
mipea Documentation

Release 2.0.0

jasLogic

Jun 23, 2019

Contents

1	Installation	1
1.1	Download the configure script	1
1.2	Building the configure script	1
1.3	Troubleshooting	2
2	Peripherals	3
2.1	Macros	3
2.2	Functions	3
3	Clock Manager	5
3.1	Macros	5
3.2	Registers	5
3.3	Enums	6
3.4	Functions	6
4	GPIOs	9
4.1	Macros	9
4.2	Registers	9
4.3	Enums	10
4.4	Functions	10
5	I2C	13
5.1	Macros	13
5.2	Registers	13
5.3	Functions	14
6	PWM	17
6.1	Macros	17
6.2	Registers	18
6.3	Enums	18
6.4	Structs	19
6.5	Functions	19
7	SPI	21
7.1	Macros	21
7.2	Registers	22
7.3	Structs	22

7.4	Functions	23
8	Mipea Wrapper	25
8.1	Functions	25
9	GNU General Public License	27
10	GNU Free Documentation License	41
11	Indices and tables	51
	Index	53

The source code is hosted on [GitHub](#). mipea uses autotools (autoconf, automake, libtool) to build and install the library. The git repository does *not* include the `configure` script and `Makefile.in` which means that you have two options for installing the library.

1.1 Download the configure script

On [GitHub](#), when you look under the tab [releases](#) you will find some release with a name like for example “v2.0.0”. Then you can download the corresponding file named “mipea_x.x.x” which *includes the configure script and Makefile.in*. After downloading just run these commands from inside the downloaded directory:

```
$ ./configure
$ make
$ sudo make install
```

1.2 Building the configure script

When you have the GNU autotools installed you can simply clone this repository and build the `configure` script and `Makefile.in` yourself:

```
$ git clone https://github.com/jasLogic/mipea.git
$ cd mipea
$ autoreconf --install
$ ./configure
$ make
$ sudo make install
```

1.3 Troubleshooting

1.3.1 Configure script can not find /proc/cpuinfo

If the configure script prints this warning:

```
configure: WARNING: cannot find file /proc/cpuinfo
```

than the script was unable to find the `cpuinfo` file which is needed to determine the SoC (BCM2835 or BCM2836/7) and the revision. Pis with a revision number where the last four digits are less than 0004 use I2C bus 0 instead of 1, like the new ones.

This error can be fixed by editing the `config.h` file ensuring that it contains these lines (depending on your Pi):

```
#define BCM2835      1
#define BCM2836_7    1
#define USE_I2C_BUS_0 1
```

1.3.2 When running a program, the shared library file is not found

I noticed that sometimes the library can be linked, but when running a program an error message appears saying: File or directory not found. If you have this problem just run `ldconfig` or follow the output from `sudo make install`:

```
-----
Libraries have been installed in:
  /usr/local/lib

If you ever happen to want to link against installed libraries
in a given directory, LIBDIR, you must either use libtool, and
specify the full pathname of the library, or use the '-LLIBDIR'
flag during linking and do at least one of the following:
  - add LIBDIR to the 'LD_LIBRARY_PATH' environment variable
    during execution
  - add LIBDIR to the 'LD_RUN_PATH' environment variable
    during linking
  - use the '-Wl,-rpath -Wl,LIBDIR' linker flag
  - have your system administrator add LIBDIR to '/etc/ld.so.conf'

See any operating system documentation about shared libraries for
more information, such as the ld(1) and ld.so(8) manual pages.
-----
```

The peripheral functions are something like the *core* of the library. They map and unmap the memory used by all other parts.

2.1 Macros

PERIPHERAL_BASE_BCM2835

```
0x20000000
```

This macro holds the value of the peripheral base, when a BCM2835 is used.

PERIPHERAL_BASE_BCM2836_7

```
0x3F000000
```

This macro holds the value of the peripheral base, when a BCM2836 or BCM2837 is used.

2.2 Functions

`uint32_t * peripheral_map (uint32_t offset, uint32_t size)`

This function maps a code memory block of size *size* at offset *offset* from the peripheral base.

Note: The *offset* must be a multiple of the page size which is 4096 on the Raspberry Pi.

The function returns a pointer to the mapped memory on success and `NULL` on error.

`void peripheral_unmap (void* map, uint32_t size)`

This function unmaps the memory mapped to pointer *map* with size *size*.

int **peripheral_ismapped** (void **map*, uint32_t *size*)

This function checks if a pointer `map` is already mapped to a memory region with the size `size`. It returns `true` if the pointer is already mapped and `false` if not.

3.1 Macros

CLOCK_MANAGER_OFFSET

0x101000

This macro defines the offset at which the clock manager registers are located relative to the peripheral base.

CLOCK_MANAGER_SIZE

0xA4

This macro holds the size of the clock manager registers which needs to be mapped.

CM_PASSWD

0x5A000000

This macro holds the clock manager password. This value must always be present when writing to a clock manager register (e.g. by OR with the value).

3.2 Registers

volatile uint32_t ***clock_manager_base_ptr**

This pointer points, when mapped, to the base of the clock manager registers.

struct **clock_manager_register_map**

This struct maps the registers of the clock manager. The names of the struct members correspond to the registers.

