Overall

Master

Nios II (Master)

Custom instruction → Counter Listener

Request for collision counter value

Nios II (Slave)

1. Set base
2. Get coll..

Custom instruction

Interface → Counter

SHA   SHA ....

Test

Dual core needed because single core blocked by collision all

This allows master to pull the result of computations

Original hardware design with custom instructions had no way to communicate the total SHA digest computed
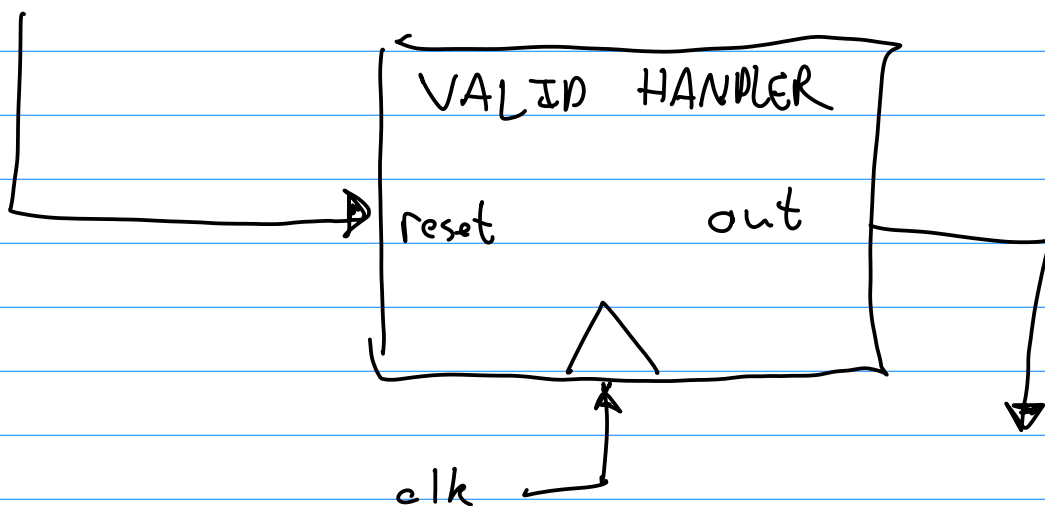
START

