Video-Conferencing System

Evan Broder
Chris Post

Goals

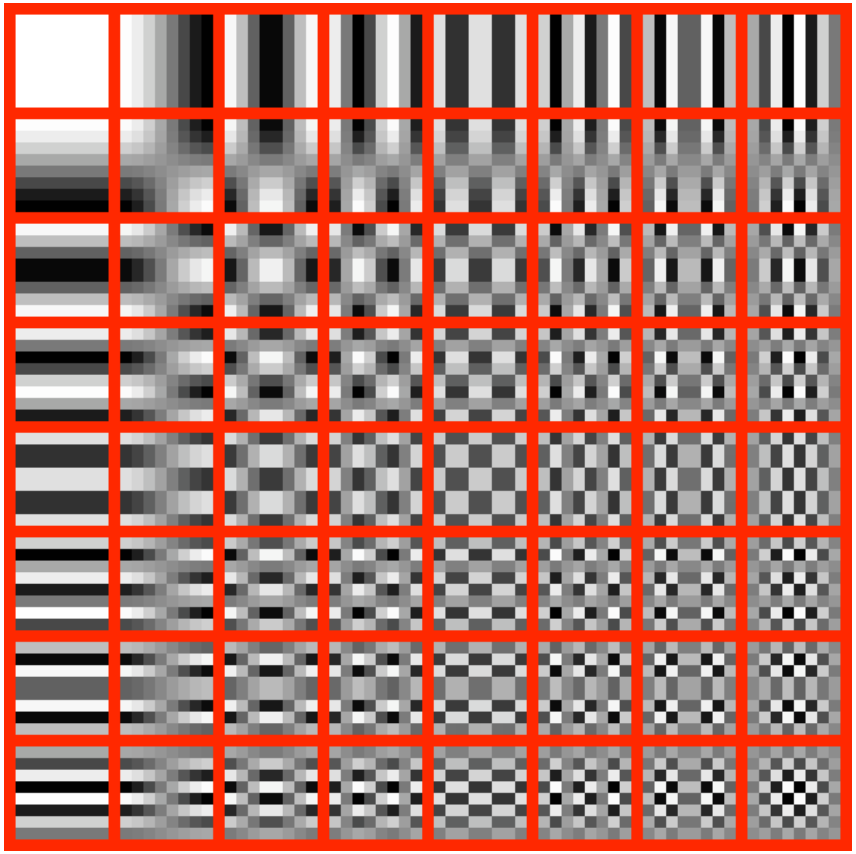
- Video conferencing system
- Transmit video and audio on low-bandwidth serial line
- Compress video
- Downsample audio
- Resistant to dropped packets

Specifications

- Video
  - 320x240
  - 15 fps
- Audio
  - 16 kHz
  - 8 bit
- 250 kbps serial
  - Approx. 56x compression

Compression: What is JPEG

- Frame-based compression
- YCrCb color space
- Chroma subsampling
- Encoding
  - DCT

