

uInt32

IMPLEMENTACE OPERACÍ S 32B ČÍSLY BEZ ZNAMÉNKA

Příručka uživatele a programátora



SofCon[®] spol. s r.o.
Střešovická 49
162 00 Praha 6
tel/fax: +420 220 180 454
E-mail: sofcon@sofcon.cz
www: <http://www.sofcon.cz>

Informace v tomto dokumentu byly pečlivě zkontrolovány a SofCon věří, že jsou spolehlivé, přesto SofCon nenese odpovědnost za případné nepřesnosti nebo nesprávnosti zde uvedených informací.

SofCon negarantuje bezchybnost tohoto dokumentu ani programového vybavení, které je v tomto dokumentu popsáno. Uživatel přebírá informace z tohoto dokumentu a odpovídající programové vybavení ve stavu, jak byly vytvořeny a sám je povinen provést validaci bezchybnosti produktu, který s použitím zde popsaného programového vybavení vytvořil.

SofCon si vyhrazuje právo změny obsahu tohoto dokumentu bez předchozího oznámení a nenese žádnou odpovědnost za důsledky, které z toho mohou vyplynout pro uživatele.

Datum vydání: 20.05.2003

Datum posledního uložení dokumentu: 20.05.2003

(Datum vydání a posledního uložení dokumentu musí být stejné)

Upozornění:

V dokumentu použité názvy výrobků, firem apod. mohou být ochrannými známkami nebo registrovanými ochrannými známkami příslušných vlastníků.

Obsah :

1.O dokumentu	5
1.1. Revize dokumentu	5
1.2. Účel dokumentu	5
1.3. Rozsah platnosti	5
1.4. Související dokumenty	5
2.Termíny a definice	5
3.Úvod	6
4.Konstanty	6
5.Typy	6
6.Operace	6
6.1. Funkce	6
6.1.1. IncUInt32	6
6.1.2. IncUInt32Ex	6
6.1.3. DecUInt32	6
6.1.4. DecUInt32Ex	7
6.1.5. AddUInt32	7
6.1.6. SubUInt32	7
6.1.7. IsLessUInt32	7
6.1.8. IsLessEqualUInt32	7
6.1.9. IsGreaterUInt32	7
6.1.10. IsEqualUInt32	7
6.1.11. MulUInt32	8
6.1.12. DivUInt32	8
6.1.13. ModUInt32	8
6.1.14. ShlUInt32	8
6.1.15. ShrUInt32	8
6.2. Inline funkce	8
6.2.1. _IncUInt32	8
6.2.2. _IncUInt32Ex	9
6.2.3. _DecUInt32	9
6.2.4. _DecUInt32Ex	9
6.2.5. _IsLessUInt32	9
6.2.6. _IsLessEqualUInt32	9
6.2.7. _IsGreaterUInt32	9
6.2.8. _IsEqualUInt32	9

1. O dokumentu

1.1. Revize dokumentu

Verze dokumentu	Verze SW	Autor	Datum vydání	Popis změn
1.00	1.XX	Hv	19.3.2003	První vydání.
1.10	1.XX	Tu	20.05.2003	Úprava dokumentu dle ISO9000.

1.2. Účel dokumentu

Tento dokument slouží jako popis knihovny UInt32, ve které jsou implementovány základní operace s 32b čísly bez znaménka.

1.3. Rozsah platnosti

Určen pro programátory a uživatele programového vybavení SofCon.

1.4. Související dokumenty

Pro čtení tohoto dokumentu není potřeba číst žádný další manuál, ale je potřeba orientovat se v používání programového vybavení SofCon.

Popis formátu verze knihovny a souvisejících funkcí je popsán v manuálu LibVer.

2. Termíny a definice

Používané termíny a definice jsou popsány v samostatném dokumentu Termíny a definice.

3. Úvod

Tato knihovna implementuje základní operace pro práci s 32b čísly bez znaménka. Tato čísla budou dále označována UNSIGNED32.

4. Konstanty

```
cVerNo = např. $0100;
```

```
cVer = např. '01.00,17.03.2003';
```

konstanty udávají verzi a poslední změnu jednotky ve standardním formátu definovaném v knihovně LibVer.

