代码实现：

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 2025/12/01 23:58:38
// Design Name:
// Module Name: vending_machine
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////

module vending_machine (
    input wire clk,
    input wire rst,
    input wire op_start,
    input wire [1:0] coin_val,
    input wire cancel_flag,

    output reg hold_ind,
    output reg drinktk_ind,
    output reg charge_ind,
    output reg [2:0] charge_val
);

    localparam S0 = 3'd0;
    localparam S1 = 3'd1;
    localparam S2 = 3'd2;
    localparam S3 = 3'd3;
    localparam S4 = 3'd4;
    localparam S5 = 3'd5;
    localparam S6 = 3'd6;

    reg [2:0] current_state, next_state;
```

```
always @(posedge clk or posedge rst) begin
    if (rst)
        current_state <= S0;
    else
        current_state <= next_state;
end

always @(*) begin
    next_state = current_state;

    case (current_state)
        S0: begin
            if (op_start) begin
                if (coin_val == 2'b01) next_state = S1;
                else if (coin_val == 2'b10) next_state = S2;
            end
        end

        S1: begin
            if (cancel_flag) next_state = S0;
            else if (coin_val == 2'b01) next_state = S2;
            else if (coin_val == 2'b10) next_state = S3;
        end

        S2: begin
            if (cancel_flag) next_state = S0;
            else if (coin_val == 2'b01) next_state = S3;
            else if (coin_val == 2'b10) next_state = S4;
        end

        S3: begin
            if (cancel_flag) next_state = S0;
            else if (coin_val == 2'b01) next_state = S4;
            else if (coin_val == 2'b10) next_state = S5;
        end

        S4: begin
            if (cancel_flag) next_state = S0;
            else if (coin_val == 2'b01) next_state = S5;
            else if (coin_val == 2'b10) next_state = S6;
        end

        S5: begin
```

```
            next_state = S0;
        end

        S6: begin
            next_state = S0;
        end

        default: next_state = S0;
    endcase
end

always @(posedge clk or posedge rst) begin
    if (rst) begin
        hold_ind <= 1'b0;
        drinktk_ind <= 1'b0;
        charge_ind <= 1'b0;
        charge_val <= 3'b000;
    end else begin
        drinktk_ind <= 1'b0;
        charge_ind <= 1'b0;
        charge_val <= 3'b000;

        if (next_state == S0)
            hold_ind <= 1'b0;
        else
            hold_ind <= 1'b1;

        case (current_state)
            S1, S2, S3, S4: begin
                if (cancel_flag) begin
                    charge_ind <= 1'b1;
                    case (current_state)
                        S1: charge_val <= 3'b001;
                        S2: charge_val <= 3'b010;
                        S3: charge_val <= 3'b011;
                        S4: charge_val <= 3'b100;
                    endcase
                end
            end

            S5: begin
                drinktk_ind <= 1'b1;
                charge_ind <= 1'b0;
            end
```

```
                S6: begin
                    drinktk_ind <= 1'b1;
                    charge_ind <= 1'b1;
                    charge_val <= 3'b001;
                end
            endcase
        end
    end

endmodule
```

仿真图形：