

# Operator Commands

This chapter covers the following topics:

- Using Operator Commands
  - Command Description
- 

## Using Operator Commands

With the exception of the EOJ command, each command may be entered in full or may be abbreviated. The minimum abbreviation required is the number of characters necessary to uniquely identify the command. The characters needed to identify the command are underlined in the description of each command in the sections that follow.

### Note:

The MODIFY (F) command and proper ID (MVS), or the REPLID (VSE) are assumed, and are not shown in command syntax in this chapter.

## Specifying Terminals

The Terminal Information Block Table (TIBTAB) defines the terminals to be used by Com-plete. The TIBTAB is loaded or generated by the Com-plete control program during Com-plete initialization.

For each local and remote terminal, the TIBTAB contains a Terminal Information Block (TIB). The TIB contains identifying information such as a one- to five-digit Terminal Identification number (TID) and a one- to eight-character Terminal Information Block Name (TIBNAME). Either the TID or TIBNAME may be used to specify a single terminal in most computer operator commands.

## Specifying User IDs

When a command argument specifies a user ID, the argument must be in the form:

```
USERID=userid
```

where *userid* is one to eight alphanumeric characters.

Note that the command will only be accepted if the user ID is currently logged onto Com-plete.

## Specifying Groups of Terminals or User IDs

In Com-plete, groups of terminals or user IDs can be defined. A group may contain any combination of user IDs and/or terminals. For details, refer to the description of function TU of the UUTIL utility in the Com-plete Utilities documentation.

To specify a group in an operator command, enter the groupname with the command.

If a command specifies a group containing a user ID that is not logged on, Com-plete will still carry out the command for all other members of the group.

## Specifying All Terminals

In general, to specify all terminals in the Com-plete system, the argument ALL is entered with the command. Note, however, that a TID may be defined in the TIBTAB as belonging to the group named ALL=NO. This indicates that this TID will not be considered part of the ALL grouping.

## Command Description

### ACCESS

The ACCESS command is used to start up or shut down Com-plete access processing.

#### Command Syntax

```
ACCESS START
ACCESS START, FORCE
ACCESS STOP
```

#### Description

The status of the Com-plete/ACCESS interface is established during Com-plete initialization. The ACCESS operator command allows the operator to control ACCESS processing after initialization. The FORCE option can be used on the START command to force the allocation of the entry in the ADABAS SVC ID table.

The START command loads, if necessary, the ADABAS interface modules and initializes the ACCESS node. A subsequent STOP command causes any further access requests to the node to be rejected by the ADABAS SVC with response 148. Only the STOP command issued during EOJ processing causes the node to be deactivated and the SVC communication to be terminated.

---

### ALLOW

The ALLOW command enables the computer operator to enable one of three Com-plete facilities:

#### Syntax

```
ALLOW, RJE
ALLOW, LOGON
ALLOW, LOAD
```

#### Description

Note that the ALLOW command is normally used in conjunction with the DISALLO command after the desired facility has been disabled and it is determined that the facility should once again be enabled.

#### Examples

```
ALLOW, RJE
ALLO, LOGON
AL, LOAD
```

---

## CANCEL

The CANCEL command is used to cause an abnormal termination of an online application program that is executing from a specific local or remote terminal (TID or TIBNAME), or executing under a specific user ID.

### Syntax

```
CANCEL, tid  
CANCEL, tibname  
CANCEL, userid
```

### Description

The CANCEL command causes termination of the application program in conversation with the specified terminal. An application program that has been rolled out of the thread while waiting for terminal I/O will not be cancelled, however, until the terminal operator presses the ENTER key. When the program is cancelled, the following message is displayed at the terminal:

---

**ABS0003 PROGRAM XXXXXXXX CANCELLED BY COMPUTER OPERATOR**

---

If a Com-plete service routine or a PV program is in control at the time the CANCEL command is entered, the CANCEL request is delayed until the routine or program is completed. The operator is notified by the following message:

---

**ZOC00013 CANCEL DELAYED; IN Com-plete OR PV-PROGRAM**

---

Entry of a second CANCEL command causes the termination of the routine or program, but may result in the loss of buffers and/or other resources.

