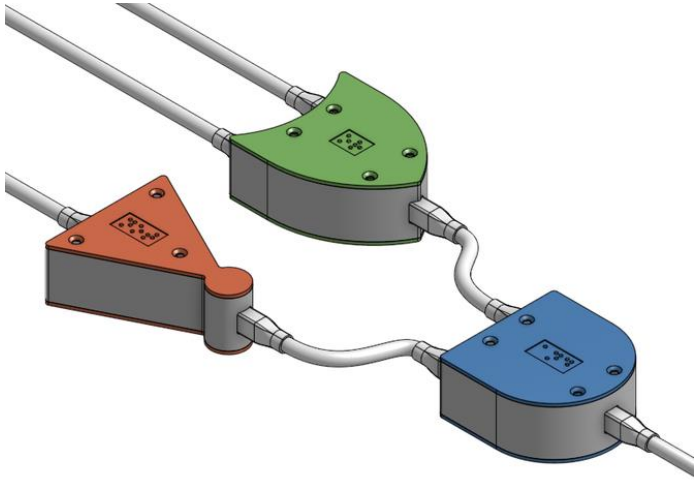
جامعة شقراء
Shaqra University



CS 211 - Digital Logic Design
211 عال - تصميم المنطق الرقمي

First Term - 1439/1440
Lecture #1

Dr. Hazem Ibrahim Shehata

Assistant Professor

College of Computing and Information Technology

Teaching Staff

- Instructor:
  - Hazem Ibrahim Shehata
  - Email: hshehata@su.edu.sa
  - Lectures: Wednesday 8:00am – 9:50am
  - Tutorials: Sunday 1:00pm – 2:50pm
  - Office Hours: TBA

Course Info

- Course website:
  - http://hshehata.github.io/courses/su/cs211/

- Textbook:
  - "Digital Fundamentals", Thomas L. Floyd, 10th Edition, 2009, http://catalogue.pearsoned.co.uk/educator/product/Digital-Fundamentals-Pearson-New-International-Edition-10E/9781292025629.page

Course Info (Cont.)

- Grading

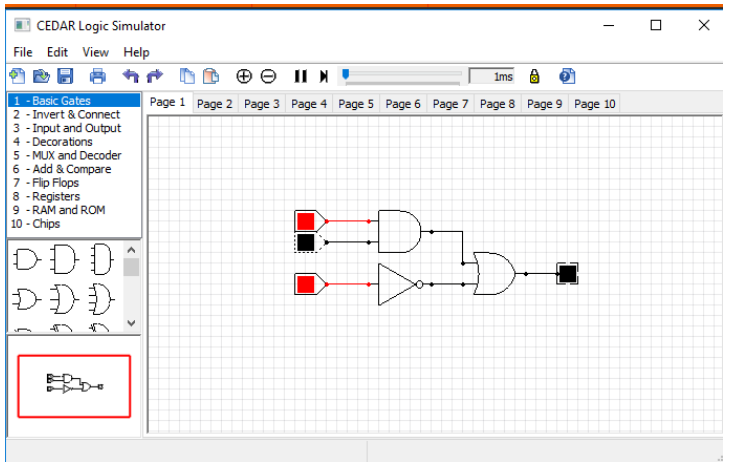
Course Work	Grade Distribution	
Attendance	10pt	60pt
Quizzes	10pt	
Assignments	10pt	
Midterm Exam (1)	15pt	
Midterm Exam (2)	15pt	
Final Exam	40pt	
Total Points	**100pt**	

Course Overview

- Number Systems, Operations, Codes → Ch. 2
- Logic Gates and Boolean Algebra → Ch. 3, 4
- Combinational Logic → Ch. 5, 6
- Sequential Logic → Ch. 7, 8, 9
- Memory → Ch. 10 [Optional!]

