```
CC       = gcc
CPPFLAGS = $(DFLAGS) $(INCLUDES)
CFLAGS   = -g -Wall -O2
LDFLAGS  =
LDLIBS   =
DFLAGS=      -D_FILE_OFFSET_BITS=64 -D_LARGEFILE64_SOURCE -D_CURSES_LIB=1
LOBJS=       bam_aux.o bam.o bam_import.o sam.o \
             sam_header.o bam_plbuf.o
AOBJS=       bam_index.o bam_plcmd.o sam_view.o \
             bam_cat.o bam_md.o bam_reheader.o bam_sort.o bedidx.o kprobaln.o \
             bam_rmdup.o bam_rmdupse.o bam_mate.o bam_stat.o bam_color.o \
             bamtk.o bam2bcf.o bam2bcf_indel.o errmod.o sample.o \
             cut_target.o phase.o bam2depth.o padding.o bedcov.o bamshuf.o \
             faidx.o stats.o stats_isize.o bam_flags.o bam_split.o \
             bam_tview.o bam_tview_curses.o bam_tview_html.o bam_lpileup.o
INCLUDES=    -I. -I$(HTSDIR)
LIBCURSES=   -lcurses # -lXCurses

prefix      = /usr/local
exec_prefix = $(prefix)
bindir      = $(exec_prefix)/bin
mandir      = $(prefix)/share/man
man1dir     = $(mandir)/man1

MKDIR_P = mkdir -p
INSTALL = install -p
INSTALL_PROGRAM = $(INSTALL)
INSTALL_DATA    = $(INSTALL) -m 644
INSTALL_DIR     = $(MKDIR_P) -m 755


PROGRAMS = samtools

BUILT_MISC_PROGRAMS = \
  misc/ace2sam misc/maq2sam-long misc/maq2sam-short \
  misc/md5fa misc/md5sum-lite misc/wgsim

MISC_PROGRAMS = \
  $(BUILT_MISC_PROGRAMS) \
  misc/blast2sam.pl misc/bowtie2sam.pl misc/export2sam.pl \
  misc/interpolate_sam.pl misc/novo2sam.pl \
  misc/plot-bamstats misc/psl2sam.pl \
  misc/sam2vcf.pl misc/samtools.pl misc/seq_cache_populate.pl \
  misc/soap2sam.pl \
  misc/varfilter.py misc/wgsim_eval.pl misc/zoom2sam.pl

BUILT_TEST_PROGRAMS = \
  test/merge/test_bam_translate \
  test/merge/test_pretty_header \
  test/merge/test_rtrans_build \
  test/merge/test_trans_tbl_init \
  test/split/test_count_rg \
  test/split/test_expand_format_string \
  test/split/test_filter_header_rg \
  test/split/test_parse_args \
  test/vcf-miniview

all: $(PROGRAMS) $(BUILT_MISC_PROGRAMS) $(BUILT_TEST_PROGRAMS)


# Adjust $(HTSDIR) to point to your top-level htslib directory
HTSDIR = htslib-1.2.1
include $(HTSDIR)/htslib.mk
HTSLIB = $(HTSDIR)/libhts.a
BGZIP  = $(HTSDIR)/bgzip


PACKAGE_VERSION = 1.2

# If building from a Git repository, replace $(PACKAGE_VERSION) with the Git
# description of the working tree: either a release tag with the same value
```

```make
# as $(PACKAGE_VERSION) above, or an exact description likely based on a tag.
# $(shell), :=, etc are GNU Make-specific.  If you don't have GNU Make,
# comment out this conditional.
ifneq "$(wildcard .git)" ""
PACKAGE_VERSION := $(shell git describe --always --dirty)

