

COSC 6385

Computer Architecture

- Correlated Branch Predictors

Edgar Gabriel

Fall 2006



COSC 6385 – Computer Architecture
Edgar Gabriel

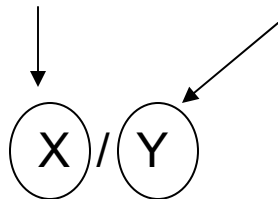


Correlated branches

- For a (1,1) predictor: each branch has two different branch prediction buffers:

Predictor used in case the previous branch in the application has not been taken

Predictor used in case the previous branch in the application has been taken



- The content of the two branch prediction buffers are determined by the branch to which they belong
- Which of the two branch prediction buffers are used is depending on the outcome of the previous branch in the application



Correlated branches - example

```

if ( d==0 )           BNEZ  R1, L1           !branch b1
    d = 1;           DADDIU R1, R0, #1
if ( d==1 )           L1:  DADDIU R3, R1, #-1
    ...              BNEZ  R3, L2           !branch b2
                    ...
                    L2:
    
```

Initial value of d	d==0?	b1	Value of d before b2	d==1?	b2
2	No	Taken	2	No	Taken
0	Yes	Not taken	1	Yes	Not taken
2	No	Taken	2	No	Taken
0	Yes	Not taken	1	Yes	Not taken



Correlated branches - example

d=?	BPB b1	b1 act.	BPB b2	B2 act.
2	NT/NT		NT/NT	

- the branch prediction buffers for the branches b1 and b2 are assumed to hold the prediction 'Not taken' for both option (previous branch not taken/taken)



Correlated branches - example

d=?	BPB b1	b1 act.	BPB b2	B2 act.
2	NT/NT		NT/NT	

- assuming BPB for b1 uses the 'Not Taken' predictor because the previous branch in the application has not been taken
→ BPB for b1 predicts that b1 will not be taken



Correlated branches - example

d=?	BPB b1	b1 act.	BPB b2	B2 act.
2	NT/NT	T	NT/NT	

- BPB for b1 predicts that b1 will not be taken
- b1 is taken (see table for d=2)

Initial value of d	d==0?	b1	Value of d before b2	d==1?	b2
2	No	Taken	2	No	Taken
0	Yes	Not taken	1	Yes	Not taken



Correlated branches - example

d=?	BPB b1	b1 act.	BPB b2	B2 act.
2	NT/NT	T	NT/NT	
	T/NT			

→ updating the 'Previous branch has not been taken' part of BPB for b1 to Taken

→ because b1 has been taken, the 'last branch has been taken' part of BPB b2 will be used

→ BPB b2 predicts, that b2 will not be taken



Correlated branches - example

d=?	BPB b1	b1 act.	BPB b2	B2 act.
2	NT/NT	T	NT/NT	T
	T/NT		NT/T	

- b2 is taken (see table for d=2)
- updating the 'Previous branch has been taken' part of BPB for b2 to Taken
- because b2 has been taken, the 'last branch has been taken' part of BPB b1 will be used
- BPB b1 predicts, that b1 will not be taken

Initial value of d	d==0?	b1	Value of d before b2	d==1?	b2
2	No	Taken	2	No	Taken
0	Yes	Not taken	1	Yes	Not taken



Correlated branches - example

d=?	BPB b1	b1 act.	BPB b2	B2 act.
2	NT/NT	T	NT/NT	T
0	T/NT	NT	NT/T	

- b1 is not taken (see table for d=0) → matches prediction!
- update of BPB b1 does not modify any entry taken
- because b1 has not been taken, the 'last branch has not been taken' part of BPB b2 will be used
- BPB b2 predicts that b2 will not be taken

Initial value of d	d==0?	b1	Value of d before b2	d==1?	b2
2	No	Taken	2	No	Taken
0	Yes	Not taken	1	Yes	Not taken



Correlated branches

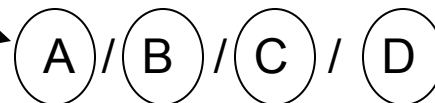
- A (2,1) correlated branch predictor
 - Uses the behavior of the last 2 branches to choose from 2^2 different predictions
 - Uses a 1 bit predictor for each of the 4 prediction buffers

Predictor used in case the previous 2 branches in the application have both not been taken (00)

Predictor used in case the previous branches have the history :second last branch not taken, last branch taken (01)

Predictor used in case the previous branches have the history: second last branch taken, last branch not taken (10)

Predictor used in case the previous 2 branches in the application have both been taken (11)



Correlated branches

- How do we know which of the four sections of our branch predictor to use
 - Need to record the behavior of all branches in the application

