lab 5 and 6 SPI and FreeRTOS

Generated by Doxygen 1.8.13

Contents

Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

Chapter 2

File Documentation

2.1 src/ConfigPerformance.c File Reference

```
#include "FreeRTOS.h"
#include "ConfigPerformance.h"
```

Include dependency graph for ConfigPerformance.c:

2.2 src/ConfigPerformance.h File Reference

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

Functions

- void vHardwareConfigurePerformance (void)
- void vHardwareUseMultiVectoredInterrupts (void)

2.2.1 Function Documentation

2.2.1.1 vHardwareConfigurePerformance()

```
void vHardwareConfigurePerformance (
            void  )
```

2.2.1.2 vHardwareUseMultiVectoredInterrupts()

```
void vHardwareUseMultiVectoredInterrupts (
            void  )
```

2.3 src/FreeRTOSConfig.h File Reference

```
#include <xc.h>
```

Include dependency graph for FreeRTOSConfig.h:

Macros

- #define configUSE_PREEMPTION 1
- #define configUSE_PORT_OPTIMISED_TASK_SELECTION 1
- #define configUSE_QUEUE_SETS 1
- #define configUSE_IDLE_HOOK 0
- #define configUSE_TICK_HOOK 1
- #define configTICK_RATE_HZ ((TickType_t) 1000)
- #define configCPU_CLOCK_HZ (200000000UL)
- #define configPERIPHERAL_CLOCK_HZ (40000000UL)
- #define configMAX_PRIORITIES (8UL)
- #define configMINIMAL_STACK_SIZE (190)
- #define configISR_STACK_SIZE (400)
- #define configTOTAL_HEAP_SIZE ((size_t) 60000)
- #define configMAX_TASK_NAME_LEN (16)
- #define configUSE_TRACE_FACILITY 0
- #define configUSE_16_BIT_TICKS 0
- #define configIDLE_SHOULD_YIELD 1
- #define configUSE_MUTEXES 1
- #define configCHECK_FOR_STACK_OVERFLOW 3 /∗ Three also checks the system/interrupt stack. ∗/
- #define configQUEUE_REGISTRY_SIZE 0
- #define configUSE_RECURSIVE_MUTEXES 1
- #define configUSE_MALLOC_FAILED_HOOK 0
- #define configUSE_APPLICATION_TASK_TAG 0
- #define configUSE_COUNTING_SEMAPHORES 1
- #define configGENERATE_RUN_TIME_STATS 0
- #define configUSE_CO_ROUTINES 0
- #define configMAX_CO_ROUTINE_PRIORITIES (2)
- #define configUSE_TIMERS 1
- #define configTIMER_TASK_PRIORITY (2)
- #define configTIMER_QUEUE_LENGTH 5
- #define configTIMER_TASK_STACK_DEPTH (configMINIMAL_STACK_SIZE ∗ 2)
- #define INCLUDE_vTaskPrioritySet 1
- #define INCLUDE_uxTaskPriorityGet 1
- #define INCLUDE_vTaskDelete 1
- #define INCLUDE_vTaskCleanUpResources 0
- #define INCLUDE_vTaskSuspend 1
- #define INCLUDE_vTaskDelayUntil 1
- #define INCLUDE_vTaskDelay 1
- #define INCLUDE_uxTaskGetStackHighWaterMark 1
- #define INCLUDE_eTaskGetState 1
- #define INCLUDE_xTimerPendFunctionCall 1
- #define configKERNEL_INTERRUPT_PRIORITY 0x01
- #define configMAX_SYSCALL_INTERRUPT_PRIORITY 0x03
- #define configASSERT(x) if((x) == 0) vAssertCalled(__FILE__, __LINE__)

Functions

- void vAssertCalled (const char *pcFile, unsigned long ulLine)

2.3.1 Macro Definition Documentation

2.3.1.1 configASSERT

```
#define configASSERT(
             x ) if( ( x ) == 0 ) vAssertCalled( __FILE__, __LINE__ )
```

2.3.1.2 configCHECK_FOR_STACK_OVERFLOW

```
#define configCHECK_FOR_STACK_OVERFLOW 3 /* Three also checks the system/interrupt stack.  */
```

2.3.1.3 configCPU_CLOCK_HZ

```
#define configCPU_CLOCK_HZ ( 200000000UL )
```

2.3.1.4 configGENERATE_RUN_TIME_STATS

```
#define configGENERATE_RUN_TIME_STATS 0
```

2.3.1.5 configIDLE_SHOULD_YIELD

```
#define configIDLE_SHOULD_YIELD 1
```

2.3.1.6 configISR_STACK_SIZE

```
#define configISR_STACK_SIZE ( 400 )
```

2.3.1.7 configKERNEL_INTERRUPT_PRIORITY

```
#define configKERNEL_INTERRUPT_PRIORITY 0x01
```

