IEEE-754 Visualiser

Generated by Doxygen 1.9.8

1 File Index	1
1.1 File List	1
2 File Documentation	3
2.1 main.c File Reference	3
2.1.1 Macro Definition Documentation	4
2.1.1.1 prec1	4
2.1.1.2 prec2	4
2.1.1.3 STR_2	4
2.1.1.4 VAR_TO_STR	4
2.1.2 Function Documentation	4
2.1.2.1 bit_print()	4
2.1.2.2 decode_mantissa()	5
2.1.2.3 main()	5
2.1.2.4 visualise()	5
Index	7

# **Chapter 1**

# File Index

1.1 File List	t
---------------	---

Here is a list of all files with brief descriptions:	
main.c	

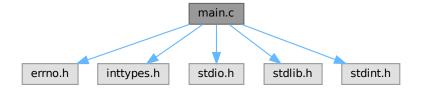
2 File Index

## **Chapter 2**

## **File Documentation**

## 2.1 main.c File Reference

```
#include <errno.h>
#include <inttypes.h>
#include <stdio.h>
#include <stdlib.h>
#include <stdint.h>
Include dependency graph for main.c:
```



### **Macros**

• #define STR\_2(X) #X

Auxiliary macro used by VAR\_TO\_STR.

• #define VAR\_TO\_STR(X) STR\_2(X)

Returns a literal with the parameter value.

• #define prec1 9

The number of digits after the decimal point for float output.

• #define prec2 16

The number of digits after the decimal point for double output.

4 File Documentation

#### **Functions**

• double decode mantissa (uint64 t u, int size)

Converts mantissa's bit representation to double.

· void bit\_print (uint64\_t const u, int const exp, int const mant)

Prints floating point number bit representation.

• void visualise (uint64\_t const u, unsigned long const size)

Prints the canonical representation of a floating point number and its bit representation.

int main (int argc, char \*\*argv)

The function accepts a floating point number and visualizes its float and double representations if the number fits within those data types.

### 2.1.1 Macro Definition Documentation

### 2.1.1.1 prec1

```
#define prec1 9
```

The number of digits after the decimal point for float output.

#### 2.1.1.2 prec2

```
#define prec2 16
```

The number of digits after the decimal point for double output.

## 2.1.1.3 STR\_2

```
#define STR_2( _{\it X} ) #X
```

Auxiliary macro used by VAR\_TO\_STR.

## 2.1.1.4 VAR\_TO\_STR

Returns a literal with the parameter value.

## 2.1.2 Function Documentation

### 2.1.2.1 bit\_print()

Prints floating point number bit representation.

Highlights Sign, Exp and Frac sections.

2.1 main.c File Reference 5

#### **Parameters**

и	Floating point number bit representation
exp	Size of exp section
mant	Size of mantissa section

## 2.1.2.2 decode\_mantissa()

```
double decode_mantissa ( \label{eq:code_mantissa} \mbox{ uint64\_t } \mbox{ u,} \\ \mbox{ int } \mbox{ size )}
```

Converts mantissa's bit representation to double.

### **Parameters**

и	Mantissa's bit representation
size	Size of the mantissa

### Returns

Mantissa value.

## 2.1.2.3 main()

```
int main (
                int argc,
                char ** argv )
```

The function accepts a floating point number and visualizes its float and double representations if the number fits within those data types.

## 2.1.2.4 visualise()

```
void visualise ( \label{eq:uint64_tonst} \mbox{uint64\_t const } \mbox{\it u,} \\ \mbox{unsigned long const } \mbox{\it size} \mbox{\it )}
```

Prints the canonical representation of a floating point number and its bit representation.

#### **Parameters**

и	Floating point number bit representation
size	The size of the number in bits.

6 File Documentation

## Warning

This function is only compatible with systems where floating point numbers are 32 and 64 bits in size.

## Index

```
bit_print
    main.c, 4
decode_mantissa
    main.c, 5
main
    main.c, 5
main.c, 3
    bit_print, 4
    decode_mantissa, 5
    main, 5
    prec1, 4
    prec2, 4
    STR_2, 4
    VAR_TO_STR, 4
    visualise, 5
prec1
    main.c, 4
prec2
    main.c, 4
STR_2
    main.c, 4
VAR_TO_STR
    main.c, 4
visualise
    main.c, 5
```