More Digital Representation

Discrete information is represented in binary (PandA), and "continuous" information is made discrete

---

Return To RGB

Images are constructed from picture elements (pixels); color uses RGB light

The RGB color intensities are specified by 3 numbers in the range (0, 255), ie 1 byte each

Black = (0, 0, 0)	0000 0000	0000 0000	0000 0000
Gray = (128,128,128)	1000 0000	1000 0000	1000 0000
White = (255,255,255)	1111 1111	1111 1111	1111 1111

White-gray-black all have same values for RGB

Colors

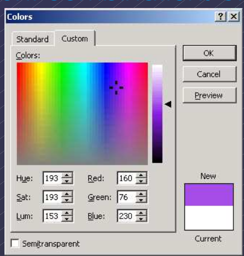
Colors use different combinations of RGB

- Husky Purple

Red=160

Green=76

Blue=230



Positional Notation

The RGB intensities are binary numbers

Binary numbers, like decimal numbers, use *place notation*

1101 = 1x1000 + 1x100 + 0x10 + 1x1

= $1\text{x}10^3$ + $1\text{x}10^2$ + $0\text{x}10^1$ + $1\text{x}10^0$

except that the base is 2 not 10

1101 = 1x8 + 1x4 + 0x2 + 1x1

= $1\text{x}2^3$ + $1\text{x}2^2$ + $0\text{x}2^1$ + $1\text{x}2^0$

Base or radix

1101 in binary is 13 in decimal

Binary Numbers

Given a binary number, add up the powers of 2 corresponding to 1s

1010 0000

$0\text{x}2^0$	= 0x1	= 0
$0\text{x}2^1$	= 0x2	= 0
$0\text{x}2^2$	= 0x4	= 0
$0\text{x}2^3$	= 0x8	= 0
$0\text{x}2^4$	= 0x16	= 0
$1\text{x}2^5$	= 1x32	= 32
$0\text{x}2^6$	= 0x64	= 0
$1\text{x}2^7$	= 1x128	= 128
		= 160

Binary Numbers

Given a binary number, add up the powers of 2 corresponding to 1s

0100 1100

$0\text{x}2^0$	= 0
$0\text{x}2^1$	= 0
$1\text{x}2^2$	= 4
$1\text{x}2^3$	= 8
$0\text{x}2^4$	= 0
$0\text{x}2^5$	= 0
$1\text{x}2^6$	= 64
$0\text{x}2^7$	= 0
	= 76

Binary Numbers

Given a binary number, add up the powers of 2 corresponding to 1s

1110 0110

0×2^0	= 0
1×2^1	= 2
1×2^2	= 4
0×2^3	= 0
0×2^4	= 0
1×2^5	= 32
1×2^6	= 64
1×2^7	= 128
	= 230

Husky Purple

Recall that Husky purple is (160,76,230) which in binary is

1010 0000 0100 1100 1110 0110
160 76 230

Suppose you decide it's not "red" enough

- Increase the red by 16 = 1 0000

```
   1010 0000
 +    1 0000
   1011 0000
```

Adding in binary is pretty much like adding in decimal

A Redder Purple

Increase by 16 more

```
   00110 000   ← Carries
   1011 0000
 +    1 0000
   1100 0000
```

The rule: When the "place sum" equals the radix or more, subtract radix & carry

Find Binary From Decimal

The conversion algorithm

Start: x is the number to convert

1. Let d the largest numbers so $2^d \le x$
2. Is $d \ge 0$, i.e. more digits to process? No, end
3. Is $x \ge 2^d$, i.e. is x at least as large as 2^d?
   - 3t. Yes, the binary place is 1; $x = x - 2^d$
   - 3f. No, the binary place is 0
4. d = d - 1, go to Step 2

Find Binary From Decimal

Start: x is the number to convert

1. Let d the largest numbers so $2^d \le x$
2. Is $d \ge 0$, i.e. more digits to process? No, end
3. Is $x \ge 2^d$, i.e. is x at least large a 2^d?
   - 3t. Y, binary place=1; $x = x - 2^d$
   - 3f. N, binary place=0
4. d = d - 1, go to Step 2

Place	x	2^d	$x \ge 2^d$	bit
7=d	230	128	yes	1
6	102	64	yes	1
5	38	32	yes	1
4	6	16	no	0
3	6	8	no	0
2	6	4	yes	1
1	2	2	yes	1
0	0	1	no	0

1110 0110

Another Example

Convert x = 141 to binary …

1. Let d the largest numbers so $2^d \le x$
2. Is $d \ge 0$, i.e. more digits to process? No, end
3. Is $x \ge 2^d$, i.e. is x at least large a 2^d?
   - 3t. Y, binary place=1; $x = x - 2^d$
   - 3f. N, binary place=0
4. d = d - 1, go to Step 2

Place	x	2^d	$x \ge 2^d$	bit
	141			

Another Example

Convert x = 141 to binary …

1. Let d the largest numbers so $2^d \le x$
2. Is $d \ge 0$, i.e. more digits to process? No, end
3. Is $x \ge 2^d$, i.e. is x at least large a 2^d?
   - 3t. Y, binary place=1;$x=x-2^d$
   - 3f. N, binary place=0
4. d = d - 1, go to Step 2

