

**AddMem**

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> AddMem		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		March 28, 2025	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>AddMem</b>	<b>1</b>
1.1	AddMem.doc . . . . .	1
1.2	AddMem.doc - Description . . . . .	1
1.3	AddMem.doc - Memory attributes . . . . .	2
1.4	AddMem.doc - Address . . . . .	3
1.5	AddMem.doc - Copyright . . . . .	3
1.6	AddMem.doc - History . . . . .	4

# Chapter 1

## AddMem

### 1.1 AddMem.doc

AddMem

Version 3.01 written on 8/29/1994  
by Martin Schlodder

This is version 3.01 of AddMem, a bugfix for version 3.0.

This program only runs with OS 2.0 or higher. It behaves like a typical CLI command through the use of ReadArgs(). If run with OS 2.1 or higher, it prints all messages in the preferred language if a suitable catalog is installed. The installation is rather quick: Just copy the program to any directory in the path of the shell, preferably "SYS:C", and the suitable catalog (in the moment there is only a german catalog available, english texts are built in) to "LOCALE:Catalogs/<language>". The documentation will not be really necessary, because AddMem has a help text built in.

Copyright  
Description  
History  
Address

//  
Thanks to \X/ Amiga for being the best computer ever !

### 1.2 AddMem.doc - Description

AddMem adds expansion memory that is not auto configuring to the system's memory list. The required flags for the memory type will be set automatically as far as possible. If these precalculated flags are not ok, they might be changed through the arguments to the program call. The start and end addresses will be validated (certain areas of the Amiga's memory range are reserved) and the program will look if there really is RAM at this position (reset proof programs and data will not be affected by this). If required, AddMem can do a bitwise memory check wich will take some time (about one minute per MByte) and wich destroys any data in this area. AddMem requires at least OS 2.0 (better

---

OS 2.1) and of course a non autoconfig memory board.

```
Usage: AddMem startaddress endaddress [CheckMem] [A1000Fast] [32Bit]
      [LOCAL] [!PUBLIC] [CHIP] [FAST] [!24BITDMA]
      [priority] [RESIDENT]
```

startaddress, endaddress:

Start and end addresses of the memory area as hexadecimal numbers. Both of them must be a multiple of eight. The numbers may begin with a '\$'. The endaddress might be either the last byte of the memory area or the first byte after it (e.g. \$200000 to \$3ffffff or \$200000 to \$400000).

CheckMem:

Memory will be checked before being added to the system's memory list.

A1000Fast:

The memory range will become PUBLIC|FAST|LOCAL|24BITDMA with a priority of 0, if it is located between \$200000 and \$A00000.

32Bit:

The memory range will become PUBLIC|FAST with a priority of +5 and depending on it's position 24BITDMAable (if it is located below \$1000000).

LOCAL:

This argument should only be set, if the memoy is directly connected to the CPU, i.e. not on the Zorro bus.

!PUBLIC, CHIP, FAST, !24BITDMA:

Memory attributes to override the precalculated type. Seldom needed.

priority:

Priority of the memory area. Defaults to 0.

RESIDENT:

An AddMem program will be placed in the boot process (to be exact, in the CoolCapture). It can be removed by pressing the left mouse button for some seconds after a reset or by switching the computer off and on again or by doing a reboot wich removes Exec, such as my program Reset does. The CoolCapture program checks for double enrties of the memory range, and in that case removes any program that used the CoolCapture and thus any boot program initialized by AddMem. You can of course add several memory areas with this option set. They will all be added after a Reset with the chosen flags.

Memory attributes - A more precise description

This program was written completely in assembly language (OMA 2.05).

## 1.3 AddMem.doc - Memory attributes

In Kickstart 1.3 the following memory flags were defined:

- PUBLIC      Was nearly always set and therefor had no meaning. Today it's used by virtual memory managers to determine which memory may be swapped. It is always set in the memory list.
- CHIP        CHIP memory is accessible by the Amiga's custom chips (who are responsible for graphics, sound, diskdrive etc.). It was origi-located in the lowest 512 KBytes of the Amiga's memory range, then 1 MByte and now 2 MBytes.
- FAST        FAST memory is all memory except CHIP memory. It is usually faster than CHIP memory, as the processor can access it directly

and is not slowed down by custom chip DMA. The only exception is the Ranger Memory of the Amiga 500 and the Amiga 2000A, which is not accessible by the custom chip DMA and still slowed down by it.

In OS 2.0 the following flags were added:

- LOCAL        This is memory, that is direktly accessible by the CPU and which is not connected through the Zorro bus. This memory will still be accessible after a RESET CPU instruction while Zorro memory might vanish.
- 24BITDMA    Memory which is located in the address range of the Zorro II bus (this is the address range of the MC68000, \$000000 to \$FFFFFF) has this flag set.

This flag is defined since OS 3.0:

- KICK        Memory which is added to the system's memory list before the KickTags are processed will be of this type.

AddMem uses all these flags. PUBLIC will always be set if the argument !PUBLIC was not set. CHIP is set if the memory range is located below \$200000 or if the argument CHIP is given. If the memory is located above \$200000 or if the argument FAST was given, the flag FAST will be set. The flag 24BITDMA is set if the memory is located below \$1000000 and if the argument !24BITDMA is not given. According to the Autodocs, KICK should not be set when adding memory. But when the memory is added by the RESIDENT CoolCapture program, it is added early enough so that KICK can be set.

## 1.4 AddMem.doc - Address

For questions or suggestions you may reach me:

via EMail (internet):  
schlodder@student.uni-tuebingen.de

Or write me:  
Martin Schlodder  
Uhlandstr. 18  
D-72336 Balingen

## 1.5 AddMem.doc - Copyright

AddMem is FreeWare. You may use it and copy it, as long as you leave it unchanged.

DISCLAIMER:

I am not responsible for loss of data, damage or other problems resulting directly or indirectly from the use of this program.

---

## 1.6 AddMem.doc - History

- V0.5: First Version. Uses the OS2.0 type PUBLIC|FAST|LOCAL|24BITDMA for the new memory.
  - V0.6: Now checks for the option 'CheckMem' before testing the memory.
  - V1.0: Works now with Kick 1.3 and earlier versions by using the memory type PUBLIC|FAST for them.
  - V2.0: This version, again, runs only with OS 2.0 and higher, for is uses now the new commandline parsing function of DOS. Any type of memory might now be added, since any combination of memory flags can be specified (except CHIP|FAST).
  - V2.01: FPutS replaced by PutStr.
  - V2.1: Accepts now the OS 3.0 memory flag KICK.
  - V3.0: Localized and RESIDENT option introduced. PUBLIC is now always set, CHIP, FAST and 24BITDMA are set according to the memory area's location. The parameter format has been changed severely.
  - V3.01: Bug removed wich prevented AddMem from accepting memory above \$100000.
-