☆ The digital abstraction

So far, we've been studying analog signals like voltage and current, and we focus on how these signals, being transformed by a circuit, will impact the behavior of a certain element in a circuit. The impact manifests itself in terms of analog signals on the element, and we also called it the "response" of the element with respect to the analog signals from a voltage source  or a current source  .

Now, how does an analog signal relates to the "digital world", the world built using some meaningful combinations of 0s and 1s? The digital abstraction serves this purpose. It specifies a transformation that interprets the analog signals into a series of binary digits, the so-called "digital signals."

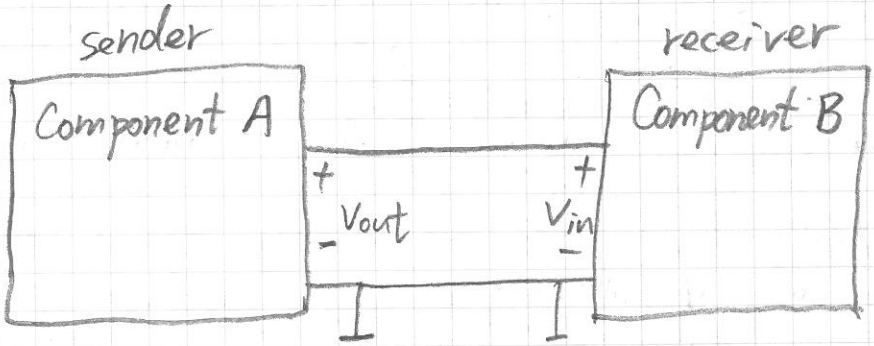
P50

Review Section 1.8 in the textbook.

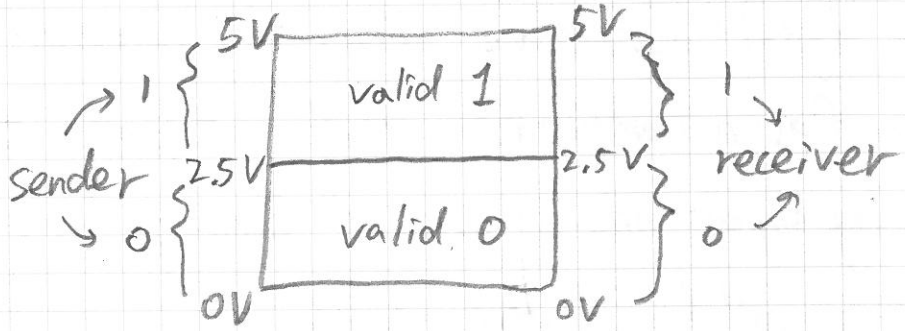
- Digital signals are "interpretations" of analog signals, so that they can be understood and be used by a digital system (such as a computer).

- Between physical components in a system, it is still analog signals that are transferred.

- How to transform digital signals into analog ones?

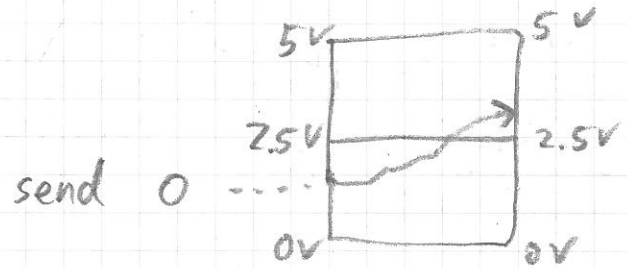


Our first attempt:



A serious problem:

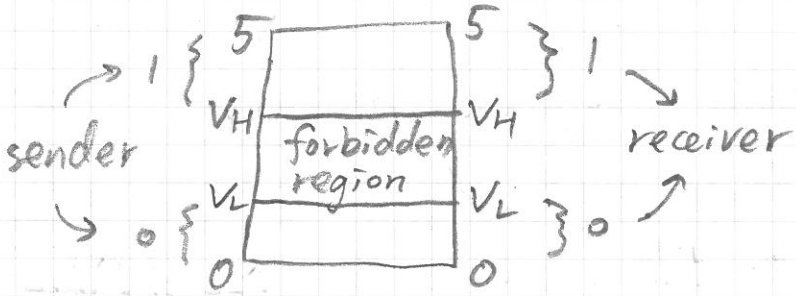
there could be noise/interference during signal transmissions



receiver interprets as 1, not 0!

