NASA TECHNICAL MEMORANDUM NASA TM-77241

DATA HIGHWAY TERMINAL CONTROLLER

T. Yoshimoto

Translation of Japanese Kokai Patent, Tokyo Shibaura Electric Co. Ltd., Kokai No. 56-40944, April 17, 1981, pp. 225-228

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
WASHINGTON, DC 20546 MAY 1983

STANDARD TITLE PAGE

1. Report No. NASA TM-77241	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle: DATA HIGHWAY TERMINAL CONTROLLER		5. Report Date: May 1983 6. Performing Organization Code
7. Author(s): T. Yoshimoto		8. Performing Organization Report No. 10. Work Unit No.
9. Performing Organization Name and Address: Leo Kanner Associates, Redwood City, CA 94063		11. Contract or Grant No. NASW-3541 13. Type of Report and Period Covered: Translation
12. Sponsoring Agency Name and Address: National Aeronautics and Space Administration, Washington, DC 20546		14. Sponsoring Agency Code

15. Supplementary Notes

Translation of Japanese Kokai Patent, Tokyo Shibaura Electric Co, Ltd., Kodai No. 56-40944, April 17, 1981, pp. 225-228

16. Abstract

The invention describes a data highway terminal controller with the following features: transmission controllers are assigned to each processor in a set of multiple processors. Shared-use terminals are loop connected to the processors via the controllers.

17. Key Words (Selected by Author(s))	18. Distribution Statement: Unclassified-Unlimited		
19. Security Classif. (of this report): Unclassified	20. Security Classif. (of this page): Unclassified	21. No. of Pages	22.

Application for Patent /1*

1. Name of Patent

Data highway terminal controller

2. Range of Patent Application

(1) The present invention is a data highway terminal controller with these distinctive features. Transmission controllers are assigned to each processor in a set of multiple processors. Shared-use terminals are loop-connected to the processors via the controllers. When a request is made to use a processor set between transmission controllers and terminals, the data highway controller receives the address data for the requested processor. If a terminal is not being used by another processor, a response goes to the first processor saying that the terminal can be used. If a terminal is in use, the controller sends data on terminal status and an estimate of the time before the processor will be able to use the terminal. A time value greater than the estimated time value is reported to the processor.

(2) The present invention is a data highway terminal controller as described in Paragraph 1, above, of this Range of Patent Application with these distinctive features: When the processor makes an /2 allocation request, the terminal controller stores data in the terminal allocation request queue located in the memory of the terminal. At the instant the terminal completes processing, it updates the queue, retrieves data for the next allocation queue, and informs the processor waiting for allocation that terminal allocation has been completed.

3. Detailed Description of the Invention

The invention is an improved, shared-use terminal controller connected in loop to data transmission lines (hereinafter called data highway) by multiple processors.

---

*Numbers in the margin indicate pagination in the foreign text.

When multiple processors in conventional systems share the use of terminals connected to a data highway, an allocation request for the processor is made first. When a reply is received saying the processor can be used, the system moves on to the next I/O operation. If the received reply says the processor cannot be used, the allocation request is sent again after a fixed period of delay as if another processor were using the terminal. This is why allocation requests must be sent frequently when a terminal is in use, and why there is an increase in the data highway's transmission load or processor load. /3

One approach to solving this problem has been a method in which the terminal controller transmits all allocation data to other processors at the time terminals are allocated. It also informs all other processors of processor allocation. When the allocation is released, another transmission of all data notifies of that intent. In another method, messages are exchanged between processors and the system controlled so there is no simultaneous use. But both methods have the disadvantage of increasing the number of excess transmissions and complicating processor operations.

The present invention was developed in light of those disadvantages, it is a terminal controller which does not transmit unnecessarily nor does it have the above mentioned disadvantages in data I/O between terminal and processor.

In the following paragraphs, we will will explain this invention and in doing so, refer to the appended charts. Figure 1 is a diagram of the invention prototype. The processors are 1A, 1B and 1C. The transmission controllers are 2A, 2B and 2C. The terminal controller in the invention is 3. The terminal is 4. The terminal queue in storage within the terminal controller is 5 and 5 is controlled by the first-in-first-out method.

Figure 2 is a chart showing the flow of control in the invention's terminal controller. Figure 3 is a message format for the /4

terminal allocation request sent from the transmission controller to the terminal controller. CMD1 is the code indicating a terminal allocation request. SA is the address of the device to which the transmission is going. PA is the address of the transmitting device. N is the number of I/O data. Figure 4 is the format for the message when a reply on completion of terminal allocation is made from the terminal controller to the transmission controller. CMD2 is the code indicating a reply. SA is the device address of the transmission destination. PA is the address of the transmitting device. DK is a code indicating the completion of terminal allocation.

