Begin forwarded message:

> From: Rick Auricchio <rick@cfcl.com>
> Date: Wed Dec 18, 2002 12:08:09 PM US/Central
> To: Steven Weyhrich <sdweyhrich@mac.com>
> Subject: Re: Apple Ile question

>> Do you remember what parts of DOS 3.2.1 and 3.3 you worked on?
>
> DOS 3.2.1 was a patch to fix disk errors. In late '78 and through part of
> '79, Paul Lutus (developer of AppleWriter) had been experiencing I/O errors
> on his dual-floppy system. He was one of the few in the world with one,
> outside of Apple.
>
> One day, while running his test script (thank goodness he had one), I heard
> the "deselected" drive click during a switch from drive 1 to 2 (or 2 to 1,
> it doesn't matter). It turns out the head on that drive was stepping a
> little bit when it shouldn't have, thus positioning itself off-track. I
> don't recall why a recalibrate didn't recover, but it could be because the
> head was 1/2 track off and it got *some* data, but failed to get it all. I
> believe if you could read address mark headers you didn't recalibrate.
>
> I called Woz over, we looked at it, and he found a new capacitor on the
> motor-control board. Shugart had added this to smooth out the power without
> Apple's knowledge; he attached a scope and saw the power stayed up on the
> drive for perhaps 100mSec after it was deselected.
>
> The seek routine immediately began stepping the "new" drive head right
> after turning off the "old" drive. Because the power didn't shut off
> immediately to the old drive, its stepper had enough power to click a
> little bit.
>
> The problem didn't occur on older drives within Apple, because they didn't
> have that capacitor.
>
> Woz and I added a 150mSec delay to the start of the seek, so there was
> enough time for the old drive to really die. This delay was invisible,
> because the seek would still complete before the spindle motor was up to
> speed.
>
> That was DOS 3.2.1.

-------------------------
For DOS 3.3, I integrated Woz's 16-sector disk "core routines."

Shephardson Microsystems did the majority of the work on the original DOS 3.

Apple bought DOS from them.

Randy Wigginton and Steve Wozniak wrote the RWTS part of it.

Correct. RWTS, Read-Write Track/Sector, was the main entry point of the "core routines," the floppy driver code.

I was also told that Dick Huston did work on modifying DOS 3.x.

Dick maintained DOS in general, fixing bugs. DOS 3.1 had plenty; DOS 3.2 fixed a lot of them. Dick also knew the core routines, so he perhaps helped with those too.

Also, legend had it (at least at one time) that there was no assembly source file for DOS (at least on the Apple II), but that it was patched via the mini-assembler. Is there any truth in that?

Not true, but close. DOS source lived on an S-100 Z80 system, but I forget what kind. That's where Dick made the changes. Some time after I got the assembler in good shape on the Apple II, DOS got moved there. I don't recall who moved it over. Could've been me, could've been Dick, or perhaps John Arkley.

GameBasic on the Apple I, predecessor of Integer Basic, was hand-entered in hex by Woz at parties. The Mini-Assembler came later than that...

David Craig told me that there was a secret in the Apple III called "Devil Mode"

"Satan Mode," a name I made up. Andy Hertzfeld discovered that you could use the /// as a II with access to the various features, if you didn't require ALL the hardware compatibility that the "Apple II switch" enabled.

Initial attempts to use it were risky, because there were occasional things you'd forget about and you'd just crash. He then created a "Satan Mode Boot" diskette, which I later used to run the SubLogic FS-1 flight simulator at the increased processor speed.

I don't believe anyone bothered to try using bank-switching, extended indirect addressing, or 80-column video in Satan Mode; it was a novelty that was neat but didn't become very popular.

Once you flipped the /// into II Emulation Mode, you couldn't flip it out without a reset, nor could you access any of the advanced features.

[made floppy driver] "correct", was it an issue of functioning correctly

Yes. It was in ROM so the system could boot. Dick Huston fixed "the last bug" and offered a $50 challenge to find another. Two days later I smugly...
handed him about seven bugs! I declined the reward, but he insisted on paying.

One of the bugs caused multiple drives to fail, so I got the task of fixing the driver. Dick was beginning on ProDos---the Disk Division was hot to sell the 5MB Profile hard drive for the II machines---so it wasn't like they took him off the floppy and gave it to me.

I knew how the core routines worked on the II, and the /// was similar, but I had not actually done a floppy driver in its entirety. I spent a few weeks reading his code, understanding what it had to do (and how it did or didn't do the job). A lot of his code, in tight assembly fashion, was intertwined. The read and write paths, for example, came through the same block of code, with tests all over saying "reading?" or "writing?" to branch here and there within the block.

I went so far as to completely flowchart a driver, realizing that it was faster and smaller to simply write multiple routines. Then create subroutines from the duplicated code. This is straightforward design, nothing special.

