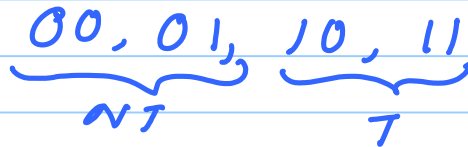


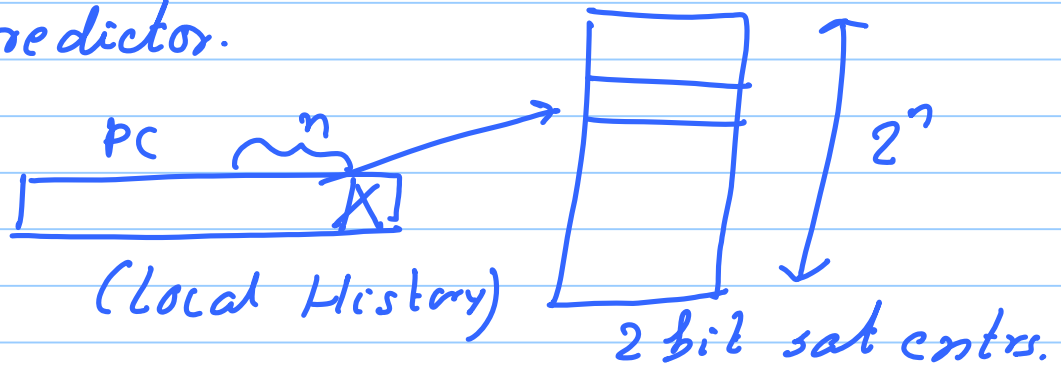
Oct 3rd

# Branch Predictors

2-bit saturating counter



1) Bimodal predictor.



Gag, Gap, Pag, Pap

Global History.

✓ if (flag == 1)  
flag1 = 2;

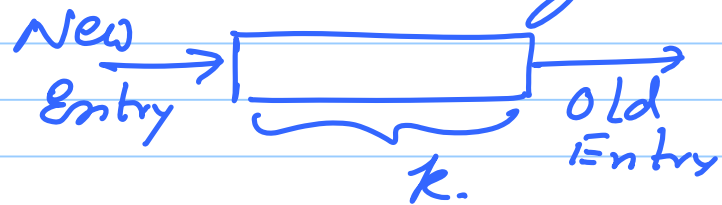
if (flag1 == 2)  
flag2 = 4;

Ex 1

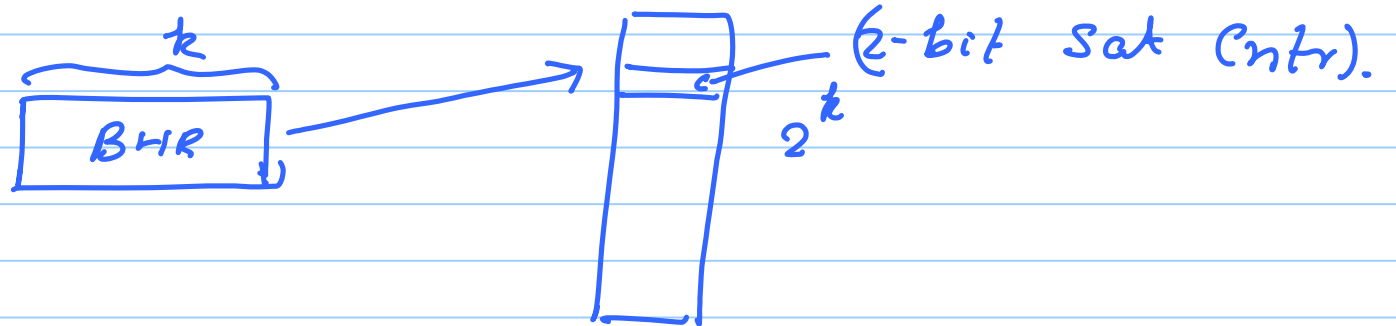
if (a == 0) {  
} else if (a == 1) {  
} else if (a == 2) { }  
}