2.3.1.8 configMAX_CO_ROUTINE_PRIORITIES

```
#define configMAX_CO_ROUTINE_PRIORITIES ( 2 )
```

2.3.1.9 configMAX_PRIORITIES

```
#define configMAX_PRIORITIES ( 8UL )
```

2.3.1.10 configMAX_SYSCALL_INTERRUPT_PRIORITY

```
#define configMAX_SYSCALL_INTERRUPT_PRIORITY 0x03
```

2.3.1.11 configMAX_TASK_NAME_LEN

```
#define configMAX_TASK_NAME_LEN ( 16 )
```

2.3.1.12 configMINIMAL_STACK_SIZE

```
#define configMINIMAL_STACK_SIZE ( 190 )
```

2.3.1.13 configPERIPHERAL_CLOCK_HZ

```
#define configPERIPHERAL_CLOCK_HZ ( 40000000UL )
```

2.3.1.14 configQUEUE_REGISTRY_SIZE

```
#define configQUEUE_REGISTRY_SIZE 0
```

2.3.1.15 configTICK_RATE_HZ

```
#define configTICK_RATE_HZ ( ( TickType_t ) 1000 )
```

2.3.1.16 configTIMER_QUEUE_LENGTH

```
#define configTIMER_QUEUE_LENGTH 5
```

2.3.1.17 configTIMER_TASK_PRIORITY

```
#define configTIMER_TASK_PRIORITY ( 2 )
```

2.3.1.18 configTIMER_TASK_STACK_DEPTH

```
#define configTIMER_TASK_STACK_DEPTH ( configMINIMAL_STACK_SIZE * 2 )
```

2.3.1.19 configTOTAL_HEAP_SIZE

```
#define configTOTAL_HEAP_SIZE ( ( size_t ) 60000 )
```

2.3.1.20 configUSE_16_BIT_TICKS

```
#define configUSE_16_BIT_TICKS 0
```

2.3.1.21 configUSE_APPLICATION_TASK_TAG

```
#define configUSE_APPLICATION_TASK_TAG 0
```

2.3.1.22 configUSE_CO_ROUTINES

```
#define configUSE_CO_ROUTINES 0
```

2.3.1.23 configUSE_COUNTING_SEMAPHORES

```
#define configUSE_COUNTING_SEMAPHORES 1
```

2.3.1.24 configUSE_IDLE_HOOK

```
#define configUSE_IDLE_HOOK 0
```

2.3.1.25 configUSE_MALLOC_FAILED_HOOK

```
#define configUSE_MALLOC_FAILED_HOOK 0
```

2.3.1.26 configUSE_MUTEXES

```
#define configUSE_MUTEXES 1
```

2.3.1.27 configUSE_PORT_OPTIMISED_TASK_SELECTION

```
#define configUSE_PORT_OPTIMISED_TASK_SELECTION 1
```

2.3.1.28 configUSE_PREEMPTION

```
#define configUSE_PREEMPTION 1
```

2.3.1.29 configUSE_QUEUE_SETS

```
#define configUSE_QUEUE_SETS 1
```

2.3.1.30 configUSE_RECURSIVE_MUTEXES

```
#define configUSE_RECURSIVE_MUTEXES 1
```

2.3.1.31 configUSE_TICK_HOOK

```
#define configUSE_TICK_HOOK 1
```

2.3.1.32 configUSE_TIMERS

```
#define configUSE_TIMERS 1
```

2.3.1.33 configUSE_TRACE_FACILITY

```
#define configUSE_TRACE_FACILITY 0
```

2.3.1.34 INCLUDE_eTaskGetState

```
#define INCLUDE_eTaskGetState 1
```

2.3.1.35 INCLUDE_uxTaskGetStackHighWaterMark

```
#define INCLUDE_uxTaskGetStackHighWaterMark 1
```

2.3.1.36 INCLUDE_uxTaskPriorityGet

```
#define INCLUDE_uxTaskPriorityGet 1
```

2.3.1.37 INCLUDE_vTaskCleanUpResources

```
#define INCLUDE_vTaskCleanUpResources 0
```

2.3.1.38 INCLUDE_vTaskDelay

```
#define INCLUDE_vTaskDelay 1
```

2.3.1.39 INCLUDE_vTaskDelayUntil

```
#define INCLUDE_vTaskDelayUntil 1
```

2.3.1.40 INCLUDE_vTaskDelete

```
#define INCLUDE_vTaskDelete 1
```

2.3.1.41 INCLUDE_vTaskPrioritySet

```
#define INCLUDE_vTaskPrioritySet 1
```

2.3.1.42 INCLUDE_vTaskSuspend

```
#define INCLUDE_vTaskSuspend 1
```

2.3.1.43 INCLUDE_xTimerPendFunctionCall

```
#define INCLUDE_xTimerPendFunctionCall 1
```

2.3.2 Function Documentation

2.3.2.1 vAssertCalled()

```
void vAssertCalled (
            const char * pcFile,
            unsigned long ulLine )
```

