Module std::uq32_32

Defines an unsigned, fixed-point numeric type with a 32-bit integer part and a 32-bit fractional part. The notation uq32_32 and UQ32_32 is based on Q notation . q indicates it a fixed-point number. The u prefix indicates it is unsigned. The 32_32 suffix indicates the number of bits, where the first number indicates the number of bits in the integer part, and the second the number of bits in the fractional part--in this case 32 bits for each.

A fixed-point numeric type with 32 integer bits and 32 fractional bits, represented by an underlying 64 bit value. This is a binary representation, so decimal values may not be exactly representable, but it provides more than 9 decimal digits of precision both before and after the decimal point (18 digits total).

The total number of bits in the fixed-point number. Used in macro invocations.

The number of fractional bits in the fixed-point number. Used in macro invocations.

Create a fixed-point value from a quotient specified by its numerator and denominator. from_quotient and from_int should be preferred over using from_raw . Unless the denominator is a power of two, fractions can not be represented accurately, so be careful about rounding errors. Aborts if the denominator is zero. Aborts if the input is non-zero but so small that it will be represented as zero, e.g. smaller than 2^{-32}. Aborts if the input is too large, e.g. larger than or equal to 2^32.

Create a fixed-point value from an integer. from_int and from_quotient should be preferred over using from_raw .

Add two fixed-point numbers, a + b. Aborts if the sum overflows.

Subtract two fixed-point numbers, a - b. Aborts if a < b.

Multiply two fixed-point numbers, truncating any fractional part of the product. Aborts if the product overflows.

Divide two fixed-point numbers, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

Convert a fixed-point number to an integer, truncating any fractional part.

Multiply a u64 integer by a fixed-point number, truncating any fractional part of the product. Aborts if the product overflows.

Divide a u64 integer by a fixed-point number, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

Less than or equal to. Returns true if and only if a <= a.

Less than. Returns true if and only if a < b.

Greater than or equal to. Returns true if and only if a >= b.

Greater than. Returns true if and only if a > b.

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

Struct

A fixed-point numeric type with 32 integer bits and 32 fractional bits, represented by an underlying 64 bit value. This is a binary representation, so decimal values may not be exactly representable, but it provides more than 9 decimal digits of precision both before and after the decimal point (18 digits total).

```bash
```

```bash
```

```bash
```

[error]

```

```bash
# [error]
```

```bash
# [error]
```

```bash
# [error]
```

```bash
# [error]
```

The total number of bits in the fixed-point number. Used in macro invocations.

```bash
```

The number of fractional bits in the fixed-point number. Used in macro invocations.

```bash
```

Create a fixed-point value from a quotient specified by its numerator and denominator. from_quotient and from_int should be preferred over using from_raw . Unless the denominator is a power of two, fractions can not be represented accurately, so be careful about rounding errors. Aborts if the denominator is zero. Aborts if the input is non-zero but so small that it will be represented as zero, e.g. smaller than 2^{-32}. Aborts if the input is too large, e.g. larger than or equal to 2^32.

```bash
```

```bash
```

Create a fixed-point value from an integer. from_int and from_quotient should be preferred over using from_raw .

```bash
```

```bash
```
```

Add two fixed-point numbers, a + b. Aborts if the sum overflows.

```bash
```

```bash
```

Subtract two fixed-point numbers, a - b. Aborts if a < b.

```bash
```

```bash
```

Multiply two fixed-point numbers, truncating any fractional part of the product. Aborts if the product overflows.

```bash
```

```bash
```

Divide two fixed-point numbers, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```bash
```

Convert a fixed-point number to an integer, truncating any fractional part.

```bash
```

```bash
```

Multiply a u64 integer by a fixed-point number, truncating any fractional part of the product. Aborts if the product overflows.

```bash
```

```bash
```

Divide a u64 integer by a fixed-point number, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```bash
```

Less than or equal to. Returns true if and only if a <= a.

```bash
```

```bash
```

Less than. Returns true if and only if a < b.

```bash
```

```bash
```

Greater than or equal to. Returns true if and only if a >= b.

```bash
```

```bash
```

Greater than. Returns true if and only if a > b.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Constants

```bash
```

[error]

```

```bash

# [error]

```

```bash

# [error]

```

```bash

# [error]

```

```bash

# [error]

```

The total number of bits in the fixed-point number. Used in macro invocations.

```bash

```

The number of fractional bits in the fixed-point number. Used in macro invocations.

```bash

```

Create a fixed-point value from a quotient specified by its numerator and denominator. from_quotient and from_int should be preferred over using from_raw . Unless the denominator is a power of two, fractions can not be represented accurately, so be careful about rounding errors. Aborts if the denominator is zero. Aborts if the input is non-zero but so small that it will be represented as zero, e.g. smaller than 2^{-32}. Aborts if the input is too large, e.g. larger than or equal to 2^32.

```bash

```

```bash

```

Create a fixed-point value from an integer. from_int and from_quotient should be preferred over using from_raw .

