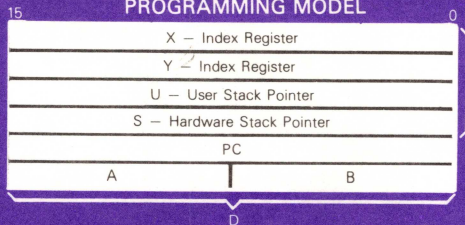
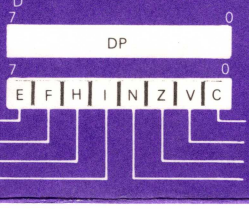
# MC6809 — MC6809E

## 8-bit microprocessor

## Reference Card


### PROGRAMMING MODEL




**MOTOROLA INC.**
*MOS Integrated Circuits Division*
3501 ED BLUESTEIN BLVD. AUSTIN, TEXAS 78721

M6809(AC3)

| OP | MNEM | MODE | ~ | # |
|---|---|---|---|---|
| 00 | NEG | DIRECT | 6 | 2 |
| 03 | COM | | 6 | 2 |
| 04 | LSR | | 6 | 2 |
| 06 | ROR | | 6 | 2 |
| 07 | ASR | | 6 | 2 |
| 08 | ASL/LSL | | 6 | 2 |
| 09 | ROL | | 6 | 2 |
| 0A | DEC | | 6 | 2 |
| 0C | INC | | 6 | 2 |
| 0D | TST | | 6 | 2 |
| 0E | JMP | | 3 | 2 |
| 0F | CLR | DIRECT | 6 | 2 |
| 12 | NOP | INHERENT | 2 | 1 |
| 13 | SYNC | INHERENT | 4 | 1 |
| 16 | LBRA | RELATIVE | 5 | 3 |
| 17 | LBSR | RELATIVE | 9 | 3 |
| 19 | DAA | INHERENT | 2 | 1 |
| 1A | ORCC | IMMED | 3 | 2 |
| 1C | ANDCC | IMMED | 3 | 2 |
| 1D | SEX | INHERENT | 2 | 1 |
| 1E | EXG | IMMED | 8 | 2 |
| 1F | TFR | IMMED | 6 | 2 |
| 20 | BRA | RELATIVE | 3 | 2 |
| 21 | BRN | | 3 | 2 |
| 22 | BHI | | 3 | 2 |
| 23 | BLS | | 3 | 2 |
| 24 | BHS/BCC | | 3 | 2 |
| 25 | BLO/BCS | | 3 | 2 |
| 26 | BNE | | 3 | 2 |
| 27 | BEQ | | 3 | 2 |
| 28 | BVC | | 3 | 2 |
| 29 | BVS | | 3 | 2 |
| 2A | BPL | | 3 | 2 |
| 2B | BMI | | 3 | 2 |
| 2C | BGE | | 3 | 2 |
| 2D | BLT | RELATIVE | 3 | 2 |
| 2E | BGT | RELATIVE | 3 | 2 |
| 2F | BLE | RELATIVE | 3 | 2 |
| 30 | LEAX | INDEXED | 4 | 2 |
| 31 | LEAY | | 4 | 2 |
| 32 | LEAS | | 4 | 2 |
| 33 | LEAU | INDEXED | 4 | 2 |
| 34 | PSHS | IMMED | 5 | 2 |
| 35 | PULS | | 5 | 2 |
| 36 | PSHU | | 5 | 2 |
| 37 | PULU | IMMED | 5 | 2 |
| 39 | RTS | INHERENT | 5 | 1 |
| 3A | ABX | | 3 | 1 |
| 3B | RTI | INHERENT | 6/15 | 1 |
| 3C | CWAI | IMMED | 20 | 2 |
| 3D | MUL | INHERENT | 11 | 1 |
| 3F | SWI | | 19 | 1 |
| 40 | NEGA | | 2 | 1 |
| 43 | COMA | INHERENT | 2 | 1 |

| OP | MNEM | MODE | ~ | # |
|---|---|---|---|---|
| 44 | LSRA | INHERENT | 2 | 1 |
| 46 | RORA | | 2 | 1 |
| 47 | ASRA | | 2 | 1 |
| 48 | ASLA/LSLA | | 2 | 1 |
| 49 | ROLA | | 2 | 1 |
| 4A | DECA | | 2 | 1 |
| 4C | INCA | | 2 | 1 |
| 4D | TSTA | | 2 | 1 |
| 4F | CLRA | | 2 | 1 |
| 50 | NEGB | | 2 | 1 |
| 53 | COMB | | 2 | 1 |
| 54 | LSRB | | 2 | 1 |
| 56 | RORB | | 2 | 1 |
| 57 | ASRB | | 2 | 1 |
| 58 | ASLB/LSLB | | 2 | 1 |
| 59 | ROLB | | 2 | 1 |
| 5A | DECB | | 2 | 1 |
| 5C | INCB | INHERENT | 2 | 1 |
| 5D | TSTB | INHERENT | 2 | 1 |
| 5F | CLRB | INHERENT | 2 | 1 |
| 60 | NEG | INDEXED | 6 | 2 |
| 63 | COM | | 6 | 2 |
| 64 | LSR | | 6 | 2 |
| 66 | ROR | | 6 | 2 |
| 67 | ASR | | 6 | 2 |
| 68 | ASL/LSL | | 6 | 2 |
| 69 | ROL | | 6 | 2 |
| 6A | DEC | | 6 | 2 |
| 6C | INC | | 6 | 2 |
| 6D | TST | | 6 | 2 |
| 6E | JMP | | 3 | 2 |
| 6F | CLR | INDEXED | 6 | 2 |
| 70 | NEG | EXTENDED | 7 | 3 |
| 73 | COM | | 7 | 3 |
| 74 | LSR | | 7 | 3 |
| 76 | ROR | EXTENDED | 7 | 3 |
| 77 | ASR | EXTENDED | 7 | 3 |
| 78 | ASL/LSL | | 7 | 3 |
| 79 | ROL | | 7 | 3 |
| 7A | DEC | | 7 | 3 |
| 7C | INC | | 7 | 3 |
| 7D | TST | | 7 | 3 |
| 7E | JMP | | 4 | 3 |
| 7F | CLR | EXTENDED | 7 | 3 |
| 80 | SUBA | IMMED | 2 | 2 |
| 81 | CMPA | | 2 | 2 |
| 82 | SBCA | | 2 | 2 |
| 83 | SUBD | | 4 | 3 |
| 84 | ANDA | | 2 | 2 |
| 85 | BITA | | 2 | 2 |
| 86 | LDA | | 2 | 2 |
| 88 | EORA | | 2 | 2 |
| 89 | ADCA | | 2 | 2 |
| 8A | ORA | IMMED | 2 | 2 |

