**Manual Técnico para el Juego de Buscaminas con Arduino, LCD, y Flip-Flops**

**Descripción Manual Técnico para el Juego de Buscaminas 3x3 con Arduino, LCD, y Flip-Flops**

**Descripción General**

**Este proyecto implementa un juego de Buscaminas en una matriz 3x3 utilizando Arduino, un display LCD I2C, comunicación serial Bluetooth, y flip-flops 74LS174 para manejar el estado de las casillas. El juego permite configurar la posición de bombas, descubrir casillas y determina si el jugador gana o pierde.**

**Componentes y Librerías Utilizadas**

1. **Arduino: Para el control y lógica del juego.**
2. **LCD I2C (LiquidCrystal\_I2C): Para mostrar mensajes en pantalla.**
3. **Flip-Flops 74LS174: Para almacenar el estado de cada casilla (descubierta o no).**
4. **Bluetooth (SoftwareSerial): Para recibir comandos a través de Bluetooth.**
5. **Librerías:**
   * **Wire.h y LiquidCrystal\_I2C.h para el manejo del display LCD.**
   * **SoftwareSerial.h para la comunicación Bluetooth.**

**Variables y Pines**

* **LCD: LiquidCrystal\_I2C lcd(0x27, 20, 4); Inicializa el LCD en la dirección 0x27 con 20x4 dimensiones.**
* **Flip-Flops:**
  + **D\_pins[9]: Pines de datos (D) para cada flip-flop (casillas).**
  + **CLK\_pin1 y CLK\_pin2: Pines de reloj (CLK) para los dos conjuntos de flip-flops.**
* **Bluetooth: SoftwareSerial BT(0, 1); RX y TX para la comunicación Bluetooth.**
* **Variables de Juego:**
  + **bombas[9]: Define la posición de las bombas.**
  + **descubiertas[9]: Guarda el estado de cada casilla (descubierta o no).**
  + **flipFlopStates[9]: Estado actual de los flip-flops (alto o bajo).**
  + **estado: Estado del juego (0: configuración, 1: jugando, 2: game over, 3: victoria).**

**Estructura y Funciones**

**setup()**

**Inicializa los componentes:**

* **Configura el LCD para mostrar mensajes de bienvenida.**
* **Define los pines de los flip-flops como salida (OUTPUT).**
* **Inicia la comunicación serial y Bluetooth.**

**loop()**

**Realiza las siguientes acciones en un bucle:**

1. **Lee Entradas: Revisa entradas de Serial y Bluetooth.**
2. **Cambia Estados: Cambia entre los modos de configuración y juego según la entrada del usuario.**
3. **Llama a Funciones de Juego:**
   * **configurarBombas(input): Configura bombas en el tablero.**
   * **jugar(input): Ejecuta la lógica del juego.**

**Funciones de Juego**

**configurarBombas(String input)**

**Permite al usuario configurar la posición de las bombas en el tablero:**

* **Cuando el usuario ingresa una posición (A0-A8), se asigna una bomba a esa casilla.**
* **Si se ingresa “A9”, se termina la configuración y se inicia el juego.**

**jugar(String input)**

**Controla la lógica del juego, detecta si el jugador elige una casilla segura o una bomba:**

* **Casilla segura: Cambia el estado del flip-flop correspondiente y muestra mensaje en el LCD.**
* **Bomba: Cambia el estado a Game Over y muestra el mensaje en el LCD y Bluetooth.**
* **Verifica Victoria: Si todas las casillas seguras están descubiertas, cambia el estado a victoria.**

**actualizarFlipFlops()**

**Activa los relojes CLK\_pin1 y CLK\_pin2 para reflejar el estado de los flip-flops en hardware.**

**gameOver()**

**Muestra el mensaje de Game Over y reinicia los flip-flops y variables de juego.**

**ganaste()**

**Muestra el mensaje de victoria y reinicia los flip-flops y variables de juego.**

**Diagrama de Flujo de Estados**

1. **Estado de Configuración (estado = 0): Configura la posición de bombas.**
2. **Estado de Juego (estado = 1): Juega hasta descubrir una bomba o ganar.**
3. **Game Over (estado = 2): Muestra Game Over y permite reiniciar.**
4. **Victoria (estado = 3): Muestra el mensaje de victoria y permite reiniciar.**

**Funcionamiento General**

* **Configura Bombas: Usa entradas A0-A8 para seleccionar posiciones de bombas.**
* **Juega: Selecciona casillas con entradas A0-A8 para descubrir casillas seguras o bombas.**
* **Estados de Flip-Flops: Actualiza el hardware de flip-flops según el estado de las casillas.**

**Mensajes en LCD y Bluetooth**

* **Mensajes de Estado:**
  + **"Buscaminas 3x3 Iniciado"**
  + **"Reiniciando el juego"**
  + **"Configura las bombas A0-A8"**
  + **"Selecciona una casilla (A0-A8)"**
  + **"¡Boom! Bomba en casilla X"**
  + **"Casilla X segura"**
  + **"¡Game Over!"**
  + **"¡Felicidades, ganaste!"**

**General**

Este proyecto implementa un juego de Buscaminas en una matriz 3x3 utilizando Arduino, un display LCD I2C, comunicación serial Bluetooth, y flip-flops 74LS174 para manejar el estado de las casillas. El juego permite configurar la posición de bombas, descubrir casillas y determina si el jugador gana o pierde.

