Digital Design & Computer Organization

CS 1050 - Week 01

About the lecturer

Dr. Thanuja D. Ambegoda

- BSc Hons Electronics & Telecommunications Engineering, University of Moratuwa
- MSc Computer Science, Swiss Federal Institute of Technology (ETH) Zurich
- PhD in Computer Vision & Machine Learning, ETH Zurich.
- Former co-founder of Enway GmbH, Germany.

Lesson Outline

1. Introduction to Digital Design
2. Number systems
3. Binary codes and arithmetic operations
4. Logic signals
5. Logic gates & digital logic families - pushed to week 02

1. Introduction to digital design

Analog vs Digital

Analog components have time varying signals that can take continuous ranges of values

Digital components are similar! But we model their inputs to be discrete values (e.g. either 0 or 1)

Why digital over analog?

1. Reproducibility of results
2. Ease of design
3. Flexibility and functionality
4. Programmability
5. Speed

Recommended reading

Digital Design by John Wakerly (3rd edition)

Chapter 1

Some useful tips

IMPORTANT THEMES IN DIGITAL DESIGN

- Good tools do not guarantee good design, but they help a lot by taking the pain out of doing things right.
- Digital circuits have analog characteristics.
- Know when to worry and when not to worry about the analog aspects of digital design.
- Always document your designs to make them understandable by yourself and others.
- Associate active levels with signal names and practice bubble-to-bubble logic design.
- Understand and use standard functional building blocks.
- Design for minimum cost at the system level, including your own engineering effort as part of the cost.
- State-machine design is like programming; approach it that way.
- Use programmable logic to simplify designs, reduce cost, and accommodate last-minute modifications.
- Avoid asynchronous design. Practice synchronous design until a better methodology comes along.
- Pinpoint the unavoidable asynchronous interfaces between different subsystems and the outside world, and provide reliable synchronizers.
- Catching a glitch in time saves nine.

From the textbook

2. Number systems

Basic definitions

- Binary numbers
    - (0, 1)
- Base conversions
    - decimal to binary, binary to decimal, hex to decimal etc
- Number systems
    - there are more than one way to express a number in binary
    - E.g. 1010 could be 10, -2, -5 or -6
- A/D and D/A conversion
    - Real world signals come in continuous/analog form. These signals and modeled as 0s and 1s in digital systems

The basics: binary numbers

Bases we use:

- Binary (2)
- Octal (8)
- Decimal (10)
- Hexadecimal (16)

Positional number system:

- $101_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$
- $63_8 = 6 \times 8^1 + 3 \times 8^0$

Base conversions

Conversion from binary to octal/hex

Binary: 10011110001

Octal: 10 | 011 | 110 | 001=2361_8

Hex: 100 | 1111 | 0001=$4F1_{16}$

Conversion from binary to decimal

$101_2 = 1×2^2 + 0×2^1 + 1×2^0 = 5_{10}$

$63.4_8 = 6×8^1 + 3×8^0 + 4×8^{–1} = 51.5_{10}$

$A1_{16} = 10×16^1 + 1×16^0 = 161_{10}$

Negative binary numbers

3 approaches:

- Sign-and-magnitude
- Ones’-complement
- Two’s-complement

For all 3, the MSB is the sign digit

- 0 positive
- 1 negative

Two’s-complement

- Widely used
- Simplifies arithmetic

Sign-and-magnitude

- The most-significant bit (MSB) is the sign digit
  - 0 ≡ positive
  - 1 ≡ negative
- The remaining bits are the number's magnitude
- Problem 1: Two representations for zero
  - 0 = 0000 and also –0 = 1000
- Problem 2: Arithmetic is cumbersome

Add		Subtract			Compare and subtract		
4	0100	4	0100	0100	– 4	1100	1100
+ 3	+ 0011	– 3	+ 1011	– 0011	+ 3	+ 0011	– 0011
= 7	= 0111	= 1	≠ 1111	= 0001	– 1	≠ 1111	= 1001