| OP | MNEM | MODE | ~ | # |
|---|---|---|---|---|
| 8B | ADDA | IMMED | 2 | 2 |
| 8C | CMPX | IMMED | 4 | 3 |
| 8D | BSR | RELATIVE | 7 | 2 |
| 8E | LDX | IMMED | 3 | 3 |
| 90 | SUBA | DIRECT | 4 | 2 |
| 91 | CMPA | | 4 | 2 |
| 92 | SBCA | | 4 | 2 |
| 93 | SUBD | | 6 | 2 |
| 94 | ANDA | | 4 | 2 |
| 95 | BITA | | 4 | 2 |
| 96 | LDA | | 4 | 2 |
| 97 | STA | | 4 | 2 |
| 98 | EORA | | 4 | 2 |
| 99 | ADCA | | 4 | 2 |
| 9A | ORA | | 4 | 2 |
| 9B | ADDA | | 4 | 2 |
| 9C | CMPX | | 6 | 2 |
| 9D | JSR | DIRECT | 7 | 2 |
| 9E | LDX | DIRECT | 5 | 2 |
| 9F | STX | DIRECT | 5 | 2 |
| A0 | SUBA | INDEXED | 4 | 2 |
| A1 | CMPA | | 4 | 2 |
| A2 | SBCA | | 4 | 2 |
| A3 | SUBD | | 6 | 2 |
| A4 | ANDA | | 4 | 2 |
| A5 | BITA | | 4 | 2 |
| A6 | LDA | | 4 | 2 |
| A7 | STA | | 4 | 2 |
| A8 | EORA | | 4 | 2 |
| A9 | ADCA | | 4 | 2 |
| AA | ORA | | 4 | 2 |
| AB | ADDA | | 4 | 2 |
| AC | CMPX | | 6 | 2 |
| AD | JSR | | 7 | 2 |
| AE | LDX | | 5 | 2 |
| AF | STX | INDEXED | 5 | 2 |
| B0 | SUBA | EXTENDED | 5 | 3 |
| B1 | CMPA | | 5 | 3 |
| B2 | SBCA | | 5 | 3 |
| B3 | SUBD | | 7 | 3 |
| B4 | ANDA | | 5 | 3 |
| B5 | BITA | | 5 | 3 |
| B6 | LDA | | 5 | 3 |
| B7 | STA | | 5 | 3 |
| B8 | EORA | | 5 | 3 |
| B9 | ADCA | | 5 | 3 |
| BA | ORA | | 5 | 3 |
| BB | ADDA | | 5 | 3 |
| BC | CMPX | | 7 | 3 |
| BD | JSR | | 8 | 3 |
| BE | LDX | | 6 | 3 |
| BF | STX | EXTENDED | 6 | 3 |
| C0 | SUBB | IMMED | 2 | 2 |
| C1 | CMPB | IMMED | 2 | 2 |

| OP | MNEM | MODE | ~ | # |
|---|---|---|---|---|
| C2 | SBCB | IMMED | 2 | 2 |
| C3 | ADDD | | 4 | 3 |
| C4 | ANDB | | 2 | 2 |
| C5 | BITB | | 2 | 2 |
| C6 | LDB | | 2 | 2 |
| C8 | EORB | | 2 | 2 |
| C9 | ADCB | | 2 | 2 |
| CA | ORB | | 2 | 2 |
| CB | ADDB | | 2 | 2 |
| CC | LDD | | 3 | 3 |
| CE | LDU | IMMED | 3 | 3 |
| D0 | SUBB | DIRECT | 4 | 2 |
| D1 | CMPB | | 4 | 2 |
| D2 | SBCB | | 4 | 2 |
| D3 | ADDD | | 6 | 2 |
| D4 | ANDB | | 4 | 2 |
| D5 | BITB | | 4 | 2 |
| D6 | LDB | DIRECT | 4 | 2 |
| D7 | STB | DIRECT | 4 | 2 |
| D8 | EORB | | 4 | 2 |
| D9 | ADCB | | 4 | 2 |
| DA | ORB | | 4 | 2 |
| DB | ADDB | | 4 | 2 |
| DC | LDD | | 5 | 2 |
| DD | STD | | 5 | 2 |
| DE | LDU | | 5 | 2 |
| DF | STU | DIRECT | 5 | 2 |
| E0 | SUBB | INDEXED | 4 | 2 |
| E1 | CMPB | | 4 | 2 |
| E2 | SBCB | | 4 | 2 |
| E3 | ADDD | | 6 | 2 |
| E4 | ANDB | | 4 | 2 |
| E5 | BITB | | 4 | 2 |
| E6 | LDB | | 4 | 2 |
| E7 | STB | | 4 | 2 |
| E8 | EORB | INDEXED | 4 | 2 |
| E9 | ADCB | INDEXED | 4 | 2 |
| EA | ORB | | 4 | 2 |
| EB | ADDB | | 4 | 2 |
| EC | LDD | | 5 | 2 |
| ED | STD | | 5 | 2 |
| EE | LDU | | 5 | 2 |
| EF | STU | INDEXED | 5 | 2 |
| F0 | SUBB | EXTENDED | 5 | 3 |
| F1 | CMPB | | 5 | 3 |
| F2 | SBCB | | 5 | 3 |
| F3 | ADDD | | 7 | 3 |
| F4 | ANDB | | 5 | 3 |
| F5 | BITB | | 5 | 3 |
| F6 | LDB | | 5 | 3 |
| F7 | STB | | 5 | 3 |
| F8 | EORB | | 5 | 3 |
| F9 | ADCB | | 5 | 3 |
| FA | ORB | EXTENDED | 5 | 3 |
| FB | ADDB | EXTENDED | 5 | 3 |
| FC | LDD | | 6 | 3 |
| FD | STD | | 6 | 3 |
| FE | LDU | | 6 | 3 |
| FF | STU | EXTENDED | 6 | 3 |
| 1021 | LBRN | RELATIVE | 5 | 4 |
| 1022 | LBHI | | 5(6) | 4 |
| 1023 | LBLS | | 5(6) | 4 |
| 1024 | LBHS/LBCC | | 5(6) | 4 |
| 1025 | LBCS/LBLO | | 5(6) | 4 |
| 1026 | LBNE | | 5(6) | 4 |
| 1027 | LBEQ | | 5(6) | 4 |
| 1028 | LBVC | | 5(6) | 4 |
| 1029 | LBVS | | 5(6) | 4 |
| 102A | LBPL | | 5(6) | 4 |
| 102B | LBMI | | 5(6) | 4 |
| 102C | LBGE | | 5(6) | 4 |
| 102D | LBLT | RELATIVE | 5(6) | 4 |
| 102E | LBGT | RELATIVE | 5(6) | 4 |
| 102F | LBLE | RELATIVE | 5(6) | 4 |
| 103F | SWI2 | INHERENT | 20 | 2 |
| 1083 | CMPD | IMMED | 5 | 4 |
| 108C | CMPY | | 5 | 4 |
| 108E | LDY | IMMED | 4 | 4 |
| 1093 | CMPD | DIRECT | 7 | 3 |
| 109C | CMPY | | 7 | 3 |
| 109E | LDY | | 6 | 3 |
| 109F | STY | DIRECT | 6 | 3 |
| 10A3 | CMPD | INDEXED | 7 | 3 |
| 10AC | CMPY | | 7 | 3 |
| 10AE | LDY | | 6 | 3 |
| 10AF | STY | INDEXED | 6 | 3 |
| 10B3 | CMPD | EXTENDED | 8 | 4 |
| 10BC | CMPY | | 8 | 4 |
| 10BE | LDY | | 7 | 4 |
| 10BF | STY | EXTENDED | 7 | 4 |
| 10CE | LDS | IMMED | 4 | 4 |
| 10DE | LDS | DIRECT | 6 | 3 |
| 10DF | STS | DIRECT | 6 | 3 |
| 10EE | LDS | INDEXED | 6 | 3 |
| 10EF | STS | INDEXED | 6 | 3 |
| 10FE | LDS | EXTENDED | 7 | 4 |
| 10FF | STS | EXTENDED | 7 | 4 |
| 113F | SWI3 | INHERENT | 20 | 2 |
| 1183 | CMPU | IMMED | 5 | 4 |
| 118C | CMPS | IMMED | 5 | 4 |
| 1193 | CMPU | DIRECT | 7 | 3 |
| 119C | CMPS | DIRECT | 7 | 3 |
| 11A3 | CMPU | INDEXED | 7 | 3 |
| 11AC | CMPS | INDEXED | 7 | 3 |
| 11B3 | CMPU | EXTENDED | 8 | 4 |
| 11BC | CMPS | EXTENDED | 8 | 4 |

