



Lab 5: UART - Buffers y recepción de caracteres

1. Objetivo

En general, cuando se realiza comunicación entre dos dispositivos, es común esperar que esta sea de carácter bidireccional. En este sentido, la “bidireccionalidad” de la comunicación apunta a que los dispositivos sean capaces tanto de transmitir como de recibir información. Siguiendo la línea de protocolos de comunicación del laboratorio anterior, la presente experiencia tiene por objetivo el introducir al alumno a la recepción de datos en el protocolo USART, mediante el uso de estructuras complejas basadas en herramientas como la gestión de memoria (punteros) y *buffers*.

Al final de la experiencia, se espera que los estudiantes sean comprendan las funciones de Transmisión y Recepción del protocolo USART ¹, a la vez que también sean capaces de generar estructuras complejas, capaces de gestionar la información a transmitir o recibir.

2. Descripción de la actividad

Para realizar esta actividad se le ha proporcionado el archivo `USART_implement_me.c`, el cual deberá completar con el código faltante. El archivo en cuestión contiene dos nuevas funciones para su librería de comunicación USART, las cuales, en caso de ser completadas correctamente, le permitirán la recepción de un carácter o *char* (función 1) y la recepción de un *string* completo (función 2) de un largo conocido.

Una vez completado lo anterior, deberá enfrentarse al problema que ocurre cuando se envía un texto de un largo considerablemente mayor al tamaño del buffer implementado. La solución a implementar será el buffer circular o *ring buffer*, el que le permitirán guardar el final de su mensaje y poder desplegar este mensaje en la pantalla del terminal serial.

¹En la vida laboral, los protocolos de comunicaciones entre fabricantes funcionan o no funcionan. No hay términos medios



Figura 1: Buffer circular a implementar

Existe una gran cantidad de información disponible acerca de los *ring buffer* y cómo implementarlos. Estudie y comprenda estas estructuras, pues en caso contrario la actividad puede volverse compleja. Ponga atención a los punteros de *tail* y *head*, de forma de saber en todo momento la posición del último carácter recibido. Esta estructura tendrá un tamaño definido por usted (un largo 20 es aceptable), de forma que si se recibe por serial “Sistemas Electronicos Programables” (frase de 34 caracteres) y su buffer es de largo 20, en consola muestre “ronicos Programables”, -siendo este un string de largo 20 que corresponde a la última parte del mensaje.

Para esta actividad, **la elección del microcontrolador es libre, y solo es necesario realizarla en uno de ellos para cumplir con la aprobación.** No obstante, es altamente recomendable que lo implemente en ambos microcontroladores, teniendo en consideración los futuros laboratorios. En cuanto a los parámetros de comunicación serial, tales como el *baud rate*, *data bits*, *parity*, etc., estos quedan a libre elección, siempre y cuando le permitan observar un correcto funcionamiento de la tarea.

NOTA: Es importante notar que a la función de `USART_init()`, se le debe agregar una rutina de inicialización para la recepción de caracteres en USART. Esto asumiendo que no implementó esto antes.



3. Lectura recomendada

- [ATmega328/P Complete Datasheet](#).
- [MSP430x5xx and MSP430x6xx Family User's Guide](#).
- [MSP430F552x, MSP430F551x Mixed-Signal Microcontrollers datasheet](#).
- [Tutorial de C](#): material de utilidad para iniciarse en el mundo de la programación en C.

4. Pauta de Evaluación

4.1. Consideraciones generales

- El laboratorio será evaluado exclusivamente con nota 1.0 (**R**eprobado), 5.5 (**A**probado) y 7.0 (**D**istinguido). En ningún caso habrán notas intermedias.
- No se reciben trabajos después del módulo de presentación. Trabajos no entregados son calificados con nota 1.0 y son considerados dentro del criterio de aprobación del curso. La hora límite para inscribir a revisión es a las 10:20 hrs, posterior a esto se asignará una posición aleatoria.
- Cualquier consulta sobre los criterios de evaluación de cada laboratorio debe ser realizada en las [issues](#), donde estará disponible para que sea revisada por todos los alumnos.

