

[LE]1 [CE] [RE] [LO] [CO] [RO]1 [LE] [CE] [RE]Generated on Fri Aug 2 2013
 15:19:43 for Sonoma (MAXREFDES14#) Code Documentation by Doxygen [LO]Generated
 on Fri Aug 2 2013 15:19:43 for Sonoma (MAXREFDES14#) Code Documentation by
 Doxygen [CO] [RO]

language=C++,inputencoding=utf8,basicstyle=,breaklines=true,breakatwhitespace=true,tabsize=4,numbers=left

Sonoma (MAXREFDES14#) Code Documentation

V01.00

Generated by Doxygen 1.8.2

Fri Aug 2 2013 15:19:45

Contents

1	Main Page	1
1.1	Introduction	1
2	Data Structure Index	3
2.1	Data Structures	3
3	File Index	5
3.1	File List	5
4	Data Structure Documentation	7
4.1	maximDateTime Struct Reference	7
4.1.1	Detailed Description	7
4.1.2	Field Documentation	7
4.1.2.1	uchDate	7
4.1.2.2	uchDow	7
4.1.2.3	uchHour	8
4.1.2.4	uchMinute	8
4.1.2.5	uchMonth	8
4.1.2.6	uchSecond	8
4.1.2.7	uchYear	8
4.2	maximOLEDDisplay Struct Reference	8
4.2.1	Detailed Description	8
4.2.2	Field Documentation	8
4.2.2.1	flippedBuffer	8
4.2.2.2	font	9
4.2.2.3	portStatus	9
4.2.2.4	writeBuffer	9
4.2.2.5	xgpioPort	9
5	File Documentation	11

5.1	src/maximDeviceSpecificUtilities.c File Reference	11
5.1.1	Detailed Description	11
5.1.2	Function Documentation	12
5.1.2.1	read_reg	12
5.1.2.2	write_reg	12
5.2	src/maximDeviceSpecificUtilities.h File Reference	13
5.2.1	Detailed Description	13
5.2.2	Function Documentation	13
5.2.2.1	read_reg	13
5.2.2.2	write_reg	14
5.3	src/MAXREFDES14.c File Reference	14
5.3.1	Detailed Description	15
5.3.2	Macro Definition Documentation	15
5.3.2.1	MAJOR_REVISION	15
5.3.2.2	MINOR_REVISION	15
5.3.3	Function Documentation	16
5.3.3.1	main	16
5.3.4	Variable Documentation	16
5.3.4.1	font	16
5.3.4.2	g_sInputString	16
5.3.4.3	g_sTempString	16
5.3.4.4	g_structureOLED	16
5.3.4.5	g_xGpioLed	16
5.4	src/MAXREFDES14.h File Reference	17
5.4.1	Detailed Description	17
5.4.2	Macro Definition Documentation	18
5.4.2.1	ABOUT_ONE_SECOND	18
5.4.2.2	DEFAULT_HYPERTERMINAL_UART_ADDRESS	18
5.4.2.3	DEFAULT_HYPERTERMINAL_UART_ID	18
5.4.2.4	INPUT_STRING_MAX_SIZE	18
5.4.3	Variable Documentation	18
5.4.3.1	g_nSampleRate	18
5.4.3.2	g_sInputString	18
5.4.3.3	g_structureOLED	19
5.4.3.4	g_tempString	19
5.4.3.5	g_uchAtoDConverterChannel	19
5.4.3.6	g_unSampleSize	19

5.4.3.7	g_xGpioLed	19
5.5	src/menu.c File Reference	19
5.5.1	Detailed Description	19
5.5.2	Function Documentation	20
5.5.2.1	menu_cls	20
5.5.2.2	menu_print_line	20
5.5.2.3	menu_print_main_menu	21
5.5.2.4	menu_print_maxim_banner	21
5.5.2.5	menu_print_maxim_banner_big	21
5.5.2.6	menu_print_prompt	21
5.5.2.7	menu_retrieve_keypress	22
5.6	src/menu.h File Reference	22
5.6.1	Detailed Description	23
5.6.2	Macro Definition Documentation	24
5.6.2.1	BASE_FUNCTION_STATE	24
5.6.2.2	KEYPRESS_ARROW_DOWN	24
5.6.2.3	KEYPRESS_ARROW_LEFT	24
5.6.2.4	KEYPRESS_ARROW_RIGHT	24
5.6.2.5	KEYPRESS_ARROW_UP	24
5.6.2.6	KEYPRESS_END	24
5.6.2.7	MAIN_MENU	24
5.6.2.8	MENU_READ_REG	25
5.6.2.9	MENU_REG_ADDR	25
5.6.2.10	MENU_WRITE_REG	25
5.6.2.11	WAIT_KEYPRESS	25
5.6.3	Function Documentation	25
5.6.3.1	menu_cls	25
5.6.3.2	menu_print_main_menu	25
5.6.3.3	menu_print_maxim_banner	26
5.6.3.4	menu_print_maxim_banner_big	26
5.6.3.5	menu_print_prompt	26
5.6.3.6	menu_retrieve_keypress	26
5.7	src/platform.c File Reference	27
5.7.1	Macro Definition Documentation	27
5.7.1.1	UART_BAUD	27
5.7.2	Function Documentation	27
5.7.2.1	cleanup_platform	27

5.7.2.2	disable_caches	27
5.7.2.3	enable_caches	27
5.7.2.4	init_platform	28
5.7.2.5	init_uart	28
5.8	src/platform.h File Reference	28
5.8.1	Function Documentation	28
5.8.1.1	cleanup_platform	28
5.8.1.2	init_platform	28
5.9	src/platform_config.h File Reference	28
5.10	src/utilities.c File Reference	28
5.10.1	Detailed Description	29
5.10.2	Macro Definition Documentation	30
5.10.2.1	OLED_DATA_COMMAND_B	30
5.10.2.2	OLED_RESET_B	30
5.10.2.3	OLED_SCLK	30
5.10.2.4	OLED_SDIN	30
5.10.2.5	OLED_VBAT	30
5.10.2.6	OLED_VDD	30
5.10.2.7	UART_BASEADDR	30
5.10.3	Function Documentation	30
5.10.3.1	checkUartEmpty	30
5.10.3.2	clearOLEDBuffer	31
5.10.3.3	delay	31
5.10.3.4	displayOLEDBuffer	31
5.10.3.5	flipAndCopyDisplayBuffer	32
5.10.3.6	getUartByte	32
5.10.3.7	htoi	33
5.10.3.8	initializeOLED	33
5.10.3.9	led_knight_rider	33
5.10.3.10	number_raised_to_power	33
5.10.3.11	print_asterisks	34
5.10.3.12	printfToBufferOLED	34
5.10.3.13	printfToOLED	34
5.10.3.14	putCharOLED	35
5.10.3.15	suchEndOLEDSPI	35
5.10.3.16	suchEndUartByte	36
5.11	src/utilities.h File Reference	36

5.11.1 Detailed Description	37
5.11.2 Function Documentation	37
5.11.2.1 checkUartEmpty	37
5.11.2.2 clearOLEDBuffer	38
5.11.2.3 delay	38
5.11.2.4 displayOLEDBuffer	38
5.11.2.5 flipAndCopyDisplayBuffer	39
5.11.2.6 getUartByte	39
5.11.2.7 htoi	39
5.11.2.8 initializeOLED	39
5.11.2.9 led_knight_rider	40
5.11.2.10 number_raised_to_power	40
5.11.2.11 print_asterisks	41
5.11.2.12 printfToBufferOLED	41
5.11.2.13 printfToOLED	41
5.11.2.14 putcharOLED	42
5.11.2.15 sendOLEDSPI	42
5.11.2.16 sendUartByte	42
5.11.3 Variable Documentation	42
5.11.3.1 t	42

Chapter 1

Main Page

1.1 Introduction

This is the code documentation for the Sonoma (MAXREFDES14#) subsystem reference design.

The Files page contains the File List page and the Globals page.

The Globals page contains the Functions, Variables, and Macros sub-pages.

