

RX Family

R20AN0228JJ0101

Rev.1.00

M3S-FIXMATH-LIB: 固定小数点ライブラリ

2012.10.30

本ドキュメントでは、RX 用固定小数点ライブラリについて説明します。

目次

1.	固定小数点ライブラリ	2
1.1	概要	2
1.2	固定小数点数の形式	2
1.3	提供ライブラリ	3
1.4	使用例	3
1.5	ライブラリ使用時の注意事項	4
2.	固定小数点ライブラリの詳細	5
2.1	"fixmath.h"	5
2.2	各関数の説明	10
3.	性能	19
3.1	測定条件	19
3.2	実行サイクル数	19
	サポート窓口	20

1. 固定小数点ライブラリ

1.1 概要

RX シリーズでの実数の演算を 固定小数点数(*1)のデータとして扱うライブラリを提供します。

固定小数点ライブラリは、特に浮動小数点ユニットを持たない CPU において、高速な実数演算をサポートします。本ライブラリは、小数部が 16、24、29 ビットの固定小数点数の型に対する以下の関数をサポートしています。

- (1) 乗除算
- (2) sin、cos、atan、sqrt の関数
- (3) 浮動小数点数との変換

アプリケーションの要求する精度に合わせて 16 ビットまたは 24 ビットの精度を選択してください。29 ビット精度は、三角関数を使用するときに必要な $-\pi \sim +\pi$ の範囲を表現できる最大の精度を提供しており、精度の高い演算が必要な場合に有用です。

また、固定小数点では、浮動小数点とことなり、表現できる値の範囲が限定されるため、演算の途中で入力値、出力値の範囲を表現できる型を使用する必要があります。このため、本ライブラリでは、乗除算、型変換に対しては小数部が 1 ビットから 31 ビットまでのすべての固定小数点型をサポートしています。

*1：固定小数点数とは、小数点が特定の位置に固定されている数値の表現手法です。

1.2 固定小数点数の形式

本ライブラリでサポートする固定小数点数の基本的な形式は下記の通りです。

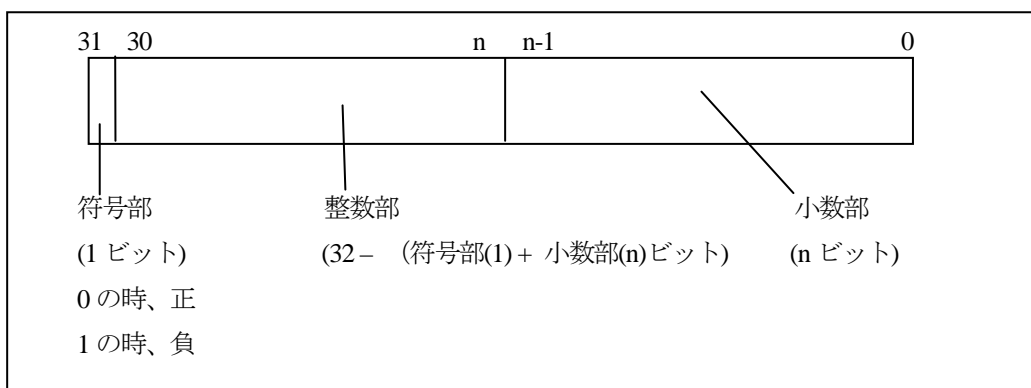


図 1-1 固定小数点数の形式

小数部のビット数に応じて FIX1 から FIX31 までの型をサポートしています。

F I X<n>の n の部分が小数部のビット数を表しています。

また、FIX1 から FIX31 を総称する型として FIX 型をサポートします。この型のデータに対しては、小数部ビット数を指定しない一般的な固定小数点演算をサポートします。

1.3 提供ライブラリ

下表に示すインクルードファイルとライブラリを提供します。

本ライブラリを使用するときは、表 1-1に示すファイルをインクルードしてください。

また、表 1-2に示すように、コンパイラオプションに対応するライブラリをリンクしてください。

表 1-1 固定小数点ライブラリ用のインクルードファイル

ライブラリの種類	内容	インクルードファイル
固定小数点ライブラリ	固定小数点演算を行うライブラリです。	"fixmath.h"

表 1-2 固定小数点ライブラリ一覧

ライブラリ名	コンパイラオプション	
	cpu	endian
RXfixmath_big.lib	RX	big
RXfixmath_little.lib	RX	little

