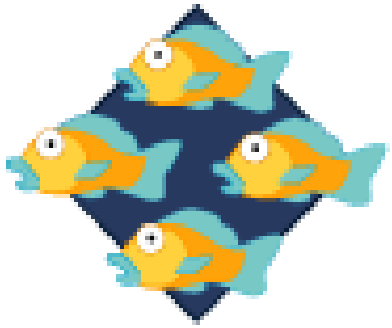
Ogg Vorbis Decoder Implementation in Bluespec



Marek Doniec
MIT CSAIL 2008

Image source: http://www.vorbis.com

Ogg What!?

Ogg Vorbis audio codec:

General purpose, perceptual.

Decoder computationally simple.

Open source! License & royalty free!

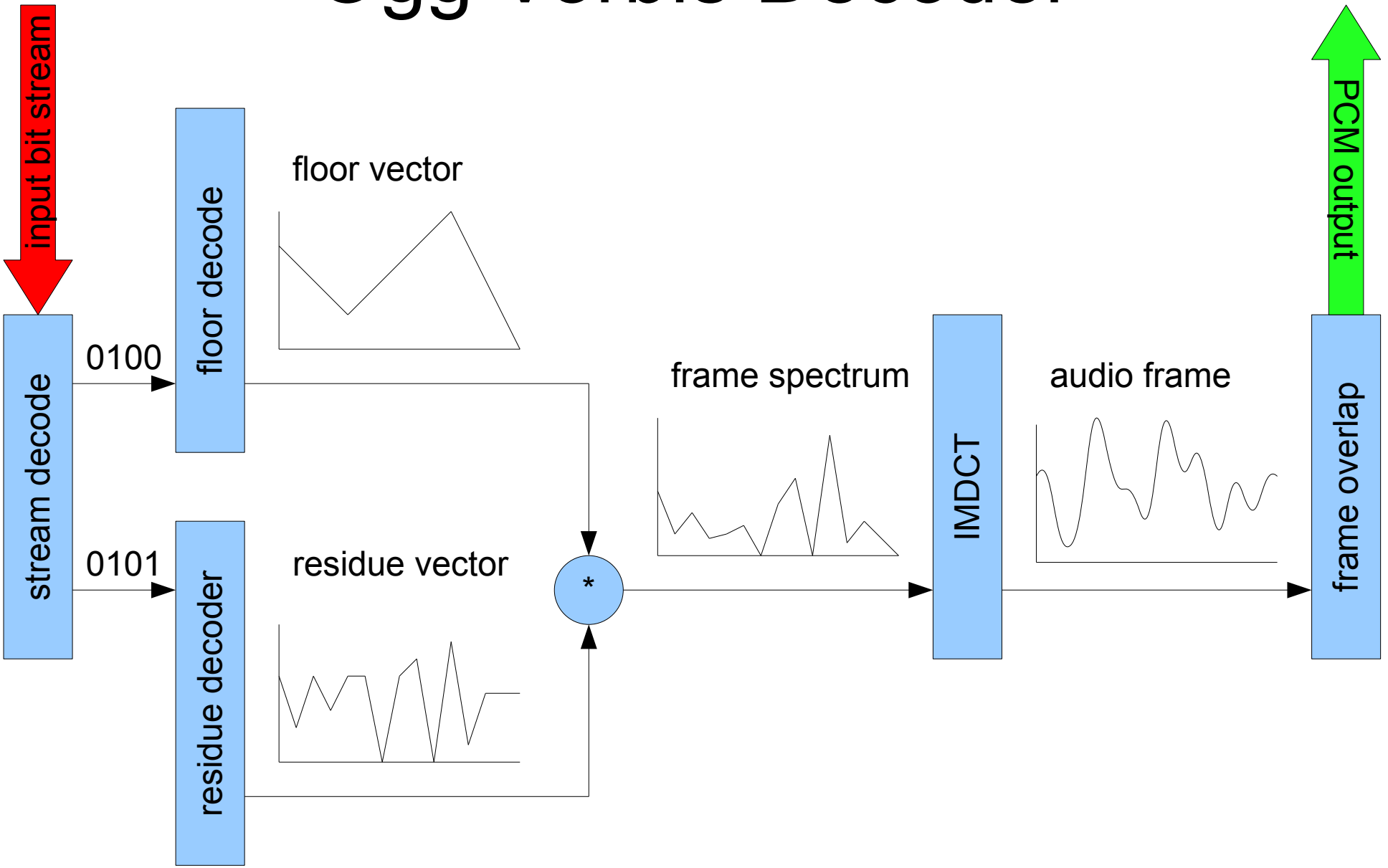
Outline

1.) Internal decoder structure.

