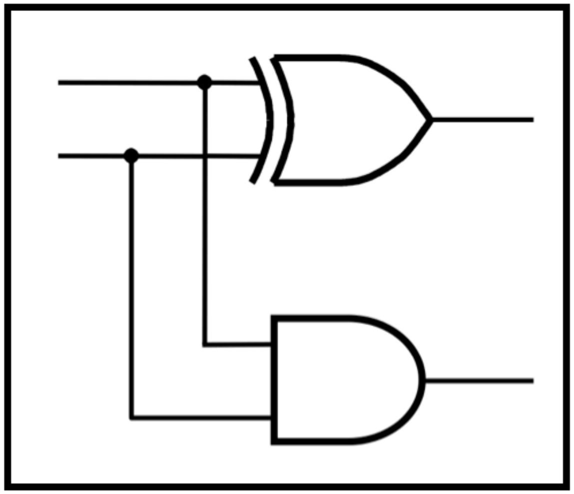


CprE 281: Digital Logic

Instructor: Alexander Stoytchev

http://www.ece.iastate.edu/~alexs/classes/

Binary Numbers

CprE 281: Digital Logic
Iowa State University, Ames, IA

Administrative Stuff

This is the official class web page:

http://www.ece.iastate.edu/~alexs/classes/2016_Fall_281/

If you missed the first lecture, the syllabus and other class materials are posted there.

Administrative Stuff

- HW1 is out

- It is due on Monday Aug 29 @ 4pm.

- Submit it on paper before the start of the lecture

Administrative Stuff

The labs and recitations start next week:

- **Section N: Monday 9:00 AM - 11:50 AM (Coover Hall, room 1318)**
- **Section P: Monday 12:10 PM - 3:00 PM (Coover Hall, room 1318)**
- **Section R: Monday 5:10 PM - 8:00 PM (Coover Hall, room 1318)**
- **Section U: Tuesday 11:00 AM - 1:50 PM (Coover Hall, room 2050)**
- **Section M: Tuesday 2:10 PM - 5:00 PM (Coover Hall, room 2050)**
- **Section J: Wednesday 8:00 AM - 10:50 AM (Coover Hall, room 2050)**
- **Section T: Wednesday 6:10 PM - 9:00 PM (Coover Hall, room 1318)**
- **Section Q: Thursday 11:00 AM - 1:50 PM (Coover Hall, room 2050)**
- **Section V: Thursday 11:00 AM - 1:50 PM (Coover Hall, room 1318)**
- **Section L: Thursday 2:10 PM - 5:00 PM (Coover Hall, room 2050)**
- **Section K: Thursday 5:10 PM - 8:00 PM (Coover Hall, room 2050)**
- **Section G: Friday 11:00 AM - 1:50 PM (Coover Hall, room 2050)**

