2.10 Hướng dẫn thiết lập

Kiến trúc 8031 có thiết lập vô cùng mạnh mẽ và linh hoạt cho phép người dùng phát triển một chương trình nhỏ gọn (nén).

2.10.1 Hướng thiết lập truyền dữ liệu

|  |  |  |
| --- | --- | --- |
| Bộ nhớ | Vận hành | Thời gian thực hiện (µs) với mỗi 12MHz |
| MOV A, <src> | A = <src> | 1 |
| MOV <dest> , A | <dest> = A | 1 |
| MOV <dest> , <src> | <dest> = <src> | 2 |
| MOV DPTR, #data16 | DPTR = hằng số tức thời 16 bit | 2 |
| PUSH <src> | INC SP : MOV”@SP”, <src> | 2 |
| POP <dest> | MOV <dest> , “@SP” : DEC SP | 2 |
| XCH A, <byte> | ACC và <byte> trao đổi dữ liệu | 1 |
| XCHD a, @Ri | ACC và @Ri trao đổi các mảnh | 1 |

2.10.2 Hướng thiết lập truyền dữ liệu trên RAM ngoài

|  |  |  |  |
| --- | --- | --- | --- |
| Độ dài địa chỉ | Bộ nhớ | Vận hành | Thời gian thực hiện (µs) với mỗi 12MHz |
| 8 bit | MOV x A, @Ri | Đọc RAM ngoài @Ri | 2 |
| 8 bit | MOV x @Ri, A | Ghi RAM ngoài @Ri | 2 |
| 16 bit | MOV x A, @DPTR | Đọc RAM ngoài @DPTR | 2 |
| 16 bit | MOV x @DPTR, A | Ghi RAM ngoài @DPTR | 2 |

2.10.3 Các tham chiếu

|  |  |  |
| --- | --- | --- |
| Bộ nhớ | Vận hành | Thời gian thực hiện (µs) với mỗi 12MHz |
| MOVC A,@A + DPTR | Đọc bộ nhớ chươn trình tại ( A + DPTR) | 2 |
| MOVC A,A + PC | Đọc bộ nhớ chương trình tại (A+ PC) | 2 |

2.10.4 Các hướng dẫn về số học

|  |  |  |
| --- | --- | --- |
| Bộ nhớ | Vận hành | Thời gian thực hiện (µs) với mỗi 12MHz |
| ADD A, <byte> | A= A+ <byte> | 1 |
| ADDC A, <byte> | A= A+ <byte> + C | 1 |
| SUBB A, <byte> | A= A - <byte> - C | 1 |
| INC A | A= A + 1 | 1 |
| INC <byte> | <byte> = <byte>+ 1 | 1 |
| INC DPTR | DPTR = DPTR + 1 | 2 |
| DEC A | A= A- 1 | 1 |
| DEC <byte> | <byte> = <byte> - 1 | 1 |
| MUL AB | B : A = B x A | 4 |
| DIV AB | A= Int [A/B]  B= Mod [A/B] | 4 |
| DA A | điều chỉnh phần thập phân | 1 |

2.10.5 Hướng dẫn về mặt logic

|  |  |  |
| --- | --- | --- |
| Bộ nhớ | Vận hành | Thời gian thực hiện (µs) với mỗi 12MHz |
| ANL A, <byte> | A= A. AND <byte> | 1 |
| ANL <byte>, a | <byte> = <byte> .AND . A | 1 |
| ANL <byte>, #data | <byte> = <byte> .AND . #data | 2 |
| ORL A, <byte> | A= A. OR . <byte> | 1 |
| ORL <byte>, A | <byte> = <byte> . OR . A | 1 |
| ORL <byte>, #data | <byte> = <byte> . OR . #data | 2 |
| XRL A, <byte> | A= A. XOR. <byte> | 1 |
| XRL <byte>, A | <byte> = <byte> . XOR . A | 1 |
| XRL <byte>, #data | <byte> = <byte> . XOR. #data | 2 |
| CLR A | A= 00H | 1 |
| CPL A | A= .NOT. A | 1 |
| RL A | Dịch ACC sang trái 1 bit | 1 |
| RLC A | Dịch trái thông qua Carry | 1 |
| RR A | Dịch ACC sang phải 1 bit | 1 |
| RRC A | Dịch phải thông qua Carry | 1 |
| SWAP A | Hoán đổi các mảnh trong A | 1 |

2.10.6 Kiểm soát chương trình – Các bước nhảy, gọi hàm và trả về kết quả

Các bước nhảy dài mã hóa các địa chỉ 16 bit, kiểm soát sử dụng AJMP.

2.10.7 Hướng thiết lập bước nhảy

|  |  |  |
| --- | --- | --- |
| Bộ nhớ | Vận hành | Thời gian thực hiện (µs) với mỗi 12MHz |
| JMP addr | Nhảy tới addr | 2 |
| JMP @A+ DPTR | Nhảy tới A + DPTR | 2 |
| CALL addr | Gọi hàm con tại addr | 2 |
| RET | Trả về từ hàm con | 2 |
| RETI | Trả về nếu lỗi | 2 |
| NOP | Không vận hành được | 1 |

2.10.8 Hướng thiết lập điều kiện các bước nhảy

|  |  |  |
| --- | --- | --- |
| Bộ nhớ | Vận hành | Thời gian thực hiện (µs) với mỗi 12MHz |
| JZ rel | Nhảy nếu A = 0 | 2 |
| JNZ rel | Nhảy nếu A ≠ 0 | 2 |
| JC rel | Nhảy nếu C= 1 | 2 |
| JNC rel | Nhảy nếu C= 0 | 2 |

2.10.9 Hướng vận hành và phân nhánh

|  |  |  |
| --- | --- | --- |
| Bộ nhớ | Vận hành | Thời gian thực hiện (µs) với mỗi 12MHz |
| DJNZ <byte>, rel | Giảm dần và nhảy nếu khác 0 | 2 |
| CJNE A< <byte>, rel | Nhảy nếu A ≠ <byte> | 2 |
| CJNE <byte>, #data , rel | Nhảy nếu <byte> ≠ #data | 2 |

2.10.10 Hướng dẫn về mặt đại số

|  |  |  |
| --- | --- | --- |
| Bộ nhớ | Vận hành | Thời gian thực hiện (µs) với mỗi 12MHz |
| ANL C, bit | C= C. AND. bit | 2 |
| ANL C, /bit | C= C. AND. NOT. bit | 2 |
| ORL C, bit | C= C. OR. bit | 2 |
| ORL C, / bit | C= C. OR. NOT. bit | 2 |
| MOV C, bit | C = bit | 1 |
| MOV bit.C | bit= C | 2 |
| CLR C | C = 0 | 1 |
| CLR nit | bit= 0 | 1 |
| SETB C | C= 1 | 1 |
| SETB bit | bit= 1 | 1 |
| CPL C | C= .NOT. C | 1 |
| CPLbit | Bit= .NOT. bit | 1 |
| JC rel | Nhảy nếu C = 1 | 2 |
| JNC rel | Nhảy nếu C = 0 | 2 |
| JB bit, rel | Nhảy nếu bit = 1 | 2 |
| JNB bit, rel | Nhảy nếu bit = 0 | 2 |
| JBC bit, rel | Nhảy nếu bit = 1, CLR bit | 2 |