Unfortunately, the official datasheet does not feature this chapter. But there is an upload of this chapter here: [BCM2835 clocks](#):

```
struct clock_manager_register_map {
    uint32_t GP0CTL;
    uint32_t GP0DIV;
    uint32_t GP1CTL;
    uint32_t GP1DIV;
    uint32_t GP2CTL;
    uint32_t GP2DIV;
    uint32_t: 32;
    uint32_t: 32;
    uint32_t: 32;
    uint32_t: 32;
    uint32_t PCMCTL;
    uint32_t PCMDIV;
    uint32_t PWMCTL;
    uint32_t PWMDIV;
}
```

CM

```
#define CM ((volatile struct clock_manager_register_map *) (clock_manager_base_ptr_
↪ + 28))
```

By using this macro, the registers of the clock manager can be accessed like this `CM->PWMCTL`.

3.3 Enums

`clock_source_t`

This enum holds the values for the different clock sources:

```
typedef enum {
    CLOCK_GND,
    CLOCK_OSC,
    CLOCK_TST0,
    CLOCK_TST1,
    CLOCK_PLLA,
    CLOCK_PLLC,
    CLOCK_PLLD,
    CLOCK_HDMI
} clock_source_t;
```

3.4 Functions

`uint32_t * clock_map (void)`

This function maps the clock manager registers. It calls `peripheral_map()` with the values `CLOCK_MANAGER_OFFSET` and `CLOCK_MANAGER_SIZE`.

`void clock_unmap (void)`

This function unmaps the clock manager.

The following functions all take a pointer to a clock manager register as an argument because all the registers for the *different clocks* have the *same structure*. This means that you just need to tell the clock manager which clock to use (by pointing to the right register). For example: `clock_enable (&CM->PWMCTL) ;`

void **clock_enable** (volatile uint32_t *reg)

This function enables the clock with the register pointed to by `reg`.

void **clock_disable** (volatile uint32_t *reg)

This function disables the clock with the register pointed to by `reg`.

void **clock_configure** (volatile uint32_t *reg, *clock_source_t* src, unsigned int divisor, unsigned int mash)

This function configures the clock with the register pointed to by `reg` and sets up the *clock_source_t* `src`, the divisor `divisor` with the mash factor `mash`.

Todo: Add a decimal places to the divisor.

4.1 Macros

GPIO_OFFSET

```
0x200000
```

This macro defines the offset at which the GPIO registers are located from the peripheral base.

GPIO_SIZE

```
0xB0
```

This macro holds the size of the GPIO registers which needs to be mapped.

4.2 Registers

`volatile uint32_t *gpio_base_ptr`

This pointer points, when mapped, to the base of the GPIO registers.

struct `gpio_register_map`

This struct maps the registers of the GPIOs. The names of the struct members correspond to the registers from the [Datasheet](#):

```
struct gpio_register_map {
    uint32_t FSEL[6];
    uint32_t: 32;
    uint32_t SET[2];
    uint32_t: 32;
    uint32_t CLR[2];
}
```

(continues on next page)

(continued from previous page)

```

uint32_t: 32;
uint32_t LEV[2];
uint32_t: 32;
uint32_t EDS[2];
uint32_t: 32;
uint32_t REN[2];
uint32_t: 32;
uint32_t FEN[2];
uint32_t: 32;
uint32_t HEN[2];
uint32_t: 32;
uint32_t LEN[2];
uint32_t: 32;
uint32_t AREN[2];
uint32_t: 32;
uint32_t AFEN[2];
uint32_t: 32;
uint32_t PUD;
uint32_t PUDCLK[2];
};

```

GP

```
#define GP ((volatile struct gpio_register_map *)gpio_base_ptr)
```

By using this macro, the registers of the GPIOs can be accessed like this `GP->SET[0]`.

4.3 Enums

pin_functions_t

This enum holds the values for the various pin functions:

```

typedef enum {
    INPUT, OUTPUT, ALT0, ALT1, ALT2, ALT3, ALT4, ALT5
} pin_functions_t;

```

pud_t

This enum holds the values for the states of the pullups / -downs:

```

typedef enum {
    PUD_DISABLE, PUD_DOWN, PUD_UP
} pud_t;

```

4.4 Functions

`uint32_t * gpio_map (void)`

This function maps the GPIO registers. It calls `peripheral_map()` with the values `GPIO_OFFSET` and `GPIO_SIZE`.

`void gpio_unmap (void)`

This function unmaps the GPIOs.

void **gpio_func** (uint32_t *pin*, *pin_functions_t* *function*)

This function sets the pin *pin* to the pin function *function*.

void **gpio_set** (uint32_t *pin*)

Set the pin *pin*.

void **gpio_clr** (uint32_t *pin*)

Clear the pin *pin*.

uint32_t **gpio_tst** (uint32_t *pin*)

Test the pin *pin*. This function returns 0 or false when the pin is low and non-zero if the pin is high.

void **gpio_pud** (uint32_t *pin*, *pud_t* *val*)

Use the pullup / -down functionality *val* on the pin *pin*.

void **gpio_inp** (uint32_t *pin*)

Make pin *pin* an input.

void **gpio_out** (uint32_t *pin*)

Make pin *pin* an output.