Chapter 2

Data Structure Index

2.1 Data Structures

Here are the data structures with brief descriptions:

maximDateTime	
< Structure to hold time, date, day of week	7
maximOLEDDisplay	8

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

src/ maximDeviceSpecificUtilities.c	11
src/ maximDeviceSpecificUtilities.h	13
src/ MAXREFDES14.c	14
src/ MAXREFDES14.h	17
src/ menu.c	19
src/ menu.h	22
src/ platform.c	27
src/ platform.h	28
src/ platform_config.h	28
src/ utilities.c	28
src/ utilities.h	36

Chapter 4

Data Structure Documentation

4.1 maximDateTime Struct Reference

< Structure to hold time, date, day of week

```
#include <MAXREFDES14.h>
```

Data Fields

- u8 [uchHour](#)
- u8 [uchMinute](#)
- u8 [uchSecond](#)
- u8 [uchMonth](#)
- u8 [uchDate](#)
- u8 [uchYear](#)
- u8 [uchDow](#)

4.1.1 Detailed Description

< Structure to hold time, date, day of week

Definition at line 85 of file MAXREFDES14.h.

4.1.2 Field Documentation

4.1.2.1 u8 uchDate

Definition at line 91 of file MAXREFDES14.h.

4.1.2.2 u8 uchDow

Definition at line 93 of file MAXREFDES14.h.

4.1.2.3 u8 uchHour

Definition at line 87 of file MAXREFDES14.h.

4.1.2.4 u8 uchMinute

Definition at line 88 of file MAXREFDES14.h.

4.1.2.5 u8 uchMonth

Definition at line 90 of file MAXREFDES14.h.

4.1.2.6 u8 uchSecond

Definition at line 89 of file MAXREFDES14.h.

4.1.2.7 u8 uchYear

Definition at line 92 of file MAXREFDES14.h.

The documentation for this struct was generated from the following file:

- [src/MAXREFDES14.h](#)

4.2 maximOLEDDisplay Struct Reference

```
#include <MAXREFDES14.h>
```

Data Fields

- XGpio [xgpioPort](#)
- u8 [portStatus](#)
- u8 [writeBuffer](#) [512]
- u8 [flippedBuffer](#) [512]
- u8 * [font](#)

4.2.1 Detailed Description

Definition at line 96 of file MAXREFDES14.h.

4.2.2 Field Documentation

4.2.2.1 u8 flippedBuffer[512]

Definition at line 101 of file MAXREFDES14.h.

4.2.2.2 **u8* font**

Definition at line 102 of file MAXREFDES14.h.

4.2.2.3 **u8 portStatus**

Definition at line 99 of file MAXREFDES14.h.

4.2.2.4 **u8 writeBuffer[512]**

Definition at line 100 of file MAXREFDES14.h.

4.2.2.5 **XGpio xgpioPort**

Definition at line 98 of file MAXREFDES14.h.

The documentation for this struct was generated from the following file:

- src/[MAXREFDES14.h](#)

Chapter 5

File Documentation

5.1 src/maximDeviceSpecificUtilities.c File Reference

```
#include "xbasic_types.h"
#include "xspips_hw.h"
#include "stdio.h"
#include "utilities.h"
```

Functions

- u32 [read_reg](#) (u8 uchRegAddr)
Reads the value of a register and display the data via the HyperTerminal.
- void [write_reg](#) (u8 uchRegAddr, u32 unRegValue)
Write a value to a register.

5.1.1 Detailed Description

```
Project: Sonoma (MAXREFDES14#)
Filename: maximDeviceSpecificUtilities.c
Description: This module uses the Zynq SPI peripheral to make
             custom functions for the Sonoma MAXREFDES14#. These
             low level functions could be cut/pasted into the user's
             application as a starting point for development of an end
             application.
```

Revision History:

05-31-13 Rev 01.00 GL Initial Release

This code follows the following naming conventions:

char chPmodValue

char (array) sPmodString[16]

float fPmodValue

int nPmodValue

int (array) anPmodValue[16]

u16 uPmodValue

u16 (array) auPmodValue[16]

u8 uchPmodValue

u8 (array) auchPmodBuffer[16]

unsigned int unPmodValue

int * punPmodValue

Definition in file [maximDeviceSpecificUtilities.c](#).

5.1.2 Function Documentation

5.1.2.1 u32 read_reg (u8 uchRegAddr)

Reads the value of a register and display the data via the HyperTerminal.

Details

Reads the value of a register and display the data via the HyperTerminal

Parameters

in	<i>uchRegAddr</i>	- Register Address
----	-------------------	--------------------

Return values

<i>Register</i>	Value
-----------------	-------

Definition at line 69 of file maximDeviceSpecificUtilities.c.

5.1.2.2 void write_reg (u8 uchRegAddr, u32 unRegValue)

Write a value to a register.

Details

Write a value to a register

Parameters

in	<i>uchRegAddr</i>	- Register Address
in	<i>uchRegAddr</i>	- Register Value

Return values

<i>None</i>

Definition at line 136 of file maximDeviceSpecificUtilities.c.

5.2 src/maximDeviceSpecificUtilities.h File Reference

Functions

- u32 [read_reg](#) (u8 uchRegAddr)
Reads the value of a register and display the data via the HyperTerminal.
- void [write_reg](#) (u8 uchRegAddr, u32 unRegValue)
Write a value to a register.

5.2.1 Detailed Description

```
Project: Sonoma (MAXREFDES14#)
Filename: maximDeviceSpecificUtilities.h
Description: This module uses the Zynq SPI peripheral to make
             custom functions for the Sonoma MAXREFDES14#. These
             low level functions could be cut/pasted into the user's
             application as a starting point for development of an end
             application.
```

Revision History:

05-31-13 Rev 01.00 GL Initial Release

This code follows the following naming conventions:

char chPmodValue

char (array) sPmodString[16]

float fPmodValue

int nPmodValue

int (array) anPmodValue[16]

u16 uPmodValue

u16 (array) auPmodValue[16]

u8 uchPmodValue

u8 (array) auchPmodBuffer[16]

unsigned int unPmodValue

int * punPmodValue

Definition in file [maximDeviceSpecificUtilities.h](#).

5.2.2 Function Documentation

5.2.2.1 u32 read_reg (u8 uchRegAddr)

Reads the value of a register and display the data via the HyperTerminal.

Details

Reads the value of a register and display the data via the HyperTerminal

Parameters

in	<i>uchRegAddr</i>	- Register Address
----	-------------------	--------------------

Return values

<i>Register</i>	Value
-----------------	-------

Definition at line 69 of file maximDeviceSpecificUtilities.c.

5.2.2.2 void write_reg (u8 uchRegAddr, u32 unRegValue)

Write a value to a register.

Details

Write a value to a register

Parameters

in	<i>uchRegAddr</i>	- Register Address
in	<i>uchRegAddr</i>	- Register Value

Return values

<i>None</i>

Definition at line 136 of file maximDeviceSpecificUtilities.c.

5.3 src/MAXREFDES14.c File Reference

```
#include "xparameters.h"
#include "stdio.h"
#include "xil_printf.h"
#include "platform.h"
#include "menu.h"
#include "utilities.h"
#include "maximDeviceSpecificUtilities.h"
#include "MAXREFDES14.h"
```

Macros

- #define [MAJOR_REVISION](#) 01
- #define [MINOR_REVISION](#) 00

Functions

- int [main](#) (void)
Main() function for MAXREFDES14.

Variables

- char [g_sInputString](#) [INPUT_STRING_MAX_SIZE]
- XGpio [g_xGpioLed](#)
- struct [maximOLEDDisplay](#) [g_structureOLED](#)
- char [g_sTempString](#) [32]
- u8 [font](#) [1024]

5.3.1 Detailed Description

```
Project: Sonoma (MAXREFDES14#)
Filename: MAXREFDES14.c
Description: This module contains the Main application for the
             Digilent Zedboard implementation of the MAXREFDES14
             example program.
```

Revision History:

05-31-13 Rev 01.00 GL Initial Release

This code follows the following naming conventions:

char chPmodValue

char (array) sPmodString[16]

float fPmodValue

int nPmodValue

int (array) anPmodValue[16]

u16 uPmodValue

u16 (array) auPmodValue[16]

u8 uchPmodValue

u8 (array) auchPmodBuffer[16]

unsigned int unPmodValue

int * punPmodValue

Definition in file [MAXREFDES14.c](#).

5.3.2 Macro Definition Documentation

5.3.2.1 #define MAJOR_REVISION 01

Definition at line 85 of file MAXREFDES14.c.

5.3.2.2 #define MINOR_REVISION 00

Definition at line 86 of file MAXREFDES14.c.

5.3.3 Function Documentation

5.3.3.1 `int main ( void )`

Main() function for MAXREFDES14.

Details

This function sets up and initializes the FPGA and hardware, displays the menu and sampled data via Hyperterminal (or equivalent Terminal program i.e. Teraterm).

