- Att_6b_IEEE_signed_review_11_08_00.pdf

Attachment 6b - Review #1 by the signed IEEE reviewer Dr. Les Rogers

At 05:04 PM 11/08/2000 -0500, Rogers wrote:
Dear Dario:

I have read your paper and think I understand the concept. Basicly you describe a pipeline processor that permits you to have one or more stages that can be flexibly expanded to perform computations that take longer than the minimum pipeline element time.

This is necessary when the computation can not be subdivided into multiple subcomputations that could be accomplished by adding multiple pipeline elements in series.

You replace the special purpose switch essentially with an intermediate result bus into which you could plug as many processor stages as necessary.

They are all operating in parallel when the parallel pipe segment is full, and their outputs are synchronized so that their outputs can be picked off a "result bus" by the next serial pipeline element.

My problem is that I do not know if this is a novel concept (replacing a switch with a bus).

Note, if this is an oversimplification, then it is important to point out the difference. The difference may be related to the nanosecond scale of the operations, so if there is something unique that warrants the 3-D flow nomenclature, that needs to be emphasized.

Thus it is unclear whether the point of the paper is to describe a truely novel pipeline architecture, or to show how this architecture is apparently much less expensive to design and build than the switch-based design.

I find the concept logical and appealing, but I am not an expert in trigger processing or pipeline processing, so I have no basis to judge the fine points of your approach.

Figure 4 and Table 1 don't really add anything, or at least I didn't get anything out of them.

Another thing that tends to distract the reader is the infusion of what I might call marketing hype such as in Figure 2 where stage 3 is radiating with a bold face label attached.

If you can let me know what you think the main point of your paper should be, perhaps I could offer some suggestions that would tighten it up.

My suggestion would be to carefully compare the pros and cons of method 4-b or c and your method in terms of cost, generality, speed of designing and modifying and trouble shooting etc. I also think you spend more time than necessary explaining both the problem and what your solution is. This is where the meat is.

--
W. Leslie Rogers
Professor & Research Scientist
Divn of Nuclear Medicine &
Dept. of Biomedical Engineering
3480 Kresge III
University of Michigan Ann Arbor, MI 48109-0552

Phone: (734) 763-3277
FAX: (734) 764-0288

****

Date: Thu, 09 Nov 2000 11:05:17 -0500
To: w.l.rogers@umich.edu
From: Dario Crosetto <crosetto@worldnet.att.net>
Subject: Re: TNS 1999 article re-review
Cc: brillab@ctrvax.vanderbilt.edu,info@3d-computing.com

Dear Les,

Thank you for accepting to dialogue and describing with your own words the concepts of my novel technology.

This will give me an opportunity to address the issues that are not clear.

I hope that you do not have objection if I also send a copy of this review process to Dr. Randy Brill because he asked me to inform him about this review.

Rather than having me attempt to describe my same novel concepts using other words (I will mainly limit myself to pointing out where my definitions of the concepts are in my previous document), allow me to ask you questions and to provide my interpretation of what I understand from your description of them.

My first question is if you agree that students of the middle school implementing my architecture in solution No. 4 on pages 51-61 of [1] (or described in in Figure 2, 3, 4d, and 4e of [2]) could achieve a throughput of a bottle every 6 seconds even if it takes a non-interruptible 28 seconds (unbreakable process) to fill one bottle? If you think that there is any flaw in the description of the exercise of the student and you believe that they cannot achieve that result, please let me know. However, if instead you agree that the result of the throughput of a bottle every 6 seconds can be achieved while the old pipelining mentioned on Solution No. 2 on pages 39-45 of [1] (or Figure 1 of [2]) could achieve only a throughput of a bottle every 30 seconds, then would you agree that my approach has a merit because it a) allows a higher input data rate to be sustained, b) allows a thorough patter recognition algorithm on the input data which best identifies the particles to be executed? In other terms would you agree that the reviewer's statements of my TNS paper that

my "technique of parallel processing within a given, to extend the allocated time for that stage is flawed... [and that my] scheme has no practical implementation merit" are wrong?

Do you agree that the limit of the throughput of Solution No. 3 on pages 46-51 of [1] (or in Figure 4b and 4c of [2]) is 12 seconds because if we attempt to add a sixth station (d) in Figure 25 of [1], that must be placed at a further distance from (c) and (e) (see Figure 25 and middle paragraph on page 50 of [1])?