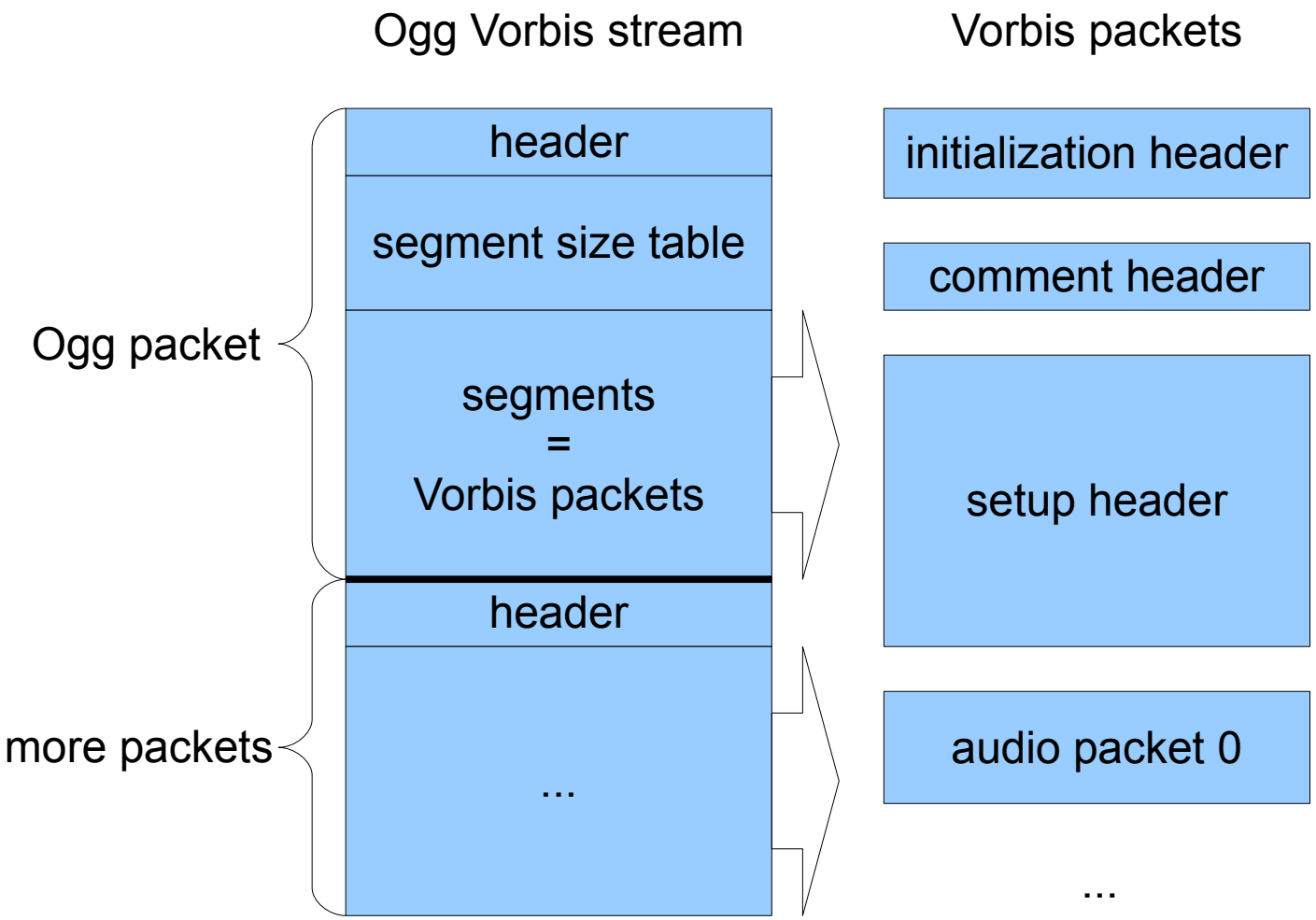
2.) Current hardware implementation.

