```
1
2   module ATM_FSM(clk,card_insert,reset,pin_1,pin_2,pin_3,menu_sel,
3                  amount,acc_num,select_options,face,otp,cash_dis,
4                  bal_D,print_receipt,CNL,card_lock,mini_s
5                  );
6
7     input wire clk, card_insert, reset;
8     input wire[3:0] pin_1;
9     input wire[3:0] pin_2;
10    input wire[3:0] pin_3;
11    input wire[1:0] menu_sel;    //menu selection
12    input wire[15:0] amount;     //withdrawal or deposit amount
13    input wire[3:0] face;        //face recoginition
14    input wire CNL;              //cancel
15    input wire select_options;
16    input wire[7:0]acc_num;      //account number for pin generation
17    input wire [7:0] otp;
18    output reg cash_dis, bal_D, print_receipt,card_lock,mini_s;         //cash dispending
19                                                                       //balance display
20                                                                       //print_receipt
      (ack) for deposit
21                                                                       //mini-statement
22                                                                       //card lock for 24
      hours
23
24    integer i,counter;
25    integer t_count;
26
27  //helper signals
28
29    reg [1:0]pin_valid[1:3];
30    reg acc_valid;
31    reg otp_valid,face_valid;
32
33  // States definition
34
35  parameter s0 = 4'b0000;
36  parameter s1 = 4'b0001;
37  parameter s2 = 4'b0010;
38  parameter s3 = 4'b0011;
39  parameter s4 = 4'b0100;
40  parameter s5 = 4'b0101;
41  parameter s6 = 4'b0110;
42  parameter s7 = 4'b0111;
43  parameter s8 = 4'b1000;
44  parameter s9 = 4'b1001;
45  parameter s10 = 4'b1010;
46  parameter s11 = 4'b1011;
47  parameter s12 = 4'b1100;
48  parameter s13 = 4'b1101;
49  parameter s14 = 4'b1110;
50  parameter s15 = 4'b1111;
51
52
53  reg [3:0] currState, nextState;
54  reg [3:0] acc_index,acc_index_Mpin;
55  reg EC;                              //EC:eject card
56  reg [31:0] balance_db [0:9];
57  reg [7:0] otp_db [0:9];
58  reg [3:0] pin_db [0:9];
59  reg [3:0] face_db [0:9];
60  reg [7:0] acc_db [0:9];
61  reg [15:0] transaction [0:4];
62  reg [15:0] balance [0:4];
63
64
65  initial begin
66  i=0;
67  counter=0;
68  t_count=0;
69  pin_valid[1]=2'b00;
70  pin_valid[2]=2'b00;
71  pin_valid[3]=2'b00;
72
73  // customer accounts, balance and other confidential details initialization
74
75                 acc_db[0] = 8'd1234;
```

```
                acc_db[1] = 8'd2345;
                acc_db[2] = 8'd3456;
                acc_db[3] = 8'd4567;
                acc_db[4] = 8'd5678;
                acc_db[5] = 8'd6789;
                acc_db[6] = 8'd7890;
                acc_db[7] = 8'd8901;
                acc_db[8] = 8'd9012;
                acc_db[9] = 8'd1230;

                balance_db[0] = 31'd50000;
                balance_db[1] = 31'd500;
                balance_db[2] = 31'd500;
                balance_db[3] = 31'd500;
                balance_db[4] = 31'd500;
                balance_db[5] = 31'd500;
                balance_db[6] = 31'd50000;
                balance_db[7] = 31'd500;
                balance_db[8] = 31'd500;
                balance_db[9] = 31'd500;

                pin_db[0] = 4'b1111;
                pin_db[1] = 4'b0001;
                pin_db[2] = 4'b0010;
                pin_db[3] = 4'b0011;
                pin_db[4] = 4'b0100;
                pin_db[5] = 4'b0101;
                pin_db[6] = 4'b0110;
                pin_db[7] = 4'b0111;
                pin_db[8] = 4'b1000;
                pin_db[9] = 4'b1001;

                //storing face data in digital format

                face_db[0] = 4'b1111;
                face_db[1] = 4'b0001;
                face_db[2] = 4'b0010;
                face_db[3] = 4'b0011;
                face_db[4] = 4'b0100;
                face_db[5] = 4'b0101;
                face_db[6] = 4'b0110;
                face_db[7] = 4'b0111;
                face_db[8] = 4'b1000;
                face_db[9] = 4'b1001;

                otp_db[0] = 8'd2749;
                otp_db[1] = 8'd2175;
                otp_db[2] = 8'd2429;
                otp_db[3] = 8'd2125;
                otp_db[4] = 8'd2178;
