

stormamiga_lib

Matthias Henze

COLLABORATORS

	<i>TITLE :</i> stormamiga_lib		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Matthias Henze	July 19, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	stormamiga_lib	1
1.1	stormamiga_lib: Inhalt	1
1.2	stormamiga_lib: Einleitung	1
1.3	stormamiga_lib: Registrierung	2
1.4	stormamiga_lib: Systemanforderungen	3
1.5	stormamiga_lib: Installation	3
1.6	stormamiga_lib: Funktionen	4
1.7	stormamiga_lib: Beispiele	6
1.8	stormamiga_lib: Bekannte Fehler	7
1.9	stormamiga_lib: Updates	7
1.10	stormamiga_lib: Kopierrecht	7
1.11	stormamiga_lib: Geschichte	8
1.12	stormamiga_lib: In Zukunft	10
1.13	stormamiga_lib: Danksagungen	10
1.14	stormamiga_lib: Autor	11

Chapter 1

stormamiga_lib

1.1 stormamiga_lib: Inhalt

stormamiga.lib Version 41.034 (31.07.1996)

© Copyright 1996 by COMPIUTECK

geschrieben von Matthias Henze

F R E E W A R E

Einleitung Informationen über die stormamiga.lib.

Registrierung Weshalb man sich registriert.

Systemanforderungen Was braucht man für die stormamiga.lib?

Installation Wie installiere ich die stormamiga.lib?

Funktionen Beschreibung der einzelnen Funktionen.

Beispiele Beschreibung der Beispielprogramme.

Bekannte Fehler Wo gibt es Probleme?

Updates Wo gibt es neue Versionen?

Kopierrecht Das Rechtliche.

Geschichte Was hat sich bisher getan?

In Zukunft Was wird sich noch ändern?

Danksagungen Danksagungen an

Autor Wie erreicht man den Autor?

1.2 stormamiga_lib: Einleitung

Einleitung:

~~~~~

Da die Funktionen der "storm.lib" in C geschrieben sind,  
werden die damit gelinkten Programme sehr groß und langsam.

Die Funktionen der "amiga.lib" sind auch nicht gerade klein  
und schnell.

---

Aus diesem Grund habe ich mich am 18.03.1996 entschlossen die "stormamiga.lib" zu schreiben.

Die "stormamiga.lib" ist komplett in Assembler geschrieben.

Dadurch werden die damit gelinkten Programme auch sehr klein und schnell.

Mein Ziel ist es, alle Funktionen der "amiga.lib", die nicht in den Pragmadateien enthalten sind, und alle Funktionen der "storm.lib" durch kurze und schnelle Assembler Routinen zu ersetzen. Außerdem will ich einige Spezialbefehle von anderen Compilern (zur Zeit nur vom GCC) und einige Routinen, die das Programmieren erleichtern, in die "stormamiga.lib" integrieren.

Wichtig

In der jetzigen Version der "stormamiga.lib" sind noch nicht alle Funktionen der "storm.lib" und der "amiga.lib" enthalten.

Außerdem wird zur Zeit nur das große Code- und Datenmodell unterstützt.

Bis auf den Startupcode "stormamiga\_startups.o" sollte die "stormamiga.lib" auch mit C++ funktionieren. Allerdings wurde das noch nicht genügend getestet.

### 1.3 stormamiga\_lib: Registrierung

Registrierung:

~~~~~

Obwohl die "stormamiga.lib" Freeware ist, was auch so bleiben wird, können Sie sich bei **mir** registrieren lassen. Ich möchte vor allem wissen, ob sich die Weiterentwicklung überhaupt lohnt. Sie können mir natürlich auch Geschenke, die Sie für angemessen halten, zuschicken. Es wäre auch sehr schön, wenn Sie mir Ihr Programm, bei dem Sie die "stormamiga.lib" verwendet haben, zuschicken würden. Außerdem würde ich gerne Ihre Meinung zur "stormamiga.lib" erfahren.

Für Fehlerberichte (möglichst mit dem entsprechenden Quelltext und einer genauen Beschreibung) und Verbesserungsvorschläge bin ich jederzeit dankbar.

Wenn Sie Fragen zur "stormamiga.lib" haben, können Sie mich gerne anrufen oder mir schreiben. Ich werde die Antworten zu Ihren Fragen in der Anleitung der nächsten Version veröffentlichen.

