

Raspberry Gpio Peripheral Driver

Generated by Doxygen 1.8.9.1

Wed Aug 19 2015 01:07:34

Contents

1	Class Index	1
1.1	Class List	1
2	File Index	1
2.1	File List	1
3	Class Documentation	1
3.1	GpioRPi Class Reference	2
3.1.1	Detailed Description	2
3.1.2	Member Function Documentation	2
3.1.3	Member Data Documentation	3
4	File Documentation	3
4.1	Gpio.cpp File Reference	3
4.1.1	Variable Documentation	3
4.2	Gpio.cpp	4
4.3	Gpio.h File Reference	4
4.3.1	Macro Definition Documentation	6
4.3.2	Variable Documentation	8
4.4	Gpio.h	8
	Index	11

1 Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

GpioRPi	2
-------------------------	---

2 File Index

2.1 File List

Here is a list of all files with brief descriptions:

Gpio.cpp	3
Gpio.h	4

3 Class Documentation

3.1 GpioRPi Class Reference

```
#include <Gpio.h>
```

Public Member Functions

- void [begin](#) ()
- void [stop](#) ()
- void [pinMode](#) (unsigned char pin, bool mode)
- void [digitalWrite](#) (unsigned char pin, bool value)
- bool [digitalRead](#) (unsigned char pin)
- void [setPin](#) (unsigned char pin)
- void [clearPin](#) (unsigned char pin)

Private Member Functions

- void [configureBits](#) (volatile unsigned int *address, unsigned int value, unsigned int mask)
- unsigned int [safeRead](#) (volatile unsigned int *address)
- void [safeWrite](#) (volatile unsigned int *address, unsigned int value)

Private Attributes

- Bcm2835::Peripheral [gpio](#)

3.1.1 Detailed Description

Definition at line 59 of file [Gpio.h](#).

3.1.2 Member Function Documentation

3.1.2.1 void GpioRPi::begin ()

Definition at line 4 of file [Gpio.cpp](#).

3.1.2.2 void GpioRPi::clearPin (unsigned char *pin*)

Definition at line 30 of file [Gpio.cpp](#).

3.1.2.3 void GpioRPi::configureBits (volatile unsigned int * *address*, unsigned int *value*, unsigned int *mask*) [private]

Definition at line 42 of file [Gpio.cpp](#).

3.1.2.4 bool GpioRPi::digitalRead (unsigned char *pin*)

Definition at line 19 of file [Gpio.cpp](#).

3.1.2.5 void GpioRPi::digitalWrite (unsigned char *pin*, bool *value*)

Definition at line 13 of file [Gpio.cpp](#).

3.1.2.6 void GpioRPi::pinMode (unsigned char *pin*, bool *mode*)

Definition at line 34 of file [Gpio.cpp](#).

3.1.2.7 `unsigned int GpioRPI::safeRead (volatile unsigned int * address) [private]`

Definition at line 48 of file [Gpio.cpp](#).

3.1.2.8 `void GpioRPI::safeWrite (volatile unsigned int * address, unsigned int value) [private]`

Definition at line 56 of file [Gpio.cpp](#).

3.1.2.9 `void GpioRPI::setPin (unsigned char pin)`

Definition at line 26 of file [Gpio.cpp](#).

3.1.2.10 `void GpioRPI::stop ()`

Definition at line 9 of file [Gpio.cpp](#).

3.1.3 Member Data Documentation

3.1.3.1 `Bcm2835::Peripheral GpioRPI::gpio [private]`

Definition at line 61 of file [Gpio.h](#).

The documentation for this class was generated from the following files:

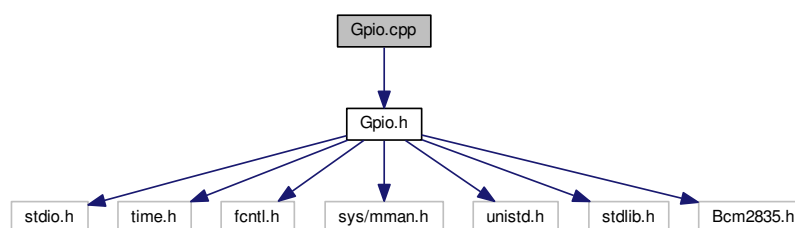
- [Gpio.h](#)
- [Gpio.cpp](#)

4 File Documentation

4.1 Gpio.cpp File Reference

```
#include "Gpio.h"
```

Include dependency graph for Gpio.cpp:



Variables

- [GpioRPI Gpio](#)

4.1.1 Variable Documentation

4.1.1.1 GpioRPI Gpio

Definition at line 63 of file [Gpio.cpp](#).

