

計算機アーキテクチャ特論 (Advanced Computer Architectures)

7. 命令フェッチと分岐予測

1

TOP500 List - November 2010 (1-100)

R_{max} and R_{peak} values are in TFlops. For more details about other fields, check the TOP500 description.
Power data in kW for entire system

Rank	Site	Computer/Year	Vendor	Cores	R_{max}	R_{peak}	Power
1	National Supercomputing Center in Tianjin, China	Tianhe-1A - NJDT TH MPP	X5670 2.93GHz / SC, NVIDIA GPU, FT-1000 GC / 2010	186368	2566.00	4701.00	4040.00
2	DOE/SC/Oak Ridge National Laboratory, United States	Jaguar - Cray XT5-HE	Opteron 6-core 2.6 GHz / 2009	224152	1759.00	2331.00	6950.60
3	National Supercomputing Center in Shenzhen (NSCSC), China	Nebulae - Dawning TC3600	Blade, Intel X5650, Nvidia Tesla C2050 GPU / 2010	120640	1271.00	2984.30	2580.00
4	GSIC Center, Tokyo Institute of Technology, Japan	TOSUBAME 2.0 - HP ProLiant SL390s G7	Xeon RC 55670, Nvidia GPU, Linux/Windows / 2010	73278	1192.00	2287.63	1398.61
5	DOE/SC/LANL/NERSC, United States	Hopper - Cray XE6	12-core 2.1 GHz / 2010	153408	1054.00	1288.63	2910.00
6	Commissariat à l'Energie Atomique (CEA), France	Tera-100 - Bull built super-nodes	392x1056030 / 2010	138368	1050.00	1254.55	4590.00
7	DOE/NSA/LANL/INL, United States	Roadrunner - BladeCenter Q620A, B21	Cluster, PowerPC Cell B1 3.2 GHz / Opteron GC 1.8 GHz, Virtuale Infiniband / 2009	122400	1042.00	1375.78	2345.50
8	National Institute for Computational Sciences, University of Tennessee, United States	Krahen XT5 - Cray XT5-HE	Opteron 6-core 2.6 GHz / 2009	98928	831.70	1028.85	3090.00
9	Forschungszentrum Juelich (FZJ), Germany	JUGENE - Blue Gene/P	Solution / 2009	294912	625.50	1002.70	2298.00
10	DOE/NSA/LANL/INL, United States	Cielo - Cray XE6	6-core 2.4 GHz / 2010	107152	816.60	1028.66	2950.00

Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

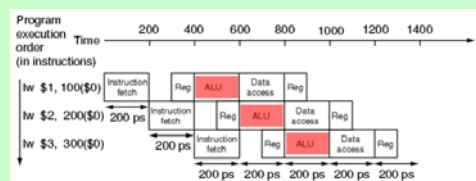
Gamma Separated Value

Green500 Rank	MFLOPS/W	Site*	Computer*	Total Power (kW)
1	1684.20	IBM Thomas J. Watson Research Center	NHSA/SC Blue Gene/Q Prototype	38.80
2	958.35	GSIC Center, Tokyo Institute of Technology	HP ProLiant SL390s G7 Xeon 6C X5670, Nvidia GPU, Linux/Windows	1243.80
3	933.06	NCSA	Hybrid Cluster Core i3 2.93GHz Dual Core, NVIDIA C2050, Infiniband	36.00
4	828.67	RIKEN Advanced Institute for Computational Science	K computer, SPARC64 VIIIfx 2.0GHz, Tofu Interconnect	57.96
5	773.38	Forschungszentrum Juelich (FZJ)	QPACE SFB TR Cluster, PowerPC Cell B1, 3.2 GHz, 3D-Torus	57.54
6	773.38	Universitaet Regensburg	QPACE SFB TR Cluster, PowerPC Cell B1, 3.2 GHz, 3D-Torus	57.54
7	773.38	Universitaet Wuppertal	QPACE SFB TR Cluster, PowerPC Cell B1, 3.2 GHz, 3D-Torus	57.54
8	740.78	Universitaet Frankfurt	Supermicro Cluster, QC Opteron 2.1 GHz, ATI Radeon GPU, Infiniband	385.00
9	677.12	Georgia Institute of Technology	HP ProLiant SL390s G7 Xeon 6C X5660 2.8GHz, Nvidia Fermi, Infiniband QDR	94.40
10	636.36	National Institute for Environmental Studies	GOSAT Research Computation Facility, nvidia	117.15

Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

スカルプロセッサ

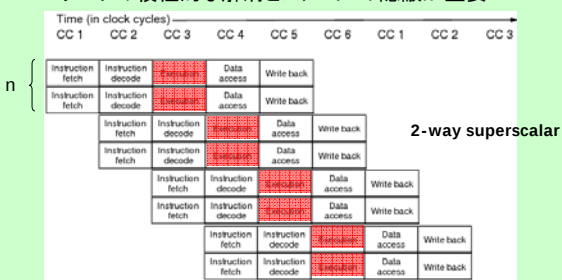
- サイクル当たりの実行命令数
Executed Instructions per Cycle (IPC)
- IPCが1を超えることはできない。



Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

スーパースカラ・プロセッサと命令レベル並列性

- 複数のパイプラインを利用して IPC を 1以上に引き上げる
 - n-way スーパースカラ
- ハザードの積極的な解消とストールの隠蔽が重要



Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

データ依存関係 (data dependence)

- 真のデータ依存 (true data dependence)
 - RAW, read after write
 - 出力依存 (output dependence)
 - WAW, write after write
 - 逆依存 (antidependence)
 - WAR, write after read
 - RAR ?, read after read
- 偽のデータ依存 (False data dependence) is indicated for WAW, WAR, and RAR.

Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

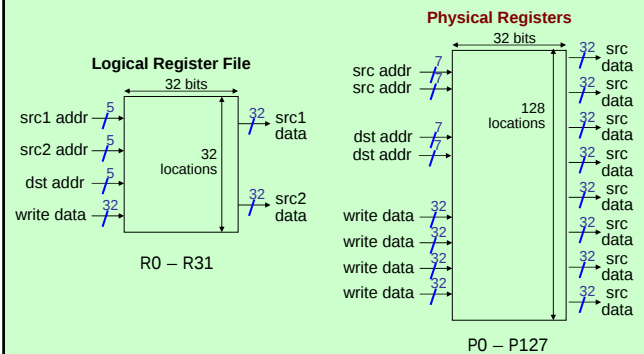
ハードウェアによるレジスタ名前換え

- 論理レジスタ (logical register)
 - プログラマやコンパイラから見えるレジスタ
 - 機械命令のフィールドで指定
 - MIPSの命令セットでは R0 – R31 という論理レジスタを利用
- 物理レジスタ (physical register)
 - プロセッサアーキテクチャから見えるレジスタ
 - プログラマやコンパイラから陽に見える必要はない。
 - 物理的に存在するレジスタ

ソフトウェアによるレジスタ名前換え

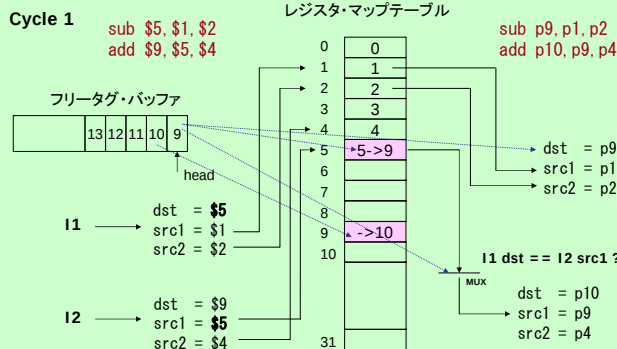
Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

MIPS Register File



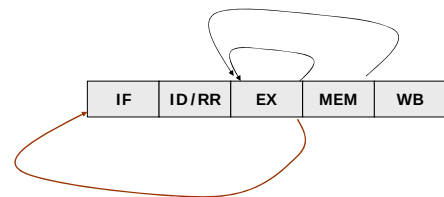
Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

2命令のレジスタ・リネーミング



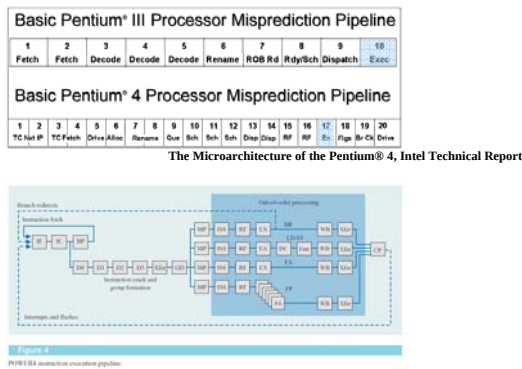
Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