### Examples

```
CAN, 33  
CANCEL, TIB00005  
CA, U=FRED
```

---

## CLOSE

The CLOSE command causes the current Com-plete capture data set to be closed and the next data set to be opened in order to continue uninterrupted service to CAPTUR requests by application programs.

### Syntax

```
CLOSE
```

## Description

The CLOSE command is normally used to make the current capture data set available for access by batch jobs.

## Examples

CL

---

## COMSEC

The COMSEC command allows the operator to control Com-plete SECURITY processing after initialization.

## Syntax

COMSEC, *parameters*

## Description

The status of the Com-plete SECURITY system is established during Com-plete initialization. Please refer to the Com-plete SECURITY Reference documentation for a full description of the available operator commands.

---

## DCTRL

The DCTRL command enables the computer operator to display the terminals in the Com-plete network that have control status. Terminals may be assigned control status by use of the SETCTL command and may have control status removed by means of the UNCTL command.

## Syntax

DCTRL

## Description

A terminal with control status is considered to be a privileged terminal and is allowed to perform functions not normally permitted. The DCTRL command enables the computer operator to periodically monitor such terminal usage and take corrective action if needed.

## Examples

DC

---

## DISALLO

The DISALLO command enables the computer operator to disallow use of the following facilities by terminal users:

## Syntax

DISALLO,RJE  
DISALLO,LOGON

## Description

The DISALLO command is normally used to temporarily disable a utility function during system maintenance. The ALLOW command can be used to reenable a function.

### MVS Only:

The DISALLO command can also be used to disallow loading of load modules from the COMPLIB load library chain. This may be useful in two cases:

1. If, since Com-plete startup, new extents of secondary space have been allocated to any PDS concatenated to COMPLIB, members stored into these extents cannot be loaded by Com-plete unless the COMPLIB DCB is closed and re-opened. You can do this by specifying:

**DISALLO ,LOAD ,REOPEN**

After successful execution of this command, LOADs are allowed again without the need of issuing an ALLOW,LOAD command.

2. A PDS concatenated to COMPLIB may become full and require compression. You can temporarily disallow LOADs from COMPLIB and free the ENQ held for one of the data sets concatenated to COMPLIB by specifying:

**DISALLO ,LOAD ,*dsn***

If an ENQ exists for the data set specified, the COMPLIB DCB will be closed, and the ENQ for the data set will be freed. You can now compress the data set and, after successful compression, issue an ALLOW,LOAD command to reestablish the ENQ, re-open the COMPLIB DCB and re-allow LOADs. Note that the data set is not deallocated from Com-plete by the DISALLO,LOAD command and, therefore, need not be relocated.

### Note:

While LOADs are disallowed, only requests to load resident programs or programs residing in the Com-plete lookaside buffer are satisfied. All requests to load a module from COMPLIB while LOADs are disallowed will be queued and the appropriate subtask(s) will be set into WAIT state until you issue the ALLOW,LOAD operator command. Other operator commands causing modules to be loaded, such as PGM, SERV, etc., may cause unpredictable results when issued in the meantime.

## Examples

DISALLO ,RJE  
DISA ,LOGON

---

## DTASKS

The DTASKS command enables the computer operator to display the status of all Com-plete tasks.

## Syntax

***DTASKS***

## Description

For a description of the output, please refer to the commandPLIST.

## Examples

**DT**

---

## DUMP (VSE Only)

Switch the device to which Com-plete abend dumps are written.

### Syntax

**DUMP,DISK**           (VSE) write dump to VSAM dataset COMDMP  
**DUMP,NODISK**       (VSE) write dump to SYSLST

### Description

The DUMP command enables the computer operator to switch the device to which Com-plete abend dumps are written from SYSLST to COMDMP and vice versa.