4.2 Gpio.cpp

```

00001
00002 #include "Gpio.h"
00003
00004 void GpioRpi::begin() {
00005     gpio.address = GPIO_ADDRESS;
00006     Bcm2835::mapPeripheral(&gpio);
00007 }
00008
00009 void GpioRpi::stop() {
00010     Bcm2835::unmapPeripheral(&gpio);
00011 }
00012
00013 void GpioRpi::digitalWrite(unsigned char pin, bool value) {
00014     volatile unsigned int* address = (unsigned int*)gpio.mem + ((value) ?
BCM2835_GPSET0 : BCM2835_GPCLR0) / 4 + pin / 32;
00015     unsigned char shift = pin % 32;
00016     safeWrite(address, 0x000001 << shift);
00017 }
00018
00019 bool GpioRpi::digitalRead(unsigned char pin) {
00020     volatile unsigned int* address = (unsigned int*)gpio.mem +
BCM2835_GPLEV0 / 4 + pin / 32;
00021     unsigned char shift = pin % 32;
00022     unsigned int v = safeRead(address);
00023     return (v & (1 << shift)) ? HIGH : LOW;
00024 }
00025
00026 void GpioRpi::setPin(unsigned char pin) {
00027     digitalWrite(pin, HIGH)
00028 }
00029
00030 void GpioRpi::clearPin(unsigned char pin) {
00031     digitalWrite(pin, LOW)
00032 }
00033
00034 void GpioRpi::pinMode(unsigned char pin, bool mode) {
00035     volatile unsigned int* address = (unsigned int*)gpio.mem +
BCM2835_GPFSEL0 / 4 + (pin / 10);
00036     unsigned char shift = (pin % 10) * 3;
00037     unsigned int mask = BCM2835_FSEL_MASK << shift;
00038     unsigned int value = (mode) ? (0x000001 << shift) : 0x000000;
00039     configureBits(address, value, mask);
00040 }
00041
00042 void GpioRpi::configureBits(volatile unsigned int* address, unsigned int value,
unsigned int mask) {
00043     unsigned int v = safeRead(address);
00044     v = (v & ~mask) | (value & mask);
00045     safeWrite(address, v);
00046 }
00047
00048 unsigned int GpioRpi::safeRead(volatile unsigned int* address) {
00049     unsigned int i = *address;
00050     #if USE_RW_BARRIER == 1
00051         unsigned int dummy = *address;
00052     #endif
00053     return i;
00054 }
00055
00056 void GpioRpi::safeWrite(volatile unsigned int* address, unsigned int value) {
00057     #if USE_RW_BARRIER == 1
00058         *address = value;
00059     #endif
00060     *address = value;
00061 }
00062
00063 GpioRpi Gpio;

```

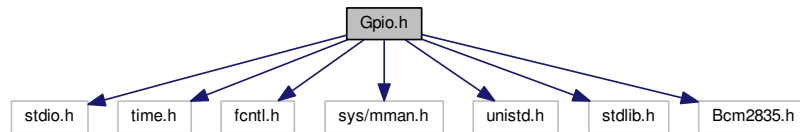
4.3 Gpio.h File Reference

```

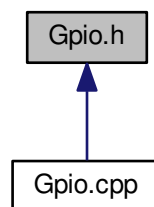
#include <stdio.h>
#include <time.h>
#include <fcntl.h>
#include <sys/mman.h>
#include <unistd.h>
#include <stdlib.h>
#include <Bcm2835.h>

```

Include dependency graph for Gpio.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [GpioRPi](#)