```bash

```

```bash

```

Add two fixed-point numbers, a + b. Aborts if the sum overflows.

```bash

```

```bash

```

Subtract two fixed-point numbers, a - b. Aborts if a < b.

```bash

```

```bash

```

Multiply two fixed-point numbers, truncating any fractional part of the product. Aborts if the product overflows.

```bash

```

```bash

```

Divide two fixed-point numbers, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash

```

```bash

```

Convert a fixed-point number to an integer, truncating any fractional part.

```bash

```

```bash

```

Multiply a u64 integer by a fixed-point number, truncating any fractional part of the product. Aborts if the product overflows.

```bash

```

```bash

```

Divide a u64 integer by a fixed-point number, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash

```

```bash
```

Less than or equal to. Returns true if and only if a <= a.

```bash
```

```bash
```

Less than. Returns true if and only if a < b.

```bash
```

```bash
```

Greater than or equal to. Returns true if and only if a >= b.

```bash
```

```bash
```

Greater than. Returns true if and only if a > b.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Function
```

Create a fixed-point value from a quotient specified by its numerator and denominator. from_quotient and from_int should be preferred over using from_raw . Unless the denominator is a power of two, fractions can not be represented accurately, so be careful about rounding errors. Aborts if the denominator is zero. Aborts if the input is non-zero but so small that it will be represented as zero, e.g. smaller than 2^{-32}. Aborts if the input is too large, e.g. larger than or equal to 2^32.

```bash
```

```bash
```

Create a fixed-point value from an integer. from_int and from_quotient should be preferred over using from_raw .

```bash
```

```bash
```

Add two fixed-point numbers, a + b. Aborts if the sum overflows.

```bash
```

```bash
```

Subtract two fixed-point numbers, a - b. Aborts if a < b.

```bash
```

```bash
```

Multiply two fixed-point numbers, truncating any fractional part of the product. Aborts if the product overflows.

```bash
```

```bash
```

Divide two fixed-point numbers, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```bash
```

Convert a fixed-point number to an integer, truncating any fractional part.

```bash
```

```

```bash
```

Multiply a u64 integer by a fixed-point number, truncating any fractional part of the product. Aborts if the product overflows.

```bash
```

```bash
```

Divide a u64 integer by a fixed-point number, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```bash
```

Less than or equal to. Returns true if and only if a <= a.

```bash
```

```bash
```

Less than. Returns true if and only if a < b.

```bash
```

```bash
```

Greater than or equal to. Returns true if and only if a >= b.

```bash
```

```bash
```

Greater than. Returns true if and only if a > b.

```bash
```

```bash
```
```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Function

Create a fixed-point value from an integer. from_int and from_quotient should be preferred over using from_raw .

```bash
```

```bash
```

Add two fixed-point numbers, a + b. Aborts if the sum overflows.

```bash
```

```bash
```

Subtract two fixed-point numbers, a - b. Aborts if a < b.

```bash
```

```bash
```

Multiply two fixed-point numbers, truncating any fractional part of the product. Aborts if the product overflows.

```bash
```

```bash
```

Divide two fixed-point numbers, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```

```bash
```

Convert a fixed-point number to an integer, truncating any fractional part.

```bash
```

```bash
```

Multiply a u64 integer by a fixed-point number, truncating any fractional part of the product. Aborts if the product overflows.

```bash
```

```bash
```

Divide a u64 integer by a fixed-point number, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```bash
```

Less than or equal to. Returns true if and only if a <= a.

```bash
```

```bash
```

Less than. Returns true if and only if a < b.

```bash
```

```bash
```

Greater than or equal to. Returns true if and only if a >= b.

```bash
```

```bash
```
```

Greater than. Returns true if and only if a > b.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Function

Add two fixed-point numbers, a + b. Aborts if the sum overflows.

```bash
```

```bash
```

Subtract two fixed-point numbers, a - b. Aborts if a < b.

```bash
```

```bash
```

Multiply two fixed-point numbers, truncating any fractional part of the product. Aborts if the product overflows.

```bash
```

```bash
```

Divide two fixed-point numbers, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```

```bash
```

Convert a fixed-point number to an integer, truncating any fractional part.

```bash
```

```bash
```

Multiply a u64 integer by a fixed-point number, truncating any fractional part of the product. Aborts if the product overflows.

```bash
```

```bash
```

Divide a u64 integer by a fixed-point number, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```bash
```

Less than or equal to. Returns true if and only if a <= a.

```bash
```

```bash
```

Less than. Returns true if and only if a < b.

```bash
```

```bash
```

Greater than or equal to. Returns true if and only if a >= b.

```bash
```

```bash
```
```

Greater than. Returns true if and only if a > b.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Function

Subtract two fixed-point numbers, a - b. Aborts if a < b.

```bash
```

```bash
```

Multiply two fixed-point numbers, truncating any fractional part of the product. Aborts if the product overflows.

```bash
```

```bash
```

Divide two fixed-point numbers, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```bash
```

Convert a fixed-point number to an integer, truncating any fractional part.

