Adding some light to computing ….

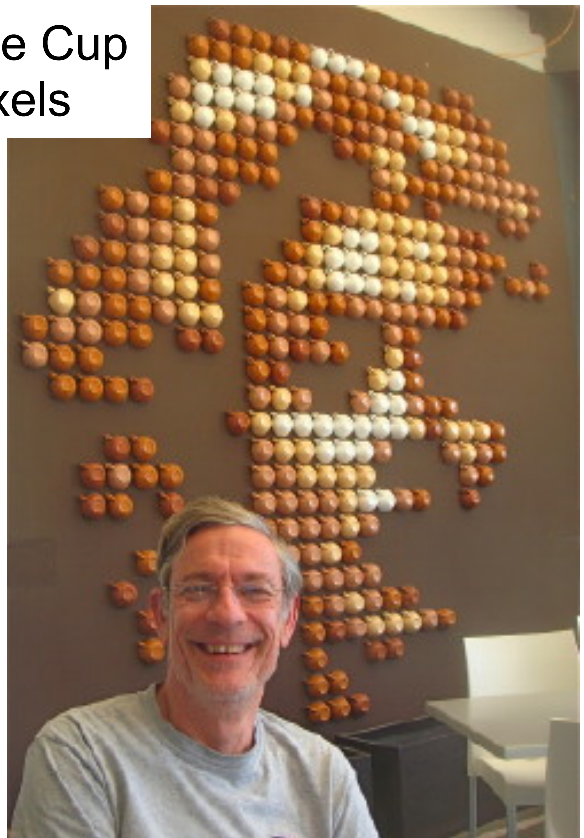
Bits of Color

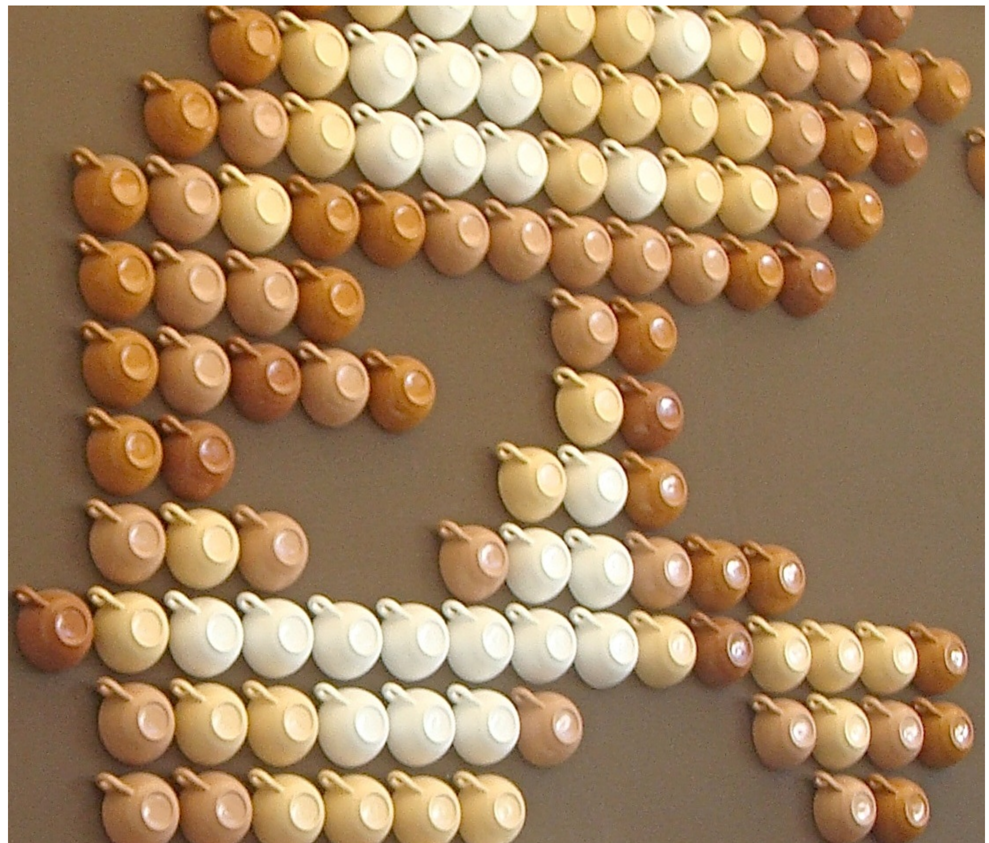
Lawrence Snyder
University of Washington, Seattle