4.2. Criterios de Aprobación

Se requiere cumplir con todos los puntos mencionados a continuación para poder aprobar. No existen casos excepcionales.

1. Funcionamiento de los requerimientos. El alumno realiza una presentación de su trabajo y se responsabiliza de exponer que su trabajo satisfaga todos los requerimientos mínimos solicitados en la *Descripción de la actividad*, los cuales incluyen en este laboratorio:
 - Implementación adecuada de las librerías de acuerdo a las especificaciones.
 - Debe idear cómo enfrentar cuando el mensaje a recibir supera al tamaño del *buffer* de recepción, existen muchísimas soluciones para eso, en esta ocasión deberán implementar un *ring buffer*



- **NO se aceptan soluciones del estilo de cortar el mensaje recibido.** Ejemplo, si se tiene un buffer de tamaño 13, y se está enviando el siguiente string ingenieria electrica, en el terminar serial se debe mostrar, ria electrica, en cambio, si se muestra ingenieria el se considerará incorrecto.
 - Se controlará este requisito también en el Laboratorio 6, razón por la cual puede resultar doblemente reprobatorio fallar en la implementación de este. Más aún, lograr este requisito en este Laboratorio trivializa significativamente el próximo.
 - **Si es que realiza otro tipo de implementación de buffer en reemplazo de la aquí señalada, no será considerado y reprobará la experiencia.**
 - **Debe comprender cada parte de su código, referenciar si corresponde y entender su solución.**
2. Preguntas. Se le realizarán diversas preguntas que abarcan (no exclusivamente) los siguientes temas. **Solo dispone de una oportunidad para la evaluación oral y será su ayudante corrector el que determinará si su nivel de conocimientos es el adecuado para esta experiencia.**
- ¿Qué es una *ring buffer*?, ¿Cómo se implementa?
 - ¿Cuáles son las ventajas y desventajas de este tipo de solución con respecto a otras (buffer lineal, linked list) ?
 - Protocolo UART en su totalidad.



4.3. Criterios de Distinción

La distinción representa un trabajo adicional que sobresale a los requerimientos mínimos para la aprobación. Agregados adicionales no constituyen por sí mismo una distinción si no representan un verdadero trabajo adicional de comprensión y/o análisis.

Los trabajos distinguidos pueden caer (no exclusivamente) en algunas de las siguientes líneas generales:

- Funcionalidades creativas :D
- Trabajos sobresalientes en la línea de *timers*, I2C, SPI u otras funcionalidades no vistas.
- Comunicación bidireccional entre microcontroladores: en este caso tenga especial cuidado con las salidas de alto voltaje de ATmega328P, pues son de 5V y si los conecta directamente a MSP430F5529 **PODRÍA QUEMAR LA TARJETA**. (En caso de comunicar, debe demostrar de alguna forma visual que la comunicación se está realizando).
- Creación de un chat entre computadores utilizando al menos dos microcontroladores como intermediario (en este caso, ambos alumnos reciben su distinguido).
- Interrupciones de Tx y Rx: en este caso, las rutinas de interrupción deben ser hechas a conciencia y conociendo a cabalidad su funcionamiento. (Además, podría adelantar trabajo para laboratorios futuros). El uso de interrupciones sin comprensión no calificará como distinguido.

Para optar a distinguido en este laboratorio debe realizar solo una idea, pudiendo ser esta en el microcontrolador escogido o en el no utilizado. No se considerarán distinguidos a quienes repitan ideas utilizadas en laboratorios anteriores o realización del laboratorio en ambos microcontroladores.

Las Distinciones son discutidas caso a caso por la totalidad del equipo de ayudantes al finalizar la corrección del laboratorio. Serán notificadas públicamente después del módulo de evaluación.