各々、任意のフォルダ下にコピーしてご使用ください。

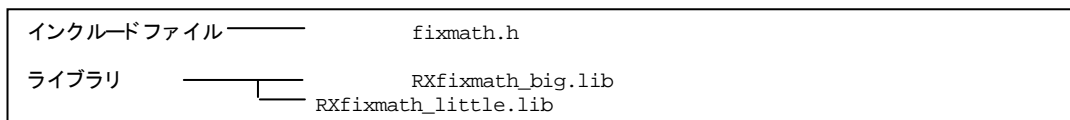


図 1-2 構成例

1.4 使用例

FIX16 型での演算のソース例と High-performance Embedded Workshop でのライブラリ設定方法を示します。

[ソース]

```

#include <stdio.h>
#include "fixmath.h" // 固定小数点ライブラリを使用する場合、インクルード必須

void main()
{
    float rflt;
    FIX16 d_fix16, r_fix16;

    d_fix16 = FIX16_fromfloat(3.14f); // float 型から FIX16 型へ変換
    r_fix16 = FIX16_sin(d_fix16); // 正弦値を求める
    rflt = FIX16_tofloat(r_fix16); // FIX16 型から float 型へ変換
    printf("%f\n", rflt);
}
  
```

[High-performance Embedded Workshop でのライブラリの設定方法]

[ビルド]メニューの[RX Standard Toolchain ...]を選択して表示される、[RX Standard Toolchain ...]ダイアログボックスにて、[最適化リンカ]タブを選択して、[カテゴリ]に“入力”、[オプション項目]に“ライブラリファイル”を選択し、[追加]ボタンをクリックしてください。表示された[Add library file]ダイアログボックスで、リンクするライブラリを指定してください。

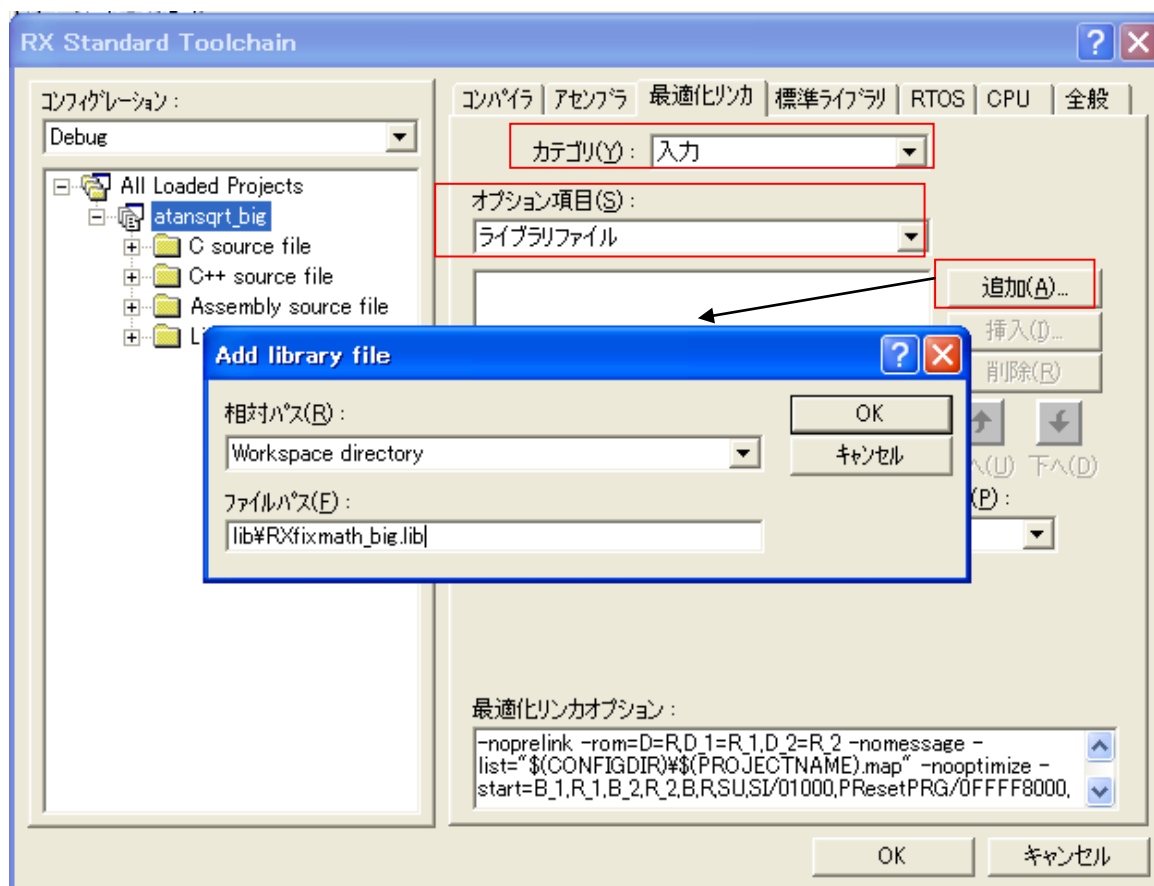


図 1-3 ライブラリの設定方法

1.5 ライブラリ使用時の注意事項

演算結果や変換結果が値の範囲を超えた場合、結果は保証されません。

2. 固定小数点ライブラリの詳細

2.1 "fixmath.h"