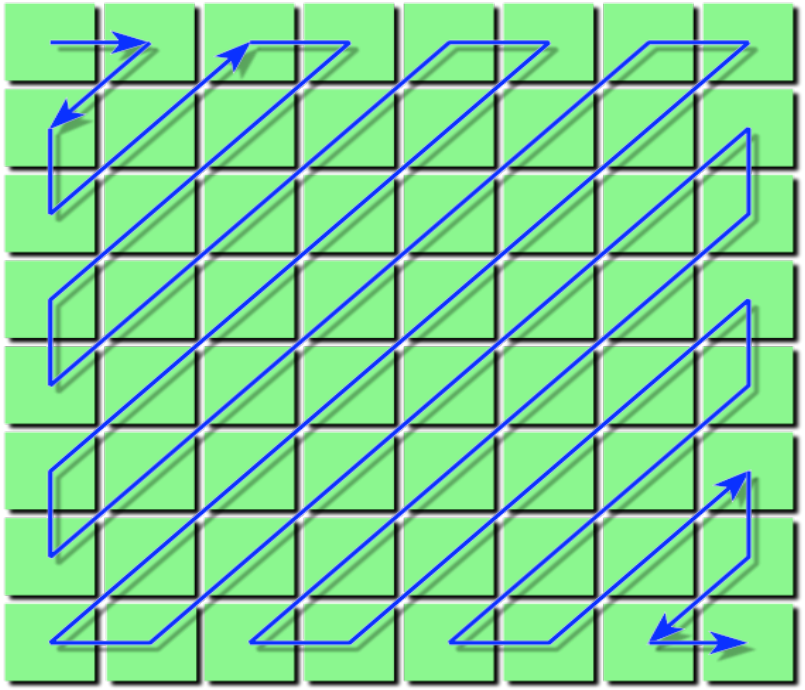


Compression: What is JPEG

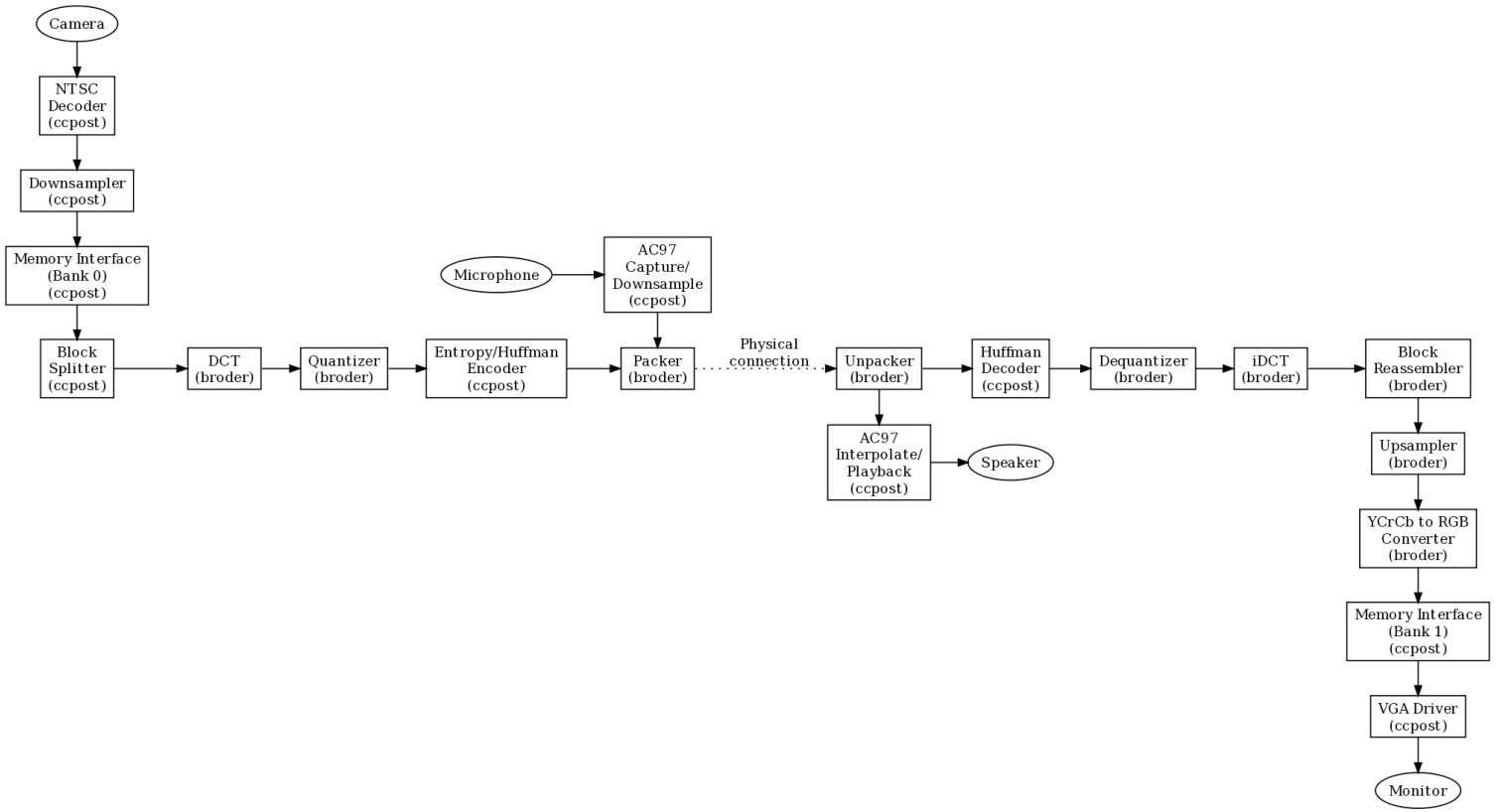
- Frame-based compression
- YCrCb color space
- Chroma subsampling
- Encoding
  - DCT
  - Quantization