AND what if received 2.5 v ?

An improved design:



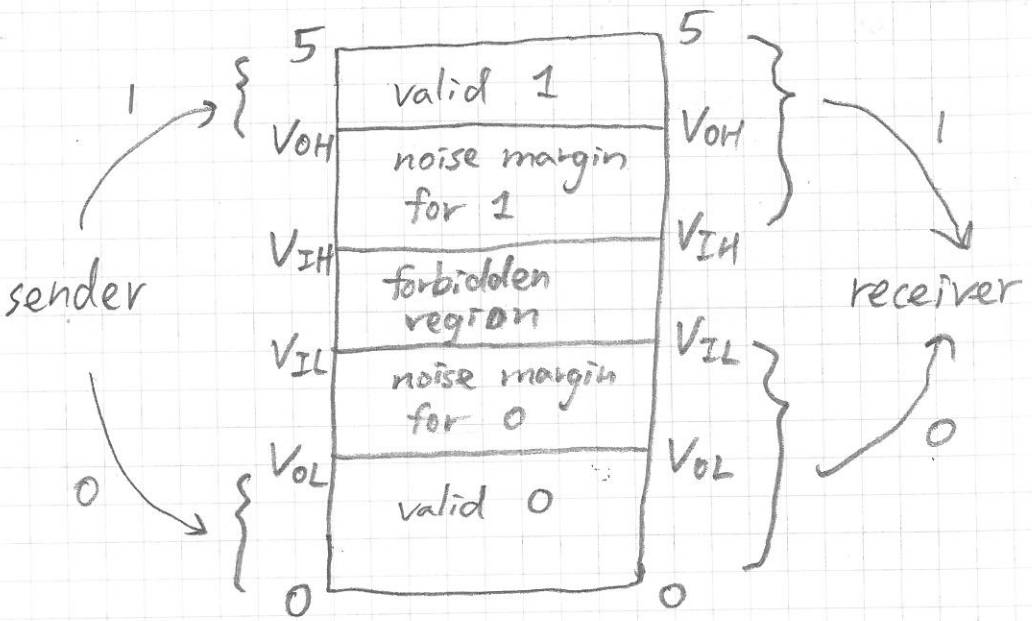
V_H and V_L are high/low voltage thresholds.

A further question:

how to quantify the immunity to noise?

and if we can do that, this can serve as a contract, and accordingly it would ensure device manufacturers to meet consumers' need, and components can be connected to form a system.

P52 A better design :



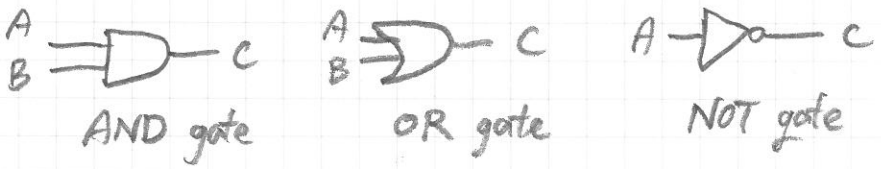
The static discipline (principle)

A device must interpret correctly voltage inputs falling within the V_{IL} or V_{IH} threshold; with a valid input, the device must produce a valid voltage output that falls within the V_{OL} or V_{OH} threshold.

↑ A specification of digital devices.

Sections 5.2, 5.3, 5.6 in the textbook are very good learning materials.
Be sure to study them yourselves.
We will briefly cover some of them at some appropriate opportunity in this course.

Logic gates (i.e., combinational gates):



A	C
1	0
0	1

NAND gate ≡ AND gate plus NOT gate

NOR gate ≡ OR gate plus NOT gate

Truth table

inputs		output (C)			
A	B	AND	OR	NAND	NOR
0	0	0	0	1	1
0	1	0	1	1	0
1	0	0	1	1	0
1	1	1	1	0	0