

**mathieeesingbas**

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> mathieeesingbas		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		March 29, 2025	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>mathieeesingbas</b>	<b>1</b>
1.1	mathieeesingbas.doc . . . . .	1
1.2	mathieeesingbas.library/IEEESPAbs . . . . .	1
1.3	mathieeesingbas.library/IEEESPAdd . . . . .	2
1.4	mathieeesingbas.library/IEEESPCeil . . . . .	2
1.5	mathieeesingbas.library/IEEESPCmp . . . . .	3
1.6	mathieeesingbas.library/IEEESPDiv . . . . .	3
1.7	mathieeesingbas.library/IEEESPFix . . . . .	4
1.8	mathieeesingbas.library/IEEESPFloor . . . . .	4
1.9	mathieeesingbas.library/IEEESPFlt . . . . .	5
1.10	mathieeesingbas.library/IEEESPMul . . . . .	5
1.11	mathieeesingbas.library/IEEESPNeg . . . . .	6
1.12	mathieeesingbas.library/IEEESPSub . . . . .	6
1.13	mathieeesingbas.library/IEEESPSt . . . . .	7

## Chapter 1

# mathieeesingbas

### 1.1 mathieeesingbas.doc

```
IEEESPAbs ()  
IEEESPAdd ()  
IEEESPCeil ()  
IEEESPCmp ()  
IEEESPDiv ()  
IEEESPFix ()  
IEEESPFloor ()  
IEEESPFlt ()  
IEEESPMul ()  
IEEESPNeg ()  
IEEESPSub ()  
IEEESPTst ()
```

### 1.2 mathieeesingbas.library/IEEESPAbs

NAME

IEEESPAbs -- compute absolute value of IEEE single precision argument

SYNOPSIS

```
x    = IEEESPAbs( y );  
d0      d0
```

```
float x,y;
```

FUNCTION

Take the absolute value of argument y and return it to caller.

INPUTS

y -- IEEE single precision floating point value

RESULT

x -- IEEE single precision floating point value

BUGS

---

SEE ALSO

### 1.3 mathieeesingbas.library/IEEESPAdd

NAME

IEEESPAdd -- add one single precision IEEE number to another

SYNOPSIS

```
x    = IEEESPAdd( y , z );  
d0    d0    d1
```

```
float x,y,z;
```

FUNCTION

Compute  $x = y + z$  in IEEE single precision.

INPUTS

y -- IEEE single precision floating point value

z -- IEEE single precision floating point value

RESULT

x -- IEEE single precision floating point value

BUGS

SEE ALSO

IEEESPSub

### 1.4 mathieeesingbas.library/IEEESPCeil

NAME

IEEESPCeil -- compute Ceil function of IEEE single precision number

SYNOPSIS

```
x    = IEEESPCeil( y );  
d0    d0
```

```
float x,y;
```

FUNCTION

Calculate the least integer greater than or equal to x and return it.  
This identity is true.  $\text{Ceil}(x) = -\text{Floor}(-x)$ .

INPUTS

y -- IEEE single precision floating point value

RESULT

x -- IEEE single precision floating point value

BUGS

SEE ALSO

---

IEEESPFloor

## 1.5 mathieeesingbas.library/IEEESPCmp

NAME

IEEESPCmp -- compare two single precision floating point numbers

SYNOPSIS

```
c    = IEEESPCmp( y , z );
d0      d0      d1
```

```
float y,z;
long  c;
```

FUNCTION

Compare y with z. Set the condition codes for less, greater, or equal. Set return value c to -1 if y<z, or +1 if y>z, or 0 if y == z.

INPUTS

y -- IEEE single precision floating point value  
z -- IEEE single precision floating point value

RESULT

```
c = 1    cc = gt      for (y > z)
c = 0    cc = eq      for (y == z)
c = -1   cc = lt      for (y < z)
```

BUGS

SEE ALSO

## 1.6 mathieeesingbas.library/IEEESPDiv

NAME

IEEESPDiv -- divide one single precision IEEE by another

SYNOPSIS

```
x    = IEEESPDiv( y , z );
d0      d0      d1
```

```
float x,y,z;
```

FUNCTION

Compute  $x = y / z$  in IEEE single precision.  
Note that the Motorola fast floating point Div routine reverses the order of the arguments for the C interface, although the dividend is still in d0 and the divisor is in d1.

