

HOPE

Turbo C++ 运行库函数 源程序与参考大全 上册

(共二册)



中国科学院希望高级电脑技术公司

Turbo C++ 运行库函数 源程序与参考大全(上册)

亦 鸥 等编

■ CLIB 库函数源程序

下册内容包括:

- ☐ MATH 库函数源程序
- ☐ EMU 库函数源程序
- ☐ 头文件和.ASI 文件
- ☐ 修改库函数源程序
- ☐ 使用批处理文件来建库

中国科学院希望高级电脑技术公司
一九九一年五月

Turbo C++ 运行库函数 源程序与参考大全(下册)

亦 鸥 等编

- MATH 库函数源程序
- EMU 库函数源程序
- 头文件和.ASI 文件
- 修改库函数源程序
- 使用批处理文件来建库

上册的内容为

- CLIB 库函数源程序

中国科学院希望高级电脑技术公司
一九九一年五月

版权所有
不许翻印
违者必究

- 北京市新闻出版局
准印证号： 3188—90188
- 订购单位：北京 8721 信箱资料部
- 电 话： 2562329
- 电 传： 01—2561057
- 电 挂： 0755
- 地 址： 海淀影剧院北侧
- 乘 车： 320、332、302路海淀黄庄下车
- 办公地点： 公司大楼 101 房间

Turbo C++ 运行库函数源程序与参考大全

《Turbo C++ 运行库函数源程序与参考大全》整理编译了 Turbo C++ 运行库中所有函数的源程序以及详细的函数使用说明。它不仅全部提供了《Turbo C++ 参考手册》中提供的函数，而且还提供了实现这些函数的所有低层支持函数。这使得 Turbo C++ 运行库成为真正的开放式结构，用户可以从多方面高效地使用库函数。

不仅如此，有了库函数源程序，用户就可以对库函数进行单步断点等符号调试，可以领会 Turbo C++ 运行库以及整个软件的设计策略和技巧，为将来开发语言系统提供实例模型。Turbo C 之所以能风靡当今 PC 世界，不仅因为它有良好的用户界面，强大的符号调试功能，还因为有设计精良、运行高速、结构优化的运行库函数。运行库函数源程序让用户能目睹其真正风采，领略其库设计策略、技巧、程序设计风格、函数依赖关系等等。

有了库函数源程序，就可以无限制地把 Turbo C 与程序设计语言揉合起来，综合 C 与其它程序设计语言的优点进行编程，使程序设计更上一层楼。

对于高级程序员来说，通过修改库函数源程序，可更高效地、多方位地使用 Turbo C++，对于一般的编程者和初学 C 的人，它又是一个很好的实例教材，是进行程序设计的良师益友。

齐全的运行库函数与低层支持函数，将给用户带来无穷无尽的新发现与高效率。

Turbo C++运行库函数源程序与参考大全

《Turbo C++运行库函数源程序与参考大全》整理编译了 Turbo C++运行库中所有函数的源程序以及详细的函数使用说明。它不仅全部提供了《Turbo C++参考手册》中提供的函数，而且还提供了实现这些函数的所有低层支持函数。这使得 Turbo C++运行库成为真正的开放式结构，用户可以从多方面高效地使用库函数。

不仅如此，有了库函数源程序，用户就可以对库函数进行单步断点等符号调试，可以领会 Turbo C++运行库以及整个软件的设计策略和技巧，为将来开发语言系统提供实例模型。Turbo C之所以能风靡当今 PC 世界，不仅因为它有良好的用户界面，强大的符号调试功能，还因为有设计精良、运行高速、结构优化的运行库函数。运行库函数源程序让用户能目睹其真正风采，领略其库设计策略、技巧、程序设计风格、函数依赖关系等等。

有了库函数源程序，就可以无限制地把 Turbo C 与程序设计语言揉合起来，综合 C 与其它程序设计语言的优点进行编程，使程序设计更上一层楼。

对于高级程序员来说，通过修改库函数源程序，可更高效地、多方位地使用 Turbo C++，对于一般的编程者和初学 C 的人，它又是一个很好的实例教材，是进行程序设计的良师益友。