Parameters

<i>None</i>	
-------------	--

Return values

<i>Always</i>	TRUE
---------------	------

Definition at line 226 of file MAXREFDES14.c.

5.3.4 Variable Documentation

5.3.4.1 `u8 font[1024]`

Definition at line 95 of file MAXREFDES14.c.

5.3.4.2 `char g_sInputString[INPUT_STRING_MAX_SIZE]`

Definition at line 90 of file MAXREFDES14.c.

5.3.4.3 `char g_sTempString[32]`

Definition at line 93 of file MAXREFDES14.c.

5.3.4.4 `struct maximOLEDDisplay g_structureOLED`

Definition at line 92 of file MAXREFDES14.c.

5.3.4.5 `XGpio g_xGpioLed`

Definition at line 91 of file MAXREFDES14.c.

5.4 src/MAXREFDES14.h File Reference

```
#include "xparameters.h"
#include "stdio.h"
#include "xgpio.h"
#include "xgpio_l.h"
#include "utilities.h"
#include <string.h>
#include "platform.h"
#include "math.h"
```

Data Structures

- struct [maximDateTime](#)
 < Structure to hold time, date, day of week
- struct [maximOLEDDisplay](#)

Macros

- #define [DEFAULT_HYPERTERMINAL_UART_ID](#) XPAR_PS7_UART_1_DEVICE_ID
 macro used to abstract Physical Port of Hyperterminal UART
- #define [DEFAULT_HYPERTERMINAL_UART_ADDRESS](#) XPAR_PS7_UART_1_BASEADDR
- #define [ABOUT_ONE_SECOND](#) 74067512
 approx 1 second delay when used as argument with function delay(numberCyclesToDelay)
- #define [INPUT_STRING_MAX_SIZE](#) 16

Variables

- XGpio [g_xGpioLed](#)
- u8 [g_uchAtoDConverterChannel](#)
- u32 [g_unSampleSize](#)
- int [g_nSampleRate](#)
- char [g_sInputString](#) [[INPUT_STRING_MAX_SIZE](#)]
- struct [maximOLEDDisplay](#) [g_structureOLED](#)
- char [g_tempString](#) [32]

5.4.1 Detailed Description

```
Project: Sonoma (MAXREFDES14#)
Filename: MAXREFDES14.h
Description: This module contains the Main application for the
             Digilent ZedBoard implementation of the
             MAXREFDES14 example program.
```

Revision History:

05-31-13 Rev 01.00 GL Initial Release

This code follows the following naming conventions:

char chPmodValue
char (array) sPmodString[16]
float fPmodValue
int nPmodValue
int (array) anPmodValue[16]
u16 uPmodValue
u16 (array) auPmodValue[16]
u8 uchPmodValue
u8 (array) auchPmodBuffer[16]
unsigned int unPmodValue
int * punPmodValue
Definition in file [MAXREFDES14.h](#).

5.4.2 Macro Definition Documentation

5.4.2.1 **#define ABOUT_ONE_SECOND 74067512**

approx 1 second delay when used as argument with function delay(numberCyclesToDelay)

Definition at line 79 of file MAXREFDES14.h.

5.4.2.2 **#define DEFAULT_HYPERTERMINAL_UART_ADDRESS XPAR_PS7_UART_1_BASEADDR**

Definition at line 77 of file MAXREFDES14.h.

5.4.2.3 **#define DEFAULT_HYPERTERMINAL_UART_ID XPAR_PS7_UART_1_DEVICE_ID**

macro used to abstract Physical Port of Hyperterminal UART

Definition at line 76 of file MAXREFDES14.h.

5.4.2.4 **#define INPUT_STRING_MAX_SIZE 16**

Definition at line 83 of file MAXREFDES14.h.

5.4.3 Variable Documentation

5.4.3.1 **int g_nSampleRate**

5.4.3.2 **char g_sInputString[INPUT_STRING_MAX_SIZE]**

Definition at line 90 of file MAXREFDES14.c.

5.4.3.3 struct maximOLEDDisplay g_structureOLED

Definition at line 92 of file MAXREFDES14.c.

5.4.3.4 char g_tempString[32]**5.4.3.5 u8 g_uchAtoDConverterChannel****5.4.3.6 u32 g_unSampleSize****5.4.3.7 XGpio g_xGpioLed**

Definition at line 91 of file MAXREFDES14.c.

5.5 src/menu.c File Reference

```
#include <stdio.h>
#include "platform.h"
#include "xparameters.h"
#include "utilities.h"
#include "string.h"
#include "MAXREFDES14.h"
#include "maximDeviceSpecificUtilities.h"
#include "menu.h"
```

Functions

- void [menu_cls](#) ()
Function to clear the screen via Hyperterminal.
- void [menu_print_maxim_banner](#) ()
Print standard Maxim banner at top of Hyperterminal screen.
- void [menu_print_maxim_banner_big](#) ()
Print large Maxim banner at top of Hyperterminal screen.
- void [menu_print_prompt](#) ()
Print a standard prompt for keyboard input ">".
- void [menu_print_line](#) ()
Print one line of dashes across the screen via Hyperterminal.
- u8 [menu_retrieve_keypress](#) (u32 nUartAddress)
Get a single keypress via Hyperterminal.
- void [menu_print_main_menu](#) ()
Print the main menu listing choice of module to test.

5.5.1 Detailed Description

Project: Sonoma (MAXREFDES14#)
 Filename: menu.c
 Description: This module contains all the functions used to

```
generate the menus and menu options used to run the
MAXREFDES14# example firmware.
```

Revision History:

05-31-13 Rev 01.00 GL Initial Release

This code follows the following naming conventions:

char chPmodValue

char (array) sPmodString[16]

float fPmodValue

int nPmodValue

int (array) anPmodValue[16]

u16 uPmodValue

u16 (array) auPmodValue[16]

u8 uchPmodValue

u8 (array) auchPmodBuffer[16]

unsigned int unPmodValue

int * punPmodValue

Definition in file [menu.c](#).

5.5.2 Function Documentation

5.5.2.1 void menu_cls ()

Function to clear the screen via Hyperterminal.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 70 of file menu.c.

5.5.2.2 void menu_print_line ()

Print one line of dashes across the screen via Hyperterminal.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 155 of file menu.c.

5.5.2.3 void menu_print_main_menu ()

Print the main menu listing choice of module to test.

Details.**Return values**

<i>None</i>	
-------------	--

Definition at line 189 of file menu.c.

5.5.2.4 void menu_print_maxim_banner ()

Print standard Maxim banner at top of Hyperterminal screen.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 84 of file menu.c.

5.5.2.5 void menu_print_maxim_banner_big ()

Print large Maxim banner at top of Hyperterminal screen.

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 108 of file menu.c.

5.5.2.6 void menu_print_prompt ()

Print a standard prompt for keyboard input " > ".

Parameters

<i>None</i>	
-------------	--

Return values

<i>None</i>	
-------------	--

Definition at line 141 of file menu.c.

5.5.2.7 u8 menu_retrieve_keypress (u32 nUartAddress)

Get a single keypress via Hyperterminal.

Details

Returns ascii character corresponding to keypress with some preprocessing.
Lower case "a"-"z" converted to uppercase "A"-"Z"

Parameters

<i>in</i>	<i>nUartAddress</i>	- address of the UART peripheral in MicroBlaze memory map
-----------	---------------------	---

Return values

<i>Character,partially</i>	decoded.
----------------------------	----------

Definition at line 167 of file menu.c.