                otp_db[5] = 8'd2647;
                otp_db[6] = 8'd2816;
                otp_db[7] = 8'd2910;
                otp_db[8] = 8'd2299;
                otp_db[9] = 8'd2689;

    end


    //sequential logic block for state transistion

    always @ (posedge clk)
    begin
          if(reset == 1'b1)
             currState <= #1 s0;
          else
             currState <= #1 nextState;
    end

    //combinational block for next state logic

    always @ (*)
    begin

       nextState = currState;

       case(currState)
```

```
153
154       s0:if(card_insert & !card_lock & !EC)
155           nextState = s8;
156           else
157           nextState = s0;
158
159       s1:if(otp_valid)
160          nextState = s15;
161          else
162          nextState = s0;
163
164       s2:
165          case(menu_sel)
166          2'b00 : nextState = s6;
167          2'b01 : nextState = s3;
168          2'b10 : nextState = s14;
169          default:nextState = s2;
170          endcase
171
172       s3:nextState = s0;
173
174       s4:if(pin_valid[1] == 2'b11)
175          nextState =  s2 ;
176          else if(pin_valid[1] == 2'b01)
177          nextState = s9;
178
179       s5:if(otp_valid)
180          nextState = s12;
181          else
182          nextState = s0;
183
184       s6:if(!CNL)
185          nextState = s7;
186          else if(CNL)
187          nextState = s0;
188          else
189          nextState = s6;
190
191       s7:if( amount > balance_db[acc_index])
192          nextState = s6;
193          else if(amount > 16'd10000 && amount <= 16'd25000 && amount < balance_db[acc_index])
194          nextState = s5;
195          else if(amount > 16'd25000 && amount < balance_db[acc_index])
196          nextState = s11;
197          else if(amount <= 16'd10000 && amount < balance_db[acc_index])
198          nextState = s12;
199
200       s8:case(select_options)
201          1'b0 : nextState = s4;  //banking
202          1'b1 : nextState = s13; //pin generation
203          endcase
204
205       s9:if(pin_valid[2] == 2'b11)
206          nextState = s2;
207          else if(pin_valid[2] == 2'b01)
208          nextState = s10;
209
210      s10:if(pin_valid[3] == 2'b11)
211          nextState = s2;
212          else if(pin_valid[3] == 2'b01)
213          nextState = s0;
214
215       s11:if(face_valid)
216           nextState = s12;
217           else
218           nextState = s0;
219
220       s12:if(EC)
221           nextState = s0;
222           else
223           nextState = s12;
224
225       s13:if(acc_valid)
226           nextState = s1;
227           else
228           nextState = s0;
229
```

```
230       s14:if(EC)
231           nextState = s0;
232           else
233           nextState = s14;
234
235       s15:if(EC)
236           nextState = s0;
237           else
238           nextState = s15;
239
240       endcase
241    end
242
243    //sequential logic for outputs
244
245    always @ (posedge clk)
246    begin
247       if(reset == 1'b1)
248          begin
249           print_receipt <= 1'b0;
250           cash_dis <= 1'b0;
251           bal_D <= 1'b0;
252           EC <=0;
253           mini_s <=1'b0;
254          end
255    else
256       card_lock <= 1'b0;
257
258       case(currState)
259
260       s0:begin
261           print_receipt <= 1'b0;
262           cash_dis <= 1'b0;
263           bal_D <= 1'b0;
264            mini_s <=1'b0;
265           end
266
267       s1:if(!otp_valid)
268            EC <= 1'b1;
269
270       s2:EC <= 1'b0;
271
272       s3:begin
273            EC <= 1'b1;
274            mini_s <= 1'b1;
275          end
276
277       s7:begin
278           print_receipt <= 1'b0;
279           cash_dis <= 1'b0;
280           bal_D <= 1'b0;
281           card_lock <= 1'b0;
282          end
283
284       s14:begin
285           print_receipt <= !amount ? 1'b0 : 1'b1;
286           EC <= 1'b1;
287           end
288
289       s12:begin
290            cash_dis <= 1'b1;
291            #20 bal_D <= 1'b1;
292            EC <= 1'b1;
293           end
294
295       s13:if(!acc_valid)
296            EC <= 1'b1;
297

298       s10:begin
299             if(pin_valid[3] == 2'd01)