---

## ENQ

The ENQ command enables the computer operator to display outstanding ENQUEUEES from the Com-plete threads.

### Syntax

**ENQ,LIST, *major,minor***  
**ENQ,TERM, *tibnr***  
**ENQ,USER, *userid***  
**ENQ,THRD, *taskid***  
**ENQ,DEQ, *qname,rname***

### Description

The ENQ command can be used to:

- Display all ENQUEUEES by specifying: **LIST, *major,minor***;  
**ENQ,LIST,SYSDSN**
- Display all ENQUEUEES for a specific terminal by specifying: **TERM, *tibnr***;  
**ENQ,TERM,5**
- Display all ENQUEUEES for a specific user by specifying: **USER, *userid***;

```
ENQ,USER,FRED
```

- Display all ENQUEUEES for a specific thread by specifying: THRD,*taskid*.

```
ENQ,THRD,THREAD1
```

- Dequeue a specific ENQUEUEE by specifying: DEQ,*qname,rname*;

```
ENQ,DEQ,SYSDSN,COM.V460.DATA
```

---

## EOJ

The EOJ command causes a logical shutdown of the Com-plete system.

### Syntax

```
EOJ
EOJ,VER=password
```

### Description

If the system programmer specified an EOJ verification password using the EOJ,VER system parameter, that password must also be entered with the EOJ command.

Note that a logical shutdown of Com-plete can also be performed with the MVS STOP (P) command.

If Com-plete does not come down after you have entered EOJ, try:

```
EOJ,FORCE
```

### Examples

```
EOJ
EOJ,VER=STOPCOMP
```

---

## FORCE

The FORCE command enables the computer operator to force the logoff of one or more Com-plete users. Before the FORCE command can be issued, a LOGOFF command must have been issued for the specified terminals or users.

### Syntax

```
FORCE,tid
FORCE,tibname
FORCE,userid
FORCE,groupname
FORCE,ALL
```

### Description

Before the FORCE command can be issued, a LOGOFF command must have been issued for the specified terminals or users.

## Examples

```
FORCE,33
FO,TIB00033
FORCE,U=CUST23
F,ACCTNG
FOR,ALL
```

---

## HMSG

The HMSG command causes Com-plete to reissue the "hello message" that is issued to all or selected terminals after Com-plete initialization.

### Syntax

```
HMSG,tid
HMSG,tibname
HMSG,userid
HMSG,groupname
HMSG,ALL
```

### Description

The HMSG command is useful to notify terminal users in the network of an end to a temporary interruption of service caused by execution of the IGNORE ALL or IGNORE GROUP commands. If GROUP is not specified, the hello message is sent to all terminals. Note that the message generated by the HMSG command will not interrupt a terminal in conversation.

The HMSG command can be used to issue the hello message to:

- A single local terminal - by specifying either the Terminal ID (TID) or the TIBNAME;

```
H,33
H,TIB00005
```

- A single user - by specifying the user ID;

```
HM,U=FRED
```

- A group of terminals or user IDs - by specifying the groupname;

```
HMS,DATAENTY
```

- All terminals - by specifying ALL (Note that any terminals assigned to the group ALL=NO in the TIBTAB will *not* be issued the hello message by the HMSG command).

```
H,ALL
```

---

## IGNORE

The IGNORE command is used to immediately terminate terminal I/O to one or more terminals or lines in the Com-plete system.

## Syntax

```
IGNORE,tid
IGNORE,tibname
IGNORE,userid
IGNORE,groupname
IGNORE,ALL
```

## Description

The IGNORE command can be used to terminate I/O to:

- A single local terminal - by specifying either the Terminal ID (TID) or the TIBNAME;

```
IGNORE,33
IG,TIB00005
```

- A single user - by specifying the user ID;

```
IG,U=FRED
```

- A group of terminals or user IDs - by specifying the groupname;

```
IGNORE,DATAENTRY
```

- All terminals - by specifying ALL (Note that any terminals assigned to the group ALL=NO in the TIBTAB will *not* be ignored by an IGNORE ALL command).

```
I,ALL
```

Processing VTAM terminals for which the IGNORE command has been issued depend on the options specified in the TIB definition (ACQUIRE/NOACQUIRE, SHARE/NOSHARE).

