

**mathieeedoubtrans**

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

	<i>TITLE :</i> mathieeedoubtrans		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		March 28, 2025	

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

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>mathieeedoubtrans</b>	<b>1</b>
1.1	mathieeedoubtrans.doc . . . . .	1
1.2	mathieeedoubtrans.library/IEEEEDPAcos . . . . .	1
1.3	mathieeedoubtrans.library/IEEEEDPAsin . . . . .	1
1.4	mathieeedoubtrans.library/IEEEEDPAtan . . . . .	2
1.5	mathieeedoubtrans.library/IEEEEDPCos . . . . .	2
1.6	mathieeedoubtrans.library/IEEEEDPCosh . . . . .	3
1.7	mathieeedoubtrans.library/IEEEEDPExp . . . . .	3
1.8	mathieeedoubtrans.library/IEEEEDPFieee . . . . .	4
1.9	mathieeedoubtrans.library/IEEEEDPLog . . . . .	4
1.10	mathieeedoubtrans.library/IEEEEDPLog10 . . . . .	5
1.11	mathieeedoubtrans.library/IEEEEDPPow . . . . .	5
1.12	mathieeedoubtrans.library/IEEEEDPSin . . . . .	6
1.13	mathieeedoubtrans.library/IEEEEDPSincos . . . . .	6
1.14	mathieeedoubtrans.library/IEEEEDPSinh . . . . .	7
1.15	mathieeedoubtrans.library/IEEEEDPSqrt . . . . .	7
1.16	mathieeedoubtrans.library/IEEEEDPTan . . . . .	8
1.17	mathieeedoubtrans.library/IEEEEDPTanh . . . . .	8
1.18	mathieeedoubtrans.library/IEEEEDPTieee . . . . .	9

## Chapter 1

# mathieeedoubtrans

### 1.1 mathieeedoubtrans.doc

IEEEDPAcos()	IEEEDPExp()	IEEEDPSin()	IEEEDPTanh()
IEEEDPAsin()	IEEEDPFieee()	IEEEDPSincos()	IEEEDPTieee()
IEEEDPAtan()	IEEEDPLog()	IEEEDPSinh()	
IEEEDPCos()	IEEEDPLog10()	IEEEDPSqrt()	
IEEEDPCosh()	IEEEDPPow()	IEEEDPTan()	

### 1.2 mathieeedoubtrans.library/IEEEDPAcos

#### NAME

IEEEDPAcos -- compute the arc cosine of a number

#### SYNOPSIS

```
x      = IEEEDPAcos( y );
d0/d1      d0/d1
```

```
double  x,y;
```

#### FUNCTION

Compute arc cosine of y in IEEE double precision

#### INPUTS

y - IEEE double precision floating point value

#### RESULT

x - IEEE double precision floating point value

#### BUGS

#### SEE ALSO

IEEEDPCos(), IEEEDPAtan(), IEEEDPAsin()

### 1.3 mathieeedoubtrans.library/IEEEDPAsin

---

## NAME

IEEEEDPAsin -- compute the arcsine of a number

## SYNOPSIS

```
    x    = IEEEEDPAsin( y );  
    d0/d1      d0/d1
```

```
double x,y;
```

## FUNCTION

Compute the arc sine of y in IEEE double precision

## INPUTS

y - IEEE double precision floating point value

## RESULT

x - IEEE double precision floating point value

## BUGS

## SEE ALSO

IEEEEDPSin(), IEEEEDPAtan(), IEEEEDPACos()

## 1.4 mathieeedoubtrans.library/IEEEEDPAtan

## NAME

IEEEEDPAtan -- compute the arctangent of a floating point number

## SYNOPSIS

```
    x    = IEEEEDPAtan( y );  
    d0/d1      d0/d1
```

```
double x,y;
```

## FUNCTION

Compute arctangent of y in IEEE double precision

## INPUTS

y - IEEE double precision floating point value

## RESULT

x - IEEE double precision floating point value

## BUGS

## SEE ALSO

IEEEEDPTan(), IEEEEDPAsin(), IEEEEDPACos()

