

Utilities library for AVR uC

Version : 1.0

Generated by Doxygen 1.7.4

Sat Jul 30 2011 12:51:38

Contents

1	Deprecated List	1
2	File Documentation	1
2.1	utils.h File Reference	1
2.1.1	Detailed Description	2
2.1.2	License	3
2.1.3	Target	3
2.1.4	Define Documentation	3
2.1.5	Function Documentation	4

1 Deprecated List

Global `nop()` Should use `_NOP` from `avr/cpufunc.h`

2 File Documentation

2.1 utils.h File Reference

Useful define's & functions for Atmel AVR.

```
#include <avr/io.h>
#include <stdbool.h>
```

Defines

I/O Macros to simplifies the #define of I/O pins

Example

```
#include "utils.h"
#define LED_PIN 1
#define LED_PORT B
...
DDR(LED_PORT) |= (1 << LED_PIN);           // DDRB |= (1 << 1)
PORT(LED_PORT) |= (1 << LED_PIN);           // PORTB |= (1 << 1)
PIN(LED_PORT)  |= (1 << LED_PIN);           // PINB  |= (1 << 1)
...
```

See also

<http://www.avrfreaks.net/index.php?name=PNphpBB2&file=viewtopic&t=73829>

- #define **DDR_**(port) DDR ## port
First macro for DDR.
- #define **DDR**(port) DDR_(port)
Second macro for DDR.
- #define **PORT_**(port) PORT ## port
First macro for PORT.
- #define **PORT**(port) PORT_(port)
Second macro for PORT.
- #define **PIN_**(port) PIN ## port
First macro for PIN.
- #define **PIN**(port) PIN_(port)
Second macro for PIN.

Enable/disable something

- #define **enable** true
- #define **disable** false

Simple macros

- #define **BIT**(n) (1 << (n))
To select bit 0-7
Not used, for backward compatibility.
- #define **nop**() __asm__ __volatile__ ("nop")
Skip a clock cycle.

Functions

- void **util_ByteString** (char *str, uint8_t data)
Convert the binary value of an unsigned 8 bits integer to a string.
- uint8_t **utils_BcdToDec** (uint8_t number)
Convert an 8 bits BCD value to its decimal representation.
- uint8_t **utils_DecToBcd** (uint8_t number)
Convert a value to its BCD representation.
- void **utils_DoubleByteString** (char *str, uint16_t data)
Convert the binary value of an unsigned 16 bits integer to a string.

2.1.1 Detailed Description

Useful define's & functions for Atmel AVR.

2.1.2 License

Disclaimer Copyright (c) 2010, 2011 Patrice Nadeau

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of Patrice Nadeau nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL Patrice Nadeau BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

2.1.3 Target

- ATmega

Author

Patrice Nadeau

email : patricen@telwarwick.net

www : <http://nadeaup.homeip.net:8080/>

2.1.4 Define Documentation

2.1.4.1 #define nop() __asm__ __volatile__ ("nop")

Skip a clock cycle.

Deprecated

Should use `_NOP` from `avr/cpufunc.h`

2.1.5 Function Documentation

2.1.5.1 `void util_ByteString (char * str, uint8_t data)`

Convert the binary value of an unsigned 8 bits integer to a string.

Parameters

in	<i>data</i>	Date of type <i>uint8_t</i>
out	<i>*str</i>	Pointer to a string

Warning

Possible buffer overflow

2.1.5.2 `uint8_t utils_BcdToDec (uint8_t number)`

Convert an 8 bits BCD value to its decimal representation.

2 x 4-digits that represent numbers from 0 to 9

Parameters

<i>number</i>	BCD value to convert
---------------	----------------------

Returns

8bits decimal number

Warning

No validation is done if 4-digit is > 1001

See also

http://en.wikipedia.org/wiki/Binary-coded_decimal

2.1.5.3 `uint8_t utils_DecToBcd (uint8_t number)`

Convert a value to its BCD representation.

2 x 4-digits that represent numbers from 0 to 9

Parameters

in	<i>number</i>	Value to convert of type <i>uint8_t</i>
----	---------------	---

Returns

BCD number in a *uint8_t* type

Warning

No validation is done if number is > 9

See also

http://en.wikipedia.org/wiki/Binary-coded_decimal

2.1.5.4 void utils_DoubleByteString (char * *str*, uint16_t *data*)

Convert the binary value of an unsigned 16 bits integer to a string.

Parameters

in	<i>data</i>	Data of type <i>uint16_t</i>
out	<i>*str</i>	Pointer to a string

Warning

Possible buffer overflow

Index

`nop`

`utils.h`, [3](#)

`util_ByteString`

`utils.h`, [3](#)

`utils.h`, [1](#)

`nop`, [3](#)

`util_ByteString`, [3](#)

`utils_BcdToDec`, [4](#)

`utils_DecToBcd`, [4](#)

`utils_DoubleByteString`, [4](#)

`utils_BcdToDec`

`utils.h`, [4](#)

`utils_DecToBcd`

`utils.h`, [4](#)

`utils_DoubleByteString`

`utils.h`, [4](#)