NAME

stdckdint - checked integer arithmetic

SYNOPSIS

#include <stdckdint.h>

bool
ckd_add(*type1 *result*, *type2 a*, *type3 b*);

bool
ckd_sub(*type1 *result*, *type2 a*, *type3 b*);

bool
ckd_mul(*type1 *result*, *type2 a*, *type3 b*);

DESCRIPTION

The function-like macros **ckd_add**, **ckd_sub**, and **ckd_mul** perform checked integer addition, subtraction, and multiplication, respectively. If the result of adding, subtracting, or multiplying *a* and *b* as if their respective types had infinite range fits in *type1*, it is stored in the location pointed to by *result* and the macro evaluates to false. Otherwise, the macro evaluates to true and the contents of the location pointed to by *result* is the result of the operation wrapped to the range of *type1*.

RETURN VALUES

The **ckd_add**, **ckd_sub**, and **ckd_mul** macros evaluate to true if the requested operation overflowed the result type and false otherwise.

EXAMPLES

```
#include <assert.h>
#include <limits.h>
#include <stdckdint.h>

int main(void)
{
        int result;

        assert(!ckd_add(&result, INT_MAX, 0));
        assert(result == INT_MAX);
        assert(ckd_add(&result, INT_MAX, 1));
        assert(result == INT_MIN);
```

```
        assert(!ckd_sub(&result, INT_MIN, 0));
        assert(result == INT_MIN);
        assert(ckd_sub(&result, INT_MIN, 1));
        assert(result == INT_MAX);

        assert(!ckd_mul(&result, INT_MAX / 2, 2));
        assert(result == INT_MAX - 1);
        assert(ckd_mul(&result, INT_MAX / 2 + 1, 2));
        assert(result == INT_MIN);

        return 0;
}
```

HISTORY

The **ckd_add**, **ckd_sub**, and **ckd_mul** macros were first introduced in FreeBSD 14.0.

AUTHORS

The **ckd_add**, **ckd_sub**, and **ckd_mul** macros and this manual page were written by Dag-Erling Smørgrav <*des@FreeBSD.org*>.