| bits | name | | Big Endian Netw | ork Order |
|----------------|----------------|-------------------|-------------------|------------------|
| 3 | primary op | | An carries | |
| | dest Addr mode | | Dn modulo size, | high no effect |
| 3 | dest reg | | , | |
| | secondary op | quicks and extra | Double(1)/Float(0 | D) Default is D |
| | size | L is default size | . , , | |
| 2 | src Addr mode | 5 | 0 equals 32 | |
| 3 | src reg | quick | branch signed fro | om PC+1 |
| | | | | |
| Addr mode | | | size | |
| #00 | An | Not CFR src | #00 | .W |
| #01 | (An) | | #01 | .L |
| #10 | Dn | Not CFR src | #10 | .Q |
| #11 | Special | | #11 | .D/.F |
| | - | | | |
| Special | | | | |
| #000 | PC | Not float | Not CFR src | |
| #001 | (PC+)/(-SP) | special write | For PCX subrout | ine |
| #010 | IP | Not float | Not CFR src | |
| #011 | (IP+)/(-RP) | special write | For IPX threaded | code |
| #100 | SP | Not float | Not CFR src | |
| #101 | (SP+)/(+SP) | Once (x)(+x) RW | NIP semantics | |
| #110 | RP | Not float | Not CFR src | |
| #111 | (RP+)/(+RP) | Once (x)(+x) RW | NIP semantics | |
| | | | | |
| ops | D | | F | |
| #000 | MOV | No dest read | MVQ | quick src args |
| #001 | ADD | | ADQ | unsigned 5 bit |
| #010 | MLL | Multiply lower | MLQ | for compact |
| #011 | SUB | | SBQ | code |
| #100 | CFR (specials) | No dest read | BPL | quick branch |
| #101 | AND | | BMI | 0 and -1 do sign |
| #110 | ORR | | BZE | of 16 bit |
| #111 | XOR | | BNZ | displacement |
| | | | | |
| D/F ops (size) | D/F | | | |
| #000 | FDV | Divide | | |
| #001 | FAD | | | |
| #010 | FML | | | |
| #011 | FSB | | | |
| #100 | FBP | quick branch | | |
| #101 | FBM | 0 and 1 do sign | | |
| #110 | FBD | of 16 bit | Branch denorma | Izero |

| #111 | FBI | displacement | Branch infinity | |
|--|-----|--------------|-----------------|--|
| | | | | |
| For efficient floats with own register set use vectors | | | | |

| CFR src | | | | | |
|----------------|----------------------|-----------------------|-------------------|-------------------|------|
| An | halves | | | | |
| Dn | fulls | | | | |
| PC | PCX | fast exchange | | | |
| IP | IPX | of registers | | | |
| SP | SPX | improves code | | | |
| RP | RPX | good stuff | | | |
| The src MUST | be an address for | CFR | | | |
| halizaa Am | | | | | |
| halves An | D.W.L. Issue | | _ | | |
| Blocks of 8 | B,W,L = half | | F | | |
| #000 | EXH | Exchange halves | | Double to float | |
| #001 | SXL | Sign extend low | | Float to double | |
| #010 | INU | Increase upper | FIR | Float Inv root | |
| #011 | DCU | Decrease upper | | Float ATN | |
| #100 | SLT | Shift left Q half | FEX | Float EXP | |
| #101 | SRT | Shift right Q half | FLN | Float LN | |
| #110 | ADU | Add upper | FSN | Float SIN | |
| #111 | SBU | Subtract upper | FCS | Float COS | |
| fulls Dn | | | | | |
| Blocks of 8 | W,L,Q | | D | | |
| #000 | ABS | Absolute | DTQ | Double to quad | |
| #001 | SGN | Sign to 0 or -1 | QTD | Quad to double | |
| #010 | NOT | Not | FIR | Double Inv root | |
| #011 | CPL | Complement | FAT | Double ATN | |
| #100 | WTO (group) | Write target over | FEX | Double EXP | |
| #101 | CLR | Clear to zero | FLN | Double LN | |
| #110 | ASR | Arith shift right | FSN | Double SIN | |
| #111 | LSR | Logic shift right | FCS | Double COS | |
| | | | | | |
| | W | L | Q | | |
| WTO group | | | WTO | | |
| WTO is set wri | te target override f | or following instruct | tion for 3 operan | ds | |
| | lowing instruction | | F - 200 | | |
| | | R but not specials e | except WTO mak | es a reserved non | pair |
| | | e assigned other fur | · | | I |
| | | | | | |
| WTO a MOV b | a basamaa a = b/ | o b = romainder for | e evample | DIV | |
| | ,c becomes a - b/ | c, b = remainder for | example | עוט | |