Macros

- `#define` [GPIO_ADDRESS](#) 0x200000
- `#define` [INPUT](#) 0
- `#define` [OUTPUT](#) 1
- `#define` [LOW](#) 0
- `#define` [HIGH](#) 1
- `#define` [USE_RW_BARRIER](#) 0
- `#define` [BCM2835_FSEL_MASK](#) 0x000007
- `#define` [BCM2835_GPFSEL0](#) 0x000000
- `#define` [BCM2835_GPFSEL1](#) 0x000004
- `#define` [BCM2835_GPFSEL2](#) 0x000008
- `#define` [BCM2835_GPFSEL3](#) 0x00000c
- `#define` [BCM2835_GPFSEL4](#) 0x000010
- `#define` [BCM2835_GPFSEL5](#) 0x000014
- `#define` [BCM2835_GPSET0](#) 0x00001c
- `#define` [BCM2835_GPSET1](#) 0x000020
- `#define` [BCM2835_GPCLR0](#) 0x000028
- `#define` [BCM2835_GPCLR1](#) 0x00002c
- `#define` [BCM2835_GPLEV0](#) 0x000034
- `#define` [BCM2835_GPLEV1](#) 0x000038
- `#define` [BCM2835_GPEDS0](#) 0x000040

- `#define BCM2835_GPEDS1 0x000044`
- `#define BCM2835_GPREN0 0x00004c`
- `#define BCM2835_GPREN1 0x000050`
- `#define BCM2835_GPFEN0 0x000048`
- `#define BCM2835_GPFEN1 0x00005c`
- `#define BCM2835_GPHEN0 0x000064`
- `#define BCM2835_GPHEN1 0x000068`
- `#define BCM2835_GPLEN0 0x000070`
- `#define BCM2835_GPLEN1 0x000074`
- `#define BCM2835_GPAREN0 0x00007c`
- `#define BCM2835_GPAREN1 0x000080`
- `#define BCM2835_GPAFEN0 0x000088`
- `#define BCM2835_GPAFEN1 0x00008c`
- `#define BCM2835_GPPUD 0x000094`
- `#define BCM2835_GPPUDCLK0 0x000098`
- `#define BCM2835_GPPUDCLK1 0x00009c`

Variables

- [GpioRPi Gpio](#)

4.3.1 Macro Definition Documentation

4.3.1.1 `#define BCM2835_FSEL_MASK 0x000007`

Definition at line 28 of file [Gpio.h](#).

4.3.1.2 `#define BCM2835_GPAFEN0 0x000088`

Definition at line 53 of file [Gpio.h](#).

4.3.1.3 `#define BCM2835_GPAFEN1 0x00008c`

Definition at line 54 of file [Gpio.h](#).

4.3.1.4 `#define BCM2835_GPAREN0 0x00007c`

Definition at line 51 of file [Gpio.h](#).

4.3.1.5 `#define BCM2835_GPAREN1 0x000080`

Definition at line 52 of file [Gpio.h](#).

4.3.1.6 `#define BCM2835_GPCLR0 0x000028`

Definition at line 37 of file [Gpio.h](#).

4.3.1.7 `#define BCM2835_GPCLR1 0x00002c`

Definition at line 38 of file [Gpio.h](#).

4.3.1.8 `#define BCM2835_GPEDS0 0x000040`

Definition at line 41 of file [Gpio.h](#).

4.3.1.9 `#define BCM2835_GPEDS1 0x000044`

Definition at line 42 of file [Gpio.h](#).

4.3.1.10 `#define BCM2835_GPFEN0 0x000048`

Definition at line 45 of file [Gpio.h](#).

4.3.1.11 `#define BCM2835_GPFEN1 0x00005c`

Definition at line 46 of file [Gpio.h](#).

4.3.1.12 `#define BCM2835_GPFSEL0 0x000000`

Definition at line 29 of file [Gpio.h](#).

4.3.1.13 `#define BCM2835_GPFSEL1 0x000004`

Definition at line 30 of file [Gpio.h](#).

4.3.1.14 `#define BCM2835_GPFSEL2 0x000008`

Definition at line 31 of file [Gpio.h](#).

4.3.1.15 `#define BCM2835_GPFSEL3 0x00000c`

Definition at line 32 of file [Gpio.h](#).

4.3.1.16 `#define BCM2835_GPFSEL4 0x000010`

Definition at line 33 of file [Gpio.h](#).

