

Propuesta de Trabajo Final de Electrónica Digital 3

Python-chess controller

Docente:

Blasco, Marcos.

Alumnos:

Soriano, Juan.

Vignolles Iván.

Introducción

La idea del proyecto es armar un módulo “joystick” utilizando la placa Nxp LPC1769, para controlar un juego de Ajedrez hosteado en una PC, hecho en Python. Además, la placa brindará información extra a los jugadores, como el tiempo total restante de cada jugador que se mostrará por 7 displays de 7 segmentos, indicará el turno de cada jugador o si se ha finalizado la partida o se ha cometido un error de movimiento de fichas. La placa recibirá información de los jugadores, como el movimiento en los ejes X e Y del joystick para manejar las fichas, un botón de selección, el paso de turno, etc. La placa, además deberá leer datos de la computadora para entregarle información a los jugadores. El link del repositorio empleado para realizar el trabajo es el siguiente: https://github.com/SorianoJuan/ED3_TPFinal .

Materiales

- **Placa Nxp LPC1769 revisión C.**
- **Módulo Joystick - KY023.** Consta de dos potenciómetros que de acuerdo a la posición en la que se encuentre el joystick, entregarán distintos valores de voltaje a la salida.
- **Pulsador – Normal abierto.** Un simple pulsador para controlar la selección de fichas para los movimientos y demás.
- **Botón – Normal abierto.** Al finalizar el turno de cada jugador, se deberá presionar el botón para indicar el final del turno del jugador.
- **Displays de 7 segmentos ánodo común - 848 BS-A51ARI (3).** En los displays se mostrará el tiempo restante total del jugador de turno.
- **Módulo LED RGB ánodo común – KY016.** Indicará el turno del jugador. Al iniciar la partida ambos jugadores podrán elegir el color que desean y el led les indicará de quién es el turno.
- **Buzzer - KY012.** Un simple buzzer que actuará como indicador de final de la partida, error en el movimiento de la ficha, etc.
- **Motor de DC – rf 300c 11440.** Actuará como vibrador del joystick.
- **Módulo USB-UART CP2102.**
- **Fuente Tenstar Robot – Input 100-240 AC; Output 3.3V 3W.** Fuente de alimentación del circuito.
- **Computador con Python 3 instalado.** El juego correrá en la computadora, hecho en Python.
- **Impresora 3D.** La base fue impresa utilizando una impresora 3D.
- **Transistores:**
 - 3x BC640 (PNP) – Multiplexado del display,
 - 7x 2N2222 (NPN) – Segmentos del display.
 - 3x BC640 (PNP) – LED RGB.
 - 1x BC640 (NPN) – Buzzer.
 - 1x BC640 (PNP) – Motor.

- **Resistencias:**

- 3x 1k Ω – Resistencia de base del multiplexado de los displays.
- 7x 33 Ω – Resistencia de colector de los displays.
- 7x 10k Ω – Resistencia de base de los displays.
- 1x 10k Ω – Resistencia de base del motor.
- 1x 33 Ω – Resistencia de colector del motor.
- 3x 10k Ω – Resistencia de base del LED RGB.
- 1x 10k Ω – Resistencia de base del buzzer.
- 1x 33 Ω – Resistencia de colector del buzzer.

Funcionamiento

1. Desde la placa realizaremos la conexión mediante el módulo UART a la PC, utilizando el CP2102.
2. Con el ADC de la placa seteado en modo ráfaga, en los canales AD.0 y AD.1 convertiremos los valores analógicos entregados por el joystick a valores digitales que serán pasados a la PC para indicar los movimientos verticales u horizontales.
3. El pulsador simplemente será leído por la placa en caso de generarse una interrupción externa y se enviará el dato a la computadora que servirá para indicar los movimientos de las fichas.
4. El LED RGB brillará con la codificación de colores correspondiente al jugador de turno.
5. Usando el Timer de la placa, se contará el tiempo del que cada jugador dispone y se irá mostrando el tiempo restante en los displays para simular una partida de ajedrez en modo “Blitz”.
6. Cuando el jugador finalice con su turno, deberá pulsar el botón para indicar que terminó su turno, en cuyo caso la placa leerá este dato, también por interrupción externa y cambiará el valor desplegado en los displays y cambiará la codificación de colores del LED RGB para indicar que es el turno del otro jugador.
7. El buzzer indicará el final de la partida, ya sea porque un jugador ha agotado su tiempo o porque la partida ha finalizado. Además, indicará cualquier error, ya sea de movimiento inválido o demás.
8. Habrá un motor de DC que será ajustado para vibrar e indicar ciertas situaciones que se pueden dar, como por ejemplo, final de la partida.