---

## LOGOFF

The LOGOFF command enables the computer operator to logoff one or more Complete users.

### Syntax

```
LOGOFF,tid
LOGOFF,tibname
LOGOFF,userid
LOGOFF,groupname
LOGOFF,ALL
```

### Description

The LOGOFF command can be used to logoff:

- A single local terminal - by specifying either the Terminal ID (TID) or the TIBNAME;

```
LOGOFF,33
LO,TIB00033
```

- A single user - by specifying the user ID;

```
LOGOFF,U=CUST23
```

- A group of terminals and/or user IDs - by specifying the groupname;

```
L,ACCTNG
```

- All terminals - by specifying ALL (Note that any terminals assigned to the group ALL=NO in the TIBTAB will *not* be logged off by the LOGOFF ALL command).

```
LOG,ALL
```

---

## MSG

The MSG command is used to send a message to one or more terminals or users in the Com-plete system:

### Syntax

```
MSG,tid
MSG,tibname
MSG,userid
MSG,groupname
MSG,ALL
```

### Description

The message class used is 1 by default, but you can override this default with an option as follows:

```
MSG,destination,ENTER,message-text      = Class 5
MSG,destination,IMMED,message-text      = Class 2
MSG,destination,message-text            = Class 1
```

### Examples

```
MSG,ALL,ENTER,Com-plete Yo-Yo in 10 minutes
MSG,ALL,IMMED,Urgent Com-plete Yo-Yo
```

### Notes:

1. TIB 1 in the TIBTAB must have authority to send class 1,2 and/or 5 messages. Otherwise a security violation will occur.
  2. If APPLYMOD 41 is not set, class 5 messages will be treated as class 1 messages.
- 

## PGM

The PGM command is used to load, delete, and refresh modules that reside in the RESIDENTPAGE area of Com-plete.

## Syntax

```
PGM,DELETE,program-name
PGM,LOAD,program-name
PGM,REFRESH,program-name
```

## Description

Note that the amount of storage GETMAINED from Com-plete's region is the module size rounded up to the next 4K boundary. The REFRESH command implies a LOAD of the new copy followed by a DELETE of the old one.

### Note:

Com-plete does not check to determine if there are any active transactions using a program being either deleted or refreshed. If there are active transactions, the effect upon those transactions will be unpredictable.

## Examples

```
PGM,DELETE,PGM1
PG,LOAD,PGM1
P,REFRESH,PGM1
```

---

## PLIST

This command will provide a list of the current tasks defined in the requested task group and the status of each.

## Syntax

```
PLIST
```

## Description

If no task group is supplied as a parameter to this command, all tasks of all task groups will be displayed. This command provides similar output to the UCTRL PL function. The following is a sample output resulting from this command.

```
09.33.15 STC06160 COMOPC0099-T COMMAND RECEIVED AT 9:33:15 FROM CONSOLE - 00 WAS PLIST
09.33.16 STC06160 COMOPC0067-T -> GrpName Status Use Wait LastOp Time Program Tid.. L
09.33.16 STC06160 COMOPC0067-T -> OC      A-Run
09.33.16 STC06160 COMOPC0067-T -> TAM      A-Wait          USTACK      4    0
09.33.16 STC06160 COMOPC0067-T -> MSGPO    A-Wait
09.33.16 STC06160 COMOPC0067-T -> PAGING   A-Wait
09.33.16 STC06160 COMOPC0067-T -> FIO      A-Wait
09.33.17 STC06160 COMOPC0067-T -> DEFAULT A-Wait 0 2  Wrtm      USTACK      4    0
09.33.17 STC06160 COMOPC0067-T -> DEFAULT A-Wait 0 2  Coexit    UPDS        4    4
09.33.17 STC06160 COMOPC0067-T -> DEFAULT A-Wait 0 2
09.33.17 STC06160 COMOPC0001-T PList command COMPLETED.
```

Where:

**GrpName**

This is the name of the task group of which the task in question is a member. In the case of system tasks, this is the name of the system task.

