

CISCとRISC



いろいろなCPUの中でどんな命令
があるのか？



いろいろなCPUの中でどんな命令？

>前に見たのは COMET II

>周りのパソコンで使われる
Core 3i/5i/7i など (通称 x86系)

>サーバーで使われるPower CPU

>スマホでよく使われるARM CPU

2

いろいろなCPUの中でどんな命令？

命令体系の比較

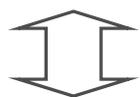
⇒ どんな組合せになっているか

3

いろいろなCPUの中でどんな命令？

命令体系の比較

RISC ⇒ 簡単な命令、簡単な体系
複雑なことを命令の組合せで

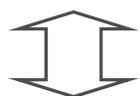


CISC ⇒ 複雑な命令、複雑な体系
複雑なことを1つの命令で

いろいろなCPUの中でどんな命令？

命令体系の比較

RISC ⇒ 簡単な命令、簡単な体系
複雑なことを命令の組合せで



Reduced Instruction Set Computer

CISC ⇒ 複雑な命令、複雑な体系
複雑なことを1つの命令で

Complex Instruction Set Computer

歴史的背景（脱線）

メインフレーム時代

⇒ ハードが機能追加され複雑化（命令も複雑化）



歴史的背景（脱線）

メインフレーム時代

⇒ ハードが機能追加され複雑化（命令も複雑化）



LSI (= マイクロプロセッサ) 時代の幕開け (1972)

⇒ 集積度低く、1チップにたくさんは載らない

⇒ 複雑なCPUを多チップに分割するより
簡単なCPUを1チップに載せた方が早い



歴史的背景（脱線）

メインフレーム時代

⇒ ハードが機能追加され複雑化（命令も複雑化）



LSI (= マイクロプロセッサ) 時代の幕開け (1972)

⇒ 集積度低く、1チップにたくさんは載らない

⇒ 複雑なCPUを多チップに分割するより

簡単なCPUを1チップに載せた方が早い



簡単なCPU (RISC) にして、クロックを早くする



歴史的背景（脱線）

メインフレーム時代

⇒ ハードが機能追加され複雑化（命令も複雑化）



LSI (= マイクロプロセッサ) 時代の幕開け (1972)

⇒ 集積度低く、1チップにたくさんは載らない

⇒ 複雑なCPUを多チップに分割するより

簡単なCPUを1チップに載せた方が早い



簡単なCPU (RISC) にして、クロックを早くする



今は
細線化により複雑なCPUでも楽に載るようになった

技術的背景（脱線）

RISCは簡単な命令を複数個組合せて
1つのCISC命令に相当することを行う

命令の組み合わせ方、レジスタの使い方などで
コンパイラを工夫する余地が大きかった
(工夫を競う風土が研究者に合った)

RISCの特徴

RISCの特徴

1つ1つの命令の動作は簡単

1つの命令でいろいろなことをするのではない

例： 演算はレジスタ・レジスタ間だけ
(演算命令でメモリをアクセスしない)

RISCの特徴

1つ1つの命令の動作は簡単

1つの命令でいろいろなことをするのではない

例： 演算はレジスタ・レジスタ間だけ
(演算命令でメモリをアクセスしない)

(その代り) レジスタ数を多くする

なるべくレジスタにデータを置いておく

RISCの特徴

1つ1つの命令の動作は簡単

1つの命令でいろいろなことをするのではない

例： 演算はレジスタ・レジスタ間だけ
(演算命令でメモリをアクセスしない)

(その代り) レジスタ数を多くする

なるべくレジスタにデータを置いておく

命令あたりのクロック数を少なくする

RISCの特徴

1つ1つの命令の動作は簡単

1つの命令でいろいろなことをするのではない

例： 演算はレジスタ・レジスタ間だけ
(演算命令でメモリをアクセスしない)

(その代り) レジスタ数を多くする

なるべくレジスタにデータを置いておく

命令あたりのクロック数を少なくする

命令の長さを固定長にする 回路簡単化・高速化

CISC/RISCの例

CISC 代表例はIntelのx86系プロセッサ
8086に始まりPentiumを経て
現在はCore2 7i/5i/3i、サーバー用xeon

16



CISC/RISCの例

CISC 代表例はIntelのx86系プロセッサ
8086に始まりPentiumを経て
現在はCore2 7i/5i/3i、サーバー用xeon
昔の代表：IBMのメインフレーム(360, 370)

