|  |  |  |
| --- | --- | --- |
| **Opcode** | **Mnemonic** | **Description** |
| 000000 | ADD | r1 = r2 + r3 |
| 000001 | SUB | r1 = r2 - r3 |
| 000010 | MUL | r1 = r2 \* r3 |
| 000011 | DIV | r1 = r2 / r3 |
| 000100 | INC | r1 = r1 + 1 |
| 000101 | DEC | r1 = r1 - 1 |
| 000110 | AND | r1 = r2 & r3 |
| 000111 | OR | r1 = r2 |
| 001000 | XOR | r1 = r2 ^ r3 |
| 001001 | NOT | r1 = ~r2 |
| 001010 | JMP | PC = addr |
| 001011 | BEQ | if (r1 == r2) PC = addr |
| 001100 | BNE | if (r1 != r2) PC = addr |
| 001101 | CALL | stack[SP] = PC + 1; SP = SP - 1; PC = addr |
| 001110 | RET | SP = SP + 1; PC = stack[SP] |
| 001111 | LD | r1 = memory[addr] |
| 010000 | ST | memory[addr] = r1 |
| 010001 | FFT | FFT(memory[r2], result); memory[r1] = result |
| 010010 | ENC | Encrypt(memory[r2]); memory[r1] = encrypted\_data |
| 010011 | DEC | Decrypt(memory[r2]); memory[r1] = decrypted\_data |