

ARTM

COLLABORATORS

	<i>TITLE :</i> ARTM		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		December 19, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	ARTM	1
1.1	ARTM Anleitung	1
1.2	Copyright	2
1.3	shareware	2
1.4	Adresse des Authors	3
1.5	installation	3
1.6	programmstart	4
1.7	tastaturbelegung	5
1.8	hilfe	5
1.9	funktionen	5
1.10	aktionen	7
1.11	tasks	8
1.12	windows	8
1.13	libraries	9
1.14	devices	10
1.15	resources	10
1.16	ports	11
1.17	residents	11
1.18	interrupts	11
1.19	vectors	12
1.20	memory	12
1.21	mount	13
1.22	assign	14
1.23	locks	14
1.24	fonts	14
1.25	hardware	15
1.26	res_cmds	16
1.27	semaphoren	16
1.28	monitor	17
1.29	last alert	17

1.30	zz	18
1.31	i	18
1.32	disksymbol	18
1.33	priority	18
1.34	remove	19
1.35	close	19
1.36	release	20
1.37	clear	20
1.38	freeze	20
1.39	unlock	21
1.40	more	21
1.41	break	22
1.42	signal	22
1.43	cancel	22
1.44	monitor_acts	23
1.45	history	23
1.46	zukunft	23
1.47	hinweis	24
1.48	Danksagungen	24
1.49	Index	24

Chapter 1

ARTM

1.1 ARTM Anleitung

ARTM Amiga Real Time Monitor V2.04

ARTM displays and controls activities in your Amiga. It is full supported by Intuition and it is very easy to use. It works under WB2.x properly. You can change many system structures.

Copyright

Copyright

shareware

shareware contribution

Author

Adressen from Author

Installation

Installation ARTM

Programstart

Start ARTM

Keyboard

Keyboard Codes

Help

ARTM Helpfunction

Funktionen

Funktions in ARTM

Aktionen

Actions in ARTM

History
 Story of ARTM

Future
 Development of ARTM

Note
 Important please read it !!!

Thank you to
 uncommentary !!?!?

```

      / ' ( ) ' ) ) )      _/_
, - / - , - _/_ / / / _ _ / _ _ _
(_ / o    / / o / ' ( (<_ / ( _ (<_ / ) _/_ ) _
      (_/

```

1.2 Copyright

Copyright and other things:

Copyright:

Copyright © 1990 - 1994 by D. Jansen & F.J. Mertens

ARTM is a

Shareware

program. The package may not be altered in any way and cannot be used for commercial purposes without the prior written permission of the author. The copyright message should be preserved.

Warranty:

No responsibility or liability will be accepted for any damage that may appear to have resulted from use of this program. All use is at your own risk. The software is provided "as is" without any warranty implied or otherwise to the fitness or accuracy of the software and documentation. The documentation is believed to be correct but the author reserves the right to update the software and/or documentation without notice.

1.3 shareware

shareware:

ARTM is a

Shareware

program. Permission is granted to freely distribute this program without profit. Copy is allowed but only completely and unchanged. It's not allow to use ARTM for commercial. If you use it often, please pay us a donation of 15 US\$. In exchange you will get an update without the troublesome requester.

Please send cash and no check. My bank is a shark. When you send a 15\$ US check: exchange in the moment 1\$ US = 1.60 DM

```
20$ US = 25.00 DM
- 11.00 DM charge to change an US$ check in German DM
-----
14.00 DM
```

Every registered ARTM-User will get all future updates free, when I get a envelop with a stamp and a formatted disk or equivalent money. When the update will go a long time, please have comprehension that I must work the whole day.

1.4 Adresse des Authors

Address from the Author:

Franz-Josef Mertens
Sonnenrain 17
72218 Wildberg
Germany

E-Mail:
UUCP: fjm@merlin.tynet.sub.org
FIDO: (2:241/7407.9) Mentopolis Wildberg

1.5 installation

Installation automatically:

You can install ARTM automatically with the C= Installer. In archive ARTM is an installer script.

