

```

idle:   if (data_ready=0) goto idle;
        if (load_coeff=1) goto load0;

store:  if (data_ready=0) goto idle;
        reg[5] = data;
        err = 0;
zero:   reg[0] = 0;
sort1:  reg[1] = reg[2];
sort2:  reg[2] = reg[3];
sort3:  reg[3] = reg[4];
sort4:  reg[4] = reg[5];

mul1:   reg[10] = reg[1] * reg[6]; // sample4 * F3
add1:   reg[0] = reg[0] + reg[10]; // add small (+)
        if (V) goto idle;

mul2:   reg[10] = reg[2] * reg[7]; // sample3 * F2
sub2:   reg[0] = reg[0] - reg[10]; // sub large (-)
        if (V) goto idle;

mul3:   reg[10] = reg[3] * reg[8]; // sample2 * F1
add3:   reg[0] = reg[0] + reg[10]; // add large (+)
        if (V) goto idle;

mul4:   reg[10] = reg[4] * reg[9]; // sample1 * F0
sub4:   reg[0] = reg[0] - reg[10]; // sub small (-)
        if (V) goto idle;
        else goto idle;

idle:   err = 1;
        if (data_ready=1) goto store;
        if (data_ready = 0) goto idle;

wait0:  if (lc == 1)
        goto load1;

wait1:  if (lc == 1)
        goto load2;

wait2:  if (lc == 1)
        goto load3;

load0:  reg[9] = F0;
        goto wait0;

load1:  reg[8] = F1;
        goto wait1;

load2:  reg[7] = F2;
        goto wait2;

load3:  reg[6] = F3;
        goto idle;

```