齐全的运行库函数与低层支持函数，将给用户带来无穷无尽的新发现与高效率。

目 录

前言

第〇章 Turbo C++ 运行库概览.....1

第一章 CLIB 库函数源程序.....14

abs.cas	14
absread.cas	14
access.c	17
allocmem.cas	18
assert.c	19
atexit.c	20
atol.cas	20
bcd1.c	22
bcd2.cpp	27
bdos.c	27
bdosptr.cas	28
bioscom.cas	29
biosdisk.cas	30
biosequ.cas	32
bioskey.cas	33
biosprin.cas	34
brk.cas	35
bsearch.c	37
calloc.c	38
cgets.c	39
chdir.cas	39
chmod.c	41
chmoda.cas	42
chsize.cas	42
clearerr.c	45
clock.cas	45
close.c	46
closea.cas	47
closeall.c	47
clreol.c	48
clrscr.c	48

color.c	48
coreleft.cas	51
country.cas	52
cplx1.cpp	53
cplx2.cpp	54
cprintf.c	57
cputs.c	59
creat.cas	59
creata.cas	61
crtnit.cas	63
cscanf.c	66
ctime.c	67
ctrlbrkc	72
ctype.c	73
cursor.c	74
cvtfak.asm	74
dbp.cpp	75
delay.cas	76
divt.cas	77
doscnd.c	78
dosenv.c	79
dosex.cas	80
dostimu.cas	81
dup2.cas	83
eof.cas	84
exec.asm	87
execl.c	94
execl.c	95
execlp.c	95
execlpe.c	96
execv.c	96
execve.c	97
execvp.c	97
execvpe.c	97
exit.c	98
farheap.asm	98

fbrk.c	108	getenv.cas	163
fcalloc.cas	110	getfat.cas	165
fclose.c	111	getftime.cas	167
fcorelft.c	112	getpass.c	168
flush.c	112	getpsp.c	169
fgetpos.c	113	gets.c	169
fgets.c	114	getswit.c	170
feapchk.asm	114	getvect.cas	171
filebuf.cpp	121	getveri.cas	172
files.c	123	getw.c	172
files2.c	124	gexit.c	173
findfirs.cas	124	gfreemem.c	173
flength.cas	126	ggetmem.c	174
flushall.c	127	gotoxy.c	174
fmode.c	128	gptext.c	175
fnmerge.c	128	gregistr.c	176
fnsplit.c	129	harderr.cas	177
fopen.c	132	heaplen.c	179
format.cpp	135	h_ldiv.asm	179
fprintf.c	137	h_llsh.asm	181
fpstklen.c	137	h_lrsh.asm	182
fputs.c	138	h_lrsh.asm	183
fread.c	138	h_pada.asm	183
freemem.cas	140	h_padd.asm	185
fscanf.c	140	h_pina.asm	186
fseek.c	141	h_pshp.asm	187
fsetpos.c	143	h_scopy.asm	188
fstat.cas	143	h_spush.asm	189
fstream.cpp	145	inport.cas	190
ftime.c	152	insline.c	191
fwrite.c	153	int86.cas	191
f_lxmul.asm	154	intdos.cas	193
f_pcmp.asm	155	intr.cas	195
getc.cas	155	ioctl.cas	197
getchr.c	158	ioerror.cas	198
getch.cas	159	ios.cpp	201
getcurdi.cas	160	iostream.cpp	204
getcwd.c	161	is.cas	206
getdate.c	162	isatty.cas	208
getdta.cas	163	istream.cpp	208