5.6 src/menu.h File Reference

```
#include "xbasic_types.h"
#include "stdio.h"
#include "MAXREFDES14.h"
```

Macros

- #define **MAIN_MENU** 0
 // Menu state machine state
- #define **MENU_REG_ADDR** 1
 // Menu state machine state
- #define **MENU_READ_REG** 2
 // Menu state machine state
- #define **MENU_WRITE_REG** 3
 // Menu state machine state
- #define **WAIT_KEYPRESS** 6
 // Menu state machine state
- #define **BASE_FUNCTION_STATE** 10

- ```
// Menu state machine state
```
- #define `KEYPRESS_ARROW_UP` 240  
*Assign up-arrow an extended ascii code which won't be used elsewhere.*
  - #define `KEYPRESS_ARROW_DOWN` 241  
*Assign up-arrow an extended ascii code which won't be used elsewhere.*
  - #define `KEYPRESS_ARROW_LEFT` 242  
*Assign up-arrow an extended ascii code which won't be used elsewhere.*
  - #define `KEYPRESS_ARROW_RIGHT` 243  
*Assign up-arrow an extended ascii code which won't be used elsewhere.*
  - #define `KEYPRESS_END` 244  
*Assign up-arrow an extended ascii code which won't be used elsewhere.*

## Functions

- void `menu_cls` ()  
*Function to clear the screen via Hyperterminal.*
- void `menu_print_maxim_banner` ()  
*Print standard Maxim banner at top of Hyperterminal screen.*
- void `menu_print_maxim_banner_big` ()  
*Print large Maxim banner at top of Hyperterminal screen.*
- void `menu_print_prompt` ()  
*Print a standard prompt for keyboard input ">".*
- u8 `menu_retrieve_keypress` (u32 nUartAddress)  
*Get a single keypress via Hyperterminal.*
- void `menu_print_main_menu` ()  
*Print the main menu listing choice of module to test.*

### 5.6.1 Detailed Description

```
Project: Sonoma (MAXREFDES14#)
Filename: menu.h
Description: This module contains all the functions used to
 generate the menus and menu options used to run the
 MAXREFDES14# example firmware.
```

Revision History:

05-31-13 Rev 01.00 GL Initial Release

This code follows the following naming conventions:

char chPmodValue

char (array) sPmodString[16]

float fPmodValue

int nPmodValue

int (array) anPmodValue[16]

u16 uPmodValue

u16 (array) auPmodValue[16]

u8 uchPmodValue

u8 (array) auchPmodBuffer[16]

unsigned int unPmodValue

int \* punPmodValue

Definition in file [menu.h](#).

## 5.6.2 Macro Definition Documentation

### 5.6.2.1 **#define BASE\_FUNCTION\_STATE 10**

// Menu state machine state

Definition at line 73 of file menu.h.

### 5.6.2.2 **#define KEYPRESS\_ARROW\_DOWN 241**

Assign up-arrow an extended ascii code which won't be used elsewhere.

Definition at line 76 of file menu.h.

### 5.6.2.3 **#define KEYPRESS\_ARROW\_LEFT 242**

Assign up-arrow an extended ascii code which won't be used elsewhere.

Definition at line 77 of file menu.h.

### 5.6.2.4 **#define KEYPRESS\_ARROW\_RIGHT 243**

Assign up-arrow an extended ascii code which won't be used elsewhere.

Definition at line 78 of file menu.h.

### 5.6.2.5 **#define KEYPRESS\_ARROW\_UP 240**

Assign up-arrow an extended ascii code which won't be used elsewhere.

Definition at line 75 of file menu.h.

### 5.6.2.6 **#define KEYPRESS\_END 244**

Assign up-arrow an extended ascii code which won't be used elsewhere.

Definition at line 79 of file menu.h.

### 5.6.2.7 **#define MAIN\_MENU 0**

// Menu state machine state

Definition at line 68 of file menu.h.

#### 5.6.2.8 #define MENU\_READ\_REG 2

// Menu state machine state

Definition at line 70 of file menu.h.

#### 5.6.2.9 #define MENU\_REG\_ADDR 1

// Menu state machine state

Definition at line 69 of file menu.h.

#### 5.6.2.10 #define MENU\_WRITE\_REG 3

// Menu state machine state

Definition at line 71 of file menu.h.

#### 5.6.2.11 #define WAIT\_KEYPRESS 6

// Menu state machine state

Definition at line 72 of file menu.h.

### 5.6.3 Function Documentation

#### 5.6.3.1 void menu\_cls ( )

Function to clear the screen via Hyperterminal.

##### Parameters

|             |  |
|-------------|--|
| <i>None</i> |  |
|-------------|--|

##### Return values

|             |  |
|-------------|--|
| <i>None</i> |  |
|-------------|--|

Definition at line 70 of file menu.c.

#### 5.6.3.2 void menu\_print\_main\_menu ( )

Print the main menu listing choice of module to test.

##### Details.

##### Return values

|             |  |
|-------------|--|
| <i>None</i> |  |
|-------------|--|

Definition at line 189 of file menu.c.

### 5.6.3.3 void menu\_print\_maxim\_banner ( )

Print standard Maxim banner at top of Hyperterminal screen.

#### Parameters

|      |  |
|------|--|
| None |  |
|------|--|

#### Return values

|      |  |
|------|--|
| None |  |
|------|--|

Definition at line 84 of file menu.c.

### 5.6.3.4 void menu\_print\_maxim\_banner\_big ( )

Print large Maxim banner at top of Hyperterminal screen.

#### Parameters

|      |  |
|------|--|
| None |  |
|------|--|

#### Return values

|      |  |
|------|--|
| None |  |
|------|--|

Definition at line 108 of file menu.c.

### 5.6.3.5 void menu\_print\_prompt ( )

Print a standard prompt for keyboard input ">".

#### Parameters

|      |  |
|------|--|
| None |  |
|------|--|

#### Return values

|      |  |
|------|--|
| None |  |
|------|--|

Definition at line 141 of file menu.c.

### 5.6.3.6 u8 menu\_retrieve\_keypress ( u32 nUartAddress )

Get a single keypress via Hyperterminal.

#### Details

Returns ascii character corresponding to keypress with some preprocessing.  
Lower case "a"-"z" converted to uppercase "A"-"Z"

**Parameters**

|    |                     |                                                           |
|----|---------------------|-----------------------------------------------------------|
| in | <i>nUartAddress</i> | - address of the UART peripheral in MicroBlaze memory map |
|----|---------------------|-----------------------------------------------------------|

**Return values**

|                             |          |
|-----------------------------|----------|
| <i>Character, partially</i> | decoded. |
|-----------------------------|----------|

Definition at line 167 of file menu.c.

## 5.7 src/platform.c File Reference

```
#include "xparameters.h"
#include "xil_cache.h"
#include "platform_config.h"
```

**Macros**

- `#define UART_BAUD 9600`

**Functions**

- void `enable_caches` ()
- void `disable_caches` ()
- void `init_uart` ()
- void `init_platform` ()
- void `cleanup_platform` ()

### 5.7.1 Macro Definition Documentation

#### 5.7.1.1 `#define UART_BAUD 9600`

Definition at line 30 of file platform.c.

### 5.7.2 Function Documentation

#### 5.7.2.1 void `cleanup_platform` ( )

Definition at line 78 of file platform.c.

#### 5.7.2.2 void `disable_caches` ( )

Definition at line 49 of file platform.c.

#### 5.7.2.3 void `enable_caches` ( )

Definition at line 33 of file platform.c.

#### 5.7.2.4 void init\_platform ( )

Definition at line 71 of file platform.c.

#### 5.7.2.5 void init\_uart ( )

Definition at line 56 of file platform.c.

## 5.8 src/platform.h File Reference

```
#include "platform_config.h"
```

### Functions

- void [init\\_platform](#) ()
- void [cleanup\\_platform](#) ()

### 5.8.1 Function Documentation

#### 5.8.1.1 void cleanup\_platform ( )

Definition at line 78 of file platform.c.

#### 5.8.1.2 void init\_platform ( )

Definition at line 71 of file platform.c.

## 5.9 src/platform\_config.h File Reference

## 5.10 src/utilities.c File Reference

```
#include "xparameters.h"
#include "stdio.h"
#include "utilities.h"
#include "MAXREFDES14.h"
```

### Macros

- #define [UART\\_BASEADDR](#) XPAR\_XUARTPS\_0\_BASEADDR
- #define [OLED\\_VBAT](#) 0x20
- #define [OLED\\_VDD](#) 0x10;
- #define [OLED\\_RESET\\_B](#) 0x08
- #define [OLED\\_DATA\\_COMMAND\\_B](#) 0x04