5. Typy

```
UNSIGNED32 = Longint;
```

Deklarace typu používaného operacemi s 32b čísly bez znaménka.

6. Operace

Operace se dělí na dva typy lišící se způsobem překladu. V prvním případě, podkapitola 6.1, se funkce překládají voláním vybrané funkce tj. CALL. Ve druhém případě, podkapitola 6.2, se kód funkce vkládá přímo na kódu programu na místo použité vybrané funkce.

Jako mnemotechnická pomůcka pro rozlišení těchto dvou operací se používá znak „_“, kterým jsou uvedeny všechny inline funkce.

6.1. Funkce

6.1.1. IncUInt32

```
function IncUInt32( X : UNSIGNED32 ) : UNSIGNED32;
```

Funkce provede inkrementaci čísla X o 1 (tj. X+1) bez kontroly přetečení čísla a vrátí toto číslo jako svoji funkční hodnotu.

6.1.2. IncUInt32Ex

```
function IncUInt32Ex( X, N : UNSIGNED32 ) : UNSIGNED32;
```

Funkce provede inkrementaci čísla X o N (tj. X+N) bez kontroly přetečení čísla a vrátí toto číslo jako svoji funkční hodnotu.

6.1.3. DecUInt32

```
function DecUInt32( X : UNSIGNED32 ) : UNSIGNED32;
```

Funkce provede dekrementaci čísla X o 1 (tj. X-1) bez kontroly podtečení čísla a toto číslo vrátí jako svoji funkční hodnotu.

6.1.4. DecUInt32Ex

```
function DecUInt32Ex( X, N : UNSIGNED32 ) : UNSIGNED32;
```

Funkce provede dekrementaci čísla X o N (tj. X-N) bez kontroly podtečení čísla a toto číslo vrátí jako svoji funkční hodnotu.

6.1.5. AddUInt32

```
function AddUInt32( A, B: UNSIGNED32 ) : UNSIGNED32;
```

Funkce provede součet čísel A a B (tj. A+B) s kontrolou přetečení čísla a tento součet vrátí jako svoji funkční hodnotu. V případě přetečení čísla se v místě použití této funkce vygeneruje RunError(215), tzn. arithmetic overflow error.

6.1.6. SubUInt32

```
function SubUInt32( A, B: UNSIGNED32 ) : UNSIGNED32;
```

Funkce provede rozdíl čísel A a B (tj. A-B) s kontrolou podtečení čísla a tento rozdíl vrátí jako svoji funkční hodnotu. V případě podtečení čísla se v místě použití této funkce vygeneruje RunError(215), tzn. arithmetic overflow error.

6.1.7. IsLessUInt32

```
function IsLessUInt32( A, B: UNSIGNED32 ): Boolean;
```

Funkce vrátí TRUE, pokud je číslo A menší jak B (tj. A<B). V opačném případě se vrací FALSE (tj. A>=B).

6.1.8. IsLessEqualUInt32

```
function IsLessEqualUInt32( A, B: UNSIGNED32 ): Boolean;
```

Funkce vrátí TRUE, pokud je číslo A je menší nebo rovno jak B (tj. A<=B). V opačném případě se vrací FALSE (tj. A>B).

6.1.9. IsGreaterUInt32

```
function IsGreaterUInt32( A, B: UNSIGNED32 ): Boolean;
```

Funkce vrátí TRUE, pokud je číslo A je větší jak B (tj. A>B). V opačném případě se vrací FALSE (tj. A<=B).

6.1.10. IsEqualUInt32

```
function IsEqualUInt32( A, B: UNSIGNED32 ): Boolean;
```

Funkce vrátí TRUE, pokud je číslo A rovno B (tj. A=B). V opačném případě se vrací FALSE (tj. A<>B).

6.1.11. MulUInt32

```
function MulUInt32( A, B: UNSIGNED32 ) : UNSIGNED32;
```

Funkce provede součin čísel A a B (tj. $A*B$) s kontrolou přetečení čísla a tento součin vrátí jako svoji funkční hodnotu. V případě přetečení čísla se v místě použití této funkce vygeneruje RunError(215), tzn. arithmetic overflow error.