istreamf.cpp	212	ostreamn.cpp	277
istreami.cpp	215	ostreamx.cpp	281
istreamn.cpp	218	ostrf.cpp	281
istreamx.cpp	223	outport.cas	281
istrf.cpp	224	parsfnn.cas	282
kbhit.cas	225	peek.c	283
keep.c	226	perror.c	284
labs.c	226	poke.c	285
ldivt.cas	227	printf.c	286
loadprog.c	228	pureerr.cpp	290
locale.c	230	putc.c	290
lock.cas	230	putch.c	293
lrotl.cas	232	putenv.cas	294
lrotr.cas	232	puts.c	297
lsearch.c	233	putw.c	297
lseek.cas	234	qsort.cas	298
ltoa.cas	235	rand.c	301
manip.cpp	237	randblk.cas	302
memccpy.cas	239	read.cas	303
memchr.cas	240	reada.cas	305
memcmp.cas	241	realcvt.asm	306
memcpy.cas	242	rename.cas	306
memcmp.c	243	rewind.c	307
memset.cas	243	rmdir.cas	307
mkdir.cas	244	rotl.cas	308
mktemp.c	245	rotr.cas	308
movedata.cas	246	scanf.c	309
movetext.c	247	scanner.cas	313
movmem.cas	247	scantod.asm	330
multbyte.c	249	scantol.cas	331
nearheap.asm	250	screen.c	336
newdel.cpp	257	scroll.c	338
nheapchk.asm	258	searchp.cas	340
n_lxmul.asm	263	segread.cas	343
n_pcmul.asm	264	setargv.asm	344
open.cas	264	setblock.cas	355
opena.cas	268	setbuf.c	355
ostream.cpp	269	setdate.cas	356
ostreamf.cpp	270	setdta.dta	357
ostreami.cpp	275	setenvp.asm	357

setenvp2.c	359	strlwr.c	403
setftime.cas	360	strmbfn.cpp	404
setjmp.cas	360	strncat.c	407
setmode.c	363	strncmp.cas	407
setupio.c	364	strncpy.cas	408
setvbuf.c	364	strncpy.cas	409
sigdata.c	365	strnset.c	410
signal.c	365	strpbrk.c	411
siosync.cpp	371	strrchr.c	411
sleep.c	372	strrev.c	412
sound.cas	372	strset.c	413
spawn.asm	374	strspn.c	413
spawnl.c	376	strstr.cas	414
spawnle.c	378	strstrea.cpp	415
spawnlp.c	378	strtok.c	420
spawnlpe.c	379	ctrtol.c	421
spawnv.c	379	strtoul.c	422
spawnve.c	380	strupr.c	423
spawnvp.c	380	strxfrm.c	424
spawnvpe.c	381	swab.c	425
sprintf.c	381	system.c	425
sscanf.c	382	tecl.c	427
stack.asm	384	textmode.c	427
stat.cas	384	timecvr.c	428
stdfile.cpp	387	tmpfile.c	430
stdlostr.cpp	387	tmpnam.c	430
stime.c	387	tolower.c	431
stklen.c	390	toupper.c	432
strcpy.c	390	tzset.cas	432
strcat.c	391	umask.c	436
strchr.cas	395	ungetc.cas	436
strcmp.cas	396	unlink.cas	437
strcoll.c	397	vapp.cpp	438
strcpy.cas	397	vappv.cpp	438
strcspn.c	398	vdel.cpp	439
strdup.c	398	vfprintf.c	440
streambf.cpp	399	vfscanf.c	441
strerror.c	400	vidinfo.c	441
stricmp.cas	401	vnew.cpp	442
strlen.cas	402	vnewv.cpp	443

vprinter.cas	444
vprintf.c	458
vram.cas	459
vscanf.c	460
wherexy.c	461
wild.asm	462
wildargs.asm	462
window.c	462
write.c	462
writea.cas	464
wscroll.c	465
xclose.c	465
xfclose.c	466
xfflush.c	466
zapctrlz.cas	466

第二章 MATH 库函数源程序.....469

acosasin.cas	469
atan.cas	472
atan2.cas	473
atof.c	475
ceil.cas	476
clear87.cas	477
cos.cas	478
cosh.cas	479
ctrl87.cas	480
difftime.c	482
efcvt.c	482
exp.cas	484
labs.cas	486
flags87.asm	486
floor.cas	487
fmod.cas	488
fperr.c	489
fpreset.c	491
frexp.cas	491
ftol.asm	492
gcvt.c	493
hugeval.c	494
hypot.cas	494

ldexp.cas	495
ldtrunc.cas	497
log.cas	499
log10.cas	500
matherr.c	501
modf.cas	503
poly.cas	504
pow.cas	505
pow10.cas	510
realevt.cas	513
scantod.cas	521
sin.cas	531
sinh.cas	533
sqrt.cas	534
stat87.cas	535
strtod.c	536
tan.cas	537
tanh.cas	539
xcvt.cas	540
_matherr.c	547