### Status

This reflects the current status of the task. The status is a combination of two state indicators separated by a dash ('-'). The primary state indicator is the letter preceding the dash indicates whether the task is Active, Quiescing or Dormant by the letters A, Q and D respectively. Active in this sense indicates that the task is available to do work. When it is quiescing, it will remain active long enough to finish any work which has been started by the task while dormant tasks cannot be used and will have no secondary state associated with them. The secondary states which may occur are as follows:

Status	Meaning
Wait	This indicates that the task is waiting. In this state, the task is waiting on new work or on events requested by programs running in threads associated with it.
Run	This indicates that the task is currently running a user program.
Disp	This indicates that the task is going through it's dispatching cycle either finishing off old work or looking for new work.

### Use

This is the current use count for the task. The use count includes the current user of the task, any users for whom a wait was issued on the task and any users with an affinity for this task.

### Wait

This is the current wait count for the task. This reflects the number events upon which the task is waiting and includes two standard events those being that work has been queued to the task group work queues or to the task's own work queue.

### LastOp

This is the last Com-plete op which was issued under control of the task.

### Time

When the task has a secondary status of 'Run', this will reflect the time in seconds that this user has spent under control of the task.

### Program

This is the name of the program currently active under control of the task, or the last program to be active under control of the task if it has a secondary status of 'wait'. If the task has never been used, this will be blank, however, once it has been used, this will always contain a value.

**Tid..**

This is the tid of the current TIB active under control of the task, or the last TIB to be active under control of the task if it has a secondary status of 'wait'. If the task has never been used, this will be blank, however, once it has been used, this will always contain a value.

**L**

This is the level number on which the user currently active under control of the task, or the last user to be active under control of the task if it has a secondary status of 'wait', is running. If the task has never been used, this will be blank, however, once it has been used, this will always contain a value. Level '0' will be displayed for a non-Com-pass user, for a Com-pass user who is running a program while something is stacked on all possible levels or for specific programs which must run on level 0 such as ULOG.

**QOVER**

This command will provide a list of the current queues defined in the system and the status of each.

**Syntax**

QOVER

**Description**

This command provides similar output to the UCTRL QO function. The following is a sample output resulting from this command.

```
09.35.17 STC06160 COMOPC0099-T COMMAND RECEIVED AT 9:36:17 FROM CONSOLE 00 WAS QOVER
09.35.18 STC06160 COMOPC0067-T -> Queue name      Fifo Max  Lifo  High  Avg Time
09.35.18 STC06160 COMOPC0067-T -> Stage Queue    0   16   0    0    0.000
09.35.18 STC06160 COMOPC0067-T -> Output Queue   0   16   0    1    0.019
09.35.18 STC06160 COMOPC0067-T -> Input Queue    0    8   0    0    0.000
09.35.18 STC06160 COMOPC0067-T -> Compl Queue    0   16   0    1    0.017
09.35.18 STC06160 COMOPC0067-T -> Message Queue  0   16   0    0    0.000
09.35.18 STC06160 COMOPC0067-T -> Paging Queue   0    8   0    0    0.000
09.35.18 STC06160 COMOPC0067-T -> DEFAULT WORK-Q 0   16   0    1    0.028
09.35.19 STC06160 COMOPC0067-T -> DEFAULT TASK-Q 0    8   0    0    0.00
09.35.19 STC06160 COMOPC0067-T -> DEFAULT TASK-Q 0    8   0    1    0.006
09.35.19 STC06160 COMOPC0067-T -> DEFAULT TASK-Q 0    8   0    0    0.000
09.35.19 STC06160 COMOPC0001-T QOVER command COMPLETED.
```

Where:

**Queue name**

This is the 16 character name of the queue for which details are provided. The first six queues are the standard system queues used by Com-plete. Following this, each of the task group's queues will follow. The first eight characters of these queue names will contain the task group name while the second eight characters indicates their purpose within the task group.

