

```

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:
11: SRCS:=api.c
12: MANS:=man3/libxbee.3 \
13:     man3/xbee_setup.3 \
14:     man3/xbee_setuplog.3 \
15:     man3/xbee_newcon.3 \
16:     man3/xbee_flushcon.3 \
17:     man3/xbee_endcon.3 \
18:     man3/xbee_getpacket.3 \
19:     man3/xbee_senddata.3 \
20:     man3/xbee_nsenddata.3 \
21:     man3/xbee_vsenddata.3 \
22:     man3/xbee_getdigital.3 \
23:     man3/xbee_hasdigital.3 \
24:     man3/xbee_getanalog.3 \
25:     man3/xbee_hasanalog.3 \
26:     man3/xbee_pkt.3
27: MANPATHS:=$(foreach dir,$(shell ls man -ln | grep ^d | tr -s ' ' | cut -d ' ' -f 9),${MANPATH}/${dir})
28:
29: PDFS:=${SRCS} ${SRCS:.c=.h} makefile main.c xbee.h globals.h
30:
31: CC:=gcc
32: CFLAGS:=-Wall -Wstrict-prototypes -pedantic -c -fPIC ${DEBUG}
33: CLINKS:=-lm ./lib/libxbee.so.1.0.1 -lpthread ${DEBUG}
34: DEFINES:=
35:
36: ifeq ($(strip $(wildcard ${MANPATH}/man3/libxbee.3.bz2)),)
37: FIRSTTIME:=TRUE
38: else
39: FIRSTTIME:=FALSE
40: endif
41:
42: ENSCRIPT:=-MA4 --color -f Courier8 -C --margins=15:15:0:20
43: ifneq ($(strip $(wildcard /usr/share/enscript/mine-web.hdr)),)
44: ENSCRIPT+= --fancy-header=mine-web
45: else
46: ENSCRIPT+= --fancy-header=a2ps
47: endif
48:
49: SRCS:=$(sort ${SRCS})
50: PDFS:=$(sort ${PDFS})
51:
52: .PHONY: FORCE
53: .PHONY: all run new clean cleanpdfs main pdfs
54: .PHONY: install install_su install_man
55: .PHONY: uninstall uninstall_su uninstall_man/
56:
57:
58: 58: # all - do everything (default) #
59: all: main
60:     @echo "**** Done! ****"
61:
62:
63: 63: # run - remake main and then run #
64: run: main
65:     LD_LIBRARY_PATH=./lib:$LD_LIBRARY_PATH ./bin/main
66:
67:
68: 68: # new - clean and do everything again #
69: new: clean all
70:
71:
72: 72: # clean - remove any compiled files and PDFs #
73: clean:
74:     rm -f ./~
75:     rm -f ./svn_version.c
76:     rm -f ./sample/~
77:     rm -f ./obj/*.o
78:     rm -f ./lib/libxbee.so*
79:     rm -f ./bin/main
80:
81: cleanpdfs:
82:     rm -f ./pdf/*.pdf
83:
84:
85: 85: # install - installs library #

