

React Game

José Alejandro Moreno Mesa, Valentina Restrepo Jaramillo
Versión 0.1

Tabla de contenidos

Table of contents

Índice de archivos

Lista de archivos

Lista de todos los archivos documentados y con breves descripciones:

C:/Users/amesa/OneDrive/Personal/PC anterior/Escritorio/React_game - doxy/botones.c (Implementación del control de los pulsadores)	3
C:/Users/amesa/OneDrive/Personal/PC anterior/Escritorio/React_game - doxy/botones.h (Archivo de cabecera para el control de los pulsadores)	4
C:/Users/amesa/OneDrive/Personal/PC anterior/Escritorio/React_game - doxy/display_7seg.c (Implementación del controlador del display de 7 segmentos)	7
C:/Users/amesa/OneDrive/Personal/PC anterior/Escritorio/React_game - doxy/display_7seg.h	8
C:/Users/amesa/OneDrive/Personal/PC anterior/Escritorio/React_game - doxy/LED.h (Definiciones y prototipos de funciones para controlar los LEDs)	9
C:/Users/amesa/OneDrive/Personal/PC anterior/Escritorio/React_game - doxy/main.c (Implementación de la lógica del juego con mensajes en pantalla)	12

Documentación de archivos

Referencia del archivo C:/Users/amesa/OneDrive/Personal/PC anterior/Escritorio/React_game - doxy/botones.c

Implementación del control de los pulsadores.

```
#include "botones.h"
#include "pico/stdlib.h"
```

Funciones

- void **botones_init** ()
Inicializa los pulsadores como pull-down.
- int **leer_botones** (int pin)
Lee el estado de un pulsador específico.

Descripción detallada

Implementación del control de los pulsadores.

Documentación de funciones

void **botones_init** ()

Inicializa los pulsadores como pull-down.

Inicializa los pulsadores.

Esta función inicializa los pines de los pulsadores y los configura como entradas con resistencia pull-down activada.

int **leer_botones** (int *pin*)

Lee el estado de un pulsador específico.

Esta función lee el estado del pulsador asociado al pin especificado.

Parámetros

<i>pin</i>	El número del pin del pulsador que se desea leer.
------------	---

Devuelve

El estado del pulsador (1 si está presionado, 0 si no lo está).

Referencia del archivo C:/Users/amesa/OneDrive/Personal/PC anterior/Escritorio/React_game - doxy/botones.h

Archivo de cabecera para el control de los pulsadores.

defines

- `#define BOTON_START_PIN 13`
Número de pin del pulsador de inicio.
- `#define BOTON_CLEAR_PIN 5`
Número de pin del pulsador del LED clear.
- `#define BOTON_YELLOW_PIN 6`
Número de pin del pulsador del LED yellow.
- `#define BOTON_RED_PIN 7`
Número de pin del pulsador del LED red.

Funciones

- `void botones_init ()`
Inicializa los pulsadores.
- `int leer_botones (int pin)`
Lee el estado de un pulsador específico.

Descripción detallada

Archivo de cabecera para el control de los pulsadores.

Documentación de «define»

`#define BOTON_CLEAR_PIN 5`

Número de pin del pulsador del LED clear.

Define el número de pin asociado al pulsador del LED clear en la Raspberry Pi Pico W.

`#define BOTON_RED_PIN 7`

Número de pin del pulsador del LED red.

Define el número de pin asociado al pulsador del LED red en la Raspberry Pi Pico W.

#define BOTON_START_PIN 13

Número de pin del pulsador de inicio.

Define el número de pin asociado al pulsador de inicio en la Raspberry Pi Pico W.

#define BOTON_YELLOW_PIN 6

Número de pin del pulsador del LED yellow.

Define el número de pin asociado al pulsador del LED yellow en la Raspberry Pi Pico W.

Documentación de funciones

void botones_init ()

Inicializa los pulsadores.

Esta función inicializa los pines de los pulsadores y los configura correctamente.

Inicializa los pulsadores.

Esta función inicializa los pines de los pulsadores y los configura como entradas con resistencia pull-down activada.

int leer_botones (int *pin*)

Lee el estado de un pulsador específico.

Esta función lee el estado del pulsador asociado al pin especificado.

Parámetros

<i>pin</i>	El número del pin del pulsador que se desea leer.
------------	---

Devuelve

El estado del pulsador (1 si está presionado, 0 si no lo está).

botones.h

Ir a la documentación de este archivo.

```
1
6 #ifndef BOTONES_H
7 #define BOTONES_H
8
15 #define BOTON_START_PIN    13
16
23 #define BOTON_CLEAR_PIN    5
24
31 #define BOTON_YELLOW_PIN   6
32
39 #define BOTON_RED_PIN      7
40
46 void botones_init();
47
56 int leer_botones(int pin);
57
58 #endif
```

Referencia del archivo C:/Users/amesa/OneDrive/Personal/PC anterior/Escritorio/React_game - doxy/display_7seg.c

Implementación del controlador del display de 7 segmentos.

```
#include "display_7seg.h"  
#include "pico/stdlib.h"
```

Funciones

- void **display_7seg_init** ()
Inicializa los pines del display de 7 segmentos.
- void **display_7seg_show_number** (int number)
Muestra un número en los 4 dígitos del display de 7 segmentos.

Variables

- const int **segment_pins** [] = {SEGMENT_A_PIN, SEGMENT_B_PIN, SEGMENT_C_PIN, SEGMENT_D_PIN, SEGMENT_E_PIN, SEGMENT_F_PIN, SEGMENT_G_PIN, SEGMENT_DP_PIN}

Descripción detallada

Implementación del controlador del display de 7 segmentos.

