

PRACTICA 8 : Buses de comunicación IV (uart)

Bucle de comunicacion UART2

Código

```
#include <Arduino.h>

void setup() {
  Serial.begin(9600);
  Serial2.begin(9600);
  Serial.write("Inicializando...");
}

void loop() {
  if(Serial.available()){
    Serial2.write(Serial.read());
    delay(10);
    if(Serial2.available()){
      Serial.write(Serial2.read());
    }
  }
}
```

Funcionamiento del programa

En primer lugar, vamos a definir nuestra función Setup. Primero vamos a crear un primer objeto de la clase Serial y lo configuramos a un baut rate de 9600. Seguidamente creamos un segundo objeto idéntico al primero al que llamaremos "Serial2".

```
#include <Arduino.h>

void setup() {
  Serial.begin(9600);
  Serial2.begin(9600);
  Serial.write("Inicializando...");
}
```

Dentro de la función Loop, en primer lugar vamos a comprobar con la funcion "available()" si nuestro primer objeto "Serial" tiene alguna información disponible para leer desde el serial port de UART0. Entonces ahora podemos escribir cualquier información y esta se guardaría en el buffer. Esa información que se guarde en el buffer la vamos a escribir en el UART2 con la función "write()". Ahora procedemos a hacer lo mismo pero al revés, aplicamos la función "available()" en nuestro segundo objeto "Serial2" y vamos a copiar la información almacenada en el buffer en el UART0 mediante la función "write()".

```
void loop() {  
  if(Serial.available()){  
    Serial2.write(Serial.read());  
    delay(10);  
    if(Serial2.available()){  
      Serial.write(Serial2.read());  
    }  
  }  
}
```

Salidas y entradas de Consola

En este caso, la salida y la entrada de la consola coinciden

Salidas:

Inicializando... Practica 8 Bucle de comunicación UART2.

Entradas:

Practica 8 Bucle de comunicación UART2.

El texto podría ser cualquiera, ya que podemos escribir cualquier información y se guardaría en el buffer.