TC_Config.cpp File Reference

Source code for a library that implements the timer/counter configuration. More...

```
#include "TC_Config.h"
#include <avr/io.h>
#include <stdint.h>
#include "IO_Config.h"
#include "Pin.h"
```

Functions

void	**TC_Init** () Sets up the timer configuration. More...
uint32_t	**micros** () Returns the system time. More...
void	**analogWrite** (**Pin** pin, uint8_t dutyCycle) Returns the system time. More...

Detailed Description

Source code for a library that implements the timer/counter configuration.

This library allows the user to configure the timers/counters in one place. System time is retrieved using the **micros()** function, meant to simulate the function on the Arduino but with 1us precision. PWM duty cycle is written on Timer E0 using the **analogWrite()** function, meant to simulate the function on the Arduino. Future updates will allow for the timer and event channel to be chosen independently by the user via parameters in the constructor.

Author

KC Egger, Rahul Goyal, Alexandros Petrakis

Date

2019-12-09

Function Documentation

◆ analogWrite()

void analogWrite (**Pin** pin,
uint8_t dutyCycle
)

Returns the system time.

This function concatenates the values of two 16-bit timers to return the system time as a 32-bit timer.

Parameters

pin The pin to write the PWM duty cycle to.

dutyCycle The duty cycle to write, between -100 and 100.

◆ micros()

uint32_t micros ( )

Returns the system time.

This function concatenates the values of two 16-bit timers to return the system time as a 32-bit timer.

Returns

The system time in microseconds as an unsigned 32-bit number.

◆ TC_Init()

void TC_Init ( )

Sets up the timer configuration.

This function sets up the system clock timer, timer interrupt, encoder counters, and PWM output.