- #define `OLED_SDIN` 0x02
- #define `OLED_SCLK` 0x01

## Functions

- void `delay` (int nStopValue)  
*Loop for nStopValue iterations to provide a delay.*
- void `led_knight_rider` (XGpio \*pLED\_GPIO, int nNumberOfTimes)  
*Blink a row of LEDs nNumberOfTimes times.*
- int `number_raised_to_power` (int nBase, int nExponent)  
*Raise nBase to the nExponent power (operates with integers only).*
- void `print_asterisks` (int nQuantity)  
*Print nQuantity of asterisks to the default Hyperterminal UART.*
- u8 `getUartByte` (u32 nUartAddress)  
*Get a byte from either the full UART in the Zynq PS, or a Xilinx UartLite.*
- void `suchEndUartByte` (u32 unUartAddress, u8 uchByte)  
*SuchEnd a byte to the UART. Either directs to the full UART in the Zynq PS, or a Xilinx UartLite.*
- u8 `checkUartEmpty` (u32 unUartAddress)  
*Check if either Uart is empty. Either directs to the full UART in the Zynq PS, or a Xilinx UartLite.*
- void `initializeOLED` (u8 \*pFont)  
*Initializes the OLED display.*
- void `suchEndOLEDSPI` (u8 uchDataToWrite)  
*Bit bang routine for Zedboard SPI OLED interface.*
- void `clearOLEDBuffer` (u8 \*pauchBuffer)  
*Clears the OLED display buffer.*
- void `displayOLEDBuffer` (u8 \*pauchBuffer)  
*Copies the display buffer into the OLED display.*
- void `putCharOLED` (int x, int y, char chCharacter)  
*Places a single ASCII character into the OLED display buffer.*
- void `printfToBufferOLED` (int x, int y, char \*chString)  
*Printf-like function to copy an ASCII string into the OLED display buffer.*
- void `printfToOLED` (int x, int y, char \*chString)  
*Printf-like function to copy an ASCII string into the OLED display buffer.*
- void `flipAndCopyDisplayBuffer` (u8 \*pauchSourceBuffer, u8 \*pauchDestinationBuffer)  
*Copies the pixels from the source display buffer, into the destination buffer, flipping their location 180 degrees.*
- int `htoi` (char \*sStr)

### 5.10.1 Detailed Description

```
Project: Maxim Plug-in Peripheral Modules
Filename: utilities.c
Description: This module contains a collection of general utility
 functions which are not specific to any particular
 module.
```

Revision History:

4-13-12 Rev 1.0 Seth Messimer Initial Release

7-20-12 Rev 1.4 Nathan Young Additional functions

This code follows the following naming conventions.

char chPmodValue char (array) sPmodString[16] float fPmodValue int nPmodValue int (array) anPmodValue[16] u16 uPmodValue u8 uchPmodValue u8 (array) auchPmodBuffer[16] unsigned int unPmodValue int \* punPmodValue

Definition in file [utilities.c](#).

## 5.10.2 Macro Definition Documentation

### 5.10.2.1 **#define OLED\_DATA\_COMMAND\_B 0x04**

Definition at line 71 of file utilities.c.

### 5.10.2.2 **#define OLED\_RESET\_B 0x08**

Definition at line 70 of file utilities.c.

### 5.10.2.3 **#define OLED\_SCLK 0x01**

Definition at line 73 of file utilities.c.

### 5.10.2.4 **#define OLED\_SDIN 0x02**

Definition at line 72 of file utilities.c.

### 5.10.2.5 **#define OLED\_VBAT 0x20**

Definition at line 68 of file utilities.c.

### 5.10.2.6 **#define OLED\_VDD 0x10;**

Definition at line 69 of file utilities.c.

### 5.10.2.7 **#define UART\_BASEADDR XPAR\_XUARTPS\_0\_BASEADDR**

Definition at line 66 of file utilities.c.

## 5.10.3 Function Documentation

### 5.10.3.1 **u8 checkUartEmpty ( u32 unUartAddress )**

Check if either Uart is empty. Either directs to the full UART in the Zynq PS, or a Xilinx UartLite.

**Parameters**

|    |                        |                     |
|----|------------------------|---------------------|
| in | <i>unUartAddress</i> . | 32 bit UART address |
|----|------------------------|---------------------|

**Return values**

|             |                         |
|-------------|-------------------------|
| <i>True</i> | if empty (false if not) |
|-------------|-------------------------|

Definition at line 215 of file utilities.c.

**5.10.3.2 void clearOLEDBuffer ( u8 \* *pauchBuffer* )**

Clears the OLED display buffer.

**Details**

This function clears the display buffer by setting all pixels to zero.

**Parameters**

|    |                     |                                                         |
|----|---------------------|---------------------------------------------------------|
| in | <i>*pauchBuffer</i> | - pointer to an u8 array used for the OLED pixel buffer |
|----|---------------------|---------------------------------------------------------|

**Return values**

|             |  |
|-------------|--|
| <i>None</i> |  |
|-------------|--|

Definition at line 415 of file utilities.c.

**5.10.3.3 void delay ( int *nStopValue* )**

Loop for *nStopValue* iterations to provide a delay.

**Details**

It is commonly used with the constant 'ABOUT\_ONE\_SECOND' defined in maximPMOD.h for setting approximate delays

**Parameters**

|    |                   |                                |
|----|-------------------|--------------------------------|
| in | <i>nStopValue</i> | - number of iterations to loop |
|----|-------------------|--------------------------------|

**Return values**

|             |  |
|-------------|--|
| <i>None</i> |  |
|-------------|--|

Definition at line 75 of file utilities.c.

**5.10.3.4 void displayOLEDBuffer ( u8 \* *pauchBuffer* )**

Copies the display buffer into the OLED display.

**Details**

This function writes the the display buffer into the OLED display using the page mode setting.

Note: The Zedboard OLED display contains a Solomon SSD1306 display controller.

**Parameters**

|           |                     |                                                         |
|-----------|---------------------|---------------------------------------------------------|
| <i>in</i> | <i>*pauchBuffer</i> | - pointer to an u8 array used for the OLED pixel buffer |
|-----------|---------------------|---------------------------------------------------------|

**Return values**

|             |
|-------------|
| <i>None</i> |
|-------------|

Definition at line 432 of file utilities.c.

**5.10.3.5 void flipAndCopyDisplayBuffer ( u8 \* pauchSourceBuffer, u8 \* pauchDestinationBuffer )**

Copies the pixels from the source display buffer, into the destination buffer, flipping their location 180 degrees.

**Details**

The (0,0) pixel location on the OLED display is in the bottom right corner. This function flips the display so that the (0,0) is in the top left corner and (131,31) is in the bottom right.

**Parameters**

|           |                                 |                                                                  |
|-----------|---------------------------------|------------------------------------------------------------------|
| <i>in</i> | <i>*pauchSource-Buffer</i>      | - pointer to an u8 buffer used as the source display buffer      |
| <i>in</i> | <i>*pauch-DestinationBuffer</i> | - pointer to an u8 buffer used as the destination display buffer |

**Return values**

|             |
|-------------|
| <i>None</i> |
|-------------|

Definition at line 586 of file utilities.c.

**5.10.3.6 u8 getUartByte ( u32 nUartAddress )**

Get a byte from either the full UART in the Zynq PS, or a Xilinx UartLite.

**Parameters**

|           |                       |                     |
|-----------|-----------------------|---------------------|
| <i>in</i> | <i>unUartAddress.</i> | 32 bit UART address |
|-----------|-----------------------|---------------------|

**Return values**

|                |                                     |
|----------------|-------------------------------------|
| <i>Returns</i> | the next byte from the Uart Rx FIFO |
|----------------|-------------------------------------|

Definition at line 174 of file utilities.c.

**5.10.3.7 int htoi ( char \* sStr )**

Definition at line 620 of file utilities.c.

**5.10.3.8 void initializeOLED ( u8 \* pFont )**

Initializes the OLED display.

**Details**

This function initializes the OLED display and the global data structure that holds the OLED display buffer and font.

**Parameters**

|    |        |                                                            |
|----|--------|------------------------------------------------------------|
| in | *pFont | - pointer to an u8 array containing the ASCII display font |
|----|--------|------------------------------------------------------------|

**Return values**

|      |
|------|
| None |
|------|

Definition at line 252 of file utilities.c.

**5.10.3.9 void led\_knight\_rider ( XGpio \* pLED\_GPIO, int nNumberOfTimes )**

Blink a row of LEDs nNumberOfTimes times.

**Details**

The Digilent NEXYS-3 and Zedboard have 8 green LEDs located above the toggle switches. This function blinks them back/forth (a bit like the KITT car from Knight Rider)

**Parameters**

|    |            |                                                                            |
|----|------------|----------------------------------------------------------------------------|
| in | *pLED_GPIO | - address of the GPIO peripheral driving the LEDs in MicroBlaze memory map |
|----|------------|----------------------------------------------------------------------------|

**Return values**

|      |
|------|
| None |
|------|

Definition at line 96 of file utilities.c.

**5.10.3.10 int number\_raised\_to\_power ( int nBase, int nExponent )**

Raise nBase to the nExponent power (operates with integers only).