- **The lab schedule is also posted on the class web page**

Labs Next Week

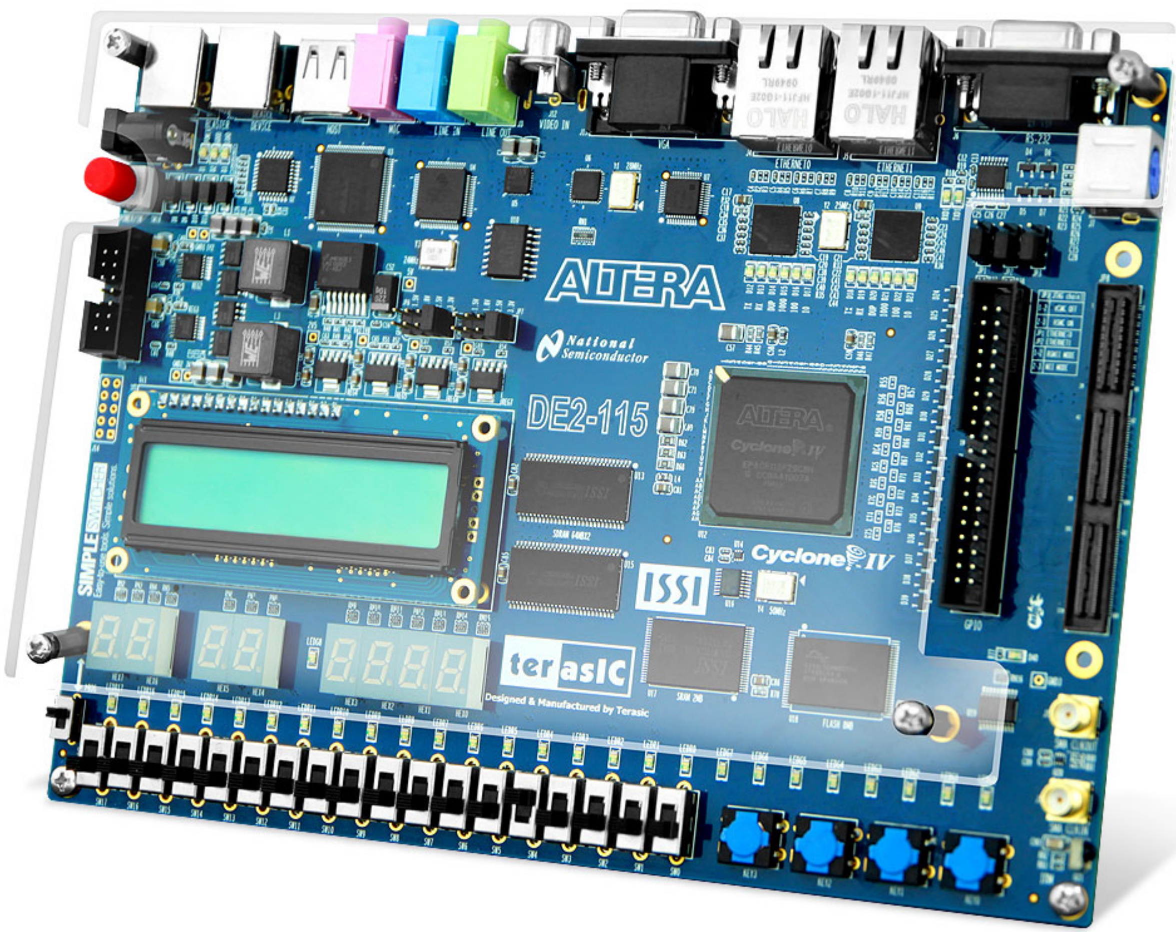
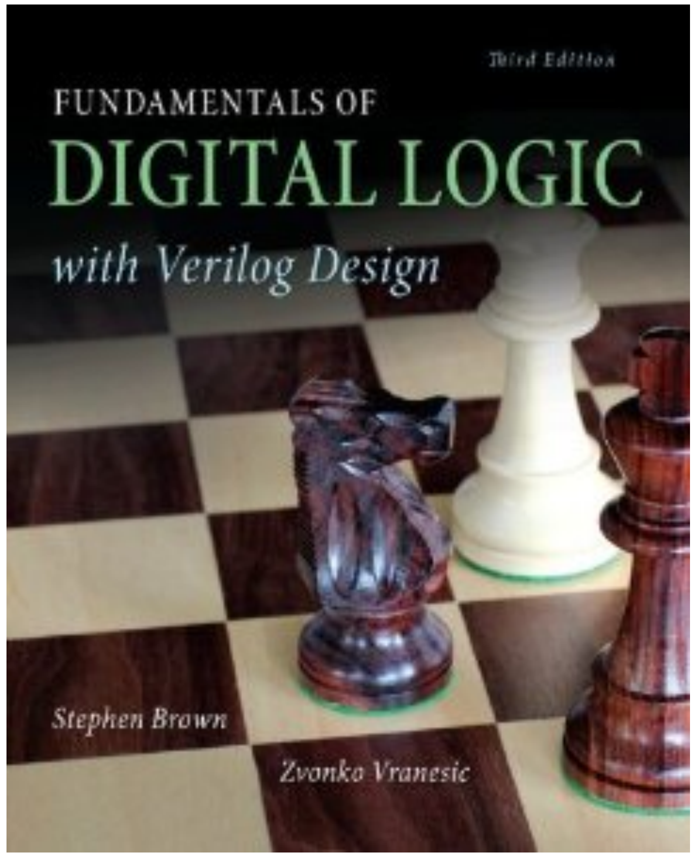


Figure 1.5 in the textbook: An FPGA board.