Compression: What is JPEG

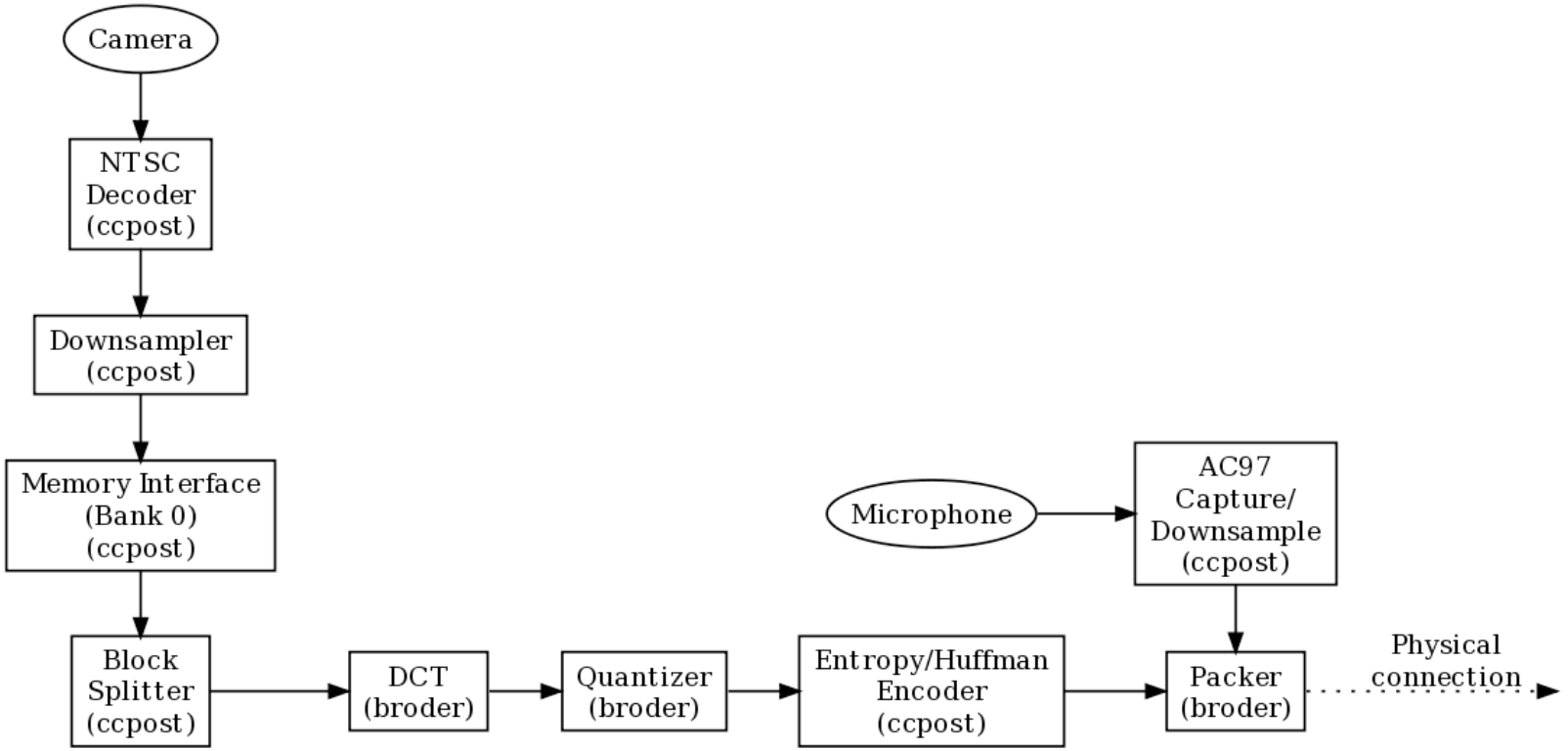
- Frame-based compression
- YCrCb color space
- Chroma subsampling
- Encoding
  - DCT
  - Quantization
  - Entropy/Huffman encoding
- Decoding