5.1 Macros

I2C_OFFSET_0

```
0x205000
```

I2C_OFFSET_1

```
0x804000
```

The BCM2835/6/7 has three BSC (I2C) controllers, from which only one is connected to the I2C pins. The older Pis (where the last four digits of the revision number are less than 0004) have BSC0 connected to the pins, all the other BSC1. The configure script tries to read the `/proc/cpuinfo` file, which includes the revision number, and then defines `USE_I2C_BUS_0` accordingly. See [Installation](#)

I2C_SIZE

```
0x18
```

This macro holds the size of the I2C registers which needs to be mapped.

```
I2C_FIFO_SIZE I2C_C_I2CEN I2C_C_ST I2C_C_CLEAR I2C_C_READ I2C_S_RXS I2C_S_TXD
I2C_S_DONE
```

5.2 Registers

```
volatile uint32_t *i2c_base_ptr
```

This pointer points, when mapped, to the base of the I2C registers.

struct **i2c_register_map**

This struct maps the registers of the BSC controller. The names of the struct members correspond to the registers from the [Datasheet](#):

```
struct i2c_register_map {
    uint32_t C;
    uint32_t S;
    uint32_t DLEN;
    uint32_t A;
    uint32_t FIFO;
    uint32_t DIV;
    uint32_t DEL;
    uint32_t CLKT;
};
```

I2C

```
#define I2C ((volatile struct i2c_register_map *)i2c_base_ptr)
```

By using this macro, the registers of the I2C can be accessed like this `I2C->FIFO`.

5.3 Functions

uint32_t * **i2c_map** (void)

This function maps the I2C registers. It calls `peripheral_map()` with the values `I2C_OFFSET` and `I2C_SIZE`. `I2C_OFFSET` is defined in `i2c.c`.

void **i2c_unmap** (void)

This function unmaps the I2C registers.

void **i2c_set_address** (uint8_t *addr*)

This function sets the address of the I2C device to communicate with. The address is a seven bit value.

void **i2c_set_clkdiv** (uint16_t *divisor*)

This function sets the clock divisor of the BSC controller.

Note: The clock source is the core clock with a frequency, according to the [Datasheet](#), of 150 MHz and according to [this file](#) and other sources of 250 MHz. When I tested the clock speed of I2C and SPI with a logic analyzer, it seems that 250 MHz **is correct** (at least for the Raspberry Pi Zero I use).

void **i2c_set_clkstr** (uint16_t *clkstr*)

This function sets the clock stretch timeout (or delay). This means that the master will wait `clkstr` cycles after the rising clock edge for the slave to respond. After this the timeout flag is set. This can often be left at reset value `0x40`.

void **i2c_start** (void)

Starts the BSC controller and clears the flag register.

void **i2c_stop** (void)

Disables the BSC controller.

void **i2c_write_byte** (uint8_t *byte*)

Write a `byte` of data.

`uint8_t i2c_read_byte (void)`

This function receives a byte of data and returns it.

`void i2c_write_data (const uint8_t *data, uint16_t length)`

This function writes `length` bytes of data pointed to by `data`.

`void i2c_read_data (uint8_t *data, uint16_t length)`

This function receives `length` bytes of data and writes them to the array `data`.

`void i2c_write_register (uint8_t reg, uint8_t data)`

This function writes to bytes of data. First `reg` and then `data`.

Note: You *cannot* use two calls to `i2c_write_byte()` instead of this function because this is only *one* transmission, while two times `i2c_write_byte()` would be *two* different transmissions.

`uint8_t i2c_read_register (uint8_t reg)`

In contrast to `i2c_write_register()` you *can* use a call to `i2c_write_byte()` and to `i2c_read_byte()`. This is because I2C needs to make two transmissions anyway to change the read / write bit.

Note: The [Datasheet](#) specifies PWM channels 0 and 1. The Raspberry Pi has pins for PWM channels 1 and 2, you just need to add one.

6.1 Macros

PWM_OFFSET

```
0x20C000
```

This macro defines the offset at which the PWM registers are located from the peripheral base.

PWM_SIZE

```
0x24
```

This macro holds the size of the I2C registers which needs to be mapped.

RNG_CHANNEL0

DAT_CHANNEL0

RNG_CHANNEL1

DAT_CHANNEL1

```
#define RNG_CHANNEL0    PWM->RNG1
#define DAT_CHANNEL0    PWM->DAT1
#define RNG_CHANNEL1    PWM->RNG2
#define DAT_CHANNEL1    PWM->DAT2
```

To prevent confusion (because the [Datasheet](#) calls the PWM channels 1 and 2 and the Raspberry Pi 0 and 1) the values of the registers which need to be used “on the fly” are :code;‘defined‘ from 2 to 1 and from 1 to 0.

6.1.1 Configuration Macros

PWM_CTL_MODE_PWM	Use PWM mode
PWM_CTL_MODE_SERIALISER	Use serialiser mode
PWM_RPTL_STOP	If serialiser mode: Transmission stops when fifo empty
PWM_RPTL_REPEAT	If serialiser mode: Repeat last data when fifo empty
PWM_SBIT_LOW	Output low when no transmission active
PWM_SBIT_HIGH	Output high when no transmission active
PWM_POLA_DEFAULT	Polarity is default
PWM_POLA_INVERTED	Polarity is innverted
PWM_USEF_DATA	Data register is transmitted
PWM_USEF_FIFO	Data from fifo is transmitted
PWM_MSEN_PWMALGORITHM	Use PWM algorithm
PWM_MSEN_MS_RATIO	Use MS ratio

6.2 Registers

`volatile uint32_t *pwm_base_ptr`

This pointer points, when mapped, to the base of the PWM registers.

`struct pwm_register_map`

This struct maps the registers of the PWM. The names of the struct members correspond to the registers from the [Datasheet](#):

```
struct pwm_register_map {
    uint32_t CTL;
    uint32_t STA;
    uint32_t DMAC;
    uint32_t: 32;
    uint32_t RNG1;
    uint32_t DAT1;
    uint32_t FIF1;
    uint32_t: 32;
    uint32_t RNG2;
    uint32_t DAT2;
};
```

PWM

```
#define PWM ((volatile struct pwm_register_map *)pwm_base_ptr)
```

By using this macro, the registers of the PWM can be accessed like this `PWM->RNG1`.