Ex: 2

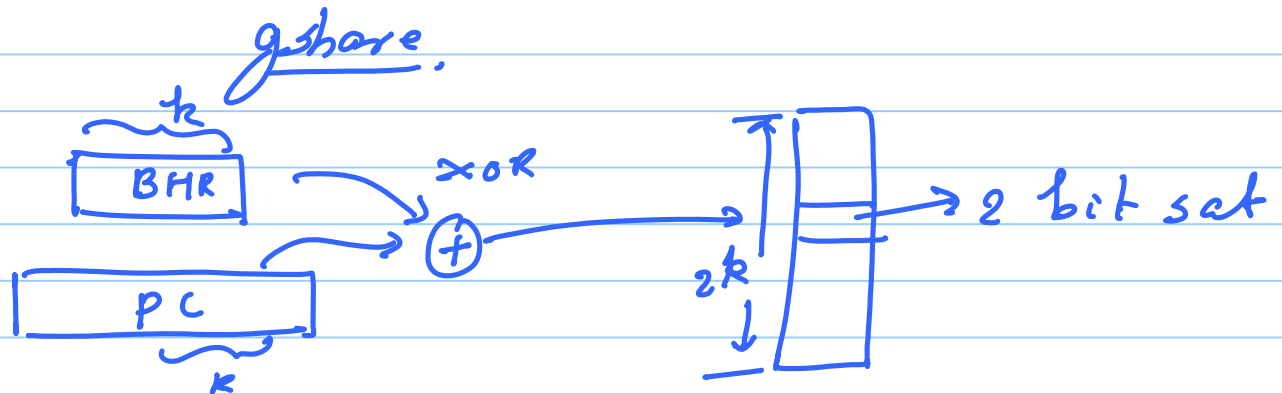
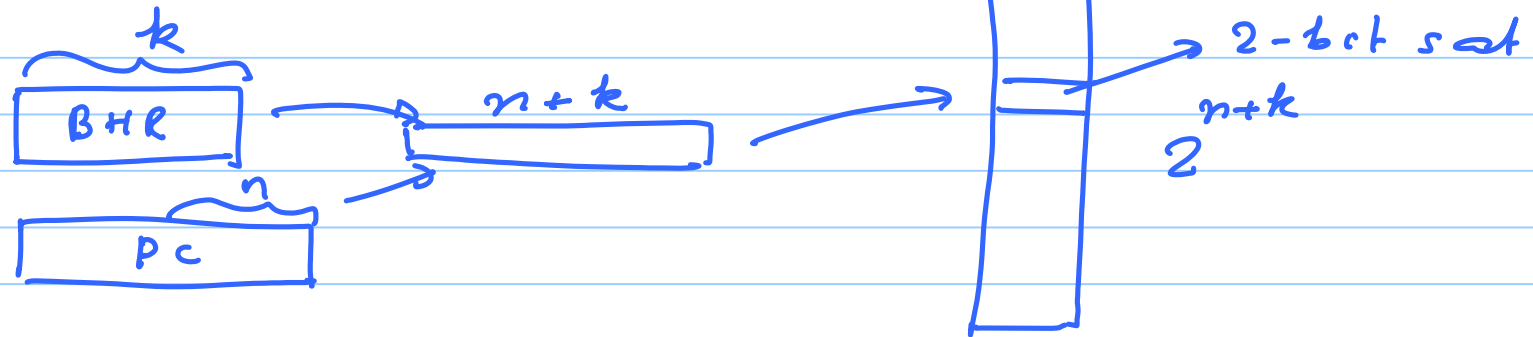
# Branch History Register.