第三章 EMU 库函数源程序.....550

EMURULES.ASI	550
FPINIT.ASM	559

第四章 头文件和.ASI 文件.....572

ASMRULES.H	572
_DIR.H	577
_FHEAP.H	578
_GRAPH.H	578
_HEAP.H	584
_IO.H	584
_MATH.H	585
_PRINTF.H	588
_PROCESS.H	589
_SCANF.H	590
_STDIO.H	591
_VIDEO.H	592
_HEAP.INC	593
EMUVAR.SASI	593

EQUATES.ASI.....	596
RULES.ASI.....	597

第五章 修改库函数源程序.....	607
-------------------	-----

第六章 使用批处理文件来建库.....	608
CLIB.BAT.....	609

CLIBREPL.BAT.....	612
CLIBRLIB.BAT.....	616
MATH.BAT.....	618
MATHREPL.BAT.....	621
MATHRLIB.BAT.....	626
CLIBRCMP.BAT.....	628
MATHRCMP.BAT.....	630

第○章 Turbo C++运行库概览

本书整理编译了 Turbo C++ 运行库中所有函数的源程序以及详细的函数使用说明。它不仅全部提供了《Turbo C++ 参考手册》中提供的函数，而且还提供了实现这些函数的所有低层支持函数。这使得 Turbo C++ 运行库成为真正的开放式结构，用户可以从多方面高效地使用库函数。

有些库例程需要用到 Turbo C++ 的 .H 文件，另外，还需要用到 TLIB 和目标文件库。需要在 TURBOC.CFG 文件中指明 Turbo C++ 头(.H)文件的位置。

运行库被组织在几个目标文件库块中。这些库文件包含了通用的库例程和标准数学库例程。

FPINIT.ASM 是用于修改浮点初始化和终止进程的。CLIB* 和 MATH* 批文件不用此文件。FPINIT.ASM 用于建立两个 OBJ 文件，一个是用于模拟器的 FPINIT.OBJ，另一个是用于嵌入式 8087 指令的 FP87INIT.OBJ。

下面两个命令行将建立这两个文件：

```
TASM /MX /t /z fpinit
```

```
TASM /MX /t /z /D87 fpinit. fp87init
```

运行库中包含的大部分模块是用 Turbo C++ 编译器编译库源程序产生的目标文件。这些文件中有些包含嵌入式汇编语言代码，因此需要用于 TASM 2.0(或兼容)的宏汇编程序来编译这些源文件。最后，有少量模块是完全用汇编语言编写的。

下面是运行库中一些文件扩展名的含义：

.C - C源文件

.H - C头文件

.CAS - 包含嵌入式汇编程序的C源文件

.ASM - 汇编语言源文件

.CPP - C++源文件

.ASI - .ASM文件所需的包括文件

.RSP - TLIB响应文件。包含指定库的所有文件清单

本书结构

第一章：“CLIB 库函数源程序”

第二章：“MATH 库函数源程序”

第三章：“EMU 库函数源程序”

第四章：“头文件和.ASI 文件”

第五章：“修改库函数源程序”

第六章：“使用批处理文件来建库”

第一章到第三章列出了 Turbo C++ 软件包中提供的各个库函数，介绍它们的功能、用法、相关函数用法、原型所在头文件、说明信息、返回值、可移植性、参阅部分和示例，并给出实现该函数的源程序，具体格式如下：

函 数 名 功 能

用 法 本函数的用法

相关函数

用 法 本函数的相关函数的用法

原 型 在 本函数原型所在的头文件

说 明 本函数的使用信息说明

返 回 值 本函数的返回值

可移植性 本函数的适用系统

参 见 和本函数有关的其它函数名

源 程 序 实现本函数的源程序

为了方便读者理解运行库源程序,在源程序中还适当地加了一些注解。

本书以文件为单位进行组织,每个文件可能包含几个函数。文件与函数的包含关系如下:

— CLIB 库

文件	函数	说明
abs.cas	abs	绝对值
absread.cas	absread	绝对磁盘读
	abswrite	绝对磁盘写
access.c	access	确定文件的访问权限
allocmem.cas	allocmem	分配DOS内存段
assert.c	__assertfail	打印错误信息并退出
atexit.c	atexit	寄存器终止函数
atol.cas	atol	将字符串转换为长整数
	atoi	将字符串转换为整数
bcd1.c		long double / binary 代码转换库
bcd2.cpp		
bdos.c	bdos	MS-DOS 系统调用
bdosptr.cas	bdosptr	MS-DOS 系统调用
bioscom.cas	bioscom	通信I/O
biosdisk.cas	biosdisk	硬盘/软盘I/O
biosequ.cas	biosequip	设备检查
	biosmemory	返回内存大小
	biostime	返回时间值
bioskey.cas	bioskey	键盘接口
biosprin.cas	biosprint	打印机I/O
brk.cas	brk	改变数据段空间分配
	sbrk	改变数据段空间分配
	__brk	改变近堆上分配的数据段空间
	__sbrk	改变近堆上分配的数据段空间
bsearch.c	bsearch	二分查找
calloc.c	calloc	分配主内存
cgets.c	cgets	从控制台读取字符串
chdir.cas	chdir	改变工作目录
	getdisk	取当前驱动器号
	setdisk	设置当前驱动器
chmod.c	chmod	改变文件的访问模式
chmoda.cas	_chmod	改变文件的访问模式
chsize.cas	chsize	改变文件长度
clearerr.c	clearerr	错误标志复位
clock.cas	clock	ANSI标准;返回自本程序开始以来已经过的时间数。
close.c	close	关闭文件句柄
closea.cas	_close	关闭文件句柄
closeall.c	fcloseall	关闭打开的流

clreol.c	clreol	在窗口中删到行尾
clrscr.c	clrscr	清除文本模式窗口
color.c	textcolor	在文本模式设置新的字符颜色
	textbackground	选择新的背景颜色
	textattr	设置正文属性
	highvideo	选择高亮度文本字符
	lowvideo	选择低亮度文本字符
	normvideo	选择正常亮度文本字符
coreleft.cas	coreleft	返回未使用内存的长度
country.cas	country	返回与国家有关的信息
cplx1.cpp		C++复杂库例程
cplx2.cpp		C++复杂库例程
cprintf.c	__cputn	字符输出函数
	cprintf	将格式化输出送到控制台
cputs.c	cputs	向控制台写一个字符串
creat.cas	dosCreat	建立一个文件
	creat	建立一个新文件或重写已存在的文件
creata.cas	CreateFile	建立一个文件
	_creat	建立一个新文件或重写一个已存在的文件
	creattemp	建立一个新文件
	creatnew	建立一个新文件
crtinit.cas	system	显示系统信息
	egainstalled	检测EGA卡
	_VideoInt	产生视频中断
	_c0crtinit	crt初始化
	_crtinit	视频变量初始化
	turboCrt	
cscanf.c	cscanf	从控制台执行格式化输入
ctime.c	comtime	将长型的时间日期转换为结构
	asctime	将日期和时间转换ASCII形式
	ctime	将日期和时间转换为字符串
	gmtime	将日期和时间转换为格林威治时间
	localtime	将日期和时间转换为结构
	mktime	日期和时间结构的普通化
	strftime	将日期和时间结构转换为字符串
ctrlbrk.c	hentry	调用用户Ctrl-break处理程序的句柄。
	ctrlbrk	设置control-break句柄
ctype.c		
cursor.c	setcursortype	设置正文光标类型
CVTFAK.ASM		当没有浮点数时退出
dbp.cpp		为流缓冲区高度输出
delay.cas	delay	等待指定的时间
divt.cas	div	整数除法
doscnd.c	__DOScmd	准备Spawn/Exec的命令行
dosenv.c	__DOSenv	准备Spawn/Exec环境
dosex.cas	dosexterr	取扩展错误
ostimu.cas	__DOStimeToU	将DOS时间转换为UNIX时间格式