6.3 Enums

`pwm_channel_t`

This enum holds the values distinguishing PWM channel 0 and 1:

```
typedef enum {
    PWM_CHANNEL0, PWM_CHANNEL1
} pwm_channel_t;
```

6.4 Structs

`pwm_channel_config_t`

This struct is used to configure a PWM channel:

```
typedef struct {
    union {
        struct {
            uint32_t: 1;
            uint32_t mode: 1;
            uint32_t rpt1: 1;
            uint32_t sbit: 1;
            uint32_t pola: 1;
            uint32_t usef: 1;
            uint32_t: 1;
            uint32_t msen: 1;
        };
        uint32_t ctl_register;
    };
    unsigned int divisor;
    uint32_t range;
} pwm_channel_config_t;
```

`uint32_t ctl_register`

This member can be directly edited by the anonymous struct inside this union. This register maps directly to the CTL register, with some offset for PWM 1. The settings of this register are described in the *Macros*.

`unsigned int divisor`

The divisor which is passed to the *Clock Manager*.

`uint32_t range`

The range to which the PWM counter counts before it starts over.

6.5 Functions

`uint32_t *pwm_map (void)`

This function maps the PWM registers. It calls `peripheral_map()` with the values `PWM_OFFSET` and `PWM_SIZE`.

`void pwm_unmap (void)`

This function unmaps the PWM registers.

`void pwm_configure (pwm_channel_t channel, pwm_channel_config_t *config)`

This function configures `pwm_channel_t` channel with a `pwm_channel_config_t` pointed to by `config`.

`void pwm_enable (pwm_channel_t channel)`

This function enables `pwm_channel_t` channel.

void **pwm_disable** (*pwm_channel_t* channel)

This function disables *pwm_channel_t* channel.

7.1 Macros

SPI_OFFSET

0x204000

This macro defines the offset at which the SPI registers are located from the peripheral base.

SPI_SIZE

0x14

This macro holds the size of the I2C registers which needs to be mapped.

7.1.1 Configuration Macros

SPI_CS_CE0	Use chip enable 0
SPI_CS_CE1	Use chip enable 1
SPI_CS_CE2	Use chip enable 2
SPI_CPHA_CLK_BEGINNING	Data on clock leading edge
SPI_CPHA_CLK_MIDDLE	Data on clock trailing edge
SPI_CPOL_RESET_LOW	Clock polarity: active low
SPI_CPOL_RESET_HIGH	Clock polarity: active high
SPI_CSPOL_ACTIVE_LOW	Chip enable: active low
SPI_CSPOL_ACTIVE_HIGH	Chip enable: active high

7.2 Registers

volatile uint32_t ***spi_base_ptr**

This pointer points, when mapped, to the base of the SPI registers.

struct **spi_register_map**

This struct maps the registers of the SPI. The names of the struct members correspond to the registers from the [Datasheet](#):

```
struct spi_register_map {
    uint32_t CS;
    uint32_t FIFO;
    uint32_t CLK;
    uint32_t DLEN;
    uint32_t LTOH;
    uint32_t DC;
};
```

SPI

```
#define SPI ((volatile struct spi_register_map *)spi_base_ptr)
```

By using this macro, the registers of the SPI can be accessed like this `SPI->CS`.

7.3 Structs

spi_channel_config_t

This struct is used to configure SPI:

```
typedef struct {
    union {
        struct {
            uint32_t: 2;
            uint32_t cpha: 1;
            uint32_t cpol: 1;
            uint32_t: 2;
            uint32_t cspol: 1;
            uint32_t: 14;
            uint32_t cspol0: 1;
            uint32_t cspol1: 1;
            uint32_t cspol2: 1;
        };
        uint32_t cs_register;
    };
    uint16_t divisor;
} spi_channel_config_t;
```

uint32_t **cs_register**

This member can be directly edited by the anonymous struct inside this union. This register maps directly to the CS register. The settings of this register are described in the [Macros](#).

uint16_t **divisor**

The master clock divisor.

Note: The clock source is the core clock with a frequency, according to the [Datasheet](#), of 150 MHz and according to [this file](#) and other sources of 250 MHz. When I tested the clock speed of I2C and SPI with a logic analyzer, it seems that 250 MHz **is correct** (at least for the Raspberry Pi Zero I use).

7.4 Functions

uint32_t * **spi_map** (void)

This function maps the SPI registers. It calls *peripheral_map()* with the values *SPI_OFFSET* and *SPI_SIZE*.

void **spi_unmap** (void)

This function unmaps the SPI registers.

void **spi_configure** (*spi_channel_config_t* **config*)

This function configures SPI with a *spi_channel_config_t* pointed to by *config*.

void **spi_set_ce** (uint8_t *ce*)

This function sets which chip enable line the SPI controller should use. This can be a 3 bit value.

void **spi_transfer_start** (void)

This function starts a SPI transfer.

void **spi_transfer_stop** (void)

This function stops the current SPI transfer.

uint8_t **spi_transfer_byte** (uint8_t *data*)

While there is a SPI transfer active you can call this function as often as needed by the slave, to send and receive. This function needs to be called between *spi_transfer_start()* and *spi_transfer_stop()*, it sends *data* over SPI and asynchronously receives data and *returns* it.

uint8_t **spi_send2_recv1** (uint8_t *data0*, uint8_t *data1*)

This function writes to bytes of data and than keeps the clock running to receive and return the third byte. *spi_transfer_start()* and *spi_transfer_stop()* *may not* be called when using this function.

