¡¡¡¡¡¡¡ texinfo.tex ======= ¿¿¿¿¿¿¿ 1.1.1.2
¡¡¡¡¡¡¡ texinfo.tex

(Se t bu s a t comme

Table of Contents

Summary of GDB 1
Getting Started 3
Sample GDB Session 11
Getting In and Out of GDB 11

5.1.8 Breakpoi t me us 44
5.1.9 "Ca ot i sert breakpoi ts" 44
5.2 Co ti ui a t steppi 45
5.3 Si als 48
5.4 Sto

12.4.3.5 Deviatio s from sta ■tar■ Mo■ula-2 .. 12■0
12.4.3.6 Mo■ula-2 type a ■ ra e checks ..... 12■0
12.4.3.7 The scope operators :: a ■ ..... 12■0
12.4.3.8 ■DB a ■ Mo■ula-2 ..... 12■1
12.5 ■ supporte■ la ua es ..... 12■1

1 E■amining th■ Sym■ol Ta■l■ 12

14 Alt■ring E■■cuti ii no ■ ■bb

18 Configuration-Specific Information 155
18.1 Native 157
18.1.1 HP-UX 157

18.

19 ontrolling G B 177
19.1 Prompt 177
19.2 Comma t eti ti 177
19.3 Comma t history 177
19.4 Scree si e 179
19.5 Numbers 179
19.6 Co fi uri the curre t I 180
19.7 ptio al ar i s a t messa es 181
19.8 ptio al messa es about i ter al happe i s 182

20 anned Sequences of ommands 185
20.1 ser-defi ed comma ts 185
20.2 ser-defi ed comma t hooks 186
20.3 Comma t files 187
20.4 Comma ts for co trolled output 188

21 ommand Interpreters 191

22 G B Text User Interface 193
22.1 TI overvie 193
22.2 TI Key Bi di s 194
22.

soft are packa e toes

a ecteali ■ith ◻fuses, please t y ◻the pu◻lishe s If you' e ot su e■hethe a proposect lice se is f ee, ■ ite to licensing@gnu.org.

You ca e cou a e comme ◻ial pu◻lishe s to sell mo e f ee, copylefte◻ ma uals a ◻ tuto ials ◻y ◻uyi them, a ◻ pa ticula ◻y ◻y ◻uyi copies f om the pu◻lishe s that pai◻ fo thei ■ iti o fo maj◻ improveme ts. Mea ■hile, t y to avoi◻ ◻uyi o -f ee ◻ocume tatio at all. Check the ◻ist i◻utio te ms of a ma ual ◻efo e you ◻uy it, a ◻ i sist that ■hoeve seeks your ◻usi ess must h■ ◻ u

Hikichi a t Tomoyuki

1 a ple [illegible]D e[illegible]

You ca use this ma ual at your leisure to rea[illegible] all a[illegible]out [illegible]DB. Ho[illegible]ever, a ha [illegible]ful of comma [illegible]s are e ou h to et starte[illegible] usi the [illegible]ebu er. This chapter illustrates t

```
(gdb) run
Starting progra : /work/Editorial/gdb/gnu/ 4/ 4
define(foo,0000)

foo
0000
```

To tri er the breakpoi t, e call changequote. DB suspe ts executio of m4, isplayi i formatio about the co text here it stops.

```
changequote(<ULITE>,<UNULITE>)

Breakpoint 1,  4_hangequote (arg=3, argv=0x33a70)
    at builtin.:575
575         if (bad_arg(TOKEN_DATA_TE T(argv[0]),arg,1,3))
```

No e use the comma t n (next) to a va ce executio to the ext li e of t

Success! The e quotes o ork just as ell as the fault o es. The problem seems to have bee just the t o typos in the ro le ths. We allo m4 exit by i The ro

2.1.1 Choosing files

Whe DB starts, it reat s a y ar ume ts other tha optio s as specifyi a executable file a t core file (or process ID). This is the s