スカラー・プロセッサ、パイプラインと制約ループ



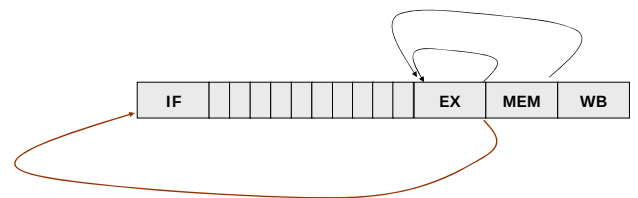
Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

プロセッサの命令パイプラインの例



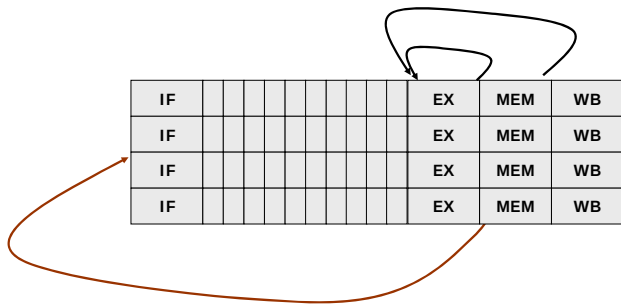
Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

スカラー・プロセッサ(パイプラインの長い、スーパーパイプライン), パイプラインと制約ループ



Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

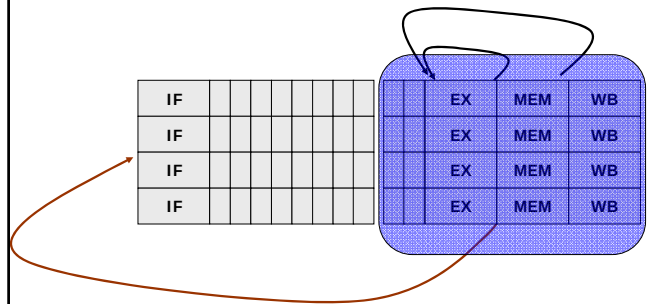
スーパースカラプロセッサ(パイプラインの長い, スーパーパイプライン), パイプラインと制約ループ



Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

13

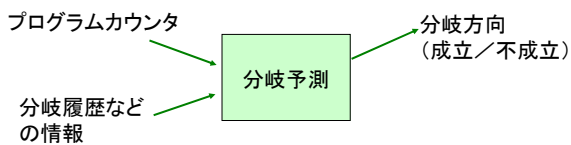
アウトオブオーダー実行のスーパースカラプロセッサ



Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

14

分岐方向の予測(分岐予測)

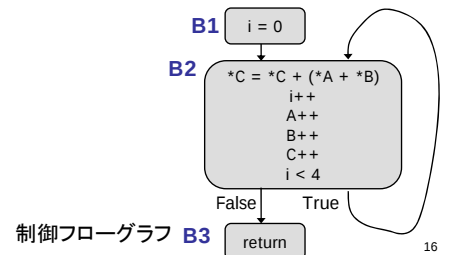


Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

15

サンプルプログラム Vector Add

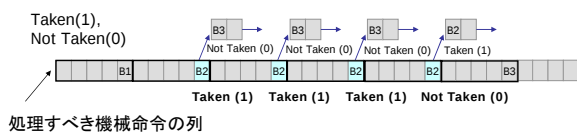
```
#define VSIZE 4
void vadd(long *A, long *B, long *C){
    for(i=0; i<VSIZE; i++){
        C[i] += (A[i] + B[i]);
    }
}
```



Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

16

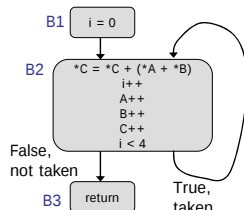
シンプルな分岐予測 Branch Always



処理すべき機械命令の列

Branch Always: 常に分岐が成立すると予測する。

- 上の例では, 予測成功率は 75%, ミス率 25%
- 予測のためのメモリを必要としない。
- 予測とよぶほどのものではない。

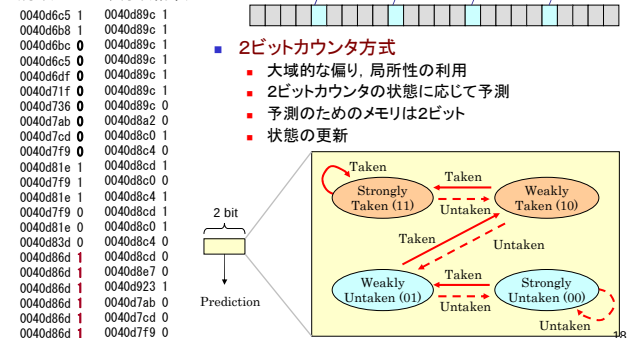


Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

17

シンプルな分岐予測 2ビットカウンタ方式

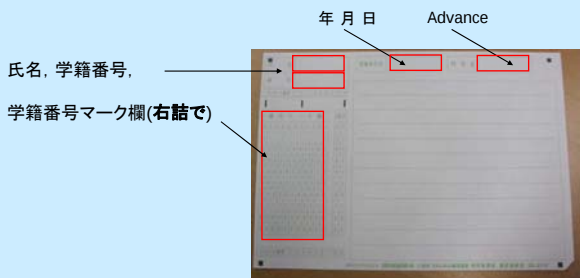
トレースデータ
(分岐アドレス, 分岐結果)



Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

18

Exercise



19

2ビットカウンタ方式の機能レベルの実装

```
class TwoBit {
    int cnt;
public:
    TwoBit(int);
    int predict();
    void update(int);
};

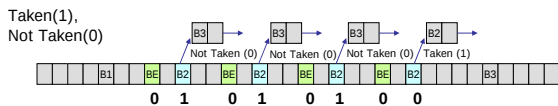
TwoBit::TwoBit() {
    cnt = 1;
}

int TwoBit::predict() {
    // ...
}

void TwoBit::update(int taken) {
    // ...
}
```

20

サンプルプログラム Vector Add

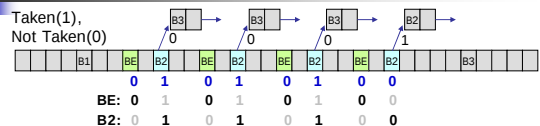


```
#define VSIZE 4
void vadd(long *A, long *B, long *C) {
    for(i=0; i<VSIZE; i++) {
        if(A[i]<0) error_routine();
        C[i] += (A[i] + B[i]);
    }
}
```

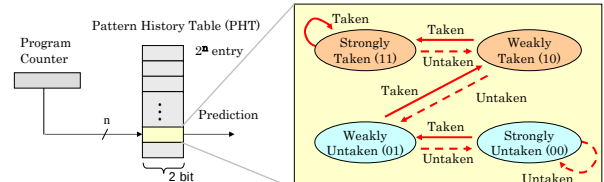
制御フローグラフ

21

Bimodal (ISCA 1981)

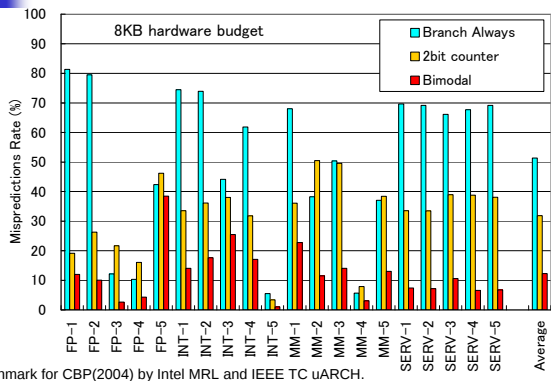


- 分岐アドレス(プログラムカウンタ)毎に履歴を切り替える
- 分岐アドレスによりパターン履歴表 (PHT) のインデックスを作成
- パターン履歴表は2ビットカウンタの配列



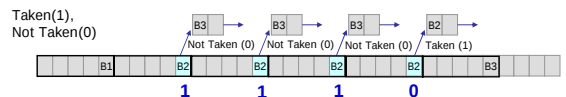
22

シンプルな分岐予測の予測精度(予測ミス率)