The `mipea.c / h` files are just a wrapper for all the other parts of the library. If you are lazy (or need all peripherals mapped) than this wrapper is usefull.

8.1 Functions

int **mipea_map** (void)

This function maps all the peripherals.

void **mipea_unmap** (void)

This function unmaps all the peripherals.

GNU General Public License

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<https://fsf.org/>>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program--to make sure it remains free
software for all its users. We, the Free Software Foundation, use the
GNU General Public License for most of our software; it applies also to
any other work released this way by its authors. You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
them if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new
free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you
these rights or asking you to surrender the rights. Therefore, you have
certain responsibilities if you distribute copies of the software, or if
you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether

(continues on next page)

(continued from previous page)

gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

(continues on next page)

(continued from previous page)

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding

(continues on next page)

(continued from previous page)

Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey,

(continues on next page)

(continued from previous page)

and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the

(continues on next page)

(continued from previous page)

product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as

(continues on next page)

(continued from previous page)

part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or

(continues on next page)

(continued from previous page)

authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

(continues on next page)

(continued from previous page)

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

(continues on next page)

(continued from previous page)

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you

(continues on next page)

(continued from previous page)

to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY

(continues on next page)

(continued from previous page)

GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
```

```
This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.
```

```
This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
```

```
You should have received a copy of the GNU General Public License
along with this program. If not, see <https://www.gnu.org/licenses/>.
```

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands

(continues on next page)

(continued from previous page)

might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see [<https://www.gnu.org/licenses/>](https://www.gnu.org/licenses/).

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read [<https://www.gnu.org/licenses/why-not-lgpl.html>](https://www.gnu.org/licenses/why-not-lgpl.html).

CHAPTER 10

GNU Free Documentation License

GNU Free Documentation License
Version 1.3, 3 November 2008

Copyright (C) 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc.
<<https://fsf.org/>>

Everyone **is** permitted to copy **and** distribute verbatim copies
of this license document, but changing it **is not** allowed.

0. PREAMBLE

The purpose of this License **is** to make a manual, textbook, **or** other
functional **and** useful document "**free**" **in** the sense of freedom: to
assure everyone the effective freedom to copy **and** redistribute it,
with or without modifying it, either commercially **or** noncommercially.
Secondarily, this License preserves **for** the author **and** publisher a way
to get credit **for** their work, **while not** being considered responsible
for modifications made by others.

This License **is** a kind of "**copyleft**", which means that derivative
works of the document must themselves be free **in** the same sense. It
complements the GNU General Public License, which **is** a copyleft
license designed **for** free software.

We have designed this License **in** order to use it **for** manuals **for** free
software, because free software needs free documentation: a free
program should come **with** manuals providing the same freedoms that the
software does. But this License **is not** limited to software manuals;
it can be used **for any** textual work, regardless of subject matter **or**
whether it **is** published **as** a printed book. We recommend this License
principally **for** works whose purpose **is** instruction **or** reference.

(continues on next page)

(continued from previous page)

1. APPLICABILITY AND DEFINITIONS

This License applies to **any** manual **or** other work, **in any** medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited **in** duration, to use that work under the conditions stated herein. The "Document", below, refers to **any** such manual **or** work. Any member of the public **is** a licensee, **and is** addressed **as "you"**. You accept the license **if** you copy, modify **or** distribute the work **in** a way requiring permission under copyright law.

A "Modified Version" of the Document means **any** work containing the Document **or** a portion of it, either copied verbatim, **or with** modifications **and/or** translated into another language.

A "Secondary Section" **is** a named appendix **or** a front-matter section of the Document that deals exclusively **with** the relationship of the publishers **or** authors of the Document to the Document's overall subject (**or** to related matters) **and** contains nothing that could fall directly within that overall subject. (Thus, **if** the Document **is in** part a textbook of mathematics, a Secondary Section may **not** explain **any** mathematics.) The relationship could be a matter of historical connection **with** the subject **or with** related matters, **or** of legal, commercial, philosophical, ethical **or** political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, **as** being those of Invariant Sections, **in** the notice that says that the Document **is** released under this License. If a section does **not** fit the above definition of Secondary then it **is not** allowed to be designated **as** Invariant. The Document may contain zero Invariant Sections. If the Document does **not** identify **any** Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, **as** Front-Cover Texts **or** Back-Cover Texts, **in** the notice that says that the Document **is** released under this License. A Front-Cover Text may be at most 5 words, **and** a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented **in** a format whose specification **is** available to the general public, that **is** suitable **for** revising the document straightforwardly **with** generic text editors **or** (**for** images composed of pixels) generic paint programs **or** (**for** drawings) some widely available drawing editor, **and** that **is** suitable **for input** to text formatters **or** **for** automatic translation to a variety of formats suitable **for input** to text formatters. A copy made **in** an otherwise Transparent file format whose markup, **or** absence of markup, has been arranged to thwart **or** discourage subsequent modification by readers **is not** Transparent. An image format **is not** Transparent **if** used **for any** substantial amount of text. A copy that **is not** "Transparent" **is** called "Opaque".

Examples of suitable formats **for** Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML **or** XML using a publicly available DTD, **and** standard-conforming simple HTML, PostScript **or** PDF designed **for** human modification. Examples of

(continues on next page)

(continued from previous page)

transparent image formats include PNG, XCF **and** JPG. Opaque formats include proprietary formats that can be read **and** edited only by proprietary word processors, SGML **or** XML **for** which the DTD **and/or** processing tools are **not** generally available, **and** the machine-generated HTML, PostScript **or** PDF produced by some word processors **for** output purposes only.