3.) Decoder performance and future work.

Ogg Vorbis Decoder



Ogg Container for Vorbis Data



Initialization and Comment Headers

- Initialization header:
  - Vorbis version
  - Number of channels, sample rate
  - Small and large frame size (block sizes)
- Comment header:
  - Song title, artist, weblink, ...
  - Not important for decoder!

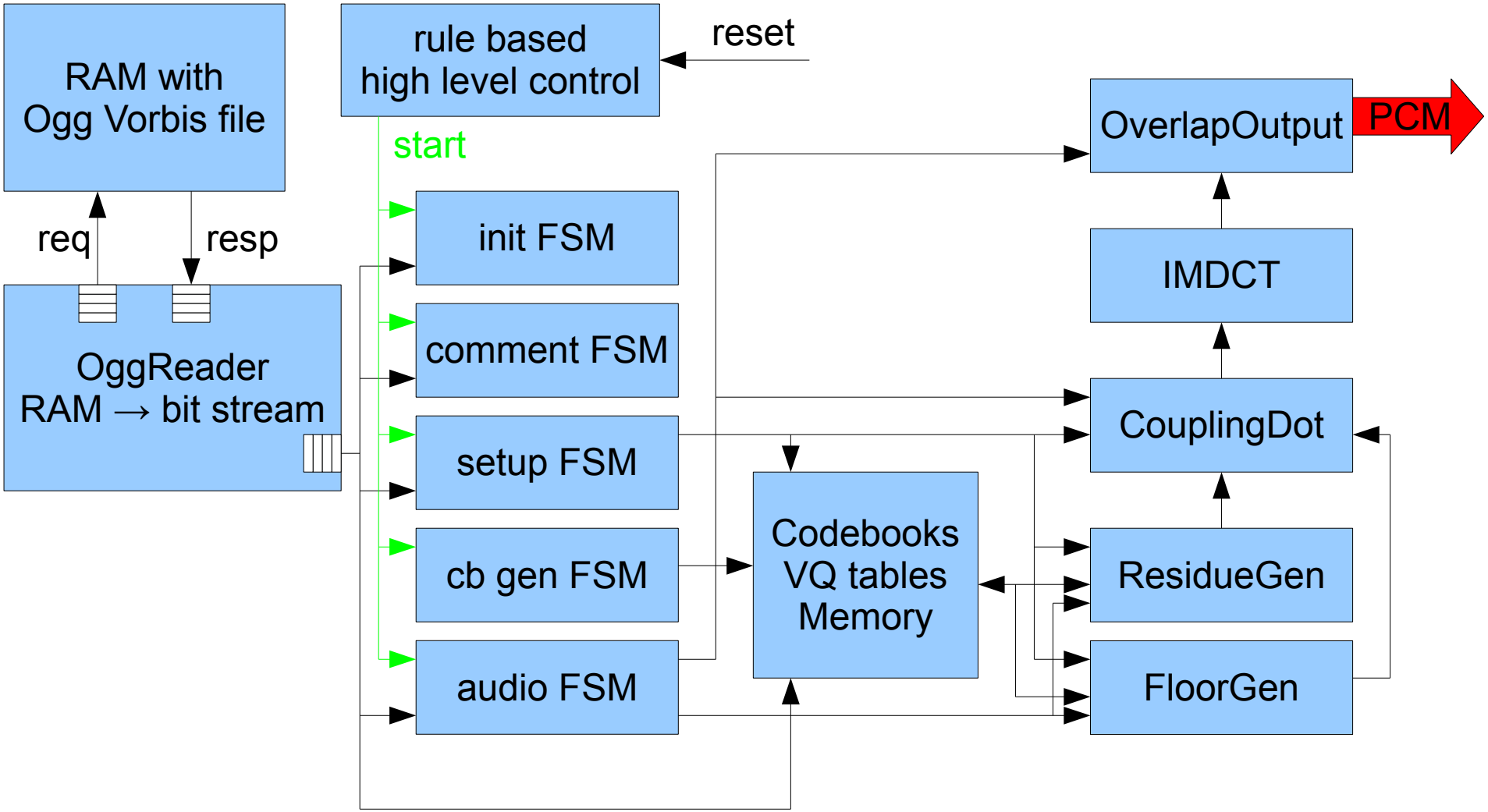
Setup Header

- Contains probability model:
  - Huffman codebooks
  - Vector quantization (VQ) tables
  - Floors (type, size, resolution, cb index)
  - Residues (type, size, resolution, passes, cb index)
  - Modes and Mappings
- Needs to be stored for further decode:
  - Large memory consumption: header > 2kb
  - When unpacked > 256kb!

