

Context selection complexity in HEVC CABAC

(JCTV-D244/m19005)

Vivienne Sze

Texas Instruments Inc.

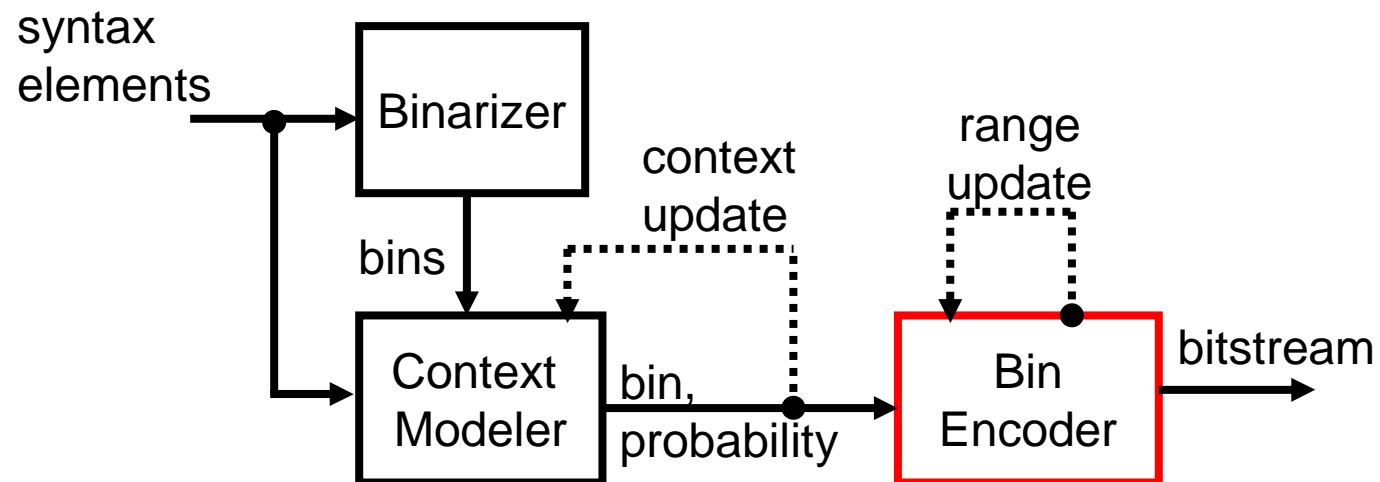
**Joint Collaborative Team on Video Coding (JCT-VC)
of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11**