4.3.1.17 `#define BCM2835_GPFSEL5 0x000014`

Definition at line 34 of file [Gpio.h](#).

4.3.1.18 `#define BCM2835_GPHEN0 0x000064`

Definition at line 47 of file [Gpio.h](#).

4.3.1.19 `#define BCM2835_GPHEN1 0x000068`

Definition at line 48 of file [Gpio.h](#).

4.3.1.20 `#define BCM2835_GPLEN0 0x000070`

Definition at line 49 of file [Gpio.h](#).

4.3.1.21 `#define BCM2835_GPLEN1 0x000074`

Definition at line 50 of file [Gpio.h](#).

4.3.1.22 `#define BCM2835_GPLEV0 0x000034`

Definition at line 39 of file [Gpio.h](#).

4.3.1.23 `#define BCM2835_GPLEV1 0x000038`

Definition at line 40 of file [Gpio.h](#).

4.3.1.24 `#define BCM2835_GPPUD 0x000094`

Definition at line 55 of file [Gpio.h](#).

4.3.1.25 `#define BCM2835_GPPUDCLK0 0x000098`

Definition at line 56 of file [Gpio.h](#).

4.3.1.26 `#define BCM2835_GPPUDCLK1 0x00009c`

Definition at line 57 of file [Gpio.h](#).

4.3.1.27 `#define BCM2835_GPREN0 0x00004c`

Definition at line 43 of file [Gpio.h](#).

4.3.1.28 `#define BCM2835_GPREN1 0x000050`

Definition at line 44 of file [Gpio.h](#).

4.3.1.29 `#define BCM2835_GPSET0 0x00001c`

Definition at line 35 of file [Gpio.h](#).

4.3.1.30 `#define BCM2835_GPSET1 0x000020`

Definition at line 36 of file [Gpio.h](#).

4.3.1.31 `#define GPIO_ADDRESS 0x200000`

Definition at line 18 of file [Gpio.h](#).

4.3.1.32 `#define HIGH 1`

Definition at line 23 of file [Gpio.h](#).

4.3.1.33 `#define INPUT 0`

Definition at line 20 of file [Gpio.h](#).

4.3.1.34 `#define LOW 0`

Definition at line 22 of file [Gpio.h](#).

4.3.1.35 `#define OUTPUT 1`

Definition at line 21 of file [Gpio.h](#).

4.3.1.36 `#define USE_RW_BARRIER 0`

Definition at line 25 of file [Gpio.h](#).

4.3.2 Variable Documentation

4.3.2.1 GpioRPi Gpio

Definition at line 63 of file [Gpio.cpp](#).

