ESP8266 Libraries

|  |  |
| --- | --- |
| Nefastor Online libraries documentation | 03/08/2016 |

DOCUMENT STATUS : DRAFT IN PROGRESS – UNFIT FOR USE

This document and the code it pertains to are stored in the GitHub repository Nefastor/ESP8266, where later versions may be found.

This document assumes the libraries are used as part of the template project stored in the same repository.

For more information, head over to [www.nefastor.com](http://www.nefastor.com).

# GPIO

## Introduction

This library provides an API for General Purpose Input Output (GPIO) pins.

### State of the library

This library is currently usable but not fully validated.

### Origin of the library

The GPIO library originates with the official Espressif SDK. Several source files were merged to provide a single library for controlling the ESP8266 I/O pins. This includes:

* Espressif’s “gpio.h” and “gpio.c” files
* Espressif’s “gpio16.h” and “gpio16.c” files

GPIO 16 is very different form GPIO 0 through 15 in that it is actually part of a different internal peripheral and is controlled through different registers. It also has reduced functionality, and notably lacks interrupt support. I consider

Relatively few modifications to the actual code were necessary. The most important modification is FreeRTOS support.

## Using the library

### Building

By default, the library’s source code is part of the ESP8266 template project and will compile and link with your code. All that may be required for building is to #include “gpio.h” in your own source files.

### Using GPIO

Typical GPIO programming requires an initialization step.

## API

### gpio\_config

### gpio\_output\_conf

### gpio\_input\_get

Returns the current state of pins GPIO 0 to 15 as an integer.

You will need to use bit masks to get the value of individual pins or sets of pins.

### gpio\_intr\_handler\_register

Status: validated.

### gpio\_pin\_wakeup\_enable

Status: untested.

### gpio\_pin\_wakeup\_disable

### gpio\_pin\_intr\_state\_set

### gpio16\_output\_conf

### gpio16\_output\_set

### gpio16\_input\_conf

### gpio16\_input\_get

# SPI (HSPI)

## Introduction

The ESP8266 has two independent SPI (Serial Peripheral Interface). One is used to connect an external Flash memory for program storage, the other is the main board-level interface between ESP8266 and hardware. It is through this interface that the most WiFi bandwidth can be exploited.

The second interface is called “HSPI” (Hardware SPI ?)

### State of the library

This library is currently usable but not fully validated.

It is missing mechanisms for reading from SPI.

### Origin of the library

The source files for this library come from an example project in the Unofficial SDK and may be related to a project by someone called “Metal Phreak”. This information will be updated once the origin of the code has been ascertained.

This original library counts two files : “hspi.h” and “hspi.c”. It was originally intended for Espressif’s own RTOS.

Nefastor added several modifications to make it compatible with FreeRTOS. This includes the creation of a second header, “spi\_registers\_2.h”, out of Espressif code. This header is used internally by the library and does not need to be included in source files using the HSPI library.

## Using the library

## Library API

### hspi\_init

### hspi\_wait\_ready

### hspi\_prepare\_tx

### hspi\_start\_tx

### hspi\_send\_uint8

### hspi\_send\_uint16

### hspi\_send\_uint32

### hspi\_send\_data

### hspi\_send\_uint16\_r

# ILI9341 SPI LCD

## Introduction

The ILI9341 is an inexpensive LCD driver for displays with 320 x 240 pixels. It supports 18-bit color. It is commonly found on cheap LCD modules.

This library is primarily intended for displaying text and numbers, but will be expanded with graphical functions in future versions.

### State of the library

This library is currently usable but not fully validated.

### Origin of the library

## Using the library

## Library API