固定小数点数値の各種演算を行います。

本ファイルで定義している型およびサポート関数の対応一覧を表 2-1に示します。

記法: 型、関数またはマクロ名の中の $\langle n \rangle$ は、1 から 31 までの数字を表し、FIX 型の小数点以下のビット数を表します。関数またはマクロ名の中の数字は、型名の中の数字と対応しています。

表 2-1 型およびサポート 関数の対応一覧

型	サポート 関数及び関数(マクロ)
FIX1~FIX31 (FIX16, FIX24, FIX29 を除く)	FIX $\langle n \rangle$ _mul_short, FIX $\langle n \rangle$ _mul, FIX $\langle n \rangle$ _div, FIX $\langle n \rangle$ _tfloat, FIX $\langle n \rangle$ _fromfloat, FIX $\langle n \rangle$ _todouble, FIX $\langle n \rangle$ _fromdouble, FIX $\langle n \rangle$ _mul_frac, FIX $\langle n \rangle$ _mul_sat
FIX16, FIX24, FIX29	FIX $\langle n \rangle$ _mul_short, FIX $\langle n \rangle$ _mul, FIX $\langle n \rangle$ _div, FIX $\langle n \rangle$ _tfloat, FIX $\langle n \rangle$ _fromfloat, FIX $\langle n \rangle$ _todouble, FIX $\langle n \rangle$ _fromdouble, FIX $\langle n \rangle$ _mul_frac, FIX $\langle n \rangle$ _mul_sat, FIX $\langle n \rangle$ _sin, FIX $\langle n \rangle$ _cos, FIX $\langle n \rangle$ _atan, FIX $\langle n \rangle$ _sqrt
FIX	FIX_mul_scale $\langle n \rangle$, FIX_mul_frac_scale $\langle n \rangle$, FIX_mul_sat_scale $\langle n \rangle$, FIX_mul_scale, FIX_mul_frac_scale, FIX_mul_sat_scale

内部的には long 型として定義しています。

固定小数点型 FIX $\langle n \rangle$ の演算で、オペランドの型と結果の型が一致する場合は、その型に対応する関数を使用します。一方、オペランドの型どうし、あるいはオペランドと結果の型が異なる場合は総称固定小数点型 FIX に対する関数を用いてください。

[固定小数点プログラミングのヒント]

- (1) アプリケーションが必要とする精度に合わせて、FIX16 または FIX24 を選択してください。
- (2) 固定小数点は、浮動小数点に対して表現できる範囲が限定されます。入力値の範囲、演算の途中結果の範囲、必要になる演算の精度に合わせて適切な型を選択する必要がある場合があります。
- (3) 異なる型の固定小数点の間の変換は、C 言語のシフト演算で行ってください。

例: FIX16 から FIX24 への変換.

```
FIX16 x, FIX24 y;
x=y>>8;
```

(4) 同じ型の固定小数点の間の加減算は、C 言語の加減算で行ってください。

例: FIX16 同士の加算

```
FIX16 x, y, z;
```

```
z=x+y;
```

(5) 浮動小数点と固定小数点との間の変換は必要な場合だけ実行してください。固定小数点による高速化の効果が得られません。ただし、プログラム内で固定小数点定数を指定するために変換関数を使用する場合は、マクロにより定数式に変換されるため、オーバーヘッドはありません。