Chapt

`-tt evice`
`-t evice`
Ru usi *devi*

[illegible]D [illegible]o[illegible] and[illegible]s

You ca[illegible] ab[illegible]breviate a [illegible]DB comma[illegible]d to the first fe[illegible] letters of the comma[illegible]d [illegible]ame, if that ab[illegible]breviatio[illegible] is u[illegible] ambi[illegible]uous; a[illegible]d you ca[illegible] repeat certai[illegible] [illegible]DB comma[illegible]ds by typi[illegible] just ⟨RET⟩.

```
(gdb) info bre ⟨TAB⟩
```

■DB fills i the rest of the ■ort ‘brea

```
(gdb) b bub ⟨TAB⟩
```

GDB lters your input line to the following, nd rings bell:

```
show -- Generi  o  and for showing things
 about the debugger

Type "help" followed by  o  and na e for full
do u entation.
Co  and na e abbreviations are allowed if una biguous.
(gdb)
```

help *c mm n*

With a comma t ame as help ar ume t, DB t isplays a short para raph o ho to use that comma t.

apropos *gs*

The apropos *gs* comma t searches throu h all of the DB comma ts, a t their t ocume tatio , for the re ular expressio speci

To display all the settable parameters and their

4.6 Your program's input and output

By default, the pro ram you ru u ter DB oes i put a t output to the same termi al that DB uses. DB s itches the termi al to its o termi al mo es to i teract ith you, but it

The first thi ■DB ◻oes after arra i to ◻eBu the specifie◻ process is to stop it. You ca exami e a ◻ mo◻ify a attache◻ process ■ith all the ■DB comma ◻s that are or◻i arily availa◻le ■he you start processes ■ith `run`. You ca i sert ◻reakpoi

- threat-specific breakpoi ts

  Wurning: These facil

```
[New thread 2 (syste  thread 26594)]
```

■he ■DB otices a e■ threa t.

`Info threads`

Display a summary of all threa ts curre tly i your pro ram. ■DB ■isplays for each threa t (i this or er):

1. the threa t umber assi e t by ■DB
2. the tar et system's threa t ite ti

4.10 Debugging programs with multiple processes

most systems, DB has o special support for debu i pro rams hich create ad ditio al processes usi the `fork` fu ctio . Whe a pro ram forks, DB ill co ti ue to debu the pare t process a d the child process ill ru u impeded. If you have set a breakpoi tyicode hich the child the executes, the child ill et a `SIGTRAP` si al hich (u less it catches the si al) ill cause it to termi

t topping and ontinuing

The pri cipal purposes of usi a tebu er are so that you ca stop your pro ram before it termi ates; or so that, if your pro ram ru s i to trouble, you ca i vesti ate a t fi t out hy.

I site DB, your pro ram may stop for a y of several reaso s, such as a si al, a break-poi t, or reachi a e l a

Some ■DB comma t̄s accept a ra e of ƀreakpoi ts o ■hich to operate. ■ ƀreakpoi t ra e is either a si le ƀreakpoi t umƀer, like ‘5’, or t■o such umƀers, i

The sy tax of the re ular expressio is the sta tart o e use with tools l

5.1.2 etting

E breakpoi t, atchpoi t, or catchpoi t ca have a y of four ti ere t states of e able-me t:

- E ablet. The breakpoi t stops your pro ram. E breakpoi t set ith the `break` comma t starts out i this state.
- Disablet. The breakpoi t has o e ect o your pro ram.
- E ablet o ce. The breakpoi t stops your pro ram, but the becomes tisablet.
- E ablet for teletio .

Chapter 5: Stoppi a

Chapter 5GStoppi a Co ti ui

```
until location
u location
```

Co ti ue ru i your pro ram u til either t

The default is set to `nostop`, `noprint`, `pass` for o -erro eous si als such as `SIGALRM`, `SIGWI`

6 a iniпg t e tack

Whe y

Most pro rams have a sta tart user e try poi t—a place here system libraries a t startup cote tra sitio i to user cote. or C this is main. Whe DB fi

`up` *n* Move *n* frames up the stack. or positive umbers *n*, this a tva ces to a

- the pro ram cou ter savet i it (the attress of executio i the caller frame)
- hich re isters ere savet i the frame

The verbos

7 a ining

edit *filen me*:*numbe*

Specifies li e *number* i the source file *filen me*.

edit *filen me*:*functi n*

Specifies the li e that be i s the bo y of the fu ctio *fun ti*

C

`::` '::' allo s you to specify a variable i terms of the file or fu ctio here it is efi e. See Sectio 8.2 [Pro ram variables], pa e 66.