Documentación de funciones

void display_7seg_init ()

Inicializa los pines del display de 7 segmentos.

Esta función configura los pines del display de 7 segmentos como salidas.

void display_7seg_show_number (int *number*)

Muestra un número en los 4 dígitos del display de 7 segmentos.

Esta función muestra un número entero de hasta 4 dígitos en los 4 displays de 7 segmentos.

Parámetros

<i>number</i>	El número que se va a mostrar en el display.
---------------	--

display_7seg.h

```
1 //Archivos de cabecera de los 7 segmentos
2 #ifndef DISPLAY_7SEG_H
3 #define DISPLAY_7SEG_H
4
5 // Definiciones de pines para los segmentos del display de 7 segmentos
6 #define SEGMENT_A_PIN 14
7 #define SEGMENT_B_PIN 15
8 #define SEGMENT_C_PIN 16
9 #define SEGMENT_D_PIN 17
10 #define SEGMENT_E_PIN 18
11 #define SEGMENT_F_PIN 19
12 #define SEGMENT_G_PIN 20
13 #define SEGMENT_DP_PIN 21 //punto decimal
14
15 // Definiciones de los dígitos del display de 7 segmentos
16 #define DIGIT_0 0b00111111
17 #define DIGIT_1 0b00000110
18 #define DIGIT_2 0b01011011
19 #define DIGIT_3 0b01001111
20 #define DIGIT_4 0b01100110
21 #define DIGIT_5 0b01101101
22 #define DIGIT_6 0b01111101
23 #define DIGIT_7 0b00000111
24 #define DIGIT_8 0b01111111
25 #define DIGIT_9 0b01101111
26 #define DIGIT_OFF 0b00000000
27
28 // Prototipos de funciones
29 void display_7seg_init(); // Inicialización
30
31 void display_7seg_show_number(int number); // Muestra un número en los 4
32 // dígitos del 7 segmentos
33 #endif
```

Referencia del archivo C:/Users/amesa/OneDrive/Personal/PC anterior/Escritorio/React_game - doxy/LED.h

Definiciones y prototipos de funciones para controlar los LEDs.

defines

- `#define LED_CLEAR_PIN 2`
Definiciones de pines de los LEDs.
- `#define LED_YELLOW_PIN 3`
- `#define LED_RED_PIN 4`

Funciones

- `void leds_init ()`
Inicializa los pines de los LEDs.
- `void led_on (int pin)`
Enciende un LED específico.
- `void led_off (int pin)`
Apaga un LED específico.
- `void led_sequence ()`
Ejecuta la secuencia inicial de luces.

Descripción detallada

Definiciones y prototipos de funciones para controlar los LEDs.

Documentación de «define»

`#define LED_CLEAR_PIN 2`

Definiciones de pines de los LEDs.

Se definen los pines correspondientes a cada LED.

Documentación de funciones

`void led_off (int pin)`

Apaga un LED específico.

Parámetros

<i>pin</i>	El número de pin del LED que se desea apagar.
------------	---

void led_on (int *pin*)

Enciende un LED específico.

Parámetros

<i>pin</i>	El número de pin del LED que se desea encender.
------------	---

void led_sequence ()

Ejecuta la secuencia inicial de luces.

Esta función enciende y apaga los LEDs según una secuencia predefinida.

Ejecuta la secuencia inicial de luces.

Esta función enciende y apaga los LEDs según la secuencia especificada. La secuencia es la siguiente: (1,1,1), (0,1,1), (0,0,1), (0,0,0).

void leds_init ()

Inicializa los pines de los LEDs.

Esta función configura los pines de los LEDs como salidas.

LED.h

Ir a la documentación de este archivo.

```
1
5 #ifndef LED_H
6 #define LED_H
7
14 #define LED_CLEAR_PIN 2 // Pin del led transparente
15 #define LED_YELLOW_PIN 3 // Pin del led amarillo
16 #define LED_RED_PIN 4 // Pin del led rojo
17
23 void leds_init(); // Inicialización
29 void led_on(int pin); // Enciende un LED específico
35 void led_off(int pin); // Apaga un LED específico
41 void led_sequence(); // Ejecuta la secuencia inicial de luces
42
43 #endif
```

Referencia del archivo C:/Users/amesa/OneDrive/Personal/PC anterior/Escritorio/React_game - doxy/main.c

Implementación de la lógica del juego con mensajes en pantalla.

```
#include "LCD_nokia5110.h"
#include "pico/stdlib.h"
#include "hardware/gpio.h"
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include "LED.h"
#include "botones.h"
#include "display_7seg.h"
```

defines

- #define LED_RED_PIN
- #define LED_YELLOW_PIN
- #define LED_CLEAR_PIN
- #define BUTTON_1_PIN
- #define BUTTON_2_PIN
- #define BUTTON_3_PIN
- #define START_BUTTON_PIN
- #define GAME_TIME 10
- #define START_DELAY_MIN_MS 1000
- #define START_DELAY_MAX_MS 10000
- #define MAX_GAME_TIME_SEC 10

Funciones

- int **generate_random_time** (int min, int max)
- void **display_press_button_message** ()
- void **display_game_over_message** ()
- int **main** ()

Función principal del programa.

Descripción detallada

Implementación de la lógica del juego con mensajes en pantalla.

Implementación de la lógica completa del juego con LED, botones y 7 segmentos.

Documentación de funciones

int main ()

Función principal del programa.

Devuelve

0 si la ejecución fue exitosa.

Índice

INDEX