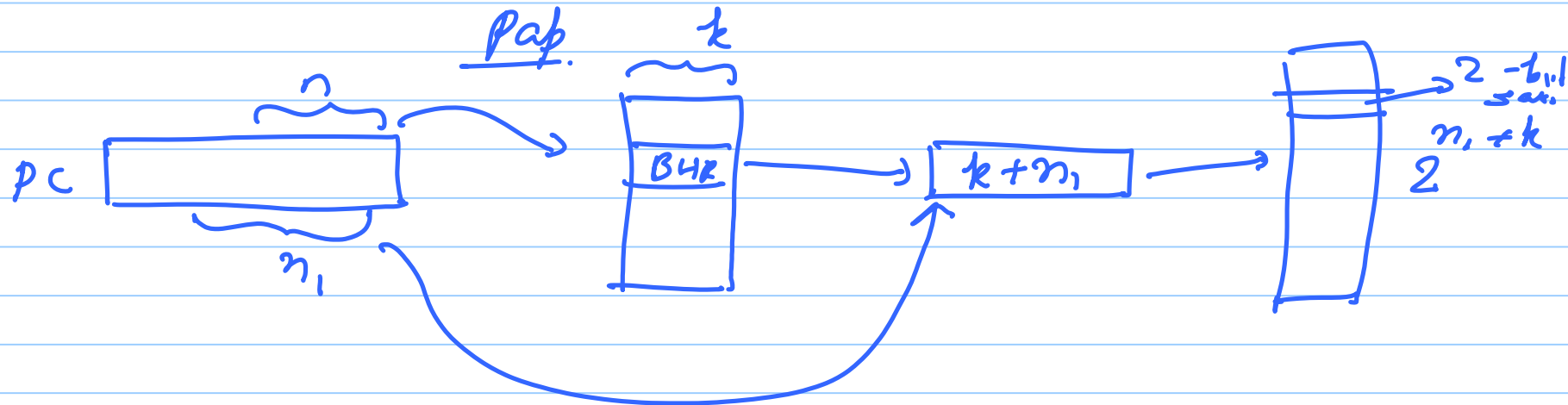
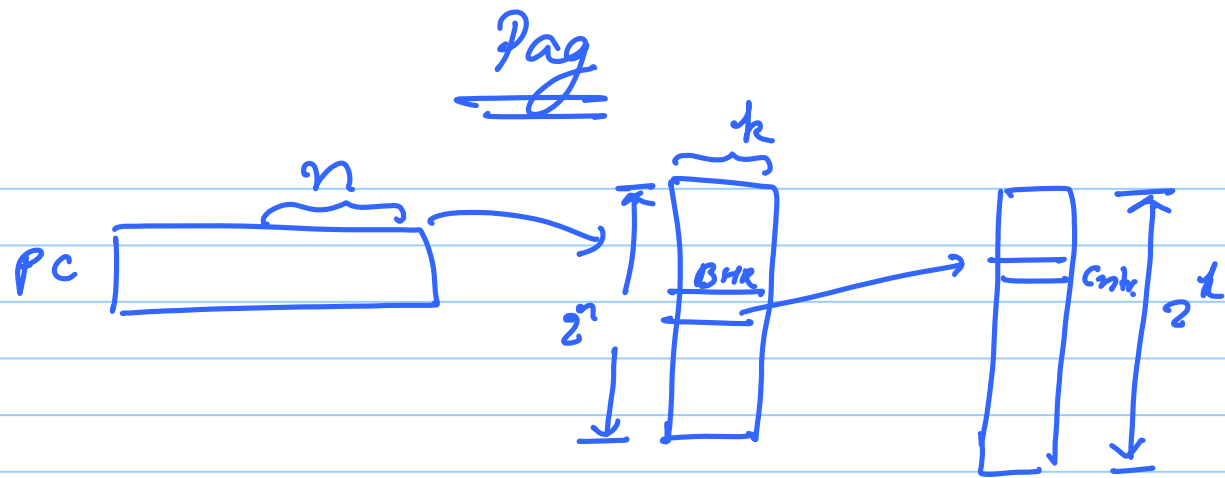


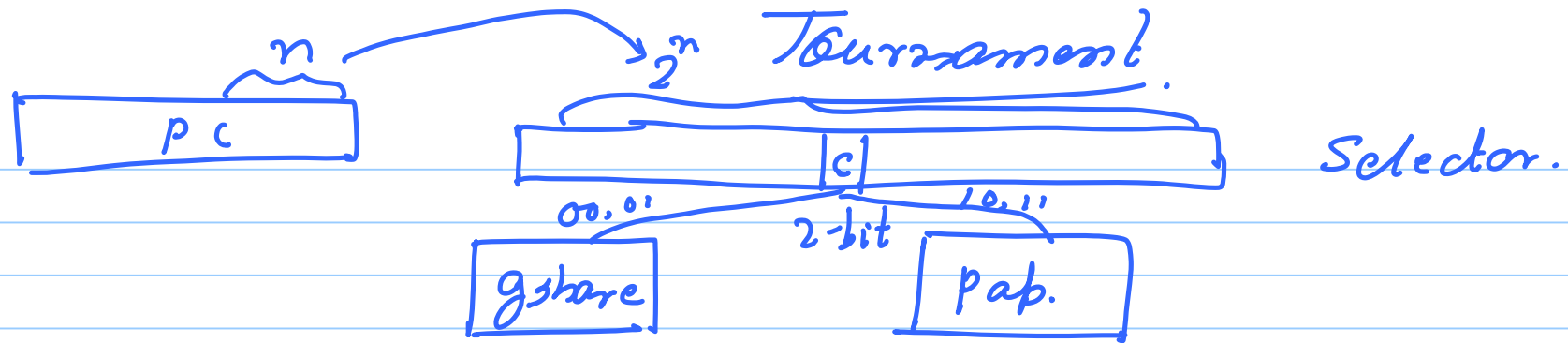
Gag.