Please note that the number of units, and students of Solution 3 in Figure 25 of [1], (or Figure 4b of [2]) is identical to the number of units and students of the solution No. 4 in Figure 33 of [1] (or Figure 4d of [2]) of my novel technique. Thus my novel architecture has an advantage compared to the other because with the same number of units and students can achieve the higher throughput of one bottle every 6 seconds compared to a throughput of one bottle every 12 seconds.

Furthermore, my approach also has an advantage in cost because if one tries to layout the "units" (bottle filling stations in a building for the example of Figure 25 of [1], or components on a printed circuit board in the example of Figure 4c of [2]), besides having paths of different length for the "general switch at the system level" of Figure 4c of [2], the routing of the connection will require a more complex, costly solution (see also the last two paragraphs on page 50 and the first two paragraphs on page 51 of [1]).

The simplification of the hardware (thus lowering the cost of the implementation) is also described on page 60 of [1] which states that "… the layout shown in Figure 34 will require only about 200 ps for the signal to go from one chip to the next (at a distance of approximately 6 cm)…" This allows to execute algorithm longer than 200 ps, however sustaining a throughput up to 5 GHz. The same thing is described in the second paragraph in the right column in Section II.D. of [2] where the reference for details is made to Section 6.6 of [3].

Please see my questions and my understanding of your description of the concepts of my novel technology inserted in between ***.....$$$.

At 05:04 PM 11/08/2000 -0500, Rogers wrote:
Dear Dario:

I have read your paper and think I understand the concept.
Basicly you describe a pipeline processor that permits you
to have one or more stages that can be flexibly expanded to
perform computations that take longer than the minimum
pipeline element time.

*** What do you mean by "minimum pipeline element time"? In the old pipeline technique, the pipeline time is given by the input data rate, there is no maximum or minimum element time. Please explain what you mean... In Section II, first page on the last paragraph of the left column of [2] I have described "...that allows for a processing time in a pipeline stage that is longer than the time interval between two consecutive input data" Please let me know if this sentence sounds clear to you or not. $$$

This is necessary when the
computation can not be subdivided into multiple
subcomputations that could be accomplished by adding
multiple pipeline elements in series.

You replace the special purpose switch essentially with an
intermediate result bus into which you could plug as many

processor stages as necessary.

*** What you call "special purpose switch" is it what I have called "general switch at the system level in between Figures 4b and 4c of [2]" ? Please explain what you mean by "intermediate result." I do not have intermediate results.

Where did you see in my document that I make the statement of intermediate results or what is inducing you to make that statement? In the fourth paragraph of Section II.B of [2] I stated that "Each 3D-Flow processor in "Stage 3" (ST_3 in Fig. 2) executes the complete task of the first-level trigger.... There is no division of the trigger algorithm into small steps, each executed by a different processor (or circuit) as would have been the case in a normal pipelined system. If, for example..." I am stating again the same thing in the caption of Figure 2 of [2]. In the first paragraph of Section II.C, I am describing the flow of input data and output results (I do not mention partial results). Figure 3 of [2] shows that the results are bypassed from Layer 1 to Layer 3 without being processed.

What do you mean by result bus? I describe a point-to-point connection where input data and output results are flowing, that is not a bus. In the first paragraph in the left column of the page where there is Figure 4, I stated again "...and each processor is allowed to execute an algorithm (or task) in its entirety. This is analogous to...." By looking at Figure 4b of [2] one can see clearly that the arrows going through one of the two center blocks cannot allow data to go through the second lower block, thus the block should generate a final result for that stage. The equivalent circuit in the boxes shown in center of Figure 4d implementing the same circuit, also generate a final result for that stage.

As you can see, I have stated clearly several times (in text form and in the figures) that was not a partial result. If this issue is not cleared up, you may get into the same misinterpretation of the previous TNS reviewer. $$$

They are all operating in
parallel when the parallel pipe segment is full, and their
outputs are synchronized so that their outputs can be picked
off a "result bus" by the next serial pipeline element.

*** Please explain what you mean by "...and their outputs are synchronized so that their outputs can be picked off a "result bus" by the next serial element." Not only the results are synchronized, but the whole system is synchronous (input, output, processing, moving data between stages and to processors). The outputs are not picked up but they flow synchronously through the system like all other elements. There is no "result bus" but point-to-point connections from processor to processor where are flowing input data and output results. Please explain what you mean by "serial pipeline element." $$$

My problem is that I do not know if this is a novel concept
(replacing a switch with a bus).