Installation by your fingers:

The installation is very easy. ARTM is only one program, so you can copy ARTM with the copy command `copy ARTM path/ARTM` in your favorite directory. Copy the .doc files with the copy command in your favorite directory too. The ARTM.guide file should you copy to `LOCALE:HELP/`.

```
copy ARTM.guid SYS:LOCALE/HELP/ARTM.guide
```

```
assign HELP: SYS:LOCAL/HELP
```

Following files are in archiv ARTM:

```
ARTM
ARTM.info
Install_ARTM
Install_ARTM.info
german.doc/artm.guide
german.doc/artm.guide.info
german.doc/artm.dok
german.doc/artm.dok.info
german.doc/artm.history
german.doc/artm.history.info
english.doc/artm.guide
english.doc/artm.guide.info
english.doc/artm.doc
english.doc/artm.doc.info
english.doc/artm.history
english.doc/artm.history.info
```

1.6 programmstart

Programstart from Shell:

```
Type ARTM [opt]
```

and ARTM will open on the WB Screen

there are many options available:

- q Will bring up ARTM in its iconified form in the upper right hand corner.
- poutputpath say where ARTM should be store his output. Possible are all devices prt: ram:file.
- ix/y Will bring up the ARTM window in its full size, but when iconified the icon will be located at x and y position.
- wx/y Start ARTM as WB Fenster, x/y is the left upper point from ARTM-Window. Should the difference from x/y to the right screenside lower than 640 Pixel ARTM will start at the left upper corner.
- hx Start ARTM as WB window, x=window.height. Is the screen shorter than x ARTM-Window.Height = 168.

Programmstart from WorkBench:

Simply double click the ARTM icon and the ARTM window will open on the Workbenhscreen

The options mentioned above can be used as Tooltype.

```
START=WINDOW|ICON
ICON_POS=x/y
OUTPUT_PATH=RAM:ARTM.output
WINDOWHEIGHT=280
WINDOW_POS=70/16
```

1.7 tastaturbelegung

Keyboardcodes are supported in ARTM:

this version of ARTM support only 3 keycodes. In one of the next versions all functiones should be useable over the keyboard.

Followed keyboardcodes are supported:

```
ESC      finished ARTM
HELP     the ARTM.GUIDE pop up and show you the actual function help
         (see function
           Help
         )
```

ARTM requester known followed keyboardcodes:

```
ESC      cancel the function is like CANCEL
RETURN   execute the function is like OK
```

1.8 hilfe

The Helpfunction in ARTM:

an extensive help is available in ARTM. After hitting helpkey, ARTM.guide pops up and the last selected function help will be show you. The amigaguide.librarie in your Libs: directory is required. Allso ARTM.guide must be correct installed. (see
Installation
)

1.9 funktionen

Followed functiones are implemented:

```
Tasks
      list all running tasks
```

```
Windows
```

list all screens and windows

Libraries
list all open libraries

Devices
list all open devices

Resources
list all Resources

Ports
list all open ports

Residents
list all resident programs

Interrupts
list all interrupts

Vectors
list the value from systemectors

Memory
list all memorybanks

Mount
list all mounted devices

Assign
list all assigned devices

Locks
list all locks

Fonts
list all disk- and romfonts

Hardware
show the most significant hardware things

Resident Cmds
list all residente programs

Semaphoren
list all semaphores

Monitor
a little memory-monitor readonly

Last Alert
show the last alert number and address

zz
the ARTM-Window goes iconify

Search search a ASCII string in functin Monitor

Cancel switch last gadget-row from function to action

1.11 tasks

Tasks:

is default at start of ARTM and lists all tasks in your Amiga. You select a task with a mouseclick on the requested entry. Now you can choose any action.

These are possibile actions:

Remove
,
Priority
,
Freeze
,
Activate
,
More
,
Break
 and
Signal
These informations are displayed:

Address, Priority, Signals, State, Code, Type, Name.

Example:

Address	Pri	Stack	Signals	State	Code	Type	Name
003f96f0	0	4094	00000010	run	ET--	process	ARTM
002f1cc0	0	2998	80000000	wait	ET--	process	« ConClip »

1.12 windows

Windows:

lists all open screens and windows. Screens are in color 3 windows are in color 2. In a standart palette WB2.0 coler 3 ar white and coler 2 is black. Select a window or screen by clicking on it and than select any

followed action.

These are possible actions:

Close
and
More
.

These informations are displayed:

Address, Position, Size, Flags, IDCMP (nur bei Windows), Title.

Example:

Address	Pos	Size	Flags	IDCMP	Title
002c1428	0,0	708,512	00004611		Workbench Screen
003da88c	68,16	640,283	2800300f	00400670	AMIGA RealTime Monitor V2.0 Beta

1.13 libraries

Libraries:

lists all Libraries. By click on any item, you can do any actions with the selected library. When you remove a library (be carefull the OpenCount must be NULL) you can do this with action close.