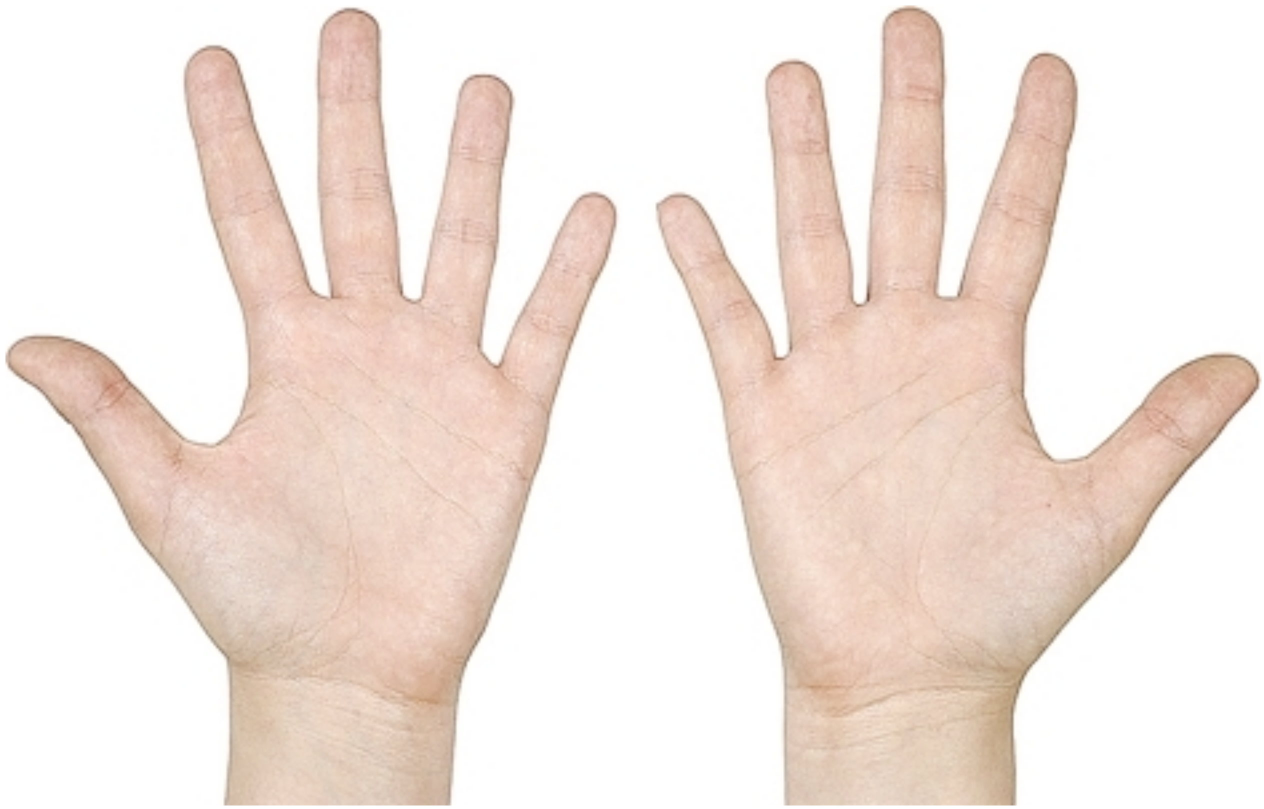
Labs Next Week

- **Please download and read the lab assignment for next week before you go to your lab section.**
- **You must print the answer sheet and do the prelab before you go to the lab.**
- **The TAs will check your answers at the beginning of the lab.**

Did you get the textbook?



The Decimal System



[http://www.chompchomp.com/images/irregular011.jpg]

What number system is this one?

[http://freedomhygiene.com/wp-content/themes/branfordmagazine/images/backgrounds/Hands_141756.jpg]

The Binary System

[http://divaprojections.blogspot.com/2011/11/alien.html]

Number Systems

$$N = d_n B^n + d_{n-1} B^{n-1} + \cdots + d_1 B^1 + d_0 B^0$$

Number Systems

$$N = d_n B^n + d_{n-1} B^{n-1} + \cdots + d_1 B^1 + d_0 B^0$$

n-th digit
(most significant)

0-th digit
(least significant)

Number Systems

base

power

$$N = d_n B^n + d_{n-1} B^{n-1} + \cdots + d_1 B^1 + d_0 B^0$$

n-th digit
(most significant)

0-th digit
(least significant)

The Decimal System

$$524_{10} = 5 \times 10^2 + 2 \times 10^1 + 4 \times 10^0$$

The Decimal System

$$524_{10} = 5\times10^2+2\times10^1+4\times10^0$$

$$= 5\times100+2\times10+4\times1$$

$$= 500+20+4$$

$$= 524_{10}$$

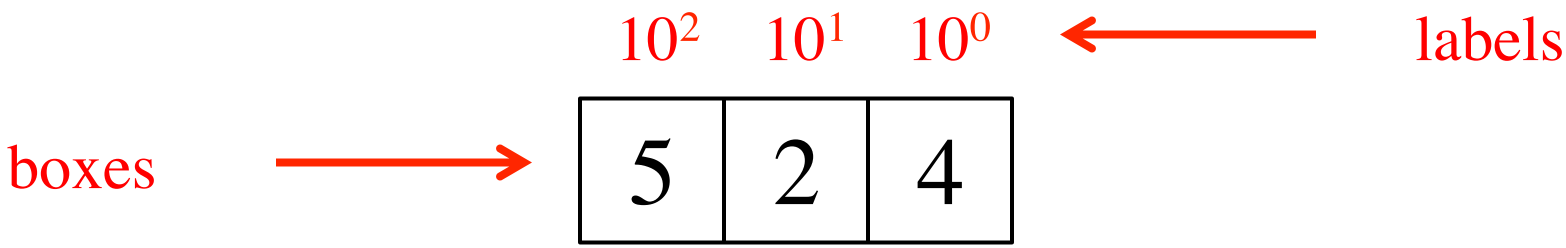
Another Way to Look at This

5	2	4

Another Way to Look at This

10^2	10^1	10^0
5	2	4

Another Way to Look at This



Each box can contain only one digit and has only one label. From right to left, the labels are increasing powers of the base, starting from 0.

Base 7

$$524_7 = 5 \times 7^2 + 2 \times 7^1 + 4 \times 7^0$$

Base 7

base power

$$524_7 = 5 \times 7^2 + 2 \times 7^1 + 4 \times 7^0$$

Base 7

base power

$$524_7 = 5 \times 7^2 + 2 \times 7^1 + 4 \times 7^0$$

most significant digit

least significant digit

Base 7

$$
\begin{aligned}
524_7 &= 5 \times 7^2 + 2 \times 7^1 + 4 \times 7^0 \\
&= 5 \times 49 + 2 \times 7 + 4 \times 1 \\
&= 245 + 14 + 4 \\
&= 263_{10}
\end{aligned}
$$

Another Way to Look at This

7^2	7^1	7^0
5	2	4

=

10^2	10^1	10^0
2	6	3

Binary Numbers (Base 2)

$$1001_2 = \quad 1 \times 2^3 \quad + \quad 0 \times 2^2 \quad + \quad 0 \times 2^1 \quad + \quad 1 \times 2^0$$