Gap (gselect)







	Pred 1	Pred 2
$C$	✓	✓
$C--$	✓	✗
$C++$	✗	✓
$C$	✗	✗

Table 1

Prediction:

1) Find appropriate 2 bit contr.

2) value is  $c$

3) If  $c = [0, 1]$  choose outcome of Pred 1  
 $c = [2, 3]$  " " " " 2

Training 1) Train selector as per table 1  
2) Train both predictors.

$$\left\{ \begin{aligned} \text{CPI} &= \text{CPI}_{\text{ideal}} + \text{bmiss-rate} \times \text{penalty} \\ &= 1 + \frac{1}{5} \times \frac{1}{20} \times 50 \\ &= \underline{1.5} \end{aligned} \right.$$

For a CSL 211 processor.

$$\text{bmisspred-penalty} = 2$$

For a realistic CSL 718 processor.

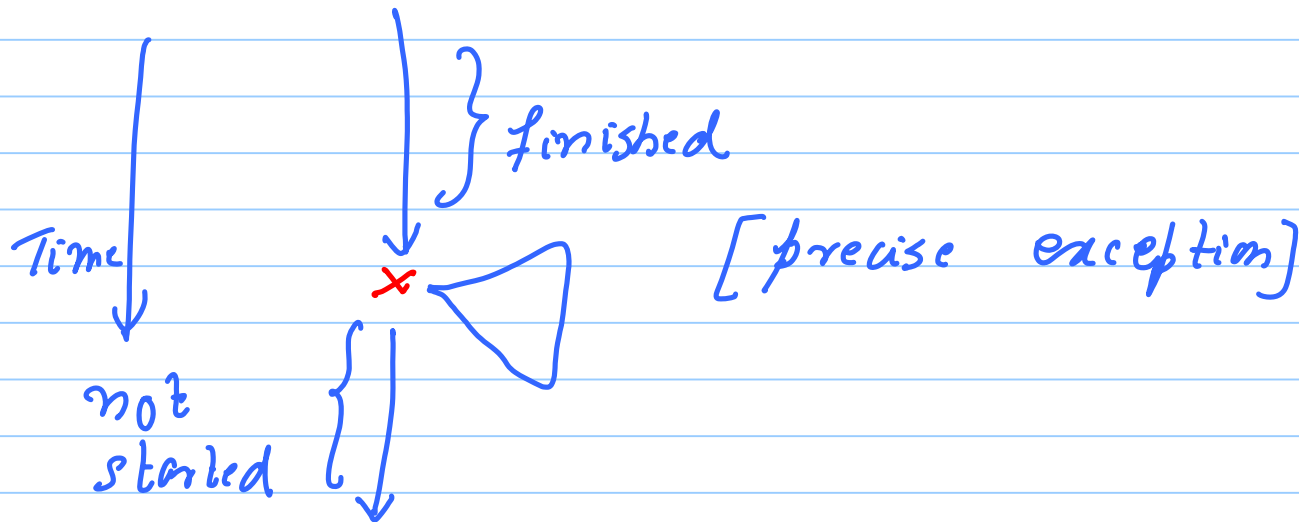
$$\text{bmisspred-penalty} = \underline{20-80},$$

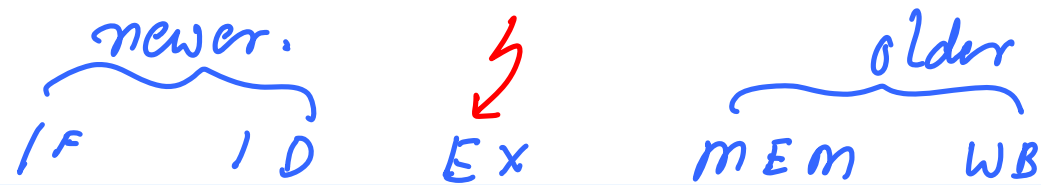


# Interrupts

Let's assume we have:

an arithmetic exception.





- 1) Discard newer instructions.
- 2) Stop fetch.
- 3) Complete older instructions.
- 4) Jump to exception handler.
- 5) You might desire to
  - a) Terminate prog.
  - OR b) Jump to exception-PC + 4