

# **SYSTEM V/68 Release 3 Programmer's Reference Manual**

**SYSTEM V/68 RELEASE 3  
PROGRAMMER'S REFERENCE MANUAL**

Part Number MU43814PR/D2

Version 2

SYSTEM V/68™ is a trademark of Motorola Inc. Dataphone®, Teletype®, and UNIX® are registered trademarks of AT&T. Diablo® is a registered trademark of Xerox. HP™ is a trademark of Hewlett-Packard. PDP™, VAX™, and DEC™ are trademarks of Digital Equipment Corporation. TEKTRONIX® is a registered trademark of Tektronix, Inc. TermiNet™ is a trademark of General Electric. Versatec® is a registered trademark of Versatec Corporation. C-68000™ is a trademark of Green Hills Software, Inc.

SYSTEM V/68 Release 3 is based on the AT&T UNIX System V, Release 3.0. The software described herein is furnished under a licensed agreement and may be used only in accordance with the terms of the agreement.

Copyright © 1986, 1987 Motorola Inc. All rights reserved. No part of this manual may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, without the prior written permission of Motorola Inc.

Portions of this document are reprinted from  
copyrighted documents by permission of AT&T, 1986.



Part Number: MU43814PR/A1

## DOCUMENT SUPPLEMENT

Date: 06/01/88

The attached pages constitute the first revision to the *SYSTEM V/68 Release 3 Programmer's Reference Manual* (Part Number MU43814PR/D2). This supplement is required to support the SYSTEM V/68 assembler (*as*) and the SYSTEM/V68 C compiler on the MC68030 microprocessor. It is also required for the Board Software Extension packages (BS82, BS83) on SYSTEM V/68 Release 3, Version 4. Additionally, corrections are included for *dial* and *dfile*.

Please replace and add pages according to the following table:

Replace old	With new	Add new
ar(1) 3/as(1) 1 as(1) 2/blank cc(1) 4/5 gcc(1) 1/2 sysfs(2) 2/sysm68k(2) 1 sysm68k(2) 2-4/blank dial(3c) 1/2 dfile(4) 5/dir(4) 1	ar(1) 3/as(1) 1 as(1) 2/3 cc(1) 4/5 gcc(1) 1/2 sysfs(2) 2/sysm68k(2) 1 sysm68k(2) 2-4/5 dial(3c) 1/2 dfile(4) 5/6	sysm68k(2) 6/7 dir(4) 1/dirent(4) 1

- A vertical bar in the outside margin of a revised page indicates a text change or addition.
- A double asterisk (\*\*) in the same position indicates a text deletion.

## PREFACE

The *Programmer's Reference Manual* (Part Number MU43814PR/D2) describes the commands, function calls, file formats and miscellaneous facilities of interest primarily to programmers on the Motorola VME-based computer.

While reasonable efforts have been made to assure the accuracy of this document, Motorola assumes no liability resulting from any omissions in this document or from the use of the information obtained therein. Motorola reserves the right to revise this document and to make changes from time to time in its content without being obligated to notify any person of such revision or changes.

# CONTENTS

1. INTRODUCTION .....	1-1
-----------------------	-----

## 1. Commands

intro(1) .....	introduction to programming commands
admin(1) .....	create and administer SCCS files
ar(1) .....	archive and library maintainer for portable archives
as(1) .....	common assembler
cb(1) .....	C program beautifier
cc(1) .....	C compiler
cdc(1) .....	change the delta commentary of an SCCS delta
cflow(1) .....	generate C flowgraph
comb(1) .....	combine SCCS deltas
cpp(1) .....	the C language preprocessor
cprs(1) .....	compress a common object file
create(1) .....	create master release media utility, R3.1
ctrace(1) .....	C program debugger
cxref(1) .....	generate C program cross-reference
delta(1) .....	make a delta (change) to an SCCS file
dis(1) .....	object code disassembler
dump(1) .....	dump selected parts of an object file
gcc(1) .....	C-68000 compiler
gcc(1M) .....	create a front-end to the cc command
get(1) .....	get a version of an SCCS file
infocmp(1M) .....	compare or print out terminfo descriptions
install(1M) .....	install commands
ld(1) .....	link editor for common object files
lex(1) .....	generate programs for simple lexical tasks
lint(1) .....	a C program checker
list(1) .....	produce C source listing from a common object file
lorder(1) .....	find ordering relation for an object library
m4(1) .....	macro processor
make(1) .....	maintain, update, and regenerate groups of programs
mcs(1) .....	manipulate the object file comment section
mkmenus(1) .....	extracts menus from labels stored in command shell scripts
nm(1) .....	print name list of common object file
prof(1) .....	display profile data
prs(1) .....	print an SCCS file