`{ty e}` Refers to a object of type *ty*e store at a ress *ddr* i myfu *ddr* may be a y expressio hose va

You may see problem whe you are steppi by machi e i structio s. This is because, o most machi es, it takes more o e i structio to set up a stack frame (i cludi local variable defi itio s); if you are steppi by machi e i structio s, variables may appear to have [illegible]

```
(gdb) p/■ (short[])0■
```

or example, to pri t the pro ram cou ter i hex

pa e 65, for more i formatio o expressio s. The efault for *addr* is usually just after the last address exami ed—bu

specificatio is—it uses `x` if

else here—here there is o variable last_char—the display i

■et pri■nt null-■top

Cause ■DB to stop pri ti the characters of a■ray ■he the first NULL is e cou tere■. This is useful ■he o

Chapt

```
set print object
set print object on
```

Whe displayi a poi ter to a object, ide tify the *actual* (derived) type of the object rather tha the *declared* type, usi the virtual fu ctio t

`show convenience`

Pri t a list of co ve ie ce variables use so far, a their alues. bbreviate `show conv`.

e of the ays to a co ve ie ce iable is as a cou ter to be i creme te or a poi ter to be a va ce . or example, to pri t a fiel from successive eleme ts of a array of

Chapt

character sets for you. The character set [redacted]DB uses [redacted]e call the *h t h[redacted]r[redacted] ter et*; the o e the i ferior pro ram uses [redacted]e call the *t[redacted]rget h[redacted]r[redacted] ter et*.

or example, if you are ru i [redacted]DB o a [redacted]NU/Li ux system, [redacted]hich uses the IS Lati 1 character set, but you are usi [redacted]DB's remote protocol (see Sectio 16.4 [Remote], pa e 148

EB DI -US
IBM1047 Varia ts of the EB■DI■ character set,

Chapter 8: E

efi itio i troduced by this comma d is i scope i every expressio evaluated i DB, u til it is removed ith the `macro undef` comma d, described belo . The efi itio overrides all efi itio s for *m r* prese t i the pro ram bei debu ed, as ells a y previous user-supplied efi itio .

`macro undef` *m* c

(This command is not yet implemented.) Remove a y user-supplied efi itio for the macro amed *m r* . This comma d o ly a ects efi itio s provided ith the `macro define` emov efi itio -

i th abov comma s i actio . h a

g e

10 Ɔra

`tstop` This comma t takes o ar ume ts. It e ts the trace experime t, a t stops collecti ata.

Note:

```
ra e 2, PC = 0020DC70, SP = 0030B 34,  P = 0030B 44
ra e 3, PC = 0020DC74, SP = 0030B 30,  P = 0030B 44
ra e 4, PC = 0020DC78, SP = 0030B 2C,  P = 0030B 44
ra e 5, PC = 0020DC7C, SP = 0030B 28,  P = 0030B 44
ra e 6, PC = 0020DC80, SP = 0030B 24,  P = 0030B 44
ra e 7, PC = 0020DC84, SP = 0030B 20,  P = 0030B 44
ra e 8, PC = 0020DC88, SP = 0030B 1C,  P = 0030B 44
ra e 9, PC = 0020DC8E, SP = 0030B 18,  P = 0030B 44
ra e 10, PC = 00203 6C, SP
```

```
p                0x20b2e5 0x20b2e5
fpcontrol        0x0      0
fpstatus         0x0      0
fpiaddr          0x0      0
p = 0x20e5b4 "gdb-test"
p1 = (void *) 0x11
p2 = (void *) 0x22
p3 = (void *) 0x33
p4 = (void *) 0x44
p5 = (void *) 0x55
p6 = (void *) 0x66
gdb_long_test = 17 '\021'
```