Binary Numbers (Base 2)

base

power

$$1001_2 = 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$

most significant bit

least significant bit

Binary Numbers (Base 2)

$$
\begin{aligned}
1001_2 = \quad & 1 \times 2^3 & + \quad & 0 \times 2^2 & + \quad & 0 \times 2^1 & + \quad & 1 \times 2^0 & = \\
= \quad & 1 \times 8 & + \quad & 0 \times 4 & + \quad & 0 \times 2 & + \quad & 1 \times 1 & = \\
= \quad & 8 & + \quad & 0 & + \quad & 0 & + \quad & 1 & = \\
= \quad & 9_{10}
\end{aligned}
$$

Another Example

$$\begin{aligned}
11101_2 &= 1\times 2^4 + 1\times 2^3 + 1\times 2^2 + 0\times 2^1 + 1\times 2^0 = \\
&= 1\times 16 + 1\times 8 + 1\times 4 + 0\times 2 + 1\times 1 = \\
&= 16 + 8 + 4 + 0 + 1 = 29_{10}
\end{aligned}$$

Powers of 2

$$
\begin{aligned}
2^{10} &= 1024 \\
2^{9} &= 512 \\
2^{8} &= 256 \\
2^{7} &= 128 \\
2^{6} &= 64 \\
2^{5} &= 32 \\
2^{4} &= 16 \\
2^{3} &= 8 \\
2^{2} &= 4 \\
2^{1} &= 2 \\
2^{0} &= 1
\end{aligned}
$$

What is the value of this binary number?

- 0 0 1 0 1 1 0 0
- 0 0 1 0 1 1 0 0
- $0*2^7 + 0*2^6 + 1*2^5 + 0*2^4 + 1*2^3 + 1*2^2 + 0*2^1 + 0*2^0$
- 0*128 + 0*64 + 1*32 + 0*16 + 1*8 + 1*4 + 0*2 + 0*1
- 0*128 + 0*64 + 1*32 + 0*16 + 1*8 + 1*4 + 0*2 + 0*1
- 32+ 8 + 4 = 44 (in decimal)

Another Way to Look at This

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
0	0	1	0	1	1	0	0

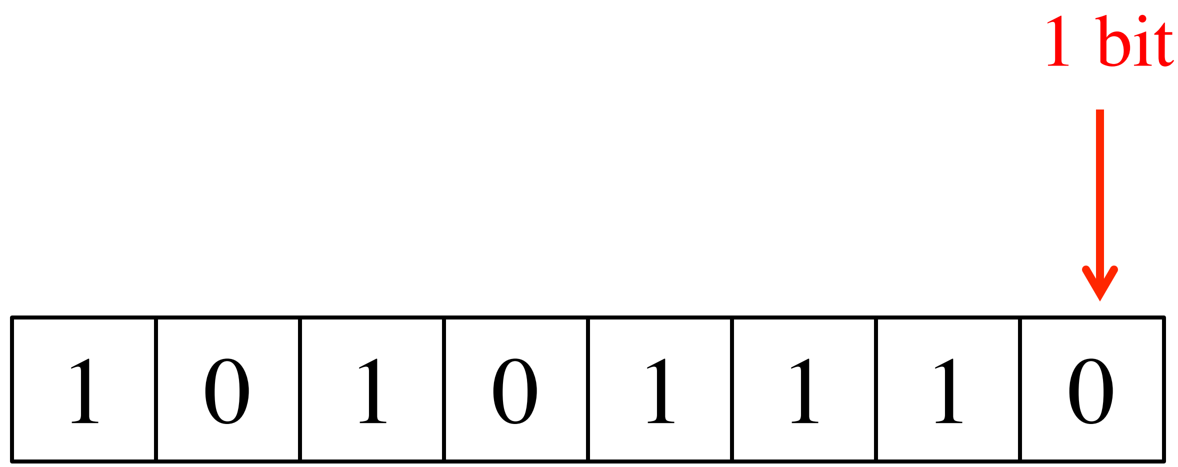
Some Terminology

- **A binary digit is called a *bit***
- **A group of eight bits is called a byte**
- **One bit can represent only two possible states, which are denoted with 1 and 0**