Return To RGB

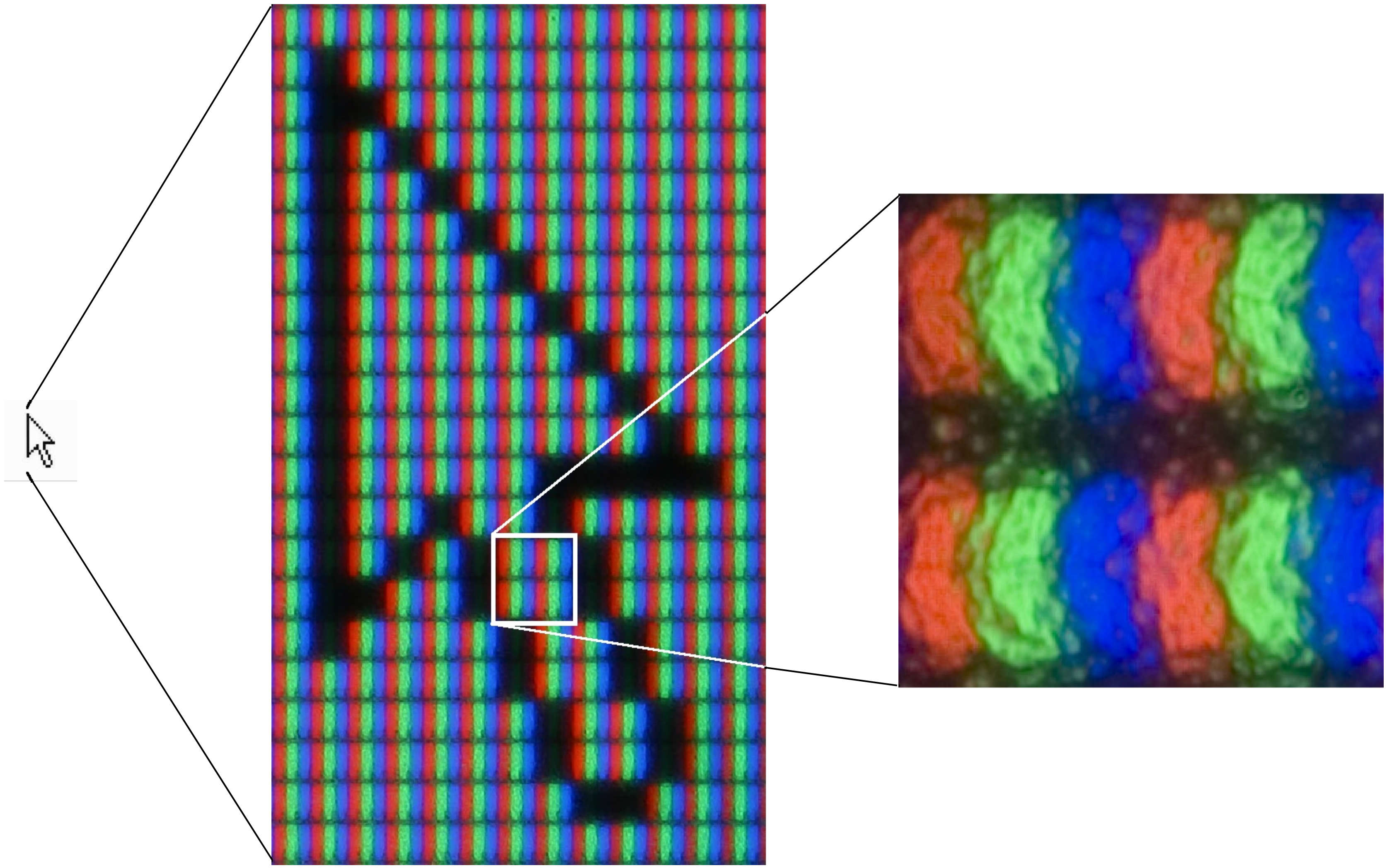
- Recall that the screen (and other video displays) use red-green-blue lights, arranged in an array of picture elements, or *pixels*

