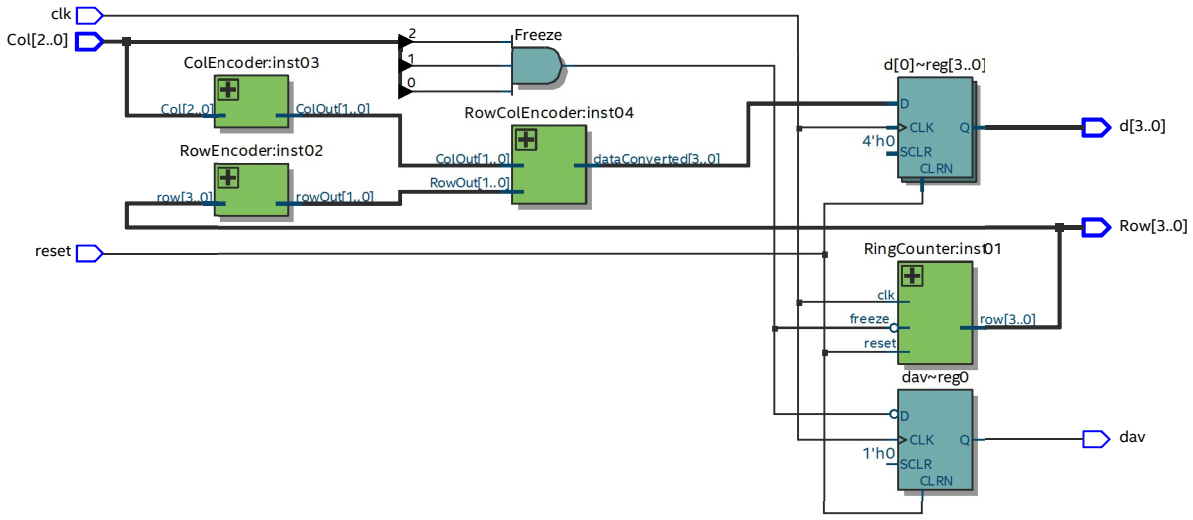
```
1   module keypadEncoder (
2       input wire clk, reset,
3       input wire[2:0] Col,
4
5       output wire[3:0] Row,
6       output reg[3:0] d,
7       output reg dav
8
9   );
10
11      wire Freeze;
12      wire[1:0] RowOut, ColOut;
13      wire[3:0] dataConverted;
14
15      RingCounter inst01 (clk, reset, Freeze, Row);
16      RowEncoder inst02 (Row, RowOut);
17      ColEncoder inst03 (Col, ColOut);
18      RowColEncoder inst04 (RowOut, ColOut, dataConverted);
19
20      assign Freeze = ~(Col[0] & Col[1] & Col[2]);
21
22      always @(posedge clk or posedge reset) begin
23          if(reset) begin
24              d = 4'b0000;
25              dav = 1'b0;
26          end
27          else begin
28              d = dataConverted;
29              dav = Freeze;
30          end
31      end
32  endmodule
```


clk
Col[2..0]
ColEncoder:inst03
Col[2..0]
ColOut[1..0]
Freeze
2
1
0
RowColEncoder:inst04
ColOut[1..0]
RowOut[1..0]
dataConverted[3..0]
RowEncoder:inst02
row[3..0]
rowOut[1..0]
reset
d[0]~reg[3..0]
D
CLK
Q
4'h0
SCLR
CLRN
d[3..0]
Row[3..0]
RingCounter:inst01
clk
freeze
reset
row[3..0]
dav~reg0
D
CLK
Q
1'h0
SCLR
CLRN
dav