Desarrollo y construcción

Se implementó el joystick con sus respectivos módulos y el producto final se muestra a continuación:

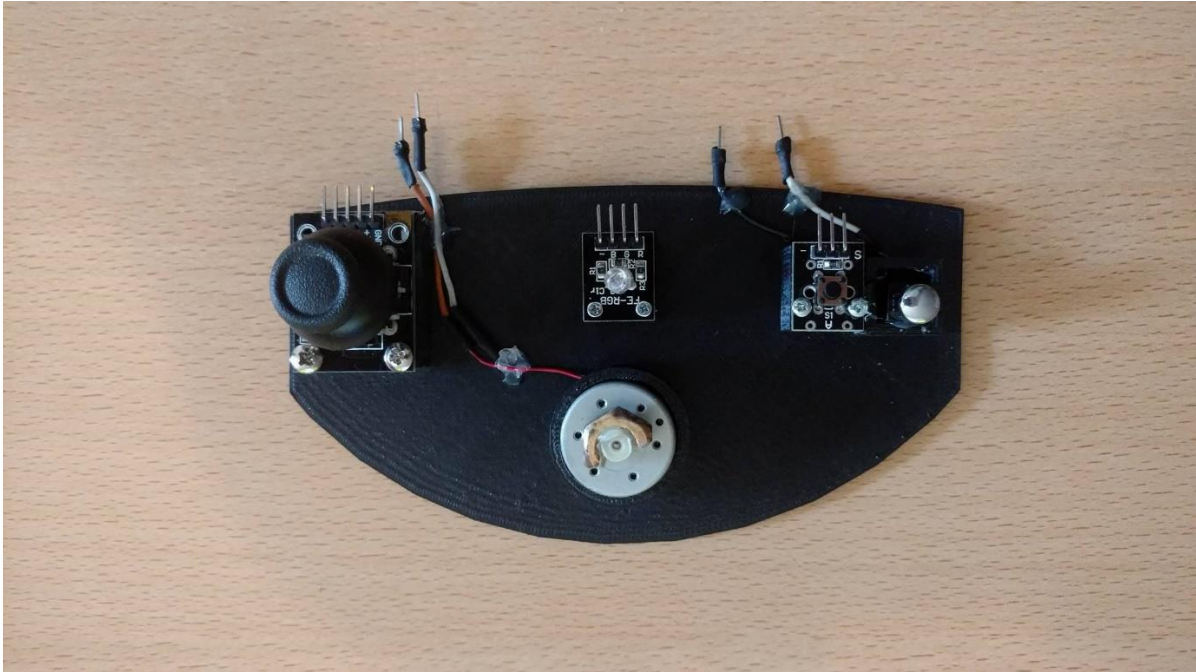


Figura 1. Joystick construido.

La parte de los displays y el buzzer, fue armada por separado con el fin de no agobiar el joystick con demasiados módulos. La base de la construcción fue diseñada e impresa con una impresora 3D.

A continuación, se presenta el diseño realizado para la base:

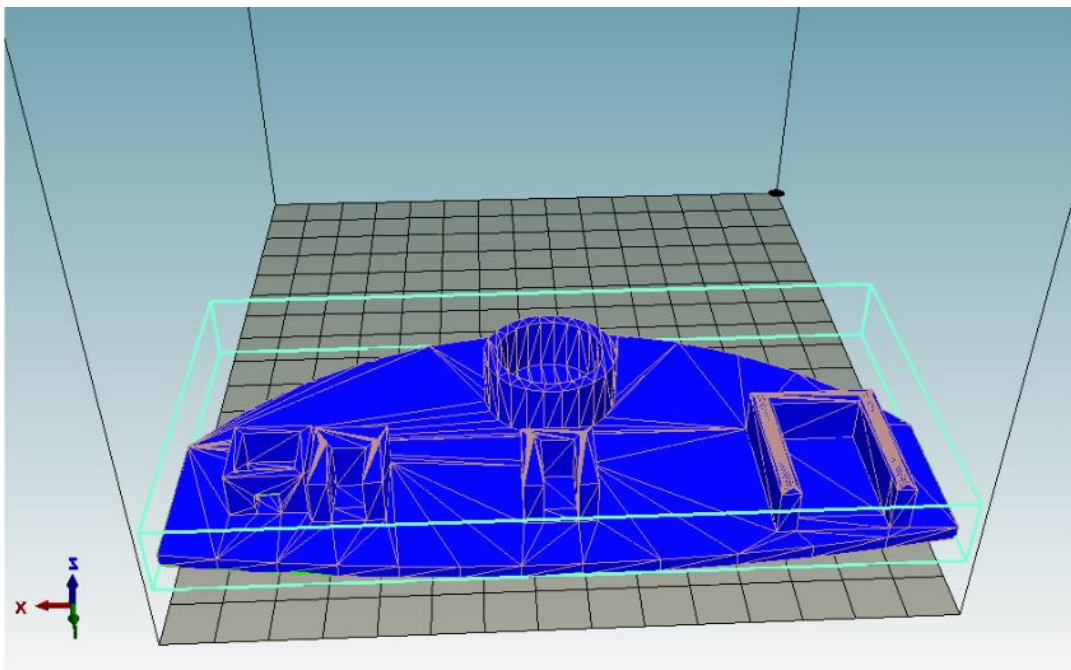


Figura 2. Diseño de la base del joystick

El conexionado de los periféricos se realizó como se indica en el siguiente diagrama ilustrativo:

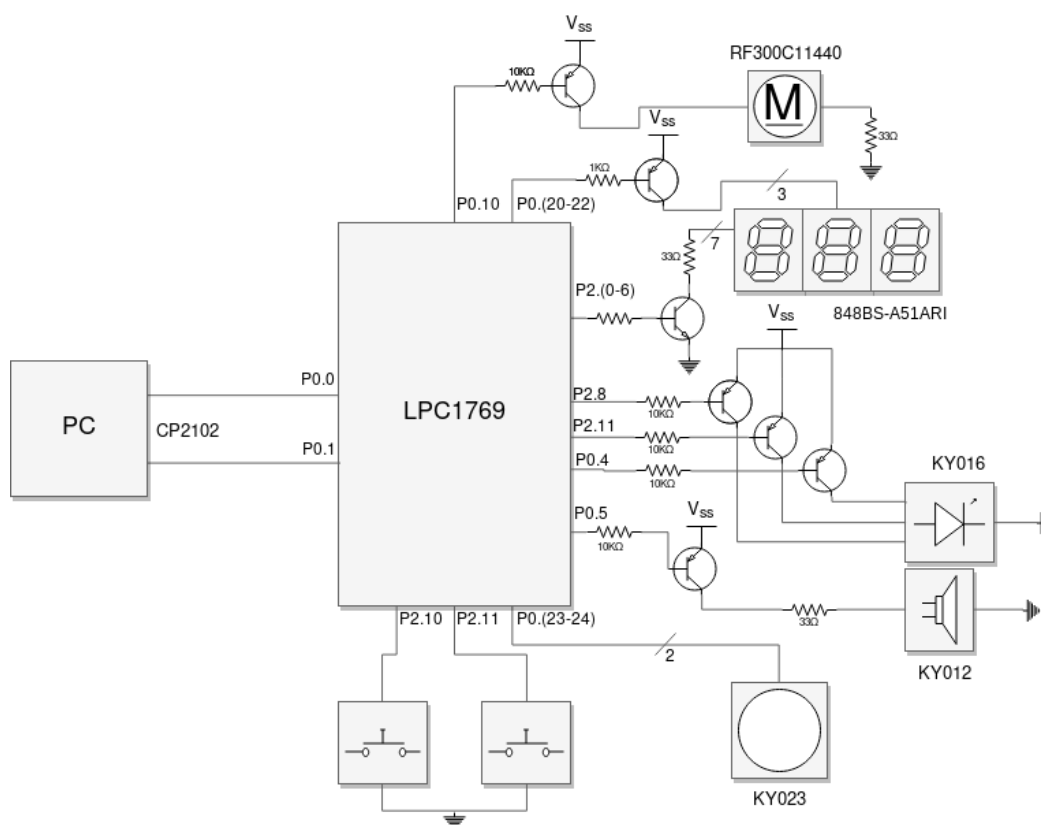


Figura 3. Esquema de conexión del circuito.

Una vez armado el joystick, el juego se ejecuta en una pc y está hecho en Python. La pc recibe el input que la placa le envía por UART a USB por medio del módulo USB-UART CP2102. Se debió implementar una interfaz para leer los comandos ingresados y enviar tramas serie de vuelta a la placa. El juego que se probó, Python-chess, es de código abierto. Fue necesario realizarle modificaciones para poder adaptarlo a nuestro proyecto. En la siguiente figura se muestra el Python-chess con algunas modificaciones:



Figura 4. Python-chess