4th Meeting: Daegu, KR, 20-28 January, 2011

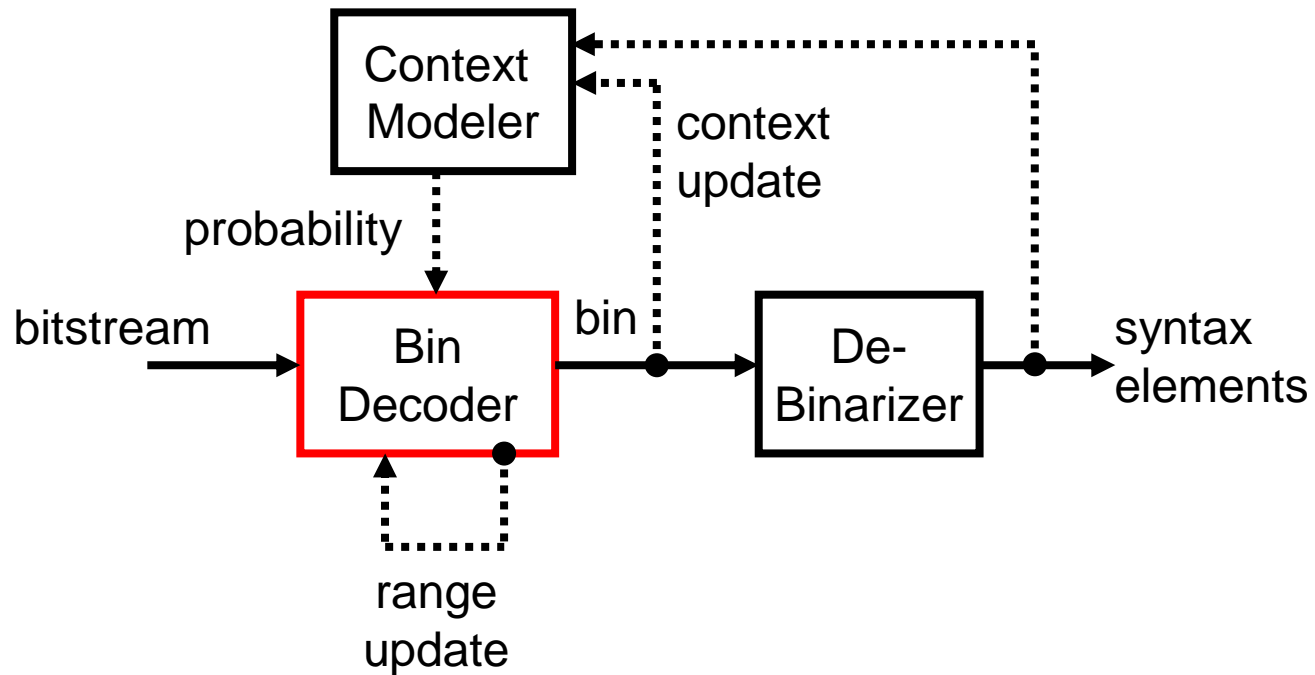
Serial Dependencies in CABAC

- CABAC is a key bottleneck in video codec implementations
- Difficult to increase throughput due to serial dependencies from multiple feedback loops
- Impact of feedback loops on implementation
 - Parallelism and pipelining common techniques used to increase throughput
 - Parallelism (increase bins per cycle)
 - Pipelining (reduce cycle time)
 - Speculative computations required due to feedback loops

Feedback Loops at Encoder



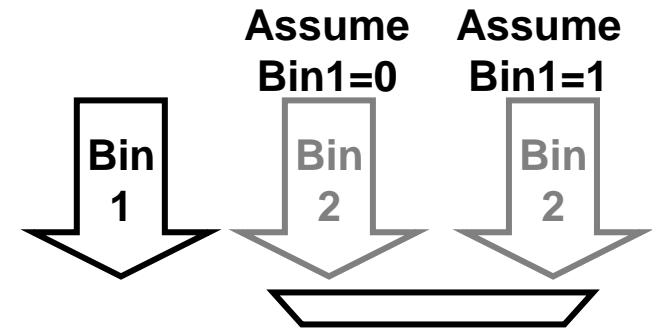
Feedback Loops in Decoder



Note that loops pass through multiple modules (e.g. context update loop goes through context modeler *and* bin decoder)

