To: Distribution

From: N. I. Morris

Date: December 31, 1973

Subject: Ideas for a GIM replacement

GIM Draw-backs and Deficiencies

This document proposes a replacement for the GIM (GIOC Interface Module). The GIM is a large hardcore ring subsystem designed to allow user ring programs to perform peripheral I/O. It was originally written to interface a GIOC and has since been retro-fitted to work with an IOM while continuing to simulate a GIOC. It performs copying of data and copying and validation of DCW's between the user ring and the hardcore ring. The GIM is slow and inefficient and requires large amounts of wired-down buffer space, even when no I/O is being performed.

The GIM allocates buffers in a wired-down area and copies data based on the tallies of DCW's supplied by the user. Using the GIM in a T & D application where larger buffers must be allocated and then checked for channel overrun would be difficult, if not impossible. (A channel overrun occurs when a data channel, through a hardware error, transfers too much data.)

I/O Interfacer Features

The replacement for the GIM proposed here would have none of the above draw-backs. This I/O Interfacer would allow the use of user ring supplied data and DCW lists with no software checking of DCW's and no copying of data. A user would be free to specify any DCW list he desired and even to actively patch it while the IOM channel was in operation. Note that such an I/O Interfacer would allow such hardcore ring programs as the printer DCM, tape DCM, and IMP DCM to be moved from the hardcore ring to the user ring with little or no effect on performance.

IOM Hardware Protection and Relocation

The I/O Interfacer would rely heavily on the relative addressing feature of the IOM to provide hardware checking of all DCW's. This IOM feature allows the system to specify the high-order 9 bits of a base address and boundary for each IOM data channel. Each user of the I/O Interfacer could be given a

---

block of core starting at a 0 mod 512 absolute address and extending for a multiple of 512 words with complete assurance that he could wreak destruction only upon this block of core. The relative addressing feature of the IOM will automatically relocate each DCW address (including Transfer DCW's) by the base address and then check the address plus tally of each DCW to insure that it does not exceed the boundary. Thus, the IOM will refuse to make any data references or DCW list services outside of the block of core assigned to the user.

IOM Hardware Deficiencies

One serious flaw does currently exist in this scheme: The address extension (high-order 6 bits of the 24-bit absolute address generated by the IOM) used by a channel can be changed during channel operation by setting appropriate control bits in an Instruction DCW. Since the relative addressing hardware in the IOM checks only the low-order 18 bits of an address, obviously a gaping security hole exists. A simple hardware modification to prohibit changing the address extension of a channel when relative addressing is in use on that channel is proposed in an appendix to this document.

Since the IOM relative addressing hardware performs address checking on a DCW prior to using that DCW for actual data transfer, a channel overrun could result in data being transmitted to or from an area of core outside the assigned block of core. Note, however, that in the current implementation of I/O software on Multics, no protection exists against such a channel malfunction. In fact, such malfunctions are rare, and it is virtually impossible to protect against their occurrence.

I/O Interfacer Operation

Only five functions would need to be provided by the new I/O Interfacer:

1. Attaching a channel
2. Allocating wired-down buffer space
3. Connecting to a DCW list
4. Returning hardware status
5. Detaching a channel

Since the user ring program would have complete access to both data and DCW lists, it would be wholly responsible for maintaining the DCW lists and performing the necessary data copying.

When the attach entry is called, a unique device index will be assigned. Also, an event channel will be supplied to the I/O Interfacer to be used in waking the user when his I/O has

completed. A buffer segment will be created in his process directory. It will be accessible only to that user in all rings from the hardcore ring to the user's ring, inclusive. The segment will not be wired-down at the time of the attach call.

The allocate entry in the I/O Interfacer will set the required length (in multiples of 512 words) of the buffer segment and wire that segment down. Some limitation will have to be provided on the size of the wired-down buffer. Also, if the buffer is longer than the size of a Multics page, steps will have to be taken to insure that the buffer segment is in contiguous core and does not cross a 256K word boundary. Upon return from the allocate entry, the user will be able to set up his DCW list and data in the buffer segment and begin to perform I/O. Subsequent calls to the allocate entry can be used to allocate more (or less) wired-down buffer space.

The connect entry will connect to any DCW in the buffer segment. A PCW will be manufactured by the I/O Interfacer and a new entry in the iom_manager will be called to connect the channel in relative mode. Illegal DCW's or illegal DCW sequences will cause IOM central or channel errors which will be reflected back to the user in his hardware status. One extra precaution will be needed in the connect entry if the I/O Interfacer is used to operate channels which have separate discrete devices (e.g. magnetic tape handlers): The first Instruction DCW (which contains the device code) will have to be validated and copied into a protected area in the hardcore ring so that the user cannot change it while the connect is taking place. The device code in subsequent Instruction DCW's is ignored by most channels.

The status entry will return any hardware status which may have come in for the channel. All types of status which can be generated for an IOM channel will be returned:

1. system fault status
2. terminate status
3. marker status
4. special interrupt status

In addition, the status entry will return the "cur_status", the current DCW list position of a channel in operation. This will enable the user to successfully add to a DCW list while the channel is in operation. This is similar to GIM operation, except that no test will be made to protect the user from accidentally overwriting a DCW that the IOM is currently using.

The detach entry of the I/O Interfacer will insure that no more I/O is taking place on the channel being detached. It will then unwire the previously allocated buffer space and detach the channel.

As in the GIM, the I/O Interfacer will call appropriate entries in the I/O Assignment Module. This will insure that if the user's process is accidentally or deliberately destroyed, a special entry in the I/O Interfacer will be automatically called to stop all I/O, release all wired-down buffer space, and detach the channel.

Minimizing the Use of Wired-down Core

An additional feature in the I/O Interfacer can be provided, if desired, to minimize tying up wired-down core when a channel is not being used heavily. A timer could be set in such a way that if no I/O has taken place on a channel for a certain period of time, the wired buffer for that channel would be unwired. The next call to the connect entry would have to rewire the buffer before issuing the connect. This technique is extremely useful on channels driving devices which remain attached in an idle state for long periods of time (e.g. card reader, punch, printer). System resources would be drawn upon only when actually needed.

Access to the I/O Interfacer

Use of the GIM is currently restricted to privileged users. However, it is expected that the I/O Interfacer will be used for a much wider range of applications. Therefore, it is desirable to make it available to any user and also to take steps to prevent its misuse. Use of the attach and allocate entries will probably be limited to the administrative ring. An administrative ring module can be provided to prevent any user from grabbing excessive amounts of system resources or from attaching any channel to which he should not have access.

I/O Interfacer Specifications

This document is intended as a preliminary draft of ideas which the author has had concerning this new I/O Interfacer. It would be premature at the point to publish a full set of specifications until design reviews have taken place and all comments and suggestions have been carefully weighed. Therefore, the detailed specifications of the I/O Interfacer will appear in a later Multics Technical Bulletin.

Appendix

IOM Design Specification Change

In order to prevent a user from changing the address extension of a channel, the specifications of the IOM must be changed as follows: Require IDCW bit 21 (the address extension control bit) to be off whenever LPW bit 23 (the relative mode bit) is on. One of two actions could be taken if IDCW bit 21 is found to be on when LPW bit 23 is on:

1. Ignore IDCW bit 21. Do not change the address extension.

2. Terminate channel operation. Reflect an IOM central status of 4 (octal) to the system. (This status code is already returned if a Transfer DCW illegally attempts to change the address extension for DCW list service.)

This change should have no effect on the use of the IOM by GCOS. Under GCOS operation, a DCW list would never contain an IDCW which attempted to change the address extension while operating in relative addressing mode.