Coffee Cup
Pixels



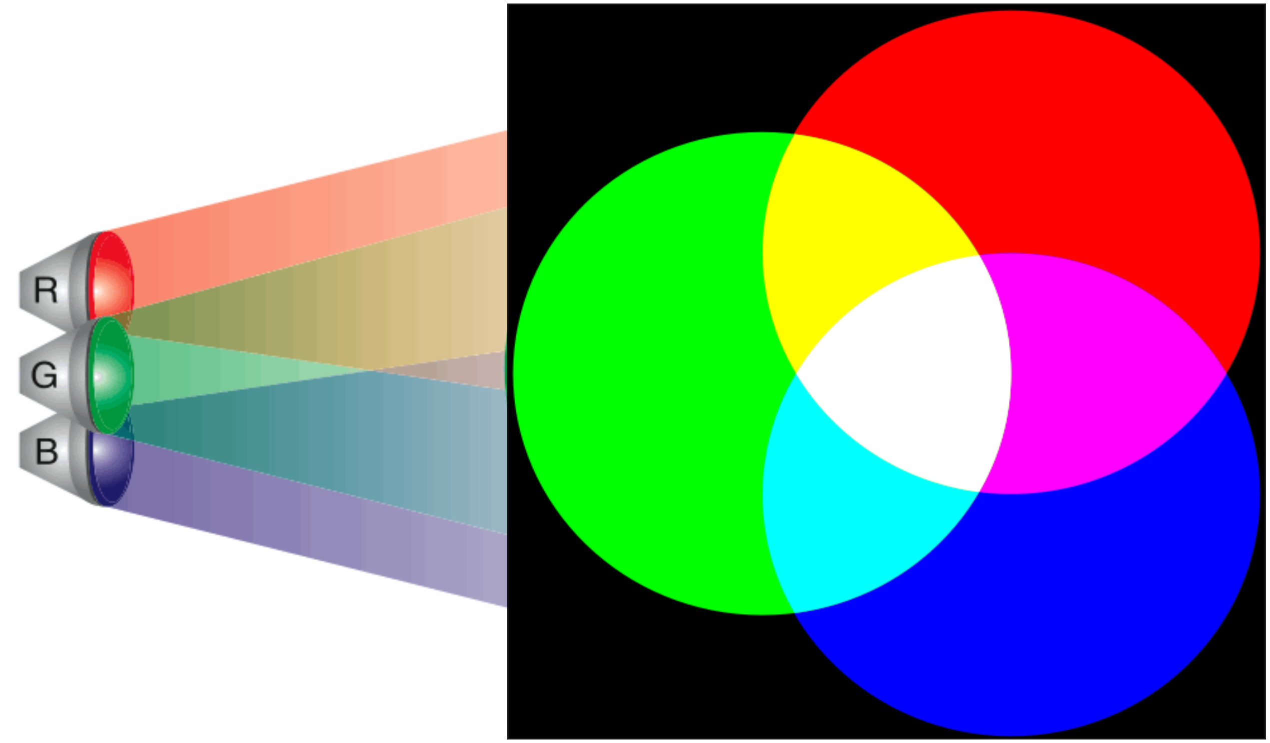


Actual Pixels From TFT LCD Display



Combining Colored Light

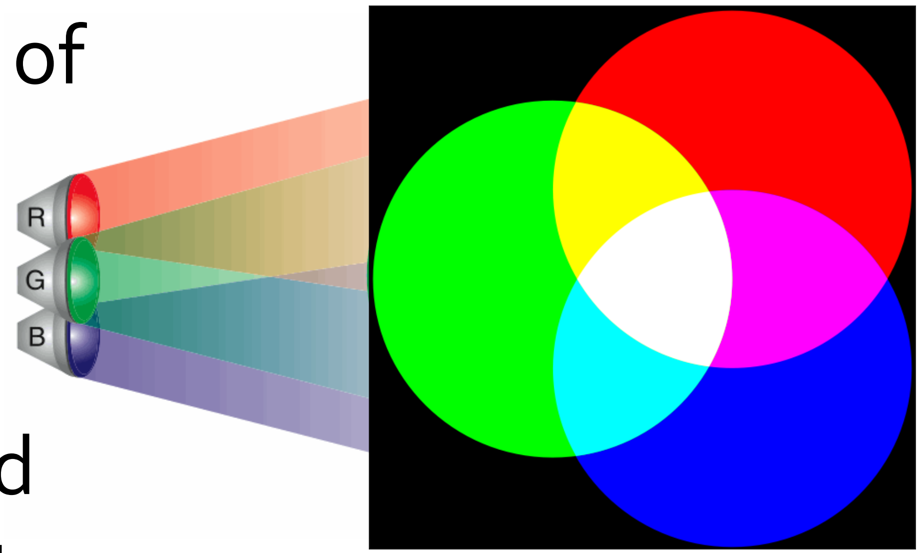
- The Amazing Properties of Colored Light!