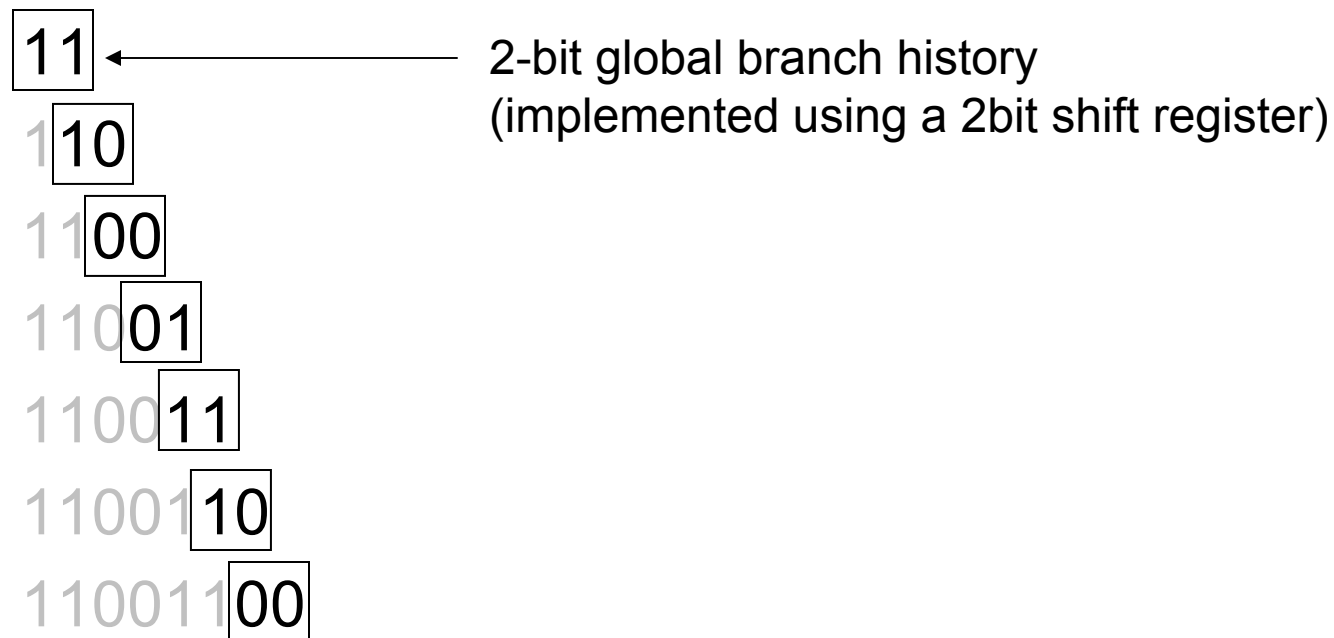
Initial value of d	d==0?	b1	Value of d before b2	d==1?	b2
2	No	Taken	2	No	Taken
0	Yes	Not taken	1	Yes	Not taken
2	No	Taken	2	No	Taken
0	Yes	Not taken	1	Yes	Not taken

- e.g. 11001100110011



Correlated branches

- For a (2,n) branch predictor, the last two branches are relevant

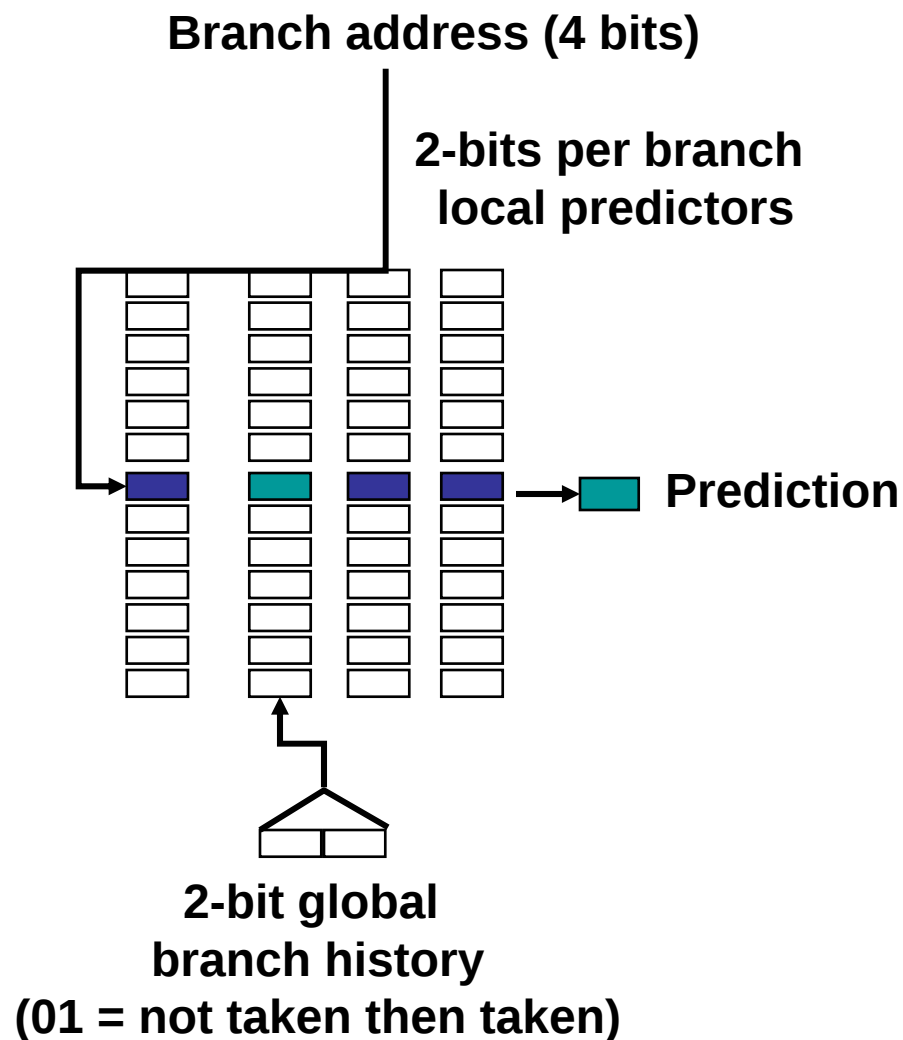


Correlating Branches

Idea: taken/not taken of recently executed branches is related to behavior of next branch (as well as the history of that branch behavior)

- Then behavior of recent branches selects between, say, 4 predictions of next branch, updating just that prediction

- (2,2) predictor: 2-bit global, 2-bit local



COSC 6385 – Computer Architecture
Edgar Gabriel

Slide based on a lecture by David A. Patterson,
University of California, Berkeley
<http://www.cs.berkeley.edu/~pattarn/252S01>

