```
1   //Aaron Hong (ahong02)
2   //Stephen Macris (smacris)
3   //5/3/23
4   //EE469 Lab3
5
6   //This module creates an asynchronous hazard unit that outputs stalling, flushing, and
    forwarding control signals.
7
8   module hazard_unit (
9      input logic Match_1E_M, Match_2E_M,
10     input logic Match_1E_W, Match_2E_W,
11     input logic Match_12D_E,
12     input logic RegWriteM, RegWriteW, MemtoRegE, BranchTakenE, PCWrPendingF, PCSrcW,
13     output logic [1:0] ForwardAE, ForwardBE,
14     output logic StallF, StallD, FlushD, FlushE
15     );
16
17     logic ldrStallD;
18
19     assign ldrStallD = Match_12D_E & MemtoRegE;
20
21     assign StallF = ldrStallD + PCWrPendingF;
22     assign FlushD = PCWrPendingF + PCSrcW + BranchTakenE;
23     assign FlushE = ldrStallD + BranchTakenE;
24     assign StallD = ldrStallD;
25
26     always_comb begin
27        if      (Match_1E_M && RegWriteM) ForwardAE = 2'b10;
28        else if (Match_1E_W && RegWriteW) ForwardAE = 2'b01;
29        else                              ForwardAE = 2'b00;
30
31        if      (Match_2E_M && RegWriteM) ForwardBE = 2'b10;
32        else if (Match_2E_W && RegWriteW) ForwardBE = 2'b01;
33        else                              ForwardBE = 2'b00;
34     end
35
36  endmodule
37
38
```