# Force version.h to be remade if $(PACKAGE_VERSION) has changed.
version.h: $(if $(wildcard version.h),$(if $(findstring "$(PACKAGE_VERSION)",$(shell cat
version.h)),,force))
endif

# If you don't have GNU Make but are building from a Git repository, you may
# wish to replace this with a rule that always rebuilds version.h:
# version.h: force
# echo '#define SAMTOOLS_VERSION "`git describe --always --dirty`"' > $@
version.h:
	echo '#define SAMTOOLS_VERSION "$(PACKAGE_VERSION)"' > $@


.SUFFIXES: .c .o

.c.o:
	$(CC) $(CFLAGS) $(CPPFLAGS) -c -o $@ $<


lib:libbam.a

libbam.a:$(LOBJS)
	$(AR) -csru $@ $(LOBJS)

samtools: $(AOBJS) libbam.a $(HTSLIB)
	$(CC) -pthread $(LDFLAGS) -o $@ $(AOBJS) libbam.a $(HTSLIB) $(LDLIBS) $(LIBCURSES) -lm -lz

bam_h = bam.h $(htslib_bgzf_h) $(htslib_sam_h)
bam2bcf_h = bam2bcf.h $(htslib_vcf_h) errmod.h
bam_lpileup_h = bam_lpileup.h $(htslib_sam_h)
bam_plbuf_h = bam_plbuf.h $(htslib_sam_h)
bam_tview_h = bam_tview.h $(htslib_hts_h) $(htslib_sam_h) $(htslib_faidx_h) $(bam2bcf_h) $(HTSDIR)/
htslib/khash.h $(bam_lpileup_h)
sam_h = sam.h $(htslib_sam_h) $(bam_h)
sample_h = sample.h $(HTSDIR)/htslib/kstring.h

bam.o: bam.c $(bam_h) sam_header.h
bam2bcf.o: bam2bcf.c $(htslib_sam_h) $(HTSDIR)/htslib/kstring.h $(HTSDIR)/htslib/kfunc.h $(bam2bcf_h)
errmod.h
bam2bcf_indel.o: bam2bcf_indel.c $(htslib_sam_h) $(bam2bcf_h) kprobaln.h $(HTSDIR)/htslib/khash.h
$(HTSDIR)/htslib/ksort.h
bam2depth.o: bam2depth.c $(htslib_sam_h) samtools.h
bam_aux.o: bam_aux.c
bam_cat.o: bam_cat.c $(htslib_bgzf_h) $(bam_h)
bam_color.o: bam_color.c $(bam_h)
bam_import.o: bam_import.c $(HTSDIR)/htslib/kstring.h $(bam_h) $(HTSDIR)/htslib/kseq.h
bam_index.o: bam_index.c $(htslib_hts_h) $(htslib_sam_h) $(HTSDIR)/htslib/khash.h
bam_lpileup.o: bam_lpileup.c $(bam_plbuf_h) $(bam_lpileup_h) $(HTSDIR)/htslib/ksort.h
bam_mate.o: bam_mate.c $(bam_h)
bam_md.o: bam_md.c $(htslib_faidx_h) $(sam_h) kprobaln.h
bam_pileup.o: bam_pileup.c $(sam_h)
bam_plbuf.o: bam_plbuf.c $(htslib_hts_h) $(htslib_sam_h) $(bam_plbuf_h)
bam_plcmd.o: bam_plcmd.c $(htslib_sam_h) $(htslib_faidx_h) $(HTSDIR)/htslib/kstring.h $(HTSDIR)/htslib/
khash_str2int.h sam_header.h samtools.h $(bam2bcf_h) $(sample_h)
bam_reheader.o: bam_reheader.c $(htslib_bgzf_h) $(bam_h)
bam_rmdup.o: bam_rmdup.c $(sam_h) $(HTSDIR)/htslib/khash.h
bam_rmdupse.o: bam_rmdupse.c $(sam_h) $(HTSDIR)/htslib/khash.h $(HTSDIR)/htslib/klist.h
bam_sort.o: bam_sort.c $(HTSDIR)/htslib/ksort.h $(HTSDIR)/htslib/khash.h $(HTSDIR)/htslib/klist.h
$(HTSDIR)/htslib/kstring.h $(htslib_sam_h)
bam_stat.o: bam_stat.c $(bam_h) samtools.h
bam_tview.o: bam_tview.c $(bam_tview_h) $(htslib_faidx_h) $(htslib_sam_h) $(htslib_bgzf_h)
bam_tview_curses.o: bam_tview_curses.c $(bam_tview_h)
bam_tview_html.o: bam_tview_html.c $(bam_tview_h)
bam_flags.o: bam_flags.c $(sam_h)
bamshuf.o: bamshuf.c $(htslib_sam_h) $(HTSDIR)/htslib/ksort.h samtools.h
bamtk.o: bamtk.c $(htslib_hts_h) version.h samtools.h
```

```
bedcov.o: bedcov.c $(HTSDIR)/htslib/kstring.h $(htslib_sam_h) $(HTSDIR)/htslib/kseq.h
bedidx.o: bedidx.c $(HTSDIR)/htslib/ksort.h $(HTSDIR)/htslib/kseq.h $(HTSDIR)/htslib/khash.h
cut_target.o: cut_target.c $(bam_h) errmod.h $(htslib_faidx_h)
errmod.o: errmod.c errmod.h $(HTSDIR)/htslib/ksort.h
kprobaln.o: kprobaln.c kprobaln.h
padding.o: padding.c sam_header.h $(sam_h) $(bam_h) $(htslib_faidx_h)
phase.o: phase.c $(htslib_sam_h) errmod.h $(HTSDIR)/htslib/kseq.h $(HTSDIR)/htslib/khash.h $(HTSDIR)/
htslib/ksort.h
sam.o: sam.c $(htslib_faidx_h) $(sam_h)
sam_header.o: sam_header.c sam_header.h $(HTSDIR)/htslib/khash.h
sam_view.o: sam_view.c $(htslib_sam_h) $(htslib_faidx_h) $(HTSDIR)/htslib/kstring.h $(HTSDIR)/htslib/
khash.h samtools.h
sample.o: sample.c $(sample_h) $(HTSDIR)/htslib/khash.h
stats_isize.o: stats_isize.c stats_isize.h $(HTSDIR)/htslib/khash.h
stats.o: stats.c $(sam_h) sam_header.h samtools.h stats_isize.h $(HTSDIR)/htslib/khash.h $(HTSDIR)/
htslib/khash_str2int.h $(htslib_faidx_h)


# test programs

