



Práctica 8:

PING PONG

Prof. Rubén Galicia Mejía

Equipo:

Los Pingüinos de Madagascar

Integrantes:

- Ayala Fuentes Sunem Gizeht
- Dávila de Jesús Sandro Armando
- García García Francisco
- Guevara Badillo Areli Alejandra

Fecha de realización: *lunes, 01 de julio de 2024*

Fecha de entrega: *martes, 02 de junio de 2024*

MATERIALES

- 2 Display de 7 segmentos de ánodo común
- 10 leds
- 10 resistencia de $3.3k\Omega$
- Placa de pruebas (protoboard)
- Cables y conectores
- 2 botones
- FPGA Cyclone IV

INTRODUCCIÓN

En esta práctica, se desarrollará un juego de ping-pong utilizando LEDs y botones, implementado mediante programación en VHDL y desplegado en una FPGA. El objetivo del proyecto es simular un juego en el que dos jugadores compiten, cada uno con un botón que deben presionar para devolver "la pelota" representada por un LED que se mueve de un lado a otro. El juego incluye un sistema de puntuación que se muestra en displays de 7 segmentos. Este proyecto abarca desde la programación en VHDL hasta el ensamblado del circuito en una placa de pruebas.

MARCO TEÓRICO

El juego de ping pong, también conocido como tenis de mesa, es un deporte que se juega entre dos o cuatro jugadores. Se originó en Inglaterra en el siglo XIX y rápidamente ganó popularidad en todo el mundo. Aquí tienes una descripción teórica sobre sus reglas, equipamiento, y técnica:

Reglas Básicas

1. Objetivo del Juego:

- El objetivo es golpear la pelota con una raqueta para que pase sobre la red y toque el lado de la mesa del oponente sin que el oponente pueda devolverla correctamente como se ilustra en la ilustración 1.

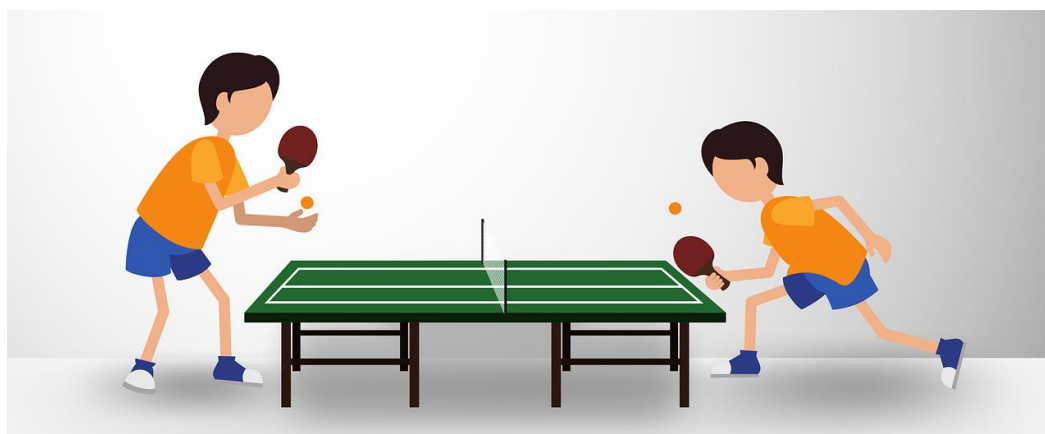


Ilustración 1. Ilustración sobre el juego de ping pong

2. Puntuación:

- Los partidos se juegan al mejor de tres, cinco o siete sets.
- Un set se juega a 11 puntos, y el jugador debe ganar por un margen de al menos 2 puntos.
- Si el puntaje llega a 10-10, se continúa jugando hasta que un jugador tenga una ventaja de 2 puntos.