4.4 Gpio.h

```
00001
00006 #ifndef __RASPBERRY_GPIO_H__
00007 #define __RASPBERRY_GPIO_H__ 1
00008
00009 #include <stdio.h>
00010 #include <time.h>
```

```

00011 #include <fcntl.h>
00012 #include <sys/mman.h>
00013 #include <unistd.h>
00014 #include <stdlib.h>
00015
00016 #include <Bcm2835.h>
00017
00018 #define GPIO_ADDRESS          0x200000
00019
00020 #define INPUT                  0
00021 #define OUTPUT                 1
00022 #define LOW                    0
00023 #define HIGH                   1
00024
00025 #define USE_RW_BARRIER      0
00026
00027 /* FSEL */
00028 #define BCM2835_FSEL_MASK    0x000007 // Function select bits mask
00029 #define BCM2835_GPFSEL0      0x000000 // GPIO Function Select 0
00030 #define BCM2835_GPFSEL1      0x000004 // GPIO Function Select 1
00031 #define BCM2835_GPFSEL2      0x000008 // GPIO Function Select 2
00032 #define BCM2835_GPFSEL3      0x00000c // GPIO Function Select 3
00033 #define BCM2835_GPFSEL4      0x000010 // GPIO Function Select 4
00034 #define BCM2835_GPFSEL5      0x000014 // GPIO Function Select 5
00035 #define BCM2835_GPSET0        0x00001c // GPIO Pin Output Set 0
00036 #define BCM2835_GPSET1        0x000020 // GPIO Pin Output Set 1
00037 #define BCM2835_GPCLR0        0x000028 // GPIO Pin Output Clear 0
00038 #define BCM2835_GPCLR1        0x00002c // GPIO Pin Output Clear 1
00039 #define BCM2835_GPLEV0        0x000034 // GPIO Pin Level 0
00040 #define BCM2835_GPLEV1        0x000038 // GPIO Pin Level 1
00041 #define BCM2835_GPEDS0        0x000040 // GPIO Pin Event Detect Status 0
00042 #define BCM2835_GPEDS1        0x000044 // GPIO Pin Event Detect Status 1
00043 #define BCM2835_GPREN0        0x00004c // GPIO Pin Rising Edge Detect Enable 0
00044 #define BCM2835_GPREN1        0x000050 // GPIO Pin Rising Edge Detect Enable 1
00045 #define BCM2835_GPFEN0        0x000048 // GPIO Pin Falling Edge Detect Enable 0
00046 #define BCM2835_GPFEN1        0x00005c // GPIO Pin Falling Edge Detect Enable 1
00047 #define BCM2835_GPHEN0        0x000064 // GPIO Pin High Detect Enable 0
00048 #define BCM2835_GPHEN1        0x000068 // GPIO Pin High Detect Enable 1
00049 #define BCM2835_GPLEN0        0x000070 // GPIO Pin Low Detect Enable 0
00050 #define BCM2835_GPLEN1        0x000074 // GPIO Pin Low Detect Enable 1
00051 #define BCM2835_GPAREN0        0x00007c // GPIO Pin Async. Rising Edge Detect 0
00052 #define BCM2835_GPAREN1        0x000080 // GPIO Pin Async. Rising Edge Detect 1
00053 #define BCM2835_GPAFEN0        0x000088 // GPIO Pin Async. Falling Edge Detect 0
00054 #define BCM2835_GPAFEN1        0x00008c // GPIO Pin Async. Falling Edge Detect 1
00055 #define BCM2835_GPPUD          0x000094 // GPIO Pin Pull-up/down Enable
00056 #define BCM2835_GPPUDCLK0      0x000098 // GPIO Pin Pull-up/down Enable Clock 0
00057 #define BCM2835_GPPUDCLK1      0x00009c // GPIO Pin Pull-up/down Enable Clock 1
00058
00059 class GpioRPi {
00060
00061     Bcm2835::Peripheral gpio;
00062
00063 public:
00064
00065     void begin();
00066
00067     void stop();
00068
00069     void pinMode(unsigned char pin, bool mode);
00070
00071     void digitalWrite(unsigned char pin, bool value);
00072
00073     bool digitalRead(unsigned char pin);
00074
00075     void setPin(unsigned char pin);
00076
00077     void clearPin(unsigned char pin);
00078
00079 private:
00080
00081     void configureBits(volatile unsigned int* address, unsigned int value, unsigned int mask);
00082
00083     unsigned int safeRead(volatile unsigned int* address);
00084
00085     void safeWrite(volatile unsigned int* address, unsigned int value);
00086 };
00087
00088 extern GpioRPi gpio;
00089
00090 #endif /* __RASPBERRY_GPIO_H__ */

```


Index

BCM2835_FSEL_MASK
 Gpio.h, [6](#)
BCM2835_GPAFEN0
 Gpio.h, [6](#)
BCM2835_GPAFEN1
 Gpio.h, [6](#)
BCM2835_GPAREN0
 Gpio.h, [6](#)
BCM2835_GPAREN1
 Gpio.h, [6](#)
BCM2835_GPCLR0
 Gpio.h, [6](#)
BCM2835_GPCLR1
 Gpio.h, [6](#)
BCM2835_GPEDS0
 Gpio.h, [6](#)
BCM2835_GPEDS1
 Gpio.h, [6](#)
BCM2835_GPFEN0
 Gpio.h, [6](#)
BCM2835_GPFEN1
 Gpio.h, [7](#)
BCM2835_GPFSEL0
 Gpio.h, [7](#)
BCM2835_GPFSEL1
 Gpio.h, [7](#)
BCM2835_GPFSEL2
 Gpio.h, [7](#)
BCM2835_GPFSEL3
 Gpio.h, [7](#)
BCM2835_GPFSEL4
 Gpio.h, [7](#)
BCM2835_GPFSEL5
 Gpio.h, [7](#)
BCM2835_GPHEN0
 Gpio.h, [7](#)
BCM2835_GPHEN1
 Gpio.h, [7](#)
BCM2835_GPLEN0
 Gpio.h, [7](#)
BCM2835_GPLEN1
 Gpio.h, [7](#)
BCM2835_GPLEV0
 Gpio.h, [7](#)
BCM2835_GPLEV1
 Gpio.h, [7](#)
BCM2835_GPPUD
 Gpio.h, [7](#)
BCM2835_GPPUDCLK0
 Gpio.h, [7](#)
BCM2835_GPPUDCLK1
 Gpio.h, [8](#)
BCM2835_GPREN0
 Gpio.h, [8](#)
BCM2835_GPREN1