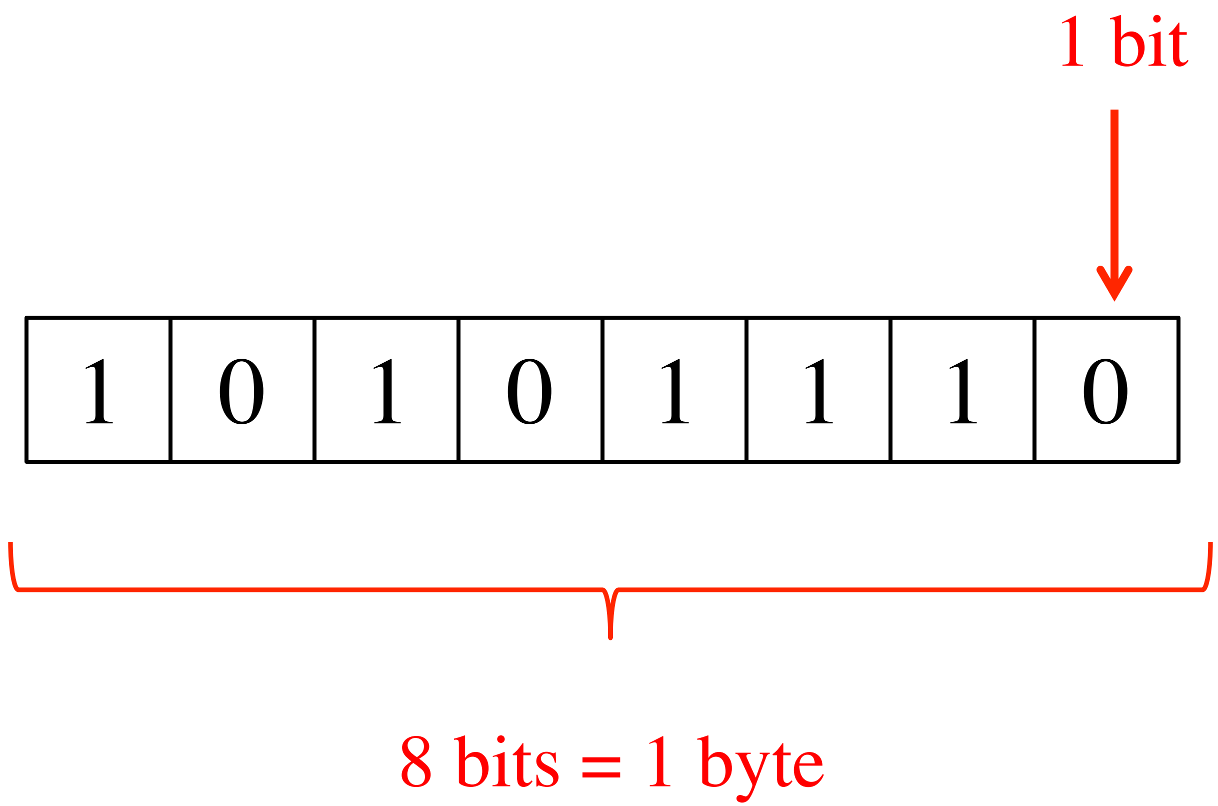
Relationship Between a Byte and a Bit

1	0	1	0	1	1	1	0

Relationship Between a Byte and a Bit



Relationship Between a Byte and a Bit



Bit Permutations

1 bit	2 bits	3 bits	4 bits	
0	00	000	0000	1000
1	01	001	0001	1001
	10	010	0010	1010
	11	011	0011	1011
		100	0100	1100
		101	0101	1101
		110	0110	1110
		111	0111	1111

Each additional bit doubles the number of possible permutations

Bit Permutations

- Each permutation can represent a particular item
- There are 2^N permutations of N bits
- Therefore, N bits are needed to represent 2^N unique items

How many items can be represented by

1 bit ?	2^1 = 2 items
2 bits ?	2^2 = 4 items
3 bits ?	2^3 = 8 items
4 bits ?	2^4 = 16 items
5 bits ?	2^5 = 32 items

What is the maximum number that can be stored in one byte (8 bits)?

What is the maximum number that can be stored in one byte (8 bits)?

- 1 1 1 1 1 1 1 1
- 1 1 1 1 1 1 1 1
- $1*2^7 + 1*2^6 + 1*2^5 + 1*2^4 + 1*2^3 + 1*2^2 + 1*2^1 + 1*2^0$
- 1*128 + 1*64 + 1*32 + 1*16 + 1*8 + 1*4 + 1*2 + 1*1
- 128 + 64 + 32 + 16 + 8 + 4 + 2 + 1 = 255 (in decimal)
- Another way is: $1*2^8 – 1 = 256 – 1 = 255$

What would happen if we try to add 1 to the largest number that can be stored in one byte (8 bits)?

```
      1 1 1 1 1 1 1 1
+
                    1
      ---------------
    1 0 0 0 0 0 0 0 0

      0 0 0 0 0 0 0 0
```

Analogy with car odometers

Analogy with car odometers

[http://www.hyperocity.com/volvo240/images/Volvo/odometerrepair/speedo999999.jpg]

Decimal to Binary Conversion (Using Guessing)

$$17 = 16 + 1 \rightarrow 10001_2$$

$$\begin{array}{rcrl}
2^7 & = & 128 & \\
2^6 & = & 64 & \\
2^5 & = & 32 & \\
2^4 & = & 16 & \checkmark \\
2^3 & = & 8 & \\
2^2 & = & 4 & \\
2^1 & = & 2 & \\
2^0 & = & 1 & \checkmark
\end{array}$$

Decimal to Binary Conversion (Using Guessing)

$$212 = 128 + 64 + 16 + 4 \rightarrow 11010100_2$$

$$\begin{array}{rcrl}
2^7 & = & 128 & \checkmark \\
2^6 & = & 64 & \checkmark \\
2^5 & = & 32 & \\
2^4 & = & 16 & \checkmark \\
2^3 & = & 8 & \\
2^2 & = & 4 & \checkmark \\
2^1 & = & 2 & \\
2^0 & = & 1 &
\end{array}$$

Converting from Decimal to Binary

				result	remainder
235	/	2	=	117	1
117	/	2	=	58	1
58	/	2	=	29	0
29	/	2	=	14	1
14	/	2	=	7	0
7	/	2	=	3	1
3	/	2	=	1	1
1	/	2	=	0	1

