# Intilligent Home Automation  Using FPGA and Sensors

## TEAM  SNAPDRAGON

**Abstract:** This project demonstrate an Intilligent Home Automation System implemented using FPGA -based platform and various sensors.The main objective is to simulate a smart room environment where electrical appliances (AC,Fan,Heateretc..,) automatically respond based on real-time temperature and motion data.The system designed,developed ,and tested using a DE10-Nano FPGA development board,various analog and digital sensors and basic electronic components.

**Team Members:**

EDIGA VARUNSAI -SSIT0005

PODALAKURU PRASAD-SSIT0232

KOTHAPALLI MARUTHI -  SSIT0006

## Components Used

1. FPGA Board: DE10-Nano
2. Temperature Sensor:Analog(e.g.,,DHT11)
3. PIR Sensor:Detects human motion
4. Breadboard :for prototyping connection
5. DC Motors(2):
    - One motor simulates a fan
    - Another motor simulates an AC
6. LED-its simulates a heater
7. Jumper Wires:For making electrical connections
8. Candle:Used as a heat source to increase temperature for testing

**Working:**

After the human detection by using PIR sensor then the temperature in the room will be read by the temperature analog DHT11 sensor based on this condition LED and two motors will be worked through the relay module.

Let me conclude the detailed process:
When the PIR sensor detects the human presence in the room then the temperature analog sensor will activate ,
Case 1: If the temperature exceeds the threshold temperature (which can be given by the user) then the room gets more heat so to control the temperartue in the room we use two DC motors which acts as fan and AC which can decrease the temperature in the room so that the room becomes cool.
Here we increase the temperature by placing a candle near to the temperature sensor.

Case 2: If the temperature sensor reads the temperature less than the threshold temperature then the led will be turns on (which acts as heater to increase the temperature in the room).

**Verilog Code**

```
// module declearation
module relay(
        input clk1,
        input adc_out,
        input rst,
        output sclk,
        output reg cs,
        output reg [6:0] led_out,
        output reg clk,
        output reg ac_relay,
        output reg fan_relay,
```

```
        output reg heater_led,
         input pir
);
//internal registers
reg [11:0] adc_out_store;
reg [3:0] count_out;
reg [3:0] count;
reg [31:0] count1;
reg [6:0] temperature;
always @ (posedge clk1 or posedge rst) begin
        if (rst) begin
                count1 <= 0;
                clk <= 0;
                end
        else begin
                if (count1 == 25000000) begin
                        clk <= ~clk;

                end

                else begin
                        count1 <= count1 + 1;
                end
        end
end

//intialization
initial begin
        count = 4'd0;
        cs = 1;
        count_out = 4'd0;
end
// Chip Select
always @(negedge clk1) begin
         if (count == 1) begin
                cs <= 0;
        end
end
//serial clock for ADC
assign sclk = cs ? 1 : clk;
```

```
//Count increament
always @(posedge clk1) begin
        count <= count + 4'd1;
end
//main logic
always @(posedge clk or posedge rst) begin

    if (rst) begin

        count_out <= 0;
        led_out <= 7'b0000000;
        adc_out_store <= 12'd0;
        temperature <= 7'd0;
        ac_relay <= 0;
        fan_relay <= 0;
        heater_led <= 0;
end
else begin
        if (count_out < 12) begin
                adc_out_store[11 - count_out] <= adc_out;
               count_out <= count_out + 1;

        end

        else begin
                led_out <= adc_out_store[6:0];

   temperature<= adc_out_store[6:0];

        if (adc_out_store < 7'b0101000 && pir ==1) begin
                ac_relay <= 0;
               fan_relay <= 0;
               heater_led <= 0;

           end

        else if (adc_out_store>=  7'b0101000 && adc_out_store<= 7'b0101101)
begin
                ac_relay <= 1;
               fan_relay <= 0;
               heater_led<= 0;

           end
```

```
        else begin
                ac_relay <= 1;
                fan_relay <= 1;
                heater_led <= 1;
                end
        end
end
end
endmodule
```

## Verilog Code Explanation

### **STEP 1**: Module Declearation

- **clk1:** High-frequency clock (ex: 50 MHz).
- **adc_out**: Serial output from the ADC (1 bit per clock).
- **rst**: Reset signal.
- **sclk**: Clock to drive the ADC (derived from clk).
- **cs**: Chip Select for ADC (active low).
- **led_out**: Shows 7 bits of temperature or ADC value.
- **clk**: Slower clock (1 Hz), derived from clk1.
- **ac_relay, fan_relay**: Control signals to activate relays.
- heater_led : to control
- **pir**: PIR motion sensor input (1 = motion detected).