# For tests that might use it, set $REF_PATH explicitly to use only reference
# areas within the test suite (or set it to ':' to use no reference areas).
# (regression.sh sets $REF_PATH to a subdirectory itself.)
check test: samtools $(BGZIP) $(BUILT_TEST_PROGRAMS)
	REF_PATH=: test/test.pl --exec bgzip=$(BGZIP)
	test/merge/test_bam_translate test/merge/test_bam_translate.tmp
	test/merge/test_pretty_header
	test/merge/test_rtrans_build
	test/merge/test_trans_tbl_init
	cd test/mpileup && ./regression.sh
	test/split/test_count_rg
	test/split/test_expand_format_string
	test/split/test_filter_header_rg
	test/split/test_parse_args


test/merge/test_bam_translate: test/merge/test_bam_translate.o test/test.o $(HTSLIB)
	$(CC) -pthread $(LDFLAGS) -o $@ test/merge/test_bam_translate.o test/test.o $(HTSLIB) $(LDLIBS) -lz

test/merge/test_pretty_header: test/merge/test_pretty_header.o $(HTSLIB)
	$(CC) -pthread $(LDFLAGS) -o $@ test/merge/test_pretty_header.o $(HTSLIB) $(LDLIBS) -lz

test/merge/test_rtrans_build: test/merge/test_rtrans_build.o $(HTSLIB)
	$(CC) -pthread $(LDFLAGS) -o $@ test/merge/test_rtrans_build.o $(HTSLIB) $(LDLIBS) -lz

test/merge/test_trans_tbl_init: test/merge/test_trans_tbl_init.o $(HTSLIB)
	$(CC) -pthread $(LDFLAGS) -o $@ test/merge/test_trans_tbl_init.o $(HTSLIB) $(LDLIBS) -lz

test/split/test_count_rg: test/split/test_count_rg.o test/test.o $(HTSLIB)
	$(CC) -pthread $(LDFLAGS) -o $@ test/split/test_count_rg.o test/test.o $(HTSLIB) $(LDLIBS) -lz

test/split/test_expand_format_string: test/split/test_expand_format_string.o test/test.o $(HTSLIB)
	$(CC) -pthread $(LDFLAGS) -o $@ test/split/test_expand_format_string.o test/test.o $(HTSLIB)
$(LDLIBS) -lz

test/split/test_filter_header_rg: test/split/test_filter_header_rg.o test/test.o $(HTSLIB)
	$(CC) -pthread $(LDFLAGS) -o $@ test/split/test_filter_header_rg.o test/test.o $(HTSLIB) $(LDLIBS) -
lz

test/split/test_parse_args: test/split/test_parse_args.o test/test.o $(HTSLIB)
	$(CC) -pthread $(LDFLAGS) -o $@ test/split/test_parse_args.o test/test.o $(HTSLIB) $(LDLIBS) -lz

test/vcf-miniview: test/vcf-miniview.o $(HTSLIB)
	$(CC) -pthread $(LDFLAGS) -o $@ test/vcf-miniview.o $(HTSLIB) $(LDLIBS) -lz

test_test_h = test/test.h $(htslib_sam_h)

test/merge/test_bam_translate.o: test/merge/test_bam_translate.c $(test_test_h) bam_sort.o
test/merge/test_pretty_header.o: test/merge/test_pretty_header.c bam_sort.o
test/merge/test_rtrans_build.o: test/merge/test_rtrans_build.c bam_sort.o
test/merge/test_trans_tbl_init.o: test/merge/test_trans_tbl_init.c bam_sort.o
test/split/test_count_rg.o: test/split/test_count_rg.c bam_split.o $(test_test_h)
```