**Details**

Many Microblaze applications will not have math.h included due to limited memory space. This is a simple functions to implement  $(nBase \wedge nExponent)$  Some Maxim devices (such as MAX44009) return values in mantissa + (power of 2) exponent format.

**Parameters**

|    |                  |            |
|----|------------------|------------|
| in | <i>nBase</i>     | - base     |
| in | <i>nExponent</i> | - exponent |

**Return values**

|                                 |
|---------------------------------|
| <i>Base<sup>nExponent</sup></i> |
|---------------------------------|

Definition at line 131 of file utilities.c.

**5.10.3.11 void print\_asterisks ( int *nQuantity* )**

Print nQuantity of asterisks to the default Hyperterminal UART.

**Parameters**

|    |                  |                                |
|----|------------------|--------------------------------|
| in | <i>nQuantity</i> | - number of asterisks to print |
|----|------------------|--------------------------------|

**Return values**

|             |
|-------------|
| <i>None</i> |
|-------------|

Definition at line 160 of file utilities.c.

**5.10.3.12 void printfToBufferOLED ( int *x*, int *y*, char \* *chString* )**

Printf-like function to copy an ASCII string into the OLED display buffer.

**Details**

This function copies a pre-formatted ASCII string into the OLED display buffer, but does not take any action to display the buffer

**Parameters**

|    |                  |                                                                                                  |
|----|------------------|--------------------------------------------------------------------------------------------------|
| in | <i>x</i>         | - The starting position of string within the 16x4 character display                              |
| in | <i>y</i>         | - The starting position of string within the 16x4 character display                              |
| in | <i>*chString</i> | - pointer to an null terminated character string. Should be generated with sprintf (or similar). |

**Return values**

|             |
|-------------|
| <i>None</i> |
|-------------|

Definition at line 527 of file utilities.c.

**5.10.3.13 void printfToOLED ( int *x*, int *y*, char \* *chString* )**

Printf-like function to copy an ASCII string into the OLED display buffer.

**Details**

This function copies a pre-formatted ASCII string into the OLED display buffer, flips/rotates the display pixels, and then suchEnds the buffer to the OLED display

**Parameters**

|    |                  |                                                                                                  |
|----|------------------|--------------------------------------------------------------------------------------------------|
| in | <i>x</i>         | - The starting position of string within the 16x4 character display                              |
| in | <i>y</i>         | - The starting position of string within the 16x4 character display                              |
| in | <i>*chString</i> | - pointer to an null terminated character string. Should be generated with sprintf (or similar). |

**Return values**

|      |
|------|
| None |
|------|

Definition at line 567 of file utilities.c.

**5.10.3.14 void putCharOLED ( int x, int y, char *chCharacter* )**

Places a single ASCII character into the OLED display buffer.

**Details**

This function copies an ASCII character from the font table into the OLED display buffer (configured as a 16x4 character display).

**Parameters**

|    |                    |                                                  |
|----|--------------------|--------------------------------------------------|
| in | <i>x</i>           | - The x position of the character location       |
| in | <i>y</i>           | - The y position of the character location       |
| in | <i>chCharacter</i> | - ASCII character to be printed (ex: 0x41 = 'A') |

**Return values**

|      |
|------|
| None |
|------|

Definition at line 495 of file utilities.c.

**5.10.3.15 void suchEndOLEDSPI ( u8 *uchDataToWrite* )**

Bit bang routine for Zedboard SPI OLED interface.

**Details**

This function suchEnds an (unsigned) byte (msb first) to the OLED SPI port

**Parameters**

|    |                       |                                      |
|----|-----------------------|--------------------------------------|
| in | <i>uchDataToWrite</i> | - u8 value to be written to OLED SPI |
|----|-----------------------|--------------------------------------|

**Return values**

|             |
|-------------|
| <i>None</i> |
|-------------|

Definition at line 382 of file utilities.c.

**5.10.3.16 void suchEndUartByte ( u32 unUartAddress, u8 uchByte )**

SuchEnd a byte to the UART. Either directs to the full UART in the Zynq PS, or a Xilinx UartLite.

**Parameters**

|    |                       |                                 |
|----|-----------------------|---------------------------------|
| in | <i>unUartAddress.</i> | 32 bit UART address             |
| in | <i>uchByte.</i>       | The byte to be sent to the UART |

**Return values**

|             |
|-------------|
| <i>None</i> |
|-------------|

Definition at line 195 of file utilities.c.

**5.11 src/utilities.h File Reference**

```
#include "xparameters.h"
#include "xbasic_types.h"
#include "stdio.h"
#include "xgpio.h"
#include "xgpio_l.h"
#include "MAXREFDES14.h"
```

**Functions**

- void [delay](#) (int nStopValue)  
*Loop for nStopValue iterations to provide a delay.*
- void [led\\_knight\\_rider](#) (XGpio \*pLED\_GPIO, int nNumberOfTimes)  
*Blink a row of LEDs nNumberOfTimes times.*
- int [number\\_raised\\_to\\_power](#) (int nBase, int nExponent)  
*Raise nBase to the nExponent power (operates with integers only).*
- void [print\\_asterisks](#) (int nQuantity)  
*Print nQuantity of asterisks to the default Hyperterminal UART.*
- u8 [getUartByte](#) (u32 nUartAddress)  
*Get a byte from either the full UART in the Zynq PS, or a Xilinx UartLite.*
- void [sendUartByte](#) (u32 unUartAddress, u8 uchByte)
- u8 [checkUartEmpty](#) (u32 unUartAddress)  
*Check if either Uart is empty. Either directs to the full UART in the Zynq PS, or a Xilinx UartLite.*
- void [initializeOLED](#) (u8 \*pFont)  
*Initializes the OLED display.*
- void [sendOLEDSPI](#) (u8 uchDataToWrite)

- void `clearOLEDBuffer` (u8 \*pauchBuffer)  
*Clears the OLED display buffer.*
- void `displayOLEDBuffer` (u8 \*pauchBuffer)  
*Copies the display buffer into the OLED display.*
- void `putCharOLED` (int x, int y, char chCharacter)  
*Places a single ASCII character into the OLED display buffer.*
- void `printfToBufferOLED` (int x, int y, char \*chString)  
*Printf-like function to copy an ASCII string into the OLED display buffer.*
- void `printfToOLED` (int x, int y, char \*chString)  
*Printf-like function to copy an ASCII string into the OLED display buffer.*
- void `flipAndCopyDisplayBuffer` (u8 \*pauchSourceBuffer, u8 \*pauchDestinationBuffer)  
*Copies the pixels from the source display buffer, into the destination buffer, flipping their location 180 degrees.*
- int `htoi` (char \*sStr)

## Variables

- struct `maximDateTime` \* t

### 5.11.1 Detailed Description

```

Project: Maxim Plug-in Peripheral Modules
Filename: utilities.h
Description: This module contains a collection of general utility
 functions which are not specific to any particular
 module.

```

Revision History:

4-13-12 Rev 1.0 Seth Messimer Initial Release

7-20-12 Rev 1.4 Nathan Young Additional functions

This code follows the following naming conventions.

char chPmodValue char (array) sPmodString[16] float fPmodValue int nPmodValue int (array) anPmodValue[16] u16 uPmodValue u8 uchPmodValue u8 (array) auchPmodBuffer[16] unsigned int unPmodValue int \* punPmodValue

Definition in file [utilities.h](#).

### 5.11.2 Function Documentation

#### 5.11.2.1 u8 checkUartEmpty ( u32 unUartAddress )

Check if either Uart is empty. Either directs to the full UART in the Zynq PS, or a Xilinx UartLite.

##### Parameters

|                 |                             |                     |
|-----------------|-----------------------------|---------------------|
| <code>in</code> | <code>unUartAddress.</code> | 32 bit UART address |
|-----------------|-----------------------------|---------------------|

##### Return values

|                   |                         |
|-------------------|-------------------------|
| <code>True</code> | if empty (false if not) |
|-------------------|-------------------------|

Definition at line 215 of file utilities.c.

#### 5.11.2.2 void clearOLEDBuffer ( u8 \* *pauchBuffer* )

Clears the OLED display buffer.

##### Details

This function clears the display buffer by setting all pixels to zero.

##### Parameters

|    |                      |                                                         |
|----|----------------------|---------------------------------------------------------|
| in | * <i>pauchBuffer</i> | - pointer to an u8 array used for the OLED pixel buffer |
|----|----------------------|---------------------------------------------------------|

##### Return values

|      |
|------|
| None |
|------|

Definition at line 415 of file utilities.c.