Cedar Logic Simulator

- Concepts learned in this course are applied using: Cedar Logic Simulator (or shortly: Cedar LS).
  - Open-source digital logic simulator
  - Developed for academic purposes.
  - Very simple and intuitive user interface.
  - Download link:
    - https://sourceforge.net/projects/cedarlogic/files/latest/download



Ch.2: Number Systems, Operations, and Codes

Decimal Numbers

- Position of each digit in a weighted number system is assigned a weight based on the base or radix of that system.
- Radix/base of decimal system is 10.
  - Ten possible values for each digit: 0, 1, 2, …, 9.
  - Example: The number 409.631_{10}
- Column weights of decimal numbers are powers of 10:

$$\ldots\ 10^3\ 10^2\ 10^1\ 10^0.\ 10^{-1}\ 10^{-2}\ 10^{-3}\ 10^{-4}\ \ldots$$

Radix Point (Decimal Point)

Decimal Numbers (Cont.)

- Decimal numbers can be expressed as sum of products of each digit times column value for that digit.
- Example:
  - Express the number 480.52_{10} as the sum of values of each digit.
- Solution:
  - $480.52_{10} = (4 \times 10^2) + (8 \times 10^1) + (0 \times 10^0) + (5 \times 10^{-1}) + (2 \times 10^{-2})$

Binary Numbers

- Digital systems use binary number system.
- Radix/base of system is 2.
  - Two possible values for each binary digit (know as bit): 0 or 1.
  - Example: The number 10010.001_2
  - Note: Values 0 and 1 can be easily represented in hardware using 2 distinct voltage values: low (e.g., 0 volt) and high (e.g., 5 volt)!
- Column weights of decimal numbers are powers of 2:

$$\ldots\ 2^3\ 2^2\ 2^1\ 2^0.\ 2^{-1}\ 2^{-2}\ 2^{-3}\ 2^{-4}\ \ldots$$

Conversion: Binary ➔ Decimal

- Method: Sum of weights (Add column values of all of bits that are 1 and discard all of bits that are 0).
- Example: Convert 100101.01_2 to decimal.
- Solution: Write column weights ➔ Add weights that correspond to 1's.

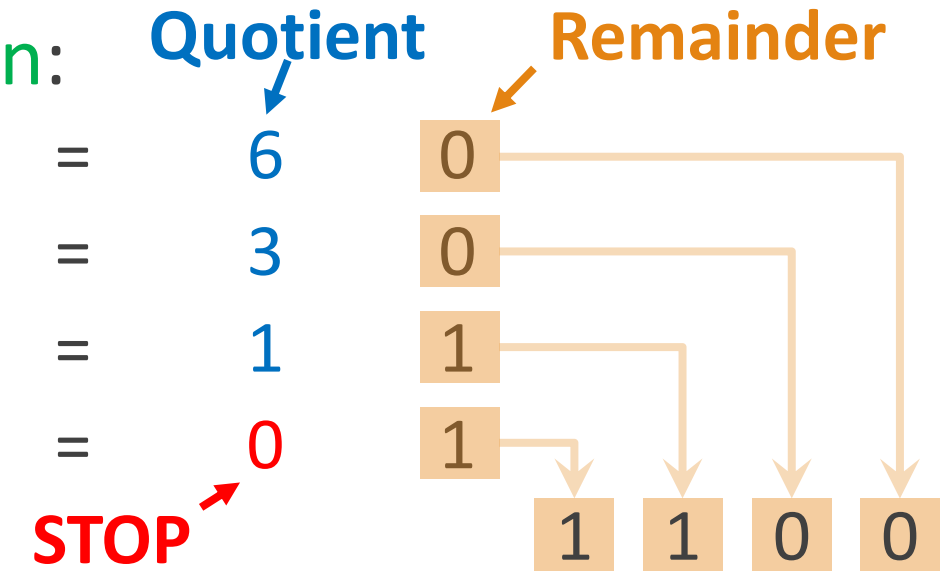
2^5	2^4	2^3	2^2	2^1	2^0	.	2^{-1}	2^{-2}	
32	16	8	4	2	1	.	½	¼	
1	0	0	1	0	1	.	0	1	
32			+4		+1			+¼	= 37.25_{10}