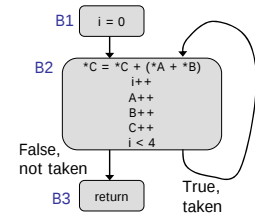
23

分岐履歴



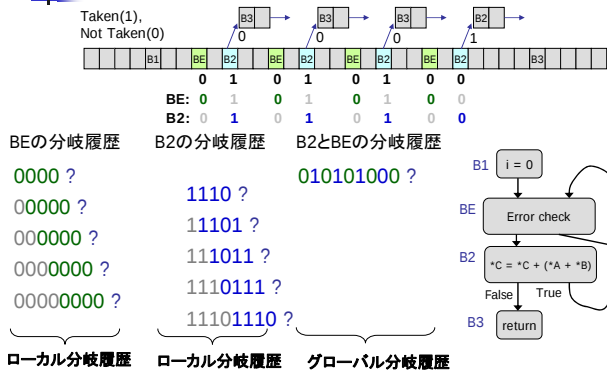
B2の分岐履歴

1110 ?
11101 ?
111011 ?
1110111 ?
11101110 ?



24

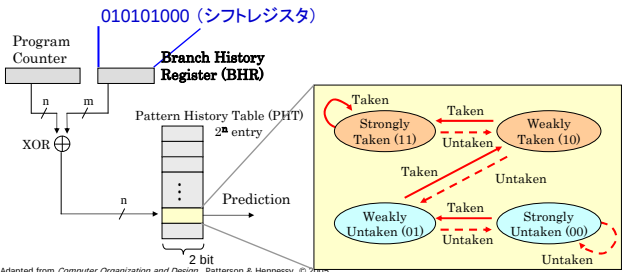
ローカル、グローバル分岐履歴



25

Gshare (TR-DEC 1993)

- グローバル分岐履歴と分岐アドレスとの排他的論理和によりパターン履歴表へのインデックスを作成
 - パターン履歴表は2ビット飽和型カウンタの配列で、選択された2ビットカウンタの値により分岐方向を予測 (bimodalと同じ)
 - 分岐結果を用いて、予測に利用したカウンタを更新



Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

Gshareの機能レベルの実装

```
typedef int data_t;

class Gshare {
    data_t bhr;
    data_t *buf;
public:
    int size;
    Gshare(int):
        int predict(data_t);
        void update(data_t, int);
};

Gshare::Gshare(int bpred_size) {
    size = bpred_size;
    buf = (data_t *)calloc(size, sizeof(data_t));
    for (int i=0; i<size; i++) buf[i] = 2;
}

int Gshare::predict(data_t pc) {
    // ...
}

void Gshare::update(data_t pc, int taken) {
    // ...
}
```

Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

27

Gshareの機能レベルの実装

```
typedef int data_t;

class Gshare {
    data_t bhr;
    data_t *buf;
public:
    int size;
    Gshare(int):
        int predict(data_t);
        void update(data_t, int);
};

Gshare::Gshare(int bpred_size) {
    size = bpred_size;
    buf = (data_t *)calloc(size, sizeof(data_t));
    for (int i=0; i<size; i++) buf[i] = 2;
}

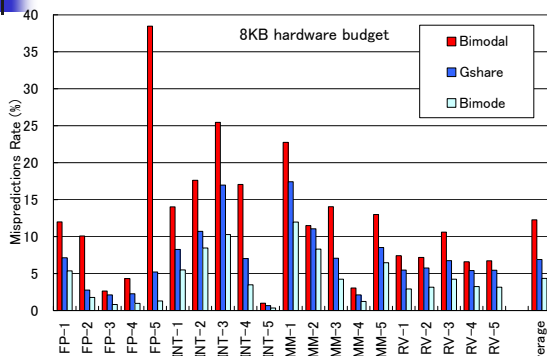
int Gshare::predict(data_t pc) {
    int index = ((pc >> 2) ^ bhr) % size;
    return (buf[index]>1);
}

void Gshare::update(data_t pc, int taken) {
    int index = ((pc >> 2) ^ bhr) % size;
    if (taken!=0 && buf[index]<2) buf[index]++;
    if (taken==0 && buf[index]>0) buf[index]--;
    bhr = (bhr << 1) | taken;
}
```

Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

28

予測精度(予測ミス率)



Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

29

アナウンス

- 講義スライド, 講義スケジュール
 - www.arch.cs.titech.ac.jp

Adapted from Computer Organization and Design, Patterson & Hennessy, © 2005

30