**Harold’s Boolean Algebra**

**Cheat Sheet**

8 September 2025

**Boolean Algebra**

|  |  |  |  |
| --- | --- | --- | --- |
| **Boolean Law** | **Boolean Expression** | **Equivalent Circuit** | **Description** |
| **Idempotent** |  | idempotent parallel circuit | A in parallel with A = “A” |
|  | idempotent series circuit | A in series with A = “A” |
| **Associative** |  |  | Allows the removal of brackets from an expression and the regrouping of the variables |
|  |  |
| **Commutative** |  | absorption parallel circuit | A in parallel with B = B in parallel with A |
|  | absorption series circuit | A in series with B = B in series with A |
| **Distributive** |  |  | Permits the multiplying or factoring out of an expression |
|  |  |
| **Identity** |  | universal parallel | A in parallel with open = “A” |
|  | universal series circuit | A in series with closed = “A” |
| **Domination**  (Annulment) |  | universal parallel circuit | A in parallel with closed = “CLOSED” |
|  | universal series | A in series with open = “OPEN” |
| **Double Compliment**  (Double Negation) |  |  | NOT NOT A (double negative) = “A” |
| **Complement** |  | complement parallel circuit | A in parallel with NOT A = “CLOSED” |
|  | complement series circuit | A in series with NOT A = “OPEN” |
| **De Morgan’s** |  |  | Invert and replace OR with AND |
|  |  | Invert and replace AND with OR |
| **Absorption**  (Absorptive) |  |  | Enables a reduction in a complicated expression to a simpler one by absorbing like terms |
|  |  |
|  |  | Reduces a complicated expression to a simpler one by absorbing complement term |

**Precedence Rules**

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Operator** | **Symbol** | **Precedence** |
| **1** | **Parenthesis** | ( ) | Highest precedence |
| **2** | **NOT** | ~ |  |
| **3** | **Quantifiers** | ∀*,* ∃ |  |
| **4** | **AND** | • | Applied Left to Right |
| **5** | **OR** | ∨ |  |
| **6** | **Conditional** | ⊃ |  |
| **7** | **Biconditional** | **↔** | Lowest precedence |

**Boolean Logic Gates**

|  |  |  |  |
| --- | --- | --- | --- |
| **Boolean Logic** | **Notation** | **Gate** | **Description** |
| **IDENTITY** | 1  T  True  S |  | On, Tautology, High voltage (typically +5V) |
| **NULL** | 0  F  False  ⊥  ∅ | **GND**  Learn more | Circuit Playground: G is for Ground | Adafruit ... | Off, Contradiction, Low voltage (typically 0V) |
| **Input** | A, B, C, D |  | Line, Wire, Connects to |
| **Output** | W, X, Y, Z |  | Line, Wire, Connects from |
| **AND** | A ∧ B  A ∩ B | AND symbol | AND, BUT, Multiply, Conjunction, Intersection |
| **OR** | A ∨ B  A ∪ B  A | B | OR symbol | Inclusive-OR, Add, Disjunction, Union |
| **NOT** |  | NOT symbol | NOT, Invert, Negation, Change, Difference |
| **NAND** | A ⊼ B  A | B\* | NAND symbol | Not AND |
| **NOR** | A ⊽ B  A ↓ B | NOR symbol | Not OR |
| **XOR** | A ⊕ B  A ⊻ B | XOR symbol | Exclusive-OR, Both A and B are different |
| **XNOR** | A ⊙ B | XNOR symbol | Exclusive-NOR, Both A and B are the same |

**NOTE:** The laws of Boolean algebra are the same in propositional and set logic.

**Boolean Logic Truth Tables**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Inputs** | | **Outputs** | | | | | | | | |
| **A** | **B** | **AND** | **NAND**  **⊼** | **OR**  **+** | **NOR**  **⊽** | **XOR**  ⊕ | **XNOR**  ⊙ | **NOT** | **VCC**  **1** | **GND**  **0** |
| **0** | **0** | 0 | 1 | 0 | 1 | 0 | 1 | A=1 | 1 | 0 |
| **0** | **1** | 0 | 1 | 1 | 0 | 1 | 0 | A=1 | 1 | 0 |
| **1** | **0** | 0 | 1 | 1 | 0 | 1 | 0 | A=0 | 1 | 0 |
| **1** | **1** | 1 | 0 | 1 | 0 | 0 | 1 | A=0 | 1 | 0 |