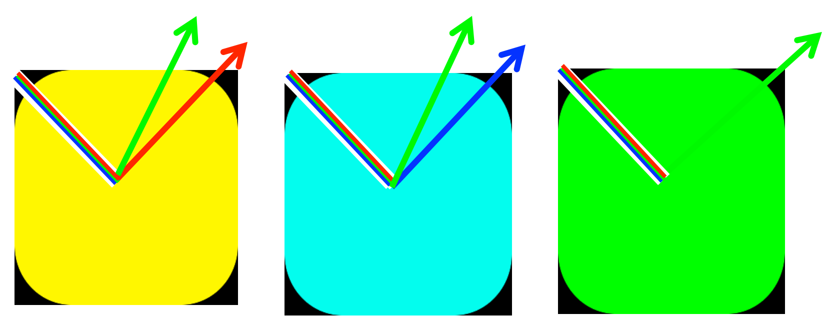


- Caution: It doesn't work like pigment

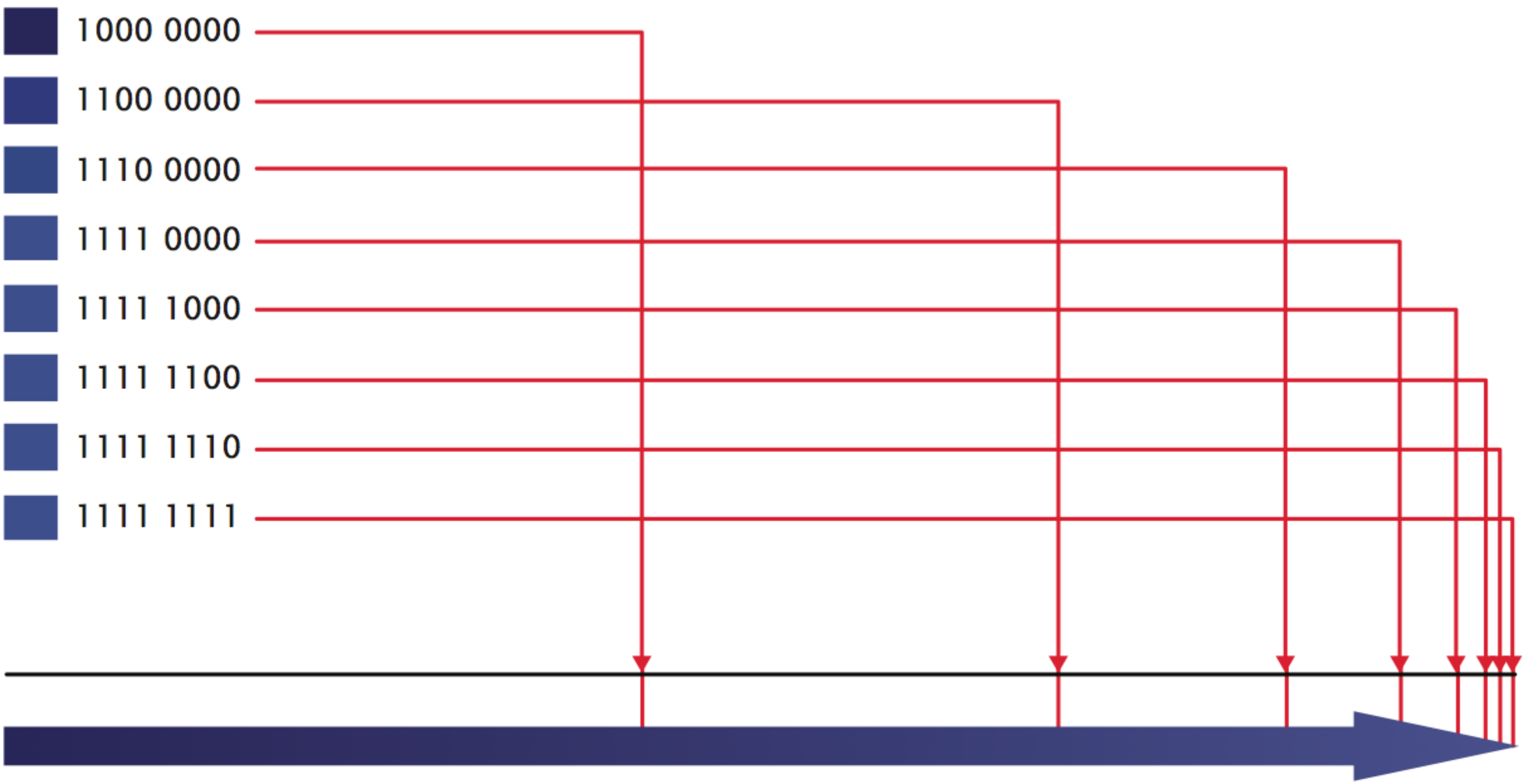
Green + Red = Yellow?

- Colored light seems to violate our grade school rule of green = blue + yellow What gives?
- In pigment, the color we see is the reflected color from white light; the other colors are absorbed



