

2nd Year's Eirbot Display for the Robotics French Cup

Generated by Doxygen 1.9.1

| | |
|-----------------------------------|-----------|
| 1 File Index | 1 |
| 1.1 File List | 1 |
| 2 File Documentation | 3 |
| 2.1 caracters.cpp File Reference | 3 |
| 2.2 functions.cpp File Reference | 3 |
| 2.2.1 Detailed Description | 4 |
| 2.2.2 Function Documentation | 4 |
| 2.2.2.1 buf_alloc() | 4 |
| 2.2.2.2 buf_free() | 5 |
| 2.2.2.3 buf_init() | 5 |
| 2.2.2.4 buf_realloc() | 5 |
| 2.2.2.5 clear_matrix() | 6 |
| 2.2.2.6 flicker() | 6 |
| 2.2.2.7 inversion() | 6 |
| 2.2.2.8 print_on_column() | 7 |
| 2.2.2.9 print_with_scroll() | 7 |
| 2.2.2.10 random_stuffs() | 8 |
| 2.2.2.11 scrolling() | 8 |
| 2.2.2.12 scrolling_reverse() | 8 |
| 2.2.2.13 test_all_points() | 9 |
| 2.3 main.cpp File Reference | 9 |
| 2.3.1 Detailed Description | 9 |
| 2.4 pixel_arts.cpp File Reference | 9 |
| 2.4.1 Detailed Description | 10 |
| 2.4.2 Function Documentation | 10 |
| 2.4.2.1 aliens() | 10 |
| 2.4.2.2 amogus() | 10 |
| 2.4.2.3 battle_aliens() | 11 |
| 2.4.2.4 laser() | 11 |
| 2.4.2.5 noot() | 11 |
| 2.4.2.6 noot_noot() | 12 |
| 2.4.2.7 pokeball() | 12 |
| 2.4.2.8 sus() | 12 |
| Index | 13 |

Chapter 1

File Index

1.1 File List

Here is a list of all documented files with brief descriptions:

| | |
|--------------------------------|----|
| characters.cpp | 3 |
| characters.hpp | ?? |
| functions.cpp | 3 |
| functions.hpp | ?? |
| main.cpp | 9 |
| pixel_arts.cpp | 9 |
| pixel_arts.hpp | ?? |

Chapter 2

File Documentation

2.1 characters.cpp File Reference

```
#include <MD_MAX72xx.h>
#include <SPI.h>
#include "functions.hpp"
#include "characters.hpp"
#include <string.h>
```

Include dependency graph for characters.cpp:

2.2 functions.cpp File Reference

```
#include "functions.hpp"
#include "characters.hpp"
#include "pixel_arts.hpp"
#include <MD_MAX72xx.h>
#include <SPI.h>
```

Include dependency graph for functions.cpp:

Macros

- `#define BASE_DELAY 70`
Define the base delay used when scrolling is applied on the display.
- `#define FLICKER_DELAY 50`
Define the delay used when flicker is applied on the display.
- `#define INVERSION_DELAY 500`
Define the delay used when pixel inversion is applied on the display.

Functions

- void [test_all_points](#) (MD_MAX72XX *M)
This fonction tests all dots of the displat starting from line 0 and column 0.
- void [scrolling](#) (MD_MAX72XX *M, int c_min, int c_max)
Function that applies one in which one iteration of right to left scrolling on an MD_MAX72XX display.
- void [scrolling_reverse](#) (MD_MAX72XX *M, int c_min, int c_max)
Function that applies one in which one iteration of right to left scrolling on a MD_MAX72XX display.
- int [test_column](#) (MD_MAX72XX *M, int c)
- void [inversion](#) (MD_MAX72XX *M, int l_min, int l_max, int c_min, int c_max)
Function that does one iteration of inversion of pixels on a MD_MAX72XX display.
- void [flicker](#) (MD_MAX72XX *M, int c_min, int c_max)
Function that makes the display flicker one time.
- void [print_with_scroll](#) (MD_MAX72XX *M, int *buf, int way, int c_min, int c_max)
Function that prints a buffer on a display in scrooling mod.
- void [print_on_column](#) (MD_MAX72XX *M, int *buf, int c)
Function that prints a buffer on a display in fixed mod.
- void [clear_matrix](#) (MD_MAX72XX *M, int c_min, int c_max)
Function that clears the display.
- int * [buf_alloc](#) (int width)
*Function that allocates dynamically a buffer of size $8 * \text{width} * \text{sizeof}(\text{int}) + 1$, this correspond to the datas of a drawing of a size of width column and his size.*
- void [buf_realloc](#) (int *buffer, int width)
*Function that changes the size of the dynamically allocated buffer to $8 * \text{width} * \text{sizeof}(\text{int}) + 1$.*
- void [buf_init](#) (int *buf, int width)
Function that entirely sets a buffer to 0.
- void [buf_free](#) (int *buf)
Function that frees the memory associated with the buffer.
- void [random_stuffs](#) (MD_MAX72XX *M)
Function that displays random stuffs on the display based on functions implemented in [characters.cpp](#) and [pixel_arts.cpp](#).