Place	x	2^d	$x \ge 2^d$	bit
7=d	141	128	yes	1

Another Example

Convert x = 141 to binary …

1. Let d the largest numbers so $2^d \le x$
2. Is $d \ge 0$, i.e. more digits to process? No, end
3. Is $x \ge 2^d$, i.e. is x at least large a 2^d?
   - 3t. Y, binary place=1;$x=x-2^d$
   - 3f. N, binary place=0
4. d = d - 1, go to Step 2

Place	x	2^d	$x \ge 2^d$	bit
7=d	141	128	yes	1
6	13			

Another Example

Convert x = 141 to binary …

1. Let d the largest numbers so $2^d \le x$
2. Is $d \ge 0$, i.e. more digits to process? No, end
3. Is $x \ge 2^d$, i.e. is x at least large a 2^d?
   - 3t. Y, binary place=1;$x=x-2^d$
   - 3f. N, binary place=0
4. d = d - 1, go to Step 2

Place	x	2^d	$x \ge 2^d$	bit
7=d	141	128	yes	1
6	13	64	no	0
5	13			

Another Example

Convert x = 141 to binary …

1. Let d the largest numbers so $2^d \le x$
2. Is $d \ge 0$, i.e. more digits to process? No, end
3. Is $x \ge 2^d$, i.e. is x at least large a 2^d?
   - 3t. Y, binary place=1;$x=x-2^d$
   - 3f. N, binary place=0
4. d = d - 1, go to Step 2

Place	x	2^d	$x \ge 2^d$	bit
7=d	141	128	yes	1
6	13	64	no	0
5	13	32	no	0
4	13	16	no	0
3	13			

Another Example

Convert x = 141 to binary …

1. Let d the largest numbers so $2^d \le x$
2. Is $d \ge 0$, i.e. more digits to process? No, end
3. Is $x \ge 2^d$, i.e. is x at least large a 2^d?
   - 3t. Y, binary place=1;$x=x-2^d$
   - 3f. N, binary place=0
4. d = d - 1, go to Step 2

Place	x	2^d	$x \ge 2^d$	bit
7=d	141	128	yes	1
6	13	64	no	0
5	13	32	no	0
4	13	16	no	0
3	13	8	yes	1
2	5			

Another Example

Convert x = 141 to binary …

1. Let d the largest numbers so $2^d \le x$
2. Is $d \ge 0$, i.e. more digits to process? No, end
3. Is $x \ge 2^d$, i.e. is x at least large a 2^d?
   - 3t. Y, binary place=1;$x=x-2^d$
   - 3f. N, binary place=0
4. d = d - 1, go to Step 2

Place	x	2^d	$x \ge 2^d$	bit
7=d	141	128	yes	1
6	13	64	no	0
5	13	32	no	0
4	13	16	no	0
3	13	8	yes	1
2	5	4	yes	1
1	1			

Another Example

Convert x = 141 to binary …

1. Let d the largest numbers so $2^d \le x$
2. Is $d \ge 0$, i.e. more digits to process? No, end
3. Is $x \ge 2^d$, i.e. is x at least large a 2^d?
   - 3t. Y, binary place=1;$x=x-2^d$
   - 3f. N, binary place=0
4. d = d - 1, go to Step 2

Place	x	2^d	$x \ge 2^d$	bit
7=d	141	128	yes	1
6	13	64	no	0
5	13	32	no	0
4	13	16	no	0
3	13	8	yes	1
2	5	4	yes	1
1	1	2	no	0
0	1	1	yes	1

Another Example

Convert x = 141 to binary …

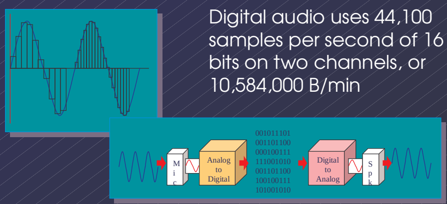
1. Let d the largest numbers so $2^d \le x$
2. Is $d \ge 0$, i.e. more digits to process? No, end
3. Is $x \ge 2^d$, i.e. is x at least large a 2^d?
   - 3t. Y, binary place=1;$x=x-2^d$
   - 3f. N, binary place=0
4. d = d - 1, go to Step 2

Place	x	2^d	$x \ge 2^d$	bit
7=d	141	128	yes	1
6	13	64	no	0
5	13	32	no	0
4	13	16	no	0
3	13	8	yes	1
2	5	4	yes	1
1	1	2	no	0
0	1	1	yes	1

1000 1101

Digitizing

"Continuous" information like light and sound must be made "discrete"



Digital audio uses 44,100 samples per second of 16 bits on two channels, or 10,584,000 B/min

Information Processing

Manipulating pixels is an example of "computing on a representation"

- Photoshop & other graphics SW manipulate pictures by computing on representation
- Audio is edited similarly to remove coughs and other odd sounds, speed up, etc.
- Searching the dictionary is another example

Information processing depends on computing on representations

Bits Are It

Bits represent information, but their interpretation gives bits meaning

0000 0000 1111 0001 0000 1000 0010 0000

- Could be a number, color, instruction, ASCII, sound samples, IP address, …

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

Summary

Bits can represent any information

- Discrete information is directly encoded using binary
- Continuous information is made discrete
- "Computing on representations" is the key to "information processing"
- Bias-free Universal Medium Principle