## STACKING ORDER

Pull Order ↓

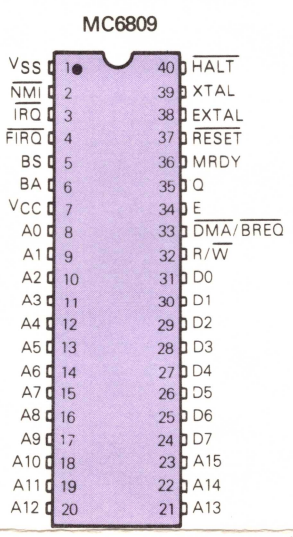
CC
A
B
DP
X Hi
X Lo
Y Hi
Y Lo
U/S Hi
U/S Lo
PC Hi
PC Lo

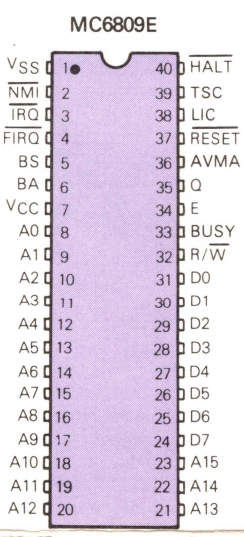
↑ Push Order

Increasing Memory ↓

## INTERRUPT VECTORS

| | |
|---|---|
| FFFE | Restart |
| FFFC | $\overline{NMI}$ |
| FFFA | SWI |
| FFF8 | $\overline{IRQ}$ |
| FFF6 | $\overline{FIRQ}$ |
| FFF4 | SWI2 |
| FFF2 | SWI3 |
| FFF0 | Reserved |




HEXADECIMAL AND DECIMAL CONVERSION

HOW TO USE THE TABLES

CONVERSION TO DECIMAL. Find the decimal weights for corresponding hexadecimal characters beginning with the least significant character. The sum of the decimal weight is the decimal value of the hexadecimal number.

CONVERSION TO HEXADECIMAL. Find the highest decimal value in the table which is lower than or equal to the decimal number to be converted. The corresponding hexadecimal character is the most significant character. Subtract the decimal value found from the decimal number to be converted. With the difference, repeat the process to find subsequent hexadecimal characters.

HEXADECIMAL AND DECIMAL CONVERSION

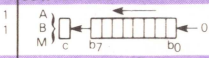
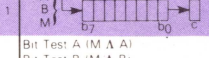
| 15 BYTE 8 | | | | 7 BYTE 0 | | | |
|---|---|---|---|---|---|---|---|
| 15 CHAR 12 | | 11 CHAR 8 | | 7 CHAR 4 | | 3 CHAR 0 | |
| HEX | DEC | HEX | DEC | HEX | DEC | HEX | DEC |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 4 096 | 1 | 256 | 1 | 16 | 1 | 1 |
| 2 | 8 192 | 2 | 512 | 2 | 32 | 2 | 2 |
| 3 | 12 288 | 3 | 768 | 3 | 48 | 3 | 3 |
| 4 | 16 384 | 4 | 1 024 | 4 | 64 | 4 | 4 |
| 5 | 20 480 | 5 | 1 280 | 5 | 80 | 5 | 5 |
| 6 | 24 576 | 6 | 1 536 | 6 | 96 | 6 | 6 |
| 7 | 28 672 | 7 | 1 792 | 7 | 112 | 7 | 7 |
| 8 | 32 768 | 8 | 2 048 | 8 | 128 | 8 | 8 |
| 9 | 36 864 | 9 | 2 304 | 9 | 144 | 9 | 9 |
| A | 40 960 | A | 2 560 | A | 160 | A | 10 |
| B | 45 056 | B | 2 816 | B | 176 | B | 11 |
| C | 49 152 | C | 3 072 | C | 192 | C | 12 |
| D | 53 248 | D | 3 328 | D | 208 | D | 13 |
| E | 57 344 | E | 3 584 | E | 224 | E | 14 |
| F | 61 440 | F | 3 840 | F | 240 | F | 15 |