Read more about the signed zero

Ones'-complement

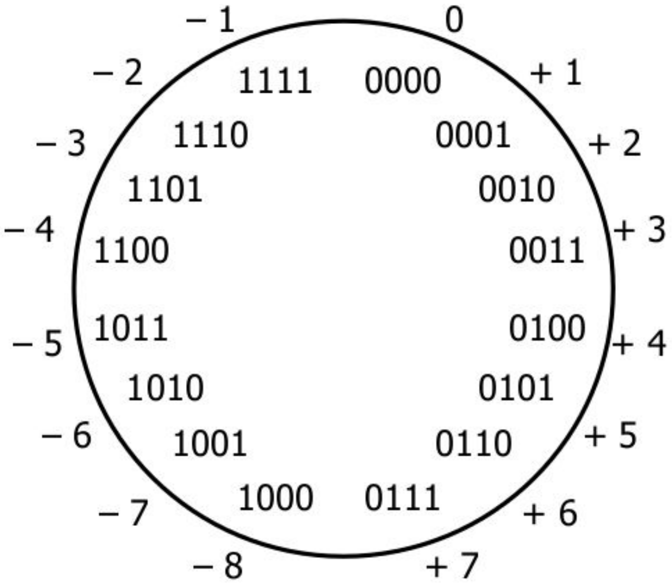
- Negative number: Bitwise complement positive number
  - $0111 \equiv 7_{10}$
  - $1000 \equiv -7_{10}$
- Solves the arithmetic problem

Add		Invert, add, add carry		Invert and add	
4	0100	4	0100	– 4	1011
+ 3	+ 0011	– 3	+ 1100	+ 3	+ 0011
= 7	= 0111	= 1	1 0000	– 1	1110
		add carry:	+1		
			= 0001		

- Remaining problem: Two representations for zero
  - 0 = 0000 and also –0 = 1111

Two's-complement

- Negative number: Bitwise complement plus one
  - $0111 \equiv 7_{10}$
  - $1001 \equiv -7_{10}$
- Number wheel



- Only one zero!
- MSB is the sign digit
  - 0 ≡ positive
  - 1 ≡ negative

Two's-complement (2)

- Complementing a complement ➜ the original number
- Arithmetic is easy
  - Subtraction = negation and addition
    - Easy to implement in hardware

Add		Invert and add		Invert and add	
4	0100	4	0100	– 4	1100
+ 3	+ 0011	– 3	+ 1101	+ 3	+ 0011
= 7	= 0111	= 1	1 0001	– 1	1111
		drop carry	= 0001		

3. Binary codes and binary arithmetic operations

Binary Coded Decimal (BCD) and Gray Codes

Decimal Symbols	BCD Code
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Decimal Symbols	Gray Code
0	0000
1	0001
2	0011
3	0010
4	0110
5	0111
6	0101
7	0100
8	1100
9	1101

Only one bit changes per step

Gray codes are widely used to prevent spurious output from electromechanical switches and to facilitate error correction in digital communications such as digital terrestrial television and some cable TV systems.

Binary Operations

1. Addition
2. Subtraction
3. Multiplication
4. Division

Supplementary reading if you need a quick review

Also Chapter 2 of the recommended text.

4. Logic signals

Logic Levels

The voltages used to represent a 1 and 0 are called logic levels.

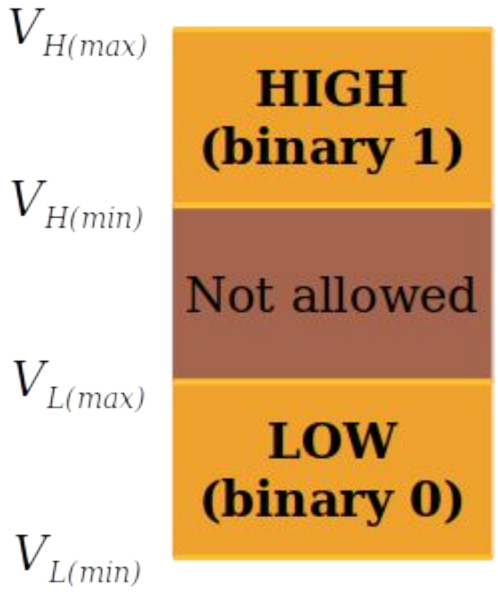
Ideally, there is only HIGH (1) and LOW (0).

Practically, there must be thresholds to determine which one is HIGH or LOW or neither of them.

