**COMPUTE**

**RESTART**

*count == WIDTH*

*count < width*

*go=0*

*go=1*

The following pseudo-code illustrates the behavior of the bit difference calculator. If there are X more 1s than 0s in the input, then the output is X. If there are Y more 0s than 1s, then the output is –Y.

One possible FSMD (used by the FSMD modules):

**START**

*go=1*

Inputs: go, data (WIDTH bits)

Outputs: result (2\*WIDTH+1 bits), done

Reset values: done = 0, result = 0, diff = 0, data\_r = 0

while(1) {

while (*go* == 0);

done = 0;

*data\_r* = *data*; // Store input in a register.

// This ensures that the code will still

// work if the input changes during the loop.

*diff* = 0;

for *width* iterations {

if data\_r[0] == 1

*diff* ++;

else

*diff* --;

*data* = data >> 1; // Shift right by 1

}

*result* = *diff*;

*done* = 1;

}