## ASCII CHARACTER SET

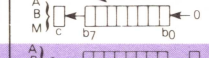
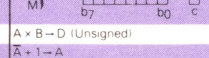
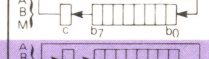
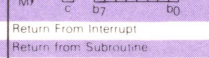
| Least Significant Character \ Most Significant Character | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| Hex | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 0 | NUL | DLE | SP | 0 | @ | P | ` | p |
| 1 | SOH | DC1 | ! | 1 | A | Q | a | q |
| 2 | STX | DC2 | " | 2 | B | R | b | r |
| 3 | ETX | DC3 | # | 3 | C | S | c | s |
| 4 | EOT | DC4 | $ | 4 | D | T | d | t |
| 5 | ENQ | NAK | % | 5 | E | U | e | u |
| 6 | ACK | SYN | & | 6 | F | V | f | v |
| 7 | BEL | ETB | ' | 7 | G | W | g | w |
| 8 | BS | CAN | ( | 8 | H | X | h | x |
| 9 | HT | EM | ) | 9 | I | Y | i | y |
| A | LF | SUB | * | : | J | Z | j | z |
| B | VT | ESC | + | ; | K | [ | k | { |
| C | FF | FS | , | < | L | \ | l | \| |
| D | CR | GS | - | = | M | ] | m | } |
| E | SO | RS | . | > | N | ^ | n | ~ |
| F | SI | US | / | ? | O | _ | o | DEL |

POWERS OF TWO

| $2^n$ | n | $2^n$ | n | $2^n$ | n |
|---|---|---|---|---|---|
| 1 | 0 | 128 | 7 | 16,384 | 14 |
| 2 | 1 | 256 | 8 | 32,768 | 15 |
| 4 | 2 | 512 | 9 | 65,536 | 16 |
| 8 | 3 | 1,024 | 10 | 131,072 | 17 |
| 16 | 4 | 2,048 | 11 | 262,144 | 18 |
| 32 | 5 | 4096 | 12 | 524,288 | 19 |
| 64 | 6 | 8,192 | 13 | 1,048,576 | 20 |

| Instruction | Forms | Immediate | | | Direct | | | Indexed[1] | | | Extended | | | Inherent | | | Description | 5 H | 3 N | 2 Z | 1 V | 0 C |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Op | ~ | # | Op | ~ | # | Op | ~ | # | Op | ~ | # | Op | ~ | # | | | | | | |
| ABX | | | | | | | | | | | | | | 3A | 3 | 1 | B + X → X (Unsigned) | • | • | • | • | • |
| ADC | ADCA | 89 | 2 | 2 | 99 | 4 | 2 | A9 | 4+ | 2+ | B9 | 5 | 3 | | | | A + M + C → A | ↕ | ↕ | ↕ | ↕ | ↕ |
| | ADCB | C9 | 2 | 2 | D9 | 4 | 2 | E9 | 4+ | 2+ | F9 | 5 | 3 | | | | B + M + C → B | ↕ | ↕ | ↕ | ↕ | ↕ |
| ADD | ADDA | 8B | 2 | 2 | 9B | 4 | 2 | AB | 4+ | 2+ | BB | 5 | 3 | | | | A + M → A | ↕ | ↕ | ↕ | ↕ | ↕ |
| | ADDB | CB | 2 | 2 | DB | 4 | 2 | EB | 4+ | 2+ | FB | 5 | 3 | | | | B + M → B | ↕ | ↕ | ↕ | ↕ | ↕ |
| | ADDD | C3 | 4 | 3 | D3 | 6 | 2 | E3 | 6+ | 2+ | F3 | 7 | 3 | | | | D + M:M + 1 → D | • | ↕ | ↕ | ↕ | ↕ |
| AND | ANDA | 84 | 2 | 2 | 94 | 4 | 2 | A4 | 4+ | 2+ | B4 | 5 | 3 | | | | A ∧ M → A | • | ↕ | ↕ | 0 | • |
| | ANDB | C4 | 2 | 2 | D4 | 4 | 2 | E4 | 4+ | 2+ | F4 | 5 | 3 | | | | B ∧ M → B | • | ↕ | ↕ | 0 | • |
| | ANDCC | 1C | 3 | 2 | | | | | | | | | | | | | CC ∧ IMM → CC | | | | | ↕ |
| ASL | ASLA | | | | | | | | | | | | | 48 | 2 | 1 |  | 7 | ↕ | ↕ | ↕ | ↕ |
| | ASLB | | | | | | | | | | | | | 58 | 2 | 1 | | 7 | ↕ | ↕ | ↕ | ↕ |
| | ASL | | | | 08 | 6 | 2 | 68 | 6+ | 2+ | 78 | 7 | 3 | | | | | 7 | ↕ | ↕ | ↕ | ↕ |
| ASR | ASRA | | | | | | | | | | | | | 47 | 2 | 1 |  | 7 | ↕ | ↕ | • | ↕ |
| | ASRB | | | | | | | | | | | | | 57 | 2 | 1 | | 7 | ↕ | ↕ | • | ↕ |
| | ASR | | | | 07 | 6 | 2 | 67 | 6+ | 2+ | 77 | 7 | 3 | | | | | 7 | ↕ | ↕ | • | ↕ |
| BIT | BITA | 85 | 2 | 2 | 95 | 4 | 2 | A5 | 4+ | 2+ | B5 | 5 | 3 | | | | Bit Test A (M ∧ A) | • | ↕ | ↕ | 0 | • |
| | BITB | C5 | 2 | 2 | D5 | 4 | 2 | E5 | 4+ | 2+ | F5 | 5 | 3 | | | | Bit Test B (M ∧ B) | • | ↕ | ↕ | 0 | • |
| CLR | CLRA | | | | | | | | | | | | | 4F | 2 | 1 | 0 → A | • | 0 | 1 | 0 | 0 |
| | CLRB | | | | | | | | | | | | | 5F | 2 | 1 | 0 → B | • | 0 | 1 | 0 | 0 |
| | CLR | | | | 0F | 6 | 2 | 6F | 6+ | 2+ | 7F | 7 | 3 | | | | 0 → M | • | 0 | 1 | 0 | 0 |
| CMP | CMPA | 81 | 2 | 2 | 91 | 4 | 2 | A1 | 4+ | 2+ | B1 | 5 | 3 | | | | Compare M from A | 7 | ↕ | ↕ | ↕ | ↕ |
| | CMPB | C1 | 2 | 2 | D1 | 4 | 2 | E1 | 4+ | 2+ | F1 | 5 | 3 | | | | Compare M from B | 7 | ↕ | ↕ | ↕ | ↕ |
| | CMPD | 10 | 5 | 4 | 10 | 7 | 3 | 10 | 7+ | 3+ | 10 | 8 | 4 | | | | Compare M:M + 1 from D | • | ↕ | ↕ | ↕ | ↕ |
| | | 83 | | | 93 | | | A3 | | | B3 | | | | | | | | | | | |
| | CMPS | 11 | 5 | 4 | 11 | 7 | 3 | 11 | 7+ | 3+ | 11 | 8 | 4 | | | | Compare M:M + 1 from S | • | ↕ | ↕ | ↕ | ↕ |
| | | 8C | | | 9C | | | AC | | | BC | | | | | | | | | | | |
| | CMPU | 11 | 5 | 4 | 11 | 7 | 3 | 11 | 7+ | 3+ | 11 | 8 | 4 | | | | Compare M:M + 1 from U | • | ↕ | ↕ | ↕ | ↕ |
| | | 83 | | | 93 | | | A3 | | | B3 | | | | | | | | | | | |
| | CMPX | 8C | 4 | 3 | 9C | 6 | 2 | AC | 6+ | 2+ | BC | 7 | 3 | | | | Compare M:M + 1 from X | • | ↕ | ↕ | ↕ | ↕ |
| | CMPY | 10 | 5 | 4 | 10 | 7 | 3 | 10 | 7+ | 3+ | 10 | 8 | 4 | | | | Compare M:M + 1 from Y | • | ↕ | ↕ | ↕ | ↕ |
| | | 8C | | | 9C | | | AC | | | BC | | | | | | | | | | | |