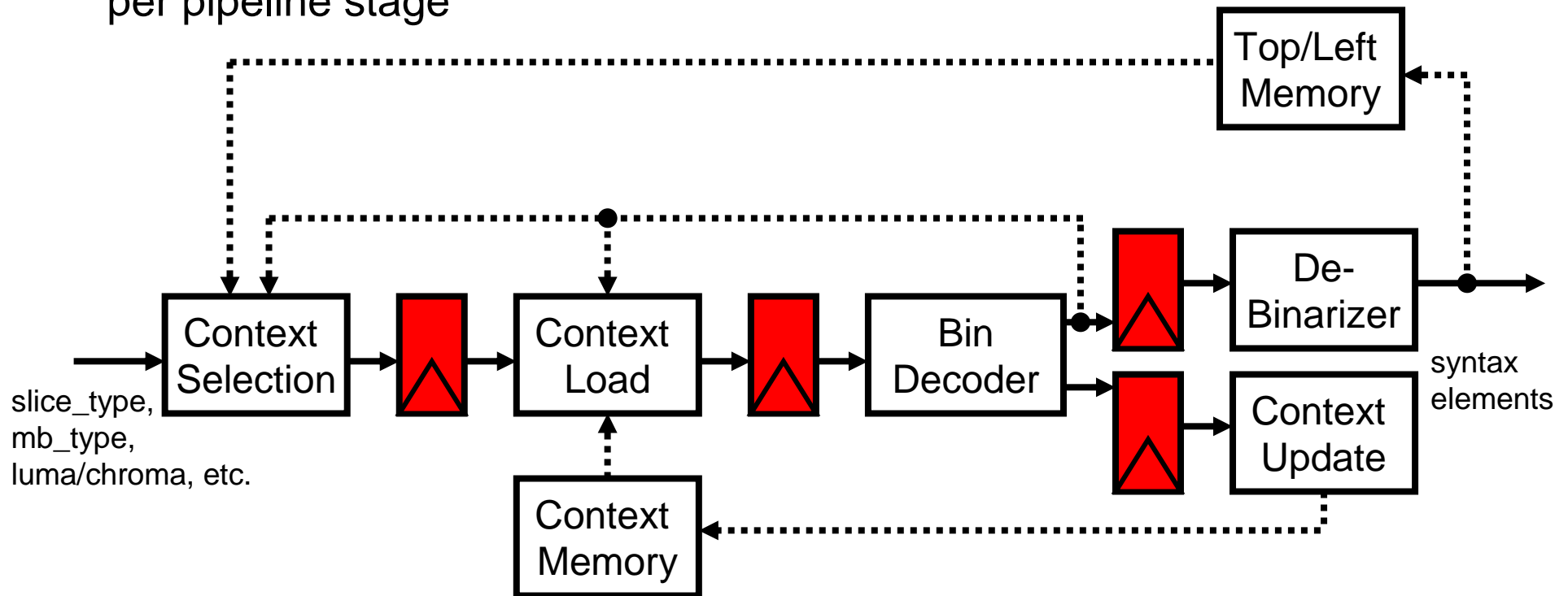
Speculative Computations for multi-bin

- If want to decode multiple bins in a cycle, need to perform speculative computations due to feedback loops
- Example: Decode two bins in parallel
- To decode bin2 before bin1 is resolved
 - Candidate0: compute bin2 assuming bin1=0
 - Candidate1: compute bin2 assuming bin1=1
 - Once bin1 is resolved, select either Candidate0 and Candidate1
- Overhead
 - Additional computations: for N bins parallel decoder requires $1+2+\dots+2^{(N-1)}$
Number of operations increases exponentially!
 - Additional delay due to selecting candidates



Pipelined CABAC

- To reduce critical path delay, CABAC is typically pipelined
- Due to feedback loops, number of speculative computations increase per pipeline stage



Pipelined 2 bin per cycle CABAC implementation can require up to 12 context selection calculations

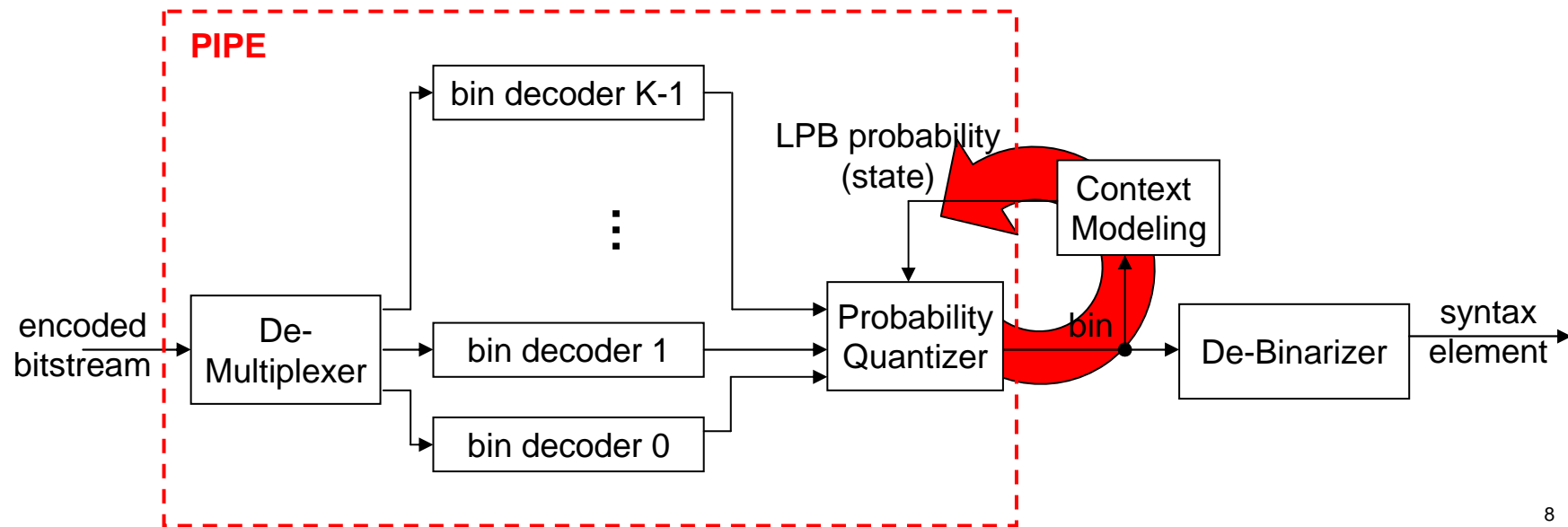
For 4 bin per cycle in future video codecs, need up to 60 context selections

- In AVC, optimizations done to reduce number of speculative computations
- Additional dependencies in HEVC
 - Increases operations per context selection
 - Makes operation reduction difficult
 - potentially increase critical path delay



Speculative Computations with PIPE/V2V

- Decode N bins in parallel using $1+2+\dots+2^{(N-1)}$ probabilities (state, MPS)
- In PIPE, read bin from bin decoder based on probabilities
- Speculatively read $1+2+\dots+2^{(N-1)}$ bins from multiple bin decoders
- Multiple bins can be read from same bin decoder
 - Need to keep track of which bins were speculatively read and which bins were actually read once bin resolved



Conclusions

- Multiple feedback loops in CABAC cause serial dependencies
- Important to look at **both** encoder and decoder
 - Loops are more complex at the decoder
- Techniques that increase implementation throughput require speculative calculations
- Additional dependencies introduced in HEVC increase number of speculative computations
- For alternative entropy coding methods, important to evaluate end-to-end throughput and decoder complexities