

My Project

Generated by Doxygen 1.8.13

Contents

1	File Index	1
1.1	File List	1
2	File Documentation	3
2.1	ADC.c File Reference	3
2.1.1	Function Documentation	3
2.1.1.1	convertWiFIREadc()	3
2.1.1.2	initWiFIREadc()	4
2.1.1.3	ReadPotentiometerWithADC()	4
2.2	ADC.h File Reference	4
2.2.1	Function Documentation	4
2.2.1.1	convertWiFIREadc()	4
2.2.1.2	initWiFIREadc()	5
2.3	configuration_bits.c File Reference	5
2.4	main.c File Reference	5
2.4.1	Detailed Description	5
2.4.2	Function Documentation	5
2.4.2.1	main()	6
2.5	user.c File Reference	6
2.5.1	Macro Definition Documentation	6
2.5.1.1	VER	6
2.5.2	Function Documentation	6
2.5.2.1	AdjustLED1Brightness()	6
2.5.2.2	InitApp()	7

2.5.2.3	InitGPIO()	7
2.5.2.4	InitTimer2AndOC5()	7
2.6	user.h File Reference	7
2.6.1	Macro Definition Documentation	7
2.6.1.1	BTN1_PORT_BIT	8
2.6.1.2	BTN2_PORT_BIT	8
2.6.1.3	LD1_PORT_BIT	8
2.6.1.4	LD2_PORT_BIT	8
2.6.1.5	LD3_PORT_BIT	8
2.6.1.6	LD4_PORT_BIT	8
2.6.1.7	MAX_ADC_VALUE	8
2.6.1.8	PWM_FREQ_HZ	8
2.6.1.9	PWM_PERIOD_COUNTS	9
2.6.1.10	VR1_AN_CHAN_NUM	9
2.6.1.11	VR2_AN_CHAN_NUM	9
2.6.2	Function Documentation	9
2.6.2.1	AdjustLED1Brightness()	9
2.6.2.2	InitApp()	9
	Index	11

Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

ADC.c	3
ADC.h	4
configuration_bits.c	5
main.c	
File containing example of doxygen usage for quick reference	5
user.c	6
user.h	7

Chapter 2

File Documentation

2.1 ADC.c File Reference

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

Functions

- void [initWiFIREadc](#) (void)
- int [convertWiFIREadc](#) (uint8_t channelNumber)
- int [ReadPotentiometerWithADC](#) (void)

2.1.1 Function Documentation

2.1.1.1 [convertWiFIREadc\(\)](#)

```
int convertWiFIREadc (
    uint8_t channelNumber )
```

[convertWiFIREadc](#)

Parameters

<i>channelNumber</i>	- The PIC32 analog channel number as in the PIC32 datasheet
----------------------	---

Returns

The converted value for that channel

Warning

If return value of zero an error may have occurred

Converts the analog signal to a digital value on the given pic32 analog channel number

2.1.1.2 initWiFIREadc()

```
void initWiFIREadc (
    void )
```

Initialization ADC

2.1.1.3 ReadPotentiometerWithADC()

```
int ReadPotentiometerWithADC (
    void )
```

Read value from potentiometer with ADC

Returns

Return the converted data

2.2 ADC.h File Reference

Functions

- void [initWiFIREadc](#) (void)
- int [convertWiFIREadc](#) (uint8_t channelNumber)

2.2.1 Function Documentation

2.2.1.1 convertWiFIREadc()

```
int convertWiFIREadc (
    uint8_t channelNumber )
```

convertWiFIREadc

Parameters

<i>channelNumber</i>	- The PIC32 analog channel number as in the PIC32 datasheet
----------------------	---

Returns

The converted value for that channel

Warning

If return value of zero an error may have occurred

Converts the analog signal to a digital value on the given pic32 analog channel number

2.2.1.2 initWiFIREadc()

```
void initWiFIREadc (
    void )
```

Initialization ADC

2.3 configuration_bits.c File Reference

2.4 main.c File Reference

File containing example of doxygen usage for quick reference.

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

Functions

- `int32_t main` (void)

2.4.1 Detailed Description

File containing example of doxygen usage for quick reference.