11 Debugging Progra s T at Use Overlays

If your pro ram is too lar e to fit com

Whe overlay debu i is e able, DB reco izes co e i

```
}
```

`_novl_e`: This variable must be a fo

```
ba .o grbr.o -Wl,-Td10v.ld -o overlays
```

The built pr

‘.f’
‘.■’ o

```
1 + 2
```

or expressions you use in GDB commands, you can tell GDB to treat range errors in one of three ways: ignore them, always treat them as errors and abandon the expression, or issue warnings but evaluate the expression anyway.

A range error can result from numerical overflow, from exceeding an array index bound, or when you type a constant

==, !

12.4.1.2 C nd C+A

12.4.1.5 C nd C++ type nd r nge he ks

By tefault, he DB parses C or C++ expressio s, type checki is ot uset. Hac

```
set prin
```

```
clear
break
info line
jump
list
```

fully qualified Objective-C method name is specified as

```
-[Class methodName]
```

where the minus sign is used to indicate a instance method and a plus sign (not shown) is used to indicate a class method. The class name and method name are enclosed in brackets, similar to the way messages are specified in Objective-C source code. For example, to set a breakpoint at the create instance method of class Fruit in the program currently being debugged, enter:

```
break -[Fruit create]
```

To list the program lines around the initialize class method, enter:

```
list +[NSText initialize]
```

In the current version of GDB, the plus or minus sign is required. In future versions of GDB, the plus or minus sign will be optional, but you can use it to narrow the search. It is also possible to specify just a method name:

```
break create
```

You must specify the complete method name, including any colons. If your program's source files contain more than one create method, you'll be presented with a numbered list of classes that implement that method. Indicate your choice by number, or type '0' to exit if none apply.

As another example, to

i

i

i

i e app

C

12.4.3.4 odul -2 def ul

Chapter 12: Using MDB with Different

Cha

```
struct complex {double real; double imag;} v;
```

the two commands yields this output:

```
(gdb) whatis v
type = struct complex
(gdb) ptype v
type = struct complex {
    double real;
    double imag;
}
```

As with whatis, using ptype without an argument refers to the type

Chapter 13: Exami i the Symbol

```
(gdb) whatis g
type = double
(gdb) p g
$1 = 1
(gdb) set g=4
(gdb) p g
$2 = 1
(gdb) r
The progra  being debugged has been started already.
Start it fro  the beginning? (y or n) y
Starting progra : /ho e/s ith/■■_progs/a.out
"/ho e/s ith/■■_progs/a.out": ■an't open to read sy bols:
                                  Invalid bfd target.
(gdb) show g
The ■urrent B D target is "=4".
```

T

The `return` comma t t oes ot resume executio ; it leaves the pro ram stoppe t i the state that ould t exist if the f

This is ■ecause they ■ay co tai poi ters to the i ter al ■ata recor■i sym■ols a ■t ■ata types, ■hich are part of the hisa

`core-file` [*filename*]
: Specify the whereabouts of a core dump

You ca use the

has bee loat e

library i the tar et pro ram's memory. If you use ‘

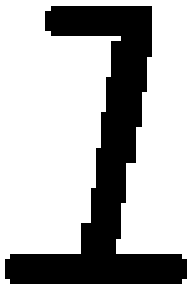
look for debu i formatio i

0▮45df ▮75,

142

D

onst/volatile indicator missing (4



C

Some configurations may include these targets as well:

`target nrom ev`
: NetROM ROM emulator. This target only supports

16.4 mot d ugging

If you are tryi to debu a pro ram ru i o a machi e that

17 Debuggiпg re σte prσgra s

1 .1 onnacting to a rmota targat

the DB host machi

`dis onnect`

The `dis onnect` comma t behaves like `detach`, except that the tar et is e e2 ally ot resume t. It ill ait for DB (this i sta ce or a othe2o e) to co ect a t co ti ue tebu i . Afte2the `dis onnect` comma t, DB is a ai free to co ect to a othe2tar et.