## 1.5 mathieeedoubtrans.library/IEEEEDPCos

## NAME

IEEEEDPCos -- compute the cosine of a floating point number

#### SYNOPSIS

```
    x    = IEEEEDPCos( y );  
d0/d1      d0/d1
```

```
double x,y;
```

#### FUNCTION

Compute cosine of y in IEEE double precision

#### INPUTS

y - IEEE double precision floating point value

#### RESULT

x - IEEE double precision floating point value

#### BUGS

#### SEE ALSO

IEEEEDPAcos(), IEEEEDPSin(), IEEEEDPTan()

## 1.6 mathieeedoubtrans.library/IEEEEDPCosh

#### NAME

IEEEEDPCosh -- compute the hyperbolic cosine of a floating point number

#### SYNOPSIS

```
    x    = IEEEEDPCosh( y );  
d0/d1      d0/d1
```

```
double x,y;
```

#### FUNCTION

Compute hyperbolic cosine of y in IEEE double precision

#### INPUTS

y - IEEE double precision floating point value

#### RESULT

x - IEEE double precision floating point value

#### BUGS

#### SEE ALSO

IEEEEDPSinh(), IEEEEDPTanh()

## 1.7 mathieeedoubtrans.library/IEEEEDPExp

#### NAME

IEEEEDPExp -- compute the exponential of e

---

## SYNOPSIS

```

    x      = IEEEEDPExp(  y  );
    d0/d1      d0/d1

```

```

double  x,y;

```

## FUNCTION

Compute  $e^y$  in IEEE double precision

## INPUTS

y - IEEE double precision floating point value

## RESULT

x - IEEE double precision floating point value

## BUGS

## SEE ALSO

IEEEEDPLog()

## 1.8 mathieeedoubtrans.library/IEEEDPFieee

## NAME

IEEEDPFieee -- convert IEEE single to IEEE double

## SYNOPSIS

```

    x      = IEEEDPFieee(  y  );
    d0/d1      d0

```

```

float  y;
double x;

```

## FUNCTION

Convert IEEE single precision number to IEEE double precision.

## INPUTS

y - IEEE single precision floating point value

## RESULT

x - IEEE double precision floating point value

## BUGS

## SEE ALSO

IEEEEDPTieee()

## 1.9 mathieeedoubtrans.library/IEEEDPLog

## NAME

IEEEDPLog -- compute the natural logarithm of a floating point number

## SYNOPSIS

```

    x    = IEEEEDPLog( y );
d0/d1      d0/d1

```

```
double x,y;
```

#### FUNCTION

Compute  $\ln(y)$  in IEEE double precision

#### INPUTS

y - IEEE double precision floating point value

#### RESULT

x - IEEE double precision floating point value

#### BUGS

#### SEE ALSO

IEEEEDPExp()

## 1.10 mathieeedoubtrans.library/IEEEEDPLog10

#### NAME

IEEEEDPLog10 -- compute logarithm base 10 of a number

#### SYNOPSIS

```

    x    = IEEEEDPLog10( y );
d0/d1      d0/d1

```

```
double x,y;
```

#### FUNCTION

Compute the logarithm base 10 of y in IEEE double precision

#### INPUTS

y - IEEE double precision floating point value

#### RESULT

x - IEEE double precision floating point value

#### BUGS

#### SEE ALSO

IEEEEDPLog()

## 1.11 mathieeedoubtrans.library/IEEEEDPPow

#### NAME

IEEEEDPPow -- raise a number to another number power

#### SYNOPSIS

```

    z    = IEEEEDPPow( x , y );
d0/d1      d2/d3 d0/d1

```



```
double x,y,z;
```

#### FUNCTION

Compute  $y^x$  in IEEE double precision

#### INPUTS

x - IEEE double precision floating point value  
y - IEEE double precision floating point value