E.g. CMOS

(2V to 3.3V HIGH)

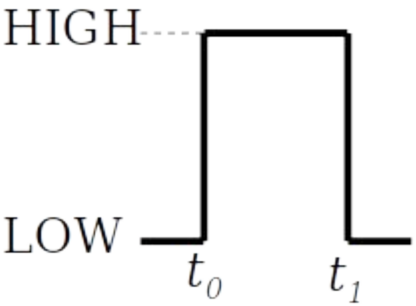
(0V. To 0.8V LOW)



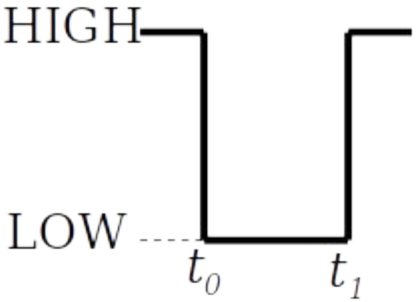
Digital Waveforms

Voltage levels that are changing back and forth between HIGH and LOW

(Ideal) pulse



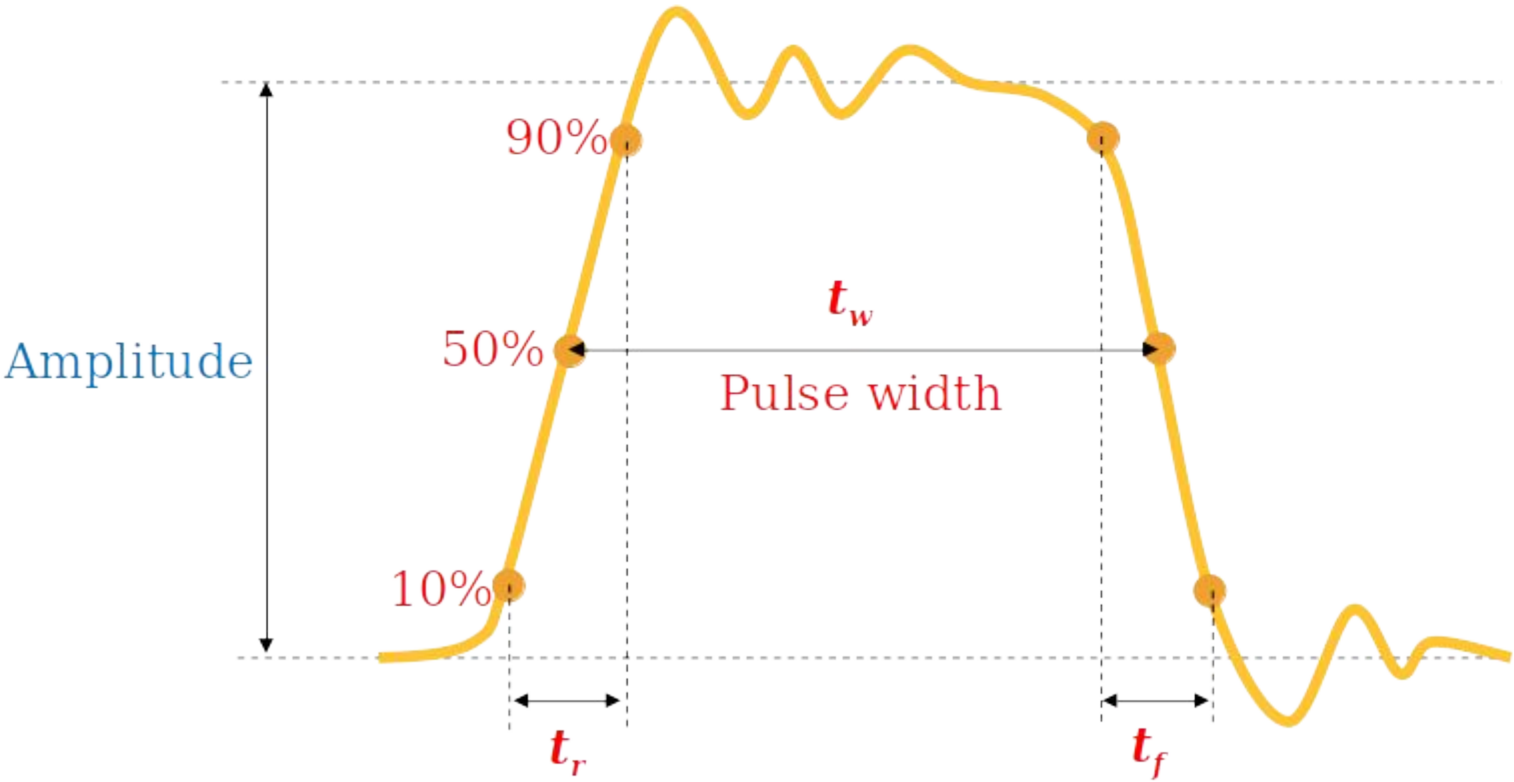
Positive-going pulse



Negative-going pulse

