

## My Project

Generated by Doxygen 1.8.13



# Contents

<b>1</b>	<b>File Index</b>	<b>1</b>
1.1	File List . . . . .	1
<b>2</b>	<b>File Documentation</b>	<b>3</b>
2.1	ADC.c File Reference . . . . .	3
2.1.1	Function Documentation . . . . .	3
2.1.1.1	convert_wifire_adc() . . . . .	3
2.1.1.2	init_wifire_adc() . . . . .	3
2.1.1.3	read_potentiometer_with_adc() . . . . .	4
2.2	ADC.h File Reference . . . . .	4
2.2.1	Detailed Description . . . . .	4
2.2.2	Function Documentation . . . . .	4
2.2.2.1	convert_wifire_adc() . . . . .	4
2.2.2.2	init_wifire_adc() . . . . .	4
2.3	configuration_bits.c File Reference . . . . .	4
2.4	main.c File Reference . . . . .	4
2.4.1	Function Documentation . . . . .	5
2.4.1.1	main() . . . . .	5
2.5	user.c File Reference . . . . .	5
2.5.1	Function Documentation . . . . .	6
2.5.1.1	__ISR() . . . . .	6
2.5.1.2	adjust_led1_brightness() . . . . .	6
2.5.1.3	blink() . . . . .	6
2.5.1.4	choose_mode() . . . . .	8

2.5.1.5	<code>delay()</code>	8
2.5.1.6	<code>init_app()</code>	9
2.5.1.7	<code>init_gpio()</code>	9
2.5.1.8	<code>init_timer2_and_oc5()</code>	9
2.5.2	Variable Documentation	10
2.5.2.1	<code>mode</code>	10
2.6	<code>user.h</code> File Reference	10
2.6.1	Macro Definition Documentation	10
2.6.1.1	<code>BTN1_PORT_BIT</code>	11
2.6.1.2	<code>BTN2_PORT_BIT</code>	11
2.6.1.3	<code>LD1_PORT_BIT</code>	11
2.6.1.4	<code>LD2_PORT_BIT</code>	11
2.6.1.5	<code>LD3_PORT_BIT</code>	11
2.6.1.6	<code>LD4_PORT_BIT</code>	11
2.6.1.7	<code>MAX_ADC_VALUE</code>	12
2.6.1.8	<code>PWM_FREQ_HZ</code>	12
2.6.1.9	<code>PWM_PERIOD_COUNTS</code>	12
2.6.1.10	<code>VR1_AN_CHAN_NUM</code>	12
2.6.2	Function Documentation	12
2.6.2.1	<code>adjust_led1_brightness()</code>	12
2.6.2.2	<code>blink()</code>	13
2.6.2.3	<code>delay()</code>	13
2.6.2.4	<code>init_app()</code>	13
	<b>Index</b>	<b>15</b>

# Chapter 1

## File Index

### 1.1 File List

Here is a list of all files with brief descriptions:

<a href="#">ADC.c</a>	3
<a href="#">ADC.h</a>	4
<a href="#">configuration_bits.c</a>	4
<a href="#">main.c</a>	4
<a href="#">user.c</a>	5
<a href="#">user.h</a>	10



## Chapter 2

# File Documentation

### 2.1 ADC.c File Reference

```
#include <stdint.h>
#include <stdbool.h>
#include "user.h"
```

#### Functions

- void [init\\_wifire\\_adc](#) (void)
- int [convert\\_wifire\\_adc](#) (uint8\_t channelNumber)
- int [read\\_potentiometer\\_with\\_adc](#) (void)

#### 2.1.1 Function Documentation

##### 2.1.1.1 [convert\\_wifire\\_adc\(\)](#)

```
int convert_wifire_adc (
    uint8_t channelNumber )
```

##### 2.1.1.2 [init\\_wifire\\_adc\(\)](#)

```
void init_wifire_adc (
    void )
```

### 2.1.1.3 read\_potentiometer\_with\_adc()

```
int read_potentiometer_with_adc (  
    void )
```

## 2.2 ADC.h File Reference

### Functions

- void [init\\_wifire\\_adc](#) (void)
- int [convert\\_wifire\\_adc](#) (uint8\_t channelNumber)

### 2.2.1 Detailed Description

#### Author

: Eugene Punov

### 2.2.2 Function Documentation

#### 2.2.2.1 convert\_wifire\_adc()

```
int convert_wifire_adc (  
    uint8_t channelNumber )
```

#### 2.2.2.2 init\_wifire\_adc()

```
void init_wifire_adc (  
    void )
```

## 2.3 configuration\_bits.c File Reference

## 2.4 main.c File Reference

```
#include <stdint.h>  
#include <stdbool.h>  
#include "user.h"
```



## Functions

- void `main` (void)

### 2.4.1 Function Documentation

#### 2.4.1.1 `main()`

```
void main (  
    void )
```

#### Author

Eugene Punov

#### Copyright

Public License.