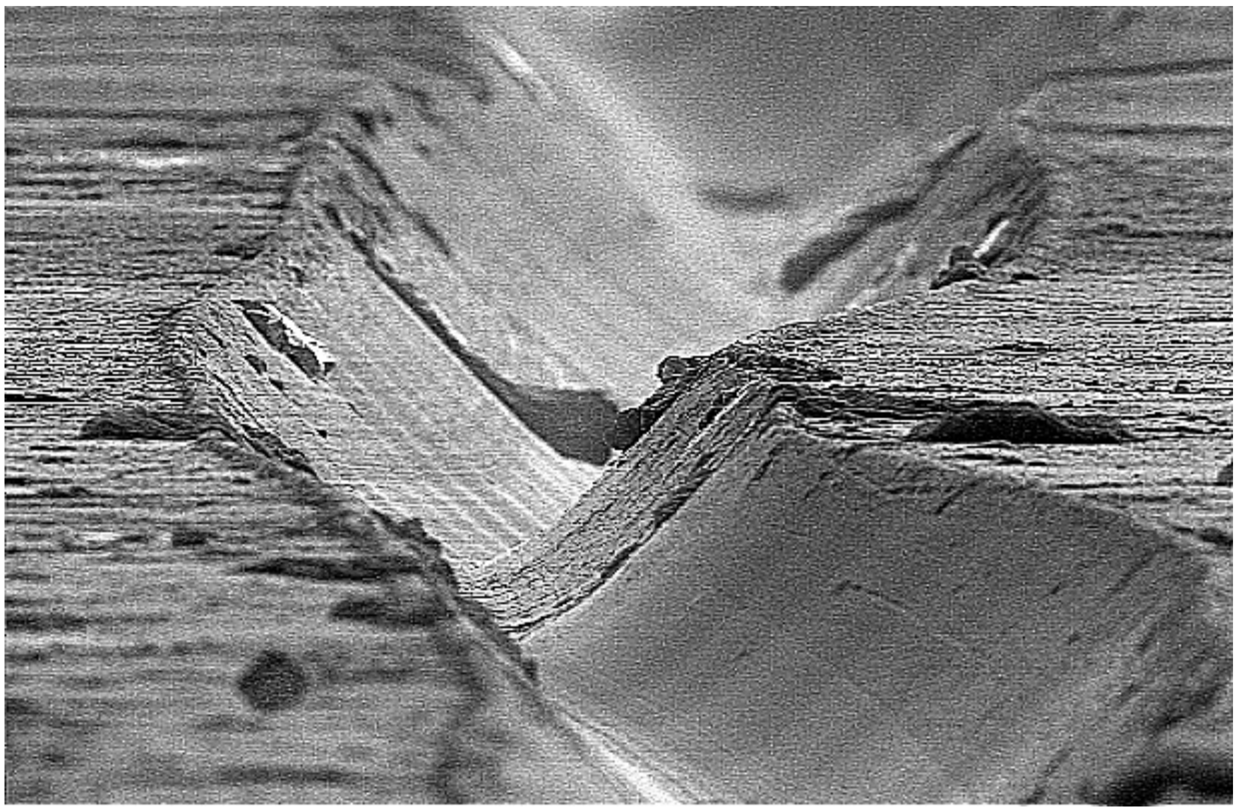


Each Bit Adds Another 1/2



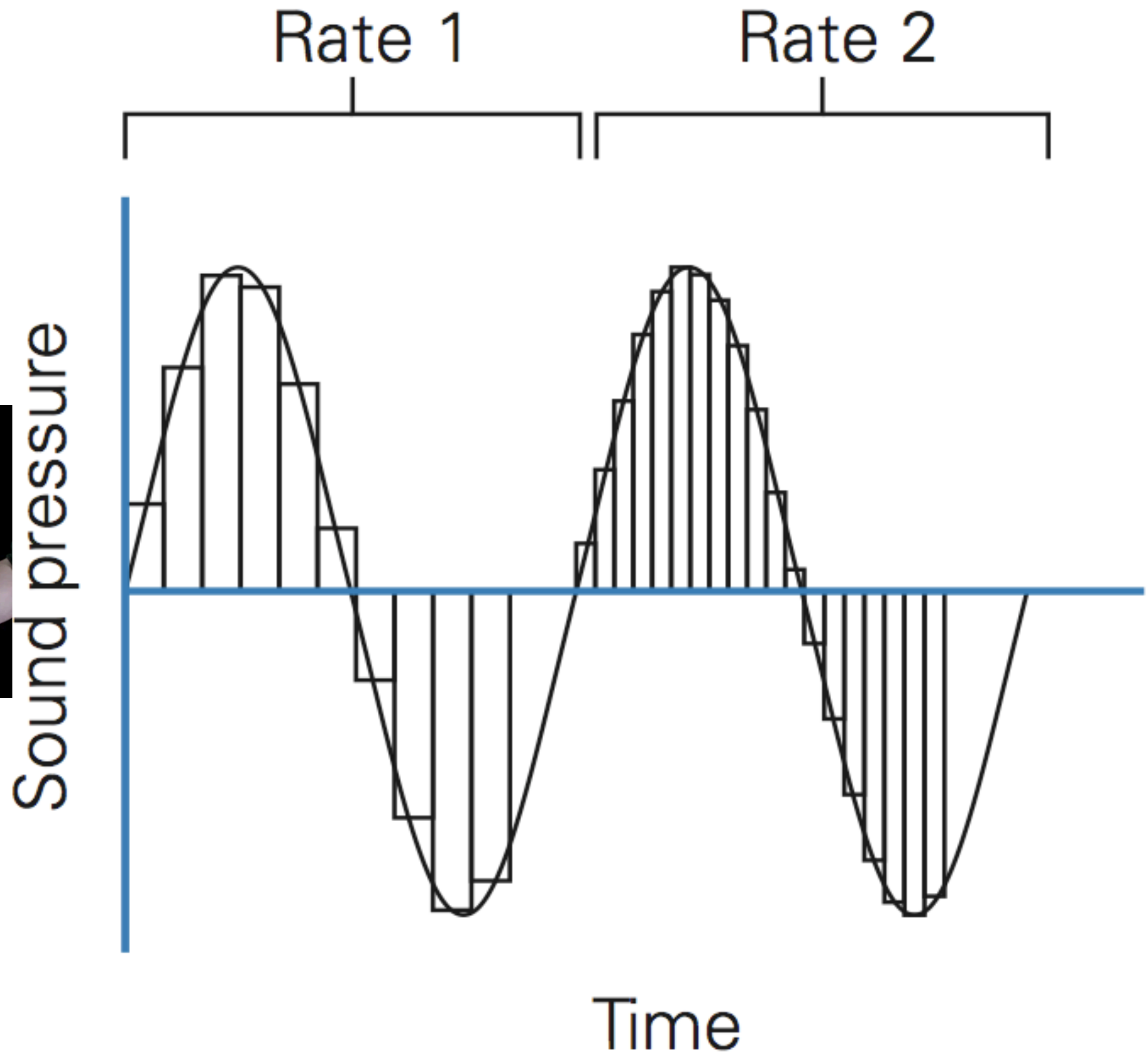
Not All Information Is Discrete

- Analogue information directly applies