INPUTS

y -- IEEE single precision floating point value  
z -- IEEE single precision floating point value

RESULT  
x -- IEEE single precision floating point value

BUGS

SEE ALSO  
IEEESPMul

## 1.7 mathieeesingbas.library/IEEESPFix

NAME  
IEEESPFix -- convert IEEE single float to integer

SYNOPSIS  
x = IEEESPFix( y );  
d0 d0

long x;  
float y;

FUNCTION  
Convert IEEE single precision argument to a 32 bit signed integer and return result.

INPUTS  
y -- IEEE single precision floating point value

RESULT  
if no overflow occurred then return  
x -- 32 bit signed integer  
if overflow return largest +- integer  
For round to zero

BUGS

SEE ALSO  
IEEESPFlt

## 1.8 mathieeesingbas.library/IEEESPFloor

NAME  
IEEESPFloor -- compute Floor function of IEEE single precision number

SYNOPSIS  
x = IEEESPFloor( y );  
d0 d0

float x,y;

FUNCTION  
Calculate the largest integer less than or equal to x and return it.

---

INPUTS  
y -- IEEE single precision floating point value

RESULT  
x -- IEEE single precision floating point value

BUGS

SEE ALSO  
IEEESPCeil

## 1.9 mathieeesingbas.library/IEEESPFlt

NAME  
IEEESPFlt -- convert integer to IEEE single precision number

SYNOPSIS  
x = IEEESPFlt( y );  
d0 d0

float x;  
long y;

FUNCTION  
Convert a signed 32 bit value to a single precision IEEE value and return it in d0. No exceptions can occur with this function.

INPUTS  
y -- 32 bit integer in d0

RESULT  
x is a 32 bit single precision IEEE value

BUGS

SEE ALSO  
IEEESPFix

## 1.10 mathieeesingbas.library/IEEESPMul

NAME  
IEEESPMul -- multiply one double precision IEEE number by another

SYNOPSIS  
x = IEEESPMul( y , z );  
d0 d0 d1

float x,y,z;

FUNCTION

---



Compute  $x = y * z$  in IEEE single precision.

#### INPUTS

y -- IEEE single precision floating point value  
z -- IEEE single precision floating point value

#### RESULT

x -- IEEE single precision floating point value

#### BUGS

#### SEE ALSO

IEEEESPDiv

## 1.11 mathieeesingbas.library/IEEESPNeg

#### NAME

IEEESPNeg -- compute negative value of IEEE single precision number

#### SYNOPSIS

```
x    = IEEESPNeg( y );
d0          d0
```

```
float x,y;
```

#### FUNCTION

Invert the sign of argument y and return it to caller.

#### INPUTS

y - IEEE single precision floating point value

#### RESULT

x - IEEE single precision floating point value

#### BUGS

#### SEE ALSO

## 1.12 mathieeesingbas.library/IEEESPSub

#### NAME

IEEESPSub -- subtract one single precision IEEE number from another

#### SYNOPSIS

```
x    = IEEESPSub( y , z );
d0          d0    d1
```

```
float x,y,z;
```

#### FUNCTION

Compute  $x = y - z$  in IEEE single precision.

INPUTS  
y -- IEEE single precision floating point value  
z -- IEEE single precision floating point value

RESULT  
x -- IEEE single precision floating point value

BUGS

SEE ALSO  
IEEESPAdd

## 1.13 mathieeesingbas.library/IEEESPTst

NAME  
IEEESPTst -- compare IEEE single precision value to 0.0

SYNOPSIS  
c = IEEESPTst( y );  
d0 d0

float y;  
long c;

FUNCTION  
Compare y to 0.0, set the condition codes for less than, greater than, or equal to 0.0. Set the return value c to -1 if less than, to +1 if greater than, or 0 if equal to 0.0.

INPUTS  
y -- IEEE single precision floating point value

RESULT  
c = 1 cc = gt for (y > 0.0)  
c = 0 cc = eq for (y == 0.0)  
c = -1 cc = lt for (y < 0.0)

BUGS

SEE ALSO

---