Conversion: Decimal ➔ Binary (Integer)

- Method #1: Reverse sum-of-weights (Write down column weights and place 1's in columns that sum to decimal num.)
- Example: Convert 49_{10} to binary.
- Solution: Write down column weights until the last number is larger than the one you want to convert.
  - 2^6 2^5 2^4 2^3 2^2 2^1 2^0
  - 64 32 16 8 4 2 1
  - 0 1 1 0 0 0 1 = 110001_2

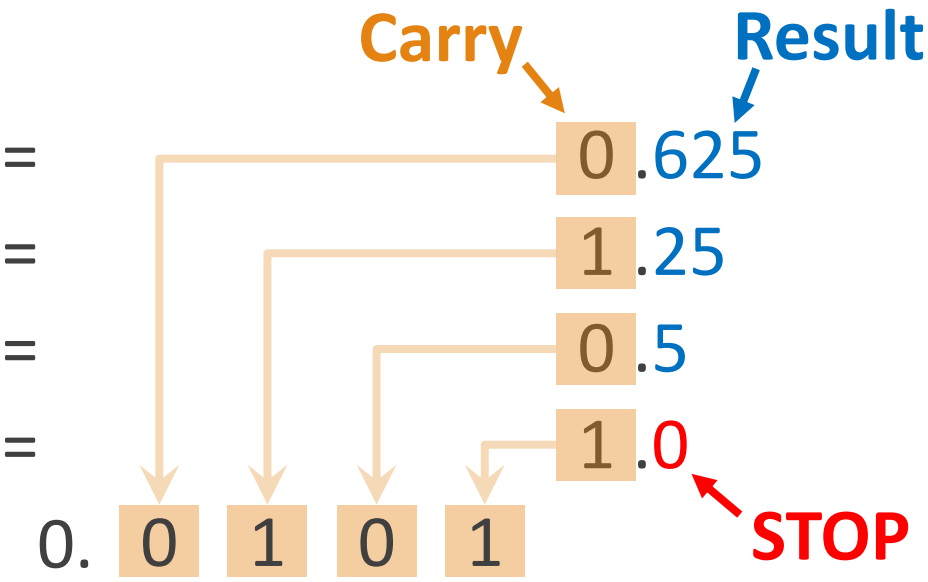
Conversion: Decimal ➔ Binary (Integer)

- Method #2: Repeated division-by-2 (divide decimal num. by 2 until quotient is 0. Remainders form binary num.)
- Example: Convert 12_{10} to binary.
- Solution:
  - 12 ÷ 2 = 6
  - 6 ÷ 2 = 3
  - 3 ÷ 2 = 1
  - 1 ÷ 2 = 0



Conversion: Decimal ➔ Binary (Fraction)

- Method: Repeated multiplication-by-2 (multiply fractional results repeatedly by 2. Carries form binary fraction)
- Example: Convert 0.3125_{10} to binary.
- Solution:
  - 0.3125 * 2 =
  - 0.625 * 2 =
  - 0.25 * 2 =
  - 0.5 * 2 =



Counting in Binary

- Using 4 bits, we can represent numbers: 0 : 2^4-1
- Using n bits, we can represent numbers: 0 : 2^n-1

Decimal Number	Binary Number
00	0 0 0 0
01	0 0 0 1
02	0 0 1 0
03	0 0 1 1
04	0 1 0 0
05	0 1 0 1
06	0 1 1 0
07	0 1 1 1
08	1 0 0 0
09	1 0 0 1
10	1 0 1 0
11	1 0 1 1
12	1 1 0 0
13	1 1 0 1
14	1 1 1 0
15	1 1 1 1

Reading Material

- Floyd, Chapter 2:
  - Pages 45 – 54