

```
1: 1: #-- set this to the man directory you would like to use
2: MANPATH:=/usr/share/man
3:
4: 4: #-- uncomment this to enable debugging
5: #DEBUG:=-g -DDEBUG
6:
7:
8: 8: ##### YOU SHOULD NOT CHANGE BELOW THIS LINE #####
9:
10: VERSION:=1.2.0
11: SHELL:=/bin/bash
12: SRCS:=api.c
13: MANS:=man3/libxbee.3 \
14:     man3/xbee_con.3 \
15:     man3/xbee_end.3 \
16:     man3/xbee_endcon.3 \
17:     man3/xbee_flushcon.3 \
18:     man3/xbee_getanalog.3 \
19:     man3/xbee_getdigital.3 \
20:     man3/xbee_getpacket.3 \
21:     man3/xbee_hasanalog.3 \
22:     man3/xbee_hasdigital.3 \
23:     man3/xbee_logit.3 \
24:     man3/xbee_newcon.3 \
25:     man3/xbee_nsnddata.3 \
26:     man3/xbee_pkt.3 \
27:     man3/xbee_snddata.3 \
28:     man3/xbee_setup.3 \
29:     man3/xbee_setupAPI.3 \
30:     man3/xbee_setuplog.3 \
31:     man3/xbee_setuplogAPI.3 \
32:     man3/xbee_vsnddata.3
33: MANPATHS:=$(foreach dir,$(shell ls man -ln | grep ^d | tr -s ' ' | cut -d ' ' -f 9),${MANPATH}/${dir})
34:
35: PDFS:=${SRCS} ${SRCS:.c=.h} makefile main.c xbee.h
36:
37: CC:=gcc
38: CFLAGS:=-Wall -Wstrict-prototypes -Wno-variadic-macros -pedantic -c -fPIC ${DEBUG}
39: CLINKS:=-lpthread -lrt ${DEBUG}
40: DEFINES:=
41:
42: ifeq ($(strip $(wildcard ${MANPATH}/man3/libxbee.3.bz2)),)
43:     FIRSTTIME:=TRUE
44: else
45:     FIRSTTIME:=FALSE
46: endif
47:
48: ENSCRYPT:=-MA4 --color -f Courier8 -C --margins=15:15:0:20
49: ifneq ($(strip $(wildcard /usr/share/enscript/mine-web.hdr)),)
50:     ENSCRYPT+= --fancy-header=mine-web
51: else
52:     ENSCRYPT+= --fancy-header=a2ps
53: endif
54:
55: SRCS:=$(sort ${SRCS})
56: PDFS:=$(sort ${PDFS})
57:
58: .PHONY: FORCE
59: .PHONY: all run new clean cleanpdfs main pdfs
60: .PHONY: install install_su install_man
61: .PHONY: uninstall uninstall_su uninstall_man/
62:
63:
64: 64: # all - do everything (default) #
65: all: ./lib/libxbee.so.${VERSION} main
66:     @echo "**** Done! ****"
67:
68:
69: 69: # run - remake main and then run #
70: run: main
71:     ./bin/main
72:
73:
74: 74: # new - clean and do everything again #
75: new: clean all
76:
77:
78: 78: # clean - remove any compiled files and PDFs #
79: clean:
80:     rm -f ./.*~
81:     rm -f ./sample/*~
82:     rm -f ./obj/*.o
83:     rm -f ./lib/libxbee.so*
84:     rm -f ./bin/main
85:
```

```

86: cleanpdfs:
87:     rm -f ./pdf/*.pdf
88:
89:
90: # install - installs library #
91: install: ./lib/libxbee.so.${VERSION}
92:     @echo
93:     @echo
94: ifneq ($(shell echo $$USER),root)
95:     @echo "#####"
96:     @echo "### To Install this library I need the root password please!"
97:     @echo "#####"
98: endif
99:     su -c "make install_su --no-print-directory"
100:    @echo
101: ifeq (${FIRSTTIME},TRUE)
102:    @echo "#####"
103:    @echo
104:    @pr -h "" -o 3 -w 86 -tT ./README
105:    @echo
106:    @echo "#####"
107: endif
108:
109: install_su: /usr/lib/libxbee.so.${VERSION} /usr/include/xbee.h install_man
110:
111: /usr/lib/libxbee.so.${VERSION}: ./lib/libxbee.so.${VERSION}
112:     cp ./lib/libxbee.so.${VERSION} /usr/lib/libxbee.so.${VERSION} -f
113:     @chmod 755 /usr/lib/libxbee.so.${VERSION}
114:     @chown root:root /usr/lib/libxbee.so.${VERSION}
115:     ln ./libxbee.so.${VERSION} /usr/lib/libxbee.so.1 -sf
116:     @chown root:root /usr/lib/libxbee.so.1
117:     ln ./libxbee.so.${VERSION} /usr/lib/libxbee.so -sf
118:     @chown root:root /usr/lib/libxbee.so
119:
120: /usr/include/xbee.h: ./xbee.h
121:     cp ./xbee.h /usr/include/xbee.h -f
122:     @chmod 644 /usr/include/xbee.h
123:     @chown root:root /usr/include/xbee.h
124:
125: install_man: ${MANPATH} ${MANPATHS} ${addsuffix .bz2,${addprefix ${MANPATH}/,${MANS}}}
126:
127: ${MANPATH} ${MANPATHS}:
128:     @echo "#####"
129:     @echo "### $@ does not exist... cannot install man files here!"