0 — 15 possible

Data Index

reset   out   4

clk
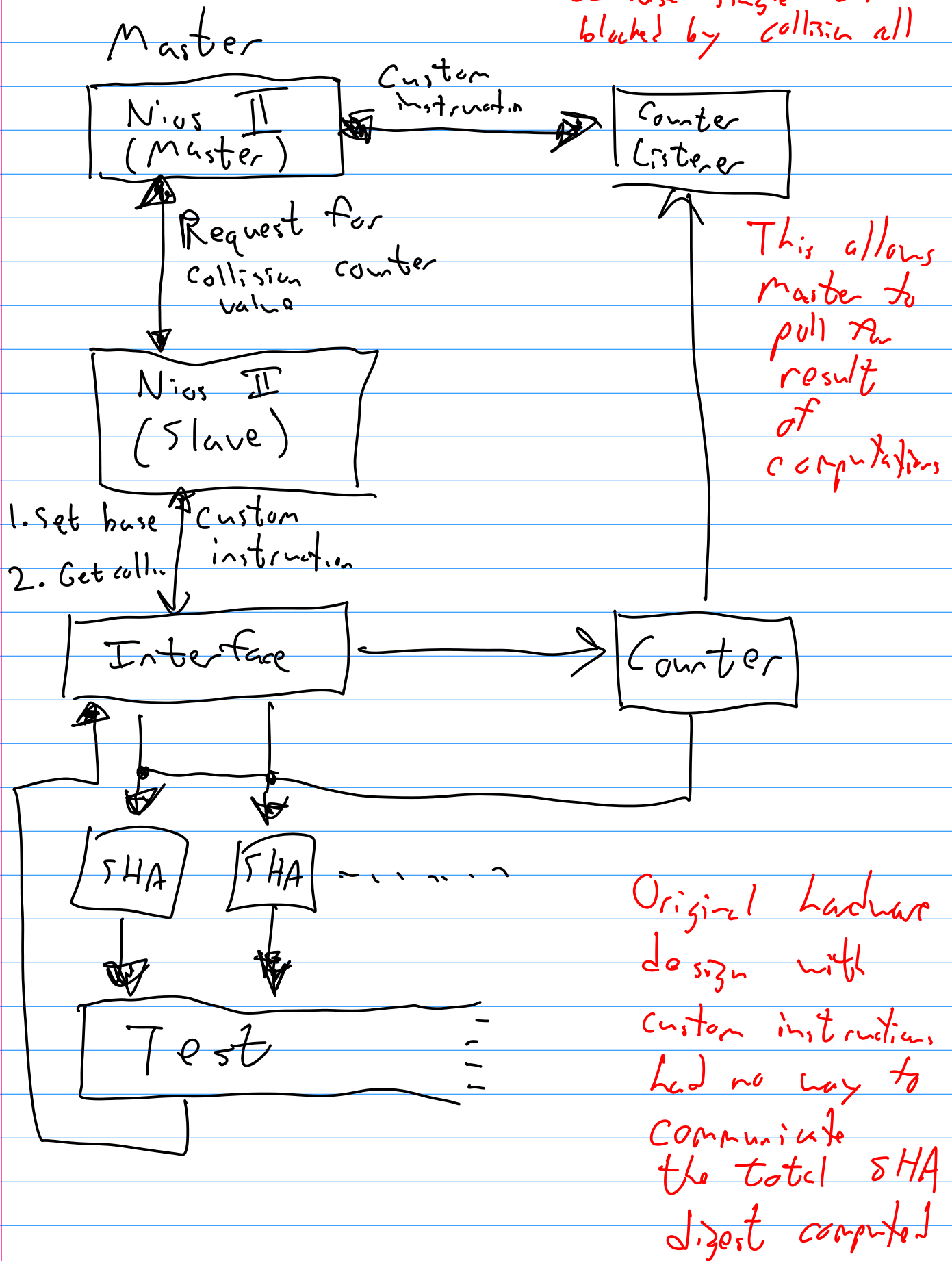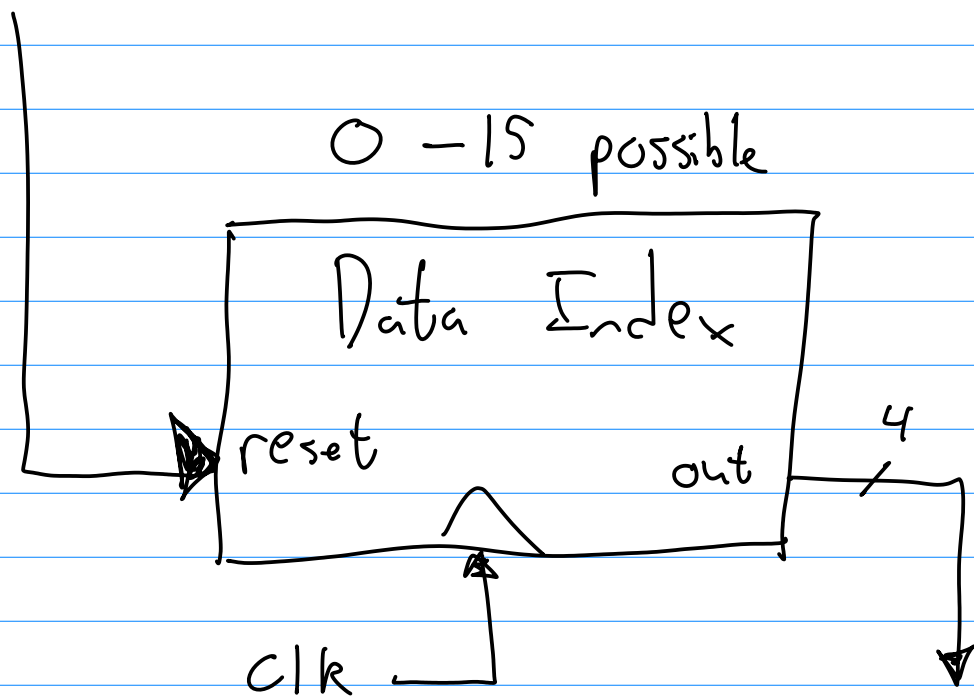
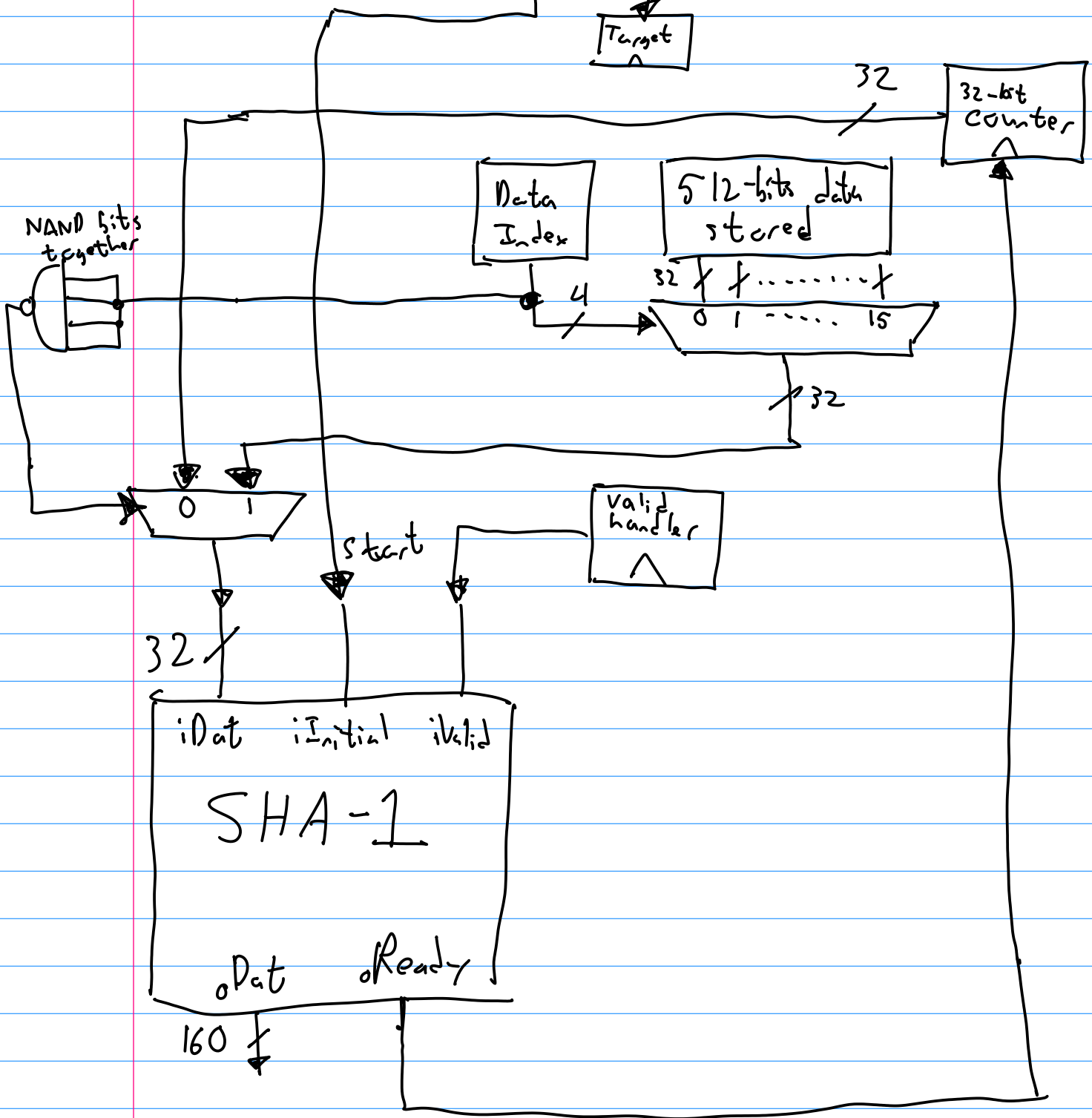increments