例: 固定小数点の定数指定。

```
FIX16 x;
```

```
x=FIX16_fromfloat(3.14f);
```

各々の型の内部表現を表 2-2に示します。

表 2-2 固定小数点数型の内部表現

型	サイズ (byte)	アライメント数 (byte)	符号の有無	値の範囲	
				最小値	最大値
FIX1	4	4	有	$-2^{30}(-1073741824.0)$	$2^{30} \cdot 2^{-1}(1073741823.5)$
FIX2	4	4	有	$-2^{29}(-536870912.0)$	$2^{29} \cdot 2^{-2}(536870911.75)$
FIX3	4	4	有	$-2^{28}(-268435456.0)$	$2^{28} \cdot 2^{-3}(268435455.875)$
FIX4	4	4	有	$-2^{27}(-134217728.0)$	$2^{27} \cdot 2^{-4}(134217727.9375)$
FIX5	4	4	有	$-2^{26}(-67108864.0)$	$2^{26} \cdot 2^{-5}(67108863.96875)$
FIX6	4	4	有	$-2^{25}(-33554432.0)$	$2^{25} \cdot 2^{-6}(33554431.984375)$
FIX7	4	4	有	$-2^{24}(-16777216.0)$	$2^{24} \cdot 2^{-7}(16777215.9921875)$
FIX8	4	4	有	$-2^{23}(-8388608.0)$	$2^{23} \cdot 2^{-8}(8388607.99609375)$
FIX9	4	4	有	$-2^{22}(-4194304.0)$	$2^{22} \cdot 2^{-9}(4194303.998046875)$
FIX10	4	4	有	$-2^{21}(-2097152.0)$	$2^{21} \cdot 2^{-10}(2097151.9990234375)$
FIX11	4	4	有	$-2^{20}(-1048576.0)$	$2^{20} \cdot 2^{-11}(1048575.99951171875)$
FIX12	4	4	有	$-2^{19}(-524288.0)$	$2^{19} \cdot 2^{-12}(524287.999755859375)$
FIX13	4	4	有	$-2^{18}(-262144.0)$	$2^{18} \cdot 2^{-13}(262143.9998779296875)$
FIX14	4	4	有	$-2^{17}(-131072.0)$	$2^{17} \cdot 2^{-14}(131071.99993896484375)$
FIX15	4	4	有	$-2^{16}(-65536.0)$	$2^{16} \cdot 2^{-15}(65535.999969482421875)$
FIX16	4	4	有	$-2^{15}(-32768.0)$	$2^{15} \cdot 2^{-16}(32767.9999847412109375)$
FIX17	4	4	有	$-2^{14}(-16384.0)$	$2^{14} \cdot 2^{-17}(16383.99999237060546875)$
FIX18	4	4	有	$-2^{13}(-8192.0)$	$2^{13} \cdot 2^{-18}(8191.999996185302734375)$
FIX19	4	4	有	$-2^{12}(-4096.0)$	$2^{12} \cdot 2^{-19}(4095.9999980926513671875)$
FIX20	4	4	有	$-2^{11}(-2048.0)$	$2^{11} \cdot 2^{-20}(2047.99999904632568359375)$
FIX21	4	4	有	$-2^{10}(-1024.0)$	$2^{10} \cdot 2^{-21}(1023.999999523162841796875)$
FIX22	4	4	有	$-2^9(-512.0)$	$2^9 \cdot 2^{-22}(511.9999997615814208984375)$
FIX23	4	4	有	$-2^8(-256.0)$	$2^8 \cdot 2^{-23}(255.99999988079071044921875)$
FIX24	4	4	有	$-2^7(-128.0)$	$2^7 \cdot 2^{-24}(127.999999940395355224609375)$
FIX25	4	4	有	$-2^6(-64.0)$	$2^6 \cdot 2^{-25}(63.9999999701976776123046875)$
FIX26	4	4	有	$-2^5(-32.0)$	$2^5 \cdot 2^{-26}(31.99999998509883880615234375)$
FIX27	4	4	有	$-2^4(-16.0)$	$2^4 \cdot 2^{-27}(15.999999992549419403076171875)$
FIX28	4	4	有	$-2^3(-8.0)$	$2^3 \cdot 2^{-28}(7.9999999962747097015380859375)$
FIX29	4	4	有	$-2^2(-4.0)$	$2^2 \cdot 2^{-29}(3.99999999813735485076904296875)$
FIX30	4	4	有	$-2^1(-2.0)$	$2^1 \cdot 2^{-30}(1.999999999068677425384521484375)$
FIX31	4	4	有	$-2^0(-1.0)$	$2^0 \cdot 2^{-31}(0.9999999995343387126922607421875)$
FIX	4	4	有	想定する小数点以下のビット数により上記のどれかの範囲をとります。	

