```
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
61              end
62              Write: begin
63                  TimerRef = 8'd10;
64                  WriteEnable = dav;
65                  MemoryEnable = 1'd1;
66              end
67              Read: begin
68                  TimerRef = 8'd10;
69                  WriteEnable = 1'd0;
70                  MemoryEnable = 1'd1;
71              end
72          endcase
73      end
74
75  endmodule
```