3. Servicio:

- El servicio debe comenzar con la pelota en la palma de la mano abierta del servidor y ser lanzada al menos 16 cm en el aire sin efecto.
- La pelota debe golpear primero el lado del servidor antes de pasar por encima de la red y golpear el lado del receptor.
- El servicio alterna cada dos puntos entre los jugadores.

4. Intercambio de Golpes:

- Después del servicio, los jugadores golpean la pelota alternativamente.
- La pelota debe golpear una vez el lado del jugador que recibe antes de ser devuelta.
- El intercambio continúa hasta que un jugador no pueda devolver la pelota correctamente.

Equipamiento

1. Mesa:

- La mesa mide 2.74 metros de largo, 1.525 metros de ancho y está a 76 cm del suelo.
- Está dividida en dos mitades por una red que mide 15.25 cm de altura.

2. Pelota:

- La pelota es de celuloide o plástico similar y tiene un diámetro de 40 mm y un peso de 2.7 gramos.
- Las pelotas son generalmente de color blanco o naranja y tienen una clasificación de estrellas para indicar su calidad.

3. Raqueta:

- Las raquetas están cubiertas con goma en uno o ambos lados, dependiendo del estilo del jugador.
- La goma puede tener diferentes texturas y espesores, lo que afecta el control y la velocidad de la pelota.

Técnica

1. Postura y Posición:

- Los jugadores deben mantener una postura equilibrada con las rodillas ligeramente flexionadas y el cuerpo inclinado hacia adelante.
- Los pies deben estar separados a la anchura de los hombros para un equilibrio óptimo.

2. Golpes Básicos:

- **Drive:** Un golpe ofensivo con un movimiento hacia adelante y hacia arriba.
- **Push:** Un golpe defensivo con un movimiento hacia abajo y hacia adelante.
- **Smash:** Un golpe fuerte y rápido utilizado para finalizar el punto.
- **Topspin:** Un golpe con efecto hacia adelante, haciendo que la pelota gire hacia el oponente.
- **Backspin:** Un golpe con efecto hacia atrás, haciendo que la pelota gire hacia el jugador que la golpeó.

3. Estrategia:

- Los jugadores deben mezclar sus golpes y efectos para desestabilizar al oponente.
- Es crucial anticipar los movimientos del oponente y posicionarse adecuadamente para cada golpe.

El tenis de mesa es un deporte que combina habilidad, estrategia y rapidez. Los jugadores deben desarrollar reflejos rápidos, precisión en los golpes y una comprensión profunda de las tácticas para ser exitosos en el juego.

DESARROLLO DE LA PRÁCTICA

El desarrollo de esta práctica se divide en dos grandes secciones: la programación del código VHDL y el ensamblado del circuito en la FPGA.

Desarrollo del código en VHDL

El código VHDL para el juego de ping-pong se presenta a continuación. Esta sección explicará en detalle cada parte del código, desde la declaración de entidades hasta la lógica de funcionamiento.

Se implementa un juego de Pong básico en una FPGA. Utiliza botones para que dos jugadores controlen el movimiento del "puck" (representado por LEDs) y muestra el puntaje de cada jugador en displays de 7 segmentos.

La entidad `pong` define las entradas y salidas del sistema:

```
entity pong is
  Port (
    clk : in std_logic; -- Reloj principal
    reset : in std_logic; -- Señal de reset
    btn_player1 : in std_logic; -- Botón del jugador 1
    btn_player2 : in std_logic; -- Botón del jugador 2
    start_game : in std_logic; -- Botón para iniciar el juego
    freq_select : in std_logic_vector(1 downto 0); -- Selector de frecuencia
    led : out std_logic_vector(9 downto 0); -- LEDs de salida
    seg_player1 : out std_logic_vector(6 downto 0); -- Display del jugador 1
    seg_player2 : out std_logic_vector(6 downto 0) -- Display del jugador 2
  );
end pong;
```

La entidad `pong` define los puertos de entrada y salida:

- `clk`: Señal de reloj.
- `reset`: Señal para resetear el juego.
- `btn_player1` y `btn_player2`: Botones de los jugadores.
- `start_game`: Botón para iniciar el juego.
- `freq_select`: Selector de frecuencia para ajustar la velocidad del juego.
- `led`: Vector de LEDs para mostrar la posición del puck.
- `seg_player1` y `seg_player2`: Displays de 7 segmentos para mostrar los puntajes.

```
architecture Behavioral of pong is
  signal led_reg : std_logic_vector(9 downto 0) := (others => '0');
  signal current_led : integer range 0 to 9 := 0;
  signal direction : std_logic := '1';
  signal game_running : std_logic := '0';
  signal start_pressed : std_logic := '0';
  signal stop_flag : std_logic := '0';
  signal player1_score : integer range 0 to 9 := 0;
```

```

signal player2_score : integer range 0 to 9 := 0;
signal update_clk_prescaler : integer := 16666667;
signal update_clk_count : integer := 0;

```

Estas señales manejan el estado del juego, incluyendo la posición actual del LED, la dirección del movimiento, si el juego está en marcha, si se ha presionado el botón de inicio, y los puntajes de los jugadores.

Proceso principal del juego

```

process(clk)
begin
    if rising_edge(clk) then
        if reset = '1' then
            ...
        elsif start_game = '1' and start_pressed = '0' then
            ...
        elsif start_game = '0' then
            start_pressed <= '0';
        end if;

        if game_running = '1' and stop_flag = '0' then
            if update_clk_count >= update_clk_prescaler then
                if direction = '1' then
                    if current_led = 9 then
                        if btn_player2 = '1' then
                            direction <= '0';
                        else
                            stop_flag <= '1';
                            player1_score <= player1_score + 1;
                            if player1_score = 9 then
                                led_reg <= (others => '1');
                                game_running <= '0';
                            end if;
                        end if;
                    end if;
                else
                    current_led <= current_led + 1;
                end if;
            else
                if current_led = 0 then
                    if btn_player1 = '1' then
                        direction <= '1';
                    else
                        stop_flag <= '1';
                        player2_score <= player2_score + 1;
                        if player2_score = 9 then
                            led_reg <= (others => '1');
                            game_running <= '0';
                        end if;
                    end if;
                else
                    current_led <= current_led - 1;
                end if;
            end if;

            if game_running = '1' then
                led_reg <= (others => '0');
                led_reg(current_led) <= '1';
            end if;
            update_clk_count <= 0;
        else
            update_clk_count <= update_clk_count + 1;
        end if;
    end if;
end process;

```

```

        elsif stop_flag = '1' then
            if (direction = '1' and btn_player2 = '1') or (direction = '0' and btn_player1
= '1') then
                if direction = '1' then
                    current_led <= 9;
                    direction <= '0';
                else
                    current_led <= 0;
                    direction <= '1';
                end if;
                stop_flag <= '0';
                game_running <= '1';
            end if;
        end if;
    end if;
end process;

```

Este proceso maneja la lógica del juego, incluyendo el movimiento del puck, la detección de colisiones con los botones de los jugadores, la actualización de los puntajes y la terminación del juego.

```

led <= led_reg;
seg_player1 <= to_7_segment_inverted(player2_score);
seg_player2 <= to_7_segment(player1_score);

```

Finalmente, se asignan los valores calculados a los LEDs y displays de 7 segmentos.

Ensamblado del Circuito y Funcionamiento

Comenzamos colocando la protoboard en el centro de la mesa. Esta sería nuestra base para conectar todos los componentes de manera ordenada y eficiente. Luego, tomamos los dos displays de 7 segmentos y los insertamos cuidadosamente en la protoboard. Nos aseguramos de que los ánodos comunes de cada display estuvieran conectados a los pines de alimentación. Utilizando cables, conectamos los pines de cada segmento de los displays a los pines correspondientes de la FPGA Cyclone IV, tal y como lo habíamos definido en nuestro código VHDL.