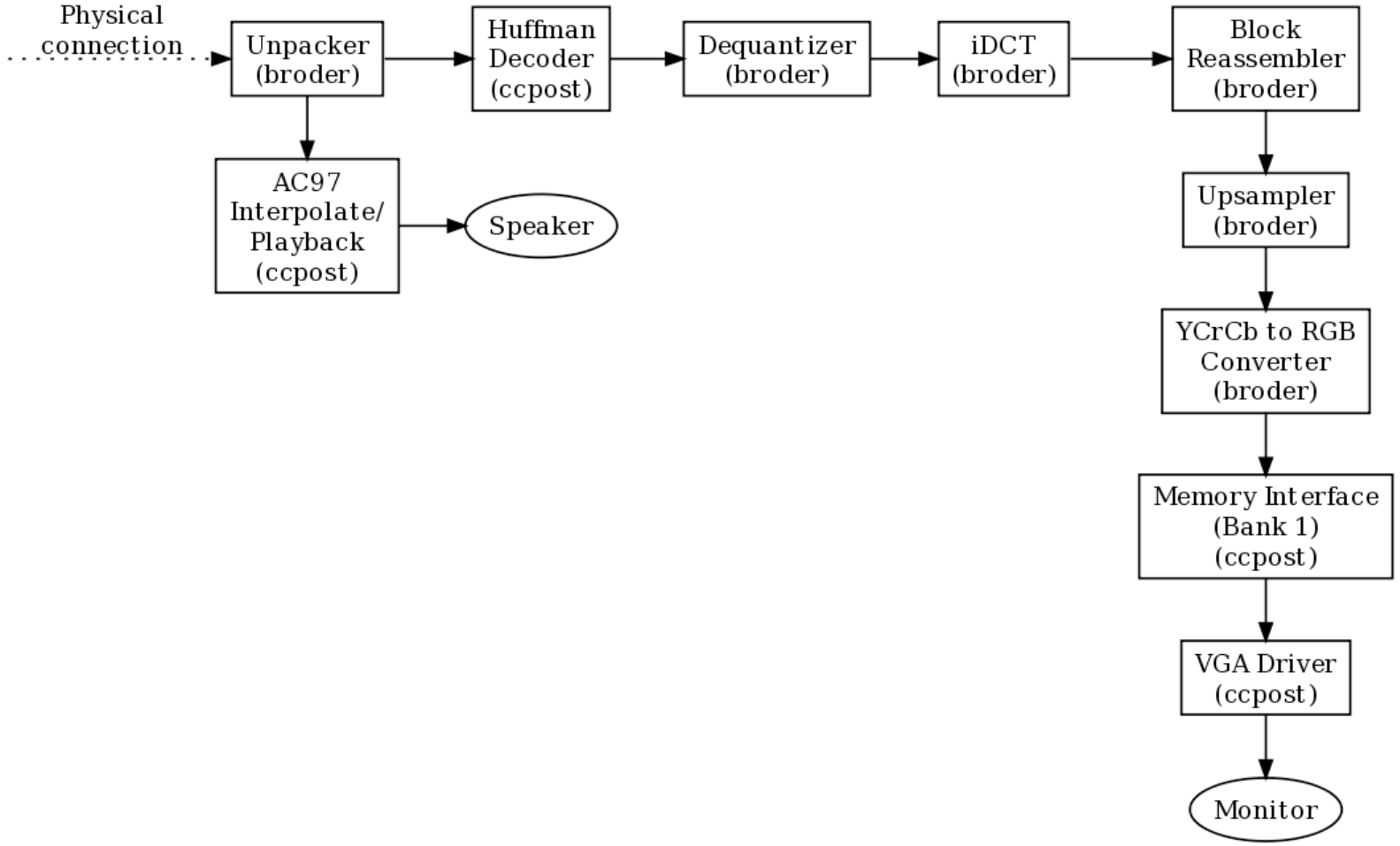
Logic Overview



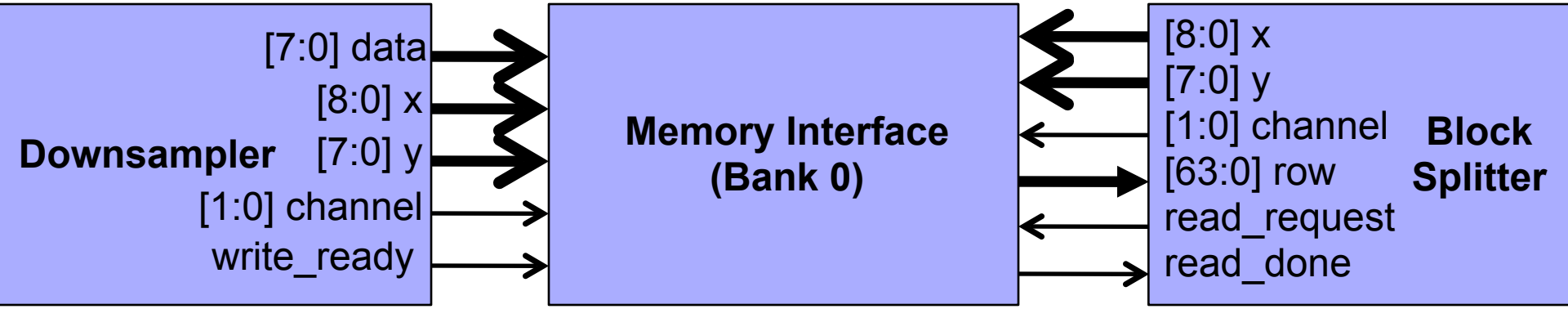
Logic Overview



Logic Overview

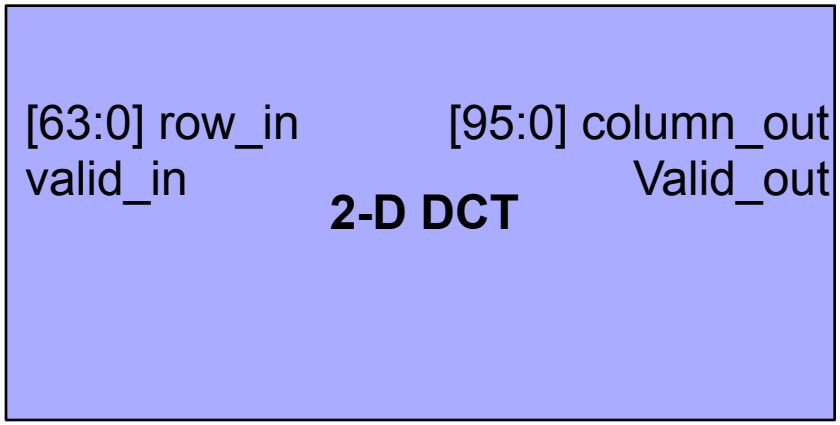