2.2.1 Detailed Description

File in which there are functions for general manipulation for a MD_MAX72XX display

2.2.2 Function Documentation

2.2.2.1 [buf_alloc\(\)](#)

```
int* buf_alloc (
    int width )
```

Function that allocates dynamically a buffer of size $8 * \text{width} * \text{sizeof}(\text{int}) + 1$, this correspond to the datas of a drawing of a size of width column and his size.

Parameters

| | |
|--------------|---|
| <i>width</i> | the maximum width that can be used to draw (typically the number of columns of the drawing) |
|--------------|---|

Returns

int* the buffer

2.2.2.2 buf_free()

```
void buf_free (
    int * buf )
```

Function that frees the memory associated with the buffer.

Parameters

| | |
|------------|-------------------------------|
| <i>buf</i> | the buffer that will be freed |
|------------|-------------------------------|

2.2.2.3 buf_init()

```
void buf_init (
    int * buf,
    int width )
```

Function that entirely sets a buffer to 0.

Parameters

| | |
|--------------|--|
| <i>buf</i> | the buffer that will be set to 0 |
| <i>width</i> | the size of the buffer that will be stored in buf[0] |

2.2.2.4 buf_realloc()

```
void buf_realloc (
    int * buffer,
    int width )
```

Function that changes the size of the dynamically allocated buffer to $8 * \text{width} * \text{sizeof(int)} + 1$.

Parameters

| | |
|---------------|---|
| <i>buffer</i> | the buffer which will be reallocated |
| <i>width</i> | the maximum width that can be use to draw (typically the number of column of the drawing) |

2.2.2.5 clear_matrix()

```
void clear_matrix (
    MD_MAX72XX * M,
    int c_min,
    int c_max )
```

Function that clears the display.

Parameters

| | |
|--------------|--|
| <i>M</i> | a MD_MAX72XX* that refers to the MD_MAX72XX display that will be cleared |
| <i>c_min</i> | the minimum column from which the erasure will begin |
| <i>c_max</i> | the maximum column from which the erasure will end |

2.2.2.6 flicker()

```
void flicker (
    MD_MAX72XX * M,
    int c_min,
    int c_max )
```

Function that makes the display flicker one time.

Parameters

| | |
|--------------|---|
| <i>M</i> | a MD_MAX72XX* that refers to the MD_MAX72XX display that will flicker |
| <i>c_min</i> | the minimum column in which the flicker will occur |
| <i>c_max</i> | the maximum column in which the flicker will occur |

2.2.2.7 inversion()

```
void inversion (
    MD_MAX72XX * M,
    int l_min,
    int l_max,
```

```
int c_min,  
int c_max )
```

Function that does one iteration of inversion of pixels on a MD_MAX72XX display.

Parameters

| | |
|--------------|---|
| <i>M</i> | a MD_MAX72XX* that refers to the MD_MAX72XX display in which the inversion is applied |
| <i>l_min</i> | the minimum line in which the inversion will be applied |
| <i>l_max</i> | the maximum line in which the inversion will be applied |
| <i>c_min</i> | the minimum column in which the inversion will be applied |
| <i>c_max</i> | the maximum column in which the inversion will be applied |

2.2.2.8 print_on_column()

```
void print_on_column (  
    MD_MAX72XX * M,  
    int * buf,  
    int c )
```

Function that prints a buffer on a display in fixed mod.