```

```

86: install: ./lib/libxbee.so.1.0.1
87:     @echo
88:     @echo
89: ifneq ($(shell echo $$USER),root)
90:     @echo "#####"
91:     @echo "### To Install this library I need the root password please!"
92:     @echo "#####"
93: endif
94:     su -c "make install_su --no-print-directory"
95:     @echo
96: ifeq (${FIRSTTIME},TRUE)
97:     @echo "#####"
98:     @echo
99:     @pr -h "" -o 3 -w 86 -tT ./README
100:    @echo
101:    @echo "#####"
102: endif
103:
104: install_su: /usr/lib/libxbee.so.1.0.1 /usr/include/xbee.h install_man
105:
106: /usr/lib/libxbee.so.1.0.1: ./lib/libxbee.so.1.0.1
107:     cp ./lib/libxbee.so.1.0.1 /usr/lib/libxbee.so.1.0.1 -f
108:     @chmod 755 /usr/lib/libxbee.so.1.0.1
109:     @chown root:root /usr/lib/libxbee.so.1.0.1
110:     cp /usr/lib/libxbee.so.1.0.1 /usr/lib/libxbee.so.1 -sf
111:     @chown root:root /usr/lib/libxbee.so.1
112:     cp /usr/lib/libxbee.so.1.0.1 /usr/lib/libxbee.so -sf
113:     @chown root:root /usr/lib/libxbee.so
114:
115: /usr/include/xbee.h: ./xbee.h
116:     cp ./xbee.h /usr/include/xbee.h -f
117:     @chmod 644 /usr/include/xbee.h
118:     @chown root:root /usr/include/xbee.h
119:
120: install_man: ${MANPATH} ${MANPATHS} ${addsuffix .bz2},${addprefix ${MANPATH}/,${MANS}}
121:
122: ${MANPATH} ${MANPATHS}:
123:     @echo "#####"
124:     @echo "### $@ does not exist... cannot install man files here!"
125:     @echo "### Please check the directory and the MANPATH variable in the makefile"
126:     @echo "#####"
127:     @false
128:
129: ${MANPATH}/%.bz2: ./man/%
130:     @echo "cat $< | bzip2 -z > $@"
131:     @cat $< | bzip2 -z > $@ || ( \
132:         echo "#####"; \
133:         echo "### Installing man page '$*' to '$@' failed..."; \
134:         echo "#####"; )
135:     @chmod 644 $@
136:     @chown root:root $@
137:
138: uninstall:
139:     @echo
140:     @echo
141: ifneq ($(shell echo $$USER),root)
142:     @echo "#####"
143:     @echo "### To Uninstall this library I need the root password please!"
144:     @echo "#####"
145: endif
146:     su -c "make uninstall_su --no-print-directory"
147:     @echo
148:     @echo
149:
150: uninstall_su: ${addprefix uninstall_man/,${MANS}}
151:     rm /usr/lib/libxbee.so.1.0.1 -f
152:     rm /usr/lib/libxbee.so.1 -f
153:     rm /usr/lib/libxbee.so -f
154:     rm /usr/include/xbee.h -f
155:
156: uninstall_man/%:
157:     rm ${MANPATH}/${*.bz2} -f
158:
159: # main - compile & link objects #
160: main: ./bin/main
161:
162: ./bin/main: ./lib/libxbee.so.1.0.1 ./bin/ ./main.c
163:     ${CC} ${CLINKS} ./main.c -o ./bin/main ${DEBUG}
164:
165: ./bin/:
166:     mkdir ./bin/
167:
168: ./lib/libxbee.so.1.0.1: ./lib/ ./obj/ ${addprefix ./obj/,${SRCS:.c=.o}} ./svn_version.c ./xbee.h
169:     gcc -shared -Wl,-soname,libxbee.so.1 -o ./lib/libxbee.so.1.0.1 ./obj/*.o -lrt
170: ifeq ($(strip $(wildcard ./lib/libxbee.so.1)),)

```

```

171:      ln ./libxbee.so.1.0.1 ./lib/libxbee.so.1 -sf
172: endif
173: ifeq ($(strip $(wildcard ./lib/libxbee.so)),)
174:      ln ./libxbee.so.1.0.1 ./lib/libxbee.so -sf
175: endif
176:
177: ./lib/:
178:      mkdir ./lib/
179:
180: ./obj/:
181:      mkdir ./obj/
182:
183: ./svn_version.c: api.c api.h globals.h xbee.h
184:      echo -n 'const char *svn_version(void) { return "" > ./svn_version.c
185: ifneq ($(strip $(wildcard /usr/bin/svnversion)),)
186:      svnversion -n . >> svn_version.c
187: else
188:      echo 'Unknown' >> svn_version.c
189: endif
190:      echo -n '";}' >> svn_version.c
191:      ${CC} ${CFLAGS} ${DEFINES} ${DEBUG} svn_version.c -o ./obj/svn_version.o
192:
193: ./obj/%.o: %.c %.h xbee.h globals.h
194:      ${CC} ${CFLAGS} ${DEFINES} ${DEBUG} $*.c -o $@
195:
196: ./obj/%.o: %.c xbee.h globals.h
197:      ${CC} ${CFLAGS} ${DEFINES} ${DEBUG} $*.c -o $@
198:
199:
200: # pdfs - generate PDFs for each source file #
201: ifneq ($(strip $(wildcard /usr/bin/ps2pdf)),)
202: ifneq ($(strip $(wildcard /usr/bin/enscript)),)
203: pdfs: ./pdf/ ${addprefix ./pdf/,${addsuffix .pdf,${PDFS}}}
204:
205: ./pdf/:
206:      mkdir ./pdf/
207:
208: ./pdf/makefile.pdf: ./makefile
209:      enscript ${ENSCRIPT} -Emakefile $< -p - | ps2pdf - $@
210:
211: ./pdf/%.pdf: %
212:      enscript ${ENSCRIPT} -Ec $< -p - | ps2pdf - $@
213:
214: ./pdf/%.pdf:
215:      @echo "**** Cannot make $@ - '$*' does not exist ****"
216: else
217: pdfs:
218:      @echo "WARNING: enscript is not installed - cannot generate PDF files"
219: endif
220: else
221: pdfs:
222:      @echo "WARNING: ps2pdf is not installed - cannot generate PDF files"
223: endif

```