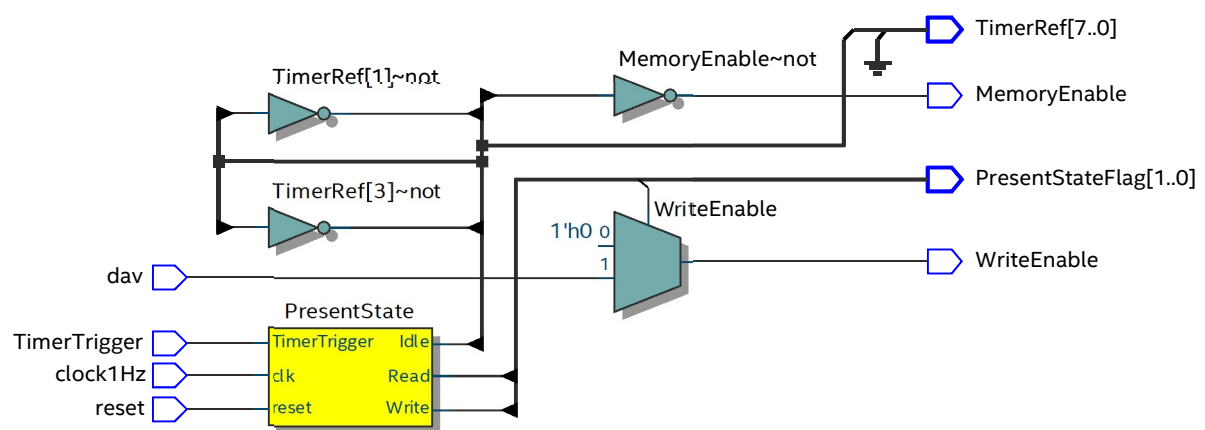


```
1  module ControlUnit (
2      input wire clock1Hz, reset,
3
4      input wire dav,
5      output reg WriteEnable, MemoryEnable,
6
7      input wire TimerTrigger,
8      output reg[7:0] TimerRef,
9
10     output wire[1:0] PresentStateFlag
11 );
12
13 assign PresentStateFlag = PresentState;
14
15 parameter
16     Idle = 2'd0,
17     Write = 2'd1,
18     Read = 2'd2;
19 reg[1:0] PresentState, NextState;
20
21 //Bloco Sequencial
22 always @(posedge clock1Hz or posedge reset) begin
23     if (reset) begin
24         PresentState <= Idle;
25     end
26     else begin
27         PresentState <= NextState;
28     end
29 end
30
31 //Transições
32 always @(*) begin
33     case(PresentState)
34     default: begin
35     end
36     Idle: begin
37         if(TimerTrigger) NextState <= Write;
38         else NextState <= Idle;
39     end
40     Write: begin
41         if(TimerTrigger) NextState <= Read;
42         else NextState <= Write;
43     end
44     Read: begin
45         if(TimerTrigger) NextState <= Idle;
46         else NextState <= Read;
47     end
48 endcase
49 end
50
51 //Bloco Combinacional
52 always @(*) begin
53     case(PresentState)
54     default: begin
55     end
56     Idle: begin
57         TimerRef = 8'd5;
58         WriteEnable = 1'd0;
59         MemoryEnable = 1'd0;
60     end
61     Write: begin
62         TimerRef = 8'd10;
63         WriteEnable = dav;
64         MemoryEnable = 1'd1;
65     end
66     Read: begin
67         TimerRef = 8'd10;
68         WriteEnable = 1'd0;
69         MemoryEnable = 1'd1;
70     end
71 endcase
72 end
73
74 endmodule
75
```



```
1  module sistema (
2      input wire clock50MHz, clock1Hz, reset,
3
4      input wire dav,
5      input wire[3:0] KeypadData,
6
7      input wire[3:0] AddressIn,
8
9      output reg[3:0] DataOut,
10
11     output wire[1:0] PresentStateFlag
12 );
13
14     wire WriteEnable, MemoryEnable, TimerTrigger;
15     wire[7:0] TimerRef;
16     ControlUnit inst00(clock1Hz, reset, dav, WriteEnable, MemoryEnable, TimerTrigger,
17     TimerRef, PresentStateFlag);
18
19     Timer inst01(TimerRef, clock1Hz, reset, TimerTrigger);
20
21     wire MemoryClock;
22     assign MemoryClock = (MemoryEnable == 1'b1) ? clock50MHz : 1'bz;
23     wire[3:0] RamData;
24     RamMemory inst02(MemoryClock, WriteEnable, AddressIn, KeypadData, RamData);
25
26     always @(*) begin
27         case (PresentStateFlag)
28             2'd0: DataOut = DataOut;
29             2'd1: DataOut = DataOut;
30             2'd2: DataOut = RamData;
31             default: begin
32
33                 end
34             endcase
35
36     end
37
38 endmodule
```