**Componentes y Librerías Utilizadas**

1. **Arduino**: Para el control y lógica del juego.
2. **LCD I2C (LiquidCrystal\_I2C)**: Para mostrar mensajes en pantalla.
3. **Flip-Flops 74LS174**: Para almacenar el estado de cada casilla (descubierta o no).
4. **Bluetooth (SoftwareSerial)**: Para recibir comandos a través de Bluetooth.
5. **Librerías**:
   * Wire.h y LiquidCrystal\_I2C.h para el manejo del display LCD.
   * SoftwareSerial.h para la comunicación Bluetooth.

**Variables y Pines**

* **LCD**: LiquidCrystal\_I2C lcd(0x27, 20, 4); Inicializa el LCD en la dirección 0x27 con 20x4 dimensiones.
* **Flip-Flops**:
  + D\_pins[9]: Pines de datos (D) para cada flip-flop (casillas).
  + CLK\_pin1 y CLK\_pin2: Pines de reloj (CLK) para los dos conjuntos de flip-flops.
* **Bluetooth**: SoftwareSerial BT(0, 1); RX y TX para la comunicación Bluetooth.
* **Variables de Juego**:
  + bombas[9]: Define la posición de las bombas.
  + descubiertas[9]: Guarda el estado de cada casilla (descubierta o no).
  + flipFlopStates[9]: Estado actual de los flip-flops (alto o bajo).
  + estado: Estado del juego (0: configuración, 1: jugando, 2: game over, 3: victoria).

**Estructura y Funciones**

**setup()**

Inicializa los componentes:

* Configura el LCD para mostrar mensajes de bienvenida.
* Define los pines de los flip-flops como salida (OUTPUT).
* Inicia la comunicación serial y Bluetooth.

**loop()**

Realiza las siguientes acciones en un bucle:

1. **Lee Entradas**: Revisa entradas de Serial y Bluetooth.
2. **Cambia Estados**: Cambia entre los modos de configuración y juego según la entrada del usuario.
3. **Llama a Funciones de Juego**:
   * configurarBombas(input): Configura bombas en el tablero.
   * jugar(input): Ejecuta la lógica del juego.

**Funciones de Juego**

**configurarBombas(String input)**

Permite al usuario configurar la posición de las bombas en el tablero:

* Cuando el usuario ingresa una posición (A0-A8), se asigna una bomba a esa casilla.
* Si se ingresa “A9”, se termina la configuración y se inicia el juego.

**jugar(String input)**

Controla la lógica del juego, detecta si el jugador elige una casilla segura o una bomba:

* **Casilla segura**: Cambia el estado del flip-flop correspondiente y muestra mensaje en el LCD.
* **Bomba**: Cambia el estado a Game Over y muestra el mensaje en el LCD y Bluetooth.
* **Verifica Victoria**: Si todas las casillas seguras están descubiertas, cambia el estado a victoria.

**actualizarFlipFlops()**

Activa los relojes CLK\_pin1 y CLK\_pin2 para reflejar el estado de los flip-flops en hardware.

**gameOver()**

Muestra el mensaje de Game Over y reinicia los flip-flops y variables de juego.

**ganaste()**

Muestra el mensaje de victoria y reinicia los flip-flops y variables de juego.

**Diagrama de Flujo de Estados**

1. **Estado de Configuración** (estado = 0): Configura la posición de bombas.
2. **Estado de Juego** (estado = 1): Juega hasta descubrir una bomba o ganar.
3. **Game Over** (estado = 2): Muestra Game Over y permite reiniciar.
4. **Victoria** (estado = 3): Muestra el mensaje de victoria y permite reiniciar.

**Funcionamiento General**

* **Configura Bombas**: Usa entradas A0-A8 para seleccionar posiciones de bombas.
* **Juega**: Selecciona casillas con entradas A0-A8 para descubrir casillas seguras o bombas.
* **Estados de Flip-Flops**: Actualiza el hardware de flip-flops según el estado de las casillas.

**Mensajes en LCD y Bluetooth**

* **Mensajes de Estado**:
  + "Buscaminas 3x3 Iniciado"
  + "Reiniciando el juego"
  + "Configura las bombas A0-A8"
  + "Selecciona una casilla (A0-A8)"
  + "¡Boom! Bomba en casilla X"
  + "Casilla X segura"
  + "¡Game Over!"
  + "¡Felicidades, ganaste!"