```bash
```

```

```bash
```

Multiply a u64 integer by a fixed-point number, truncating any fractional part of the product. Aborts if the product overflows.

```bash
```

```bash
```

Divide a u64 integer by a fixed-point number, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```bash
```

Less than or equal to. Returns true if and only if a <= a.

```bash
```

```bash
```

Less than. Returns true if and only if a < b.

```bash
```

```bash
```

Greater than or equal to. Returns true if and only if a >= b.

```bash
```

```bash
```

Greater than. Returns true if and only if a > b.

```bash
```

```bash
```
```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Function

Multiply two fixed-point numbers, truncating any fractional part of the product. Aborts if the product overflows.

```bash
```

```bash
```

Divide two fixed-point numbers, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```bash
```

Convert a fixed-point number to an integer, truncating any fractional part.

```bash
```

```bash
```

Multiply a u64 integer by a fixed-point number, truncating any fractional part of the product. Aborts if the product overflows.

```bash
```

```bash
```

Divide a u64 integer by a fixed-point number, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```bash
```

Less than or equal to. Returns true if and only if a <= a.

```bash
```

```bash
```

Less than. Returns true if and only if a < b.

```bash
```

```bash
```

Greater than or equal to. Returns true if and only if a >= b.

```bash
```

```bash
```

Greater than. Returns true if and only if a > b.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Function

Divide two fixed-point numbers, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash

```

```bash

```

Convert a fixed-point number to an integer, truncating any fractional part.

```bash

```

```bash

```

Multiply a u64 integer by a fixed-point number, truncating any fractional part of the product. Aborts if the product overflows.

```bash

```

```bash

```

Divide a u64 integer by a fixed-point number, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash

```

```bash

```

Less than or equal to. Returns true if and only if a <= a.

```bash

```

```bash

```

Less than. Returns true if and only if a < b.

```bash

```

```bash

```

Greater than or equal to. Returns true if and only if a >= b.

```bash
```

```

```bash
```

Greater than. Returns true if and only if a > b.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Function

Convert a fixed-point number to an integer, truncating any fractional part.

```bash
```

```bash
```

Multiply a u64 integer by a fixed-point number, truncating any fractional part of the product. Aborts if the product overflows.

```bash
```

```bash
```

Divide a u64 integer by a fixed-point number, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```bash

```

Less than or equal to. Returns true if and only if a <= a.

```bash

```

```bash

```

Less than. Returns true if and only if a < b.

```bash

```

```bash

```

Greater than or equal to. Returns true if and only if a >= b.

```bash

```

```bash

```

Greater than. Returns true if and only if a > b.

```bash

```

```bash

```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash

```

```bash

```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash

```

```bash

```

Function

Multiply a u64 integer by a fixed-point number, truncating any fractional part of the product. Aborts if the product overflows.

```bash

```

```bash
```

Divide a u64 integer by a fixed-point number, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```bash
```

Less than or equal to. Returns true if and only if a <= a.

```bash
```

```bash
```

Less than. Returns true if and only if a < b.

```bash
```

```bash
```

Greater than or equal to. Returns true if and only if a >= b.

```bash
```

```bash
```

Greater than. Returns true if and only if a > b.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```
```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Function

Divide a u64 integer by a fixed-point number, truncating any fractional part of the quotient. Aborts if the divisor is zero. Aborts if the quotient overflows.

```bash
```

```bash
```

Less than or equal to. Returns true if and only if a <= a.

```bash
```

```bash
```

Less than. Returns true if and only if a < b.

```bash
```

```bash
```

Greater than or equal to. Returns true if and only if a >= b.

```bash
```

```bash
```

Greater than. Returns true if and only if a > b.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash
```

```

```bash
```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Function

Less than or equal to. Returns true if and only if a <= a.

```bash
```

```bash
```

Less than. Returns true if and only if a < b.

```bash
```

```bash
```

Greater than or equal to. Returns true if and only if a >= b.

```bash
```

```bash
```

Greater than. Returns true if and only if a > b.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```
```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Function

Less than. Returns true if and only if a < b.

```bash
```

```bash
```

Greater than or equal to. Returns true if and only if a >= b.

```bash
```

```bash
```

Greater than. Returns true if and only if a > b.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Function

Greater than or equal to. Returns true if and only if a >= b.

```bash
```

```bash
```

Greater than. Returns true if and only if a > b.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Function

Greater than. Returns true if and only if a > b.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash
```

```bash
```

```

## Function

Accessor for the raw u64 value. Can be paired with from_raw to perform less common operations on the raw values directly.

```bash

```

```bash

```

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash

```

```bash

```

Function

Accessor for the raw u64 value. Can be paired with to_raw to perform less common operations on the raw values directly.

```bash

```

```bash

```
```