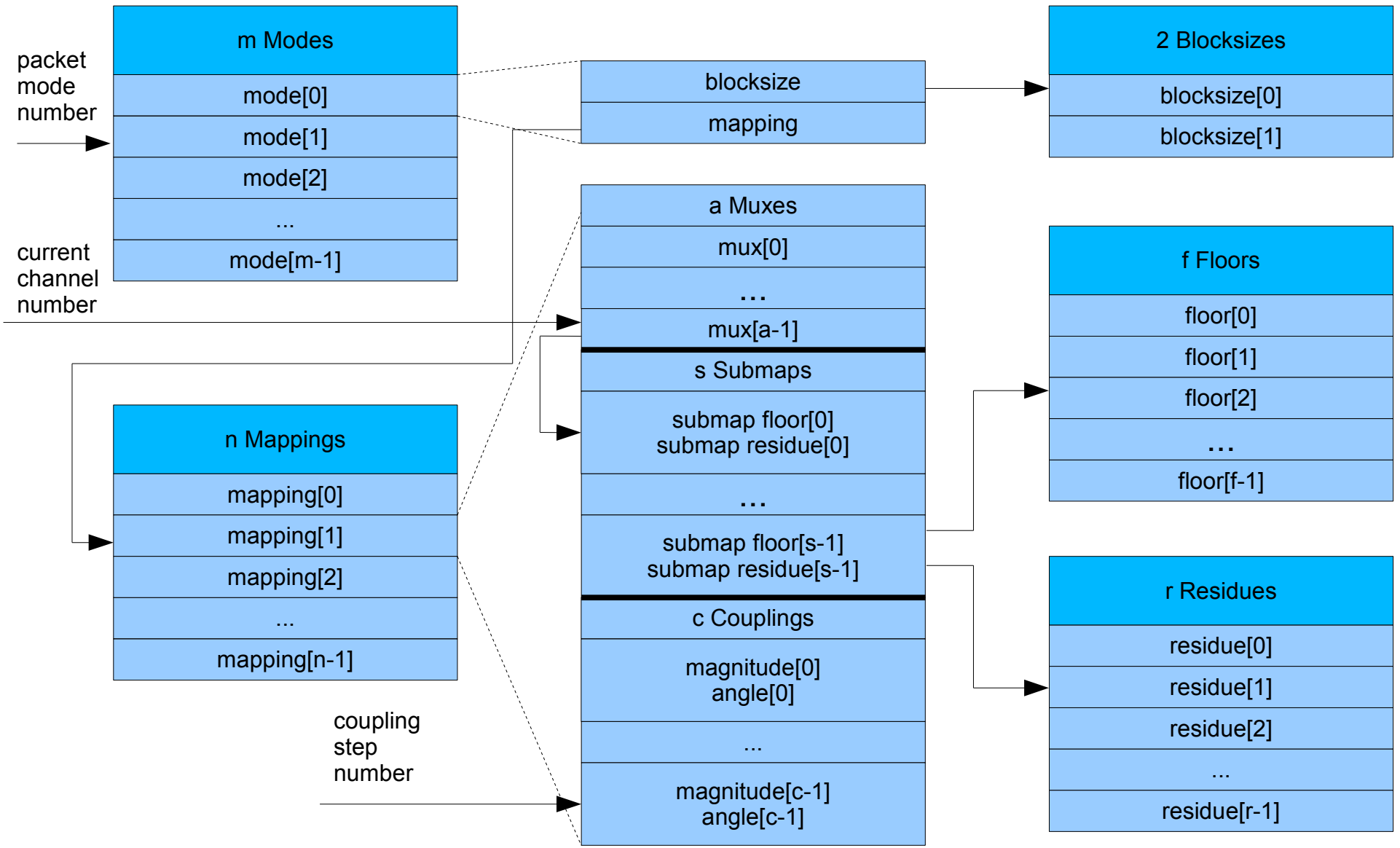
Audio Packets

- Each packet contains one frame for all channels.
- First Floors are stored (coarse spectrum)
- Residues follow (spectral details)
- Decoder reconstructs spectrum and feeds it to IMDCT. The result is overlapped with previous frames to generate output.