| Instruction | Forms | Immediate | | | Direct | | | Indexed[1] | | | Extended | | | Inherent | | | Description | 5 H | 3 N | 2 Z | 1 V | 0 C |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Op | ~ | # | Op | ~ | # | Op | ~ | # | Op | ~ | # | Op | ~ | # | | | | | | |
| COM | COMA | | | | | | | | | | | | | 43 | 2 | 1 | $\overline{A} \rightarrow A$ | • | ↕ | ↕ | 0 | 1 |
| | COMB | | | | | | | | | | | | | 53 | 2 | 1 | $\overline{B} \rightarrow B$ | • | ↕ | ↕ | 0 | 1 |
| | COM | | | | 03 | 6 | 2 | 63 | 6+ | 2+ | 73 | 7 | 3 | | | | $\overline{M} \rightarrow M$ | • | ↕ | ↕ | 0 | 1 |
| CWAI | | 3C | ≥20 | 2 | | | | | | | | | | | | | CC ∧ IMM → CC Wait for Interrupt | | | | | 7 |
| DAA | | | | | | | | | | | | | | 19 | 2 | 1 | Decimal Adjust A | • | ↕ | ↕ | 0 | ↕ |
| DEC | DECA | | | | | | | | | | | | | 4A | 2 | 1 | A − 1 → A | • | ↕ | ↕ | ↕ | • |
| | DECB | | | | | | | | | | | | | 5A | 2 | 1 | B − 1 → B | • | ↕ | ↕ | ↕ | • |
| | DEC | | | | 0A | 6 | 2 | 6A | 6+ | 2+ | 7A | 7 | 3 | | | | M − 1 → M | • | ↕ | ↕ | ↕ | • |
| EOR | EORA | 88 | 2 | 2 | 98 | 4 | 2 | A8 | 4+ | 2+ | B8 | 5 | 3 | | | | A ⊻ M → A | • | ↕ | ↕ | 0 | • |
| | EORB | C8 | 2 | 2 | D8 | 4 | 2 | E8 | 4+ | 2+ | F8 | 5 | 3 | | | | B ⊻ M → B | • | ↕ | ↕ | 0 | • |
| EXG | R1, R2 | 1E | 8 | 2 | | | | | | | | | | | | | R1 ↔ R2[2] | • | • | • | • | • |
| INC | INCA | | | | | | | | | | | | | 4C | 2 | 1 | A + 1 → A | • | ↕ | ↕ | ↕ | • |
| | INCB | | | | | | | | | | | | | 5C | 2 | 1 | B + 1 → B | • | ↕ | ↕ | ↕ | • |
| | INC | | | | 0C | 6 | 2 | 6C | 6+ | 2+ | 7C | 7 | 3 | | | | M + 1 → M | • | ↕ | ↕ | ↕ | • |
| JMP | | | | | 0E | 3 | 2 | 6E | 3+ | 2+ | 7E | 4 | 3 | | | | EA[3] → PC | • | • | • | • | • |
| JSR | | | | | 9D | 7 | 2 | AD | 7+ | 2+ | BD | 8 | 3 | | | | Jump to Subroutine | • | • | • | • | • |
| LD | LDA | 86 | 2 | 2 | 96 | 4 | 2 | A6 | 4+ | 2+ | B6 | 5 | 3 | | | | M → A | • | ↕ | ↕ | 0 | • |
| | LDB | C6 | 2 | 2 | D6 | 4 | 2 | E6 | 4+ | 2+ | F6 | 5 | 3 | | | | M → B | • | ↕ | ↕ | 0 | • |
| | LDD | CC | 3 | 3 | DC | 5 | 2 | EC | 5+ | 2+ | FC | 6 | 3 | | | | M:M + 1 → D | • | ↕ | ↕ | 0 | • |
| | LDS | 10 | 4 | 4 | 10 | 6 | 3 | 10 | 6+ | 3+ | 10 | 7 | 4 | | | | M:M + 1 → S | • | ↕ | ↕ | 0 | • |
| | | CE | | | DE | | | EE | | | FE | | | | | | | | | | | |
| | LDU | CE | 3 | 3 | DE | 5 | 2 | EE | 5+ | 2+ | FE | 6 | 3 | | | | M:M + 1 → U | • | ↕ | ↕ | 0 | • |
| | LDX | 8E | 3 | 3 | 9E | 5 | 2 | AE | 5+ | 2+ | BE | 6 | 3 | | | | M:M + 1 → X | • | ↕ | ↕ | 0 | • |
| | LDY | 10 | 4 | 4 | 10 | 6 | 3 | 10 | 6+ | 3+ | 10 | 7 | 4 | | | | M:M + 1 → Y | • | ↕ | ↕ | 0 | • |
| | | 8E | | | 9E | | | AE | | | BE | | | | | | | | | | | |
| LEA | LEAS | | | | | | | 32 | 4+ | 2+ | | | | | | | EA[3] → S | • | • | • | • | • |
| | LEAU | | | | | | | 33 | 4+ | 2+ | | | | | | | EA[3] → U | • | • | • | • | • |
| | LEAX | | | | | | | 30 | 4+ | 2+ | | | | | | | EA[3] → X | • | • | ↕ | • | • |
| | LEAY | | | | | | | 31 | 4+ | 2+ | | | | | | | EA[3] → Y | • | • | ↕ | • | • |

