

THE DATASAAB FLEXIBLE CENTRAL PROCESSING UNIT

Harold (Bud) Lawson

Lawson Konsult AB, Lidingö, Sweden; bud@lawson.se

Abstract: This panel presentation described some computer architecture developments around the late 1960s and early 1970s. In particular, it placed the Datasaab Flexible Central Processing Unit (FCPU) in perspective to computer industry developments in general. The presentation consisted of several articles as noted in the reference section. It identified papers specifically published about the FCPU as well as other papers related to fundamental aspects of the computer industry. In this paper, I present some further background to the presentation and discussion at the panel session.

Key words: Computer architecture, microprogramming, hardware-software tradeoffs, computer history

1. A CRITICAL TURNING POINT IN THE COMPUTER INDUSTRY

This article reflects my panel presentation given at the HiNC1 conference. Perhaps the most critical turning point in the computer industry occurred in 1963 when IBM introduced in System/360 series. As the suite of system software developed and led to Operating System/360, it became clear that it required significant amounts of machine code to implement basic computing tasks. This led to enormous trade-off problems in decision making at the individual programmer level, for the projects related to major software components (compilers, utilities, and so on) as well as for those responsible for the entire product and its release to customers. Thus, a

mismatch between the hardware architecture and the software systems caused fundamental problems.

As Operating System/360 unfolded, it turned out to be the most complex suite of software ever written. In this era Dr. Fred Brooks, co-architect of the hardware inherited the task of managing OS/360. It is in this era that he wrote his classic book "The Mythical Man Month". The descent deeper and deeper into decaying software architecture marked the beginning of what I have called, "The March into the Black Hole of Complexity". One can compare this decay with the notion of Entropy in thermodynamics. All initial energy dissipated and the system collapsed. Further, all attempts to improve it made it even more complex and subject to further decay.

For the first time, the world was more or less forced (by the clever marketing of IBM 360 products) into accepting unstable platforms of hardware and software upon which they were to build their critical "value added" software applications. Reflecting the decayed state of the software, a new phenomenon evolved; namely, as bug reports appeared and bugs corrected in OS/360, the new releases introduced new bugs. It started a new mental attitude on the part of user communities that gradually accepted poor product quality. A move toward accepting software instability from which the world has never recovered. (The source of many current computer security and reliability related problems).

2. THE DATASAAB FLEXIBLE CENTRAL PROCESSING UNIT

During the late 1960s and early 1970s, a few companies were involved in developing alternative solutions to the IBM computer system architecture. Some of these machines could implement multiple instruction repertoires via flexible microprogrammable architectures. Harold Lawson participated in the design of the Standard Computer MLP-900 in the late 1960s and was the architect of the Datsaab Flexible Central Processing Unit (FCPU) in the early 1970s. The purpose of the FCPU was to emulate the previous D22 instruction repertoire thus providing a D23 identity. There were several innovative features of the 64-bit machine and in particular, an asynchronous approach to the execution of microinstructions. This led to significant gains in simplifying the development of microprogram logic. An important aspect since they believed that they would develop a significant quantity of microprograms. Plans were made to implement higher-level language and system software oriented instruction repertoires, thus simplifying the production of future software. This would have provided a new profile for Datsaab as a supplier of computer systems. The approaches used machines

like the Burroughs B5000, 5500, 6500 provided inspiration as to what high-level architectures could accomplish.

3. WHAT HAPPENED?

The introduction of the microprocessor technology in the mid-1970s and its rapid growth changed hardware economics. This, in retrospect has led to a false view of the economics of the entire suite of hardware and software. The microprocessors of today still reflect the primitive instruction repertoires of the 1970s. While the microprocessors of the 1970s were quite okay for simpler applications, today they are used as the platforms for systems of enormous unprecedented complexity. They were not designed for this purpose. An absurd situation that is just accepted!!! The world pays the price repeatedly for the costs of poor software quality partially resulting from the fact that the software runs on a low-level machine where complex trade-offs must be made. We can contrast this to a high-level machine architecture where there is typically only one or only a small number of best ways of accomplishing a particular computing task. That makes decision making much simpler.

4. WHAT COULD HAVE HAPPENED?

Had the microprocessor not made an entry, it is interesting to speculate what could have happened in respect to the FCPU. The FCPU stirred up significant interest in the Nordic countries as well as some places in the USA and England. There were discussions between Datasaab and Regnecentralen concerning cooperation in using their high quality operating system RC-4000 as a basis for exploiting the flexibility in the FCPU – that is, tailored instruction sets to support the OS. At a later stage, some meetings with Burroughs occurred to discuss cooperation. There was also discussion with Motorola about cooperating on specially designed integrated circuits for the future. None of these business opportunities came to fruition.