These are possible actions:

Remove
,
Priority
, and
Close
These informations are displayed:

Address, Type, Priority, OpenCount, Version (setzt sich zusammen aus Version & Revision), Negsize, Possize, Name.

Example:

Address	Type	Pri	OpenC	Version	Neg	Pos	Name
002801b0	library	0	25	40.1	296	48	utility.library
00283d74	library	0	1	40.24	1076	544	graphics.library

1.14 devices

Devices:

lists all devices. Select any item to do any action with it. Before you remove a device be sure that it's closed.

These are possible actions:

Remove
and
Priority
These informations are displayed:

Address, Type, Priority, OpenCount, Version (consist of version & revision), Negsize, Possize, Name.

Example:

Address	Type	Pri	OpenC	Version	Neg	Pos	Name
00286e74	device	0	1	40.1	36	338	gameport.device
0028700c	device	0	19	39.4	68	380	timer.device

1.15 resources

Resources:

lists all resources. By click on any item, you can do any actions with the selected resource.

These are possible actions:

Remove
and
Priority
These informations are displayed:

Address, Type, Priority, OpenCount, Version (consist of version & revision), Negsize, Possize, Name.

Example:

Address	Type	Pri	OpenC	Version	Neg	Pos	Name
00280254	resource	0	0	37.0	20	42	potgo.resource
002802b8	resource	0	0	0.0	24	136	ciaa.resource

1.16 ports

Ports:

shows all ports and the name of its task. IDCMP_Ports are owned by a window. Be sure that the process is killed when you remove his port, else the ALERT visit you.

These are possible actions:

Remove

These informations are displayed:

Address, PortName, Type, SigBit, Flags, TaskName.

Example:

Address	Pri	Portname	Type	SigBit	Flags	TaskName
00300d50	0	fBlanker	msgport	30	signal	fBlanker
0032b864	0	REXX	msgport	31	signal	RexxMaster

1.17 residents

Residents:

lists all residents in your Amiga. Residents which were added by the KickTagPtr (i.e. a virus or RAD:) will be displayed in color 3 and if the residents are in the ResModules-table too, they will be displayed in color 3.

These informations are displayed:

Address, Type, Priority, Version, Flags, Name.

Example:

Address	Type	Pri	Vers	Flags	Name
002036d0	library	110	40	00000002	expansion.library
002000b6	library	105	40	00000002	exec.library

1.18 interrupts

Interrupts:

shows all interrupts in your Amiga. It's only allowed to remove an interrupt controlled by a server.

These are possible actions:

Remove

These informations are displayed:

Address, Data, Code, Priority, Type, Nr., InterruptName, Name.

Example:

Address	Data	Code	Pri	Type	Ctrl	Nr	IntrName	Name
00283882	00283834	0021507c	0	unknown	hand	1	Disk Block	disk.resource
002802e2	002802b8	0020e6ba	120	interrupt	serv	3	CIA-A	ciaa.resource

1.19 vectors

Vectors:

show vectors, i.e. Cool-, Cold-, WarmCapture and Kickpointers. It's helpful when searching for Viruses, but remember that the RAD: device in Kick 1.3 uses the KickTagPtr. Use function clear to set any Vector to NULL. If you close KickMemPtr, KickTagPtr or KickCheckSum, all of these three would be set to NULL. Should the KickTagPtr point to a resident table, you get the Address & Name from this residents. Note, that only the first item of the table would be displayed. If you want to see the others, select function Residents. The residents of the KickTagPtrtable would be displayed colored. (see above).

These are possible actions:

Clear

These informations are displayed:

Address, VectorName und Content, Should the KickTagPtr point to a resident table, you get the Address & Name from this residents.

Example:

Address	Content	Name	Address	Name
0028083a	00000000	ColdCapture		
0028083e	00000000	CoolCapture		
00280842	00000000	WarmCapture		
00280a32	00000000	KickMemPtr		
00280a36	00000000	KickTagPtr		
00280a3a	00000000	KickCheckSum		

1.20 memory

Memory:

show the memory in chip, fast and expansion ram (like avail). Function priority changes the priority. If you select the memory (chip or fast) and klick more then ARTM will show you the memoryfragmentation.

These are possibile actions:

Priority
and
More
These informations are displayed:

Address, Priority, Attribute, Lower, Upper, Maximum, in Use, Free, Largest, Name and in the next colum the totals.

Example:

Address	Pri	Lower	Upper	Maximum	In-Use	Free	Largest	Name
00280000	0	00280020	00600000	3583k	1692k	1891k	1853k	expansion mem
00000400	-10	00000420	00100000	1022k	237k	785k	763k	chip memory

1.21 mount

Mount:

list all mounted Devices in your Amiga.

