COEN 313 HW 2

Branch Prediction

Assume a program hits a loop with the following structure

    top: ...

        ....

        nice branch

        ....

        naughty branch

        branch X

        branch back to top

1. All the branches in the loop are shown.

2. Branch X is the one we are interested in predicting correctly and it COMPLETELY depends on nice branch (i.e if you know nice branch outcome, then you know branch X outcome).

3. Naughty branch get taken or not taken randomly without any pattern.

4. the branch at the end of the loop is always taken (the loop is executed for a very large number of times).

 a. Show that a hybrid global branch predictor with a shift register size of 3 bits will predict branch X correctly after a few iterations.

	init	i = 0	i = 1	i = 2	i = 3	i = 4	i = 5	i = 6
	000	000	100	010	101	010	101	010
000	0	0	1	1	1	1	1	1
001	0	0	0	0	0	0	0	0
010	0	0	0	0	1	1	1	1
011	0	0	0	0	0	0	0	0
100	0	0	0	0	0	0	0	0
101	0	0	0	0	0	0	0	0

110	0	0	0	0	0	0	0	0
111	0	0	0	0	0	0	0	0
Actual	0	1	0	1	0	1	0	1
Prediction	0	0	0	0	0	1	0	1

b. What if there are N naughty branches preceding branch x in the loop code? Will a 3-bit shift register work? If not, then what shift register size do we need?

If there are N naughty branches preceding branch x in the loop code, a 3-bit shift register will no longer work (assuming N is a fairly large integer). Instead, we will need a shift register size increase relative to how large the given number is.