regcmp(1) ..... regular expression compile  
 rmdel(1) ..... remove a delta from an SCCS file  
 sact(1) ..... print current SCCS file editing activity  
 sccsdiff(1) ..... compare two versions of an SCCS file  
 sdb(1) ..... symbolic debugger  
 size(1) ..... print section sizes in bytes of common object files  
 strip(1) ..... strip symbol and line number information from a common object file  
 tic(1M) ..... terminfo compiler  
 tsort(1) ..... topological sort  
 unget(1) ..... undo a previous get of an SCCS file  
 val(1) ..... validate SCCS file  
 vc(1) ..... version control  
 what(1) ..... identify SCCS files  
 yacc(1) ..... yet another compiler-compiler

## 2. System Calls

intro(2) ..... introduction to system calls and error numbers  
 access(2) ..... determine accessibility of a file  
 acct(2) ..... enable or disable process accounting  
 advfs(2) ..... advertise a directory for remote access  
 alarm(2) ..... set a process alarm clock  
 brk(2) ..... change data segment space allocation  
 chdir(2) ..... change working directory  
 chmod(2) ..... change mode of file  
 chown(2) ..... change owner and group of a file  
 chroot(2) ..... change root directory  
 close(2) ..... close a file descriptor  
 creat(2) ..... create a new file or rewrite an existing one  
 dup(2) ..... duplicate an open file descriptor  
 exec(2) ..... execute a file  
 exit(2) ..... terminate process  
 fcntl(2) ..... file control  
 fork(2) ..... create a new process  
 getdents(2) ..... read directory entries and put in a file  
 getmsg(2) ..... get next message off a stream  
 getpid(2) ..... get process, process group, and parent process IDs  
 getuid(2) ..... get real user, effective user, real group, and effective group IDs  
 ioctl(2) ..... control device  
 kill(2) ..... send a signal to a process or a group of processes  
 link(2) ..... link to a file  
 lseek(2) ..... move read/write file pointer  
 mkdir(2) ..... make a directory  
 mknod(2) ..... make a directory, or a special or ordinary file

mount(2) ..... mount a file system  
 msgctl(2) ..... message control operations  
 msgget(2) ..... get message queue  
 msgop(2) ..... message operations  
 nice(2) ..... change priority of a process  
 open(2) ..... open for reading or writing  
 pause(2) ..... suspend process until signal  
 pipe(2) ..... create an interprocess channel  
 plock(2) ..... lock process, text, or data in memory  
 poll(2) ..... STREAMS input/output multiplexing  
 profil(2) ..... execution time profile  
 ptrace(2) ..... process trace  
 putmsg(2) ..... send a message on a stream  
 read(2) ..... read from file  
 rfstart(2) ..... start the Remote File Sharing environment  
 rfstop(2) ..... stop the Remote File Sharing environment  
 rmdir(2) ..... remove a directory  
 rmount(2) ..... mount a remote directory  
 rumount(2) ..... unmount a remote directory  
 semctl(2) ..... semaphore control operations  
 semget(2) ..... get set of semaphores  
 semop(2) ..... semaphore operations  
 setpgrp(2) ..... set process group ID  
 setuid(2) ..... set user and group IDs  
 shmctl(2) ..... shared memory control operations  
 shmget(2) ..... get shared memory segment identifier  
 shmop(2) ..... shared memory operations  
 signal(2) ..... specify what to do upon receipt of a signal  
 sigset(2) ..... signal management  
 stat(2) ..... get file status  
 statf(2) ..... get file status  
 statfs(2) ..... get file system information  
 stime(2) ..... set time  
 sync(2) ..... update super block  
 sysfs(2) ..... get file system type information  
 sysm68k(2) ..... machine specific functions  
 time(2) ..... get time  
 times(2) ..... get process and child process times  
 uadmin(2) ..... administrative control  
 ulimit(2) ..... get and set user limits  
 umask(2) ..... set and get file creation mask  
 umount(2) ..... unmount a file system  
 unadv(2) ..... unadvertise a directory



uname(2) ..... get name of current SYSTEM V/68 system  
 unlink(2) ..... remove directory entry  
 ustat(2) ..... get file system statistics  
 utime(2) ..... set file access and modification times  
 wait(2) ..... wait for child process to stop or terminate  
 write(2) ..... write on a file

