

**break**

nsoggia@telnetwork.it

**COLLABORATORS**

	<i>TITLE :</i> break		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	nsoggia@telnetwork.it	February 12, 2023	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

---

# Contents

<b>1</b>	<b>break</b>	<b>1</b>
1.1	Break 37.01 user manual . . . . .	1
1.2	Foreword on Break 37.01 . . . . .	1
1.3	Usage of Break 37.01 . . . . .	2
1.4	Tutorial on Break 3701 . . . . .	2
1.5	Localizing Break 3701 . . . . .	3

---



When a CLI number does not exist, the original message is Process X does not exist, the new message is CLI "X": object not found (the "object not found" sentence is localized). This may affect scripts or programs that parse the stdout message.

When all objects cannot be found, a fail (20) return code is set as the original program. When not all objects are found but at least one is found, a return code of 15 is set.

## INSTALLATION

Before installing the new program you may want to check if it is already installed. Clicking on [1] you may see the command version string: break 37.1 (localized 1991 date here) is the original Commodore program, otherwise you will also read my name after the date. The version installed by this guide is 37.01.

To keep a backup of the original command, you may click on [2]. If you are upgrading from a previous version of my program and you already renamed the original program, this Rename command will simply fail.

To install the new command you have to click on [3]. If you didn't rename the original program, the new program will overwrite the original one.

Version C:Break FULL [1] Show the version of the installed program Rename C:Break C:\_Break [2] Keep a backup of the original program Copy Break C: CLONE [3] Install the new program (Break 37.01)

## 1.3 Usage of Break 37.01

### REQUIREMENTS

Break is a CLI/shell only program, this means it is not possible to execute it directly from Workbench. The program is pure and can be made resident, it needs Kickstart 2.0 (or later) and runs on any Amiga with a 3 Kb (or greater) stack.

### ARGUMENTS

OBJECT DEFINITIONS CMD=COMMAND/S TASK/S PORT/S SEARCH PREFERENCES TFB=TASKFALLBACK/S NS=NOSTRIP/S  
CASE/S OBJECTS LIST CLI=PROCESS/A/M SIGNAL MASK ALL/S KILL=C/S SKIP=D/S REMCX=E/S SYNC=F/S

Object definitions tell Break what kind of names are listed in the objects list. Only one of them can be set. Default value is CLI=PROCESS, meaning that objects list contains CLI task numbers. CMD or COMMAND objects list contains command names (names of commands run from a shell) TASK objects list contains task names (names of tasks or names of processes run from Workbench) PORT objects list contains public message port names (names of public message ports owned by a task or process) CLI or PROCESS objects list contains CLI task numbers (number of CLI tasks as returned by the Status command. this is the default value, keyword can be omitted)

Search preferences tell Break how to search names in system lists. It is possible to combine them. Default value is that none of them is set, meaning that all string searches are case-independent and that when searching for processes, Break will first scan the names list for the command full name, if a name is not found, Break will rescan the list using the command file name only. TFB or TASKFALLBACK when a command name is not found, Break may look in the task names list hoping the process was run from Workbench. NS or NOSTRIP when a command name is not found, Break must not do a second pass on the commands list looking for names with the path part stripped. CASE use a case-sensitive search routine.

Objects list is the only required argument. It contains the name (or the names) of the objects to look for. It has no default value, meaning that at least one argument must be given. When looking for a CLI task (decimal) number, all characters except 0 1 2 3 4 5 6 7 8 9 are ignored: this is useful in guessing what process to stop given the Run command output.