Memory Interface



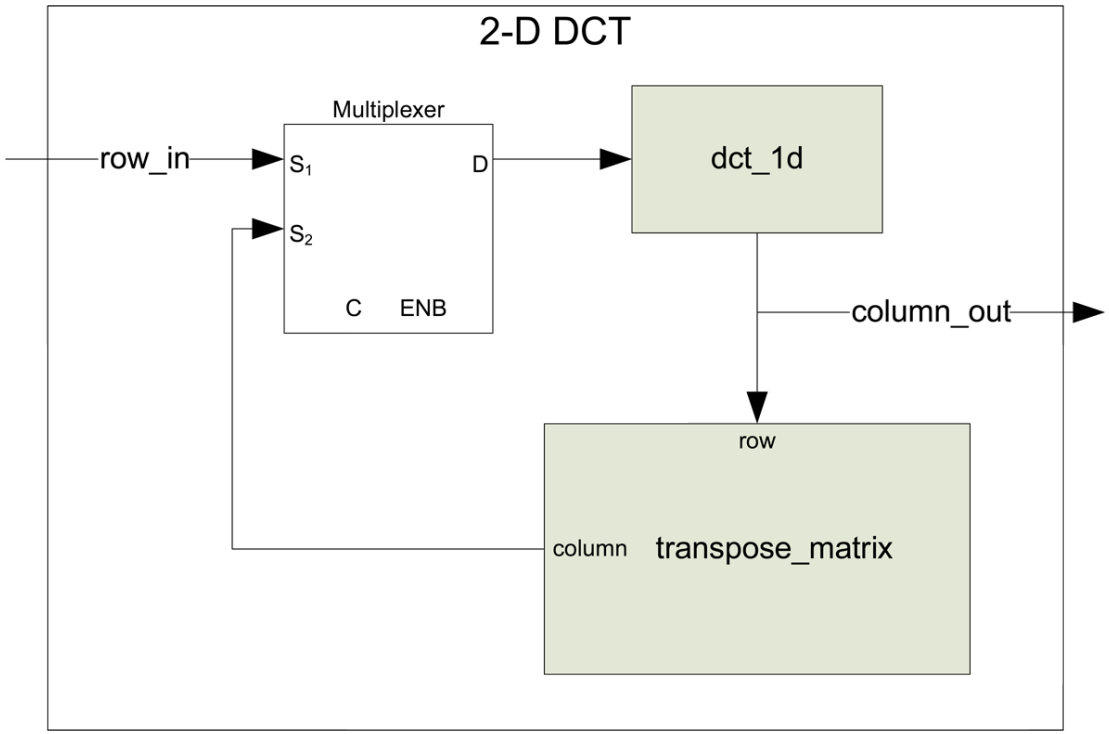
DCT

- Input is one row per clock cycle
- Output is one column per clock cycle
- Width changes because DCT outputs more information