17



CISC/RISCの例

CISC 代表例はIntelのx86系プロセッサ
8086に始まりPentiumを経て
現在はCore2 7i/5i/3i、サーバー用xeon
昔の代表：IBMのメインフレーム(360, 370)

RISC 代表例はPowerPC、ARM、MIPSなど

18



CISC/RISCの例

CISC 代表例はIntelのx86系プロセッサ
8086に始まりPentiumを経て
現在はCore2 7i/5i/3i、サーバー用xeon
昔の代表：IBMのメインフレーム(360, 370)

RISC 代表例はPowerPC、ARM、MIPSなど

ビジネス的に数の論理が強く、Intelとその互換機
が圧倒的に多い（必ずしも技術的優位ではない）

19



RISCとCISCの比較

| | RISC | CISC |
|-----------|------|------|
| 回路の複雑さ | | |
| 個々の命令の動作 | | |
| 命令当りクロック | | |
| 命令の長さ | | |
| 命令の種類 | | |
| レジスタ数 | | |
| プログラムの命令数 | | |

RISCとCISCの比較

| | RISC | CISC |
|-----------|------|------|
| 回路の複雑さ | 単純 | 複雑 |
| 個々の命令の動作 | | |
| 命令当りクロック | | |
| 命令の長さ | | |
| 命令の種類 | | |
| レジスタ数 | | |
| プログラムの命令数 | | |

RISCとCISCの比較

| | RISC | CISC |
|-----------|------|------|
| 回路の複雑さ | 単純 | 複雑 |
| 個々の命令の動作 | 簡単 | いろいろ |
| 命令当りクロック | | |
| 命令の長さ | | |
| 命令の種類 | | |
| レジスタ数 | | |
| プログラムの命令数 | | |

RISCとCISCの比較

| | RISC | CISC |
|-----------|------|------|
| 回路の複雑さ | 単純 | 複雑 |
| 個々の命令の動作 | 簡単 | いろいろ |
| 命令当りクロック | 少ない | いろいろ |
| 命令の長さ | | |
| 命令の種類 | | |
| レジスタ数 | | |
| プログラムの命令数 | | |

RISCとCISCの比較

| | RISC | CISC |
|-----------|------|------|
| 回路の複雑さ | 単純 | 複雑 |
| 個々の命令の動作 | 簡単 | いろいろ |
| 命令当りクロック | 少ない | いろいろ |
| 命令の長さ | 固定長 | 可変長 |
| 命令の種類 | | |
| レジスタ数 | | |
| プログラムの命令数 | | |

RISCとCISCの比較

| | RISC | CISC |
|-----------|------|------|
| 回路の複雑さ | 単純 | 複雑 |
| 個々の命令の動作 | 簡単 | いろいろ |
| 命令当りクロック | 少ない | いろいろ |
| 命令の長さ | 固定長 | 可変長 |
| 命令の種類 | 少ない | 多い |
| レジスタ数 | | |
| プログラムの命令数 | | |

RISCとCISCの比較

| | RISC | CISC |
|-----------|------|------|
| 回路の複雑さ | 単純 | 複雑 |
| 個々の命令の動作 | 簡単 | いろいろ |
| 命令当りクロック | 少ない | いろいろ |
| 命令の長さ | 固定長 | 可変長 |
| 命令の種類 | 少ない | 多い |
| レジスタ数 | 多い | 少ない |
| プログラムの命令数 | | |

RISCとCISCの比較

| | RISC | CISC |
|-----------|------|------|
| 回路の複雑さ | 単純 | 複雑 |
| 個々の命令の動作 | 簡単 | いろいろ |
| 命令当りクロック | 少ない | いろいろ |
| 命令の長さ | 固定長 | 可変長 |
| 命令の種類 | 少ない | 多い |
| レジスタ数 | 多い | 少ない |
| プログラムの命令数 | 多くなる | 少ない |