2.4 src/main.c File Reference

```
#include "FreeRTOS.h"
#include "task.h"
#include "semphr.h"
#include "queue.h"
#include "user.h"
#include "OLED.h"
#include "OLED/OledChar.h"
#include "OLED/OledGrph.h"
#include "UART.h"
#include "ConfigPerformance.h"
```

Include dependency graph for main.c:

Functions

- int main (void)
- void vApplicationStackOverflowHook (TaskHandle_t pxTask, char *pcTaskName)
- void vApplicationTickHook (void)
- void vAssertCalled (const char *pcFile, unsigned long ulLine)
- void _general_exception_handler (void)

2.4.1 Function Documentation

2.4.1.1 _general_exception_handler()

```
void _general_exception_handler (
            void  )
```

2.4.1.2 main()

```
int main (
            void  )
```

Create the demo tasks then start the scheduler.

2.4.1.3 vApplicationStackOverflowHook()

```
void vApplicationStackOverflowHook (
            TaskHandle_t pxTask,
            char * pcTaskName )
```

2.4.1.4 vApplicationTickHook()

```
void vApplicationTickHook (
             void  )
```

2.4.1.5 vAssertCalled()

```
void vAssertCalled (
             const char * pcFile,
             unsigned long ulLine )
```

2.5 src/user.c File Reference

initialization and system functions

```
#include <xc.h>
#include <stdint.h>
#include <sys/attribs.h>
#include "FreeRTOS.h"
#include "task.h"
#include "semphr.h"
#include "user.h"
#include "OLED.h"
#include "OledChar.h"
#include "OledGrph.h"
#include "UART.h"
#include <string.h>
#include <stdlib.h>
```

Include dependency graph for user.c:

Macros

- #define X_MAX 128
- #define Y_MAX 28
- #define X_center 64
- #define Y_center 14

Functions

- void InitGPIO (void)

  GPIO initialization for board, Initialization for buttons and LEDs Disabling analog mode and setting pins directions.
- void InitBIOSGPIO (void)

  GPIO initialization for schield, Initialization for buttons and LEDs Disabling analog mode and setting pins directions.
- void InitApp (void)

  App initialization, Calling initialization functions for GPIO and OLED Basic OLED configuration.
- void Task1 (void ∗pvParameters)

  Task #1, takes semaphore and than outputs text to the OLED.
- void Task2 (void ∗pvParameters)

  Task #2, takes semaphore and than outputs text to the OLED.
- void DelayMs (int t)

Variables

- volatile int n
- volatile int i =0

2.5.1 Detailed Description

initialization and system functions

2.5.2 Macro Definition Documentation

2.5.2.1 X_center

```
#define X_center 64
```

2.5.2.2 X_MAX

```
#define X_MAX 128
```

2.5.2.3 Y_center

```
#define Y_center 14
```

2.5.2.4 Y_MAX

```
#define Y_MAX 28
```

2.5.3 Function Documentation

2.5.3.1 DelayMs()

```
void DelayMs (
            int t )
```

2.5.3.2 InitApp()

```
void InitApp (
             void  )
```

App initialization, Calling initialization functions for GPIO and OLED Basic OLED configuration.

2.5.3.3 InitBIOSGPIO()

```
void InitBIOSGPIO (
             void  )
```

GPIO initialization for schield, Initialization for buttons and LEDs Disabling analog mode and setting pins directions.

2.5.3.4 InitGPIO()

```
void InitGPIO (
             void  )
```

GPIO initialization for board, Initialization for buttons and LEDs Disabling analog mode and setting pins directions.

2.5.3.5 Task1()

```
void Task1 (
             void * pvParameters )
```

Task #1, takes semaphore and than outputs text to the OLED.

2.5.3.6 Task2()

```
void Task2 (
             void * pvParameters )
```

Task #2, takes semaphore and than outputs text to the OLED.

