

IR Proximity Alert System with Visual & Audio Feedback

TEAM MEMBERS:

N.CHANDANA [23BEC1219], HARINI KAMATCHI VELRAJAN [23BEC1136]

PROBLEM STATEMENT:

Develop an IR Proximity Alert System using a VEGA Aries board that:

- Activates a Red LED + Buzzer when an object is detected (warning state).
- Shows a Green LED + Silent mode when no object is present (safe state).
- Provides real-time debugging via Serial Monitor (for calibration and testing).

COMPONENTS USED:

Aries Development Board v3, USB Cable, IR sensor, Buzzer, RYG LED Strip

PIN CONNECTIONS:

Component	VEGA Aries Connection
IR Sensor (HW-201)	VCC → 3.3V
	GND → GND
	OUT → GPIO1 (Analog Input)
Buzzer	VCC → 3.3V
	I/P → GPIO0
	GND → GND
LED Strip	GND → GND
	Red → PWM0
	Yellow → PWM1 (unused)
	Green → PWM2

CODE:

```
// Pin Definitions
```

```
const int BUZZER_PIN = 0;    // GPIO0 (Buzzer)
const int PROX_SENSOR_PIN = 1; // GPIO1 (Digital Input from IR Sensor)
const int RED_LED_PIN = 0;    // PWM0 (Red LED)
const int GREEN_LED_PIN = 2;  // PWM2 (Green LED)

void setup() {
  pinMode(BUZZER_PIN, OUTPUT);
  pinMode(RED_LED_PIN, OUTPUT);
  pinMode(GREEN_LED_PIN, OUTPUT);

  // Set sensor pin as INPUT (digital read)
  pinMode(PROX_SENSOR_PIN, INPUT);

  Serial.begin(115200); // For debugging
}

void loop() {
  // Read digital value (HIGH = object detected, LOW = no object)
  bool objectDetected = digitalRead(PROX_SENSOR_PIN);

  Serial.print("Sensor State: ");
  Serial.println(objectDetected ? "NOT DETECTED" : "OBJECT DETECTED");

  if (objectDetected) {
    // Object detected → Red LED + Buzzer ON
    analogWrite(RED_LED_PIN, 0); // Full brightness
    analogWrite(GREEN_LED_PIN, 255); // Turn off green
    digitalWrite(BUZZER_PIN, LOW);
    // Buzzer ON
  } else {
    // No object → Green LED + Buzzer OFF
    analogWrite(RED_LED_PIN, 255); // Turn off red
    analogWrite(GREEN_LED_PIN, 0); // Full brightness
    digitalWrite(BUZZER_PIN, HIGH); // Buzzer OFF
  }

  delay(10); // Small delay to avoid flickering
}
```

The image shows a laptop screen with a terminal window displaying a series of log messages. The messages are as follows:

```

11:29:22.103 -> Sensor State: OBJECT DETECTED
11:29:22.151 -> Sensor State: OBJECT DETECTED
11:29:22.151 -> Sensor State: OBJECT DETECTED
11:29:22.151 -> Sensor State: OBJECT DETECTED
11:29:22.151 -> Sensor State: OBJECT DETECTED
11:29:22.199 -> Sensor State: OBJECT DETECTED
11:29:22.199 -> Sensor State: OBJECT DETECTED
11:29:22.199 -> Sensor State: OBJECT DETECTED
11:29:22.247 -> Sensor State: OBJECT DETECTED
11:29:22.247 -> Sensor State: OBJECT DETECTED
11:29:22.247 -> Sensor State: OBJECT DETECTED
11:29:22.247 -> Sensor State: OBJECT DETECTED
11:29:22.294 -> Sensor State: OBJECT DETECTED
11:29:22.294 -> Sensor State: OBJECT DETECTED
11:29:22.294 -> Sensor State: OBJECT DETECTED
11:29:22.294 -> Sensor State: OBJECT DETECTED
11:29:22.341 -> Sensor State: OBJECT DETECTED
11:29:22.341 -> Sensor State: OBJECT DETECTED
11:29:22.341 -> Sensor State: OBJECT DETECTED
11:29:22.388 -> Sensor State: OBJECT DETECTED
11:29:22.388 -> Sensor State: OBJECT DETECTED
11:29:22.435 -> Sensor State: OBJECT DETECTED

```

Below the terminal window, the following code is visible:

```

// GR100 (Buzzer)
// GR101 (Digital I
// PWM (cmd LED)
// PWM (cmd LED)

```

The laptop screen also shows a Windows taskbar with various icons and a system tray with the date and time (11:29:22.388).

In the foreground, a breadboard circuit is connected to the laptop. The circuit includes an Arduino Uno microcontroller, a breadboard, several resistors, and a small motor. A hand is pointing at the circuit, indicating a specific component or connection.