 Gpio.h, [8](#)
BCM2835_GPSET0
 Gpio.h, [8](#)
BCM2835_GPSET1
 Gpio.h, [8](#)
begin
 GpioRPi, [2](#)

clearPin
 GpioRPi, [2](#)
configureBits
 GpioRPi, [2](#)

digitalRead
 GpioRPi, [2](#)
digitalWrite
 GpioRPi, [2](#)

GPIO_ADDRESS
 Gpio.h, [8](#)
Gpio
 Gpio.cpp, [3](#)
 Gpio.h, [8](#)
gpio
 GpioRPi, [3](#)
Gpio.cpp, [3](#)
 Gpio, [3](#)
Gpio.h, [4](#)
 BCM2835_FSEL_MASK, [6](#)
 BCM2835_GPAFEN0, [6](#)
 BCM2835_GPAFEN1, [6](#)
 BCM2835_GPAREN0, [6](#)
 BCM2835_GPAREN1, [6](#)
 BCM2835_GPCLR0, [6](#)
 BCM2835_GPCLR1, [6](#)
 BCM2835_GPEDS0, [6](#)
 BCM2835_GPEDS1, [6](#)
 BCM2835_GPFEN0, [6](#)
 BCM2835_GPFEN1, [7](#)
 BCM2835_GPFSEL0, [7](#)
 BCM2835_GPFSEL1, [7](#)
 BCM2835_GPFSEL2, [7](#)
 BCM2835_GPFSEL3, [7](#)
 BCM2835_GPFSEL4, [7](#)
 BCM2835_GPFSEL5, [7](#)
 BCM2835_GPHEN0, [7](#)
 BCM2835_GPHEN1, [7](#)
 BCM2835_GPLEN0, [7](#)
 BCM2835_GPLEN1, [7](#)
 BCM2835_GPLEV0, [7](#)
 BCM2835_GPLEV1, [7](#)
 BCM2835_GPPUD, [7](#)
 BCM2835_GPPUDCLK0, [7](#)
 BCM2835_GPPUDCLK1, [8](#)
 BCM2835_GPREN0, [8](#)
 BCM2835_GPREN1, [8](#)

- BCM2835_GPSET0, [8](#)
- BCM2835_GPSET1, [8](#)
- GPIO_ADDRESS, [8](#)
- Gpio, [8](#)
- HIGH, [8](#)
- INPUT, [8](#)
- LOW, [8](#)
- OUTPUT, [8](#)
- USE_RW_BARRIER, [8](#)
- GpioRPI, [2](#)
 - begin, [2](#)
 - clearPin, [2](#)
 - configureBits, [2](#)
 - digitalRead, [2](#)
 - digitalWrite, [2](#)
 - gpio, [3](#)
 - pinMode, [2](#)
 - safeRead, [2](#)
 - safeWrite, [3](#)
 - setPin, [3](#)
 - stop, [3](#)
- HIGH
 - Gpio.h, [8](#)
- INPUT
 - Gpio.h, [8](#)
- LOW
 - Gpio.h, [8](#)
- OUTPUT
 - Gpio.h, [8](#)
- pinMode
 - GpioRPI, [2](#)
- safeRead
 - GpioRPI, [2](#)
- safeWrite
 - GpioRPI, [3](#)
- setPin
 - GpioRPI, [3](#)
- stop
 - GpioRPI, [3](#)
- USE_RW_BARRIER
 - Gpio.h, [8](#)