Each task group will have four queues with names Prty-0 to Prty-3. These queues represent the work queues upon which work is queued depending on the priority of the work. In addition, each task within the group will have its own work queue with the identifier 'TASK-Q' associated with it. These queues are used for work which has an affinity to a particular task.

**Fifo**

This is the number of TIBs currently on the First In First Out (FIFO) queue.

**Max**

This is the maximum number of TIBS that can be handled at any one time on the First In First Out (FIFO) queue.

**Lifo**

This is the number of TIBS currently on the Last In First Out (LIFO) queue.

**High**

This is the highest number of TIBS on both FIFO and LIFO queues at any one time since the queue was created. If this is higher than the FIFO Hig count, it indicates that at some point in the run, there was not enough space on the FIFO queue and so the LIFO queue had to be used.

**Avg Time**

This is average amount of time each TIB spent on the queue in thousandths of seconds.

---

## QUIESCE

This command causes the system to quiesce.

**Syntax**

QUIESCE

**Description**

This command will result in the following:

The VTAM application will be quiesced which will prevent any further log-ons to this Com-plete's APPL, however, current sessions will be allowed to continue to completion. If Com-plete runs as a cluster in a Parallel Sysplex, requests to establish new sessions with Com-plete's generic VTAM resource name will automatically be routed to one of the remaining parts of the cluster.

Access will be quiesced which means that any existing sessions will be allowed to continue but any new sessions will received a 'node not active' return code.

Users returning to their COM-PASS menu will be given a warning message to the effect that the system is quiescing and that they should finish their work and log off.

Once the quiesce command has been issued, Com-plete must be stopped and restarted before normal service to users can be resumed.

---

## RESTART

The RESTART command causes an immediate restart of terminal I/O to one or more terminals or lines in the Com-plete system.

### Syntax

```
RESTART, tid
RESTART, tibname
RESTART, groupname
RESTART, ALL
```

### Description

The RESTART command can be used to restart I/O to:

- A single local terminal - by specifying the Terminal ID (TID) or TIBNAME;

```
RESTART, 33
RES, TIB00033
```

- A group of terminals - by specifying the groupname;

```
RESTART, DATAENTY
```

- All terminals - by specifying ALL (Note that any terminals assigned to the group ALL=NO in the TIBTAB will *not* be restarted with a RESTART ALL command).

```
R, ALL
```

---

## SERV

The SERV command enables the computer operator to pass commands to a server.

### Syntax

```
SERV, [keyword], server-id, [server-parameters]
```

### Description

Servers can be started and terminated using this command, and requests can be sent to servers. These servers must be specified in the Com-plete sysparm SERVER. For more information of Com-plete Servers, see **Com-plete Servers** in the Com-plete System Programming documentation.

### Examples

1.

```
SERV TERM, server-id
```

This command terminates the specified server.

2.

```
SERV INIT,server-id
```

This command initializes the specified server.

3.

```
SERV server-id,server-dependent parameters
```

This command sends a request to the specified server.

The SERV operator command syntax deviates slightly from the above examples when sending commands to the JES2 or JES3 servers. See the description on the JES subsystem in the Complete System Programming documentation for more information.

## SETCTL

Set one or more local or remote terminals to control status

### Syntax

```
SETCTL,tid  
SETCTL,tibname  
SETCTL,userid  
SETCTL,groupname  
SETCTL,ALL
```

### Description

The SETCTL command enables the computer operator to set one or more local or remote terminals to control status. A terminal with control status is considered a privileged terminal and is allowed to perform functions normally not permitted.

