Introduction to Digital Design

Week 1: Introduction

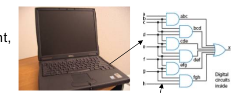
Yao Zheng
University of Hawai'i at Mānoa
Department of Electrical and Computer Engineering

1

1.1

Why Study Digital Design?

- Look "under the hood" of computers
  - Solid understanding --> confidence, insight, even better programmer when aware of hardware resource issues
- Electronic devices becoming digital
  - Enabled by shrinking and more capable chips
  - Enables:
    - Better devices: Sound recorders, cameras, cars, cell phones, medical devices,...
    - New devices: Video games, virtual reality...
  - Known as "embedded systems", SoC,
    - Thousands of new devices every year
    - Designers needed: Potential career direction



Satellites, DVD players, Video recorders, Musical instruments
Portable music players, Cell phones, Cameras, TVs, ???
1995 1997 1999 2001 2003 2005 2007

- Years shown above indicate when digital version began to *dominate*
  - (*Not* the first year that a digital version appeared)

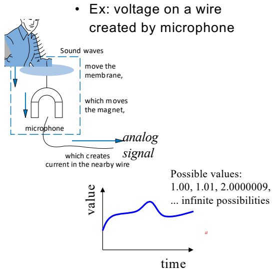
2

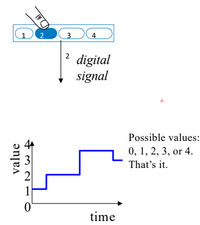
2

1.2

What Does "Digital" Mean?

- Analog signal
  - Infinite possible values
    - Ex: voltage on a wire created by microphone
- Digital signal
  - Finite possible values
    - Ex: button pressed on a keypad



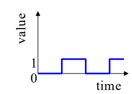


3

3

Digital Signals with Only Two Values: Binary

- ***Binary*** digital signal -- only *two* possible values
  - Typically represented as **0** and **1**
  - One ***b****inary dig****it*** is a ***bit***
  - We'll only consider *binary* digital signals
  - Binary is popular because
    - Transistors, the basic digital electric component, operate using *two* voltages (more in Chpt. 2)
    - Storing/transmitting one of *two* values is easier than three or more (e.g., loud beep or quiet beep, reflection or no reflection)



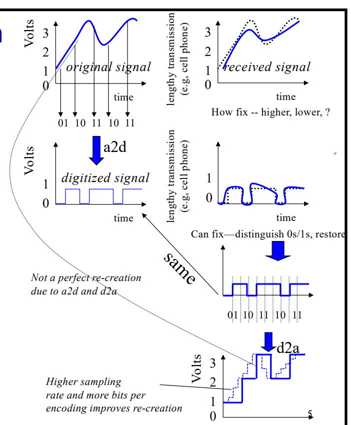
4

4

Example of Digitization Benefit

- Analog signal (e.g., audio, video) may lose quality
  - Voltage levels not saved/copied/transmitted perfectly
- Digitized version enables near-perfect save/cpy/tran.
  - "Sample" voltage at particular rate, save sample using bit encoding
  - Voltage levels still not kept perfectly
  - But we can distinguish 0s from 1s

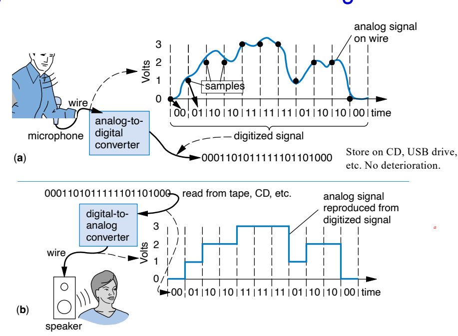
Let bit encoding be:
1 V: "01"
2 V: "10"
3 V: "11"



5

5

Digitization Benefit: Can Store on Digital Media



6

6

Digitized Audio: Compression Benefit

- Digitized audio can be compressed
  - e.g., MP3s
  - A CD can hold about 20 songs uncompressed, but about 200 compressed
- Compression also done on digitized pictures (jpeg), movies (mpeg), and more
- Digitization has many other benefits too

Example compression scheme:
00 means 0000000000
01 means 1111111111
1X means X

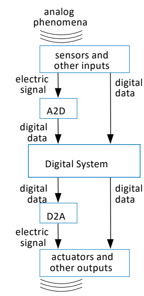
0000000000 0000000000 0000001111 1111111111

00 00 10000001111 01

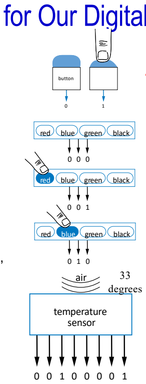
7

7

How Do We Encode Data as Binary for Our Digital System?