dup2.cas	dup	复制文件句柄
	dup2	复制文件句柄
eof.cas	eof	检测文件结束标志
EXEC.ASM		以覆盖方式执行一个程序
execl.c	execl	装入并运行一个程序
execle.c	execle	装入并运行一个程序
execlp.c	execlp	装入并运行一个程序
execlepe.c	execlepe	装入并运行一个程序
execv.c	execv	装入并运行一个程序
execve.c	execve	装入并运行一个程序
execvp.c	execvp	装入并运行一个程序
execvpe.c	execvpe	装入并运行一个程序
exit.c	exit	终止程序
FARHEAP.ASM		
fbrk.c	normalize	普通化一个指针
	_brk	改变数据段空间的分配
	_sbrk	改变数据段空间的分配
fcalloc.cas	fcalloc	从远堆中改变内存的分配
	lsetmem	向内存分配一个值
fclose.c	fclose	关闭一个流
fcoreleft.c	fcoreleft	返回远堆中未使用存储区大小
fflush.c	fflush	刷新一个流
fgetpos.c	fgetpos	取当前文件指针
fgets.c	fgets	从流中取一个字符中
FEAPCHK.ASM		
filebuf.cpp		处理文件缓冲区的C++流I/O函数
files.c		
files2.c		
findfirs.cas	findfirst	查找磁盘目录
	findnext	查找与findfirst匹配的文件
flength.cas	filelength	以字节为单位取文件长度
flushall.c	flushall	清除所缓冲区
fmode.c		
fnmerge.c	fnmerge	构建新的文件名
fnspiit.c	CopyIt	串拷贝
	DotFound	检查指定的目录名
	fnsplit	将完整的路径名分开
fopen.c	fopen	打开一个流
	fdopen	将一个流与一个文件句柄相联
	freopen	替换一个流
	_openfp	打开一个文件
	getfp	取文件指针
	CheckOpenType	返回FILE标志位值,如类型字符串无效则返回零
format.cpp		C++输出转换函数
fprintf.c	fprintf	将格式化输出送到一个流中
fpstklen.c		
fputs.c	fputs	将一个串送到一个流中

fread.c	fread	从流中读取数据
	_fgetn	从流中读取数据
freemem.cas	freemem	释放上次分配的DOS内存块
fscanf.c	fscanf	从一个流执行格式化输入
fseek.c	fseek	在流中重设文件指针
	ftell	返回当前文件指针
	Displacement	内部函数
fsetpos.c	fsetpos	在流中设置文件指针
fstat.cas	fstat	取打开的文件信息
fstream.cpp		filebuf分类和fstream成员函数
ftime.c	ftime	取当前时间并将它存到timeb结构中
fwrite.c	fwrite	向一个流写入数据
F_LXMUL.ASM		长整型数乘法
F_PCMP.ASM		长型指针比较
getc.cas	FlushOutputStreams	刷新输出流
	_fill	填充预读缓冲区
	fgetc	从流中取字符
	_fgetc	从流中取字符
	fgetchar	从流中取字符
getcbkr.c	getcbkr	取control-break状态设置
	setcbkr	设置control-break状态
getch.cas	getch	从控制台取字符
	getche	从控制台取字符
	ungetch	将字符回送给控制台
getcurdi.cas	getcurdir	取指定驱动器的当前目录
getcwd.c	getcwd	取当前工作目录
getdate.c	getdate	取MS-DOS日期
	gettime	取MS-DOS时间
getdta.cas	getdta	取磁盘传送区地址
getenv.cas	getenv	取环境中
getfat.cas	getfatd	取文件分配表信息
	getfat	取文件分配表信息
	getdfree	取磁盘空闲空间大小
getftime.cas	getftime	取文件日期和时间
getpass.c	_KbdFlush	刷新键盘缓冲区
	getpass	读取口令
getpsp.c	getpsp	取程序段前缀
gets.c	gets	从流中取串
getswit.c	getswitchar	取MS-DOS开关字符
	setswitchar	设置MS-DOS开关字符
getvect.cas	getvect	取中断向量项
	setvect	设置中断向量项
getveri.cas	getverify	取校验状态
	setverify	设校验状态
getw.c	getw	从流中取一个单词
gexit.c	_graphexit	退出程序(对图形库)
gfreemem.c	_graphfreemem	为图形库释放内存