2.5.4 Variable Documentation

2.5.4.1 i

```
volatile int i =0
```

2.5.4.2 n

```
volatile int n
```

2.6 src/user.h File Reference

contains definitions, macroses and function prototypes

```
#include "FreeRTOS.h"
#include "semphr.h"
```

Include dependency graph for user.h: This graph shows which files directly or indirectly include this file:

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 BIOS_LD1_PORT_BIT LATEbits.LATE0
- #define BIOS_LD2_PORT_BIT LATEbits.LATE1
- #define BIOS_LD3_PORT_BIT LATEbits.LATE2
- #define BIOS_LD4_PORT_BIT LATEbits.LATE3
- #define BIOS_LD5_PORT_BIT LATEbits.LATE4
- #define BIOS_LD6_PORT_BIT LATEbits.LATE5
- #define BIOS_LD7_PORT_BIT LATEbits.LATE6
- #define BIOS_LD8_PORT_BIT LATEbits.LATE7
- #define BTN_1_SCHLD PORTGbits.RG7
- #define BTN_2_SCHLD PORTDbits.RD5
- #define BTN_3_SCHLD PORTFbits.RF1
- #define BTN_4_SCHLD PORTAbits.RA2
- #define BIOS_SW1_PORT_BIT PORTEbits.RE8
- #define BIOS_SW2_PORT_BIT PORTEbits.RE9
- #define BIOS_SW3_PORT_BIT PORTAbits.RA14
- #define BIOS_SW4_PORT_BIT PORTCbits.RC1

Functions

- void InitApp (void)

  App initialization, Calling initialization functions for GPIO and OLED Basic OLED configuration.
- void Task1 (void ∗pvParameters)

  Task #1, takes semaphore and than outputs text to the OLED.
- void Task2 (void ∗pvParameters)

  Task #2, takes semaphore and than outputs text to the OLED.
- void DelayMs (int t)

Variables

- SemaphoreHandle_t xMutexOLED

2.6.1 Detailed Description

contains definitions, macroses and function prototypes

2.6.2 Macro Definition Documentation

2.6.2.1 BIOS_LD1_PORT_BIT

```
#define BIOS_LD1_PORT_BIT LATEbits.LATE0
```

2.6.2.2 BIOS_LD2_PORT_BIT

```
#define BIOS_LD2_PORT_BIT LATEbits.LATE1
```

2.6.2.3 BIOS_LD3_PORT_BIT

```
#define BIOS_LD3_PORT_BIT LATEbits.LATE2
```

2.6.2.4 BIOS_LD4_PORT_BIT

```
#define BIOS_LD4_PORT_BIT LATEbits.LATE3
```

2.6.2.5 BIOS_LD5_PORT_BIT

```
#define BIOS_LD5_PORT_BIT LATEbits.LATE4
```

2.6.2.6 BIOS_LD6_PORT_BIT

```
#define BIOS_LD6_PORT_BIT LATEbits.LATE5
```

2.6.2.7 BIOS_LD7_PORT_BIT

```
#define BIOS_LD7_PORT_BIT LATEbits.LATE6
```

2.6.2.8 BIOS_LD8_PORT_BIT

```
#define BIOS_LD8_PORT_BIT LATEbits.LATE7
```

2.6.2.9 BIOS_SW1_PORT_BIT

```
#define BIOS_SW1_PORT_BIT PORTEbits.RE8
```

2.6.2.10 BIOS_SW2_PORT_BIT

```
#define BIOS_SW2_PORT_BIT PORTEbits.RE9
```

2.6.2.11 BIOS_SW3_PORT_BIT

```
#define BIOS_SW3_PORT_BIT PORTAbits.RA14
```

2.6.2.12 BIOS_SW4_PORT_BIT

```
#define BIOS_SW4_PORT_BIT PORTCbits.RC1
```

2.6.2.13 BTN1_PORT_BIT

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

2.6.2.14 BTN2_PORT_BIT

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

2.6.2.15 BTN_1_SCHLD

```
#define BTN_1_SCHLD PORTGbits.RG7
```

2.6.2.16 BTN_2_SCHLD

```
#define BTN_2_SCHLD PORTDbits.RD5
```

2.6.2.17 BTN_3_SCHLD

```
#define BTN_3_SCHLD PORTFbits.RF1
```

2.6.2.18 BTN_4_SCHLD

```
#define BTN_4_SCHLD PORTAbits.RA2
```

2.6.2.19 LD1_PORT_BIT

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

2.6.2.20 LD2_PORT_BIT

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

2.6.2.21 LD3_PORT_BIT

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

2.6.2.22 LD4_PORT_BIT

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

2.6.3 Function Documentation

2.6.3.1 DelayMs()

```
void DelayMs (
            int t )
```

2.6.3.2 InitApp()

```
void InitApp (
            void  )
```

App initialization, Calling initialization functions for GPIO and OLED Basic OLED configuration.

2.6.3.3 Task1()

```
void Task1 (
            void * pvParameters )
```

Task #1, takes semaphore and than outputs text to the OLED.

2.6.3.4 Task2()

```
void Task2 (
            void * pvParameters )
```

Task #2, takes semaphore and than outputs text to the OLED.

2.6.4 Variable Documentation

2.6.4.1 xMutexOLED

```
SemaphoreHandle_t xMutexOLED
```

Index