Converting from Decimal to Binary

				result	*remainder*
235	/	2	=	117	1
117	/	2	=	58	1
58	/	2	=	29	0
29	/	2	=	14	1
14	/	2	=	7	0
7	/	2	=	3	1
3	/	2	=	1	1
1	/	2	=	0	1

$$235_{10} = 11101011_2$$

Convert $(857)_{10}$

			Remainder	
$857 \div 2$	=	428	1	LSB
$428 \div 2$	=	214	0	
$214 \div 2$	=	107	0	
$107 \div 2$	=	53	1	
$53 \div 2$	=	26	1	
$26 \div 2$	=	13	0	
$13 \div 2$	=	6	1	
$6 \div 2$	=	3	0	
$3 \div 2$	=	1	1	
$1 \div 2$	=	0	1	MSB

Result is $(1101011001)_2$

[Figure 1.6 in the textbook]

Octal System (Base 8)

0	1	2	3	4	5	6	7
10	11	12	13	14	15	16	17
20	21	22	23	24	25	26	27
30	31	32	33	34	35	36	37
40	41	42	43	44	45	46	47
50	51	52	53	54	55	56	57
60	61	62	63	64	65	66	67
70	71	72	73	74	75	76	77

Binary to Octal Conversion

$$
\begin{aligned}
000 &\rightarrow 0 \\
001 &\rightarrow 1 \\
010 &\rightarrow 2 \\
011 &\rightarrow 3 \\
100 &\rightarrow 4 \\
101 &\rightarrow 5 \\
110 &\rightarrow 6 \\
111 &\rightarrow 7
\end{aligned}
$$

Binary to Octal Conversion

$$101110010111_2 = ?_8$$

Binary to Octal Conversion

$$101110010111_2 = ?_8$$

101 110 010 111

Binary to Octal Conversion

$$101110010111_2 = ?_8$$

$$\underbrace{101}_{5}\ \underbrace{110}_{6}\ \underbrace{010}_{2}\ \underbrace{111}_{7}$$

Binary to Octal Conversion

$$101110010111_2 = ?_8$$

$$\underbrace{101}_{5}\ \underbrace{110}_{6}\ \underbrace{010}_{2}\ \underbrace{111}_{7}$$

Thus, $101110010111_2 = 5627_8$

Hexadecimal System (Base 16)

$$52_{16} = 5 \times 16^1 + 2 \times 16^0 =$$

$$5 \times 16 + 2 \times 1 =$$

$$80 + 2 = 82_{10}$$

The 16 Hexadecimal Digits

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

The 16 Hexadecimal Digits

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

10, 11, 12, 13, 14, 15

Hexadecimal to Decimal Conversion

$$C3_{16} = C \times 16^1 + 3 \times 16^0$$

$$= 12 \times 16 + 3 \times 1$$

$$= 192 + 3$$

$$= 195_{10}$$

Hexadecimal to Decimal Conversion

$$BEEF_{16} = ?_{10}$$

Hexadecimal to Decimal Conversion

$$
\begin{aligned}
BEEF_{16} &= B_{16} \times 16^3 &+& \quad E_{16} \times 16^2 &+& \quad E_{16} \times 16^1 &+& \quad F_{16} \times 16^0 \\
&= 11 \times 16^3 &+& \quad 14 \times 16^2 &+& \quad 14 \times 16^1 &+& \quad 15 \times 16^0 \\
&= 11 \times 4096 &+& \quad 14 \times 256 &+& \quad 14 \times 16 &+& \quad 15 \times 1 \\
&= 45056 &+& \quad 3584 &+& \quad 224 &+& \quad 15 \\
&= 48879_{10}
\end{aligned}
$$

Binary to Hexadecimal Conversion

$$
\begin{array}{rcc}
0000 & \rightarrow & 0 \\
0001 & \rightarrow & 1 \\
0010 & \rightarrow & 2 \\
0011 & \rightarrow & 3 \\
0100 & \rightarrow & 4 \\
0101 & \rightarrow & 5 \\
0110 & \rightarrow & 6 \\
0111 & \rightarrow & 7 \\
1000 & \rightarrow & 8 \\
1001 & \rightarrow & 9 \\
1010 & \rightarrow & A \\
1011 & \rightarrow & B \\
1100 & \rightarrow & C \\
1101 & \rightarrow & D \\
1110 & \rightarrow & E \\
1111 & \rightarrow & F
\end{array}
$$

Binary to Hexadecimal Conversion

$$
\begin{array}{rcrcr}
0000 & \rightarrow & 0 & \rightarrow & 0 \\
0001 & \rightarrow & 1 & \rightarrow & 1 \\
0010 & \rightarrow & 2 & \rightarrow & 2 \\
0011 & \rightarrow & 3 & \rightarrow & 3 \\
0100 & \rightarrow & 4 & \rightarrow & 4 \\
0101 & \rightarrow & 5 & \rightarrow & 5 \\
0110 & \rightarrow & 6 & \rightarrow & 6 \\
0111 & \rightarrow & 7 & \rightarrow & 7 \\
1000 & \rightarrow & 8 & \rightarrow & 8 \\
1001 & \rightarrow & 9 & \rightarrow & 9 \\
1010 & \rightarrow & 10 & \rightarrow & A \\
1011 & \rightarrow & 11 & \rightarrow & B \\
1100 & \rightarrow & 12 & \rightarrow & C \\
1101 & \rightarrow & 13 & \rightarrow & D \\
1110 & \rightarrow & 14 & \rightarrow & E \\
1111 & \rightarrow & 15 & \rightarrow & F
\end{array}
$$