6.1.12. DivUInt32

```
function DivUInt32( A, B: UNSIGNED32 ) : UNSIGNED32;
```

Funkce provede celočíselné dělení čísla A číslem B (tj. A/B) a tento výsledek vrátí jako svoji funkční hodnotu. V případě dělení nulou se generuje RunError(200), tzn. division by zero.

6.1.13. ModUInt32

```
function ModUInt32( A, B: UNSIGNED32 ) : UNSIGNED32;
```

Funkce provede celočíselné dělení čísla A číslem B (tj. A/B) a zbytek po dělení vrátí jako svoji funkční hodnotu. V případě dělení nulou se generuje RunError(200), tzn. division by zero.

6.1.14. ShlUInt32

```
function ShlUInt32( A : UNSIGNED32; N : BYTE ) : UNSIGNED32;
```

Funkce provede posun čísla A doleva o N bitů (tj. $A \text{ shl } N$) a výsledek vrátí jako svoji funkční hodnotu. Číslo N by mělo nabývat hodnot od 0 do 31, pokud ale bude toto číslo větší použije se zbytek po dělení čísla 32 (tj. $A \text{ mod } 32$). Na místo nejnižšího bitu se vkládá 0.

6.1.15. ShrUInt32

```
function ShrUInt32( A : UNSIGNED32; N : BYTE ) : UNSIGNED32;
```

Funkce provede posun čísla A doprava o N bitů (tj. $A \text{ shr } N$) a výsledek vrátí jako svoji funkční hodnotu. Číslo N by mělo nabývat hodnot od 0 do 31, pokud ale bude toto číslo větší použije se zbytek po dělení čísla 32 (tj. $A \text{ mod } 32$). Na nejvyšší bit se vkládá 0.

6.2. Inline funkce

6.2.1. _IncUInt32

```
function _IncUInt32( X : UNSIGNED32 ) : UNSIGNED32;
```

Funkce provede inkrementaci čísla X o 1 (tj. $X+1$) bez kontroly přetečení čísla a vrátí toto číslo jako svoji funkční hodnotu.

6.2.2. `_IncUInt32Ex`

```
function _IncUInt32Ex( X, N : UNSIGNED32 ) : UNSIGNED32;
```

Funkce provede inkrementaci čísla X o N (tj. X+N) bez kontroly přetečení čísla a vrátí toto číslo jako svoji funkční hodnotu.

6.2.3. `_DecUInt32`

```
function _DecUInt32( X : UNSIGNED32 ) : UNSIGNED32;
```

Funkce provede dekrementaci čísla X o 1 (tj. X-1) bez kontroly podtečení čísla a toto číslo vrátí jako svoji funkční hodnotu.

6.2.4. `_DecUInt32Ex`

```
function _DecUInt32Ex( X, N : UNSIGNED32 ) : UNSIGNED32;
```

Funkce provede dekrementaci čísla X o N (tj. X-N) bez kontroly podtečení čísla a toto číslo vrátí jako svoji funkční hodnotu.

6.2.5. `_IsLessUInt32`

```
function _IsLessUInt32( A, B: UNSIGNED32 ): Boolean;
```

Funkce vrátí TRUE, pokud je číslo A menší jak B (tj. A<B). V opačném případě se vrací FALSE (tj. A>=B).

6.2.6. `_IsLessEqualUInt32`

```
function _IsLessEqualUInt32( A, B: UNSIGNED32 ): Boolean;
```

Funkce vrátí TRUE, pokud je číslo A je menší nebo rovno jak B (tj. A<=B). V opačném případě se vrací FALSE (tj. A>B).

6.2.7. `_IsGreaterUInt32`

```
function _IsGreaterUInt32( A, B: UNSIGNED32 ): Boolean;
```

Funkce vrátí TRUE, pokud je číslo A je větší jak B (tj. A>B). V opačném případě se vrací FALSE (tj. A<=B).

6.2.8. `_IsEqualUInt32`

```
function _IsEqualUInt32( A, B: UNSIGNED32 ): Boolean;
```

Funkce vrátí TRUE, pokud je číslo A rovno B (tj. A=B). V opačném případě se vrací FALSE (tj. A<>B).