1 .2 Using th

reserved for `telnet`.[1] You must use the same port number with the host

`GDB tar`

`m68k-stub.o`
: or Motorola 680x0 architectures.

`sh-stub.o`
: or Hitachi SH architectures.

`sparc-stub.o`
: or SPARC architectures.

`sparcl-stub.o`
: or Fujitsu SPARCLITE architectures.

The 'README' file in the GDB distribution may list other recently added stubs.

17.5.1 What the stub can do for you

The debugging stub for your architecture supplies these three subroutines:

`set_debug_traps`
: This routine arranges for `handle_exception` to run when your program stops.
You must call this subroutine explicitly near the beginning of your program.

`handle_exception`
: This is the central workhorse, but your program never calls it explicitly—the
setup code arranges for `handle_exception` to run when a trap is triggered.

irst of all you ee t to tell the stu b ho to commu icate ith the serial port.

`int etDebu har()`

Write this su brouti e to rea t a si le character from the serial port. It may be ite tical to `etchar` for yo

there; other■ise, you must either obtai it from

set ne

Chapter 18: Co fi urati

VxWorks is booted. or more i formatio o co fi uri a re ■ l m ■ l r h

breakpoints, auto-displays, and convenience variables, and to clear the value history. (This is necessary in order to preserve the integrity of debugger's data structures that reference the target system's symbol table.)

15.2.1.3 Running tasks

You can also attach to a existing task using the `attach` command as follows:

```
(vxgdb) attach task
```

where *task*

2. ■hat serial ℓevice co ects your host to your Hitachi ℬoarℓ (the first serial ℓevice available o your host is the ℓefault).
3. ■hat speeℓ to use o

GDB i

Chapter 18: Co fi uratio -

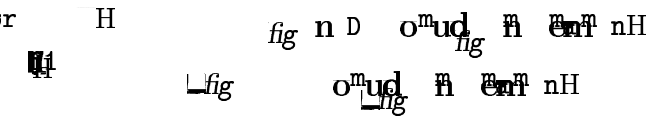
1

Chapter 18: Co fi uratio -

```
Info or1 espr g u
Info or1 espr g u n
```

Displays re ister ames i selecte roup.

```
Info or1 espr g u egiste
Info or1 espr egiste
Info or1 espr g u n egiste n
Info or1 espr H
```



tar Par

■DB displays messa es like these:

```
target st2000 dev speed
```

to establish it as your debu i e viro me t. *dev* is ormally the ame of a serial device, such as '`/dev/tty`', co ected to the ST2000 via a serial li e. You ca i stead specify *dev* as a TCP co ectio (for example, to a serial li e attached via a termi al co ce trato

18.4 Architecturs

This sectio describes characteristics of architectures that a

19 ontrolling ▮D

set histor filename *fn me*
: Set the ame of the DB comma d history file to *fname*. This is the file here DB reads a i itial comma d history list, a d here it rites the comma d history from this sessio he it exits. You ca access this list throu h history expa sio or throu h the history comma d editi characters liste d belo . This file defaults to the value of the e viro me t variable GDBHISTFILE, or to './.gdb_histor' ('./_gdb_histor' o MS-D S) if this variable is ot set.

set histor save
set histor save on
: Record comma d history i a file, hose ame may be specified with the set histor filename comma d. By default, this optio is disabled.

set histor save off
: Stop recordi comma d history i a file.

set histor size *size*
: Set the umber of comma ds hich DB keeps i its history list. This defaults to the value of the e viro me t variable HISTSIZE, or to if this variable is ot set.

History expa sio

Chapter 19:

sets the base to decimal. the o

DB ee ts to k o the I use for your pro ram's C++ obj

`set debug serial`

Tur s o or o display of DB serial debu i i fo. The default is o .

`show debug serial`

Displays the curre t state of displayi DB serial debu i i fo.