Author

My Self

Date

07 Nov 2017

2.4.2 Function Documentation

2.4.2.1 main()

```
int32_t main (  
    void )
```

2.5 user.c File Reference

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

Macros

- `#define VER 2`

Functions

- void [InitTimer2AndOC5](#) (void)
- void [AdjustLED1Brightness](#) (void)
- void [InitGPIO](#) (void)
- void [InitApp](#) (void)

2.5.1 Macro Definition Documentation

2.5.1.1 VER

```
#define VER 2
```

2.5.2 Function Documentation

2.5.2.1 AdjustLED1Brightness()

```
void AdjustLED1Brightness (  
    void )
```

2.5.2.2 InitApp()

```
void InitApp (  
    void )
```

Call functions which initialize peripherals: GPIO, ADC, Timer2, OC5

2.5.2.3 InitGPIO()

```
void InitGPIO (  
    void )
```

Initialization GPIO

2.5.2.4 InitTimer2AndOC5()

```
void InitTimer2AndOC5 (  
    void )
```

initialization Timer 2 and OC5

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)
- `#define VR2_AN_CHAN_NUM` (45)

Functions

- void `InitApp` (void)
- void `AdjustLED1Brightness` (void)

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
```

#define Macros

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)
```

2.6.1.11 VR2_AN_CHAN_NUM

```
#define VR2_AN_CHAN_NUM (45)
```

2.6.2 Function Documentation

2.6.2.1 AdjustLED1Brightness()

```
void AdjustLED1Brightness (  
    void )
```

2.6.2.2 InitApp()

```
void InitApp (  
    void )
```

Call functions which initialize peripherals: GPIO, ADC, Timer2, OC5

Index

ADC.c, [3](#)
 convertWiFIREadc, [3](#)
 initWiFIREadc, [4](#)
 ReadPotentiometerWithADC, [4](#)

ADC.h, [4](#)
 convertWiFIREadc, [4](#)
 initWiFIREadc, [5](#)

AdjustLED1Brightness
 user.c, [6](#)
 user.h, [9](#)

BTN1_PORT_BIT
 user.h, [7](#)

BTN2_PORT_BIT
 user.h, [8](#)

configuration_bits.c, [5](#)

convertWiFIREadc
 ADC.c, [3](#)
 ADC.h, [4](#)

InitApp
 user.c, [6](#)
 user.h, [9](#)

InitGPIO
 user.c, [7](#)

InitTimer2AndOC5
 user.c, [7](#)

initWiFIREadc
 ADC.c, [4](#)
 ADC.h, [5](#)

LD1_PORT_BIT
 user.h, [8](#)

LD2_PORT_BIT
 user.h, [8](#)

LD3_PORT_BIT
 user.h, [8](#)

LD4_PORT_BIT
 user.h, [8](#)

MAX_ADC_VALUE
 user.h, [8](#)

main
 main.c, [5](#)

main.c, [5](#)
 main, [5](#)

PWM_FREQ_HZ
 user.h, [8](#)

PWM_PERIOD_COUNTS

user.h, [8](#)

ReadPotentiometerWithADC
 ADC.c, [4](#)

user.c, [6](#)
 AdjustLED1Brightness, [6](#)
 InitApp, [6](#)
 InitGPIO, [7](#)
 InitTimer2AndOC5, [7](#)
 VER, [6](#)

user.h, [7](#)
 AdjustLED1Brightness, [9](#)
 BTN1_PORT_BIT, [7](#)
 BTN2_PORT_BIT, [8](#)
 InitApp, [9](#)
 LD1_PORT_BIT, [8](#)
 LD2_PORT_BIT, [8](#)
 LD3_PORT_BIT, [8](#)
 LD4_PORT_BIT, [8](#)
 MAX_ADC_VALUE, [8](#)
 PWM_FREQ_HZ, [8](#)
 PWM_PERIOD_COUNTS, [8](#)
 VR1_AN_CHAN_NUM, [9](#)
 VR2_AN_CHAN_NUM, [9](#)

VER

 user.c, [6](#)

VR1_AN_CHAN_NUM
 user.h, [9](#)

VR2_AN_CHAN_NUM
 user.h, [9](#)