#### 5.11.2.3 void delay ( int *nStopValue* )

Loop for nStopValue iterations to provide a delay.

##### Details

It is commonly used with the constant 'ABOUT\_ONE\_SECOND' defined in maximPMOD.h for setting approximate delays

##### Parameters

|    |                   |                                |
|----|-------------------|--------------------------------|
| in | <i>nStopValue</i> | - number of iterations to loop |
|----|-------------------|--------------------------------|

##### Return values

|      |
|------|
| None |
|------|

Definition at line 75 of file utilities.c.

#### 5.11.2.4 void displayOLEDBuffer ( u8 \* *pauchBuffer* )

Copies the display buffer into the OLED display.

##### Details

This function writes the the display buffer into the OLED display using the page mode setting.  
Note: The Zedboard OLED display contains a Solomon SSD1306 display controller.

##### Parameters

|    |                      |                                                         |
|----|----------------------|---------------------------------------------------------|
| in | * <i>pauchBuffer</i> | - pointer to an u8 array used for the OLED pixel buffer |
|----|----------------------|---------------------------------------------------------|

**Return values**

|             |
|-------------|
| <i>None</i> |
|-------------|

Definition at line 432 of file *utilities.c*.

**5.11.2.5 void flipAndCopyDisplayBuffer ( u8 \* *pauchSourceBuffer*, u8 \* *pauchDestinationBuffer* )**

Copies the pixels from the source display buffer, into the destination buffer, flipping their location 180 degrees.

**Details**

The (0,0) pixel location on the OLED display is in the bottom right corner. This function flips the display so that the (0,0) is in the top left corner and (131,31) is in the bottom right.

**Parameters**

|    |                                 |                                                                  |
|----|---------------------------------|------------------------------------------------------------------|
| in | <i>*pauchSource-Buffer</i>      | - pointer to an u8 buffer used as the source display buffer      |
| in | <i>*pauch-DestinationBuffer</i> | - pointer to an u8 buffer used as the destination display buffer |

**Return values**

|             |
|-------------|
| <i>None</i> |
|-------------|

Definition at line 586 of file *utilities.c*.

**5.11.2.6 u8 getUartByte ( u32 *nUartAddress* )**

Get a byte from either the full UART in the Zynq PS, or a Xilinx UartLite.

**Parameters**

|    |                       |                     |
|----|-----------------------|---------------------|
| in | <i>unUartAddress.</i> | 32 bit UART address |
|----|-----------------------|---------------------|

**Return values**

|                |                                     |
|----------------|-------------------------------------|
| <i>Returns</i> | the next byte from the Uart Rx FIFO |
|----------------|-------------------------------------|

Definition at line 174 of file *utilities.c*.

**5.11.2.7 int htoi ( char \* *sStr* )**

Definition at line 620 of file *utilities.c*.

**5.11.2.8 void initializeOLED ( u8 \* *pFont* )**

Initializes the OLED display.

**Details**

This function initializes the OLED display and the global data structure that holds the OLED display buffer and font.

**Parameters**

|    |               |                                                            |
|----|---------------|------------------------------------------------------------|
| in | <i>*pFont</i> | - pointer to an u8 array containing the ASCII display font |
|----|---------------|------------------------------------------------------------|

**Return values**

|             |
|-------------|
| <i>None</i> |
|-------------|

Definition at line 252 of file utilities.c.

**5.11.2.9 void led\_knight\_rider ( XGpio \* pLED\_GPIO, int nNumberOfTimes )**

Blink a row of LEDs nNumberOfTimes times.

**Details**

The Digilent NEXYS-3 and Zedboard have 8 green LEDs located above the toggle switches. This function blinks them back/forth (a bit like the KITT car from Knight Rider)

**Parameters**

|    |                   |                                                                            |
|----|-------------------|----------------------------------------------------------------------------|
| in | <i>*pLED_GPIO</i> | - address of the GPIO peripheral driving the LEDs in MicroBlaze memory map |
|----|-------------------|----------------------------------------------------------------------------|

**Return values**

|             |
|-------------|
| <i>None</i> |
|-------------|

Definition at line 96 of file utilities.c.

**5.11.2.10 int number\_raised\_to\_power ( int nBase, int nExponent )**

Raise nBase to the nExponent power (operates with integers only).

**Details**

Many Microblaze applications will not have math.h included due to limited memory space. This is a simple functions to implement  $(nBase \wedge nExponent)$  Some Maxim devices (such as MAX44009) return values in mantissa + (power of 2) exponent format.

**Parameters**

|    |                  |            |
|----|------------------|------------|
| in | <i>nBase</i>     | - base     |
| in | <i>nExponent</i> | - exponent |

**Return values**

|                        |
|------------------------|
| $Base \wedge Exponent$ |
|------------------------|

Definition at line 131 of file *utilities.c*.

#### 5.11.2.11 void print\_asterisks ( int *nQuantity* )

Print *nQuantity* of asterisks to the default Hyperterminal UART.

##### Parameters

|    |                  |                                |
|----|------------------|--------------------------------|
| in | <i>nQuantity</i> | - number of asterisks to print |
|----|------------------|--------------------------------|

##### Return values

|      |
|------|
| None |
|------|

Definition at line 160 of file *utilities.c*.

#### 5.11.2.12 void printfToBufferOLED ( int *x*, int *y*, char \* *chString* )

Printf-like function to copy an ASCII string into the OLED display buffer.

##### Details

This function copies a pre-formatted ASCII string into the OLED display buffer, but does not take any action to display the buffer

##### Parameters

|    |                   |                                                                                                 |
|----|-------------------|-------------------------------------------------------------------------------------------------|
| in | <i>x</i>          | - The starting position of string within the 16x4 character display                             |
| in | <i>y</i>          | - The starting position of string within the 16x4 character display                             |
| in | * <i>chString</i> | - pointer to a null terminated character string. Should be generated with sprintf (or similar). |

##### Return values

|      |
|------|
| None |
|------|

Definition at line 527 of file *utilities.c*.

#### 5.11.2.13 void printfToOLED ( int *x*, int *y*, char \* *chString* )

Printf-like function to copy an ASCII string into the OLED display buffer.

##### Details

This function copies a pre-formatted ASCII string into the OLED display buffer, flips/rotates the display pixels, and then sends the buffer to the OLED display

##### Parameters

|    |                   |                                                                                                 |
|----|-------------------|-------------------------------------------------------------------------------------------------|
| in | <i>x</i>          | - The starting position of string within the 16x4 character display                             |
| in | <i>y</i>          | - The starting position of string within the 16x4 character display                             |
| in | * <i>chString</i> | - pointer to a null terminated character string. Should be generated with sprintf (or similar). |

**Return values**

|             |  |
|-------------|--|
| <i>None</i> |  |
|-------------|--|

Definition at line 567 of file utilities.c.

**5.11.2.14 void putCharOLED ( int x, int y, char *chCharacter* )**

Places a single ASCII character into the OLED display buffer.

**Details**

This function copies an ASCII character from the font table into the OLED display buffer (configured as a 16x4 character display).

**Parameters**

|           |                    |                                                  |
|-----------|--------------------|--------------------------------------------------|
| <i>in</i> | <i>x</i>           | - The x position of the character location       |
| <i>in</i> | <i>y</i>           | - The y position of the character location       |
| <i>in</i> | <i>chCharacter</i> | - ASCII character to be printed (ex: 0x41 = 'A') |

**Return values**

|             |  |
|-------------|--|
| <i>None</i> |  |
|-------------|--|

Definition at line 495 of file utilities.c.

**5.11.2.15 void sendOLEDSPI ( u8 *uchDataToWrite* )****5.11.2.16 void sendUartByte ( u32 *unUartAddress*, u8 *uchByte* )****5.11.3 Variable Documentation****5.11.3.1 struct maximDateTime\* t**

Definition at line 76 of file utilities.h.