Figure 5 shows the format for a reply message when a terminal allocation is incomplete and another processor is using that terminal. CMD2 is the code indicating a reply. SAS is the device address of the transmission destination. PA is the address of the transmitting device. BSY is the code indicating that terminal allocation is incomplete. T is the estimated wait time for the processor.

When terminal 4 is in unused status and there is an I/O request from processor 1A, Figure 3 shows that transmission controller 2A edits terminal allocation request code CMD1, transmission destination device address SA, transmitting device address PA and the number of I/O data words N and sends the terminal allocation request message to terminal controller 3.

When terminal controller 3 receives the terminal allocation request message, it checks the terminal allocation queue within its own memory. Since the queue is empty, Figure 4 shows that reply code CMD1, the allocation request message PA, received at transmission destination device address, SA and allocation request message SA, received at transmitting device address PA, are edited and terminal allocation complete message SA is sent to transmission controller 2A. along with this, terminal controller 3 stores allocation request messages SA, PA and N into the terminal allocation queue in its memory. Transmission controller 2A receives the terminal allocation complete message from terminal controller 3 and by doing so starts

/5

I/O operations, but we will omit any explanation of subsequent steps, because the invention has nothing to do with that processing.

If processor 1B makes an I/O request while processor 1A is processing I/O, as Figure 3 shows, transmission controller 2B will load a terminal allocation request message into itself and transmit it to terminal controller 3 in the same way as described above. When terminal controller 3 receives this message, the controller determines whether the terminal is in use according to whether or not data is in the header of terminal allocation queue 5 in controller memory. If it determines there is data and the terminal is in use, Figure 5 shows that it edits 1) terminal allocation request message PA to transmission destination address SA and reply code CMD2, 2) terminal allocation request message SA to transmitting device address PA and 3) code BSY indicating terminal allocation complete. A calculation is then made of the number of I/O data words in data queued in the terminal allocation queue and, using the following formula, provides the operator T for estimated time, creates a terminal allocation complete message, and sends it to transmission controller 2B. /6

$$T = \alpha \times (\text{total number of I/O data words}) + \beta$$

Here,

α is the terminal's average processing time per word, and

β is the spare time.

When transmission controller 2B receives this message it sends the estimated time T to processor 2B and during this time places I/O requests into wait status. If processor 1C makes an I/O request in the meantime, the processing is the same as when process 1B makes such a request, the value of predicted wait time for a terminal allocation complete message the sum of the number of I/O process words in processors 1A and 1B. /7

Now, when I/O processing by terminal 4 is complete, terminal controller 3 updates the terminal allocation request queue and

retrieves the next terminal allocation request. Then, as Figure 4 shows, code CND2 indicating a response, transmission destination device address SA, transmitting device address PA and code DK indicating terminal allocation complete are edited and a terminal allocation complete message is sent to transmission controller 2.

When transmission controller 2 receives the message, it tells processor 1 that terminal 4 has been allocated, ends output request wait status, and makes another I/O request. The reader should note that since terminal 4 is already allocated, transmission controller 2 does not transmit a terminal allocation request message in response to the I/O request, and this, of course, starts data I/O. /8

It should also be noted that when terminal 4's real operations take longer than the estimated time, we can predict that a fault of one type or another will be generated between processor 1 and terminal 4 and that processor 1 will then begin fault diagnosis processing. However, we will omit any further explanation since this has no direct connection with the invention. The above explanation is one in which 3 processors share the use of one terminal, but the range of the terminal allocation request queue is such that there is no limit on the number of processors.

The invented data highway controller eliminates the inefficiencies caused by repeated processor use requests when multiple processors are sharing the use of a terminal.