At t0 leading edge, at t1 trailing edge

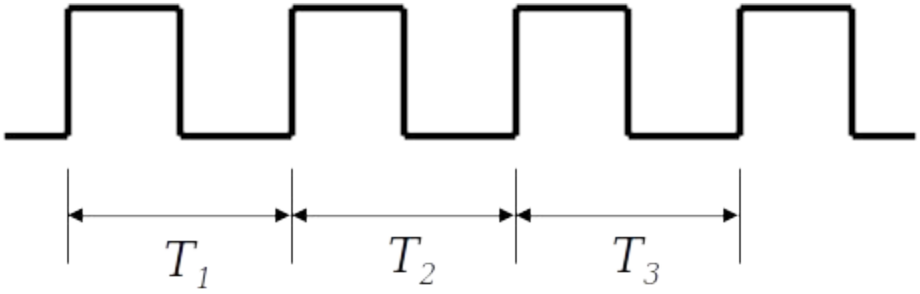
Non-ideal pulse



Waveform Characteristics

Waveforms = series of pulses (called pulse train)

Periodic



Period (T) = T1 = T2 = T3 = … = Tn

Frequency (f) = 1/T

Nonperiodic

Duty cycle

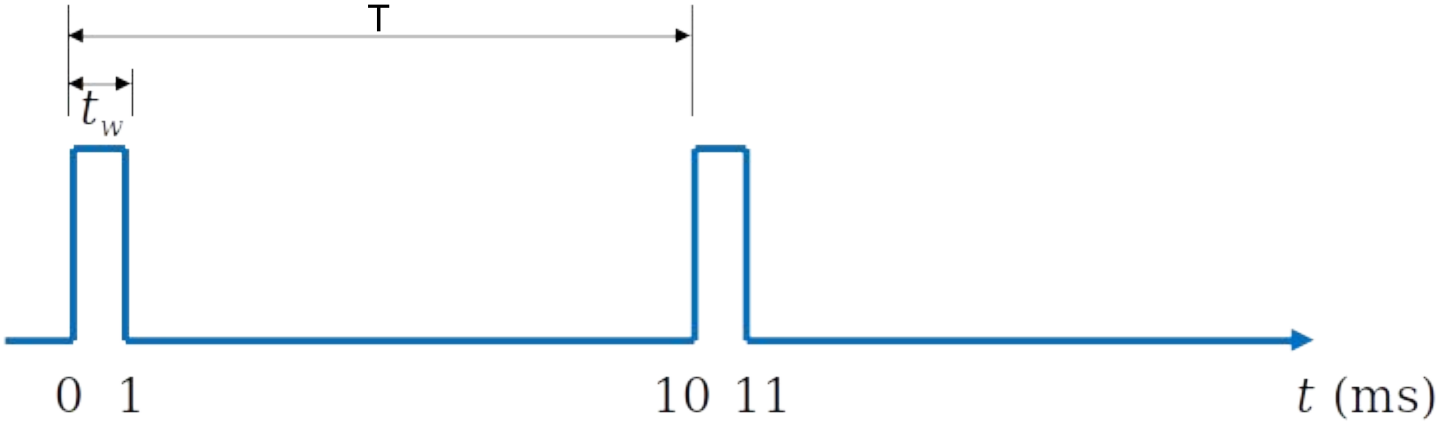
Ratio of the pulse width (t_w) to the period (T)