関数 (マクロ) の一覧を表 2-3に示します。

表 2-3 関数(マクロ) 一覧

項目	関数(マクロ) 名	引数の型	リターン値の型	説明
乗算	FIX<n>_mul_short	FIX<n> n=1~31	FIX<n> n=1~31	固定小数点値の乗算を計算します。 (演算の途中結果が32ビット以上のものは結果が保証されません)
	FIX<n>_mul_frac	FIX<n> n=1~31	FIX<n> n=1~31	固定小数点値の乗算結果の小数部分 f (0<=f<1.0) を求めます。
	FIX<n>_mul_sat	FIX<n> n=1~31	FIX<n> n=1~31	固定小数点値の乗算を計算し、オーバフローした場合は最大値または最小値を返します。
除算	FIX<n>_div	FIX<n> n=1~31	FIX<n> n=1~31	固定小数点の除算を計算します。
変換	FIX<n>_tofloat	FIX<n> n=1~31	FIX<n> n=1~31	FIX<n> 型を float 型へ変換します。
	FIX<n>_fromfloat	FIX<n> n=1~31	FIX<n> n=1~31	float 型を FIX<n>型へ変換します。
	FIX<n>_todouble	FIX<n> n=1~31	FIX<n> n=1~31	FIX<n>型を double 型へ変換します。
	FIX<n>_fromdouble	FIX<n> n=1~31	FIX<n> n=1~31	double 型を FIX<n>型へ変換します。
総称固定小数点の乗算	FIX_mul_scale<n>	FIX	FIX	総称固定小数点の乗算を計算します。
	FIX_mul_frac_scale<n>	FIX	FIX	総称固定小数点の乗算結果の小数部分を計算します。
	FIX_mul_sat_scale<n>	FIX	FIX	総称固定小数点の乗算を計算し、オーバフローの場合は最大値または最小値を返します。
	FIX_mul_frac_scale	FIX	FIX	総称固定小数点の乗算結果の小数部分を計算します。

値の範囲を超える演算結果は不定となります。値の範囲を超えないようご注意ください。

関数の一覧を表 2-4に示します。

表 2-4 関数一覧

項目	関数名	引数の型	リターン値の型	説明
乗算	FIX_mul	FIX	FIX	固定小数点値の乗算を計算します。
正弦関数	FIX<n>_sin	FIX<n> n=16, 24, 29	FIX<n> n=16, 24, 29	固定小数点値のラジアン値の正弦を計算します。
余弦関数	FIX<n>_cos	FIX<n> n=16, 24, 29	FIX<n> n=16, 24, 29	固定小数点値のラジアン値の余弦を計算します。
逆正接関数	FIX<n>_atan	FIX<n> n=16, 24, 29	FIX<n> n=16, 24, 29	固定小数点値のラジアン値の逆正接を計算します。
平方根関数	FIX<n>_sqrt	FIX<n> n=16, 24, 29	FIX<n> n=16, 24, 29	固定小数点値の正の平方根を計算します。
総称固定小数点の乗算	FIX_mul_scale	FIX	FIX	総称固定小数点の乗算を計算します。
	FIX_mul_sat_scale	FIX	FIX	総称固定小数点の乗算を計算し、オーバーフローの場合は最大値または最小値を返します。

値の範囲を超える演算結果は不定となります。値の範囲を超えないようご注意ください。

2.2 各関数の説明

2.2.1 乗算関数（マクロ）

- 【書式】 `FIX<n> FIX_mul_short(FIX<n> x, FIX<n> y)`
n: 1~31
- 【説明】 2つの32ビットデータの乗算結果を、nビット右シフトをすることで、FIX<n>型の固定小数点数の乗算をします。
- 【ヘッダ】 "fixmath.h"
- 【リターン値】 乗算結果
- 【引数】 x 固定小数点値
Y 固定小数点値
- 【例】

```
#include "fixmath.h"
FIX16 x, y, ret;
ret = FIX16_mul_short( x, y );
```
- 【備考】 演算の途中結果が32ビット以上のものは結果が保証されません。

2.2.2 除算関数

- 【書式】 `FIX<n> FIX<n>_div(FIX<n> x, FIX<n> y)`
n: 1~31
- 【説明】 固定小数点値の除算を計算します。
- 【ヘッダ】 "fixmath.h"
- 【リターン値】 除数結果
- 【引数】 x 被除数
Y 除数
- 【例】

```
#include "fixmath.h"
FIX16 x, y, ret;
ret = FIX16_div( x, y );
```

2.2.3 変換関数（マクロ）

(1)float型→固定小数点型変換

- 【書式】 `FIX<n> FIX<n>_fromfloat(float x)`
n:1~31
- 【説明】 float型を固定小数点型へ変換します。
- 【ヘッダ】 "fixmath.h"
- 【リターン値】 変換結果
- 【引数】 x 変換元データ

【例】

```
#include "fixmath.h"

float x;

FIX16 ret;

ret = FIX16_fromfloat( x );
```

(2)double 型→固定小数点型変換

【書式】

```
FIX<n> FIX<n>_fromdouble(double x)

n: 1~31
```

【説明】 double 型を固定小数点型へ変換します。

【ヘッダ】 "fixmath.h"

