Digital Logic Design

University Electrical Engineering · Vocabulary in context

Word	Example
digital logic design /ˈdɪdʒɪtl ˈlɒdʒɪk dɪˈzaɪn/	Use **digital logic design** to build a binary counter.
binary /ˈbaɪnəri/	In **binary**, the number 5 is written 101. 5 101
hexadecimal /ˌheksəˈdesɪml/	In **hexadecimal**, the number 255 is written FF. 255 FF
two's complement /tuːz ˈkɒmplɪmənt/	In eight-bit **two's complement**, 11111111 represents -1. 11111111 -1
range /reɪndʒ/	More exponent bits give a bigger **range**.
overflow /ˌəʊvəˈfləʊ/	Adding 200 and 100 in one byte causes **overflow**. 200 100
boolean algebra /ˈbuːlɪən ˈældʒɪbrə/	**Boolean algebra** manipulates AND, OR and NOT expressions. AND、OR NOT
absorption /əbˈsɔːpʃn/	By **absorption**, A OR (A AND B) is just A. A OR (A AND B) A
de morgan's theorem /də ˈmɔːgənz ˈθɪərəm/	**De Morgan's theorem** says NOT (A AND B) equals (NOT A) OR (NOT B). NOT (A AND B) (NOT A) OR (NOT B)。
karnaugh map /ˈkɑːnɔː mæp/	The **Karnaugh map** groups minterms in Gray-code order.
don't-cares	Treat **don't-cares** as 0 or 1, whichever makes the logic simpler. 0 1
combinational logic /ˌkɒmbɪˈneɪʃənl ˈlɒdʒɪk/	**Combinational logic** is a pure function of present inputs.

exercises.lol exercises.com.cn

Word	Example
multiplexer /ˈmʌltɪpleksə/	A **multiplexer** uses a select line to choose one of its inputs.
decoder /dɪˈkəʊdə/	An enabled active-high **decoder** selects one output for each input code.
adder /ˈædə/	An **adder** sums bits and produces a carry.
half adder /hɑːf ˈædə/	A **half adder** adds two bits.
full adder /fʊl ˈædə/	A **full adder** adds two bits and a carry.
hazards	**Hazards** are short glitches caused by unequal delays.
sequential logic /siːˈkwenʃl ˈlɒdʒɪk/	**Sequential logic** has memory: its output depends on past inputs.
latch /lætʃ/	A **latch** is level-sensitive memory.
flip-flops	Clocked **flip-flops** store bits at their specified active edges.
at the clock edge	The flip-flop stores its input **at the clock edge**.
toggles	A T flip-flop **toggles**: it flips from 0 to 1 or from 1 to 0. T 0 1 1 0
excitation table /eksɪˈteɪʃn ˈteɪbl/	The **excitation table** maps desired transitions to required inputs.
finite state machine /ˈfaɪnaɪt steɪt məˈʃiːn/	A **finite state machine** moves between a fixed set of states.

exercises.lol exercises.com.cn

Word	Example
state diagram /steɪt ˈdaɪəgræm/	The **state diagram** draws bubbles and labelled arrows.
state minimisation	**State minimisation** merges equivalent states, so the circuit needs fewer flip-flops.
register /ˈredʒɪstə/	The processor keeps the number it is using in a **register**.
shift register /ʃɪft ˈredʒɪstə/	A **shift register** passes bits along stage by stage.
ripple counter /ˈrɪpl ˈkaʊntə/	A **ripple counter** clocks each stage from the previous output.
synchronous counter /ˈsɪŋkrənəs ˈkaʊntə/	All stages of a **synchronous counter** share the same clock.
memory /ˈmeməri/	With ten address bits, this word-addressed **memory** selects 1024 words. 1024
hardware description language /ˈhɑːdweə dɪˈskrɪpʃn ˈlæŋgwɪdʒ/	Verilog is a **hardware description language**: it describes circuits as code. Verilog
synthesizable /ˈsɪnθəsaɪzəbl/	**Synthesizable** code can be turned into real gates.
testbench /ˈtestbentʃ/	A **testbench** drives simulated inputs and checks the resulting outputs.
moore machines	**Moore machines** produce outputs from the state only.
mealy machines	**Mealy machines** produce outputs from the state and the inputs.

exercises.lol exercises.com.cn