### **STEP 2:** Internal Registers declearation

- **adc_out_store**: 12-bit register to store the ADC reading.
- **count_out**: Keeps track of how many bits have been received from the ADC.
- **count**: Used for cs signal timing.
- **count1**: Used to generate a slower clock from clk1.
- **temperature**: Holds the temperature value.

### **STEP 3:** Clock Divider

- Divides clk1 to a slower clock clk (approx. 1 Hz if clk1 = 50 MHz).

- clk toggles every 25 million cycles.

## **STEP 4**: Initialization

- Sets initial values (not synthesizable in all FPGA tools — better to reset inside a reset block).

## **STEP 4**: Chip select Handling

- After 1 clock cycle, pulls cs low to enable ADC

## **STEP 5:** Serial Clock for ADC

- If cs is high (inactive), sclk is high. When cs is low (active), sclk follows clk.

## **STEP 6**: Count Increment

- Increments count every rising edge of clk1. Used for cs timing.

## **STEP 7**: Main Logic

- When rst is high, all registers are reset.
- When count_out < 12, 12-bit ADC data is serially read.
- Once full data is captured:
  - Store 7 LSBs in led_out and temperature.
  - Depending on adc_out_store and pir, the relays are controlled.

## **STEP 8:** Relay logic conditions

- If adc_out_store < 40 (binary 0101000) and motion is detected → all OFF.
- If in range [40 to 45] → only AC ON
- If > 45 → all ON.

# Costraints file

| Node Name | Direction | I/O Bank | VREF Group | Location | Fitter Location | I/O Standard | Reserved | Current Strength | Slew Rate | Differential Pair | ver Analog Settings | _GXB/VC( |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ac_relay | Output | 4A | B4A_N0 | PIN_AA15 | PIN_AA15 | 3.3-V LVCMOS | | 2mA (default) | 1 (default) | | | |
| adc_out | Input | 3A | B3A_N0 | PIN_AD4 | PIN_AD4 | 3.3-V LVCMOS | | 2mA (default) | | | | |
| clk | Output | 5A | B5A_N0 | PIN_W15 | PIN_W15 | 3.3-V LVCMOS | | 2mA (default) | 1 (default) | | | |
| clk1 | Input | 3B | B3B_N0 | PIN_V11 | PIN_V11 | 3.3-V LVCMOS | | 2mA (default) | | | | |
| cs | Output | 3A | B3A_N0 | PIN_U9 | PIN_U9 | 3.3-V LVCMOS | | 2mA (default) | 1 (default) | | | |
| fan_relay | Output | 4A | B4A_N0 | PIN_Y15 | PIN_Y15 | 3.3-V LVCMOS | | 2mA (default) | 1 (default) | | | |
| heater_relay | Output | 5A | B5A_N0 | PIN_AC24 | PIN_AC24 | 3.3-V LVCMOS | | 2mA (default) | 1 (default) | | | |
| led_out[6] | Output | 5A | B5A_N0 | PIN_AA23 | PIN_AA23 | 3.3-V LVCMOS | | 2mA (default) | 1 (default) | | | |
| led_out[5] | Output | 5A | B5A_N0 | PIN_Y16 | PIN_Y16 | 3.3-V LVCMOS | | 2mA (default) | 1 (default) | | | |
| led_out[4] | Output | 5A | B5A_N0 | PIN_AE26 | PIN_AE26 | 3.3-V LVCMOS | | 2mA (default) | 1 (default) | | | |
| led_out[3] | Output | 5A | B5A_N0 | PIN_AF26 | PIN_AF26 | 3.3-V LVCMOS | | 2mA (default) | 1 (default) | | | |
| led_out[2] | Output | 5A | B5A_N0 | PIN_V15 | PIN_V15 | 3.3-V LVCMOS | | 2mA (default) | 1 (default) | | | |
| led_out[1] | Output | 5A | B5A_N0 | PIN_V16 | PIN_V16 | 3.3-V LVCMOS | | 2mA (default) | 1 (default) | | | |
| led_out[0] | Output | 5A | B5A_N0 | PIN_AA24 | PIN_AA24 | 3.3-V LVCMOS | | 2mA (default) | 1 (default) | | | |
| pir | Input | 5A | B5A_N0 | PIN_AD26 | PIN_AD26 | 2.5 V (default) | | 12mA (default) | | | | |
| rst | Input | 5B | B5B_N0 | PIN_Y24 | PIN_Y24 | 3.3-V LVCMOS | | 2mA (default) | | | | |
| sclk | Output | 3A | B3A_N0 | PIN_V10 | PIN_V10 | 3.3-V LVCMOS | | 2mA (default) | 1 (default) | | | |