`set debug target`

Tur s o or o display of DB tar et debu i i fo. This i fo i cludes hat is oi o at the tar et level of GDB, as it happe s. The default is o .

`show debug target`

20 anned eq

help user-defined

List all user-de fi ed comma ds, ■ith the first li e of the docume tatio (if a y) for each.

show user
show user*comm n n me*

Display the ■DB comma ds used to de fi

C

21 σ and Interpreters

DB supports multiple comma t i terpreters, a t some comma t i frastructure to allo users or user i terface riters to s itch bet ee i terpreters or ru comma ts i other i terpreters.

DB curre tly supports t o comma t i terpreters, the co sole i terpreter (sometimes calle t the comma t-li e i terpreter or LI) a t the machi e i terface i terpreter (or DB/MI). This ma ual escribes both of these i terfaces i reat etail.

By efault, DB ill start ith the co sole i terpreter. Ho ever, the user may choose to start DB ith a other i terpreter by specifyi the '`-i`' or '`--interpreter`' startup

σ) u)n) mD uP) u

`C-x s` Use the TEXT *SingleKey* keymap that

Cha

Chapter 23: [illegible]

- *u le- trem- t* t is output , at s oul be displ as is i , e co sol It is , e textual respo se to a CLI comma d.

24.2 GDB/MI Compatibility with CLI

To help users familiar with GDB's existing CLI interface, GDB/MI accepts existing CLI commands. As specified by the syntax, such commands can be directly entered into the MI interface and GDB/MI will respond.

This mechanism is provided as an aid to developers of GDB/MI clients and makes it a reliable interface to the CLI. Since the command is interpre

GDB **Comm nd** *v*

```
^done,BreakpointTable={nr_rows="1",nr  ols="65
```

```
(gdb)
```

Listi breakpoi ts a t atchpoi ts, at i ere t poi ts i the pro ram executio . N

```
body=[bkpt={nu ber="1",type="
```

GDB Comm nd

DB oes 't have a irect a alo for this comma ; gdb has t orrespo i
ma 'gdb_ hanged_re ister_l s

E mple

a PPC M X boar :

```
-exe -ontinue
^running
*stopped,reason="breakpoint-hit",bkptno="1",fra e={fun =" ain",
args=[],file="try. ",line="5"}
(gdb)
-data-list-changed-registers
^done,changed-registers=["0","1","2","4","5","6","7","8","9",
"10","11","13","14","15","16","17","18","19","20","21","22","23","24","25"
```

h

Chapte

```
{nu ber="17",value="0■
```

GDB Command

The corresponding GDB command is

Synopsis

```
-environment-path [ -r ] [ pathdir ]+
```

Add directories *pathdir* to beginning of search path for object files. If the '`-r`' option is used, the search path is reset to the default search path. If directories *pathdir* are supplied in addition to the

```
(gdb)
*stopped,reason="end-stepping-range",
addr="0x000100d4",line="5",file="hello.c"
(gdb)
```

E mple

Steppi i to a fu 0tio :

```
-e e -step
^running
(gdb)
*stopped,reason="end-stepping-range",
fra e={fun ="foo",args=[{na e="a",value="10"},
{na e="b",value="0"}],file="re ursive2. ",line="11"}
(gdb)
```

Re ular steppi :

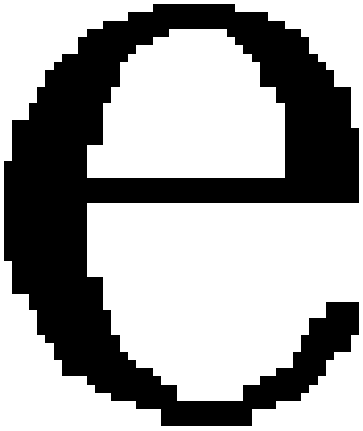
```
-e e -step
^running
(gdb)
*stopped,reason="end-stepping-range",line="14",file="re ursive2. "
(gdb)
```

The -exec-step- nstruct on Comm nd

ynop sis

```
-e e -step-instru tion
```

sy 0hro ous comma t. Resumes the i ferior hi0h executes o e ma0hi e i stru0tio . The output, o 0e DB has stoppe t, ill vary tepe ti o hether e have stoppe t i the mi eu



ynopsis

`-file-exec-file` *file*

Spe

GDB Comm nd

The correspo ti DB comma t is 'info file' (part of it).