【リターン値】 変換結果

【引数】 x 変換元データ

【例】

```
#include "fixmath.h"

double x;

FIX16 ret;

ret = FIX16_fromdouble( x );
```

(3)固定小数点型→float 型変換

【書式】

```
float FIX<n>_tofloat(FIX<n> x)

n: 1~31
```

【説明】 固定小数点型を float 型へ変換します。

【ヘッダ】 "fixmath.h"

【リターン値】 変換結果

【引数】 x 変換元データ

【例】

```
#include "fixmath.h"

FIX16 x;

float ret;

ret = FIX16_tofloat( x );
```

(4)固定小数点型→double 型変換

【書式】

```
double FIX<n>_todouble(FIX<n> x)

n: 1~31
```

【説明】 固定小数点型を double 型へ変換します。

【ヘッダ】 "fixmath.h"

【リターン値】 変換結果

【引数】 x 変換元データ

【例】

```
#include "fixmath.h"

FIX16 x;

double ret;

ret = FIX16_todouble( x );
```

2.2.4 乗算関数

【書式】

```
FIX FIX_mul(FIX x, FIX y, int n)
           n:1~31
```

【説明】 固定小数点値の乗算を計算します。x と y の小数部がnビットの時に、x と y を long 型とみなした場合、x と y の積を右に n ビットシフトした値を求めます。

【ヘッダ】 "fixmath.h"

【リターン値】 乗算結果

【引数】

x	固定小数点値
y	固定小数点値
n	固定小数点値

【例】

```
#include "fixmath.h"

FIX16 x, y, ret;

ret = (FIX16)FIX_mul( (FIX)x , (FIX)y , 16 );
```

2.2.5 正弦関数

【書式】

```
FIX<n> FIX<n>_sin(FIX<n> x)
           n:16, 24, 29
```

【説明】 固定小数点値のラジアン値の正弦を計算します。

【ヘッダ】 "fixmath.h"

【リターン値】 x の正弦値

【引数】

x	正弦を求めるラジアン値
---	-------------

【例】

```
#include "fixmath.h"

FIX16 x, ret;

ret = FIX16_sin( x );
```

2.2.6 余弦関数

【書式】

```
FIX<n> FIX<n>_cos(FIX<n> x)
           n:16, 24, 29
```

【説明】 固定小数点値のラジアン値の余弦を計算します。

【ヘッダ】 "fixmath.h"

【リターン値】 x の余弦値

【引数】

x	余弦を求めるラジアン値
---	-------------

【例】

```
#include "fixmath.h"

FIX16 x, ret;

ret = FIX16_cos( x );
```

2.2.7 逆正接関数

【書式】	$\text{FIX}\langle n \rangle \text{ FIX}\langle n \rangle_atan(\text{FIX}\langle n \rangle x)$ n:16, 24, 29
【説明】	固定小数点値のラジアン値の逆正接を計算します。
【ヘッダ】	"fixmath.h"
【リターン値】	x の逆正接値
【引数】	x 逆正接を求めるラジアン値
【例】	<pre>#include "fixmath.h" FIX16 x, ret; ret = FIX16_atan(x);</pre>

2.2.8 平方根関数

【書式】	$\text{FIX}\langle n \rangle \text{ FIX}\langle n \rangle_sqrt(\text{FIX}\langle n \rangle x)$ n:16, 24, 29
【説明】	固定小数点値の正の平方根を計算します。
【ヘッダ】	"fixmath.h"
【リターン値】	x の正の平方根の値
【引数】	x 正の平方根を求める固定小数点値
【例】	<pre>#include "fixmath.h" FIX16 x, ret; ret = FIX16_sqrt(x);</pre>
【備考】	x が負の値の場合、結果は保証されません。

2.2.9 乗算 (小数部分) (マクロ)

【書式】	$\text{FIX}\langle n \rangle \text{ FIX}\langle n \rangle_mul_frac(\text{FIX}\langle n \rangle x, \text{FIX}\langle n \rangle y)$ n: 1~31
【説明】	固定小数点値の乗算結果の小数部分を計算します。結果の値は常に正 ($0 \leq \text{結果} < 1.0$) になります。
【ヘッダ】	"fixmath.h"
【リターン値】	x と y の積の小数部分。
【引数】	x 固定小数点値 y 固定小数点値
【例】	<pre>#include "fixmath.h" FIX16 x, y, ret; ret = FIX16_mul_frac(x, y);</pre>

2.2.10 乗算 (飽和演算) (マクロ)