In addition, the SETCTL command enables the computer operator to assign control status dynamically to one or more terminals, when necessary. Note that control status assigned using the SETCTL command is in effect for the duration of the current Com-plete session only, and is removed when the user logs off or when the UNCTL command is used.

The SETCTL command can be used to assign control status to:

- A single local terminal - by specifying the Terminal ID (TID) or the TIBNAME;

```
SETCTL,33  
SE,TIB00033
```

- A single user - by specifying the user ID;

```
SETCTL,U=USER33
```

- A group of terminals and/or users - by specifying the groupname;

**SET,ACCTNG**

- All terminals - by specifying ALL (Note that any terminals assigned to the group ALL=NO will *not* be given control status with a SETCTL ALL command).

**SE,ALL**

**Note:**

If a user is logged onto a terminal to which control status is assigned with the SETCTL command, control status will be lost when the user logs off from that terminal.

---

## STATS

The STATS command causes the Com-plete EOJ statistics to be written to the data set specified by the SYSPRINT DDNAME in the Com-plete startup procedure.

### Syntax

STATS

### Description

---

## TASKS

The TASKS command can be used to attach additional operating system tasks in a Com-plete task group or to detach some of the tasks of a task group.

### Syntax

**TASKS, *task\_group\_name*, *number\_of\_tasks***

### Description

The *number\_of\_tasks* parameter you specify is the number of tasks you wish to exist in the task group after execution of the command, *not* the number of tasks you want to be attached additionally or detached!

*number\_of\_tasks* must be greater than zero and the resulting number off all processor tasks in Com-plete must not exceed the number specified or defaulted by the MAXTASKS sysparm.

### Examples

**TA,DEFAULT,10**

---

## TLIST

This command will provide a list of the current threads defined in the requested thread group and the status of each.

## Syntax

`TLIST, thread_group_name`

## Description

If no thread group name is provided as a parameter to the request, all threads of all thread groups will be displayed. This command produces a similar output to the UCTRL TL function.

### Subgrp

This is the name of the thread subgroup of which the thread in question is a member.

### Status

This reflects the current status of the thread. The status is a combination of two state indicators separated by a dash ('-'). The primary state indicator is the letter preceding the dash indicates whether the thread is Active, Quiescing or Dormant by the letters A, Q and D respectively. Active in this sense indicates that the thread is available to do work. When it is quiescing, it will remain active long enough to finish any work which has been started in the thread, while a dormant thread cannot be used and will have no secondary state associated with it. The secondary states which may occur are as follows:

#### 'Free' Status

This indicates that the thread is free to run other work. If there was a previous user of the thread, this state indicates that this user's program ended or has been rolled out.

#### 'Occ' Status

The 'occupied' status indicates that the thread is available to do work, however, the user program currently occupying the thread must first be rolled out prior to starting any other new work in the thread.

#### 'Disp' Status

This indicates that the thread is reserved and the dispatcher is currently in the process of either starting a new user program or rolling in a user program which was previously rolled out.

#### 'Run' Status

This indicates that the user program in the thread is currently running.

#### 'Susp' Status

This indicates that the user program has been temporarily suspended as a wait was issued either directly by the user program or indirectly by a function used by the program. In this state, the user program may not be rolled out. Internally it indicates that the operating system task associated with the work is active elsewhere. Once the condition for the wait is satisfied, the task will continue processing this work.

**Use**

This is the current use count for the thread. The use count includes the current user of the thread plus any other non relocatable users previously rolled out from this thread.

**Wait**

This is the current wait count for the thread. This reflects the number of users waiting to run in the thread at the present time.

**LastOp**

This is the last Com-plete op which was issued in the thread.

**Time**

When the thread has a secondary status of 'Susp' or 'Run', this will reflect the time in seconds that this user has spent in the thread.