Simplified Explanation of Charts

Figure 1 is a block diagram of an example prototype for this invention. Figure 2 is a flow chart showing the operations of the terminal controller. Figure 3 is the format of the terminal allocation request message. Figure 4 is the format of the terminal allocation complete message. Figure 5 is the format for the terminal allocation incomplete message. /9

1....processor

2....transmission controller

3....terminal controller

4....terminal

5....terminal allocation queue

Representing the Applicant

Patent Attorney Kensuke Sokkon (7317) and associate

Figure 1

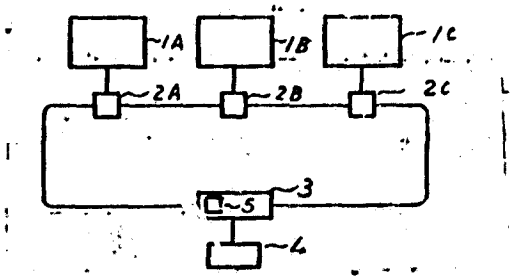
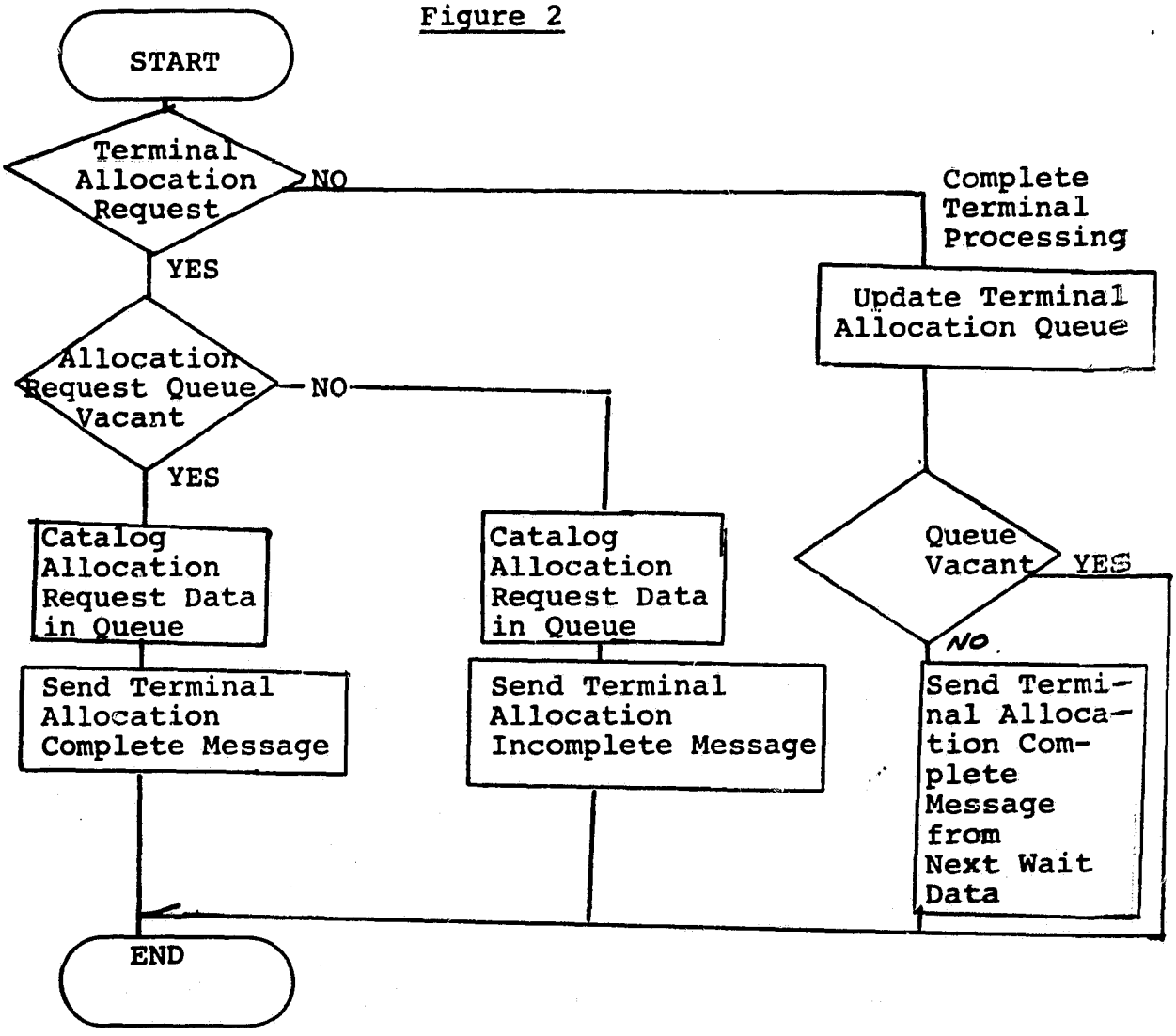


Figure 2



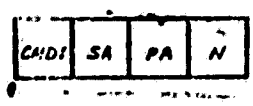


Figure 3

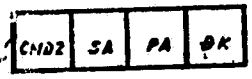


Figure 4

Figure 5