Signal mask tells Break what signals send to the target task or process. It is possible to combine them. Default value is "kill". ALL send all the following four signals KILL or C equivalent to a keyboard ctrl+c. generally used to end a program SKIP or D equivalent to a keyboard ctrl+d. generally used to end a script or to end the current action of a program REMCX or E equivalent to a keyboard ctrl+e. generally used to end a commodity SYNC or F equivalent to a keyboard ctrl+f. generally used to "wake up" a program

## 1.4 Tutorial on Break 37.01

Here are some examples on how to use this command and how to solve the most common problems. Let's assume that task "nothing" does not exist and let's suppose that the current status is:

Process 1: Loaded as command: foo Process 2: Loaded as command: Foo Process 3: Loaded as command: c:foobar Process 4: Loaded as command: foobar Process 5: Loaded as command: Workbench Process 6: Loaded as command: twin Process 7: Loaded as command: twin Process 8: No command loaded

Q: I have to signal foo A: Break CMD foo CASE When Break finds two or more equal names it can't guess what command you want to stop and does nothing else showing an error message. By default the search is case insensitive. Here foo and Foo differ in the first character when we make a case-sensitive search.

Q: I have to signal both foo and Foo A: Break CMD foo Foo CASE Break accepts multiple target names.

Q: I have to signal twin A: Break 6 ; signal the first A: Break 7 ; signal the last A: Break 6 7 ; signal both When Break finds two or more equal names it can't guess what command you want to stop and does nothing else showing an error message. Here commands 6 and 7 have the same name so we can only use the CLI task number.

Q: I have to signal c:foobar A: Break CMD c:foobar Using "Break CMD foobar" will Break process 4 instead of 5. Infact Break first scans the names list for exact match, and only when a match is not found it rescans the list stripping the path part in the names.

Q: I run Exchange from Workbench, I want to pop it up A: Break TASK exchange SYNC All processes run from Workbench have their name (the icon name) in the task structure because they are not associated to a shell. They are processes, but they must be considered tasks to be found...

Q: I wish to wakeup SMTPd by Michael B. Smith A: break PORT smtpd-wakeup SYNC SMTPd has a public message port, so we can find the program even if we don't know the name or the number of the process. Note that SMTPd gets a ctrl-f, and not the signal bit set in the port!

Q: How can I later break a program I now have to run? A: Setenv breakme `Run foobar` Now in the global \$breakme variable you have the unique CLI number of your program, so Break "\$breakme" will be able to stop it. The double quotes are needed because the variable contains a space ("[CLI 9]").

Q: What happens if I write 'Break CMD nothing TASKFALLBACK' A: You get a 'not found' error message Break first scans the command names list for an exact match, then rescans the same list for a name-only match, then scans the task names list and then it stops.

Q: What happens if I write 'Break CMD nothing TASKFALLBACK NOSTRIP' A: You get a 'not found' error message Break first scans the command names list for an exact match, then scans the task names list, and then it stops.

Q: What happens if I write 'Break PORT nothing NOSTRIP TASKFALLBACK' A: You get a 'not found' error message Break scans the public port names list and then it stops.

Q: What happens if I write 'Break nothing' A: You get a 'bad number' error message "Nothing" does not contain any digit.

Q: What happens if I write 'Break 8 ALL' A: nothing Process 8 is a shell process which ignores any signal sent by Break.

## 1.5 Localizing Break 3701

```
[...] 0550: 25732022 2573223A 20002D2D 2D2D2D00 %s "%s": :-----. 0560: 434C4900 2D2D2D2D 2D2D2D2D 2D2D2D00
CLI-----. 0570: 436F6D6D 616E6400 2D2D2D2D 2D2D2D00 Command-----. 0580: 5461736B 002D2D2D 2D2D2D2D
2D2D2D00 Task-----. 0590: 506F7274 002D2D2D 2D2D2D2D 2D2D2D00 Port-----. 05A0: 000003F2 ...ð
```

The original C:Break command was not localized maybe to let it run on disk based system without opening requesters asking the Locale volume, and my program has only four words to be localized. These two reasons are, in my opinion, enough to not support directly the locale.library, but this does not mean that Break has to talk only one language. If you feel yourself hacker enough, you can manually patch the program to talk your preferred language!

The words at +\$560, +\$570, +\$580, +\$590, are four null-terminated strings (not longer than 16 bytes each, including terminator).

The string at \$550 is the message format string, and it is magic: words appear in different order according to the position of the first double quote character (") relative to the first percent character (%). Example:

```
%s "%s": --> % then " --> CLI "10": object not found "%s" %s: --> " then % --> "10" CLI: object not found
```

Have phun! ;-)