**Program**

This is the name of the program currently active in the thread, or the last program to be active in the thread if the thread has a status of 'free' or 'occ'. If the thread has never been used, this will be blank, however, once the thread has been used, this will always contain a value.

**Tid..**

This is the tid of the current TIB active in the thread, or the last TIB to be active in the thread if the thread has a status of 'free' or 'occ'. If the thread has never been used, this will be blank, however, once the thread has been used, this will always contain a value.

**Active L**

This is the level number on which the user currently active in the thread, or the last user to be active in the thread if the thread has a status of 'free' or 'occ', is running. If the thread has never been used, this will be blank, however, once the thread has been used, this will always contain a value. Level '0' will be displayed for a non-Com-pass user, for a Com-pass user who is running a program while something is stacked on all possible levels or for specific programs which must run on level 0 such as ULOG.

**UNCTL**

The UNCTL command enables the computer operator to remove control status from one or more local or remote terminals or users.

**Syntax**

```
UNCTL,tid
UNCTL,tibname
UNCTL,userid
UNCTL,groupname
UNCTL,ALL
```

## Description

A terminal with control status is considered to be a privileged terminal and is allowed to perform functions not normally permitted. The UNCTL command enables the computer operator to remove control status dynamically from a designated terminal whenever necessary.

The UNCTL command can be used to remove control status from:

- A single local terminal - by specifying the Terminal ID (TID) or the TIBNAME;

```
UNCTL,33
UNC,TIB00033
```

- A single user - by specifying the user ID;

```
UNCTL,U=FRED
```

- A group of terminals and/or users - by specifying the groupname;

```
UNC,ACCTNG
```

- All terminals - by specifying ALL (Note that any terminals assigned to the group ALL=NO in the TIBTAB will *not* have control status removed by the UNCTL ALL command).

```
UNC,ALL
```

---

## USER

The USER command allows the computer operator to attach a specific user program and optionally pass data to that program.

### Syntax

```
USER
```

### Description

Programs that are started using the USER command receive passed data in the same format as if they were started by the ATTCH function in an online program.

A special parameter, UFILE, allows the computer operator to change the status of up to five files in a single input operation.

### Examples

```
US,UFILE, status (file1,file2,...file5)
```

where:

*status* is the status to which the files are to be set. Possible options: BTCH, ONLN

*file1...* are the DD/DLBL names to be changed. Up to five can be specified, separated by commas and enclosed in parentheses. If only one file is specified, the parentheses are optional.

---

## UU

The UU command is used to display status and account information about one or more logged on users.

### Syntax

```
UU, tid
UU, tibname
UU, userid
UU, groupname
UU, ALL      UU, ALL
```

### Description

The UU command can be used to display status and account information about :

- A single local terminal - by specifying the Terminal ID (TID) or the TIBNAME;

```
UU, 33
UU, TIB00005
```

- A single user - by specifying the user ID;

```
UU, U=FRED
```

- A group of terminals and/or users - by specifying the groupname;

```
UU, DATAENTRY
```

- All terminals - by specifying ALL

```
UU, ALL
```

---

## VTAM

The VTAM command can be used to documentionly perform certain functions upon Com-plete's VTAM interface.

### Syntax

```
VTAM, parameter
```

### Description

Usually, VTAM communication is started automatically during Com-plete startup and stopped during shutdown. This includes connection to a generic resource name when requested by sysparm VTAMGENERIC and disconnection from it.

The following parameters are available with the VTAM command:

START	Start VTAM communication
STOP	stop VTAM communication
QUIESCE	stop accepting new sessions, but continue serving existing ones. This function is also invoked implicitly by the QUIESCE operator command.
DISCONNECT	disconnect from the generic resource name if any. No new sessions will be accepted through the generic resource name. Existing sessions will continue being served. Log-on is still possible through the unique VTAM APPL-ID. This function can be used to prepare a planned downtime for one Com-plete of a cluster in a Parallel Sysplex.

## Examples

V, START  
VT, STOP

---