



**Universidad de San Carlos de Guatemala**  
**Facultad de Ingeniería**  
**Escuela de Ingeniería Mecánica Eléctrica**  
**Laboratorio de Electrónica**  
**Proyecto de Microcontroladores**

## **DESCRIPCIÓN**

El proyecto consiste en crear una versión portátil del juego "BATTLESHIP". El diseño utilizará el microcontrolador TM4C123GH6PM para ser el dispositivo encargado de manejar la lógica del juego, además, servirá para controlar los distintos periféricos conectados a él. Básicamente el juego consiste en determinar la posición de los barcos del adversario. El tablero tendrá 2 caras de cartón u otro material que tendrán 64 cuadros. Utilizando dos potenciómetros el jugador controlará el movimiento de dos servomotores (cada uno controla un eje) los cuales moverán un puntero láser. Cuando el jugador ubique el láser sobre la casilla deseada presionará un botón y en este momento el microcontrolador debe guardar dicha posición en memoria y compararla con la ubicación de los barcos del adversario. Si el jugador acierta en la ubicación del barco, se encenderá un led de color rojo al igual que un indicador sonoro. Si el jugador falla, se encenderá un led de color azul con su correspondiente indicador sonoro. Cuando el jugador haya hundido el barco, el led rojo parpadeará y se encenderá otro indicador sonoro.

## MATERIALES

1 microcontrolador TM4C123GH6PM (TIVAC)

2 servomotores

1 módulos bluetooth HC05 (la versión que puede ser maestro y esclavo)

2 potenciómetros de 100k

1 Buzzer Pasivo

## PROCOLO DE COMUNICACIÓN

El intercambio de datos entre los jugadores se realizará mediante el módulo bluetooth. El protocolo de comunicación del módulo al microcontrolador será de tipo UART. Los parámetros que todos los módulos deben tener son:

NOMBRE DEL DISPOSITIVO	Será el número de grupo.
PASSWORD	"SaleEnSemestre"
VELOCIDAD DEL UART	921600
BIT'S DE PARADA	0 (Indica 1 BIT en el comando AT)
BIT DE PARIDAD	0

Los parámetros que deben modificarse son:

TIPO DE ROL	Maestro o Esclavo (Debe ser lo contrario al del otro jugador)
MODO DE CONEXIÓN	Los jugadores deben intercambiar la ADDRESS de sus dispositivos y conectarse únicamente entre ellos

## SECUENCIA DE COMPARACIÓN DE DATOS





1. Jugador **X** envía la posición en la que considera que está el barco de **Y**
2. Controlador (TivaC) de **Y** compara dicha información con la posición de los barcos. Si jugador **X** acierta, controlador **Y** envía la letra "A". Si jugador **X** hundió el barco, controlador de **Y** envía "HB". Si jugador **X** falla, controlador **Y** envía "F". Si jugador **X** hundió todos los barcos, controlador **Y** envía "HTB".
3. Si controlador del jugador **X** recibe "A" o "HB" o "HTB" se debe encender un led rojo en la posición que **X** envió, pero para cada situación lo que cambiará es el sonido del buzzer. Si recibe "F", se debe encender un led de color azul.
4. Jugador **Y** envía la posición en la que considera que está el barco de **A** y se repite la secuencia 2.
5. Controlador **Y** compara todos los datos y ejecuta la misma lógica de la secuencia 4.

## ALMACENAMIENTO DE LA POSICIÓN DE BARCOS

Todos los jugadores deben tener 4 barcos. Uno de longitud de 5 posiciones, uno de 3 posiciones y dos de 2 posiciones. El jugador al iniciar el juego debe colocar primero sus barcos y lo hará en el orden establecido anteriormente (5, 3, 2, 2). El jugador moverá los servomotores (utilizando para ello los potenciómetros) y a su vez el indicador laser. La orientación se cambiará con un pulsador y se encenderá un led para indicar si se va a colocar el barco de forma horizontal y otro si se coloca de forma vertical. Al presionar el botón el jugador habrá colocado un barco en la posición indicada.

El controlador debe enviar una señal de error por el buzzer si el jugador coloca barcos en la misma posición o si estos comparten una posición en común.

 Representa el lugar donde el láser está enfocado.

	A	B	C	D	E	F	G	H
1								
2								
3								
4								
5								
6								
7								
8								

## RESTRICCIONES Y RECOMENDACIONES

- La programación del microcontrolador debe realizarse exclusivamente en Code Composer.
- No se permite el uso de partes del proyecto provenientes de otros grupos.
- Se sugiere el uso de elementos reciclables para la fabricación del proyecto.