

Sept. 20, 1963

FROM: Daley, Helliwig, Hastings

SUBJ: Execution of a Chain of Commands

Five new CTSS supervisor subroutines permit execution of a chain of commands. These subroutines have been implemented and have been partially but not entirely checked out to date.

By execution of a program using these routines, up to five commands may be chained in a designated sequence. Along with the user's machine conditions are kept five buffers containing the command lists (command name plus arguments) of the commands in the chain, and a command location counter (CLC) to govern the chaining. The designated commands are numbered from 1 to $n \leq 5$, in the order of execution; the CLC contains the number of the next command to be executed and the number of the terminal command in the chain. The command lists may be set, read, and modified, as may the CLC.

The call

```
TSX      SETCLS, 4
PZE      An, , n
```

where

```
An BCI   1, NAME
BCI     1, ARG1
```

```
...
@CT     7777777777
```

Sets the n^{th} command list to the command NAME and the associated arguments. The word of sevens marks the end of the list.

The call

```
TSX      GETCLS, 4
PZE      BUFF, , n
```

fills the 20-word buffer BUFF with the contents of the n^{th} command list.

It must be noted that the Supervisor does not scan the command list before moving it in or out of the user's buffer. Twenty words are copied; the fence of sevens is interpreted

by the particular command program itself. Consequently the user must provide for GETCLS a 20-word buffer regardless of the length of the command list; care must be taken that for short lists SETCLS does not produce a protection violation.

To set the CLC, a call

TSX SETCLS,4

is given with the contents of the AC as follows:

PZE m, n

This will set the CLC so that the nth command will be executed next, and chaining will terminate after execution of the nth command. The call

TSX GETCLC,4

will return in the AC the current contents of the CLC. During execution of the ith command, the address of the CLC contains $i + 1$.

The call

TSX CHNCOM,4

causes execution of the next command according to the contents of the CLC. Every command program should issue a call to CHNCOM on normal exit, so that chaining may continue if the command is part of the chain. If there is no "next command" waiting, the call to CHNCOM will be equivalent to a call to DEAD or DORMNT, depending on the contents of the AC; if the address of the AC is zero, it is as if a call to DEAD had been given; if the address of the AC is one, a call to DORMNT is effected.

Requirements, etc.

1. All CTSS commands must be recoded to terminate in a call to CHNCOM.
2. It seems reasonable to change GETCLS and SETCLS so that they will scan the command list to look for the fence, transmitting twenty words only if it is not found. This will prevent spurious protection violations and eliminate a rather sloppy restriction on the user.

3. With respect to the function of QUIT during a command chain the following should obtain:

- a.) a QUIT on level zero should have normal effect, i.e., the user is put in dormant status;
- b.) upon issue from the console of the start command from dormant status, normal restart will be effected, and there will be no disruption of the command chain;
- c.) any command other than start issued from the console will cause the CLC to be reset to zero, i.e. the command chain will be discontinued.

CAS