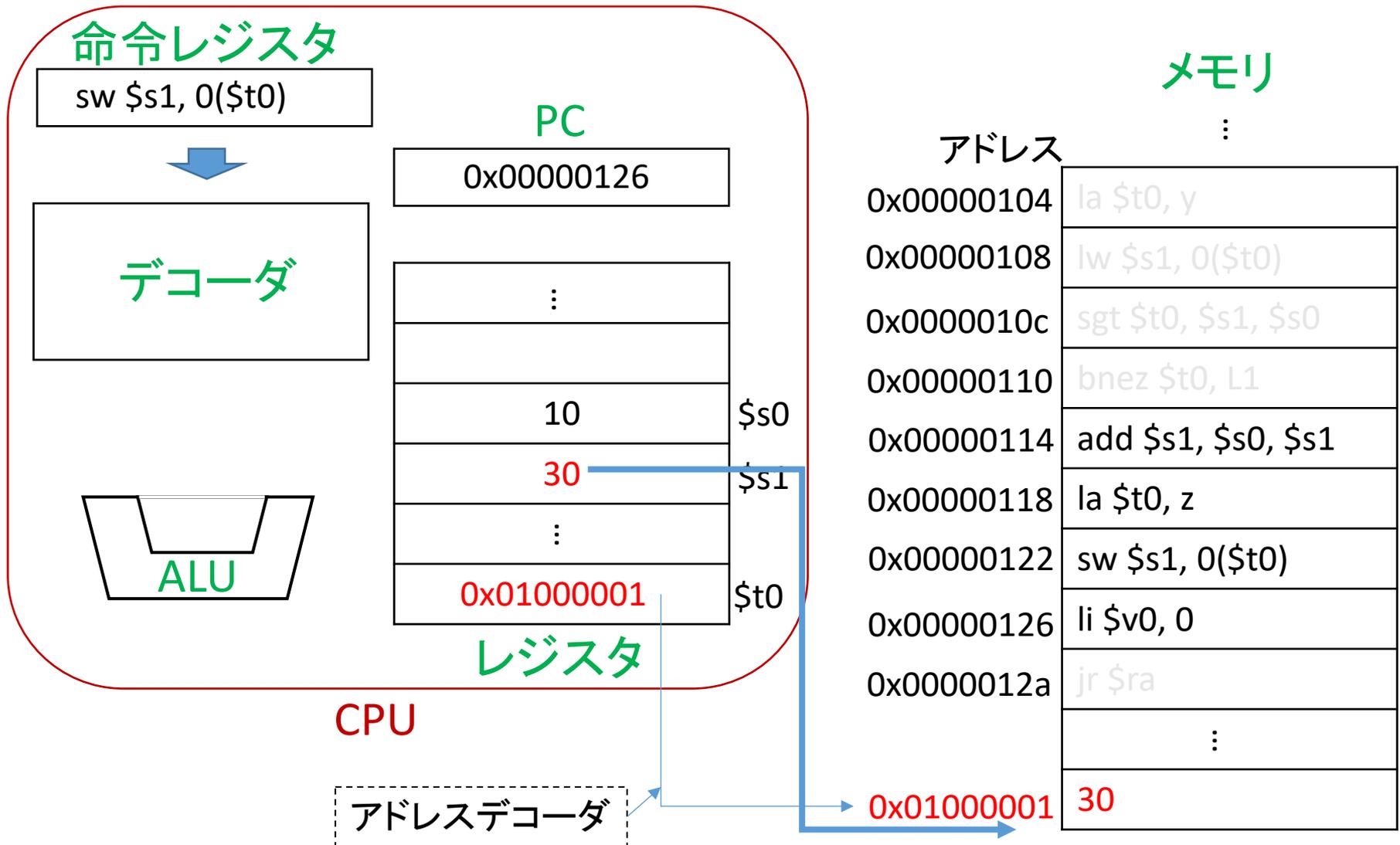
アドレス	メモリ
0x00000104	la \$t0, y
0x00000108	lw \$s1, 0(\$t0)
0x0000010c	sgt \$t0, \$s1, \$s0
0x00000110	bnez \$t0, L1
0x00000114	add \$s1, \$s0, \$s1
0x00000118	la \$t0, z
0x00000122	sw \$s1, 0(\$t0)
0x00000126	li \$v0, 0
0x0000012a	jr \$ra
0x01000001	

処理の流れ



メモリ	
アドレス	
0x00000104	la \$t0, y
0x00000108	lw \$s1, 0(\$t0)
0x0000010c	sgt \$t0, \$s1, \$s0
0x00000110	bnez \$t0, L1
0x00000114	add \$s1, \$s0, \$s1
0x00000118	la \$t0, z
0x00000122	sw \$s1, 0(\$t0)
0x00000126	li \$v0, 0
0x0000012a	jr \$ra
	⋮
0x01000001	

処理の流れ



演習06-1

- 「処理の流れ」に関する問題
- manaba, 10分

演習06-2

今回で習った内容を確認するためのyes, no問題

1. CPUとコンピュータは同義語である
2. ...

manaba, 10分。8点を4点に換算

本授業での問い

1. CPUの構成要素であるレジスタ・プログラムカウンタ・デコーダ・演算装置はどのようにハードウェア的に実現されているのか
 - 論理回路の話
2. CPUは命令に従って処理するというが、そもそも、命令ってどんなもの？だいたいどんな命令があるの？コンピュータの中はすべて0, 1表現と聞いているけど、`add $1, $0, $s1`は何？
 - 命令セット、機械語・アセンブリ言語の話