CISC vs RISC 論争の結果

一方の絶対優位はない

短い命令を多数使うか、長い命令を少数か
CISCの方が命令読出し回数は減るが、
命令種類が多く解読に時間がかかる

28

CISC vs RISC 論争の結果

一方の絶対優位はない

短い命令を多数使うか、長い命令を少数か
CISCの方が命令読出し回数は減るが、
命令種類が多く解読に時間がかかる

チップ複雑度の上限制約はどんどん緩和し
CISCに対する制限ではなくなった

29

CISC vs RISC 論争の結果

一方の絶対優位はない

短い命令を多数使うか、長い命令を少数か
CISCの方が命令読出し回数は減るが、
命令種類が多く解読に時間がかかる

チップ複雑度の上限制約はどんどん緩和し
CISCに対する制限ではなくなった

現在は、CISC命令セットの一部をRISCで
実現するなどの手法が取られる

30



CISC・RISCのまとめ

RISCとは

がなCPU ('80頃に新提案)

⇒ 単純化 何が?

⇒ 化 何が?

⇒ 代償

CISCとは

命令体系がな(従来の)CPU

比較して、

31



CISC・RISCのまとめ

RISCとは

命令体系が簡単なCPU ('80頃に新提案)

⇒ 単純化 何が?

⇒ 化 何が?

⇒ 代償

CISCとは

命令体系がな(従来の)CPU

比較して、



CISC・RISCのまとめ

RISCとは

命令体系が簡単なCPU ('80頃に新提案)

⇒ 単純化 何が? 命令の内容、命令の数

⇒ 化 何が?

⇒ 代償

CISCとは

命令体系がな(従来の)CPU

比較して、



CISC・RISCのまとめ

RISCとは

命令体系が簡単なCPU ('80頃に新提案)

⇒ 単純化 何が? 命令の内容、命令の数

⇒ 高速化 何が?

⇒ 代償

CISCとは

命令体系がな(従来の)CPU

比較して、



CISC・RISCのまとめ

RISCとは

命令体系が簡単なCPU ('80頃に新提案)

⇒ 単純化 何が? 命令の内容、命令の数

⇒ 高速化 何が? 命令の実行、クロック

⇒ 代償

CISCとは

命令体系がな(従来の)CPU

比較して、



CISC・RISCのまとめ

RISCとは

命令体系が簡単なCPU ('80頃に新提案)

⇒ 単純化 何が？ 命令の内容、命令の数

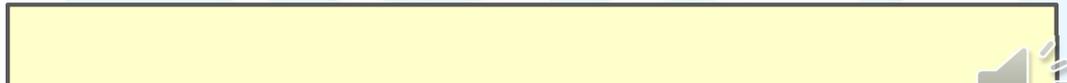
⇒ 高速化 何が？ 命令の実行、クロック

⇒ 代償 プログラムの命令数が増える

CISCとは

命令体系が□な(従来の)CPU

比較して、



CISC・RISCのまとめ

RISCとは

命令体系が簡単なCPU ('80頃に新提案)

⇒ 単純化 何が？ 命令の内容、命令の数

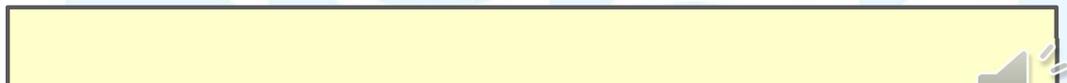
⇒ 高速化 何が？ 命令の実行、クロック

⇒ 代償 プログラムの命令数が増える

CISCとは

命令体系が複雑な(従来の)CPU

比較して、



CISC・RISCのまとめ

RISCとは

命令体系が簡単なCPU ('80頃に新提案)

⇒ 単純化 何が？ 命令の内容、命令の数

⇒ 高速化 何が？ 命令の実行、クロック

⇒ 代償 プログラムの命令数が増える

CISCとは

命令体系が複雑な(従来の)CPU

比較して、一方が優位ということはない



東邦大学

38

CISCとRISCの考え方について
分かりましたか？



↓
次へ

(ハーバードアーキテクチャ)



東邦大学

39