

---

Subject: text<->double conversions refactored  
Posted by [mirek](#) on Sun, 29 Aug 2021 11:57:30 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

After month of development I have today committed refactored text<->double conversion routines (FormatDouble, ScanDouble...)

New routines should(\*) guarantee that

- ScanDouble(FormatDouble(x)) == x for all finite doubles
- FormatG(x, 17) returns the same text as printf("%.17g", x) for all doubles

New routines are in the same speed class as ryu, wich seems to be current performance standard (conversion in both directions is around 20ns on modern CPU). (well, I believe my algorithm would be actually faster than ryu if it was doing the same thing, but U++ provides a lot of options which slow it down a bit to basically "ryu equivalent speed").

(\*) Well, the theory behind is a bit complicated and I am still ironing details, but from practical point of view I have tested it so far with 300 billions of random double numbers and continue to do so...

<https://github.com/ultimatepp/ultimatepp/blob/420697e4630d892aaf61b0ab8ba797a810ab7e14/uppsrc/Core/CvFlt.cpp>

---

---

Subject: Re: text<->double conversions refactored  
Posted by [zsolt](#) on Sun, 12 Sep 2021 18:50:00 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Thanks!

---

---

Subject: Re: text<->double conversions refactored  
Posted by [Tom1](#) on Tue, 16 Nov 2021 08:44:23 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi Mirek,

Was it this change that caused compilation issue with my call:  
FormatDoubleFix(value, 3, FD\_SIGN|FD\_ZEROS);  
The compiler tells me the flags are not supported anymore.

How do I do get the same effect as with FD\_SIGN|FD\_ZEROS?

Best regards,

Tom

---

Subject: Re: text<->double conversions refactored  
Posted by [Tom1](#) on Tue, 16 Nov 2021 09:03:33 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Nevermind, I found it. Now it's:  
`FormatF(value, 3, FD_ZEROS|FD_SIGN);`  
Best regards,

Tom

---