every clock pulse
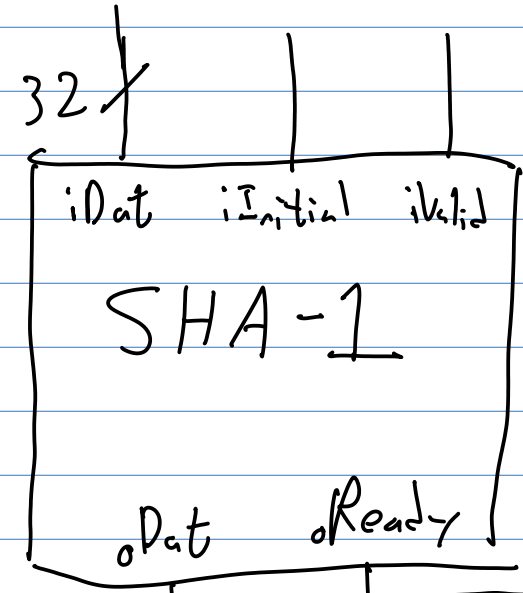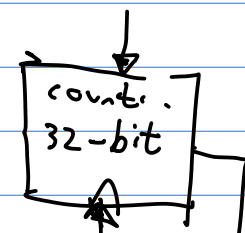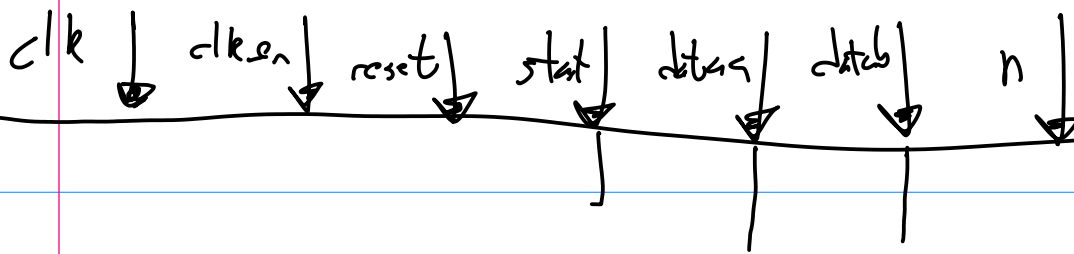
START