| Instruction | Forms | Immediate | | | Direct | | | Indexed[1] | | | Extended | | | Inherent | | | Description | 5 H | 3 N | 2 Z | 1 V | 0 C |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Op | ~ | # | Op | ~ | # | Op | ~ | # | Op | ~ | # | Op | ~ | # | | | | | | |
| LSL | LSLA | | | | | | | | | | | | | 48 | 2 | 1 |  | • | ↕ | ↕ | ↕ | ↕ |
| | LSLB | | | | | | | | | | | | | 58 | 2 | 1 | | • | ↕ | ↕ | ↕ | ↕ |
| | LSL | | | | 08 | 6 | 2 | 68 | 6+ | 2+ | 78 | 7 | 3 | | | | | • | ↕ | ↕ | ↕ | ↕ |
| LSR | LSRA | | | | | | | | | | | | | 44 | 2 | 1 |  | • | 0 | ↕ | • | ↕ |
| | LSRB | | | | | | | | | | | | | 54 | 2 | 1 | | • | 0 | ↕ | • | ↕ |
| | LSR | | | | 04 | 6 | 2 | 64 | 6+ | 2+ | 74 | | 3 | | | | | • | 0 | ↕ | • | ↕ |
| MUL | | | | | | | | | | | | | | 3D | 11 | 1 | A × B → D (Unsigned) | • | • | ↕ | • | 8 |
| NEG | NEGA | | | | | | | | | | | | | 40 | 2 | 1 | $\overline{A} + 1 \rightarrow A$ | ↕ | ↕ | ↕ | ↕ | ↕ |
| | NEGB | | | | | | | | | | | | | 50 | 2 | 1 | $\overline{B} + 1 \rightarrow B$ | ↕ | ↕ | ↕ | ↕ | ↕ |
| | NEG | | | | 00 | 6 | 2 | 60 | 6+ | 2+ | 70 | 7 | 3 | | | | $\overline{M} + 1 \rightarrow M$ | ↕ | ↕ | ↕ | ↕ | ↕ |
| NOP | | | | | | | | | | | | | | 12 | 2 | 1 | No Operation | • | • | • | • | • |
| OR | ORA | 8A | 2 | 2 | 9A | 4 | 2 | AA | 4+ | 2+ | BA | 5 | 3 | | | | A ∨ M → A | • | ↕ | ↕ | 0 | • |
| | ORB | CA | 2 | 2 | DA | 4 | 2 | EA | 4+ | 2+ | FA | 5 | 3 | | | | B ∨ M → B | • | ↕ | ↕ | 0 | • |
| | ORCC | 1A | 3 | 2 | | | | | | | | | | | | | CC ∨ IMM → CC | | | | 6 | |
| PSH | PSHS | 34 | 5+[4] | 2 | | | | | | | | | | | | | Push Registers on S Stack | • | • | • | • | • |
| | PSHU | 36 | 5+[4] | 2 | | | | | | | | | | | | | Push Registers on U Stack | • | • | • | • | • |
| PUL | PULS | 35 | 5+[4] | 2 | | | | | | | | | | | | | Pull Registers from S Stack | • | • | • | • | • |
| | PULU | 37 | 5+[4] | 2 | | | | | | | | | | | | | Pull Registers from U Stack | • | • | • | • | • |
| ROL | ROLA | | | | | | | | | | | | | 49 | 2 | 1 |  | • | ↕ | ↕ | ↕ | ↕ |
| | ROLB | | | | | | | | | | | | | 59 | 2 | 1 | | • | ↕ | ↕ | ↕ | ↕ |
| | ROL | | | | 09 | 6 | 2 | 69 | 6+ | 2+ | 79 | 7 | 3 | | | | | • | ↕ | ↕ | ↕ | ↕ |
| ROR | RORA | | | | | | | | | | | | | 46 | 2 | 1 |  | • | ↕ | ↕ | • | ↕ |
| | RORB | | | | | | | | | | | | | 56 | 2 | 1 | | • | ↕ | ↕ | • | ↕ |
| | ROR | | | | 06 | 6 | 2 | 66 | 6+ | 2+ | 76 | 7 | 3 | | | | | • | ↕ | ↕ | • | ↕ |
| RTI | | | | | | | | | | | | | | 3B | 6/15 | 1 | Return From Interrupt | | | | | 6 |
| RTS | | | | | | | | | | | | | | 39 | 5 | 1 | Return from Subroutine | • | • | • | • | • |
| SBC | SBCA | 82 | 2 | 2 | 92 | 4 | 2 | A2 | 4+ | 2+ | B2 | 5 | 3 | | | | A − M − C → A | 8 | ↕ | ↕ | ↕ | ↕ |
| | SBCB | C2 | 2 | 2 | D2 | 4 | 2 | E2 | 4+ | 2+ | F2 | 5 | 3 | | | | B − M − C → B | 8 | ↕ | ↕ | ↕ | ↕ |
| SEX | | | | | | | | | | | | | | 1D | 2 | 1 | Sign Extend B into A | • | ↕ | ↕ | 0 | • |