Binary to Hexadecimal Conversion

$$101110010111_2 = ?_{16}$$

Binary to Hexadecimal Conversion

$$101110010111_2 = ?_{16}$$

1011 1001 0111

Binary to Hexadecimal Conversion

$$101110010111_2 = ?_{16}$$

$$\underbrace{1011}_{B}\quad\underbrace{1001}_{9}\quad\underbrace{0111}_{7}$$

Binary to Hexadecimal Conversion

$$101110010111_2 = ?_{16}$$

$$\underbrace{1011}_{B}\quad\underbrace{1001}_{9}\quad\underbrace{0111}_{7}$$

Thus, $101110010111_2 = B97_{16}$

Decimal to Hexadecimal Conversion

$$1396_{10} = 574_{16}$$

				result	*remainder*
1396	/	16	=	87	4
87	/	16	=	5	7
5	/	16	=	0	5

Decimal to Hexadecimal Conversion

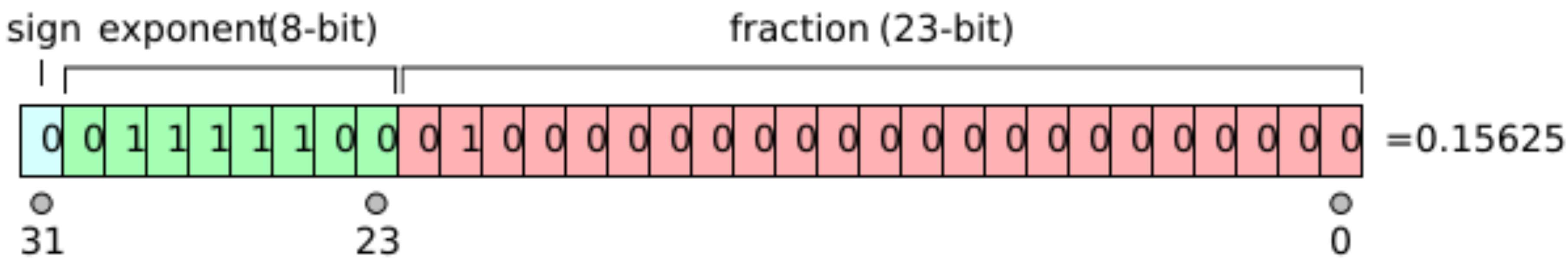
$$502_{10} = 1F6_{16}$$

$$\begin{array}{rcrcrr} & & & & result & remainder \\ \hline 502 & / & 16 & = & 31 & 6 \\ 31 & / & 16 & = & 1 & 15 \\ 1 & / & 16 & = & 0 & 1 \end{array}$$

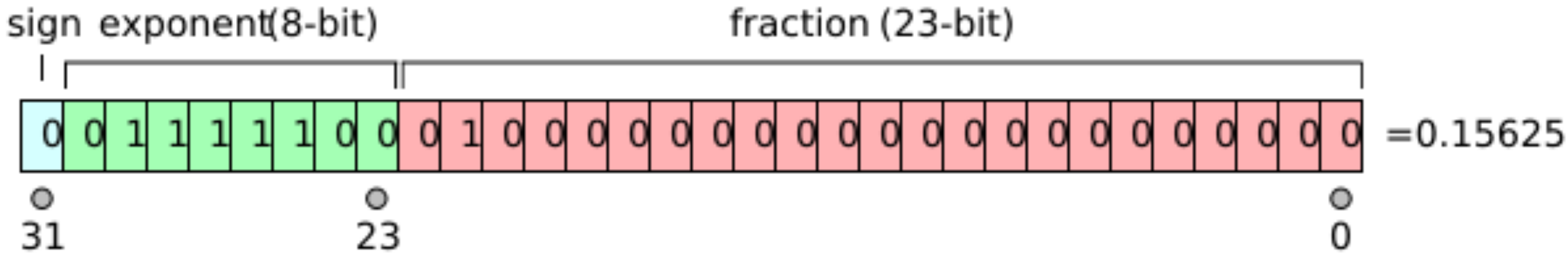
Signed integers are more complicated

We will talk more about them when we start with Chapter 3 in a couple of weeks.

The story with floats is even more complicated
IEEE 754-1985 Standard



[http://en.wikipedia.org/wiki/IEEE_754]



$$v = (-1)^{sign} \times 2^{exponent-exponent\ bias} \times 1.fraction$$

s = +1 (positive numbers and +0) when the sign bit is 0

s = −1 (negative numbers and −0) when the sign bit is 1

e = *exponent* − 127 (in other words the exponent is stored with 127 added to it, also called "biased with 127")

In the example shown above, the *sign* is zero so s is +1, the *exponent* is 124 so e is −3, and the significand m is 1.01 (in binary, which is 1.25 in decimal). The represented number is therefore +1.25 × 2^{-3}, which is +0.15625.

[http://en.wikipedia.org/wiki/IEEE_754]

On-line IEEE 754 Converter

- http://www.h-schmidt.net/FloatApplet/IEEE754.html

- More about floating point numbers in Chapter 3.