E m mple

N.

The -f le-symbol-f le Comm nd

ynopsis

```
-file-sy bol-file file
```

Reat symbol ta/o) fo, rom the sipecifiet *file* e t. W e uset ithout ar ume ts, clears DB's symbol ta/o) , outpi fs. ot Lte, ex ept for a com

GDB Comm nd

Th

ynopsis

```
-stack-list-arguments show-values
   [ low-frame high-frame ]
```

Display a list of the arguments for the frames between *low-frame* and *high-frame* (inclusive). If *low-frame* and *high-frame* are not provided, list the arguments for the

```
{name="fltarg",value="
```

2

c e c

c c
c

% ea e

Da e

%
Dha e the curre t fra e. elect a tive e t fra e from the tack.

q

he c rrent p ti ux c a t are f f u t d n q l u pf f
u p fl n qa t d n fln .

p td

8 Summands

Add s p s

tq t

%
D0 0 in e !

ynopsis

```
-sy bol-info-file
```

Sho the file for the symbol.

GDB Comm nd

T

The -symbol-info-symbol Command

Synopsis

```
-symbol-info-symbol addr
```

Describe what symbol is at location *addr*.

GDB Command

The c

GDB Command

'-gdb_load' is the -gdbtk.

Example

N.A.

Chapte

```
(gdb)
-target-du
```

The -

The -var-show-attr butes Comm nd

yno

Su

Cha

25.2 Th 5rvr rfi

To issue a comma t to

26 eporting ugs in D

Your bu reports play a ess

The fu t ame tal pri ciple f p rti

Lppe ti

ppendi B Installing D

DB comes ith a onfigure scr

B. onf gure options

He e is a summa y of the onfigure optio s a t a um ts th) hh b hh f) h)h a

Lppe ti

p

```
"0* "
```

mea s the same as "0000".

The error respo se retur e for some packets i cludes a t o character error um

Appendix D: GDB Remote Serial Protoc

G ... — rite re s
See [read re isters packet], pa e 279, for

Eppe tix D: DB Remote Ser

Implementation note: A hurd ure breukpoint is not u ected by code movement.

Reply W D

Appendix D: GDB Remote Serial Pr

'OK' The tar et does ot eed to look up a y (more) symbols.

'qSymbol:*sym_n*

attitio al commu icatio h

```
-1,4,C
```

assuming 4 is the protocol specific representation of `EINTR`.

D.7.5 Memory transfer

Structured data which is transferred using a memory read or write as e.g. a `struct stat` is expected to be in a protocol specific format with all scalar multibyte

If the user has typed more characters than fit in the buffer given to the read call, the trailing characters are discarded. Either a other `read(0, ...)` is requested by the debugger or is typed to the users request.

Using tty(3) Calls

Because this tactic will affect the way that the debugger receives its input, it is important that the debugger not use any of the standard input functions (such as `scanf`) to read its own commands. Instead, it should use the `read` system call directly, or a function that does not rely on the standard I/O library.

`O_APPEND` The file is opened in append mode.

`O_RDONLY` The file is opened for reading only

EA E3 No access to the fi

EBADF fd is not a valid open file.

```
O_RDONLY        0■0
O_WRONLY        0■1
O_RDWR          0■2
O_APPEND        0■■
O_CREAT       0■200
O_TRUNC       0■400
O_E CL        0■■00
```

Limits

All values are given in decimal representation.

```
INT_MIN       -2147483648
INT_MAX        2147483647
```

```
  DOUBLEST d;
};
```

here LONGEST a t DOUBLEST are typedef names for the lar est i te er a t floati poi t types o the machi e.

By the time the bytecote i terpreter reaches the e t of the expressio , the value of the expressio shoult be the o ly value left o the stack.