Wenn ich Ihnen die Antworten zu Ihren Fragen persönlich zuschicken soll, dann müssen Sie mir natürlich einen frankierten Briefumschlag mitschicken. Wenn ich Ihnen die neueste Version der "stormamiga.lib" zuschicken soll, dann müssen Sie neben dem frankierten Briefumschlag auch eine Diskette (HD oder DD) mitschicken.

1.4 stormamiga_lib: Systemanforderungen

Systemanforderungen:

~~~~~

- Ein Amiga
- AmigaOS 2.0 oder höher
- MC68EC020 oder höher
- StormC Version 1.05 oder höher

## 1.5 stormamiga\_lib: Installation

Installation:

~~~~~

Im Verzeichnis stormamiga_lib befinden sich die Dateien "stormamiga.h", "stormamiga.lib", "stormamiga_lib.guide" und "stormamiga_startups.o".

Kopieren Sie die Includedatei "stormamiga.h" nach StormC: Include, die Bibliothek "stormamiga.lib" nach StormC:Lib und den Startupcode "stormamiga_startups.o" nach StormC: StormSYS. Die Anleitung "stormamiga_lib.guide" können Sie an eine beliebige Stelle kopieren.

Fügen Sie die "stormamiga.lib" an erster Stelle in das Projekt ein. Da in dieser Version der "stormamiga.lib" noch nicht alle Funktionen enthalten sind, müssen Sie noch die "storm.lib" in das Projekt einfügen.

Wenn Sie die Spezialfunktionen der "stormamiga.lib" nutzen wollen, müssen Sie noch die Includedatei "stormamiga.h", mit "#include <stormamiga.h>", in Ihren Quelltext einbinden. Die Includedatei "stormamiga.h" sollten Sie als letzte einbinden.

Um auch den neuen Startupcode "stormamiga_startups.o" zu nutzen, müssen Sie, bei den Linker Optionen der Projekteinstellungen, die Option "Eigener Startup-Code" einschalten und die Datei "stormamiga_startups.o" auswählen.

Wichtig

Der Startupcode "stormamiga_startups.o" kann nur für Ansi-C verwendet werden!