| Instruction | Forms | Immediate | | | Direct | | | Indexed[1] | | | Extended | | | Inherent | | | Description | 5 H | 3 N | 2 Z | 1 V | 0 C |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Op | ~ | # | Op | ~ | # | Op | ~ | # | Op | ~ | # | Op | ~ | # | | | | | | |
| ST | STA | | | | 97 | 4 | 2 | A7 | 4+ | 2+ | B7 | 5 | 3 | | | | A → M | • | ↕ | ↕ | 0 | • |
| | STB | | | | D7 | 4 | 2 | E7 | 4+ | 2+ | F7 | 5 | 3 | | | | B → M | • | ↕ | ↕ | 0 | • |
| | STD | | | | DD | 5 | 2 | ED | 5+ | 2+ | FD | 6 | 3 | | | | D → M:M + 1 | • | ↕ | ↕ | 0 | • |
| | STS | | | | 10 | 6 | 3 | 10 | 6+ | 3+ | 10 | 7 | 4 | | | | S → M:M + 1 | • | ↕ | ↕ | 0 | • |
| | | | | | DF | | | EF | | | FF | | | | | | | | | | | |
| | STU | | | | DF | 5 | 2 | EF | 5+ | 2+ | FF | 6 | 3 | | | | U → M:M + 1 | • | ↕ | ↕ | 0 | • |
| | STX | | | | 9F | 5 | 2 | AF | 5+ | 2+ | BF | 6 | 3 | | | | X → M:M + 1 | • | ↕ | ↕ | 0 | • |
| | STY | | | | 10 | 6 | 3 | 10 | | | 10 | 7 | 4 | | | | Y → M:M + 1 | • | ↕ | ↕ | 0 | • |
| | | | | | 9F | | | AF | 6+ | 3+ | BF | | | | | | | | | | | |
| SUB | SUBA | 80 | 2 | 2 | 90 | 4 | 2 | A0 | 4+ | 2+ | B0 | 5 | 3 | | | | A − M → A | ↕ | ↕ | ↕ | ↕ | ↕ |
| | SUBB | C0 | 2 | 2 | D0 | 4 | 2 | E0 | 4+ | 2+ | F0 | 5 | 3 | | | | B − M → B | ↕ | ↕ | ↕ | ↕ | ↕ |
| | SUBD | 83 | 4 | 3 | 93 | 6 | 2 | A3 | 6+ | 2+ | B3 | 7 | 3 | | | | D − M:M + 1 → D | • | ↕ | ↕ | ↕ | ↕ |
| SWI | SWI[5] | | | | | | | | | | | | | 3F | 19 | 1 | Software Interrupt 1 | • | • | • | • | • |
| | SWI2[5] | | | | | | | | | | | | | 10 3F | 20 | 2 | Software Interrupt 2 | • | • | • | • | • |
| | SWI3[5] | | | | | | | | | | | | | 11 3F | 20 | 1 | Software Interrupt 3 | • | • | • | • | • |
| SYNC | | | | | | | | | | | | | | 13 | ≥4 | 1 | Synchronize to Interrupt | • | • | • | • | • |
| TFR | R1, R2 | 1F | 6 | 2 | | | | | | | | | | | | | R1 → R2[2] | • | • | • | • | • |
| TST | TSTA | | | | | | | | | | | | | 4D | 2 | 1 | Test A | • | ↕ | ↕ | 0 | • |
| | TSTB | | | | | | | | | | | | | 5D | 2 | 1 | Test B | • | ↕ | ↕ | 0 | • |
| | TST | | | | 0D | 6 | 2 | 6D | 6+ | 2+ | 7D | 7 | 3 | | | | Test M | • | ↕ | ↕ | 0 | • |

Legend:

OP Operation Code (Hexadecimal)
~ Number of MPU Cycles
# Number of Program Bytes
+ Arithmetic Plus

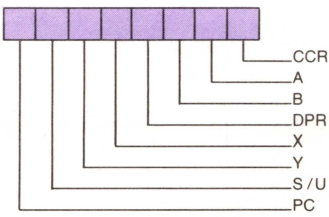
$\overline{M}$ Complement of M
→ Transfer Into
H Half-carry (from bit 3)
N Negative (sign bit)
Z Zero Result

↕ Test and set if true, cleared otherwise
• Not Affected
CC Condition Code Register
: Concatenation
V Logical or

Notes:

1. This column gives a base cycle and byte count. To obtain total count, add the values obtained from the INDEXED ADDRESSING MODES table.
2. R1 and R2 may be any pair of 8 bit or any pair of 16 bit registers.
   The 8 bit registers are: A, B, CC, DP
   The 16 bit registers are: X, Y, U, S, D, PC
3. EA is the effective address.
4. The PSH and PUL instructions require 5 cycles plus 1 cycle for each **byte** pushed or pulled.
5. SWI sets I and F bits. SWI2 and SWI3 do not affect I and F.
6. Conditions Codes set as a direct result of the instruction.
7. Value of half-carry flag is undefined.
8. Special Case – Carry set if b7 is SET.

## PUSH/PULL POST BYTE




## TRANSFER/EXCHANGE POST BYTE


### REGISTER FIELD (Source or Destination)

| | |
|---|---|
| 0000 = D (A:B) | 0101 = PC |
| 0001 = X | 1000 = A |
| 0010 = Y | 1001 = B |
| 0011 = U | 1010 = CCR |
| 0100 = S | 1011 = DPR |

## INDEXED ADDRESSING MODES

| Type | Forms | Non Indirect | | ~ | # | Indirect | | ~ | # |
|---|---|---|---|---|---|---|---|---|---|
| | | Assembler Form | Postbyte OP Code | | | Assembler Form | Postbyte OP Code | | |
| Constant Offset From R (twos complement offset) | No Offset | ,R | 1RR00100 | 0 | 0 | [,R] | 1RR10100 | 3 | 0 |
| | 5 Bit Offset | n, R | 0RRnnnnn | 1 | 0 | defaults to 8-bit | | | |
| | 8 Bit Offset | n, R | 1RR01000 | 1 | 1 | [n, R] | 1RR11000 | 4 | 1 |
| | 16 Bit Offset | n, R | 1RR01001 | 4 | 2 | [n, R] | 1RR11001 | 7 | 2 |
| Accumulator Offset From R (twos complement offset) | A — Register Offset | A, R | 1RR00110 | 1 | 0 | [A, R] | 1RR10110 | 4 | 0 |
| | B — Register Offset | B, R | 1RR00101 | 1 | 0 | [B, R] | 1RR10101 | 4 | 0 |
| | D — Register Offset | D, R | 1RR01011 | 4 | 0 | [D, R] | 1RR11011 | 7 | 0 |
| Auto Increment/Decrement R | Increment By 1 | ,R+ | 1RR00000 | 2 | 0 | not allowed | | | |
| | Increment By 2 | ,R++ | 1RR00001 | 3 | 0 | [,R++] | 1RR10001 | 6 | 0 |
| | Decrement By 1 | ,-R | 1RR00010 | 2 | 0 | not allowed | | | |
| | Decrement By 2 | ,--R | 1RR00011 | 3 | 0 | [,--R] | 1RR10011 | 6 | 0 |
| Constant Offset From PC (twos complement offset) | 8 Bit Offset | n, PCR | 1XX01100 | 1 | 1 | [n, PCR] | 1XX11100 | 4 | 1 |
| | 16 Bit Offset | n, PCR | 1XX01101 | 5 | 2 | [n, PCR] | 1XX11101 | 8 | 2 |
| Extended Indirect | 16 Bit Address | — | — | — | -- | [n] | 10011111 | 5 | 2 |

R = X, Y, U or S  X = 00  Y = 01
X = Don't Care  U = 10  S = 11

~ and # Indicate the number of additional cycles and bytes for the particular variation.

