

```
// Definir pines de los componentes

const int trigPin = 9;    // Pin Trig del sensor ultrasónico
const int echoPin = 10;   // Pin Echo del sensor ultrasónico
const int ledVerde = 7;   // Pin del LED verde
const int ledRojo = 6;    // Pin del LED rojo
const int buzzerPin = 5;  // Pin del buzzer
const int bombaPin = 3;   // Pin de la bomba de agua (controlada por transistor TIP31C)
```

```
// Variables

long duracion;           // Duración del pulso del sensor ultrasónico
int distancia;           // Distancia calculada
const int distanciaActivacion = 15; // Distancia mínima para activar el dispensador (15 cm)
```

```
void setup() {

    // Configurar los pines de los componentes

    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    pinMode(ledVerde, OUTPUT);
    pinMode(ledRojo, OUTPUT);
    pinMode(buzzerPin, OUTPUT);
```

```
pinMode(bombaPin, OUTPUT);

// Inicializar comunicación serial para depuración
Serial.begin(9600);
}

void loop() {
    // Enviar pulso del sensor ultrasónico
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);

    // Leer el pulso de retorno
    duracion = pulseIn(echoPin, HIGH);

    // Calcular la distancia en cm
    distancia = duracion * 0.034 / 2;
```

```
// Mostrar la distancia medida para depuración
Serial.print("Distancia: ");
Serial.print(distancia);
Serial.println(" cm");

// Activar el dispensador si la distancia es menor a la distancia de activación (15 cm)
if (distancia <= distanciaActivacion) {
    // Encender LED verde y activar la bomba por 500 ms
    digitalWrite(ledVerde, HIGH);
    digitalWrite(bombaPin, HIGH);

    // Generar un tono en el buzzer
    tone(buzzerPin, 1000); // Tono de 1000 Hz
    delay(150);           // Esperar 500 ms (duración de la bomba)

    // Apagar la bomba y el buzzer
    digitalWrite(bombaPin, LOW);
    noTone(buzzerPin);

    // Apagar LED verde y encender LED rojo por 4 segundos
```

```
digitalWrite(ledVerde, LOW);  
digitalWrite(ledRojo, HIGH);  
delay(4000); // Esperar 4 segundos
```

```
// Apagar el LED rojo
```

```
digitalWrite(ledRojo, LOW);
```

```
}
```

```
// Esperar un corto tiempo antes de la próxima medición
```

```
delay(200); // Asegurar que no haya lecturas falsas
```

```
}
```