These informations are displayed:

Address, Device, Heads, Sectors, Buffer, Cylinder, Interleavefaktor, Handlername. By memoryunits ARTM will dispaly Size (in MB or KB), Full (in %), rw (read/write), Volume (name) too.

Example:

Address	Device	Hd	Sec	Buff	Cyl	I	HandlerName	Size	Full	RW	Volume
002d080c	PIP:										
002c03d4	CNC:										
002bb40c	PIPE:						L:Queue-Handle				
00282c5c	RAM:							304	100%	rw	Ram Disk
00282c1c	CON:										
00282bdc	RAW:										
00282b9c	SER:						L:port-handler				
00282b5c	PAR:						L:port-handler				
00282b2c	PRT:						L:port-handler				
002915b0	DH0:	4	42	30	586	0	gvpscsi.device	48	98%	rw	Amiga
00291438	DF0:	2	22	20	80	0	trackdisk.devi	1	98%	rw	PSFonts_abd
00291688	DH1:	4	42	30	366	0	gvpscsi.device	30	94%	rw	Platte
00291748	DH2:	4	42	32	74	0	gvpscsi.device	6	98%	rw	Mail

```
002917f8 DH3:    4  42   32  196  0 gvpscsi.device  16   87% rw dh3
002918b8 DH5:    2  42   32 1215  0 gvpscsi.device  49   94% rw dh5
00291378 DF1:    2  11    5   80  0 trackdisk.devi      no disk present
```

1.22 assign

Assign:

show all assigned pathes and devices.

These informations are displayed:

Address, Directory, Path.

Example:

Address	Directory	Path
002bebc4	plz:	Amiga:plz
002be9ec	Documentum:	Amiga:Arbeit

1.23 locks

Locks:

show all Locks (read and write). Select a Lock and chose unlock and remove a selected item.

These are possibile actions:

Unlock

These informations are displayed:

Address, Access, Path.

Example:

Address	Access	Path
003eec1c	read	Ram Disk:
002ba6bc	read	Ram Disk:ENV
00365758	read	dh5:

1.24 fonts

Fonts:

list all loaded fonts, in KickRom too. With remove you can remove the

selected Font. If you finish a program that use the font (Count = NULL), the system should release the requested memory.

These are possible actions:

Remove

These informations are displayed:

Address, Count, YSize, XSize, Style, Flgs, LoChar, HiChar und Name.

Example:

Address	Cnt	Y	X	Style	Flags	LoChar	HiChar	Type	Name
00286cf8	25	8	8	00	41	32	255	ROM	topaz.font
00286ca8	3	9	10	08	49	32	255	ROM	topaz.font
002df3ea	9	8	8	00	42	32	255	disk	pearl.font

1.25 hardware

Hardware:

list all installed autoconfig-boards (ExpansionBase) ARTM will show you which processor and mathcoprocessor is in your Amiga. In addition ARTM indicate the new ECS-Chips Fat Agnus or Fat Denise (PAL or NTSC) or normal Agnus and Denise. Now ARTM will detect the new AA Chipset. At last you can see the state from the CPU 68020 and up, and the used Kickstart (from ROM or kickfile) and Workbench revision.

These informations are displayed:

Address, Boardadr, Boardsize, Slotadr, Slotsize, Driver, Product, Type, Manufacturer und Board.

My Hardware as example:

Address	Boardadr	Bo.Size	Slotadr	Ssize	Driver	Prod	Type	Manuf	Boardname
00000420	00200000	2048kB	00000020	32	no	81	230	514	A2630
00000468	00e90000	64kB	000000e9	1	yes	11	209	2017	GVP-II HD
000004b0	00400000	2048kB	00000040	32	no	10	230	2017	GVPMemory

Processor	68030
Math CoProcessor.....	68882
MMU	68030 (OFF)
DMA CustomChip	ECS Agnus (PAL)
Graphic CustomChip	Normal Denise
CPU Instruction Cache	ON
CPU Instruction Burst	ON
CPU Data Cache	ON
CPU Data Burst	ON
Kickstart Rom	2.04
Rom Kickstart Version	37.175

Used Kickstart Version 37.175
 Workbench Version 38.12

1.26 res_cmds

Resident Cmds:

list all Residents commands in the ROM and all programs they are made resident with resident programm pure include DataHunks.