#### RESULT

z - IEEE double precision floating point value

#### BUGS

#### SEE ALSO

## 1.12 mathieeedoubtrans.library/IEEEDPSin

#### NAME

IEEEDPSin -- compute the sine of a floating point number

#### SYNOPSIS

```
x      = IEEEDPSin( y );
d0/d1      d0/d1
```

```
double x,y;
```

#### FUNCTION

Compute sine of y in IEEE double precision

#### INPUTS

y - IEEE double precision floating point value

#### RESULT

x - IEEE double precision floating point value

#### BUGS

#### SEE ALSO

IEEEDPAsin(), IEEEDPTan(), IEEEDPCos()

## 1.13 mathieeedoubtrans.library/IEEEDPSincos

#### NAME

IEEEDPSincos -- compute the arc tangent of a floating point number

#### SYNOPSIS

```
x      = IEEEDPSincos( z , y );
d0/d1      a0 d0/d1
```

```
double x,y,*z;
```

**FUNCTION**

Compute sin and cosine of y in IEEE double precision.  
Store the cosine in \*z. Return the sine of y.

**INPUTS**

y - IEEE double precision floating point value  
z - pointer to IEEE double precision floating point number

**RESULT**

x - IEEE double precision floating point value

**BUGS****SEE ALSO**

IEEEEDPSin(), IEEEEDPCos()

## 1.14 mathieeedoubtrans.library/IEEEEDPSinh

**NAME**

IEEEEDPSinh -- compute the hyperbolic sine of a floating point number

**SYNOPSIS**

```
x      = IEEEEDPSinh( y );  
d0/d1      d0/d1
```

```
double x,y;
```

**FUNCTION**

Compute hyperbolic sine of y in IEEE double precision

**INPUTS**

y - IEEE double precision floating point value

**RESULT**

x - IEEE double precision floating point value

**BUGS****SEE ALSO**

IEEEEDPCosh, IEEEEDPTanh

## 1.15 mathieeedoubtrans.library/IEEEEDPSqrt

**NAME**

IEEEEDPSqrt -- compute the square root of a number

**SYNOPSIS**

```
x      = IEEEEDPSqrt( y );  
d0/d1      d0/d1
```

```
double x,y;
```

---

FUNCTION  
    Compute square root of y in IEEE double precision

INPUTS  
    y - IEEE double precision floating point value

RESULT  
    x - IEEE double precision floating point value

BUGS

SEE ALSO

## 1.16 mathieeedoubtrans.library/IEEEDPTan

NAME  
    IEEEDPTan -- compute the tangent of a floating point number

SYNOPSIS  
    x = IEEEDPTan( y );  
    d0/d1           d0/d1  
  
    double x,y;

FUNCTION  
    Compute tangent of y in IEEE double precision

INPUTS  
    y - IEEE double precision floating point value

RESULT  
    x - IEEE double precision floating point value

BUGS

SEE ALSO  
    IEEEDPatan(), IEEEDPSin(), IEEEDPCos()

## 1.17 mathieeedoubtrans.library/IEEEDPTanh

NAME  
    IEEEDPTanh -- compute the hyperbolic tangent of a floating point number

SYNOPSIS  
    x = IEEEDPTanh( y );  
    d0/d1           d0/d1  
  
    double x,y;

FUNCTION  
    Compute hyperbolic tangent of y in IEEE double precision

---

## INPUTS

y - IEEE double precision floating point value

## RESULT

x - IEEE double precision floating point value

## BUGS

## SEE ALSO

IEEEEDPSinh(), IEEEEDPCosh()

## 1.18 mathieeedoubtrans.library/IEEEEDPTieee

## NAME

IEEEEDPTieee -- convert IEEE double to IEEE single

## SYNOPSIS

```
x    = IEEEEDPTieee( y );
d0           d0/d1
```

```
double y;
float  x;
```

## FUNCTION

Convert IEEE double precision number to IEEE single precision.

## INPUTS

y - IEEE double precision floating point value

## RESULT

x - IEEE single precision floating point value

## BUGS

## SEE ALSO

IEEEEDPFieee()