# Index

BASE\_FUNCTION\_STATE  
menu.h, 24

checkUartEmpty  
utilities.c, 30  
utilities.h, 37

cleanup\_platform  
platform.c, 27  
platform.h, 28

clearOLEDBuffer  
utilities.c, 31  
utilities.h, 38

delay  
utilities.c, 31  
utilities.h, 38

disable\_caches  
platform.c, 27

displayOLEDBuffer  
utilities.c, 31  
utilities.h, 38

enable\_caches  
platform.c, 27

flipAndCopyDisplayBuffer  
utilities.c, 32  
utilities.h, 39

flippedBuffer  
maximOLEDDisplay, 8

font  
maximOLEDDisplay, 8  
MAXREFDES14.c, 16

g\_nSampleRate  
MAXREFDES14.h, 18

g\_slInputString  
MAXREFDES14.c, 16  
MAXREFDES14.h, 18

g\_sTempString  
MAXREFDES14.c, 16

g\_structureOLED  
MAXREFDES14.c, 16  
MAXREFDES14.h, 18

g\_tempString  
MAXREFDES14.h, 19

g\_uchAtoDConverterChannel  
MAXREFDES14.h, 19

g\_unSampleSize  
MAXREFDES14.h, 19

g\_xGpioLed  
MAXREFDES14.c, 16  
MAXREFDES14.h, 19

getUartByte  
utilities.c, 32  
utilities.h, 39

htoi  
utilities.c, 32  
utilities.h, 39

init\_platform  
platform.c, 27  
platform.h, 28

init\_uart  
platform.c, 28

initializeOLED  
utilities.c, 33  
utilities.h, 39

KEYPRESS\_ARROW\_DOWN  
menu.h, 24

KEYPRESS\_ARROW\_LEFT  
menu.h, 24

KEYPRESS\_ARROW\_RIGHT  
menu.h, 24

KEYPRESS\_ARROW\_UP  
menu.h, 24

KEYPRESS\_END  
menu.h, 24

led\_knight\_rider  
utilities.c, 33  
utilities.h, 40

MAIN\_MENU  
menu.h, 24

MAJOR\_REVISION  
MAXREFDES14.c, 15

MAXREFDES14.c  
font, 16  
g\_slInputString, 16

- g\_sTempString, 16
  - g\_structureOLED, 16
  - g\_xGpioLed, 16
  - main, 16
- MAXREFDES14.h
  - g\_nSampleRate, 18
  - g\_sInputString, 18
  - g\_structureOLED, 18
  - g\_tempString, 19
  - g\_uchAtoDConverterChannel, 19
  - g\_unSampleSize, 19
  - g\_xGpioLed, 19
- MENU\_READ\_REG
  - menu.h, 24
- MENU\_REG\_ADDR
  - menu.h, 25
- MENU\_WRITE\_REG
  - menu.h, 25
- MINOR\_REVISION
  - MAXREFDES14.c, 15
- main
  - MAXREFDES14.c, 16
- maximDateTime, 7
  - uchDate, 7
  - uchDow, 7
  - uchHour, 7
  - uchMinute, 8
  - uchMonth, 8
  - uchSecond, 8
  - uchYear, 8
- maximDeviceSpecificUtilities.c
  - read\_reg, 12
  - write\_reg, 12
- maximDeviceSpecificUtilities.h
  - read\_reg, 13
  - write\_reg, 14
- maximOLEDDisplay, 8
  - flippedBuffer, 8
  - font, 8
  - portStatus, 9
  - writeBuffer, 9
  - xgpioPort, 9
- menu.c
  - menu\_cls, 20
  - menu\_print\_line, 20
  - menu\_print\_main\_menu, 21
  - menu\_print\_maxim\_banner, 21
  - menu\_print\_maxim\_banner\_big, 21
  - menu\_print\_prompt, 21
  - menu\_retrieve\_keypress, 22
- menu.h
  - BASE\_FUNCTION\_STATE, 24
  - KEYPRESS\_ARROW\_DOWN, 24
  - KEYPRESS\_ARROW\_LEFT, 24
  - KEYPRESS\_ARROW\_UP, 24
  - KEYPRESS\_END, 24
  - MAIN\_MENU, 24
  - MENU\_READ\_REG, 24
  - MENU\_REG\_ADDR, 25
  - MENU\_WRITE\_REG, 25
  - menu\_cls, 25
  - menu\_print\_main\_menu, 25
  - menu\_print\_maxim\_banner, 26
  - menu\_print\_maxim\_banner\_big, 26
  - menu\_print\_prompt, 26
  - menu\_retrieve\_keypress, 26
  - WAIT\_KEYPRESS, 25
- menu\_cls
  - menu.c, 20
  - menu.h, 25
- menu\_print\_line
  - menu.c, 20
- menu\_print\_main\_menu
  - menu.c, 21
  - menu.h, 25
- menu\_print\_maxim\_banner
  - menu.c, 21
  - menu.h, 26
- menu\_print\_maxim\_banner\_big
  - menu.c, 21
  - menu.h, 26
- menu\_print\_prompt
  - menu.c, 21
  - menu.h, 26
- menu\_retrieve\_keypress
  - menu.c, 22
  - menu.h, 26
- number\_raised\_to\_power
  - utilities.c, 33
  - utilities.h, 40
- OLED\_RESET\_B
  - utilities.c, 30
- OLED\_SCLK
  - utilities.c, 30
- OLED\_SDIN
  - utilities.c, 30
- OLED\_VBAT
  - utilities.c, 30
- OLED\_VDD
  - utilities.c, 30
- platform.c
  - cleanup\_platform, 27
  - disable\_caches, 27
  - enable\_caches, 27
  - init\_platform, 27
  - init\_uart, 28

- UART\_BAUD, 27
- platform.h
  - cleanup\_platform, 28
  - init\_platform, 28
- portStatus
  - maximOLEDDisplay, 9
- print\_asterisks
  - utilities.c, 34
  - utilities.h, 41
- printfToBufferOLED
  - utilities.c, 34
  - utilities.h, 41
- printfToOLED
  - utilities.c, 34
  - utilities.h, 41
- putCharOLED
  - utilities.c, 35
  - utilities.h, 42
- read\_reg
  - maximDeviceSpecificUtilities.c, 12
  - maximDeviceSpecificUtilities.h, 13
- sendOLEDSPI
  - utilities.h, 42
- sendUartByte
  - utilities.h, 42
- src/MAXREFDES14.c, 14
- src/MAXREFDES14.h, 17
- src/maximDeviceSpecificUtilities.c, 11
- src/maximDeviceSpecificUtilities.h, 13
- src/menu.c, 19
- src/menu.h, 22
- src/platform.c, 27
- src/platform.h, 28
- src/platform\_config.h, 28
- src/utilities.c, 28
- src/utilities.h, 36
- suchEndOLEDSPI
  - utilities.c, 35
- suchEndUartByte
  - utilities.c, 36
- t
  - utilities.h, 42
- UART\_BASEADDR
  - utilities.c, 30
- UART\_BAUD
  - platform.c, 27
- uchDate
  - maximDateTime, 7
- uchDow
  - maximDateTime, 7
- uchHour
  - maximDateTime, 7
- uchMinute
  - maximDateTime, 8
- uchMonth
  - maximDateTime, 8
- uchSecond
  - maximDateTime, 8
- uchYear
  - maximDateTime, 8
- utilities.c
  - checkUartEmpty, 30
  - clearOLEDBuffer, 31
  - delay, 31
  - displayOLEDBuffer, 31
  - flipAndCopyDisplayBuffer, 32
  - getUartByte, 32
  - htoi, 32
  - initializeOLED, 33
  - led\_knight\_rider, 33
  - number\_raised\_to\_power, 33
  - OLED\_RESET\_B, 30
  - OLED\_SCLK, 30
  - OLED\_SDIN, 30
  - OLED\_VBAT, 30
  - OLED\_VDD, 30
  - print\_asterisks, 34
  - printfToBufferOLED, 34
  - printfToOLED, 34
  - putCharOLED, 35
  - suchEndOLEDSPI, 35
  - suchEndUartByte, 36
  - UART\_BASEADDR, 30
- utilities.h
  - checkUartEmpty, 37
  - clearOLEDBuffer, 38
  - delay, 38
  - displayOLEDBuffer, 38
  - flipAndCopyDisplayBuffer, 39
  - getUartByte, 39
  - htoi, 39
  - initializeOLED, 39
  - led\_knight\_rider, 40
  - number\_raised\_to\_power, 40
  - print\_asterisks, 41
  - printfToBufferOLED, 41
  - printfToOLED, 41
  - putCharOLED, 42
  - sendOLEDSPI, 42
  - sendUartByte, 42
  - t, 42
- WAIT\_KEYPRESS
  - menu.h, 25
- write\_reg

- maximDeviceSpecificUtilities.c, [12](#)
  - maximDeviceSpecificUtilities.h, [14](#)
- writeBuffer
  - maximOLEDDisplay, [9](#)
- xgpioPort
  - maximOLEDDisplay, [9](#)