300              EC <= 1'b1;
301              for(i=0;i<24;i=i+1)
302              #10 card_lock <= 1'b1;
303           end
304
305       s15: EC <= 1'b1;
```

```
306
307    default:begin
308             print_receipt <= 1'b0;
309             cash_dis <= 1'b0;
310             bal_D <= 1'b0;
311             card_lock <= 1'b0;
312             EC <=0;
313             mini_s <=1'b0;
314            end
315
316    endcase
317    end
318
319    //combinational block for performing operations
320
321    always @(currState)
322    begin
323
324       case(currState)
325
326       s0:$display("insert card");
327
328       s1:begin
329          $display("enter the otp sent to registerd mobile number");
330          otp_valid = ( otp == otp_db[acc_index_Mpin]) ? 1'b1 : 1'b0;
331          if(!otp_valid)
332             $display("Invalid otp");
333          end
334
335       s2:begin
336          $display("enter 00 for withdrawal");
337          $display("enter 01 for  mini_statement");
338          $display("enter 10 to deposit money");
339       end
340
341       s3:begin
342            for ( i = 0; i<5 ; i=i+1)
343            $display(" %d . amount_transacted = %d      main_balance = %d",i,transaction[i],
    balance[i]);
344            $display("balance is %d",balance_db[acc_index]);
345          end
346
347       s4:begin
348            $display("enter the 4 digit pin number");
349             for (i=0;i<9 && pin_1!=pin_db[i];i=i+1)
350              if(i==9) pin_valid[1] = 2'b01;
351
352             if(i<9)
353                begin
354                acc_index = i;
355                pin_valid[1] = 2'b11;
356                end
357             else
358                pin_valid[1] = 2'b01;
359          end
360
361       s5:begin
362          $display("enter the otp sent to mobile number");
363          otp_valid = (otp == otp_db[acc_index]) ? 1'b1 : 1'b0;
364          end
365
366       s6:$display("if u wish to cancel press CNl or else continue");
367
368       s8:begin
369           $display("press '0' for banking");
370           $display("press '1' for pin generation");
371           end
372
373       s9:begin
374         $display("incorrect pin_1");
375         $display("enter pin again");
376         for (i=0;i<9 && pin_2!=pin_db[i];i=i+1)
377           if(i==9)  pin_valid[2] = 2'b01;
378
379         if(i<9) begin
380            acc_index = i;
381            pin_valid[2] = 2'b11;
```

```
382         end
383         else
384            pin_valid[2] = 2'b01;
385         end
386
387     s10:begin
388         $display("incorrect pin_2");
389         $display("enter pin again");
390         for (i=0;i<9 && pin_3!=pin_db[i];i=i+1)
391           if(i==9) pin_valid[3] = 2'b01;
392
393         if(i<9) begin
394            acc_index = i;
395            pin_valid[3] = 2'b11;
396            end
397         else
398            begin
399            $display("card has been locked for 24 hours");
400            pin_valid[3] = 2'b01;
401            end
402         end
403
404     s11:begin
405         $display("recognizing face");
406         face_valid = (face == face_db[acc_index]) ? 1'b1 : 1'b0;
407         if(!face_valid)
408            $display("face does not match");
409         end
410
411     s12:begin
412         $display("old_balance is %d",balance_db[acc_index]);
413         balance_db[acc_index] = balance_db[acc_index] - amount;
414         $display("new_balance is %d",balance_db[acc_index]);
415
416         //storing transaction details
417         transaction[t_count] = amount;
418         balance[t_count] = balance_db[acc_index];
419         t_count = (t_count+1) % 5;
420         end
421
422     s13:begin
423         $display("enter the Account number");
424         for (i=0;i<9 && acc_num!=acc_db[i];i=i+1)
425           if(i==9) acc_valid <= 1'b0;
426
427         if(i<9) begin
428           acc_index_Mpin = i;
429           acc_valid = 1'b1;
430         end
431         else
432           acc_valid = 1'b0;
433         if(!acc_valid)
434           $display("the entered account number does not exist");
435         end
436
437     s14:begin
438            $display("place the amount");