| D (dst\src) | PC vectors | IP | SP | RP | |
|--|---------------------|---------------------|-----------------|------------------|---------|
| PC | BRK | Reserved | Reserved extens | sion opcodes | |
| IP | TRP | system design | 8 * 8 = 64 | | |
| SP | HPR (P) | supervisor (P) | | | |
| RP | CIV (H) | hypervisor (H) | | | |
| F | PC | IP () | SP | RP | |
| PC | COM (0 to 31) | User opcode | Reserved vector | prefix opcodes | |
| IP | compressed | extensions | 8 * 8 = 64 | | |
| SP | pair | eval as nop | | | |
| RP | extensions | if not used (32) | | | |
| | | , | | | |
| register D | PC | IP | SP | RP | |
| is src | | FCN (float | t constant) | | |
| is dest | @BRK | @TRP (P) | @HVR (H) | @CIV (H) | Set d64 |
| register F | PC | IP , | SP | RP | |
| is src | | FCN (float | t constant) | 1 | |
| is dest | | , | OM | | Set d32 |
| | | | | | |
| Blocks of 8 | supervisor (P) | | hypervisor (H) | | |
| #000 | SNP | super nop | HNP | hyper nop | |
| #001 | SUP | set user PT d64 | SSP | set super PT d64 | 1 |
| #010 | EIN | enable interrupt | EVM | enable virt mem | |
| #011 | DIN | disable interrupt | DVM | disable virt mem | |
| #100 | WTI | await interrupt | CLI | clear interrupt | |
| #101 | CLS | clear supervisor | CLH | clear hypervisor | |
| #110 | RSS | reset super | RSH | reset hyper | |
| #111 | SIL | illegal super | HIL | illegal hyper | |
| | | | | | |
| HPR is hyperviso | or trap | | | | |
| BRK is user mod | e breakpoint trap | | | | |
| CIV is set call int | errupt vector (inte | rrupt 0 is page fau | ult) | | |
| | | | | | |
| SUP and SSP fo | llowed by d64 imr | nediate | | | |
| COM instruction | pairs last can be t | tail recursive | | | |
| COM first can no | t be | | | | |
| This makes (@COM n COM m d16) a spare opcode space | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |
| | | | | | |

| (@COM n (COM | /I m d16)) spare c | pcode space | | | |
|---------------------|--|--------------------|--------------------|-------|--|
| #nnn11nn0 011s | ssss opcode form | points to | | | |
| #mmm11mm0 0 | 11000 | | | | |
| #dddddddd dddd | #dddddddd ddddddd | | | | |
| | | | | | |
| Difficult to use di | rectly | | | | |
| Only first of pair, | branches delay s | econd as delay slo | ot. | | |
| Second after bra | nch may or may n | ot speculative exe | ecute and commit | | |
| Allow setting up | and special COM | ? | | | |
| #mmm11mm0 0 | 11000 | | | | |
| #dddddddd dddd | ldddd | is the instruction | replaced by profil | e COM | |
| | | | | | |
| mmm | profile counter nu | umber to increase | | | |
| mm | recall to register | except PC | | | |
| | | | | | |
| If register is PC j | ust increment prof | file counter | | | |
| | | | | | |
| Any immediates | are ordered in the | main instruction | stream | | |
| (BRK/TRP/HPR | /CIV m d16) spare | e opcode space | | | |
| #mmm11mm0 1 | 11000 | | | | |
| #dddddddd dddd | ddddd | | | | |
| | | | | | |
| Except when mn | n is PC nnn is regi | ster special below | 1 | | |
| #mmm | | | | | |
| #000 | Processor serial number | | | | |
| #001 | Clock counter for cycles since power on (timing) | | | | |
| #010 | Clocks stalled since on for efficiency measure | | | | |
| #011 | Clock with boost | cycles since on | | | |
| #100 | Branch miss count since on | | | | |
| #101 | Branch hit count since on | | | | |
| #110 | Data cache miss since on | | | | |
| #111 | Instruction cache | miss since on | | | |
| | | | | | |
| When nn is PC o | clear profile registe | er nnn | | | |
| The register spe | cials are only clea | red via power cyc | le | | |

| NOP stability co | ode watermark s | et | | |
|-------------------|-------------------|--------------------|----------|-----|
| src,dst | PC,PC | PC,other | other,PC | PC |
| MLL | 1 | 31 | 31 | |
| MLQ | | | | 32 |
| halves | | | | 32 |
| fulls | | | | 32 |
| BPL | | | | 32 |
| BMI | | | | 32 |
| BZE | | | | 32 |
| BNZ | | | | 32 |
| AND | 1 | 31 | 31 | |
| ORR | 1 | 31 | 31 | |
| XOR | 1 | 31 | 31 | |
| 476 | 4 | 124 | 124 | 224 |
| All evaluate as n | op unless assigne | ed | | |
| Possible extensi | on opcodes * size | S | 1428 | |
| There are many | other nops. Mainl | y on src,dest choi | ces | |