## BRANCH INSTRUCTIONS

| Instruction | Forms | Addressing Mode Relative OP | ~ | # | Description | 5 H | 3 N | 2 Z | 1 V | 0 C |
|---|---|---|---|---|---|---|---|---|---|---|
| BCC | BCC | 24 | 3 | 2 | Branch C = 0 | • | • | • | • | • |
| | LBCC | 10 5(6) 24 | | 4 | Long Branch C = 0 | • | • | • | • | • |
| BCS | BCS | 25 | 3 | 2 | Branch C = 1 | • | • | • | • | • |
| | LBCS | 10 5(6) 25 | | 4 | Long Branch C = 1 | • | • | • | • | • |
| BEQ | BEQ | 27 | 3 | 2 | Branch Z = 1 | • | • | • | • | • |
| | LBEQ | 10 5(6) 27 | | 4 | Long Branch Z = 1 | • | • | • | • | • |
| BGE | BGE | 2C | 3 | 2 | Branch ≥ Zero | • | • | • | • | • |
| | LBGE | 10 5(6) 2C | | 4 | Long Branch ≥ Zero | • | • | • | • | • |
| BGT | BGT | 2E | 3 | 2 | Branch > Zero | • | • | • | • | • |
| | LBGT | 10 5(6) 2E | | 4 | Long Branch > Zero | • | • | • | • | • |
| BHI | BHI | 22 | 3 | 2 | Branch Higher | • | • | • | • | • |
| | LBHI | 10 5(6) 22 | | 4 | Long Branch Higher | • | • | • | • | • |
| BHS | BHS | 24 | 3 | 2 | Branch Higher or Same | • | • | • | • | • |
| | LBHS | 10 5(6) 24 | | 4 | Long Branch Higher or Same | • | • | • | • | • |
| BLE | BLE | 2F | 3 | 2 | Branch ≤ Zero | • | • | • | • | • |
| | LBLE | 10 5(6) 2F | | 4 | Long Branch ≤ Zero | • | • | • | • | • |
| BLO | BLO | 25 | 3 | 2 | Branch lower | • | • | • | • | • |
| | LBLO | 10 5(6) 25 | | 4 | Long Branch Lower | • | • | • | • | • |

| Instruction | Forms | Addressing Mode Relative OP | ~ | # | Description | 5 H | 3 N | 2 Z | 1 V | 0 C |
|---|---|---|---|---|---|---|---|---|---|---|
| BLS | BLS | 23 | 3 | 2 | Branch Lower or Same | • | • | • | • | • |
| | LBLS | 10 5(6) 23 | | 4 | Long Branch Lower or Same | • | • | • | • | • |
| BLT | BLT | 2D | 3 | 2 | Branch < Zero | • | • | • | • | • |
| | LBLT | 10 5(6) 2D | | 4 | Long Branch < Zero | • | • | • | • | • |
| BMI | BMI | 2B | 3 | 2 | Branch Minus | • | • | • | • | • |
| | LBMI | 10 5(6) 2B | | 4 | Long Branch Minus | • | • | • | • | • |
| BNE | BNE | 26 | 3 | 2 | Branch Z = 0 | • | • | • | • | • |
| | LBNE | 10 5(6) 26 | | 4 | Long Branch Z = 0 | • | • | • | • | • |
| BPL | BPL | 2A | 2 | 2 | Branch Plus | • | • | • | • | • |
| | LBPL | 10 5(6) 2A | | 4 | Long Branch Plus | • | • | • | • | • |
| BRA | BRA | 20 | 3 | 2 | Branch Always | • | • | • | • | • |
| | LBRA | 16 5 | | 3 | Long Branch Always | • | • | • | • | • |
| BRN | BRN | 21 | 3 | 2 | Branch Never | • | • | • | • | • |
| | LBRN | 10 5 21 | | 4 | Long Branch Never | • | • | • | • | • |
| BSR | BSR | 8D | 7 | 2 | Branch to Subroutine | • | • | • | • | • |
| | LBSR | 17 | 9 | 3 | Long Branch to Subroutine | • | • | • | • | • |
| BVC | BVC | 28 | 3 | 2 | Branch V = 0 | • | • | • | • | • |
| | LBVC | 10 5(6) 28 | | 4 | Long Branch V = 0 | • | • | • | • | • |
| BVS | BVS | 29 | 3 | 2 | Branch V = 1 | • | • | • | • | • |
| | LBVS | 10 5(6) 29 | | 4 | Long Branch V = 1 | • | • | • | • | • |

### SIMPLE BRANCHES

| | OP | ~ | # |
|---|---|---|---|
| BRA | 20 | 3 | 2 |
| LBRA | 16 | 5 | 3 |
| BRN | 21 | 3 | 2 |
| LBRN | 1021 | 5 | 4 |
| BSR | 8D | 7 | 2 |
| LBSR | 17 | 9 | 3 |

### SIMPLE CONDITIONAL BRANCHES (NOTES 1-4)

| Test | True | OP | False | OP |
|---|---|---|---|---|
| N = 1 | BMI | 2B | BPL | 2A |
| Z = 1 | BEQ | 27 | BNE | 26 |
| V = 1 | BVS | 29 | BVC | 28 |
| C = 1 | BCS | 25 | BCC | 24 |

### SIGNED CONDITIONAL BRANCHES (NOTES 1-4)

| Test | True | OP | False | OP |
|---|---|---|---|---|
| r > m | BGT | 2E | BLE | 2F |
| r ≥ m | BGE | 2C | BLT | 2D |
| r = m | BEQ | 27 | BNE | 26 |
| r ≤ m | BLE | 2F | BGT | 2E |
| r < m | BLT | 2D | BGE | 2C |

### UNSIGNED CONDITIONAL BRANCHES (NOTES 1-4)

| Test | True | OP | False | OP |
|---|---|---|---|---|
| r > m | BHI | 22 | BLS | 23 |
| r ≥ m | BHS | 24 | BLO | 25 |
| r = m | BEQ | 27 | BNE | 26 |
| r ≤ m | BLS | 23 | BHI | 22 |
| r < m | BLO | 25 | BHS | 24 |

Notes:

1. All conditional branches have both short and long variations.
2. All short branches are 2 bytes and require 3 cycles.
3. All conditional long branches are formed by prefixing the short branch opcode with $10 and using a 16-bit destination offset.
4. All conditional long branches require 4 bytes and 6 cycles if the branch is taken or 5 cycles if the branch is not taken.