A continuación, posicionamos los diez LEDs en la protoboard, en una línea recta que simularía el movimiento del puck en el juego. Para proteger los LEDs de posibles daños, conectamos una resistencia de 3.3kΩ en serie con cada uno de ellos. Luego, unimos el otro extremo de cada resistencia a los pines correspondientes en la FPGA, asegurándonos de seguir el esquema de conexiones que habíamos planeado.

Los botones para los jugadores fueron colocados estratégicamente en la protoboard. Cada botón sería utilizado para detectar cuando los jugadores intentan devolver el puck. Conectamos un extremo de cada botón a la alimentación y el otro extremo a los pines asignados en la FPGA, utilizando cables de conexión. Estas conexiones eran críticas, ya que determinarían si los jugadores podían interactuar correctamente con el juego.

Con la FPGA Cyclone IV situada al lado de la protoboard, procedimos a realizar las conexiones necesarias para alimentar todos los componentes. Utilizamos cables para conectar la alimentación y la tierra desde la FPGA a las líneas de alimentación de la protoboard.

Revisamos cuidadosamente que todos los pines necesarios de la FPGA estuvieran conectados correctamente a los componentes en la protoboard, asegurándonos de evitar cualquier error que pudiera afectar el funcionamiento del circuito.

Finalmente, conectamos la FPGA al ordenador y cargamos el código VHDL que habíamos desarrollado y verificado previamente. Una vez cargado, encendimos la FPGA y observamos con entusiasmo cómo el puck comenzaba a moverse entre los LEDs como se puede observar en la Ilustración 2.

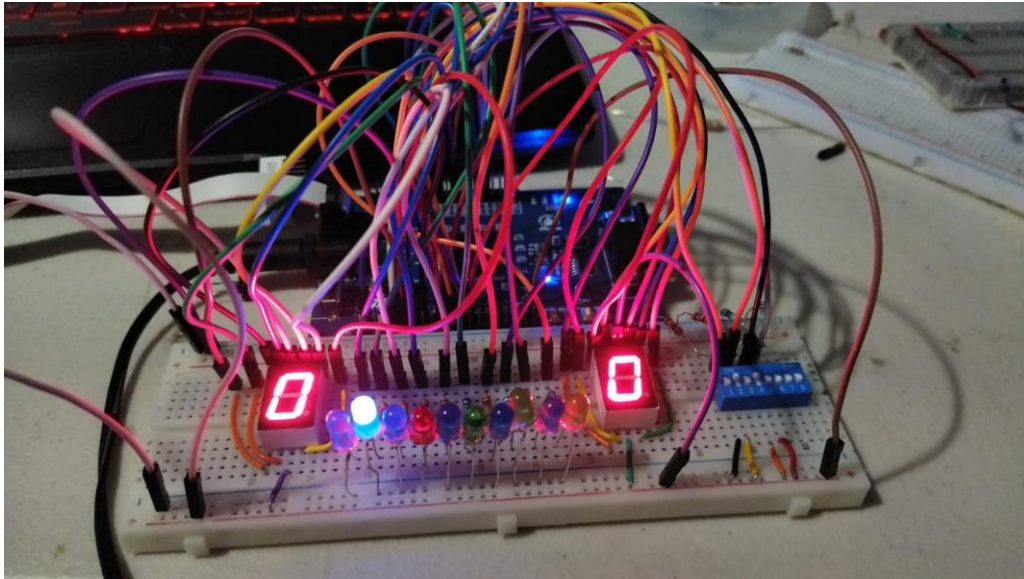


Ilustración 2. Circuito ensamblado y en primeras etapas de funcionamiento,

Una vez conectado, comprobamos que se implementara de forma correcta la lógica del juego.

