

-----  
From: Tony Tebby  
To: ql-us...@q-v-d.com  
Sent: Tuesday, 21 March 2017, 10:40  
Subject: Re: [QI-Users] Stella

The Stella project can be viewed in two ways

Negatively

- 1) You need a system development environment that
  - a) will support different processors - ruling out assembler
  - b) does not impose UNIX type structures - ruling out C, C++, etc
- 2) You need to develop interface modules (drivers?) for a wide range of "peripheral" devices - the device manufacturers will not develop them for you.
- 3) You need to develop an applications base for a new market
- 4) You need to develop entirely different hardware architectures - this century, architectures have become ever more tightly dedicated to UNIX (Linux, Windows NT, ...)

Therefore you need \$.

Positively

- 1) A first version of the Stella core was coded in MC68000 assembler and benchmarked against Solaris 2 (UNIX SVR4) showing orders of magnitude (under unrealistic, best case, benchmark conditions) lower system overheads. Sun Microsystems rejected the idea of developing the system saying that it would not work on either SPARC systems or symmetric multiprocessor systems (false) and that benchmarks were not reliable indicators (true). Instead they bought Chorus, a UNIX variant, which disappeared without trace. Not everybody will always be so stupid
- 2) There are signs that the "you can always use a more powerful computer" is becoming less acceptable as an excuse for chronically inefficient software.
- 3) There is still no end in sight to the discovery of new "exploits" in all sorts of Unix based devices. Unix is fundamentally unsound as well as chronically inefficient.
- 4) Unix type systems will inevitably be wiped out by other systems. We could hope that, unlike Unix, these will be theoretically sound and fit

for purpose. This may not happen in our lifetime.

5) The only real barrier is ignorance. The story of Android is illustrative. If, in 2005, Google had known that they could have developed an entirely new operating system and had it up and running sooner than trying to kludge something out of Linux, would they have bought Android Inc. for \$50M? Of course not, but they did not know.

Tony

-----

Re: [QI-Users] Stella  
2017-03-21 Thread Peter Graf  
Hello Tony,

it is good to read something from you.

- > 1) You need a system development environment that
- > a) will support different processors - ruling out assembler

That depends on where one looks for a niche market. Things have changed inasmuch small to medium sized FPGA have become large enough to contain a whole 32 bit CPU.

In addition to that, patents on the original 68000 have expired.

Because of the high code density, I found the 68K architecture best suited for use with the limited internal RAM resources of an FPGA. I have tried several 32 bit architectures, and 68K allows the smallest memory footprint.

Within small to medium sized FPGA systems, having a real-time operating system would often advantageous, but existing systems are simply too large.

For such targets, the best would be an assembler written OS, still allowing application code in C. Like SMSQ/E and C68, but capable of real time operation and oriented toward embedded systems instead of personal computing.

- > 2) You need to develop interface modules (drivers?) for a wide range of
- > "peripheral" devices - the device manufacturers will not develop them
- > for you.

This also improves with FPGA - once there is a peripheral device in HDL

(Hardware Definition Language) you can at least keep it the same for the next product generation.

- > 4) You need to develop entirely different hardware architectures - this
- > century, architectures have become ever more tightly dedicated to UNIX
- > (Linux, Windows NT, ...)

Maybe you can take the Q68 as a modest proof that a new hardware architecture can be close to 68K and the driver requirements of an assembler OS. I developed the Q68 as a hobby project, so is now 10 years old. Therefore it can not demonstrate the latest chip performance, and of course the Q68 has a homecomputing flavour.

But updated to latest chip generation, and adapted to an embedded application, such a system suddenly could make a lot of sense with Stella.

- > 1) A first version of the Stella core was coded in MC68000 assembler and
- > benchmarked [...]

So nothing would be better to continue development of the Stella than the 68K architecture.

- > 3) There is still no end in sight to the discovery of new "exploits" in
- > all sorts of Unix based devices.

This is an important point. I have been joking with friends, that we might have to go back to 68K for internet use, simply because the architecture is too exotic to be exploited. Also, security requires lowest possible complexity.

All the best  
Peter

-----

Re: [QI-Users] Stella  
2017-03-23 Thread Arnould  
Nothing is happening with Stella. First this document was not intended for publication. Thierry Godefroy insisted to do it.

Then in fact Stella is a concept for a range of operating systems. Tony kept telling me that it would be more efficient to develop hardware first and then dedicated OSes rather than try to port monster OSes to different devices.

Let us find a billionaire who would like a real "smart" phone and who could pay for 10 good programmers during a year, and then Stella could

become something.

I have also come to the conclusion that not only money but also language is a major brake to evolution. Google "Performance Impact of Lock-Free Algorithms on Multicore Communication APIs" and try to read it. This paper more or less comes to the same conclusions as Tony and a few others. And then the authors conclude that existing systems should be patched... But where are the "managers" who can understand this jargon? I think that they do not exist.

Arnould