

Tienda PJRC

- [Teensy 4.1, \\$31.50](#)
- [Teensy 4.0, \\$23.80](#)

Teensy

- [Página principal](#)
- ▣ [Hardware Tutorial](#)
- ▣ [de introducción](#)
- [Consejos prácticos](#)
- Biblioteca de códigos
- ▣
- ▣
-
- [Mensaje de depuración USB](#)
- [Teclado USB](#)
- [Ratón USB](#)
- [Serie USB Serie](#)
- [HID sin procesar](#)
-
- [Proyectos](#)
- ▣ [Teensyduino](#)
- ▣ [Referencia](#)

UART: Puerto serie

Este código implementa funciones básicas para utilizar el UART como puerto serie.

Descargar archivos fuente

- [UART, versión 1.1](#) .

Ejemplo de aplicación

Un ejemplo muy sencillo imprime "Ejemplo UART" al iniciar y espera las pulsaciones de teclas. Cada byte se devuelve con un mensaje corto.

TODO: captura de pantalla

Funciones

```
#include <uart.h>
```

uart_disponible()

¿Cuántos caracteres hay esperando en el búfer de recepción?

Esta función devuelve el número de bytes almacenados en el búfer, o 0 si no hay nada en el búfer de recepción. Si su aplicación debe realizar otras funciones mientras espera datos, o si desea implementar tiempos de espera, debe llamarse antes de `uart_getchar()` para comprobar si hay datos disponibles.

uart_getchar()

Recibir un personaje.

Se devuelve el siguiente byte (de 0 a 255). Si no hay datos disponibles, esta función espera la llegada de un byte.

uart_putchar(carácter)

Transmitir un solo caracter.

Se utiliza un búfer de transmisión, por lo que esta función retorna rápidamente si el byte se puede almacenar en el búfer. De lo contrario, espera a que haya espacio disponible en el búfer.

uart_init(baud)

Inicializar el controlador UART a una velocidad en baudios particular.

Ejemplo de uso

```
int principal(vacio)
{
    uint8_t c;

    CPU_PRESCALE(0); // ejecutar a 16 MHz
    uart_init(TASA_EN_BAUDIOS);
    uart_print("Ejemplo de UART\r\n");
    mientras (1) {
        si (uart_disponible()) {
            c = uart_getchar();
            uart_print("Byte: ");
            uart_putchar(c);
            uart_putchar('\r');
            uart_putchar('\n');
        }
    }
}
```