I then began coding and had the driver operational in a couple of weeks. Coding and testing went quickly once problems had been thought-out in the design phase. Separating fundamental things like the read and write path made it easier to debug; with a common path you can't debug the first write when it's going to do loads of reads first!

The ROM-based driver was fine for booting, because it didn't have to handle multiple drives, and, if there was a Write bug we wouldn't be doing writes. So we kept the ROM as is and made the floppy driver installable. When SOS loaded, it had a ram-based floppy driver.

>> do you recall if the driver functioned differently from the one in DOS 3.x on the Apple II?

The driver was essentially identical in function, though the /// could have four floppy drives connected. The API was different, since SOS used the unix-style read/write/open/close/ioctl paradigm. The core routines were rewritten, but they were still about the same as those Woz had written in the II. They would be considered derivative work if you pushed on the copyright, for sure.

>> Were you the sole author of SCP [/// system configuration program]?

Yes.

I also did much (or all?) of "Selector," which was a menu-based program launcher for the ///. I don't know how much of it I did, but I know I did the visual stuff. All that text-based folder-image scrolling and drawing.

Because Thomas Root did a great job with a smart-terminal interface for the console, a lot of the animation was done with a stream of control characters! I'd just package up viewport-setting controls, scrolling operations, and so forth, and the animation would happen in the driver.

>> I know that making the changes in the ESC cursor movement code to allow the use of the arrow keys was a BIG improvement for me, as I did a lot
of Applesoft programming at the time the IIe came out, and it just made
more sense than did the IJKM keys. Also, having the inverse "+" appear
when using the 80-column firmware to indicate that you were in cursor
movement mode was a BIG help also.

I forgot about that stuff. Remember uppercase-restrict? That was something
that I recalled from the Xerox CP-V timesharing system: if you typed in
lowercase, the system would upshift it to interpret it. I think that went
in too.

Did you see any difference in the corporate culture when you returned
for the second time, or for the third time? Was it very different as a
(I presume) bigger company? Is there a time of working at Apple that
you feel was most enjoyable for you?

My times were: Mar 79-Jan 83, Sep 85-Oct 92, Sep 95-Oct 01.

The first stint was when we had 75 people in Engineering. It was a blast,
because I'd worked with a friend building a 6502 computer from scratch in
1975. I bought my Apple II in 1977, and the learning environment in the
early Apple days was great.

Andy Hertzfeld joined about six months after I did. He and I had been
writing articles for Micro Magazine, and we both wrote an article showing
an ONERR-GOTO patch for Integer Basic. We "recognized" each other from
that. Alan Watson, who may still be at Apple writing technical manuals,
also wrote an article on which DRAMs would work in the II. At the time he
was working for Fairchild or another chip manufacturer (Fairchild?). I
mentioned the article when we were introduced.

The second stint was after I'd learned UNIX and C. Five of us started the
A/UX project. There I got to learn unix in great detail, wrote more drivers
for various Macs around the Mac-II timeframe, and we did lots of good stuff
with A/UX.

The third stint, after Taligent, was working on the CHRP project. I then
moved to the Rhapsody project shortly after the Gil Amelio layoff binge.
There, I worked with former-A/UXers and former-Taligent folks along with
the NeXT people.

As time went on, more after 1995, you could see the big-corporate changes.
Benefits became weaker and cost more, even though the employees were older
with families (when we were young and single, we didn't need all the good
benefits!)

Corporate bean-counters began cutting expenses as they always do: they cut
what's easy, but not what really COUNTS.

For example, at one point in the early 90s, someone decided to cut out the
T-Shirt budget. Why? Because it was an easy-to-spot line item on someone's
budget. Never mind the morale backlash. That got overturned at some point,
but not after it had done some morale damage.

You see this all the time in organizations where the finance people are out
of touch with the organization. They cut a line item without really knowing
whether it'll make a difference. You have to cut the EXPENSIVE stuff, not a
> little cheap thing.
>
> For example, they'd cut some minor thing that people liked, then the
> Repro/Printing department would print thousands of notices about changing
> out the Xerox machines for another brand. Instead of putting one notice at
> the machine they plastered them all over the building. Waste.
>
> The Telecom people would print thousands of GLOSSY card posters telling us
> they're adding a new prefix for extensions (974-xxxx and now 862-xxxx). BIG
> expense, and for what?
>
> Anyway, I'm rambling...it's just that I see this stupidity all over.
> Someone wants to cut expenses and he begins cutting things without first
> looking at where the money is actually going.
>
> --
> - rick
> -
> ___________________________________________________
> -- -
> Rick Auricchio  rick@cfcl.com http://www.cfcl.com/rick
> Acoustic Legacy Studios  Cambria, CA USA 93428  805-927-7305
> Years ago, I sent my mojo in for repairs. I still don't know if it's
> workin'.

Steven Weyhrich <IX0YE>--<
Apple II History
http://apple2history.org