DCT

- Only one 1-D DCT module
- Latency is approximately 20 clock cycles



1-D DCT

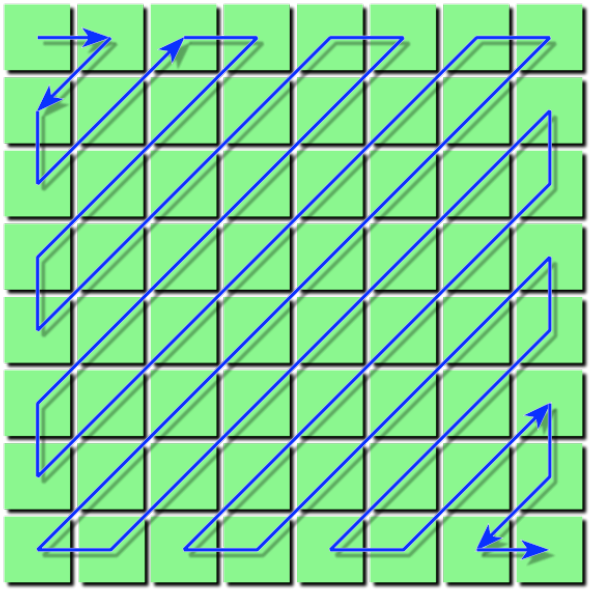
- Using Winograd algorithm
- Original algorithm
  - 64 multiplies
  - 64 adds
- Winograd
  - 6-stage pipelined
  - 29 add/subs
  - 5 multiplies

TABLE 1 – 1-D DCT algorithm

STEP 1	STEP 2	STEP 3	STEP 4	STEP 5	STEP 6
b0=a0+a7	c0=b0+b5	d0=c0+c3	e0=d0	f0=e0	S0=f0
b1=a1+a6	c1=b1–b4	d1=c0–c3	e1=d1	f1=e1	S1=f4+f7
b2=a3–a4	c2=b2+b6	d2=c2	e2=m3xd2	f2=e5+e6	S2=f2
b3=a1–a6	c3=b1+b4	d3=c1+c4	e3=m1xd7	f3=e5–e6	S3=f5–f6
b4=a2+a5	c4=b0–b5	d4=c2–c5	e4=m4xd6	f4=e3+e8	S4=f1
b5=a3+a4	c5=b3+b7	d5=c4	e5=d5	f5=e8–e3	S5=f5+f6
b6=a2–a5	c6=b3+b6	d6=c5	e6=m1xd3	f6=e2+e7	S6=f3
b7=a0–a7	c7=b7	d7=c6	e7=m2xd4	f7=e4+e7	S7=f4–f7
		d8=c7	e8=d8		

Entropy/Huffman Encoder

- Zigzag ordering from upper-left
- Result is Huffman coded
- Less bits to store 0 and small numbers
- Run-length to pack runs of same value
- Special end of block code when only 0s left



Packer

- Buffers output from each channel
- When it has all the data it needs for one packet, it frames and transmits the packet
- Keep track of coordinates of video blocks

Video:

Start	Channel	Len	X	Y	Payload	CRC

Audio:

Start	Channel	Len	Payload	CRC

Reception and Decoding

- Blocks decoded in order received
- Block decoding operations are simple inverses of the encoding counterparts
- Each block gets coordinates transmitted with it through decoding
- Block reassembler uses these coordinates to do block buffering

Timeline

- Now: First version of 2-D DCT
- After Thanksgiving
  - Final version of 2-D (i)DCT
  - (De)Quantizer
  - NTSC Decoder
  - Downsampler
  - VGA Driver
- Nov 28
  - Entropy/Huffman Encoder/ Decoder
  - Packer/Unpacker
- Dec 5
  - Memory Interfaces
  - Splitter
  - Reassembler
  - Upsampler
  - Color Space Converter

Questions?