Los jugadores presionaban sus botones para devolver el puck, y los puntajes se actualizaban correctamente en los displays de 7 segmentos. El circuito funcionaba perfectamente, demostrando que nuestras conexiones y nuestro código eran correctos.

Video del funcionamiento:

<https://drive.google.com/file/d/1Ur1DT17pjDmtC6yTAc5oplu5sSSLsvfP/view?usp=sharing>

CONCLUSIONES

CONCLUSIONES 1 (Ayala Fuentes Sunem Gizeth)

Durante esta práctica, pude consolidar mis conocimientos sobre el diseño y la implementación de circuitos digitales utilizando VHDL y FPGAs. La experiencia de llevar un diseño digital desde el entorno de simulación hasta su implementación en hardware real me permitió comprender la importancia de una correcta sincronización y manejo de señales. Asimismo, aprendí a valorar la necesidad de una planificación detallada, ya que cualquier error en la asignación de pines o en las conexiones físicas puede resultar en un mal funcionamiento del sistema completo. Este proyecto me hizo más consciente de los desafíos y las precisiones requeridas en el diseño de sistemas digitales complejos."

CONCLUSIONES 2 (Dávila de Jesús Sandro Armando)

El proyecto de Pong fue una excelente oportunidad para mejorar mis habilidades en la depuración y verificación de código VHDL. A lo largo de la práctica, pude apreciar la relación directa entre el código escrito y las operaciones físicas del circuito, lo que reforzó mi comprensión sobre la importancia de la lógica secuencial y combinacional. Además, la necesidad de ajustar la frecuencia del reloj y asegurar la sincronización adecuada de los componentes resaltó la relevancia de los conceptos de temporización en el diseño de sistemas digitales. Este proyecto también subrayó la importancia de realizar pruebas exhaustivas y de mantener una actitud meticulosa durante el desarrollo y la implementación del código.

CONCLUSIONES 3 (García García Francisco)

Esta práctica fue una lección práctica en la importancia del trabajo en equipo y la colaboración efectiva en proyectos de ingeniería. Coordinar los esfuerzos para ensamblar el circuito y verificar cada conexión

resaltó la necesidad de una comunicación clara y de roles bien definidos dentro del equipo. Desde un punto de vista técnico, aprendí a manejar con mayor precisión las conexiones en la protoboard y a utilizar las resistencias adecuadas para proteger los componentes electrónicos. Además, la experiencia de solucionar problemas inesperados durante el montaje me permitió desarrollar habilidades de resolución de problemas y entender mejor las complejidades del diseño de hardware digital.

CONCLUSIONES 4 (Guevara Badillo Areli Alejandra)

El proyecto de Pong me brindó una valiosa experiencia en el uso de FPGAs y en la implementación de lógica digital. La traducción de nuestro diseño VHDL a un sistema físico me permitió observar de primera mano la importancia de una correcta asignación de pines y una configuración precisa de los componentes. También aprendí sobre la importancia de los procedimientos de verificación y depuración en cada etapa del desarrollo, ya que cualquier error menor puede afectar significativamente el rendimiento del circuito. Este proyecto también enfatizó la importancia de la documentación y del seguimiento de un proceso metódico en la construcción y prueba de circuitos digitales, habilidades esenciales para cualquier ingeniero en electrónica.

BIBLIOGRAFÍAS

- La historia del tenis de mesa - Cornilleau. (n.d.). Recuperado de <https://es.cornilleau.com/es/content/36-the-history-of-table-tennis>
- Wikipedia. (2024, June 25). Tenis de mesa. Recuperado de https://es.wikipedia.org/wiki/Tenis_de_mesa
- Tenis de Mesa |. (n.d.). Recuperado de <https://www.ull.es/servicios/deportes/actividades/tenis-mesa/>
- Redacción. (2016, October 23). “La metafísica del ping-pong”, un viaje por la filosofía a través del deporte. La Vanguardia. Recuperado de <https://www.lavanguardia.com>