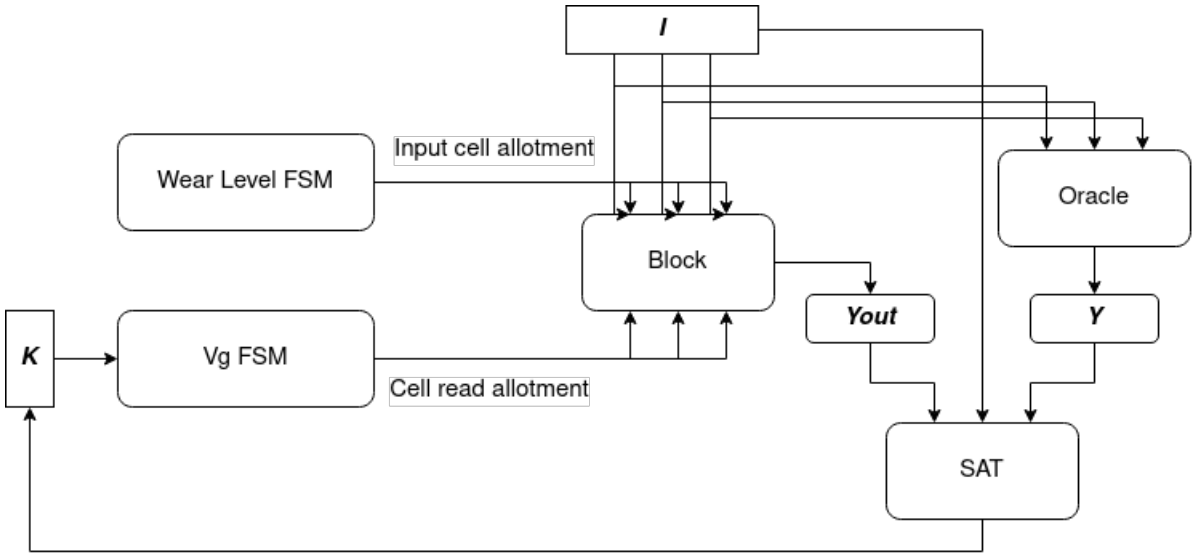
Week 11

Possible Attack Methods

- Assumptions during defense idea:
  - There exists some manner to apply Vg vector on a block, which gets decided by some state machine inside the memory controller.
  - Adversary won't have any idea about the state transition matrix of this state machine.
  - Adversary can use any number of blocks, and program them to any function he wants the memory to calculate.
- Any Flash memory is managed by wear techniques. The blocks are usually categorised either under dynamic or static wear technique.
- Dynamic means each physical block is mapped to a logical block address, and when data is written to that particular physical address, it is marked invalid and it's logical block address gets mapped to another block, via some state machine. This idea can be extended directly to page level addressing, leading to a method of alloting inputs of functions in Logic-in-memory technique. This would be using one-by-one allotment of input places, each corresponding to a transition.
- Static technique uses rotational allotment, using least used blocks, leading to better lifetimes. But due to rotational and predictable allotments, we won't be considering this method right now.
- The controller's wear levelling state machine should match with Vg state machine, which should give out one-by-one plane number as well.
- Thought experiment: Suppose someone other than the adversary is programming functions in blocks, while adversary has control over getting output by applying Vg values at planes of the block. He wants to know the correct output of the functions everytime, which can only be achieved by applying correct Vg vector sequences on the planes, which in turn can be done by initialising the state machine correctly.
- The key difference between a usual IP, and a "circuit" made here, is that output depends on an external sequential circuit. This leads to the whole CNF format needing the sequential form to be included.
- Therefore SAT boils down to the same algorithm:

Week 11


I
Wear Level FSM
Input cell allotment
Oracle
Block
Yout
Y
K
Vg FSM
Cell read allotment
SAT