- Some inputs inherently binary
  - Button: not pressed (0), pressed (1)
- Some inputs inherently digital
  - Just need encoding in binary
  - e.g., multi-button input: encode red=001, blue=010, ...
- Some inputs analog
  - Need analog-to-digital conversion
  - As done in earlier slide -- sample and encode with bits



8

8

How to Encode Text: ASCII, Unicode

- ASCII: 7- (or 8-) bit encoding of each letter, number, or symbol
- Unicode: Increasingly popular 16-bit encoding
  - Encodes characters from various world languages

Sample ASCII encodings

Encoding	Symbol
010 0000	<space>
010 0001	!
010 0010	"
010 0011	#
010 0100	$
010 0101	%
010 0110	&
010 0111	'
010 1000	(
010 1001	)
010 1010	*
010 1011	+
010 1100	,
010 1101	-
010 1110	.
010 1111	/

Encoding	Symbol
100 0001	A
100 0010	B
100 0011	C
100 0100	D
100 0101	E
100 0110	F
100 0111	G
100 1000	H
100 1001	I
100 1010	J
100 1011	K
100 1100	L
100 1101	M

Encoding	Symbol
100 1110	N
100 1111	O
101 0000	P
101 0001	Q
101 0010	R
101 0011	S
101 0100	T
101 0101	U
101 0110	V
101 0111	W
101 1000	X
101 1001	Y
101 1010	Z

Encoding	Symbol
110 0001	a
110 0010	b
...	
111 1001	y
111 1010	z
011 0000	0
011 0001	1
011 0010	2
011 0011	3
011 0100	4
011 0101	5
011 0110	6
011 0111	7
011 1000	8
011 1001	9

Question:
What does this ASCII bit sequence represent?

1010010 1000101 1010011 1010100

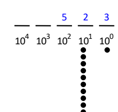
R E S T

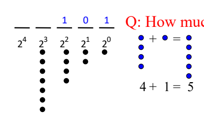
9

9

How to Encode Numbers: Binary Numbers

- Each position represents a quantity; symbol in position means how many of that quantity
  - Base ten (***decimal***)
    - Ten symbols: 0, 1, 2, ..., 8, and 9
    - More than 9 -- next position
      - So each position power of 10
    - Nothing special about base 10 -- used because we have 10 fingers
  - Base two (***binary***)
    - Two symbols: 0 and 1
    - More than 1 -- next position
      - So each position power of 2



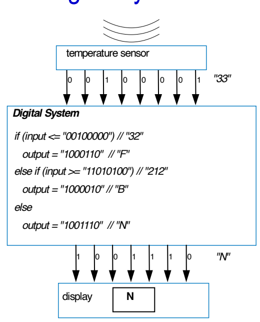


10

10

Using Digital Data in a Digital System

- A temperature sensor outputs temperature in binary
- The system reads the temperature, outputs ASCII code:
  - "F" for freezing (0-32)
  - "B" for boiling (212 or more)
  - "N" for normal
- A display converts its ASCII input to the corresponding letter



11

11

Bytes, Kilobytes, Megabytes, and More

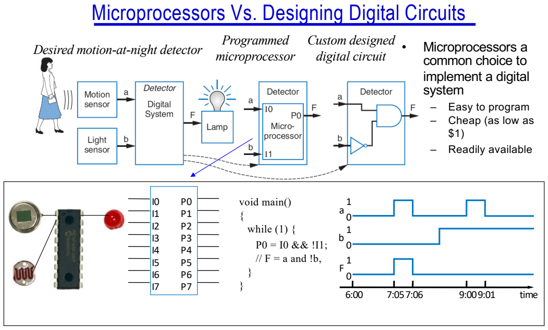
- Byte: 8 bits
- Common metric prefixes:
  - kilo (thousand, or 10^3), mega (million, or 10^6), giga (billion, or 10^9), and tera (trillion, or 10^{12}), e.g., kilobyte, or KByte
- BUT, metric prefixes also commonly used inaccurately
  - 2^{16} = 65536 commonly written as "64 Kbyte"
  - Typical when describing memory sizes
- Also watch out for "KB" for kilobyte vs. "Kb" for kilobit

12

12

Implementing Digital Systems: Programming Microprocessors Vs. Designing Digital Circuits

1.3



- Microprocessors a common choice to implement a digital system
  - Easy to program
  - Cheap (as low as $1)
  - Readily available

13

13

Digital Design: When Microprocessors Aren't Good Enough

- With microprocessors so easy, cheap, and available, why design a digital circuit?
  - Microprocessor may be too slow
  - Or too big, power hungry, or costly

Wing controller computation task:

- 50 ms on microprocessor
- 5 ms as custom digital circuit

If must execute 100 times per second:

- 100 * 50 ms = 5000 ms = 5 seconds
- 100 * 5 ms = 500 ms = 0.5 seconds

Microprocessor too slow, circuit OK.

14

14

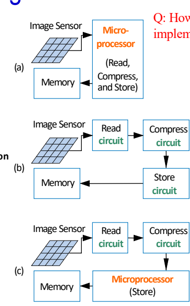
Digital Design: When Microprocessors Aren't Good Enough

- Commonly, designers partition a system among a microprocessor and custom digital circuits

Sample digital camera task execution times (in seconds) on a microprocessor versus a digital circuit:

Task	Microprocessor	Custom Digital Circuit
Read	5	0.1
Compress	8	0.5
Store	1	0.8

Q: How long for each implementation option?



(a) 5+8+1 =14 sec

(b) .1+.5+.8 =1.4 sec

(c) .1+.5+1 =1.6 sec

Good compromise

15

15

Summary

- Digital systems surround us
  - Inside computers
  - Inside many other electronic devices (embedded systems)
- Digital systems use 0s and 1s
  - Encoding analog signals to digital can provide many benefits
    - e.g., audio—higher-quality storage/transmission, compression, etc.
  - Encoding integers as 0s and 1s: Binary numbers
- Microprocessors (themselves digital) can implement many digital systems easily and inexpensively
  - But often not good enough—need custom digital circuits

16

16