Bluespec Implementation

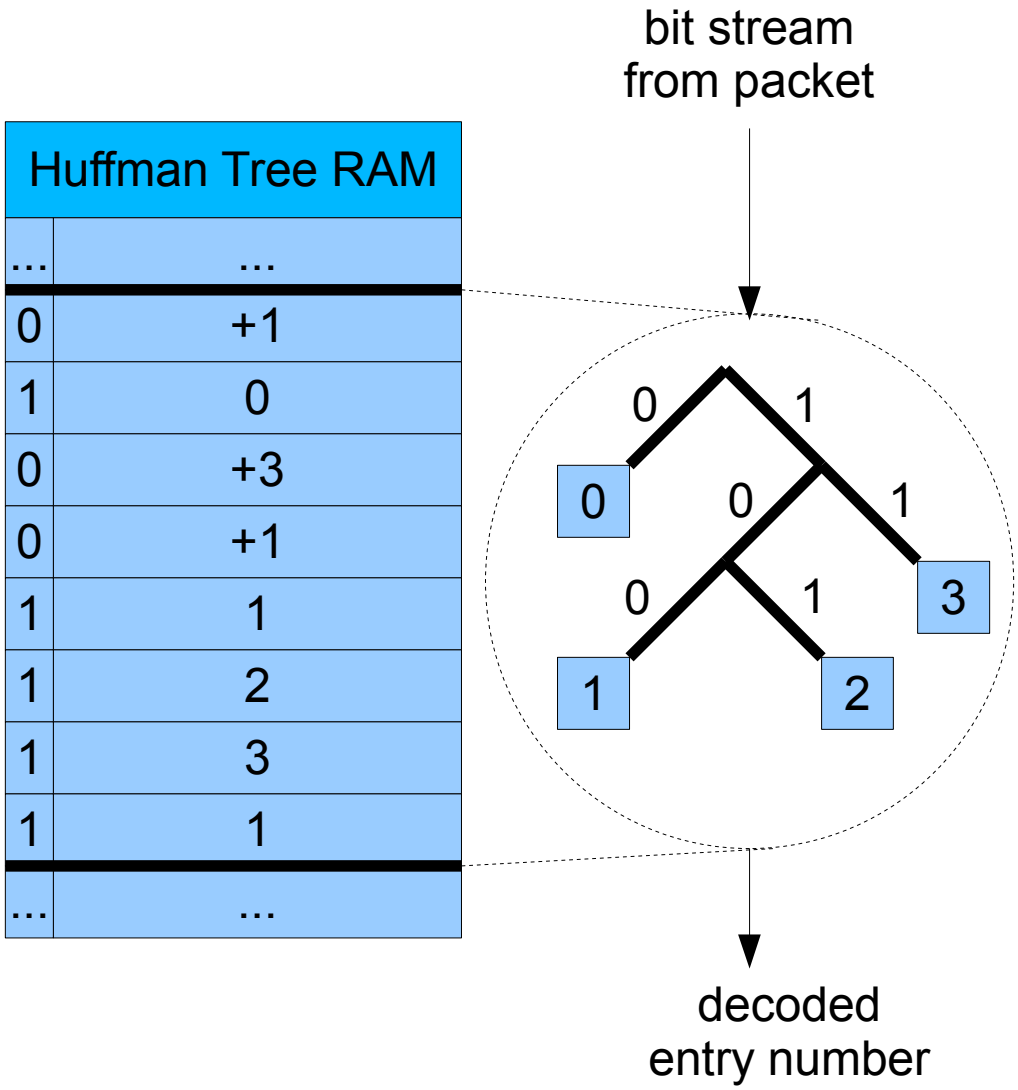


Audio Packet Setup

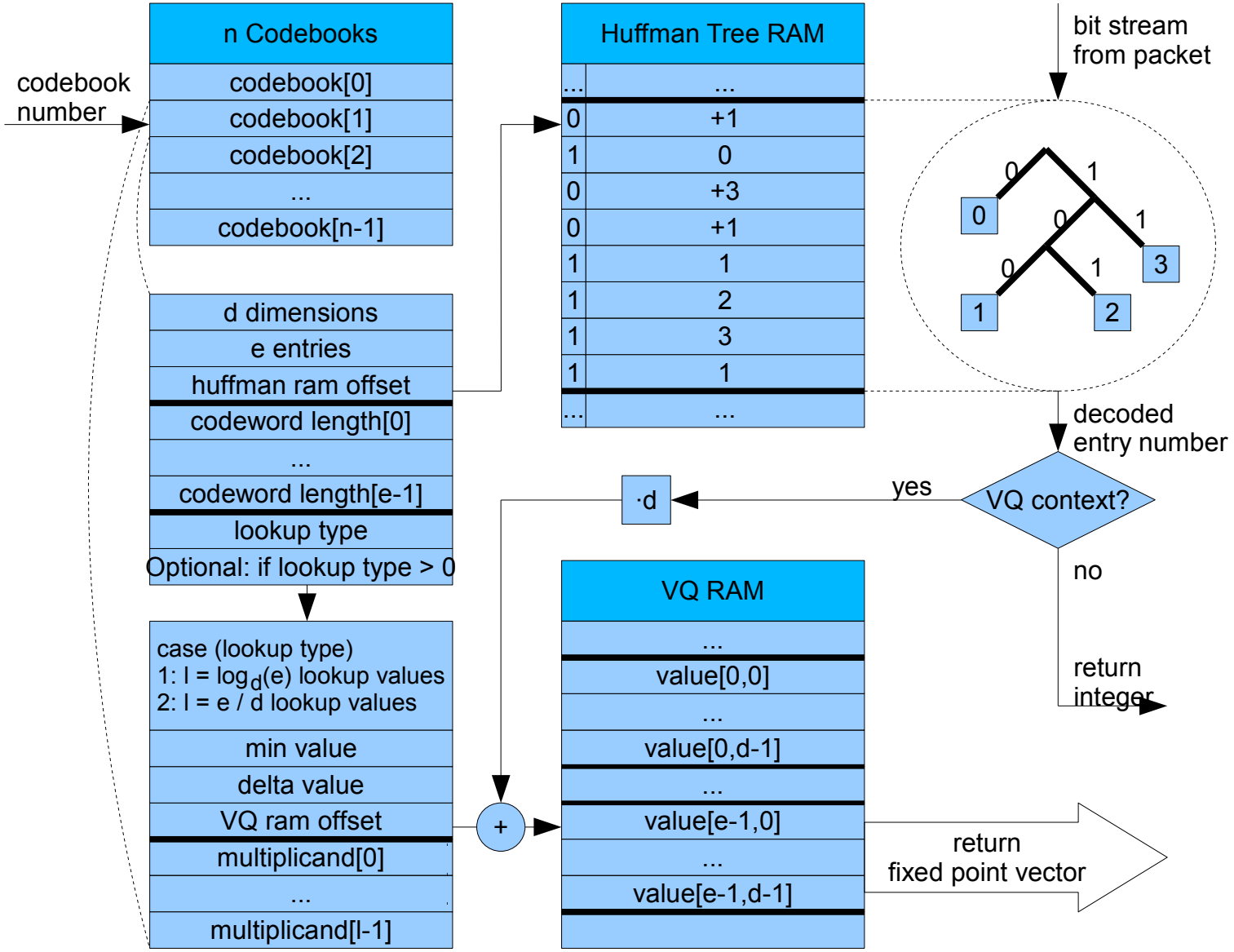


Huffman Codebooks

- Need low memory usage + fast decode!
- Binary tree with one 16 bit entry per node
  - MSB: data?
  - 15LSB: data / right child offset
  - Left child is in next entry.
- Current usage: ~16kb



Codebooks and VQ Tables



Performance and Future Work

- Tested with 16 bit mono, 22050 Hz audio files:
  - Lowest quality < 1 MHz / sec
  - Highest quality < 2 MHz / sec
  - Verilog multiplications and Divisions are used!
- To be done next:
  - Place queues and pipeline!
  - Optimize VQ tables (clock cycles vs RAM)
  - Test multichannel decode.