1.6 stormamiga_lib: Funktionen

Funktionen:

~~~~~

Da die normalen Ansi-C Funktionen bereits bei StormC beschrieben werden, erkläre ich nur die Spezialfunktionen und -befehle.

Hinweis:

Wenn Sie sich die Map-Datei, eines mit der "stormamiga.lib" gelinkten Programmes ansehen, werden Sie feststellen, daß einige Funktionen den Vorsatz "intern\_\_" (z.B. intern\_\_form\_in) besitzen. Diese Funktionen werden in der "stormamiga.lib" nur intern verwendet und können nicht als Befehle benutzt werden.

Der Startupcode:

Der Startupcode "stormamiga\_startups.o" ist speziell für Ansi-C gedacht.

Der einzige Unterschied, zum Startupcode "startup.o" des StormC, ist der Aufruf der Funktionen "main" und "wbmain".

Der Startupcode "stormamiga\_startups.o" ruft die Funktion "\_main\_\_", das entspricht der Funktion "main\_\_()" in Ansi-C, auf. Wenn es diese Funktion in Ihrem Programm, nicht gibt, wird sie aus der "stormamiga.lib" dazugelinkt. Die Funktion "\_main\_\_" ruft die Funktion "\_main", das entspricht der Funktion "main()" in Ansi-C, auf. Wenn es diese Funktion nicht gibt, meldet der Linker einen Fehler.

Bei einem Start von der Workbench ruft der Startupcode "stormamiga\_startups.o" die Funktion "\_wbmain", das entspricht der Funktion "wbmain()" in Ansi-C, auf. Wenn es diese Funktion nicht gibt, wird sie aus der "stormamiga.lib" dazugelinkt.

Die Funktion "main\_\_()":

Wenn Sie für Ihr Programm keine Auswertung von Argumenten benötigen, können Sie die Funktion "main()" in "main\_\_()" umbenennen. Dadurch wird Ihr Programm etwas kleiner.

Wichtig

Da der Compiler die Funktion "main\_\_()" nicht als normale main-Funktion erkennt, setzt er auch nicht den Returncode auf 0. Deshalb müssen Sie den Returncode, mit "return 0", selber auf 0 setzen.

Die Funktion "main\_\_()" benötigt den Startupcode "stormamiga

\_startups.o" und funktioniert nur in Ansi-C.

Kurzbeschreibung der neuen Befehle:

SPRINTF

VSPRINTF

Der Befehl "SPRINTF" entspricht in Funktion und Aufruf dem Befehl "sprintf" der "amiga.lib".

Der Befehl "VSPRINTF" entspricht dem Befehl "SPRINTF", unterstützt aber eine Parameterliste "vl".

Ich habe diese Befehle in die "stormamiga.lib" aufgenommen, weil sie die "exec.library" benutzen und dadurch sehr klein sind.

...printf\_

Bei diesen Spezialversionen werden nur die Typen "d", "i", "o", "x", "X", "u", "c", "s", "p", "n" und "%" unterstützt. Alle anderen Funktionen und der Aufruf entsprechen den Befehlen ohne "\_" am Namensende. Zur Zeit sind die Befehle "printf\_", "fprintf\_", "sprintf\_", "vprintf\_", "vfprintf\_" und "vsprintf\_" in der "stormamiga.lib" enthalten.

...printf\_\_

Bei diesen Spezialversionen werden nur die Typen "d", "i", "c", "s", "n" und "%" unterstützt. Alle anderen Funktionen und der Aufruf entsprechen den Befehlen ohne "\_\_" am Namensende. Zur Zeit sind die Befehle "printf\_\_", "fprintf\_\_", "sprintf\_\_", "vprintf\_\_", "vfprintf\_\_" und "vsprintf\_\_" in der "stormamiga.lib" enthalten.

v...scanf

Da die Befehle "vscanf", "vfscanf" und "vsscanf" nicht in der aktuellen "storm.lib" enthalten sind, sie aber von anderen Compilern unterstützt werden, habe ich sie in die "stormamiga.lib" aufgenommen. Sie entsprechen den Befehlen "scanf", "fscanf" und "sscanf", unterstützen aber eine Parameterliste "vl".

...scanf\_

Bei diesen Spezialversionen werden nur die Typen "d", "i", "o", "x", "c", "s", "p", "n", "[...]", "[^...]" und "%%" unterstützt. Alle anderen Funktionen und der Aufruf entsprechen den Befehlen ohne "\_" am Namensende. Zur Zeit sind die Befehle "scanf\_", "fscanf\_", "sscanf\_", "vscanf\_", "vfscanf\_" und "vsscanf\_" in der "stormamiga.lib" enthalten.

...scanf\_\_

Bei diesen Spezialversionen werden nur die Typen "d", "i", "c", "s",

"n", "[...]", "[^...]" und "%%" unterstützt. Alle anderen Funktionen und der Aufruf entsprechen den Befehlen ohne "\_" am Namensende. Zur Zeit sind die Befehle "scanf\_\_", "fscanf\_\_", "sscanf\_\_", "vscanf\_\_", "vfscanf\_\_" und "vsscanf\_\_" in der "stormamiga.lib" enthalten.

**muls**

Der Befehl "muls" ist ein schnellerer Ersatz für den Befehl "SMult32" der utility.library. Die Funktion und der Aufruf beider Befehle sind identisch.

**mulu**

Der Befehl "mulu" ist ein schnellerer Ersatz für den Befehl "UMult32" der utility.library. Die Funktion und der Aufruf beider Befehle sind identisch.

**divsl**

Der Befehl "divsl" ist ein schnellerer Ersatz für den Befehl "SDivMod32" der utility.library. Die Funktion und der Aufruf beider Befehle sind identisch.

**divul**

Der Befehl "divul" ist ein schnellerer Ersatz für den Befehl "UDivMod32" der utility.library. Die Funktion und der Aufruf beider Befehle sind identisch.

**Hinweis:**

Die Beschreibung der Befehle "bcmp", "bcopy", "bzero", "ffs", "index", "rindex", "memccpy", "strsep", "swab", "strcoll" und "strxfrm" entnehmen Sie bitte der Dokumentation des GNU C Compilers.

## 1.7 stormamiga\_lib: Beispiele

Beispiele:

~~~~~

Um die Vorteile und Anwendungsmöglichkeiten der "stormamiga.lib" etwas zu verdeutlichen, habe ich ein paar Beispiele beigelegt.

Die Beispiele mit dem Namen "...-storm" werden mit der "storm.lib" und dem Startupcode "startup.o" gelinkt.

Die Beispiele mit dem Namen "...-stormamiga" werden mit der "stormamiga.lib" und dem Startupcode "stormamiga_startups.o" gelinkt.

Die Beispiele mit dem Namen "...-stormamiga-2" werden mit der "stormamiga.lib" und dem Startupcode "stormamiga_startups.o" gelinkt.

Außerdem wurde der Quelltext für die "stormamiga.lib" optimiert.