### 3. Subroutines

intro(3) ..... introduction to functions and libraries  
 a64l(3C) ..... convert between long integer and base-64 ASCII string  
 abort(3C) ..... generate an IOT fault  
 abs(3C) ..... return integer absolute value  
 access881(3C) ..... provide access to floating point chip  
 bsearch(3C) ..... binary search a sorted table  
 clock(3C) ..... report CPU time used  
 conv(3C) ..... translate characters  
 crypt(3C) ..... generate hashing encryption  
 ctermid(3S) ..... generate file name for terminal  
 ctime(3C) ..... convert date and time to string  
 ctype(3C) ..... classify characters  
 cuserid(3S) ..... get character login name of the user  
 dial(3C) ..... establish an out-going terminal line connection  
 drand48(3C) ..... generate uniformly distributed pseudo-random numbers  
 dup2(3C) ..... duplicate an open file descriptor  
 ecvt(3C) ..... convert floating-point number to string  
 end(3C) ..... last locations in program  
 fclose(3S) ..... close or flush a stream  
 ferror(3S) ..... stream status inquiries  
 fopen(3S) ..... open a stream  
 fread(3S) ..... binary input/output  
 frexp(3C) ..... manipulate parts of floating-point numbers  
 fseek(3S) ..... reposition a file pointer in a stream  
 ftw(3C) ..... walk a file tree  
 getc(3S) ..... get character or word from a stream  
 getcwd(3C) ..... get path-name of current working directory  
 getenv(3C) ..... return value for environment name  
 getgrent(3C) ..... get group file entry  
 getlogin(3C) ..... get login name  
 getopt(3C) ..... get option letter from argument vector  
 getpass(3C) ..... read a password  
 getpw(3C) ..... get name from UID  
 getpwent(3C) ..... get password file entry  
 gets(3S) ..... get a string from a stream

getut(3C) ..... access utmp file entry  
 hsearch(3C) ..... manage hash search tables  
 l3tol(3C) ..... convert between 3-byte integers and long integers  
 lockf(3C) ..... record locking on files  
 lsearch(3C) ..... linear search and update  
 malloc(3C) ..... main memory allocator  
 memory(3C) ..... memory operations  
 mktemp(3C) ..... make a unique file name  
 monitor(3C) ..... prepare execution profile  
 nlist(3C) ..... get entries from name list  
 perror(3C) ..... system error messages  
 popen(3S) ..... initiate pipe to/from a process  
 printf(3S) ..... print formatted output  
 putc(3S) ..... put character or word on a stream  
 putenv(3C) ..... change or add value to environment  
 putpwent(3C) ..... write password file entry  
 puts(3S) ..... put a string on a stream  
 qsort(3C) ..... quicker sort  
 rand(3C) ..... simple random-number generator  
 scanf(3S) ..... convert formatted input  
 setbuf(3S) ..... assign buffering to a stream  
 setjmp(3C) ..... non-local goto  
 sleep(3C) ..... suspend execution for interval  
 ssignal(3C) ..... software signals  
 stdio(3S) ..... standard buffered input/output package  
 stdipc(3C) ..... standard interprocess communication package  
 string(3C) ..... string operations  
 strtod(3C) ..... convert string to double-precision number  
 strtol(3C) ..... convert string to integer  
 swab(3C) ..... swap bytes  
 system(3S) ..... issue a shell command  
 tmpfile(3S) ..... create a temporary file  
 tmpnam(3S) ..... create a name for a temporary file  
 tsearch(3C) ..... manage binary search trees  
 ttyname(3C) ..... find name of a terminal  
 ttyslot(3C) ..... find the slot in the utmp file of the current user  
 ungetc(3S) ..... push character back into input stream  
 vprintf(3S) ..... print formatted output of a varargs argument list  
**bessel(3M) ..... Bessel functions**  
 erf(3M) ..... error function and complementary error function  
 exp(3M) ..... exponential, logarithm, power, square root functions  
 floor(3M) ..... floor, ceiling, remainder, absolute value functions  
 gamma(3M) ..... log gamma function