**Blank Truth Tables**

|  |  |  |
| --- | --- | --- |
| **Inputs** | | **Output** |
| **A** | **B** | **X** |
| **0** | **0** |  |
| **0** | **1** |  |
| **1** | **0** |  |
| **1** | **1** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Inputs** | | | **Output** | |
| **A** | **B** | **C** | **X** | **Y** |
| **0** | **0** | **0** |  |  |
| **0** | **0** | **1** |  |  |
| **0** | **1** | **0** |  |  |
| **0** | **1** | **1** |  |  |
| **1** | **0** | **0** |  |  |
| **1** | **0** | **1** |  |  |
| **1** | **1** | **0** |  |  |
| **1** | **1** | **1** |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Inputs** | | | | **Output** | | |
| **A** | **B** | **C** | **D** | **X** | **Y** | **Z** |
| **0** | **0** | **0** | **0** |  |  |  |
| **0** | **0** | **0** | **1** |  |  |  |
| **0** | **0** | **1** | **0** |  |  |  |
| **0** | **0** | **1** | **1** |  |  |  |
| **0** | **1** | **0** | **0** |  |  |  |
| **0** | **1** | **0** | **1** |  |  |  |
| **0** | **1** | **1** | **0** |  |  |  |
| **0** | **1** | **1** | **1** |  |  |  |
| **1** | **0** | **0** | **0** |  |  |  |
| **1** | **0** | **0** | **1** |  |  |  |
| **1** | **0** | **1** | **0** |  |  |  |
| **1** | **0** | **1** | **1** |  |  |  |
| **1** | **1** | **0** | **0** |  |  |  |
| **1** | **1** | **0** | **1** |  |  |  |
| **1** | **1** | **1** | **0** |  |  |  |
| **1** | **1** | **1** | **1** |  |  |  |

**Karnaugh Mapping (K-Map)**

|  |  |  |  |
| --- | --- | --- | --- |
| 2-Bit  K-Map | | **A** | |
| 0 | 1 |
| **B** | 0 |  |  |
| 1 |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 3-Bit  K-Map | | **AB** | | | |
| 00 | 01 | 11 | 10 |
| **C** | 0 |  |  |  |  |
| 1 |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 4-Bit  K-Map | | **AB** | | | |
| 00 | 01 | 11 | 10 |
| **CD** | 00 |  |  |  |  |
| 01 |  |  |  |  |
| 11 |  |  |  |  |
| 10 |  |  |  |  |

2x2 Group

1x4 Group

**K-Map Rules**

1) Circle only 1s (ones) and don’t cares for Sum of Products (SOP), .

a. Circle only 0s (zeros) and don’t cares for Product of Sums (POS), .

b. Don’t cares may be used or ignored.

2) No diagonals, only horizontal or vertical connections.

3) Group only adjacent cells in groups with powers of 2 (1x1, 1x2, 2x1, 2x2, 2x4, 4x2, 1x4, 4x1).

4) Make groups as large as possible.

5) Must group all 1s (ones) for SOP or all 0s (zeros) for POS.

6) Overlapping is allowed.

7) Wrapping around all edges allowed, both top-bottom edges and left-right edges.

8) Fewest groups possible (OPTIMAL).

9) For each circle, determine which inputs do not contribute to the logic (is both 0 and 1).

10) Write down the equation as a SOP,

**Sources**

* [SNHU MAT 230](https://www.snhu.edu/admission/academic-catalogs/coce-catalog#/courses/4kVhSZLtg) - Discrete Mathematics, zyBooks.
* <https://www.electronics-tutorials.ws/boolean/bool_6.html>
* Toomey, H. A. (2024). Logical Connective Laws, Harold’s Logic Cheat Sheet, <https://www.toomey.org/tutor/discrete_math.html>

**See Also**

* [Harold’s Logic Cheat Sheet](https://www.toomey.org/tutor/harolds_cheat_sheets/Harolds_Logic_Cheat_Sheet.pdf)
* [Harold’s Logic (Philosophy) Cheat Sheet](https://www.toomey.org/tutor/harolds_cheat_sheets/Harolds_Logic_(Philosophy)_Cheat_Sheet.pdf)
* [Harold’s Sets Cheat Sheet](https://www.toomey.org/tutor/advanced_math.html)
* [Harold’s Boolean Algebra Cheat Sheet](https://www.toomey.org/tutor/discrete_math.html)
* [Harold’s Proofs Cheat Sheet](https://www.toomey.org/tutor/advanced_math.html)