$$\text{Duty cycle} = (t_w / T) \times 100\%$$

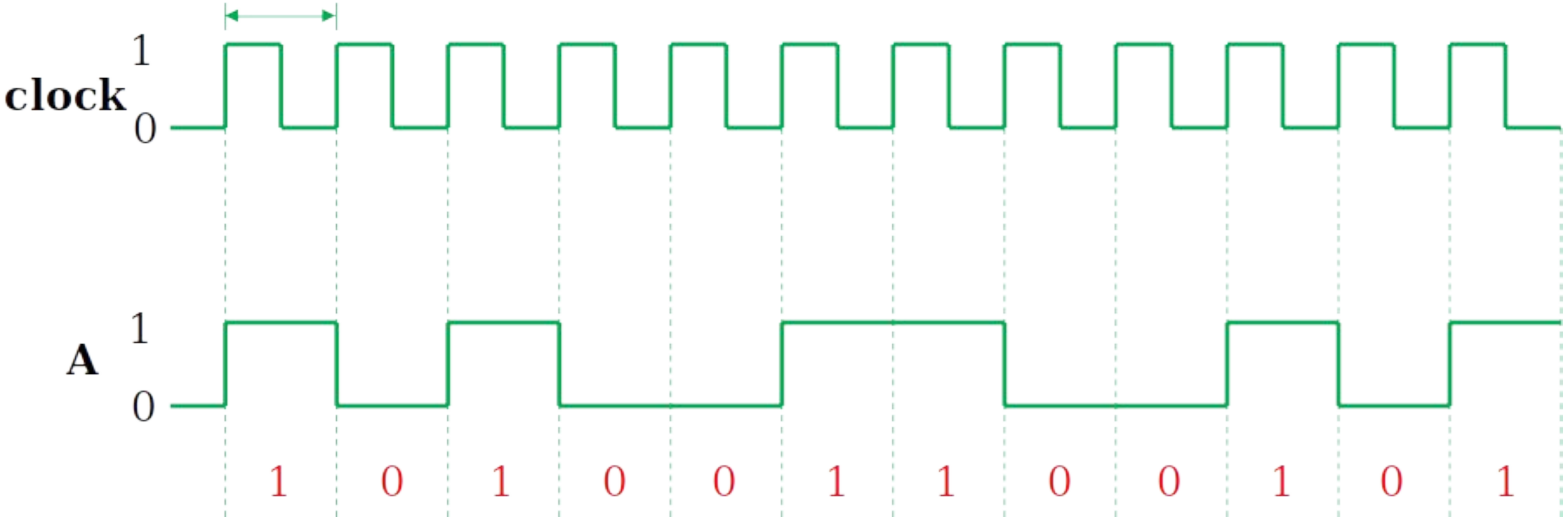
Example

From a portion of a periodic waveform (as shown) determine:

- Period
- Frequency
- Duty cycle



Waveform & binary information



Data Transfer

Binary data are transferred in two ways:

- Serial – bits are sent one bit at a time
- Parallel – all the bits in a group are sent out on separate lines at the same time (one line for each bit)

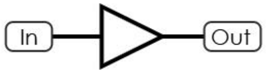
Serial over Parallel

- Advantage: less transmission lines
- Disadvantage: takes more time

5. Logic gates and digital logic families

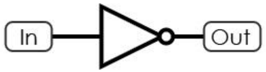
Logic Gates - Symbols and Truth Tables

BUF
(Buffer)



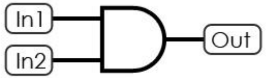
In	Out
0	0
1	1

NOT
(Inverter)



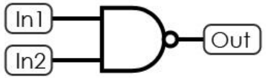
In	Out
0	1
1	0

AND



In1	In2	Out
0	0	0
0	1	0
1	0	0
1	1	1

NAND
(NOT AND)



In1	In2	Out
0	0	1
0	1	1
1	0	1
1	1	0

OR

In1	In2	Out
0	0	0
0	1	1
1	0	1
1	1	1

NOR
(NOT OR)

In1	In2	Out
0	0	1
0	1	0
1	0	0
1	1	0

XOR
(Exclusive Or)

In1	In2	Out
0	0	0
0	1	1
1	0	1
1	1	0

XNOR
(NOT XOR)

In1	In2	Out
0	0	1
0	1	0
1	0	0
1	1	1

A circle behind a symbol indicates that the output signal is inverted.

What is a digital logic family?

A collection of different integrated circuit (IC) chips that have:

- Similar input, output and internal **circuit characteristics**
- But, perform different logic functions

Chips from the same family can be interconnected to perform any desired logic function

Chips from different logic families may not be compatible, therefore can’t be interconnected successfully

To be continued next week