1.8 stormamiga_lib: Bekannte Fehler

Bekannte Fehler:

~~~~~

- Der Befehl "printf" hat bei der Ausgabe von Fließkommazahlen einen Rundungsfehler.

## 1.9 stormamiga\_lib: Updates

Updates:

~~~~~

Wenn Sie Zugang zum Internet haben, dann können Sie die Updates auf der HomePage "<http://home.pages.de/~haage>" der Haage & Partner Computer GmbH bekommen.

Sie können die neueste Version natürlich auch direkt von **mir** bekommen.

Allerdings müssen Sie mir dann einen frankierten Briefumschlag und eine Diskette (HD oder DD) zuschicken.

1.10 stormamiga_lib: Kopierrecht

Kopierrecht:

~~~~~

Die "stormamiga.lib" ist FREEWARE. Sie darf frei kopiert werden, solange sie in KEINSTER Weise verändert wird und solange ALLE dazugehörigen Dateien UNVERÄNDERT mitkopiert werden.

Die "stormamiga.lib" darf auch im Zusammenhang mit anderen Programmen verwendet und vertrieben werden, solange KLARGESTELLT ist, daß es sich um FREEWARE handelt UND solange ALLE Dateien UNVERÄNDERT mitkopiert werden.

Mit der Weitergabe der "stormamiga.lib" darf kein Gewinn erzielt werden.

Der Verkaufspreis einer Diskette, die die "stormamiga.lib" enthält, darf nicht mehr als 5,- DM betragen. Ausgenommen davon sind Disketten, die es zu Zeitschriften gibt.

Eine Reassemblierung der "stormamiga.lib" ist selbstverständlich NICHT gestattet.

AM WICHTIGSTEN:

Die Benutzung der "stormamiga.lib" erfolgt AUSSCHLIEßLICH auf eigenes Risiko.

Der Autor kann auf KEINEN FALL für einen Schaden oder Datenverlust der direkt oder indirekt mit dem Gebrauch der "stormamiga.lib" entstehen sollte verantwortlich gemacht werden.

Alle Rechte vorbehalten. Für Fehlermitteilungen oder Verbesserungsvorschläge bin ich jederzeit dankbar.

## 1.11 stormamiga\_lib: Geschichte

Geschichte:

~~~~~

V41.000 alpha - V41.002 alpha (18.03. - 21.03.1996):

interne Entwicklungsphase

- Funktionen der "amiga.lib" für MC68000 oder höher geschrieben

V41.003 alpha - V41.021 alpha (22.03. - 16.05.1996):

interne Entwicklungsphase

- Funktionen der "amiga.lib" für MC68EC020 oder höher umgeschrieben (ab

V41.003 alpha wird ein MC68EC020 oder höher benötigt)

- Ein- und Ausgaberroutinen geschrieben (GCC-kompatibel)

- Funktionen zum automatischen Öffnen und Schließen der Libraries geschrieben

- die meisten ctype-Funktionen geschrieben

- einige stdio-, string- und stdlib-Funktionen geschrieben

- Startupcode für die "stormamiga.lib" optimiert

- einige andere Funktionen geschrieben

- einige Optimierungen und Fehlerkorrekturen

- Includedatei "stormamiga.h" geschrieben

V41.022 alpha - V41.029 alpha (20.05. - 11.06.1996):

interne Entwicklungsphase

- Ein- und Ausgaberroutinen komplett neu geschrieben (StormC-kompatibel)

- einige stdio-Funktionen geschrieben

- 64Bit Befehle der "storm.lib" für MC68EC020+ optimiert

- einige andere Funktionen geschrieben

- einige Optimierungen und Fehlerkorrekturen

- Includedatei "stormamiga.h" erweitert

- Anleitung geschrieben

Hinweis:

Durch ein Versehen wurde die Version 41.028 alpha, einige uralte Beispielprogramme und Teile der Anleitung, mit der Version 1.1 von StormC, veröffentlicht. Diese Version hat aber noch einige größere Fehler und funktioniert nicht mit den alten Beispielprogrammen.

Die erste, zur Veröffentlichung gedachte Version, ist die Version 41.032 beta.