hypot(3M) ..... Euclidean distance function  
 math881(3M) ..... floating point math functions  
 matherr(3M) ..... error-handling function  
 sinh(3M) ..... hyperbolic functions  
 trig(3M) ..... trigonometric functions  
 t\_accept(3N) ..... accept a connect request  
 t\_alloc(3N) ..... allocate a library structure  
 t\_bind(3N) ..... bind an address to a transport endpoint  
 t\_close(3N) ..... close a transport endpoint  
 t\_connect(3N) ..... establish a connection with another transport user  
 t\_error(3N) ..... produce error message  
 t\_free(3N) ..... free a library structure  
 t\_getinfo(3N) ..... get protocol-specific service information  
 t\_getstate(3N) ..... get the current state  
 t\_listen(3N) ..... listen for a connect request  
 t\_look(3N) ..... look at the current event on a transport endpoint  
 t\_open(3N) ..... establish a transport endpoint  
 t\_optmgmt(3N) ..... manage options for a transport endpoint  
 t\_rcv(3N) ..... receive data or expedited data sent over a connection  
 t\_rcvconnect(3N) ..... receive the confirmation from a connect request  
 t\_rcvdis(3N) ..... retrieve information from disconnect  
 t\_rcvrel(3N) ..... acknowledge receipt of an orderly release indication  
 t\_rcvudata(3N) ..... receive a data unit  
 t\_rcvuderr(3N) ..... receive a unit data error indication  
 t\_snd(3N) ..... send data or expedited data over a connection  
 t\_snddis(3N) ..... send user-initiated disconnect request  
 t\_sndrel(3N) ..... initiate an orderly release  
 t\_sndudata(3N) ..... send a data unit  
 t\_sync(3N) ..... synchronize transport library  
 t\_unbind(3N) ..... disable a transport endpoint  
 assert(3X) ..... verify program assertion  
 crypt(3X) ..... password and file encryption functions  
 curses(3X) ..... terminal screen handling and optimization package  
 directory(3X) ..... directory operations  
 getnum(3X) ..... calculate an integer value from a string  
 getperms(3X) ..... read the *permissions* file  
 ldahread(3X) ..... read the archive header of a member of an archive file  
 ldclose(3X) ..... close a common object file  
 ldhread(3X) ..... read the file header of a common object file  
 ldgetname(3X) ..... retrieve symbol name for common object file symbol table entry  
 ldhread(3X) ..... manipulate line number entries of a common object file function  
 ldlseek(3X) ..... seek to line number entries of a section of a common object file  
 ldohseek(3X) ..... seek to the optional file header of a common object file

ldopen(3X) ..... open a common object file for reading  
 ldrseek(3X) ..... seek to relocation entries of a section of a common object file  
 ldshread(3X) ..... read an indexed/named section header of a common object file  
 ldsseek(3X) ..... seek to an indexed/named section of a common object file  
 ldtbindex(3X) ..... compute the index of a symbol table entry of a common object file  
 ldtbread(3X) ..... read an indexed symbol table entry of a common object file  
 ldtbseek(3X) ..... seek to the symbol table of a common object file  
 logname(3X) ..... return login name of user  
 malloc(3X) ..... fast main memory allocator  
 regcmp(3X) ..... compile and execute regular expression  
 sputl(3X) ..... access long integer data in a machine-independent fashion

#### 4. File Formats

intro(4) ..... introduction to file formats  
 a.out(4) ..... common assembler and link editor output  
 acct(4) ..... per-process accounting file format  
 ar(4) ..... common archive file format  
 checklist(4) ..... list of file systems processed by fsck and ncheck  
 core(4) ..... format of core image file  
 cpio(4) ..... format of cpio archive  
 dfile(4) ..... device information file  
 dir(4) ..... format of directories  
 dirent(4) ..... file system independent directory entry  
 errfile(4) ..... error-log file format  
 filehdr(4) ..... file header for common object files  
 fs(4) ..... format of system volume  
 fspec(4) ..... format specification in text files  
 fstab(4) ..... file-system-table  
 gettydefs(4) ..... speed and terminal settings used by getty  
 group(4) ..... group file  
 host(4) ..... system host name  
 hosts(4) ..... host name data base  
 hosts.equiv(4) ..... names of hosts with "equivalent" users  
 inittab(4) ..... script for the init process  
 inode(4) ..... format of an i-node  
 issue(4) ..... issue identification file  
 ldfcn(4) ..... common object file access routines  
 limits(4) ..... file header for implementation-specific constants  
 linenum(4) ..... line number entries in a common object file  
 master(4) ..... master configuration database  
 mnttab(4) ..... mounted file system table  
 passwd(4) ..... password file  
 perms(4) ..... permissions file used by the value-added disk access utilities

profile(4)	setting up an environment at login time
protocols(4)	protocol name data base
reloc(4)	relocation information for a common object file
rfmaster(4)	Remote File Sharing name server master file
rhosts(4)	user-specified file of equivalent hosts and users
sccsfile(4)	format of SCCS file
scrhdr(4)	section header for a common object file
scr_dump(4)	format of curses screen image file
services(4)	service name data base
syms(4)	common object file symbol table format
term(4)	format of compiled term file
terminfo(4)	terminal capability data base
timezone(4)	set default system time zone
unistd(4)	file header for symbolic constants
utmp(4)	utmp and wtmp entry formats

## 5. Miscellaneous Facilities

intro(5)	introduction to miscellany
ascii(5)	map of ASCII character set
environ(5)	user environment
fcntl(5)	file control options
math(5)	math functions and constants
prof(5)	profile within a function
regexp(5)	regular expression compile and match routines
stat(5)	data returned by stat system call
term(5)	conventional names for terminals
types(5)	primitive system data types
values(5)	machine-dependent values
varargs(5)	handle variable argument list