The "Title Page" means, **for** a printed book, the title page itself, plus such following pages **as** are needed to hold, legibly, the material this License requires to appear **in** the title page. For works **in** formats which do **not** have any title page **as** such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person **or** entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either **is** precisely XYZ **or** contains XYZ **in** parentheses following text that translates XYZ **in** another language. (Here XYZ stands **for** a specific section name mentioned below, such **as** "Acknowledgements", "Dedications", "Endorsements", **or** "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers **next** to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference **in** this License, but only **as** regards disclaiming warranties: any other implication that these Warranty Disclaimers may have **is** void **and** has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy **and** distribute the Document **in** any medium, either commercially **or** noncommercially, provided that this License, the copyright notices, **and** the license notice saying this License applies to the Document are reproduced **in** all copies, **and** that you add no other conditions whatsoever to those of this License. You may **not** use technical measures to obstruct **or** control the reading **or** further copying of the copies you make **or** distribute. However, you may accept compensation **in** exchange **for** copies. If you distribute a large enough number of copies you must also follow the conditions **in** section 3.

You may also lend copies, under the same conditions stated above, **and** you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (**or** copies **in** media that commonly have printed covers) of the Document, numbering more than 100, **and** the Document's license notice requires Cover Texts, you must enclose the copies **in** covers that carry, clearly **and** legibly, all these Cover Texts: Front-Cover Texts on the front cover, **and** Back-Cover Texts on the back cover. Both covers must also clearly **and** legibly identify you **as** the publisher of these copies. The front cover must present

(continues on next page)

(continued from previous page)

the full title **with** all words of the title equally prominent **and** visible. You may add other material on the covers **in** addition. Copying **with** changes limited to the covers, **as** long **as** they preserve the title of the Document **and** satisfy these conditions, can be treated **as** verbatim copying **in** other respects.

If the required texts **for** either cover are too voluminous to fit legibly, you should put the first ones listed (**as** many **as** fit reasonably) on the actual cover, **and** **continue** the rest onto adjacent pages.

If you publish **or** distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along **with** each Opaque copy, **or** state **in** **or** **with** each Opaque copy a computer-network location **from** **which** the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies **in** quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly **or** through your agents **or** retailers) of that edition to the public.

It **is** requested, but **not** required, that you contact the authors of the Document well before redistributing **any** large number of copies, to give them a chance to provide you **with** an updated version of the Document.

4. MODIFICATIONS

You may copy **and** distribute a Modified Version of the Document under the conditions of sections 2 **and** 3 above, provided that you release the Modified Version under precisely this License, **with** the Modified Version filling the role of the Document, thus licensing distribution **and** modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things **in** the Modified Version:

- A. Use **in** the Title Page (**and** on the covers, **if** **any**) a title distinct **from** **that** of the Document, **and** **from** **those** of previous versions (which should, **if** there were **any**, be listed **in** the History section of the Document). You may use the same title **as** a previous version **if** the original publisher of that version gives permission.
- B. List on the Title Page, **as** authors, one **or** more persons **or** entities responsible **for** authorship of the modifications **in** the Modified Version, together **with** at least five of the principal authors of the Document (**all** of its principal authors, **if** it has fewer than five), unless they release you **from** **this** requirement.
- C. State on the Title page the name of the publisher of the Modified Version, **as** the publisher.
- D. Preserve **all** the copyright notices of the Document.
- E. Add an appropriate copyright notice **for** your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, **in** the form shown **in** the Addendum below.

(continues on next page)

(continued from previous page)

- G. Preserve **in** that license notice the full lists of Invariant Sections **and** required Cover Texts given **in** the Document's **license notice**.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "**History**", Preserve its Title, **and** add to it an item stating at least the title, year, new authors, **and** publisher of the Modified Version **as** given on the Title Page. If there **is** no section Entitled "**History**" **in** the Document, create one stating the title, year, authors, **and** publisher of the Document **as** given on its Title Page, then add an item describing the Modified Version **as** stated **in** the previous sentence.
- J. Preserve the network location, **if any**, given **in** the Document **for** public access to a Transparent copy of the Document, **and** likewise the network locations given **in** the Document **for** previous versions it was based on. These may be placed **in** the "**History**" section. You may omit a network location **for** a work that was published at least four years before the Document itself, **or if** the original publisher of the version it refers to gives permission.
- K. For **any** section Entitled "**Acknowledgements**" **or** "**Dedications**", Preserve the Title of the section, **and** preserve **in** the section **all** the substance **and** tone of each of the contributor acknowledgements **and/or** dedications given therein.
- L. Preserve **all** the Invariant Sections of the Document, unaltered **in** their text **and in** their titles. Section numbers **or** the equivalent are **not** considered part of the section titles.
- M. Delete **any** section Entitled "**Endorsements**". Such a section may **not** be included **in** the Modified Version.
- N. Do **not** retitle **any** existing section to be Entitled "**Endorsements**" **or** to conflict **in** title **with any** Invariant Section.
- O. Preserve **any** Warranty Disclaimers.

If the Modified Version includes new front-matter sections **or** appendices that qualify **as** Secondary Sections **and** contain no material copied **from the** Document, you may at your option designate some **or all** of these sections **as** invariant. To do this, add their titles to the **list** of Invariant Sections **in** the Modified Version's **license notice**. These titles must be distinct **from any** other section titles.

You may add a section Entitled "**Endorsements**", provided it contains nothing but endorsements of your Modified Version by various parties--**for** example, statements of peer review **or** that the text has been approved by an organization **as** the authoritative definition of a standard.

You may add a passage of up to five words **as** a Front-Cover Text, **and** a passage of up to 25 words **as** a Back-Cover Text, to the end of the **list** of Cover Texts **in** the Modified Version. Only one passage of Front-Cover Text **and** one of Back-Cover Text may be added by (**or** through arrangements made by) **any** one entity. If the Document already includes a cover text **for** the same cover, previously added by you **or** by arrangement made by the same entity you are acting on behalf of, you may **not** add another; but you may replace the old one, on explicit permission **from the** previous publisher that added the old one.