Storing Characters

- This requires some convention that maps binary numbers to characters.
- ASCII table
- Unicode

ASCII Table

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Source: www.LookupTables.com

Extended ASCII Codes

128	Ç	144	É	161	í	177	▒	193	┴	209	╤	225	ß	241	±
129	ü	145	æ	162	ó	178	▓	194	┬	210	╥	226	Γ	242	≥
130	é	146	Æ	163	ú	179	│	195	├	211	╙	227	π	243	≤
131	â	147	ô	164	ñ	180	┤	196	─	212	╘	228	Σ	244	⌠
132	ä	148	ö	165	Ñ	181	╡	197	┼	213	╒	229	σ	245	⌡
133	à	149	ò	166	ª	182	╢	198	╞	214	╓	230	µ	246	÷
134	å	150	û	167	º	183	╖	199	╟	215	╫	231	τ	247	≈
135	ç	151	ù	168	¿	184	╕	200	╚	216	╪	232	Φ	248	°
136	ê	152	_	169	_	185	╣	201	╔	217	┘	233	Θ	249	·
137	ë	153	Ö	170	¬	186	║	202	╩	218	┌	234	Ω	250	·
138	è	154	Ü	171	½	187	╗	203	╦	219	█	235	δ	251	√
139	ï	156	£	172	¼	188	╝	204	╠	220	▄	236	∞	252	_
140	î	157	¥	173	¡	189	╜	205	═	221	▌	237	φ	253	²
141	ì	158	_	174	«	190	╛	206	╬	222	▐	238	ε	254	■
142	Ä	159	ƒ	175	»	191	┐	207	╧	223	▀	239	∩	255	
143	Å	160	á	176	░	192	└	208	╨	224	α	240	≡		

Source: www.LookupTables.com

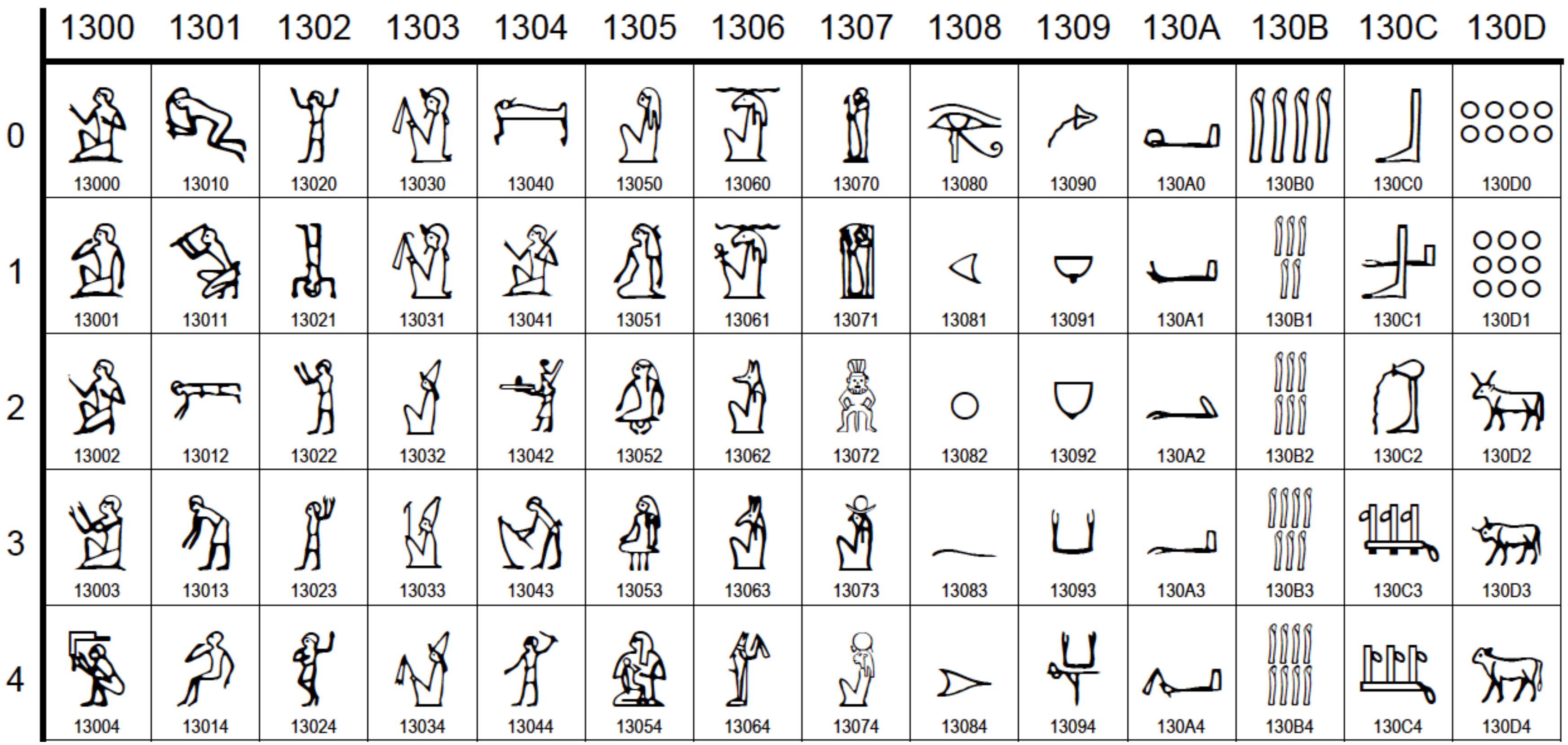
The Unicode Character Code

- http://www.unicode.org/charts/

Egyptian Hieroglyphs

http://www.unicode.org/charts/

Close up



http://www.unicode.org/charts/

Questions?

THE END