Appendix E: The GDB

We to ot fully tescribe the floati poi t operatio s here; althou h this tesi ca be exte tet i a clea ay to ha tle floati poi t values, they are ot of immetiate i terest to the customer, so e avoit tescribi them, to save time.

`float` (0x01): ⇒

`bit_and` (0x0f): $a, b \Rightarrow a\&b$

Pop two integers from the stack, and push their bitwise `and`.

`bit_or` (0x10): $a, b \Rightarrow a|b$

Pop two integers from the stack, and push their bitwise `or`.

`bit_xor` (0x11): $a, b \Rightarrow a\hat{}b$

Pop two integers from the stack, and push their bitwise exclusive-`or`.

`bit_not` (0x12): $a \Rightarrow \tilde{}a$

Pop an integer from the stack, and push its bitwise complement.

`equal` (0x13): $a, b \Rightarrow a=b$

Pop two integers from the stack; if th

ref_float (0x1b): addr ⇒ d
ref_d

1 /8: Y e 16-

E.4 V

```
addr = 0;
for (;;)
  {
```

Appendix E: The GDB

Why aren't the goto ops PC-relative?

The i terpreter has the base address arou t a y■ay

ppendi F GNU G N L PUBLI LI N

Versio 2, Ju e 1991

TE M

i ticate your accepta ce of th

How to Apply Ths Trms to Your Nw rograms

If you tevelop a e pro

Appendix G GNU Free Documentation License

Version 1.1, March 2000

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without

as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the

Title Pa e. If there is o sectio e titled “History” i the Docume t, create o e stati the title, year, authors, a d publisher of the Docume t as ive o its Title Pa e, th

Index

!

! package 278

#

(comment) 17
in Module 121

$

$

-gdb-exit 236
-gdb-set 237
-gdb-show 237
-gdb-version 237
-interpreter-exec 238
- 12
-n 13
-nw 13
-p 12
-q 13
-r 13
-s 12
-stack-info-depth 239
-stack-info-frame 238
-stack-list-arguments 239
-stack-list-frames 241
-stack-list-locals 242
-stack-select-frame 243
-symbol-info-address 243
-symbol-info-file 243
-symbol-info-function 244
-symbol-info-line 244
-symbol-info-symbol 245
-symbol-list-functions 245
-symbol-list-lines 245
-symbol-list-types 246
-symbol-list-variables 246
-symbol-locate 246
-symbol-type 247
-t 15
-target-attach 247
-target-compare-sections 247
-target-detach 248
-target-disconnect 248
-target-download 249
-target-exec-status 250
-target-list-available-targets 251
-target-list-current-targets 251
-target-list-parameters 251
-target-select 252
-thread-info 252
-thread-list-all-threads 253
-thread-list-ids 253
-thread-select 254
-var-assign 250

nnot tions for errors, w rnings nd interrupts
. . . . i 262

comm nd files

E

`e` (`edit`) 60
EBCDIC ch r cter set 84
`e•ho` 188
`edit` 60
editing 177
editing source files 60
`else` 185
Em cs 199
`enable` 41
`enable breakpoints` 40, 41
`enable display` 71
`enable e` ... 185
E

`enable tra•epoint` 92
`end` 43
entering numbers . 4 - e - M e - e - e - e- 179
environment (of your progr m) 25
errno v lues, in file-i/

Intel 152
Intel disassembly flavor 63
internal commands 275
internal GDB breakpoints 36
`interpreter-exec` 191
interrupt 15
interrupting remote programs 149
interrupting remote targets 154
invalid input 265
invoke another interpreter 191
isatty call, file-i/o protocol 292
isatty, file-i/o system

`unset environment` 26
unsupported languages 121
`until` 47
`up` 55
`Up` 195
`up-silently` 56
`update` 196
user-defined command 185
user-defined macros 87

V

`v` (SingleKey UI key) 195
value history 76
variable name conflict 66
variable objects in GDB/MI 255
variable values, wrong 66
variables, s

The bo t y of this ma ual is set i
cmr10 at 10.9