The author(s) **and** publisher(s) of the Document do **not** by this License give permission to use their names **for** publicity **for or** to **assert or** imply endorsement of **any** Modified Version.

(continues on next page)

(continued from previous page)

5. COMBINING DOCUMENTS

You may combine the Document **with** other documents released under this License, under the terms defined **in** section 4 above **for** modified versions, provided that you include **in** the combination **all** of the Invariant Sections of **all** of the original documents, unmodified, **and** **list** them **all** **as** Invariant Sections of your combined work **in** its license notice, **and** that you preserve **all** their Warranty Disclaimers.

The combined work need only contain one copy of this License, **and** multiple identical Invariant Sections may be replaced **with** a single copy. If there are multiple Invariant Sections **with** the same name but different contents, make the title of each such section unique by adding at the end of it, **in** parentheses, the name of the original author **or** publisher of that section **if** known, **or else** a unique number. Make the same adjustment to the section titles **in** the **list** of Invariant Sections **in** the license notice of the combined work.

In the combination, you must combine **any** sections Entitled "History" **in** the various original documents, forming one section Entitled "History"; likewise combine **any** sections Entitled "Acknowledgements", **and** **any** sections Entitled "Dedications". You must delete **all** sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document **and** other documents released under this License, **and** replace the individual copies of this License **in** the various documents **with** a single copy that **is** included **in** the collection, provided that you follow the rules of this License **for** verbatim copying of each of the documents **in** **all** other respects.

You may extract a single document **from such** a collection, **and** distribute it individually under this License, provided you insert a copy of this License into the extracted document, **and** follow this License **in** **all** other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document **or** its derivatives **with** other separate **and** independent documents **or** works, **in** **or** on a volume of a storage **or** distribution medium, **is** called an "aggregate" **if** the copyright resulting **from the** compilation **is not** used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document **is** included **in** an aggregate, this License does **not** apply to the other works **in** the aggregate which are **not** themselves derivative works of the Document.

If the Cover Text requirement of section 3 **is** applicable to these copies of the Document, then **if** the Document **is** less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, **or** the

(continues on next page)

(continued from previous page)

electronic equivalent of covers **if** the Document **is in** electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation **is** considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections **with** translations requires special permission **from their** copyright holders, but you may include translations of some **or all** Invariant Sections **in** addition to the original versions of these Invariant Sections. You may include a translation of this License, **and all** the license notices **in** the Document, **and any** Warranty Disclaimers, provided that you also include the original English version of this License **and** the original versions of those notices **and** disclaimers. In case of a disagreement between the translation **and** the original version of this License **or** a notice **or** disclaimer, the original version will prevail.

If a section **in** the Document **is** Entitled "**Acknowledgements**", "**Dedications**", **or** "**History**", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may **not** copy, modify, sublicense, **or** distribute the Document **except as** expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, **or** distribute it **is** void, **and** will automatically terminate your rights under this License.

However, **if** you cease **all** violation of this License, then your license **from a** particular copyright holder **is** reinstated (a) provisionally, unless **and** until the copyright holder explicitly **and finally** terminates your license, **and** (b) permanently, **if** the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license **from a** particular copyright holder **is** reinstated permanently **if** the copyright holder notifies you of the violation by some reasonable means, this **is** the first time you have received notice of violation of this License (**for any** work) **from that** copyright holder, **and** you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does **not** terminate the licenses of parties who have received copies **or** rights **from you** under this License. If your rights have been terminated **and not** permanently reinstated, receipt of a copy of some **or all** of the same material does **not** give you **any** rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the

(continues on next page)

(continued from previous page)

GNU Free Documentation License **from time** to time. Such new versions will be similar **in** spirit to the present version, but may differ **in** detail to address new problems **or** concerns. See <https://www.gnu.org/licenses/>.

Each version of the License **is** given a distinguishing version number. If the Document specifies that a particular numbered version of this License "**or any later version**" applies to it, you have the option of following the terms **and** conditions either of that specified version **or** of **any** later version that has been published (**not as** a draft) by the Free Software Foundation. If the Document does **not** specify a version number of this License, you may choose **any** version ever published (**not as** a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's **public statement of acceptance of a** version permanently authorizes you to choose that version **for** the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (**or "MMC Site"**) means **any** World Wide Web server that publishes copyrightable works **and** also provides prominent facilities **for** anybody to edit those works. A public wiki that anybody can edit **is** an example of such a server. A "**Massive Multiauthor Collaboration**" (**or "MMC"**) contained **in** the site means **any set** of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a **not-for-profit** corporation **with** a principal place of business **in** San Francisco, California, **as well as** future copyleft versions of that license published by that same organization.

"Incorporate" means to publish **or** republish a Document, **in** whole **or in** part, **as** part of another Document.

An MMC **is** "**eligible for relicensing**" **if** it **is** licensed under this License, **and if** all works that were first published under this License somewhere other than this MMC, **and** subsequently incorporated **in** whole **or in** part into the MMC, (1) had no cover texts **or** invariant sections, **and** (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained **in** the site under CC-BY-SA on the same site at **any** time before August 1, 2009, provided the MMC **is** eligible **for** relicensing.

ADDENDUM: How to use this License **for** your documents

To use this License **in** a document you have written, include a copy of the License **in** the document **and** put the following copyright **and** license notices just after the title page:

```
Copyright (c)  YEAR  YOUR NAME.
Permission is granted to copy, distribute and/or modify this document
under the terms of the GNU Free Documentation License, Version 1.3
or any later version published by the Free Software Foundation;
```

(continues on next page)

(continued from previous page)

with no Invariant Sections, no Front-Cover Texts, **and** no Back-Cover Texts.
A copy of the license **is** included **in** the section entitled "GNU
Free Documentation License".

If you have Invariant Sections, Front-Cover Texts **and** Back-Cover Texts,
replace the "with...Texts." line **with** this:

with the Invariant Sections being LIST THEIR TITLES, **with** the
Front-Cover Texts being LIST, **and with** the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, **or** some other
combination of the three, merge those two alternatives to suit the
situation.

If your document contains nontrivial examples of program code, we
recommend releasing these examples **in** parallel under your choice of
free software license, such **as** the GNU General Public License,
to permit their use **in** free software.

CHAPTER 11

Indices and tables

- `genindex`
- `modindex`
- `search`

C

Clock Manager, 4
clock_configure (*C function*), 7
clock_disable (*C function*), 7
clock_enable (*C function*), 7
clock_manager_base_ptr (*C variable*), 5
CLOCK_MANAGER_OFFSET (*C macro*), 5
clock_manager_register_map (*C type*), 5
CLOCK_MANAGER_SIZE (*C macro*), 5
clock_map (*C function*), 6
clock_source_t (*C type*), 6
clock_unmap (*C function*), 6
CM (*C macro*), 6
CM_PASSWD (*C macro*), 5

D

DAT_CHANNEL0 (*C macro*), 17
DAT_CHANNEL1 (*C macro*), 17

G

GP (*C macro*), 10
gpio_base_ptr (*C variable*), 9
gpio_clr (*C function*), 11
gpio_func (*C function*), 10
gpio_inp (*C function*), 11
gpio_map (*C function*), 10
GPIO_OFFSET (*C macro*), 9
gpio_out (*C function*), 11
gpio_pud (*C function*), 11
gpio_register_map (*C type*), 9
gpio_set (*C function*), 11
GPIO_SIZE (*C macro*), 9
gpio_tst (*C function*), 11
gpio_unmap (*C function*), 10
GPIOs, 7

I

I2C, 11
I2C (*C macro*), 14

i2c_base_ptr (*C variable*), 13
i2c_map (*C function*), 14
I2C_OFFSET_0 (*C macro*), 13
I2C_OFFSET_1 (*C macro*), 13
i2c_read_byte (*C function*), 14
i2c_read_data (*C function*), 15
i2c_read_register (*C function*), 15
i2c_register_map (*C type*), 13
i2c_set_address (*C function*), 14
i2c_set_clkdiv (*C function*), 14
i2c_set_clkstr (*C function*), 14
I2C_SIZE (*C macro*), 13
i2c_start (*C function*), 14
i2c_stop (*C function*), 14
i2c_unmap (*C function*), 14
i2c_write_byte (*C function*), 14
i2c_write_data (*C function*), 15
i2c_write_register (*C function*), 15
Installation, 1

M

Mipea Wrapper, 23
mipea_map (*C function*), 25
mipea_unmap (*C function*), 25

P

PERIPHERAL_BASE_BCM2835 (*C macro*), 3
PERIPHERAL_BASE_BCM2836_7 (*C macro*), 3
peripheral_ismapped (*C function*), 4
peripheral_map (*C function*), 3
peripheral_unmap (*C function*), 3
Peripherals, 2
pin_functions_t (*C type*), 10
pud_t (*C type*), 10
PWM, 15
PWM (*C macro*), 18
pwm_base_ptr (*C variable*), 18
pwm_channel_config_t (*C type*), 19
pwm_channel_config_tctl_register (*C member*), 19

`pwm_channel_config_t.divisor` (*C member*),
19
`pwm_channel_config_t.range` (*C member*), 19
`pwm_channel_t` (*C type*), 18
`pwm_configure` (*C function*), 19
`pwm_disable` (*C function*), 19
`pwm_enable` (*C function*), 19
`pwm_map` (*C function*), 19
`PWM_OFFSET` (*C macro*), 17
`pwm_register_map` (*C type*), 18
`PWM_SIZE` (*C macro*), 17
`pwm_unmap` (*C function*), 19

R

`RNG_CHANNEL0` (*C macro*), 17
`RNG_CHANNEL1` (*C macro*), 17

S

`SPI`, 20
`SPI` (*C macro*), 22
`spi_base_ptr` (*C variable*), 22
`spi_channel_config_t` (*C type*), 22
`spi_channel_config_t.cs_register` (*C member*), 22
`spi_channel_config_t.divisor` (*C member*),
22
`spi_configure` (*C function*), 23
`spi_map` (*C function*), 23
`SPI_OFFSET` (*C macro*), 21
`spi_register_map` (*C type*), 22
`spi_send2_recv1` (*C function*), 23
`spi_set_ce` (*C function*), 23
`SPI_SIZE` (*C macro*), 21
`spi_transfer_byte` (*C function*), 23
`spi_transfer_start` (*C function*), 23
`spi_transfer_stop` (*C function*), 23
`spi_unmap` (*C function*), 23

T

Troubleshooting, 1