physical phenomena, e.g. vinyl records

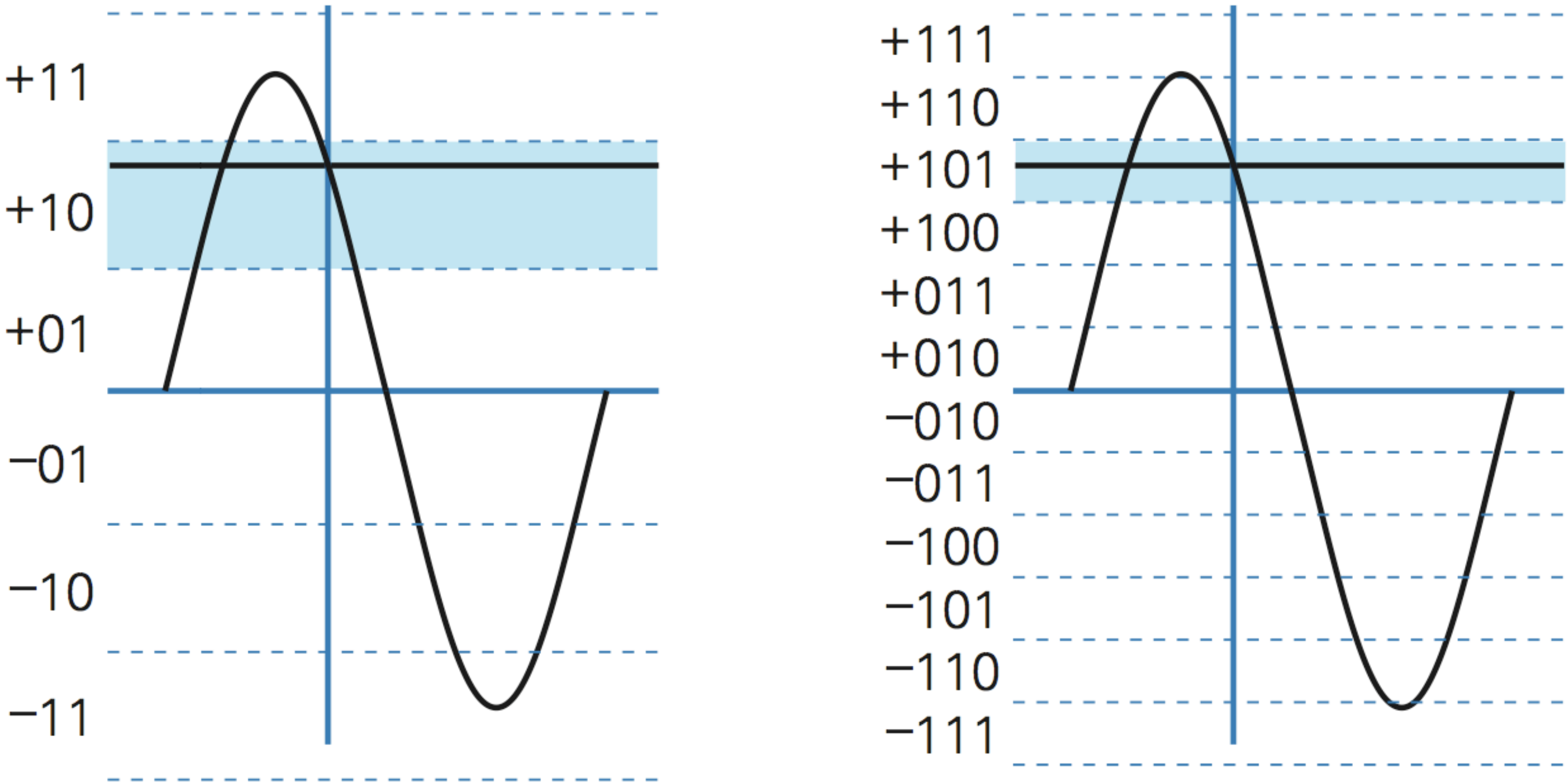


Analog Signals Become Discrete

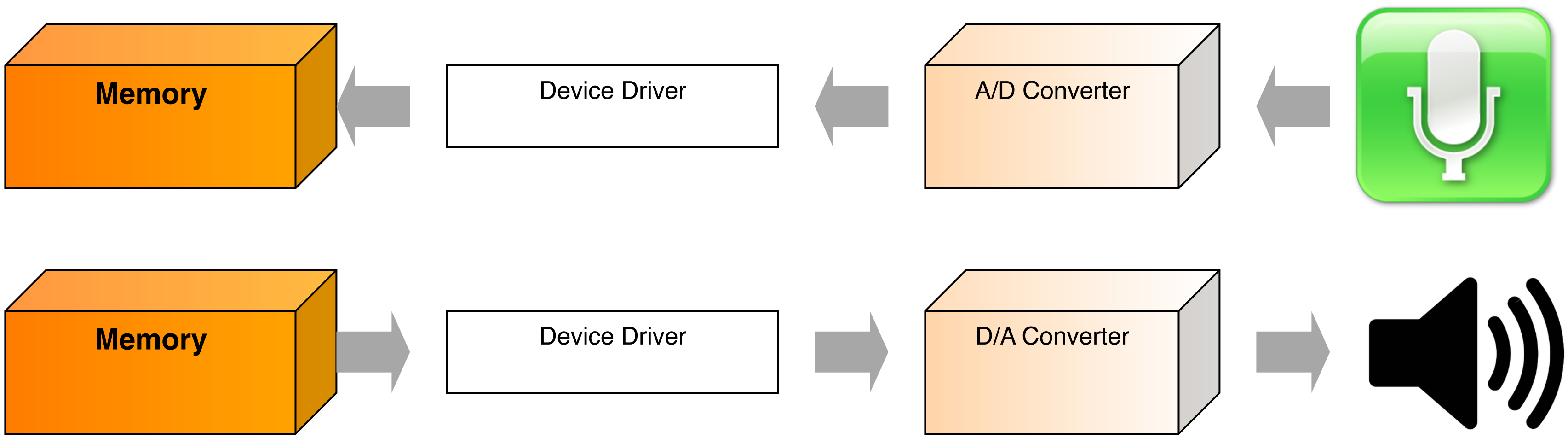
Sampling
the wave ...