Parameters

| | |
|------------|---|
| <i>M</i> | a MD_MAX72XX* that refers to the MD_MAX72XX display in which the buffer will be print out |
| <i>buf</i> | a buffer that stores the datas that will be print out |
| <i>c</i> | the column from which the display of datas will begin |

2.2.2.9 print_with_scroll()

```
void print_with_scroll (  
    MD_MAX72XX * M,  
    int * buf,  
    int way,  
    int c_min,  
    int c_max )
```

Function that prints a buffer on a display in scrolling mod.

Parameters

| | |
|--------------|---|
| <i>M</i> | a MD_MAX72XX* that refers to the MD_MAX72XX display in which the buffer will be print out |
| <i>buf</i> | a buffer that stores the datas that will be print out |
| <i>way</i> | the way a the scroll |
| <i>c_min</i> | the minimum column of the scroll |
| <i>c_max</i> | the maximum column of the scroll |

2.2.2.10 random_stuffs()

```
void random_stuffs (
    MD_MAX72XX * M )
```

Function that displays random stuffs on the display based on functions implemented in [characters.cpp](#) and [pixel_arts.cpp](#).

Parameters

| | |
|----------|---|
| <i>M</i> | a MD_MAX72XX* that refers to the MD_MAX72XX display in which this will be displayed |
|----------|---|

2.2.2.11 scrolling()

```
void scrolling (
    MD_MAX72XX * M,
    int c_min,
    int c_max )
```

Function that applies one in which one iteration of right to left scrolling on an MD_MAX72XX display.

Parameters

| | |
|--------------|--|
| <i>M</i> | a MD_MAX72XX* that refers to the MD_MAX72XX display in which the scrolling is applied on |
| <i>c_max</i> | |

2.2.2.12 scrolling_reverse()

```
void scrolling_reverse (
    MD_MAX72XX * M,
    int c_min,
    int c_max )
```

Function that applies one in which one iteration of right to left scrolling on a MD_MAX72XX display.

Parameters

| | |
|--------------|---|
| <i>M</i> | a MD_MAX72XX* that refers to the MD_MAX72XX display in which the reverse scrolling is applied |
| <i>c_min</i> | |
| <i>c_max</i> | |

2.2.2.13 test_all_points()

```
void test_all_points (
    MD_MAX72XX * M )
```

This fonction tests all dots of the displat starting from line 0 and column 0.

Parameters

| | |
|----------|--|
| <i>M</i> | a MD_MAX72XX* that referes to the MD_MAX72XX display in which the dots will be print out |
|----------|--|

2.3 main.cpp File Reference

```
#include <Arduino.h>
#include "characters.hpp"
#include "functions.hpp"
#include "pixel_arts.hpp"
#include <MD_MAX72xx.h>
#include <SPI.h>
```

Include dependency graph for main.cpp:

Macros

- `#define Max7219_pinCLK 2`
- `#define Max7219_pinCS 3`
- `#define Max7219_pinDIN 4`

Functions

- `void setup ()`
- `void loop ()`

Variables

- `MD_MAX72XX * M = new MD_MAX72XX(M->FC16_HW, 4, 2, 3, 4)`

2.3.1 Detailed Description

Main file that stores the main program that will be executed the the arduino

2.4 pixel_arts.cpp File Reference

```
#include <MD_MAX72xx.h>
#include <SPI.h>
#include <stdlib.h>
#include "pixel_arts.hpp"
#include "functions.hpp"
#include "characters.hpp"
Include dependency graph for pixel_arts.cpp:
```

Functions

- `int * pokeball ()`
Function that returns the draw of a pokeball.
- `int * laser (int length)`
Function that returns the drawing of a laser.
- `int * aliens (int type)`
Function that returns the drawing of an alien.
- `void battle_aliens (MD_MAX72XX *M)`
Function that generates a battle between alien 1 and 2, they will shoot 5x3 laser alternatively until a winner is randomly chosen.
- `int * amogus ()`
Function that returns the drawing of a crew member of a famous game.
- `void sus (MD_MAX72XX *M)`
Function that draws a famous crew with his iconic line.
- `int * noot ()`
function that returns the drawing of a famous penguin
- `void noot_noot (MD_MAX72XX *M)`
Function that draws a famous penguin and his line.