These informations are displayed:

Name, ROM INTERN,

and if the program is made resident with resident programm pure you will see the Lower, Upper und Size from the DataHunks.

Example:

Name

```
-----
ls
Hunks:
Lower      Upper      Size
-----
004351c0   00438324   12652
004046b0   0040530c   3172
total:                15824
```

```
Alias          ROM INTERN
Ask            ROM INTERN
CD             ROM INTERN
```

1.27 semaphoren

Semaphoren:

show all Semaphores from AmigaDos. With remove you can remove the selected Semaphore but before, the Semaphore will automaticly be released. With release you can release any Semaphores.

These are possibile actions:

```
Remove
and
Release
These informations are displayed:
```

Address, Priority, NestCount, WaitQueue, Name, OwnerTask.

Example:

Address	Pri	Nest	Queue	Name	OwnerTask
002d775c	0	0	-1	« IPrefs »	NO_OWNER_TASK
0038ae74	-100	0	-1	Alert Patch	NO_OWNER_TASK

1.28 monitor

Monitor:

is a little memorymonitor and shows the whole memory. The memory will be loaded 100 lines in the ARTM stringgadget. With forward and backward you can view the next or previous 100 records. A click on address will bring up a requester, type an address in it and the Monitor list the value from memory and the next 100 records from the address. By klick on search a requester pop up. Fill in an ASCII string and klick on return. If ARTM found the string it will be display on top and the next 100 lines.

These are possible actions:

```

Forward
'
Backward
'
Address
and
Search
These informations are displayed:

```

Address, Bytes (Hex), Bytes (ASCII)

Example:

Address	Bytes (Hex)	Bytes (ASCII)
00200000	11 16 4e f9 00 20 00 d2	. . N ù . . Ò
00200008	00 00 ff ff 00 28 00 3e	. . ŷ ŷ . (. >
00200010	00 28 00 0a ff ff ff ff	. (. . ŷ ŷ ŷ ŷ
00200018	00 41 4d 49 47 41 20 52	. A M I G A R
00200020	4f 4d 20 4f 70 65 72 61	O M O p e r a
00200028	74 69 6e 67 20 53 79 73	t i n g S y s
00200030	74 65 6d 20 61 6e 64 20	t e m a n d
...		
..		
.		

1.29 last alert

Last Alert:

show the last Alertnumber and address in the comment field above the list-view gadget.

1.30 zz

zz:

Iconify function. If you chose the zz gadget ARTM disappears and popup as an icon. A doubleclick on this icon and ARTM will bring its Window back.

1.31 i

I:

show the AboutRequester

1.32 disksymbol

Disksymbol:

store all items from the current called funktion in a file or put it out to PRT: see

Programstart
option -poutputpath.

1.33 priority

Priority:

changes the priority of a selected function. Values may range from -128 to 127, better use values between -10 and 10. If you change a programpriority to 10 and up please remember important processes could run much slower.

These functiones are supported:

Tasks
list all running tasks

Libraries
list all open libraries

Devices

```
list all open devices

Resources
list all Resources

Ports
list all open ports

Interrupts
list all interrupts

Memory
list all memorybanks
```

1.34 remove

Remove:

removes selected function. Select (before remove) with a mouseclick on the requested entry. But remember libs and devices must be closed.

These functiones are supported:

```
Tasks
list all running tasks

Libraries
list all open libraries

Devices
list all open devices

Resources
list all Resources

Ports
list all open ports

Interrupts
list all interrupts

Fonts
list all disk- and romfonts

Semaphoren
list all semaphores
```

1.35 close

Close:

closes screens/windows and libraries. Use it as remove.

These functiones are supported:

Windows

list all screens and windows

Libraries

list all open libraries

1.36 release

Release:

relase Semaphores from SignalSemaphoreList.

These functiones are supported:

Semaphoren

list all semaphores

1.37 clear

Clear:

set the SystemVectors to NULL. Use it as remove or close. Is any System-vector not = NULL so a Virus could be resident in your Amiga. Function Clear will remov him from the resident structure. After the next reset, he is removed. Some usefull programs could be resident too Rad etc.

These functiones are supported:

Vectors

list the value from systemectors

1.38 freeze

Freeze:

freeze a runnig Task and removed it form the nodestructur. The frozen task-structur is stored. You recognize it function Task on STATE "Frozen".

If you select a frozen task and click Activate the task get a new live.

Activate:

is the counterpart from Freeze. Show above.

1.39 unlock

Unlock:

remove a Lock (read or write) from the list. Attention if the system know the Lock, your Amiga will get an Alert.