439            $display("transaction completed");
440            balance_db[acc_index] = balance_db[acc_index] + amount;
441
442            //soring transaction details
443            transaction[t_count] = amount;
444            balance[t_count] = balance_db[acc_index];
445            t_count = (t_count+1) % 5;
446         end
447
448     s15:begin
449           $display("enter new Mpin");
450           pin_db[acc_index_Mpin] = pin_1;
451           $display("pin updated successfully");
452         end
453
454     endcase
455     end
456
457     endmodule
458
```

```
459  //Test bench
460  module ATM_SM_tb;
461
462    reg clk;
463    reg card_insert, reset;
464    reg [3:0] pin_1;
465    reg [3:0] pin_2;
466    reg [3:0] pin_3;
467    reg [2:0] menu_sel;
468    reg [15:0] amount;
469    reg [7:0] acc_num;
470    reg select_options;
471    reg [3:0] face;
472    reg [7:0] otp;
473    reg CNL;
474    wire cash_dis,bal_D,print_receipt;
475    wire card_lock;
476    wire mini_s;
477
478    // Instantiate the ATM_SM module
479    ATM_FSM atm_sm (
480      .clk(clk),
481      .card_insert(card_insert),
482      .reset(reset),
483      .pin_1(pin_1),
484      .pin_2(pin_2),
485      .pin_3(pin_3),
486      .menu_sel(menu_sel),
487      .amount(amount),
488      .acc_num(acc_num),
489      .select_options(select_options),
490      .face(face),
491      .otp(otp),
492      .cash_dis(cash_dis),
493      .bal_D(bal_D),
494      .print_receipt(print_receipt),
495      .CNL(CNL),
496      .card_lock(card_lock),
497      .mini_s(mini_s)
498    );
499
500  initial begin
501  clk = 0;
502  reset = 1;
503
504  //test case 1                          card_lock after 3 pin attempts failed
505  //=================================================================================================
     ==========
506
507
508  #10   CNL=0;reset = 0 ;card_insert = 1;
509  #10 pin_1 = 4'b1110;
510  #10 pin_2 = 4'b1110;
511  #10 pin_3 = 4'b1110;
512   select_options = 0;
513   menu_sel = 2'b00;
514   amount = 16'd33000;
515   face = 4'b1111;
516
517
518
519  //test case 2                            deposit
520  //=================================================================================================
     ==========
521  #400 reset=1;
522  #10 CNL=0;reset = 0 ;
523   card_insert = 1;
524   select_options = 0;
525   pin_1 = 4'b1111;
526   menu_sel = 2'b10;
527   amount = 16'd2000;
528
529  //test case 3                         pin generation / pin change
530  //=================================================================================================
     ==========
531
532  #80 reset=1;
```

```
533   #10   CNL=0;reset = 0 ;
534    card_insert = 1;
535    select_options = 1;
536    acc_num = 8'd1234;
537    otp = 8'd2749;
538    pin_1 = 4'b1010;
539
540   // verifying whether the pin is updated
541
542   #200 reset=1;
543   #10   CNL=0;reset = 0 ;
544    card_insert = 1;
545    select_options = 0;
546    pin_1 = 4'b1010;
547    menu_sel = 2'b01;
548
549   //test case 4                          withdrawal
550   //=============================================================================================
      ==========
551   #80   reset =1 ;
552   #10 CNL=0;reset = 0;
553    card_insert = 1;
554    pin_1 = 4'b1010;
555    select_options = 0;
556    menu_sel = 2'b00;
557    amount = 16'd33000;
558    face = 4'b1111;
559
560   #150 reset =1;
561   #10 CNL=0;reset = 0;
562    card_insert = 1;
563    pin_1 = 4'b1010;
564    select_options = 0;
565    menu_sel = 2'b00;
566    amount = 16'd12000;
567    otp = 8'd2749;
568
569   #150 reset =1;
570   #10 CNL=0;reset = 0;
571    card_insert = 1;
572    pin_1 = 4'b1010;
573    select_options = 0;
574    menu_sel = 2'b00;
575    amount = 16'd5000;
576
577
578   //test case 5                          printing recent 5 transactions
579   //=============================================================================================
      ==========
580
581   #150 reset=1;
582   #10   CNL=0;reset = 0 ;
583    card_insert = 1;
584    select_options = 0;
585    pin_1 = 4'b1010;
586    menu_sel = 2'b01;
587
588
589   #500 $finish;
590     end
591
592     always #5 clk = ~clk;
593
594   endmodule
```