130:     @echo "### Please check the directory and the MANPATH variable in the makefile"
131:     @echo "#####"
132:     @false
133:
134: ${MANPATH}/%.bz2: ./man/%
135:     @echo "cat $< | bzip2 -z > $@"
136:     @cat $< | bzip2 -z > $@ || ( \
137:         echo "#####"; \
138:         echo "### Installing man page '$*' to '$@' failed..."; \
139:         echo "#####"; )
140:     @chmod 644 $@
141:     @chown root:root $@
142:
143: ./doc/:
144:     mkdir ./doc/
145:
146: html: ./doc/ ./man/
147:     cd ./doc/; mkdir -p `find ../man/ -type d -not -path *.svn* | cut -b 2-`;
148:     find ./man/ -type f -not -path *.svn* | cut -d / -f 3- | sort > .html_todo
149:     for item in `cat .html_todo`; do \
150:         man2html -r ./man/$$item | tail -n +3 > ./doc/man/$$item.html; \
151:         done 2> /dev/null
152:     rm .html_todo
153:
154: uninstall:
155:     @echo
156:     @echo
157: ifneq ($(shell echo $$USER),root)
158:     @echo "#####"
159:     @echo "### To Uninstall this library I need the root password please!"
160:     @echo "#####"
161: endif
162:     su -c "make uninstall_su --no-print-directory"
163:     @echo
164:     @echo
165:
166: uninstall_su: ${addprefix uninstall_man/,${MANS}}
167:     rm /usr/lib/libxbee.so.${VERSION} -f
168:     rm /usr/lib/libxbee.so.1 -f
169:     rm /usr/lib/libxbee.so -f
170:     rm /usr/include/xbee.h -f

```

```
171:
172: uninstall_man%:
173:     rm ${MANPATH}/${*.bz2} -f
174:
175: # main - compile & link objects #
176: main: ./bin/main
177:
178: ./bin/main: ./obj/api.o ./bin/ ./main.c
179:     ${CC} ${CLINKS} ./main.c ./obj/api.o -o ./bin/main ${DEBUG}
180:
181: ./bin/:
182:     mkdir ./bin/
183:
184: ./lib/libxbee.so.${VERSION}: ./lib/ ${addprefix ./obj/,${SRCS:.c=.o}} ./xbee.h
185:     gcc -shared -Wl,-soname,libxbee.so.1 ${CLINKS} -o ./lib/libxbee.so.${VERSION} ./obj/*.o
186: ifeq ($(strip $(wildcard ./lib/libxbee.so.1)),)
187:     ln ./libxbee.so.${VERSION} ./lib/libxbee.so.1 -sf
188: endif
189: ifeq ($(strip $(wildcard ./lib/libxbee.so)),)
190:     ln ./libxbee.so.${VERSION} ./lib/libxbee.so -sf
191: endif
192:
193: ./lib/:
194:     mkdir ./lib/
195:
196: ./obj/:
197:     mkdir ./obj/
198:
199: ./obj/%.o: ./obj/ %.c %.h xbee.h
200:     ${CC} ${CFLAGS} ${DEFINES} ${DEBUG} $.c -o $@
201:
202: ./obj/%.o: ./obj/ %.c xbee.h
203:     ${CC} ${CFLAGS} ${DEFINES} ${DEBUG} $.c -o $@
204:
205:
206: # pdfs - generate PDFs for each source file #
207: ifneq ($(strip $(wildcard /usr/bin/ps2pdf)),)
208: ifneq ($(strip $(wildcard /usr/bin/enscript)),)
209: pdfs: ./pdf/ ${addprefix ./pdf/,${addsuffix .pdf,${PDFS}}}
210:
211: ./pdf/:
212:     mkdir ./pdf/
213:
214: ./pdf/makefile.pdf: ./makefile
215:     enscript ${ENSCRIPT} -Emakefile $< -p - | ps2pdf - $@
216:
217: ./pdf/%.pdf: %
218:     enscript ${ENSCRIPT} -Ec $< -p - | ps2pdf - $@
219:
220: ./pdf/%.pdf:
221:     @echo "*** Cannot make $@ - '$*' does not exist ***"
222: else
223: pdfs:
224:     @echo "WARNING: enscript is not installed - cannot generate PDF files"
225: endif
226: else
227: pdfs:
228:     @echo "WARNING: ps2pdf is not installed - cannot generate PDF files"
229: endif
```