**Laboratorio 7: Introducción a programación en C y MCU QE16**

**Preguntas:**

1. Que significa comunicación unidireccional?, ¿que sería comunicación bidireccional?

Cuando se tiene un solo canal unidireccional: el origen puede transmitir al destino pero el destino no puede comunicarse con el origen. Por ejemplo, la radio y la televisión. Es aquel en el que una estación siempre actúa como fuente y la otra siempre como colector, o sea se permite la transmisión de información en un único sentido.

En la comunicación bidireccional se permite la transmisión en ambas direcciones,

1. ¿Qué es un protocolo de comunicación síncrono y asíncrono?

En el síncrono hay un clock para regir la comunicación y en el asíncrono no hay clock de base para establecer la comunicación

1. ¿Qué es ​baud rate​?

Es la velocidad de transmisión de datos debido a la variación de la señal por segundo.

1. ¿Qué es ​bit rate​?

Es la razón de transmisión en bits por segundo

1. ¿Qué es código ASCII?

Es una convención internacional de la American Standard Code for Information Interchange en donde se establece una codificación de caracteres y símbolos fijos asignados a una numeración según una tabla convenida.

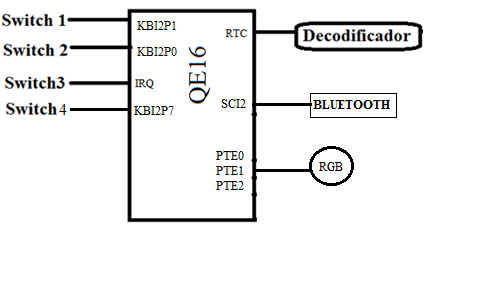
1. Realice un diagrama de flujo para un algoritmo que envíe el nombre en código ASCII de uno de los integrantes del grupo por medio del módulo serial a un receptor como un PC.

**Requerimientos:**

Se debe implementar una interfaz gráfica que permita hacer las siguientes tareas

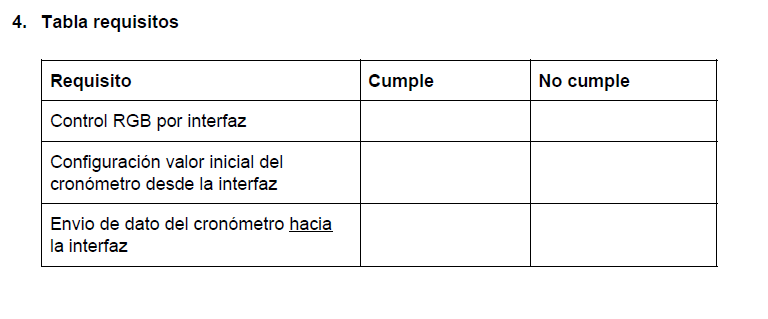
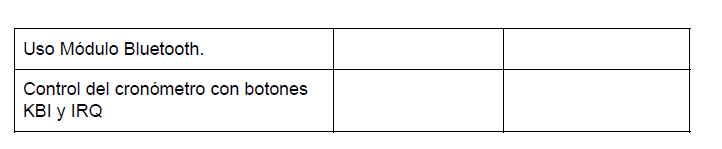
* R1 = Controlar la conexión o el emparejamiento a el MCU (a través del Bluetooth) con dos botones, uno para iniciar las comunicaciones y otro para detener las comunicaciones.
* R2 = Controlar un LED RGB: Con tres “botones” de la interfaz se debe controlar el encendido y apagado individual de cada color de un LED RGB.
* R3 = Se debe controlar un **cronómetro descendente** por medio de dos campos de texto donde se indique el tiempo desde donde iniciará este, se implementará un botón para enviar el dato a el MCU.
* R4 = Con el valor de partida ya configurado se debe poder controlar el cronómetro descendente con dos botones de KBI para inicio y parada tal como se hizo en la práctica anterior.
* R5 = Después de dejar correr el cronómetro y detenerlo en algún tiempo arbitrario, se deberá enviar el valor en el que se encuentre el cronómetro a el PC y este debe visualizarse en la interfaz.
* R6 = Un botón que resetee el cronómetro por interrupción IRQ.
* R7 = Como siempre el código debe ir debidamente documentado además de que deben emplearse técnicas de programación como el uso de etiquetas.
* R8 = Diagramas de flujo y esquemas de conexiones.

**Diagrama de concepto:**



Asignación de pines:

|  |  |  |
| --- | --- | --- |
| **PIN** | **Nombre** | **Funciones** |
| 4 | VDD | 3V3 |
| 9 | VSS | GND |
| 19 | PTC1 | ON.DG4-7SEG |
| 20 | PTC0 | ON.DG3-7SEG |
| 21 | PTB3 | ON.DG2-7SEG |
| 22 | PTB2 | ON.DG1-7SEG |
| 32 | PTA3 | D-7SEG |
| 33 | PTA2 | C-7SEG |
| 34 | PTA1 | B-7SEG |
| 35 | PTA0 | A-7SEG |
| 43 | SWICTH3 | IRQ (RESET-TIME) |
| 44 | BKGD | PROGRAM-LOADER |
| 1 | SWITCH1 | KBI2P1 (PLAY) |
| 2 | SWITCH2 | KBI2P0 (STOP) |
| 16 | SWITCH4 | KBI2P7(Envió dato del Cronometro a la interfaz) |
| 36 | ICS2-Tx | TxD2(Transmisor al Bluetooth) |
| 37 | ICS2-Rx | RxD2(Receptor al Bluetooth) |
| 38 | PTE2 | RGB-GREEN |
| 39 | PTE1 | RGB-BLUE |
| 40 | PTE0 | RGB-RED |



**Diseño de Hardware:**

Ver anexo.