【書式】	<code>FIX<n> FIX<n>_mul_sat(FIX<n> x, FIX<n> y)</code> n: 1~31
【説明】	固定小数点値の乗算を計算します。結果がオーバーフローした場合、結果の符合にしたがって最大値または最小値を返します。
【ヘッダ】	"fixmath.h"
【リターン値】	x と y の積 (飽和演算)。
【引数】	x 固定小数点値 y 固定小数点値
【例】	<pre>#include "fixmath.h" FIX16 x, y, ret; ret = FIX16_mul_sat(x, y);</pre>

2.2.11 乗算 (FIX 型) (マクロ)

【書式】	<code>FIX FIX_mul_scale<n>(FIX x, FIX y)</code> n: 1~31
【説明】	総称固定小数点値の乗算を計算します。x と y を long 型とみなした場合、x と y の積を右に n ビットシフトした値を求めます。
【ヘッダ】	"fixmath.h"
【リターン値】	x と y の積 (FIX 型)。
【引数】	x 固定小数点値 y 固定小数点値
【例】	<pre>#include "fixmath.h" FIX x, y, ret; ret = FIX_mul_scale16(x, y);</pre>
【備考】	結果の値が 32 ビットで表現できないとき、その値は保証されません。 FIX<n ₁ >, FIX<n ₂ > 型を乗算して FIX<n ₃ > 型を求める場合は、シフトするビット数は n ₁ +n ₂ -n ₃ になりますので、n としてこの値 (1~31 の範囲) を指定してください。

2.2.14 乗算 (小数部分、FIX 型) (マクロ)

- 【書式】 `FIX FIX_mul_frac_scale(FIX x, FIX y, int n)`
n: 1~31
- 【説明】 総称固定小数点値の乗算結果の小数部分を計算します。x と y を long 型とみなした
場合、x と y の積を右に n ビットシフトした値の下位 n ビットを求めます。
- 【ヘッダ】 "fixmath.h"
- 【リターン値】 x と y の積 (FIX 型)。
- 【引数】 x 固定小数点値
y 固定小数点値
- 【例】

```
#include "fixmath.h"

FIX x, y, ret;

int n=16;

ret = FIX_mul_frac_scale ( x, y, n );
```
- 【備考】 `FIX<n+d>`, `FIX<n-d>` 型を乗算して `FIX<n>` 型を求める場合に使用することができます。

2.2.15 乗算 (飽和演算、FIX 型) (マクロ)

- 【書式】 `FIX FIX_mul_sat_scale<n>(FIX x, FIX y)`
n: 1~31
- 【説明】 総称固定小数点値の乗算を計算します。x と y を long 型とみなした場合、x と y の
積を右に n ビットシフトした値を求めます。結果がオーバーフローした場合、結果の符合
にしたがって最大値または最小値を返します。
- 【ヘッダ】 "fixmath.h"
- 【リターン値】 x と y の積 (FIX 型)。
- 【引数】 x 固定小数点値
y 固定小数点値
- 【例】

```
#include "fixmath.h"

FIX x, y, ret;

ret = FIX_mul_sat_scale16( x, y );
```
- 【備考】 結果の値が 32 ビットで表現できないとき、その値は保証されません。
`FIX<n1>`, `FIX<n2>` 型を乗算して `FIX<n3>` 型を求める場合は、シフトするビット数は
`n1+n2-n3` になりますので、n としてこの値 (1~31 の範囲) を指定してください。

2.2.16 乗算関数 (FIX 型)

【書式】 `FIX FIX_mul_sat_scale(FIX x, FIX y, int n)`

n: 1~31

【説明】 総称固定小数点値の乗算を計算します。x と y を long 型とみなした場合、x と y の積を右に n ビットシフトした値を求めます。結果がオーバーフローした場合、結果の符号にしたがって最大値または最小値を返します。

【ヘッダ】 "fixmath.h"

【リターン値】 x と y の積 (FIX 型)。

【引数】 x 固定小数点値

y 固定小数点値

【例】

```
#include "fixmath.h"
```

```
FIX x, y, ret;
```

```
int n=16;
```

```
ret = FIX_mul_sat_scale ( x, y, n );
```

【備考】 結果の値が 32 ビットで表現できないとき、その値は保証されません。

FIX<n₁>, FIX<n₂> 型を乗算して FIX<n₃> 型を求める場合は、シフトするビット数は n₁+n₂-n₃ になりますので、n としてこの値 (1~31 の範囲) を指定してください。

