Alan García - 5008 HW
STACKS AND QUEUES

**Circular queues**

Circular queues are different from regular queues because they implement two indexing values (back and front) and a system that defines a dynamic start and end to the queue. This allows the implementation to be more space efficient and use all available space without the need to shift or move data.

Embedded systems usually employ circular buffers in a communication protocol called **Universal Asynchronous Receiver-Transmitter** or **UART**s. Where two devices interchange data continuously and this data needs to be quickly stored.They need to split time critical data with non time critical data. In other words, the stream of bytes could be same in the buffer to momentarily save data that takes time to process. We can use circular queues, when space and time is important, but we don't mind too much if data gets overridden.

"Circular buffer is extremely useful in buffering serial communication data for some reasons: it makes possible avoid data loss when processing the received data; permits memory to be reusable; is an array that wraps around – where it overwrites old data in the buffer if the microcontroller processing speed is unable to momentarily keep up with the incoming data rate;" - *Instituto Federal de Santa Catalina*

Resources:

https://stackoverflow.com/questions/2553637/what-are-the-uses-of-circular-buffer
https://wiki.sj.ifsc.edu.br/index.php/ESTE:_UART_-_Interrupts_and_Circular_Buffer
Spanish:
https://www.rohde-schwarz.com/lat/productos/prueba-y-medicion/osciloscopios/educational-content/entendiendo-el-uart_254524.html

Alan García - 5008 HW
STACKS AND QUEUES

## STACKS

Stack is a widely used abstract data type (ADT), in fact it's so widely used that almost literally every language has some sort of function stack. Where the last function to be called is the first one, generally the process of assigning and freeing memory uses this ADT. Additionally, compilers use stacks to define the hierarchy of operations in a line of code or mathematical expression.

Lastly, other common uses are for backtracking problems where you need to start from the very end to the desired state, undoing processes (CTRL + Z) and any situation where the last in is the last out. Stacks are extremely common and useful, we should always consider using them when we are storing data linearly.

Resources

https://www.faceprep.in/data-structures/stack-applications-in-data-structure/