VALID HANDLER

reset   out

clk

- Counts down
- Outputs "1" while internally > 1
- Does not reset from 0 normally, from overflow

Reset sends internal counter back to 15

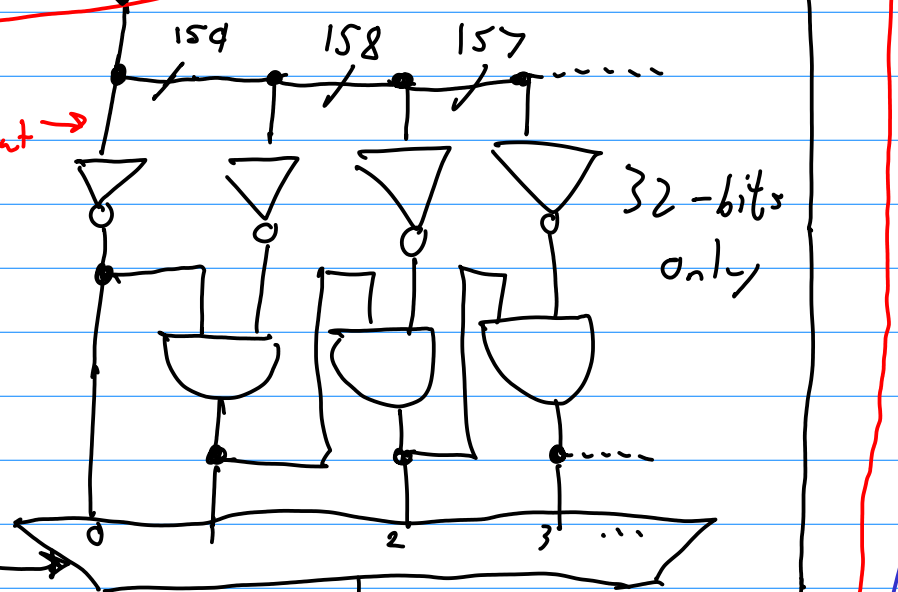clk   clken   reset   start   data_a   data_b   n

Target

32

32-bit counter

Data Index

512-bits data stored

32 ......
0 1 15

NAND bits together

32

0   1

Valid handler

start

32

iDat   iInitial   iValid

SHA-1

oDat   oReady

160

clk   clken   reset   start   dataa   datab   n

Target

counter 32-bit

~1

SHA-1

iDat   iInitial   iValid

32

oDat   oReady

160

32-bit Collision checker

most significant bit

154   158   157   . . . . . .

32-bits only

Counter Reduce

Needed because counter is incremented above

~1

(counter-1)

0   2   3   . . .
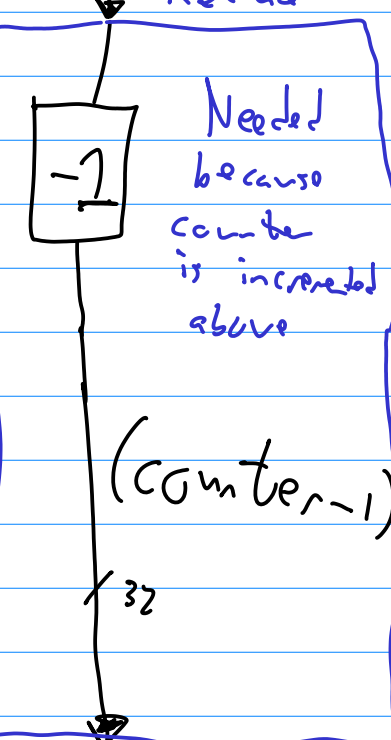
if "1" collision found

data clk

32

32

done

result

State Machine

start=0

IDLE

start=1

DETERMINE INSTRUCTION

n=0

n=1

APPEND BYTES TO MESSAGE

SET TARGET FROM DATAA

LOAD MESSAGE INTO SHA-1 (INCREMENT DATA INDEX)

loading=1

loading=0

ready=0

WAIT FOR READY FROM SHA-1

Collision=0

Collision=1

DONE

CHECK IF COLLISION