5. WHAT HAPPENED TO THE FCPU?

The Styrelsen för Tekniska Utveckling created Project P about 1973 and managed to get several researchers interested in the FCPU. Amongst other activities, Jaak Urmi did work on a LISP machine at Uppsala and Sven Tavelin at Chalmers performed work on operating systems.

Commercially, I believe there were nine processors built. Several were delivered to customers and ran as D23s, some into the late 1970s. The FCPU resulted in significant interest in the US and at several universities in the USA and Europe where the FCPU concepts were taught as a part of computer architecture courses. As noted in the reference list, several scientific paper contributions were made. In particular, the paper on the FCPU by Lawson and Magnhagen was selected for the Best Paper Award at the Second Annual International Symposium on Computer Architecture. Some work was done on providing higher-level instructions; in particular, vector arithmetic instructions were implemented via microprograms that resulted in significant performance improvements. However, the FCPU was not further commercially developed according to original plans to support the implementation of higher-level languages and operating system functions. Very unfortunate!!

In the late 1970s, the LYSATOR computer club at Linköping University installed one of the few remaining FCPUs. Some experimental work, including a Forth language interpreter became operational on the machine. LYSATOR members had fun and learned a lot in experimenting with the novel features of the machine. That FCPU hardware eventually wound up in a barn in Östergötland county and is, as far as I know, the only remaining machine. Hopefully, it will become part of the Linköping Computer Museum.

6. PATENT CASES

The FCPU has been referenced in two patent infringement cases. One involving infringement upon byte instruction logic of the Intel 386/486 microprocessors. While much more primitive than the general concepts of the FCPU, the byte instructions of the Intel processors operated asynchronously, like the FCPU. Thus prior art existed and was published, thus Intel should not have received a patent. In another case, a smaller computer manufacturer in the US used many FCPU concepts in developing a computer in the late 1970s and later claimed that a very large manufacturer (who also remains unnamed) had copied these architectural concepts.

7. CONCLUSION

A permanent documentation of the concepts and ideas behind this innovative asynchronously controlled 64-bit microprogrammable processor is provided in the papers produced. A modern version would be a good

instrument for improving function distribution between hardware and software and lead the way toward more stable and secure platforms—just waiting for someone to reinvent the wheel!!! I would imagine that this would happen sometime, perhaps not in my lifetime.

REFERENCES

FCPU Related References

- Harold Lawson and Bengt Malm , A Flexible Asynchronous Microprocessor, The Scandinavian Computer Journal BIT, Volume 13, Number 2, June 1973.
- Harold Lawson and Lars Blomberg. The Datasaab FCPU Microprogramming Language, Proceedings of the SIGPLAN/SIGMICRO Interface Meeting, May 1973. Also published as Saab Scania Report GM-72:296, November 1972.
- Lars Blomberg and Harold Lawson, SILL Specification (A System Implementation Language Proposal), Internal Saab Scania Document, May 1974.
- Harold Lawson and Bengt Magnhagen, Advantages of Structured Hardware, Proceedings of the Second Annual Symposium on Computer Architecture, Houston, Texas, January 1975. [Selected as the best paper at the Symposium.]
- Harold Lawson and Bengt Malm, The Datasaab Flexible Central Processing Unit, A chapter appearing in the Infotek State of the Art Series Report 17 on Computer Design, 1974.
- Harold Lawson, An Asynchronous Approach to Microprogramming, A chapter appearing in Microprogramming and Firmware Engineering Methods (editor: Prof. Stanley Habib), Van Nostrand-Reinhold, 1988.

Other References

The following references relate to papers where the fundamental problems of the computer industry are discussed.

- Harold Lawson, Function Distribution in Computer Systems Architectures, Proc. 3rd Annual Symposium on Computer Architecture, 1976.
- Harold Lawson, Salvation from System Complexity, IEEE Computer, Vol. 31, No. 2, Feb 1998, pp 118-120.
- Harold Lawson, Infrastructure Risk Reduction, Communications of the ACM, Vol. 40, No. 6, June 1998, pp120.
- Harold Lawson, From Busyware to Stableware, IEEE Computer, Vol. 31, No. 10, Oct 1998, pp117-119.
- Harold Lawson, Rebirth of the Computer Industry, Communications of the ACM, Vol. 45, No. 6, June, 2002, pp 25-29.