Precision of the Sample



The World Is Analog – Go Between



Analog is needed for the "real world"

Digital is best for "information world"

- Can be modified, enhanced, remixed, etc
- Shared, stored permanently, reproduced, …

Sampling Sound …

- Going too slowly misses waves
- Going too fast keeps lots of redundant info
- The range of human hearing is 20-20,000 hz
  - Faster or slower, only the dog can hear it
  - Nyquist Rule: Sampling rate must be twice as fast as fastest frequency to be captured
- For technical reasons, the number is 44,100 hz
- How precise to sample: 16 bits gives -32k to 32k

Multimedia

- Many different forms of online information with special representations
  - JPG, MP3, MPEG, WAV …
  - Most forms of multimedia require many, many bits
    - A minute of digital audio:
      - 60 seconds x
      - 44,100 samples per second x
      - 16 bits each
      - x 2 for stereo
    - Is 84,672,000 bits, or 10,584,000 B
    - 1 hour is 635 MB!

Compress: Change Representation

- Often, most of the bits are not needed – MP3 audio is less than 1MB/min because many sounds can be eliminated – we can't hear them
- Compression … comes in two forms
  - Lossless – eliminated bits can be recovered
  - Lossy – eliminated bits are gone for good … MP3

Susanne Vega sings *Tom's Diner*
https://www.youtube.com/watch?v=VGw3W10QxLA

Lossless Compression

- Lossless compression seems strange – it eliminates bits that can be recovered again … weren't they necessary in the first place???
- Consider a fax –
  - Usually faxes use a scanner that produces rows of 0s and 1s.
  - Compress by counting … it's *run-length encoding*:

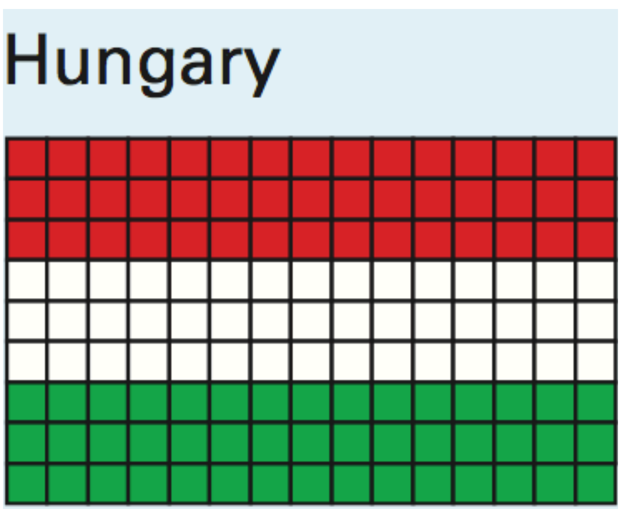
0000000000000000000000111111100000000011

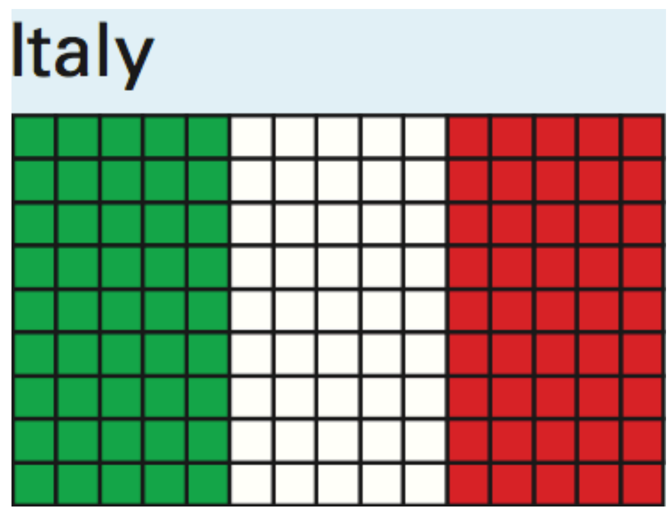
== 22:0,7:1,8:0,2:1

GIF Uses Same Idea

- Graphics Interchange Format (GIF) uses several kinds of compression
  - Color Table
  - Run Length Encoding
  - Lemple/Ziv/Welch Encoding