V41.030 beta - V41.031 beta (13.06. - 15.06.1996):

interne Entwicklungs- und Testphase

- Optimierung der Ein- und Ausgaberroutinen
- Fehlerkorrekturen
- Anleitung überarbeitet

V41.032 beta (16.06.1996):

Erste öffentliche Version

V41.033 beta (17.06. - 15.07.1996):

- umfangreiches Betatesting
- Fehlerkorrektur der string-Funktion "memcpy" (Die Benutzung von "memcpy" führte zu einem Fehler oder zum Systemabsturz. Die Ursache war ein Tippfehler. Ich hatte "a2" statt "a1" geschrieben.)
- die mathematischen Funktionen "acos", "asin", "atan", "ceil", "cos", "cosh", "exp", "fabs", "floor", "log", "log10", "pow", "sin", "sinh", "sqrt", "tan" und "tanh" geschrieben
- die string-Funktionen "memccpy", "strcoll", "strlwr", "strsep", "strupr", "strxfrm" und "swab" geschrieben
- die stdlib-Funktionen "abs", "labs", "atoi", "atol" geschrieben
- die stdlib-Funktion "rand" optimiert
- die string-Funktionen "memchr", "memmove", "memset", "strcspn", "strpbrk", "strspn" und "strtok" optimiert
- Includedatei "stormamiga.h" überarbeitet und erweitert

V41.034 (16.07. - 31.07.1996):

- Fehlerkorrektur der internen Funktion "amigawrite" (Der auszugebende Text wurde erst nach einigen Sekunden angezeigt. Bei dem Programm "GadTools" wurde der Text erst ausgegeben, wenn 5 Schalter gedrückt wurden.)
 - die string-Funktionen "strerror" und "stricmp" geschrieben
 - die string-Funktionen "memcmp", "strcat", "strcmp", "strcpy", "strncat", "strncmp", "strncpy" und "strstr" optimiert
 - die stdlib-Funktionen "malloc" und "free" optimiert
 - die stdio-Funktionen "vfprintf", "vfprintf_", "vfprintf__", "vfscanf", "vfscanf_", "vfscanf__", "fflush" und "setvbuf" optimiert
 - die internen Funktionen "amigaread", "amigareadunget", "amigawrite", "amiga_eof", "amigaseek", "amigagetc", "amigagetcunget", "amigaungetc", "amigaputc", "amigaflush", "amigaclose", "SMult64", "UMult64", "SDiv64", "INIT_0_InitFiles", "EXIT_5_InitFiles", "INIT_5_InitStdIOFiles", "EXIT_5_InitStdIOFiles" und "EXIT_4_free" optimiert
 - "aufräumen" der "stormamiga.lib" (dadurch wird das Linken beschleunigt, die Programme etwas schneller und die Map-Datei übersichtlicher)
 - Anleitung überarbeitet
-

1.12 stormamiga_lib: In Zukunft

In Zukunft:

~~~~~

Die folgenden Punkte habe ich mir für die nächsten Versionen der "stormamiga.lib" vorgenommen.

- alle, noch fehlenden, Funktionen der "storm.lib" und der "amiga.lib" in die "stormamiga.lib" integrieren
- Unterstützung des kleinen Datenmodells a4 (eventuell auch a6)
- spezielle Version der "stormamiga.lib" für die Koprozessoren MC68881 und MC68882 (mit "68040.library" bzw. "68060.library" auch für den MC68040 und MC68060)
- spezielle Version der "stormamiga.lib" für den MC68040 (mit "68060.library" auch für den MC68060)
- neue Anleitung mit ausführlicher Beschreibung der einzelnen Funktionen

## 1.13 stormamiga\_lib: Danksagungen

Danksagungen:

~~~~~

Als erstes möchte ich mich bei der Haage & Partner Computer GmbH bedanken, weil sie mir ihre Entwicklerunterlagen und Quelltexte kostenlos überlassen haben. Ohne diese Unterlagen wäre ich wahrscheinlich an den Ein- und Ausgaberroutinen für die Befehle printf und scanf verzweifelt.

Besonderen Dank an Jochen Becher, der auch an Sonntagen zu später Stunde Zeit für meine Probleme hatte.

Ohne die Unterstützung der Haage & Partner Computer GmbH würde es die "stormamiga.lib", in dieser Form, nicht geben.

Außerdem möchte ich mich bei folgenden Leuten bedanken:

- Uwe Schienbein, für Betatesting, Bugreports und neue Ideen
- Karsten Bornholz, für Betatesting
- Dietmar Heidrich, für seinen "OMA"
- Frank Wille, für seinen "FreePhxAss"

1.14 stormamiga_lib: Autor

Autor:

~~~~~

Matthias Henze

Gorkistraße 119

04347 Leipzig

Deutschland

Telefon: 0341/2326414

---