1.40 more

More:

show detail informations about task, process, CLI, window, screen und memory structure. It's not ever the complete structur but the most important things.

These functiones are supported:

Tasks	list all running tasks
Windows	list all screens and windows
Memory	list all memorybanks

Example:

Klick on the requested task and then on More. This is the result from:

Task- Process- and ComandLineInterface Structure plus Hunks from ToolManger

Task Structure:

```

-----
IDNestCnt:          -1                TDNestCnt:          0
SigAlloc:          c000ffff           SigWait:           c000f000
SigRecvd:          00000100           SigExcept:         00000000
TrapAlloc:         00008000           TrapAble:          00000000
ExceptData:        00000000           ExceptCode:        002033f0
TrapData:          00000000           TrapCode:          0021678e
SPLower:           0037ca48           SPUpper:           0037da48
SPReg:             0037d946
Switch():          00000000           Launch():          00000000
MemEntry:          00389b16           UserData:          00000000
Process Structure:

```

```

WindowPtr:      00000000          Title:
CurrentDir:    00371fa8          Path: Amiga:
MsgPort:       00000000
StackBase:     00389bac          StackSize:      3200
CIS:           0036577c          COS:            003654a4
ConsoleTask:   00000000          FileSystemTask: 00294ca4
SegList:       00367328          SegPointer:     00000000
PktWait:       00000000          CLI:            000d98ff
CommandLineInterface Structure:
CommandDir:    002c94a4
StandardInput: 0036577c          StandardOutput: 003654a4
CurrentInput:  0036577c          CurrentOutput:  003654a4
Module:        0038c9dc
Hunks:
Lower      Upper      Size
-----
0038c9e0  00396520    39752
0037b7b8  0037ca40     4752
00023378  00023408     152
total:                    44656

```

1.41 break

Break:

cancel a runnig Task. It will be set Control D/E/F/C

1.42 signal

Signal:

A requester pops up where you can change the value from tc_SigAlloc. Then Dosfunction Signal get it back.

See function

Break

1.43 cancel

Cancel:

switch the last gadgetrow from action (color 4) back to function (color 2). So function monitor, Hardware etc. are visible.

1.44 monitor_acts

Address:

Pop up a little Requester. Put here your favorite address. Function Monitor search the address and will display it in the first item plus the next 100 rows.

Forward:

list the next 100 rows.

Backward:

list the previous 100 rows

Search:

Pop up a little Requester. Put here the searched ASCII string. Function Monitor search the string and when it is found ARTM will display it in the first item plus the next 100 rows.

1.45 history

History:

The program-history from ARTM is a separated file like ARTM.History. The reason to not include the history in this guide is, the guide grows up and would be too big.

1.46 zukunft

Future:

I would like to add to ARTM some new features in the near time. Some things go slower than I've a fulltime job too. These listed functions will be added in one of the next versions:

show the CPU-Usage

switch on- and off the CPU-Caches

A little preference program instead of tooltypes and shell-options

a trap handler to catch some alerts

specified output to function

 Last Alert

 a little AREXX port

a shared Library board.library for additional hardware boards

If you have some suggestions please let it me know.

1.47 hinweis

Note:

If you use the function hardware and have installed autoconfig-boards (ExpansionBase). Please send me a mail and tell us the output from ARTM (PROD, TYPE and MANUFACTRER) and also the name and Type of your expansion-boards.

You should ever know what you do! You could change some things in your Amiga so, that the alert visit you. Be careful.

Please send bug reports and a list of features you would like to have in the next version to the Author. The easy way to call the author is E-Mail see

Author

.

1.48 Danksagungen

Danksagungen:

Special thanks to all, who helped me with little routines, tips and failure tracking and all Betatesters. Special thanks to:

Matthias Zepf
Markus Stoll.
Matthias Scheler
Kai Bolay

and all others who have send bugreports and tips. My sepcial thanks to all they have payed the shareware contribution, and so helped me to develop ARTM in the future.

1.49 Index

Index:

Activate

Address

Aktionen
Assign
Author
Backward
Break
Cancel
Clear
Close
Copyright
Devices
Disksymbol
Fonts
Forward
Freeze
Funktionen
Future
Hardware
History
I
Installation
Interrupts
Last Alert
Libraries
Locks
Memory
Monitor
More
Mount
Note

Ports

Priority

Programmstart

Release

Remove

Resident Cmds

Residents

Resources

Search

Semaphoren

shareware

Signal

Tasks

Thank you to

Unlock

Vectors

Windows

zz