#### Precondition

main file of our programm

< For `uint32_t` definition

< For true/false definition

< User funct/params, such as `InitApp` < Initialize I/O and beginiing main cycle

## 2.5 user.c File Reference

```
#include <stdint.h>  
#include <stdbool.h>  
#include "user.h"  
#include <sys/attrs.h>  
#include "ADC.h"
```

## Functions

- void `init_timer2_and_oc5` (void)
- void `adjust_led1_brightness` (void)
- void `init_gpio` (void)
- void `init_app` (void)
- void `blink` (void)
- void `delay` (uint32\_t n)
- void `choose_mode` ()
- void `__ISR` (\_CHANGE\_NOTICE\_A\_VECTOR, IPL2SRS)

## Variables

- `uint8_t mode = 0`

*Different modes of programm.*

## 2.5.1 Function Documentation

### 2.5.1.1 \_\_ISR()

```
void __ISR (
    _CHANGE_NOTICE_A_VECTOR ,
    IPL2SRS )
```

#### Parameters

out	<i>none</i>	
in	<i>interruption</i>	

#### Returns

*none*

### 2.5.1.2 adjust\_led1\_brightness()

```
void adjust_led1_brightness (
    void )
```

for control led brightness

#### Parameters

out	<i>none</i>	
in	<i>none</i>	

#### Returns

*none*

### 2.5.1.3 blink()

```
void blink (
    void )
```

function for control blinking speed

**Parameters**

out	<i>none</i>	
in	<i>none</i>	

**Returns**

none

**2.5.1.4 choose\_mode()**

```
void choose_mode ( )
```

choose one of two modes

**Parameters**

out	<i>none</i>	
in	<i>none</i>	

**Returns**

none

**2.5.1.5 delay()**

```
void delay (
    uint32_t n )
```

delay function

**Parameters**

out	<i>none</i>	
in	<i>uint32_t</i> <i>_t</i>	n dellay in mills

**Returns**

none

### 2.5.1.6 init\_app()

```
void init_app (
    void )
```

begins program

#### Parameters

out	<i>none</i>	
in	<i>none</i>	

#### Returns

none

### 2.5.1.7 init\_gpio()

```
void init_gpio (
    void )
```

initialize gpio

#### Parameters

out	<i>none</i>	
in	<i>none</i>	

#### Returns

none

### 2.5.1.8 init\_timer2\_and\_oc5()

```
void init_timer2_and_oc5 (
    void )
```

Initialize Timer2

#### Parameters

out	<i>none</i>	
in	<i>none</i>	

**Returns**

none

< Select OC5

## 2.5.2 Variable Documentation

### 2.5.2.1 mode

```
uint8_t mode = 0
```

Different modes of programm.

**Precondition**

Main programmFor uint32\_t definition

## 2.6 user.h File Reference

**Macros**

- #define [LD1\\_PORT\\_BIT](#) LATGbits.LATG6
- #define [LD2\\_PORT\\_BIT](#) LATDbits.LATD4
- #define [LD3\\_PORT\\_BIT](#) LATBbits.LATB11
- #define [LD4\\_PORT\\_BIT](#) LATGbits.LATG15
- #define [BTN1\\_PORT\\_BIT](#) PORTAbits.RA5
- #define [BTN2\\_PORT\\_BIT](#) PORTAbits.RA4
- #define [PWM\\_FREQ\\_HZ](#) (1000)
- #define [PWM\\_PERIOD\\_COUNTS](#) (100000000/(256\*[PWM\\_FREQ\\_HZ](#)))
- #define [MAX\\_ADC\\_VALUE](#) (4095)
- #define [VR1\\_AN\\_CHAN\\_NUM](#) (8)

**Functions**

- void [init\\_app](#) (void)
- void [adjust\\_led1\\_brightness](#) (void)
- void [blink](#) (void)
- int [delay](#) (int n)

### 2.6.1 Macro Definition Documentation

### 2.6.1.1 BTN1\_PORT\_BIT

```
#define BTN1_PORT_BIT PORTAbits.RA5
```

### 2.6.1.2 BTN2\_PORT\_BIT

```
#define BTN2_PORT_BIT PORTAbits.RA4
```

### 2.6.1.3 LD1\_PORT\_BIT

```
#define LD1_PORT_BIT LATGbits.LATG6
```

#### Author

Eugene Punov

#### Copyright

Public License.