Color Table		
1	FF 00 00	
2	FF FF FF	
3	00 FF 00	



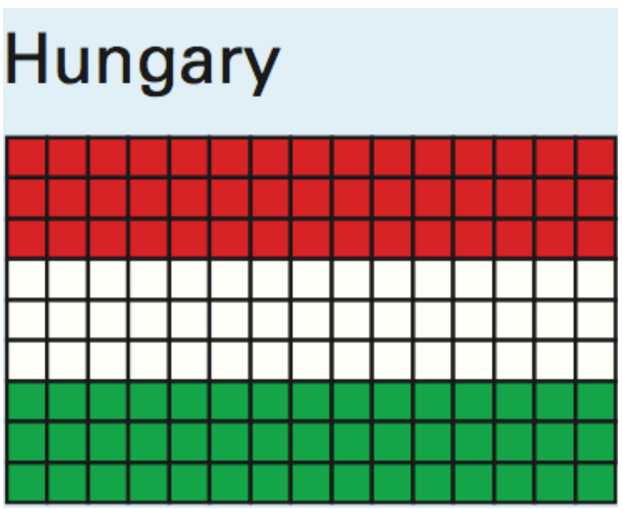


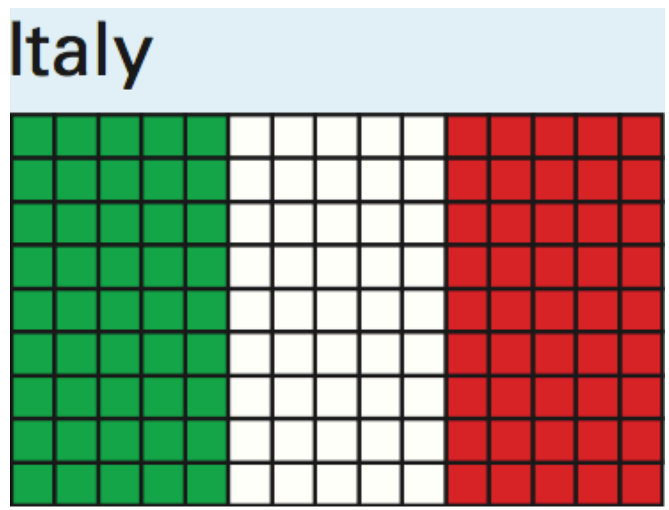
Compare Images Using GIf

- Compare Hungarian Flag and Italian Flag
  - huFlag: [15 × 9] 45:1, 45:2, 45:3
  - itFlag: [15 x 9]

5:3,5:2,5:1,5:3,5:2,5:1,5:3,5:2,5:1,
5:3,5:2,5:1,5:3,5:2,5:1,5:3,5:2,5:1,
5:3,5:2,5:1,5:3,5:2,5:1,5:3,5:2,5:1

Color Table		
1	FF 00 00	
2	FF FF FF	
3	00 FF 00	





JPG is Lossy

- Areas of similar color are represented by one shade … it's OK for a while

Review What We Know About Bits

- Facts about physical representation:
  - Information is represented by the presence or absence of a physical phenomenon (PandA)
    - Hole punched in a card; no hole [Hollerith]
    - Dog barks in the night; no barking in the night [Holmes]
    - Wire is electrically charged; wire is neutral
    - ETC
- Abstract all these cases with 0 and 1; it unifies them so we don't have to consider the details

Bits Work For Arithmetic

- Binary is sufficient for number representation (place/value) and arithmetic
  - The number base is 2, instead of 10
  - Binary addition is just like addition in any other base except it has fewer cases … better for circuits
  - All arithmetic and standard calculations have binary equivalents
  - Pixels represented by amount of light intensity
- We conclude: bits “work” for quantities

Bytes – 8 bits in a row

- Bytes illustrate that bits can be grouped in sequence to generate unique patterns
  - 2 bits in sequence, $2^2 = 4$ patterns: 00, 01, 10, 11
  - 4 bits in sequence, $2^4 = 8$ patterns: 0000, 0001, …
  - 8 bits in sequence, 2^8=256 patterns: 0000 0000, …
- ASCII groups 8 bits in sequence
  - They seem to be assigned intelligently, but they're just patterns

ASCII	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0000	NU	SH	SX	EX	ET	EQ	AK	BL	BS	HT	LF	YT	FF	CR	S0	SI
0001	DL	D1	D2	D3	D4	NK	SY	EΣ	CN	EM	SB	EC	FS	GS	RS	US
0010		!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
0011	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
0100	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
0101	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
0110	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
0111	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DT
1000	80	81	82	83	IN	NL	SS	ES	HS	HJ	YS	PD	PV	RI	S2	S3
1001	DC	P1	PZ	SE	CC	MM	SP	EP	Q8	QQ	QA	CS	ST	OS	PM	AP
1010	AO	¡	¢	£	♀	¥	¦	§	¨	©	♂	«	¬	-	®	¯
1011	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
1100	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
1101	Đ	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	β
1110	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
1111	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	Þ	ÿ

Representing Anything

- Compare binary arithmetic to ASCII
  - Binary encodes the positions to make using the information (numbers) easy, like for addition
  - ASCII assigns some pattern to each letter
- Given any finite set of things – colors, computer addresses, English words, etc.
  - We might figure out a smart way to represent them as bits – colors can give light intensity of RGB
  - We can just assign patterns, and manipulate them by pattern matching – red can be 0000 0001, dark red 0000 0010, etc.

Bits Have No Inherent Meaning

- What does this represent:

  0000 0000 1111 0001 0000 1000 0010 0000?
- You don't know until you know how it was encoded
  - As a binary number: 15,796,256
  - As a color, RGB(241,8,32)
  - As a computer instruction: Add 1, 7, 17
  - As ASCII: $^{n}{}_{U}$ $^{b}{}_{s}$ ñ <space>
  - IP Address: 0.241.8.32
  - Hexadecimal number: 00 F1 08 20
  - … → to infinity and beyond

A Bias-free Universal Medium

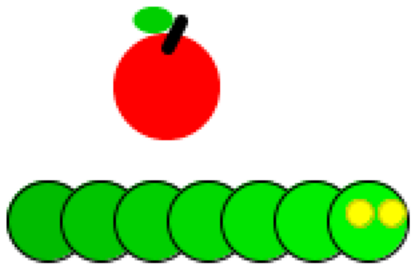
- This is the principle:

Bias-free Universal Medium Principle: Bits can represent all discrete information; bits have no inherent meaning

- Bits are it!!!
- "Computers encode information with bits, not numbers … the bits might be numbers, but they might be a lot of other stuff instead"

Assignment 11 – Two Parts

- Goal



- Part 1: HW 11, due Tuesday
- Part 2: Lab 7, do it in lab

Just Do It!