*** No one in our field has used the approach that I am putting forth, no one has provided me references of similar approaches used elsewhere, if the reviewer is not aware of the same technique used elsewhere and he assesses that is different from previous techniques, he/she could certainly assesses that is novel to him/her. My technique is not replacing a "general switch at the system level" (see Figure 4b) with a "bus" but with a bypass switch described briefly in the fifth paragraph of Section II.B and extensively in Section II.C of [2] $$$

Note, if this is an oversimplification, then it is important
to point out the difference. The difference may be related

to the nanosecond scale of the operations, so if there is something unique that warrants the 3-D flow nomenclature, that needs to be emphasized.

*** The difference of my novel approach compared to the old pipeline approach is defined in one sentence "extending the processing time in a pipeline stage beyond the time interval between two consecutive input data" and it is described more extensively in the paper, with references to more detailed explanations in [3]. As it is demonstrated in [1] the difference of my novel architecture is not related only to the nanosecond, but it can be related to many other applications, even related to seconds as it is the bottling example of [1]. $$$

Thus it is unclear whether the point of the paper is to describe a truely novel pipeline architecture, or to show how this architecture is apparently much less expensive to design and build than the switch-based design.

*** The point of the paper is both to describe a truly novel pipeline architecture which provides several benefits such as higher throughput, simplified circuit implementation, lower cost. $$$

I find the concept logical and appealing, but I am not an expert in trigger processing or pipeline processing, so I have no basis to judge the fine points of your approach.

*** As is shown in [1], the concept can be explained to 8th grades students and does not require an understanding of the trigger in HEP. The other classical pipeline processing is applied everywhere even to assembly lines in a factory. $$$

Figure 4 and Table 1 don't really add anything, or at least I didn't get anything out of them.

*** Given the number of times I have cited Figure 4 in this email, it seems to contain much useful clarifying information. The simplified "intrinsic switch" shown in between Figure 4d and 4e should help to compare the difference with the implementation shown in Figure 4b and 4c, while for understanding its full functionality, the same switch is shown in more detail in the fourth column (from the left) of Figure 3 of [2]. Please let me know what you did not understand in Table 1. $$$

Another thing that tends to distract the reader is the infusion of what I might call marketing hype such as in Figure 2 where stage 3 is radiating with a bold face label attached.

*** It was not my intention to make what you have called "…infusion of marketing hype such as in Figure 2 where stage 3 is radiating with a bold face label attach…" but my intention is to explain and make clear. I intended with the "radiating" symbol you have mentioned to visualize with a picture the need to extend the processing time in a pipeline stage beyond the time interval between two consecutive input data. $$$

If you can let me know what you think the main point of your paper should be, perhaps I could offer some suggestions that would tighten it up.

*** If you could explain where I may have oversimplified in my article (e.g. that processors generate intermediate results, the result bus, etc.), then I will be able to make it clearer. $$$

My suggestion would be to carefully
compare the pros and cons of method 4-b or c and your method
in terms of cost, generality, speed of designing and
modifying and trouble shooting etc. I also think you spend
more time than necessary explaining both the problem and
what your solution is. This is where the meat is.

*** Figure 4 (that you mentioned does not add anything) summarizes the comparisons, the advantages in simplifying the circuit, lowering the implementation cost, etc. Please let me know how you would modify that figure in order to better compare the different solutions).
I do not understand your statement: "I also think you spend more time than necessary explaining both the problem and what your solution is. This is where the meat is."

Please do not interpret any of my text as anything different than an attempt to clarify, explain and to make more clear my novel approach, if anything might be interpreted as offensive it was not my intention. I am looking forward to receive from you explanations as to where our readings do not coincide (e.g. partial results, the result bus, etc.) and the answer to all my questions, in particular as if my novel technology has merit, or if you see some flaws that were not substantiated by the TNS reviewer. $$$

Best regards,

Dario Crosetto

[1] Understanding a new idea for a Cancer Screening device. Book ISBN 0-9702897-1-5
[2] The article submitted to TNS "System Design and Verification Process for LHC Programmable Trigger Electronics" subject of this review, presented at the IEEE NSS-MIC conference in Seattle on Oct. 1999.
[3] Nuclear Instrument and Methods in Physics Research Section A vol. 436 (1999) 341-385

Anyway I look forward to hearing your response

--
W. Leslie Rogers
Professor & Research Scientist
Divn of Nuclear Medicine &
Dept. of Biomedical Engineering
3480 Kresge III
University of Michigan Ann Arbor, MI 48109-0552

Phone: (734) 763-3277
FAX: (734) 764-0288