3. 性能

3.1 測定条件

コンパイラ : Renesas RX Standard コンパイラ V.1.0.0.0

ビルド条件 : 表 3-1に示す条件でライブラリを構築

表 3-1 ライブラリ 構築時条件

条件	ライブラリ 構築時のオプション	
	cpu	endian
1	RX600	big
2	RX600	little

3.2 実行サイクル数

固定小数点ライブラリの演算速度を表 3-2に示します。

表 3-2 固定小数点ライブラリの演算速度

CPU		RX600	RX600
ライブラリ 構築時条件(表 3-1 参照)		1	2
正弦関数	FIX16_sin	56	56
	FIX24_sin	56	56
	FIX29_sin	55	55
余弦関数	FIX16_cos	52	52
	FIX24_cos	52	52
	FIX29_cos	53	53
逆正接関数	FIX16_atan	118	118
	FIX24_atan	118	118
	FIX29_atan	112	112
平方根関数	FIX16_sqrt	82	82
	FIX24_sqrt	82	82
	FIX29_sqrt	81	81

【注】単位は Cycle. 演算速度の測定値には誤差が含まれています。

本ライブラリの演算結果の最大誤差は最下位ビットで±2です。

サポート窓口

ルネサステクノロジホームページ

<http://japan.renesas.com/>

お問い合わせ先

<http://japan.renesas.com/inquiry>

csc@renesas.com

改訂記録

Rev.	発行日	改訂内容	
		ページ	ポイント
1.00	2010.12.28	-	初版発行
1.01	2012.10.30	-	FIX<n>_atan(-1)で pai/4 を返すバグの修正 ドキュメント番号変更 R21AN0003 -> R20AN0228

製品ご使用上の注意事項

ここでは、マイコン製品全体に適用する「使用上の注意事項」について説明します。個別の使用上の注意事項については、本ドキュメントおよびテクニカルアップデートを参照してください。

1. 未使用端子の処理

【注意】未使用端子は、本文の「未使用端子の処理」に従って処理してください。

CMOS 製品の入力端子のインピーダンスは、一般に、ハイインピーダンスとなっています。未使用端子を開放状態で動作させると、誘導現象により、LSI 周辺のノイズが印加され、LSI 内部で貫通電流が流れたり、入力信号と認識されて誤動作を起こす恐れがあります。未使用端子は、本文「未使用端子の処理」で説明する指示に従い処理してください。

2. 電源投入時の処置

【注意】電源投入時は、製品の状態は不定です。

電源投入時には、LSI の内部回路の状態は不確定であり、レジスタの設定や各端子の状態は不定です。

外部リセット端子でリセットする製品の場合、電源投入からリセットが有効になるまでの期間、端子の状態は保証できません。

同様に、内蔵パワーオンリセット機能を使用してリセットする製品の場合、電源投入からリセットのかかる一定電圧に達するまでの期間、端子の状態は保証できません。

3. リザーブアドレス（予約領域）のアクセス禁止

【注意】リザーブアドレス（予約領域）のアクセスを禁止します。

アドレス領域には、将来の機能拡張用に割り付けられているリザーブアドレス（予約領域）がありません。これらのアドレスをアクセスしたときの動作については、保証できませんので、アクセスしないようにしてください。

4. クロックについて

【注意】リセット時は、クロックが安定した後、リセットを解除してください。

プログラム実行中のクロック切り替え時は、切り替え先クロックが安定した後に切り替えてください。

リセット時、外部発振子（または外部発振回路）を用いたクロックで動作を開始するシステムでは、クロックが十分安定した後、リセットを解除してください。また、プログラムの途中で外部発振子（または外部発振回路）を用いたクロックに切り替える場合は、切り替え先のクロックが十分安定してから切り替えてください。

5. 製品間の相違について

【注意】型名の異なる製品に変更する場合は、製品型名ごとにシステム評価試験を実施してください。

同じグループのマイコンでも型名が違っていると、内部 ROM、レイアウトパターンの相違などにより、電気的特性の範囲で、特性値、動作マージン、ノイズ耐量、ノイズ輻射量などが異なる場合があります。型名が異なる製品に変更する場合は、個々の製品ごとにシステム評価試験を実施してください。

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
 2. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
 3. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
 4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
 5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.
"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.
Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
 6. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
 7. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or systems manufactured by you.
 8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
 9. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
 10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
 11. This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
 12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.
- (Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.
(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.



SALES OFFICES

Renesas Electronics Corporation

<http://www.renesas.com>

Refer to "<http://www.renesas.com/>" for the latest and detailed information.

Renesas Electronics America Inc.
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K.
Tel: +44-1628-651-700, Fax: +44-1628-651-804

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-3390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
11F., Samik Laved or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141