# Chip8 Opcode Table

|  |  |
| --- | --- |
| Opcode | Explanation |
| 0NNN | Calls [RCA 1802](https://en.wikipedia.org/wiki/RCA_1802) program at address NNN. Not necessary for most ROMs. |
| 00E0 | Clears the screen. |
| 00EE | Returns from a subroutine. |
| 1NNN | Jumps to address NNN. |
| 2NNN | Calls subroutine at NNN. |
| 3XNN | Skips the next instruction if VX equals NN. |
| 4XNN | Skips the next instruction if VX doesn't equal NN. |
| 5XY0 | Skips the next instruction if VX equals VY. |
| 6XNN | Sets VX to NN. |
| 7XNN | Adds NN to VX. |
| 8XY0 | Sets VX to the value of VY. |
| 8XY1 | Sets VX to VX [or](https://en.wikipedia.org/wiki/Logical_disjunction) VY. |
| 8XY2 | Sets VX to VX [and](https://en.wikipedia.org/wiki/Logical_conjunction) VY. |
| 8XY3 | Sets VX to VX [xor](https://en.wikipedia.org/wiki/Xor" \o "Xor) VY. |
| 8XY4 | Adds VY to VX. VF is set to 1 when there's a carry, and to 0 when there isn't. |
| 8XY5 | VY is subtracted from VX. VF is set to 0 when there's a borrow, and 1 when there isn't. |
| 8XY6 | Shifts VX right by one. VF is set to the value of the [least significant bit](https://en.wikipedia.org/wiki/Least_significant_bit) of VX before the shift.[[2]](https://en.wikipedia.org/wiki/CHIP-8#cite_note-shift-2) |
| 8XY7 | Sets VX to VY minus VX. VF is set to 0 when there's a borrow, and 1 when there isn't. |
| 8XYE | Shifts VX left by one. VF is set to the value of the [most significant bit](https://en.wikipedia.org/wiki/Most_significant_bit) of VX before the shift.[[2]](https://en.wikipedia.org/wiki/CHIP-8#cite_note-shift-2) |
| 9XY0 | Skips the next instruction if VX doesn't equal VY. |
| ANNN | Sets I to the address NNN. |
| BNNN | Jumps to the address NNN plus V0. |
| CXNN | Sets VX to the result of a bitwise and operation on a random number and NN. |
| DXYN | Sprites stored in memory at location in index register (I), 8bits wide. Wraps around the screen. If when drawn, clears a pixel, register VF is set to 1 otherwise it is zero. All drawing is XOR drawing (i.e. it toggles the screen pixels). Sprites are drawn starting at position VX, VY. N is the number of 8bit rows that need to be drawn. If N is greater than 1, second line continues at position VX, VY+1, and so on. |
| EX9E | Skips the next instruction if the key stored in VX is pressed. |
| EXA1 | Skips the next instruction if the key stored in VX isn't pressed. |
| FX07 | Sets VX to the value of the delay timer. |
| FX0A | A key press is awaited, and then stored in VX. |
| FX15 | Sets the delay timer to VX. |
| FX18 | Sets the sound timer to VX. |
| FX1E | Adds VX to I.[[3]](https://en.wikipedia.org/wiki/CHIP-8#cite_note-onlgame-3) |
| FX29 | Sets I to the location of the sprite for the character in VX. Characters 0-F (in hexadecimal) are represented by a 4x5 font. |
| FX33 | Stores the [Binary-coded decimal](https://en.wikipedia.org/wiki/Binary-coded_decimal) representation of VX, with the most significant of three digits at the address in I, the middle digit at I plus 1, and the least significant digit at I plus 2. (In other words, take the decimal representation of VX, place the hundreds digit in memory at location in I, the tens digit at location I+1, and the ones digit at location I+2.) |
| FX55 | Stores V0 to VX in memory starting at address I.[[4]](https://en.wikipedia.org/wiki/CHIP-8#cite_note-memi-4) |
| FX65 | Fills V0 to VX with values from memory starting at address I.[[4]](https://en.wikipedia.org/wiki/CHIP-8#cite_note-memi-4) |