

Date: 22 June 1981 1:14 pm PDT (Monday)

From: Ogus.PA

Subject: Dandelion RS232C DMA problem

To: Wick, Lauer

cc: Schwartz, Danielson, Garlick, Hutson, Charnley, DDavies, Yamanaka, Ogus

I am forwarding Dan's message which sums up the findings of his RS232C DMA testing. Basically, due to an (undocumented) incompatibility between the Zilog SIO chip and the Intel DMA controller, it is not possible to reliably implement DMA transfers in the current system. There are two problems (see below), the first of which could perhaps be worked around in software. The second is not possible to fix reliably. Dan lists four approaches to attempt to obtain 56 kbps data rates. The first (Dan's #2) is to attempt to do so without using the Dma controller, using the current interrupt mechanism. The current code could be streamlined, and perhaps unnecessary function removed through the use of overlays. This would work with the current hardware, all would be the quickest solution to the problem. I recommend that this approach be focused on right now.

The second (Dan's #1) would be to ignore the setup time problem and to implement the software workaround. This solution would probably work in a lot of machines, but would not work in general. It cannot be used in a product. The third and fourth solutions involve redesign of hardware. The first redesign would involve attempting to salvage the DMA function through the use of other chips instead of the SIO. We would result in another Options card which had the Dma capability, but still had the limitations of the current Options card (only one RS232C channel). There is also some risk in that the alternate chips are not yet available. The fourth approach is to expand the Dandelion by adding a high speed serial port which could handle several RS232C ports, including one (or more) at 56 kbps.

We do not recommend designing another limited Options card, but favor the approach of "doing it right" and building the expanded RS232C capability. This would save the extra development effort and cost, and not produce an extra board which would soon be obsolete. We are working on the expanded system design schedule, and will need to discuss with you the match with the 56 kbps schedule requirements.

Roy.

-----

Date: 18 June 1981 2:20 pm PDT (Thursday)

From: DDavies.PA

Subject: Zilog SIO in Intel System

To: Ogus

cc: Garner, BELLEVILLE, DDavies

Roy,

The Zilog SIO chip cannot be used in applications requiring DMA in our current Intel system. There are two fundamental problems.

1. The Zilog DMA chip inserts 3 clock cycles before any activity. The Intel 8257 does not. In a Zilog system, the SIO would have time after any access to prepare for the next one. The Intel DMA chip may start an access to the SIO one clock cycle after the processor has finished one. The exact SIO response to this is not known (we can't look inside the chip), but the SIO does generate a spurious DMA request immediately. In experimentation, the interference has been seen to cause the SIO to insert or delete data bytes, decide all packets had bad checksums or simply stop receiving.

2. The Zilog SIO chip uses the IOPClk to sample bus control signals (CE', IORQ', etc). Intel chips use the processor clock only to provide rough internal timing, not to strobe bus signals. Thus the bus signals produced by Intel parts do not satisfy any particular set-up or hold times with respect to the processor clock. In particular, timings for the signals produced by the 8085 may differ from those produced by the 8257 by 130 to 150 ns. The SIO's clock in our present system has been adjusted so the signals produced by the 8085 meet its set-up and hold requirements. Because of this, the signals produced by the 8257 DMA chip do not meet the requirements. This seems to have had no detrimental effects to date but would probably cause problems in manufacturing. Because of the loose relationship between the processor clock and the bus signals, no hazard-free method of synchronization has been found.

There are at least four proposed methods of obtaining 56 kbaud data rates.

1. Ignore the SIO's bus timing problems and disable DMA access whenever accessing the SIO chip from the 8085.

This would probably cure the DMA problems seen in the lab. There is no guarantee that even those boards initially meeting the timing requirements would continue to do so in the field. We do not recommend this option.

2. Attempt to develop IOP code that could maintain a full-duplex SDLC channel at 56 kbaud using interrupt routines to process each byte.

There is considerable pessimism on this subject. At 56 kb, a byte arrives roughly every 142 fIS. For full duplex, the IOP must handle a byte every 71 fIS on the average. When data is being transferred through the CP port, the DMA controller takes about 4 cycles out of every 12, leaving the CP running at 75% normal speed. Thus, an interrupt routine should take no more than 53 fIS per byte maximum, 40 fIS to be safe. This is hard. If we assume roughly 2 fIS per instruction, this is 20 instructions. Maybe it could be done, maybe not. Since it could be a low cost, quick and reliable solution, it should be tried.

3. Design an interim Options and/or IOP to handle 56 kbaud, then design another Options and/or IOP to handle multiple ports, magnetic tape, etc.

The interim design would probably involve using the not yet qualified Intel 8274 in place of the SIO chip or replacing the 8257 DMA with the newer 8237. The 8274 is quite similar to the SIO, but is just now being sampled. We would have to gamble that it would be available in quantity and would be qualified quickly. We could also keep the Zilog SIO and try the newer Intel DMA controller (8237). Zilog thinks this works, but they thought our current system

should have worked until the problems were explained to them.

This is the approach least favored by us as it would involve a reasonably large amount of design work and the resulting product would quickly be obsolete. It would also delay the introduction of a board capable of handling Dandelion expansion.

4. Wait until a new Options card having a high-speed serial port is ready.

As currently envisaged, this board would provide access to the expansion everyone wants. The serial channel would probably be connected directly to the CP so it could run at up to 7 MHz in server applications (less if a display was needed since it would need to share someone else's clock). An exterior box could hold multiple controllers for RS232 lines, Magnetic tape drives, voice, etc. Alternately, the line could be used as a high-speed RS232C line directly and the protocols could be handled in microcode. This approach would allow us to provide a single link with almost arbitrary speed and signalling convention.

This is the favored long term approach. Much of the time needed for any design is taken in overhead, debugging, layout, test specification, drawing packages, consultations with manufacturing, pre-prototypes, prototypes, pre-production prototypes, and so on. The time taken for many of these steps is independent of the similarity between the new board and an old one. The new Options board will be needed with or without an interim model. The new board will be significantly delayed if an interim model is required.

In conclusion, the current system cannot support DMA access to the SIO chip. As a quick fix, we should try to attain 56 kBaud using only interrupt processing. In the long term, we should proceed with the design of a new Options card to be used to provide multiple RS232 lines, high-speed lines and access to new peripherals.

Dan

---