|  |  |  |
| --- | --- | --- |
| **OPCODE** | **OPERATIE** | **DESCRIERE** |
| 00 0000 | HLT |  |
| 00 0001 | ADD | ADD RX, RY |
| 00 0010 | SUB | SUB RX, RY |
| 00 0011 | LSR | LSR RX (RX >> 1) logic |
| 00 0100 | LSL | LSL RX (RX << 1) logic |
| 00 0101 | RSR | RSR RX (RX >> 1) arith |
| 00 0110 | RSL | RSL RX (RX << 1) arith |
| 00 0111 | MOV | MOV RX, RY |
| 00 1000 | MUL | MUL RX, RY |
| 00 1001 | DIV | DIV RX, RY |
| 00 1010 | MOD | MOD RX, RY |
| 00 1011 | AND | AND RX, RX, RY //RX = RX \* RY |
| 00 1100 | OR | OR RX, RX, RY |
| 00 1101 | XOR |  |
| 00 1110 | ● NOT | NOT RX //RX = !RX |
| 00 1111 | ● CMP | CMP RX, RY |
| 01 0000 | ● TST | TST RX, RY |
| 01 0001 | ● INC | INC RX //RX++ |
| 01 0010 | ● DEC | DEC RX //RX-- |
| 01 0011 | ADD | ADD RX, #imd |
| 01 0100 | SUB | SUB RX, #imd |
| 01 0101 | MOV | MOV RX, #imd |
| 01 0110 | MUL | MUL RX, #imd |
| 01 0111 | DIV | DIV RX, #imd |
| 01 1000 | MOD | MOD RX, #imd |
| 01 1001 | AND | AND RX, RX, #imd //RX = RX & #imd |
| 01 1010 | OR |  |
| 01 1011 | XOR |  |
| 01 1100 | CMP | CMP RX, #imd |
| 01 1101 | TST |  |
| 01 1110 | STR | STR RX |
| 01 1111 | LDR | LDR RX |
| 10 0000 | PSH | PSH {RX, RY, LR} |
| 10 0001 | POP | PSH {RX, RY, PC} |
|  |  |  |
|  |  |  |
|  |  |  |
| 10 0010 | BRZ = branch if zero |  |
| 10 0011 | ● BRN = branch if negative |  |
| 10 0100 | ● BRC = branch if carry |  |
| 10 0101 | ● BRO = branch if overflow |  |
| 10 0110 | ● BRA = branch always (unconditional branch) |  |
| 10 0111 | ● JMP = call procedure |  |
| 10 1000 | ● RET = return from procedure |  |