2.4.1 Detailed Description

File in which there are functions that generates and draws some pixel-arts for/on a MD_MAX72XX display

2.4.2 Function Documentation

2.4.2.1 aliens()

```
int* aliens (
    int type )
```

Function that returns the drawing of an alien.

Parameters

| | |
|-------------|-------------------------------|
| <i>type</i> | the type of the alien, 1 or 2 |
|-------------|-------------------------------|

Returns

`int*` the buffer that stores the datas of the alien drawing

2.4.2.2 amogus()

```
int* amogus ( )
```

Function that returns the drawing of a crew member of a famous game.

Returns

int* the buffer that stores the datas of the crew member

2.4.2.3 battle_aliens()

```
void battle_aliens (
    MD_MAX72XX * M )
```

Function that generates a battle between alien 1 and 2, they will shoot 5x3 laser alternatively until a winner is randomly chosen.

Parameters

| | |
|----------|---|
| <i>M</i> | a MD_MAX72XX* that refers to the MD_MAX72XX display in which the battle will be displayed |
|----------|---|

2.4.2.4 laser()

```
int* laser (
    int length )
```

Function that returns the drawing of a laser.

Parameters

| | |
|---------------|----------------------------------|
| <i>length</i> | the lenght of the 3xlenght laser |
|---------------|----------------------------------|

Returns

int* the buffer that stores the datas of the laser drawing

2.4.2.5 noot()

```
int* noot ( )
```

function that returns the drawing of a famous penguin

Returns

int* the buffer that stores the datas of the crew penguin

2.4.2.6 noot_noot()

```
void noot_noot (
    MD_MAX72XX * M )
```

Function that draws a famous penguin and his line.

Parameters

| | |
|----------|---|
| <i>M</i> | a MD_MAX72XX* that refers to the MD_MAX72XX display in which this will be displayed |
|----------|---|

2.4.2.7 pokeball()

```
int* pokeball ( )
```

Function that returns the draw of a pokeball.

Returns

int* the buffer that stores the datas of the drawing

2.4.2.8 sus()

```
void sus (
    MD_MAX72XX * M )
```

Function that draws a famous crew with his iconic line.

Parameters

| | |
|----------|---|
| <i>M</i> | a MD_MAX72XX* that refers to the MD_MAX72XX display in which this will be displayed |
|----------|---|

Index

- aliens
 - pixel_arts.cpp, 10
- amogus
 - pixel_arts.cpp, 10
- battle_aliens
 - pixel_arts.cpp, 11
- buf_alloc
 - functions.cpp, 4
- buf_free
 - functions.cpp, 5
- buf_init
 - functions.cpp, 5
- buf_realloc
 - functions.cpp, 5
- characters.cpp, 3
- clear_matrix
 - functions.cpp, 6
- flicker
 - functions.cpp, 6
- functions.cpp, 3
 - buf_alloc, 4
 - buf_free, 5
 - buf_init, 5
 - buf_realloc, 5
 - clear_matrix, 6
 - flicker, 6
 - inversion, 6
 - print_on_column, 7
 - print_with_scroll, 7
 - random_stuffs, 8
 - scrolling, 8
 - scrolling_reverse, 8
 - test_all_points, 8
- inversion
 - functions.cpp, 6
- laser
 - pixel_arts.cpp, 11
- main.cpp, 9
- noot
 - pixel_arts.cpp, 11
- noot_noot
 - pixel_arts.cpp, 11
- pixel_arts.cpp, 9
 - aliens, 10
 - amogus, 10
 - battle_aliens, 11
 - laser, 11
 - noot, 11
 - noot_noot, 11
 - pokeball, 12
 - sus, 12
- pokeball
 - pixel_arts.cpp, 12
- print_on_column
 - functions.cpp, 7
- print_with_scroll
 - functions.cpp, 7
- random_stuffs
 - functions.cpp, 8
- scrolling
 - functions.cpp, 8
- scrolling_reverse
 - functions.cpp, 8
- sus
 - pixel_arts.cpp, 12
- test_all_points
 - functions.cpp, 8