

## Re: Alternative SZ IntToStrB&V v0.04

---

*Source:*

<http://coding.derkeiler.com/Archive/Delphi/borland.public.delphi.language.basm/2005-09/msg00493.html>

---

- *From:* "Sasa Zeman" <[public@xxxxxxxxxxxx](mailto:public@xxxxxxxxxxxx)>
  - *Date:* 14 Sep 2005 02:18:52 -0700
- 

Dennis wrote:

- > I have to disagree with you again.
- >
- > You just come in and tell us how to do things without showing any signs of
- > respect for what we have been doing for the last 3 years.

Dennis, please do not write that I never wrote.

If I understand you correctly, suggestions to improve benchmark you interpreting as attack on any 3 years effort of fastcode existion?! That is just your own interpretation and your own assumptions. I fail to see why suggestion based on facts are attack on project integrity or perhaps to someone personally.

I really do not prefer conversation like this one and I do not have a time to spend on this. Should we simply abandon this and professionally continue with arguments and facts about benchmarking issues?

- >> In generally it also return far below 1%,
- >> but in shown example that is dramatically incorrect.
- > It is an estimation of the worst case error.

This formula shows "maximum spreading" value only.

Fact is that in special mentioned cases, it can create completly wrong picture.

- > It is standard for Fastcode. This is a Fastcode project.

Clear.

- > Why should it be taken from anywhere. Perhaps we just made it.

As long as I see it is derived from standard deviation formula – based on calculation for one element deviation.

- > When we use the benchmark, we do one run on each of the target PC's. We do
- > not know if this one run is a "perfect" one, a "good" one or one of those
- > single runs that potentially can be far off. We need to assume that it can
- > be the worst possible result. This is why we need to know the worst case
- > error and have a low worst case error.

Problem is that "worst case" depend on situation to situation even on the same PC. That is the problem on multi-application environment and that is main reason why that is very problematic. I'm fail to run current benchmark to test "spreading" value, but on my benchmark that value is not stable (talking in small interval resolution, it vary inside 0.09% – 0.09 %, however sometime exceed 2%, as I mentioned and gives example). Since this is not plain DOS, background activity will always create a problem with that.

That is reasonable argument against showing maximum spreading value.

- > You say that one run that is 10% off matters very little if it is one out of
- > 10. It matters a lot because it could be this +-10% error run we have gotten
- > on eg. the Prescott target. Then we could have taken as winner any of the
- > functions that are within 20% of the winner.

Please do not write that I never wrote.

About though to change scoring. That was just a though, and didn't elaborate it since it is not a suggestion.

I was wrote that calculating index of calculate speed performance vary depending on:

1. CPU and OS
2. Stability
3. Datarange
4. Speed calculation method

If index become stable (under 1%) covering all problematic issues which influence on index, that will possibly be reaslistic.

Points base should be changed, for example to:

1. Ceil(Base point \* Index over RTL)  
Base point may be 10 points.
2. Or, Ceil(Fixed poins \* Index under winning funtion)  
Fixed poins constant may be 100 points.

Index could be also corrected depending on testing on other CPU (probably some correction exist in current scoring calculation), etc...

Anyway, this s not worthed of further detailed analyse.

Sasa

---

www.szutils.net

.

---

• **Follow-Ups:**

- ◆ **Re: Alternative SZ IntToStrB&V v0.04**  
◇ From: Anders Isaksson
- ◆ **Re: Alternative SZ IntToStrB&V v0.04**  
◇ From: Dennis
- ◆ **Re: Alternative SZ IntToStrB&V v0.04**  
◇ From: Avatar Zondertau

• **References:**

- ◆ **Alternative SZ IntToStrB&V v0.04**  
◇ From: Sasa Zeman
- ◆ **Re: Alternative SZ IntToStrB&V v0.04**  
◇ From: Avatar Zondertau
- ◆ **Re: Alternative SZ IntToStrB&V v0.04**  
◇ From: Sasa Zeman
- ◆ **Re: Alternative SZ IntToStrB&V v0.04**  
◇ From: Avatar Zondertau
- ◆ **Re: Alternative SZ IntToStrB&V v0.04**  
◇ From: Sasa Zeman
- ◆ **Re: Alternative SZ IntToStrB&V v0.04**  
◇ From: Avatar Zondertau
- ◆ **Re: Alternative SZ IntToStrB&V v0.04**  
◇ From: Sasa Zeman
- ◆ **Re: Alternative SZ IntToStrB&V v0.04**  
◇ From: Dennis

- Prev by Date: **Fastcode IntToStr B&V v0.19**
- Next by Date: **Re: Fastcode Gettarget unit**
- Previous by thread: **Re: Alternative SZ IntToStrB&V v0.04**
- Next by thread: **Re: Alternative SZ IntToStrB&V v0.04**
- Index(es):
  - ◆ **Date**
  - ◆ **Thread**