#### Precondition

Configuration and description for easier usage programmm

### 2.6.1.4 LD2\_PORT\_BIT

```
#define LD2_PORT_BIT LATDbits.LATD4
```

### 2.6.1.5 LD3\_PORT\_BIT

```
#define LD3_PORT_BIT LATBbits.LATB11
```

### 2.6.1.6 LD4\_PORT\_BIT

```
#define LD4_PORT_BIT LATGbits.LATG15
```

### 2.6.1.7 MAX\_ADC\_VALUE

```
#define MAX_ADC_VALUE (4095)
```

### 2.6.1.8 PWM\_FREQ\_HZ

```
#define PWM_FREQ_HZ (1000)
```

### 2.6.1.9 PWM\_PERIOD\_COUNTS

```
#define PWM_PERIOD_COUNTS (100000000/(256*PWM_FREQ_HZ))
```

### 2.6.1.10 VR1\_AN\_CHAN\_NUM

```
#define VR1_AN_CHAN_NUM (8)
```

## 2.6.2 Function Documentation

### 2.6.2.1 adjust\_led1\_brightness()

```
void adjust_led1_brightness (  
    void )
```

I/O and Peripheral Initialization

for control led brightness

#### Parameters

out	<i>none</i>	
in	<i>none</i>	

#### Returns

none



### 2.6.2.2 blink()

```
void blink (
    void )
```

control led brightness

function for control blinking speed

#### Parameters

out	<i>none</i>	
in	<i>none</i>	

#### Returns

none

### 2.6.2.3 delay()

```
int delay (
    int n )
```

Control speed of led blinking

#### Parameters

in	<i>n</i>	integer value for delay v.
----	----------	----------------------------

### 2.6.2.4 init\_app()

```
void init_app (
    void )
```

begins program

#### Parameters

out	<i>none</i>	
in	<i>none</i>	

#### Returns

none



# Index

- [\\_\\_ISR](#)
    - [user.c, 6](#)
- [ADC.c, 3](#)
  - [convert\\_wifire\\_adc, 3](#)
  - [init\\_wifire\\_adc, 3](#)
  - [read\\_potentiometer\\_with\\_adc, 3](#)
- [ADC.h, 4](#)
  - [convert\\_wifire\\_adc, 4](#)
  - [init\\_wifire\\_adc, 4](#)
- [adjust\\_led1\\_brightness](#)
  - [user.c, 6](#)
  - [user.h, 12](#)
- [BTN1\\_PORT\\_BIT](#)
  - [user.h, 10](#)
- [BTN2\\_PORT\\_BIT](#)
  - [user.h, 11](#)
- [blink](#)
  - [user.c, 6](#)
  - [user.h, 12](#)
- [choose\\_mode](#)
  - [user.c, 8](#)
- [configuration\\_bits.c, 4](#)
- [convert\\_wifire\\_adc](#)
  - [ADC.c, 3](#)
  - [ADC.h, 4](#)
- [delay](#)
  - [user.c, 8](#)
  - [user.h, 13](#)
- [init\\_app](#)
  - [user.c, 8](#)
  - [user.h, 13](#)
- [init\\_gpio](#)
  - [user.c, 9](#)
- [init\\_timer2\\_and\\_oc5](#)
  - [user.c, 9](#)
- [init\\_wifire\\_adc](#)
  - [ADC.c, 3](#)
  - [ADC.h, 4](#)
- [LD1\\_PORT\\_BIT](#)
  - [user.h, 11](#)
- [LD2\\_PORT\\_BIT](#)
  - [user.h, 11](#)
- [LD3\\_PORT\\_BIT](#)
  - [user.h, 11](#)
- [LD4\\_PORT\\_BIT](#)
  - [user.h, 11](#)
- [user.h, 11](#)
- [MAX\\_ADC\\_VALUE](#)
  - [user.h, 11](#)
- [main](#)
  - [main.c, 5](#)
- [main.c, 4](#)
  - [main, 5](#)
- [mode](#)
  - [user.c, 10](#)
- [PWM\\_FREQ\\_HZ](#)
  - [user.h, 12](#)
- [PWM\\_PERIOD\\_COUNTS](#)
  - [user.h, 12](#)
- [read\\_potentiometer\\_with\\_adc](#)
  - [ADC.c, 3](#)
- [user.c, 5](#)
  - [\\_\\_ISR, 6](#)
  - [adjust\\_led1\\_brightness, 6](#)
  - [blink, 6](#)
  - [choose\\_mode, 8](#)
  - [delay, 8](#)
  - [init\\_app, 8](#)
  - [init\\_gpio, 9](#)
  - [init\\_timer2\\_and\\_oc5, 9](#)
  - [mode, 10](#)
- [user.h, 10](#)
  - [adjust\\_led1\\_brightness, 12](#)
  - [BTN1\\_PORT\\_BIT, 10](#)
  - [BTN2\\_PORT\\_BIT, 11](#)
  - [blink, 12](#)
  - [delay, 13](#)
  - [init\\_app, 13](#)
  - [LD1\\_PORT\\_BIT, 11](#)
  - [LD2\\_PORT\\_BIT, 11](#)
  - [LD3\\_PORT\\_BIT, 11](#)
  - [LD4\\_PORT\\_BIT, 11](#)
  - [MAX\\_ADC\\_VALUE, 11](#)
  - [PWM\\_FREQ\\_HZ, 12](#)
  - [PWM\\_PERIOD\\_COUNTS, 12](#)
  - [VR1\\_AN\\_CHAN\\_NUM, 12](#)
- [VR1\\_AN\\_CHAN\\_NUM](#)
  - [user.h, 12](#)