

```
1: SRCS:=api.c
2: PDFS:=${SRCS} ${SRCS:.c=.h} makefile main.c xbee.h globals.h
3:
4: CC:=gcc
5: CFLAGS:=-Wall -Wstrict-prototypes -pedantic -c -fPIC ${DEBUG}
6: CLINKS:=-lm ./lib/libxbee.so.1.0.1 -lpthread ${DEBUG}
7: #DEBUG:=-g -DDEBUG
8: DEFINES:=
9:
10: ENSCRIPT:=-MA4 --color -f Courier8 -C --margins=15:15:0:20
11: ifneq (${strip $(wildcard /usr/share/enscript/mine-web.hdr)},)
12:     ENSCRIPT+= --fancy-header=mine-web
13: else
14:     ENSCRIPT+= --fancy-header=a2ps
15: endif
16:
17:
18: SRCS:=${sort ${SRCS}}
19: PDFS:=${sort ${PDFS}}
20:
21: .PHONY: all run new clean cleanpdfs main pdfs install install_su uninstall uninstall_su
22:
23:
24: # all - do everything (default) #
25: all: main
26:     @echo "**** Done! ****"
27:
28:
29: # run - remake main and then run #
30: run: main
31:     LD_LIBRARY_PATH=./lib:$LD_LIBRARY_PATH ./bin/main
32:
33:
34: # new - clean and do everything again #
35: new: clean all
36:
37:
38: # clean - remove any compiled files and PDFs #
39: clean:
40:     rm -f ./~
41:     rm -f ./obj/*.o
42:     rm -f ./lib/libxbee.so*
43:     rm -f ./bin/main
44:
45: cleanpdfs:
46:     rm -f ./pdf/*.pdf
47:
48:
49: # install - installs library #
50: install: /usr/lib/libxbee.so.1.0.1
51:
52: /usr/lib/libxbee.so.1.0.1: ./lib/libxbee.so.1.0.1
53:     @echo
54:     @echo
55:     @echo "#####"
56:     @echo "### To Install this library I need the root password please! ###"
57:     @echo "#####"
58:     su -c "make install_su --no-print-directory"
59:     @echo
60:     @echo
61:
62: install_su:
63:     cp ./lib/libxbee.so.1.0.1 /usr/lib/libxbee.so.1.0.1 -f
64:     chmod 755 /usr/lib/libxbee.so.1.0.1
65:     chown root:root /usr/lib/libxbee.so.1.0.1
66:
67:     cp /usr/lib/libxbee.so.1.0.1 /usr/lib/libxbee.so.1 -sf
68:     chown root:root /usr/lib/libxbee.so.1
69:
70:     cp /usr/lib/libxbee.so.1.0.1 /usr/lib/libxbee.so -sf
71:     chown root:root /usr/lib/libxbee.so
72:
73:     cp ./xbee.h /usr/include/xbee.h -f
74:     chmod 644 /usr/include/xbee.h
75:     chown root:root /usr/include/xbee.h
76:
77: uninstall:
78:     @echo
79:     @echo
80:     @echo "#####"
81:     @echo "### To Uninstall this library I need the root password please! ###"
82:     @echo "#####"
83:     su -c "make uninstall_su --no-print-directory"
84:     @echo
85:     @echo
```

```
86:
87: uninstall_su:
88:     rm /usr/lib/libxbee.so.1.0.1 -f
89:     rm /usr/lib/libxbee.so.1 -f
90:     rm /usr/lib/libxbee.so -f
91:     rm /usr/include/xbee.h -f
92:
93: # main - compile & link objects #
94: main: ./bin/main
95:
96: ./bin/main: ./lib/libxbee.so.1.0.1 ./bin/ ./main.c
97:     ${CC} ${CLINKS} ./main.c -o ./bin/main ${DEBUG}
98:
99: ./bin/:
100:     mkdir ./bin/
101:
102: ./lib/libxbee.so.1.0.1: ./lib/ ./obj/ ${addprefix ./obj/,${SRCS:.c=.o}} ./xbee.h
103:     gcc -shared -Wl,-soname,libxbee.so.1 -o ./lib/libxbee.so.1.0.1 ./obj/*.o -lrt
104: ifeq (${strip $(wildcard ./lib/libxbee.so.1)},)
105:     ln ./libxbee.so.1.0.1 ./lib/libxbee.so.1 -sf
106: endif
107: ifeq (${strip $(wildcard ./lib/libxbee.so)},)
108:     ln ./libxbee.so.1.0.1 ./lib/libxbee.so -sf
109: endif
110:
111: ./lib/:
112:     mkdir ./lib/
113:
114: ./obj/:
115:     mkdir ./obj/
116:
117: ./obj/%.o: %.c %.h xbee.h globals.h
118:     ${CC} ${CFLAGS} ${DEFINES} ${DEBUG} $.c -o $@
119:
120: ./obj/%.o: %.c xbee.h globals.h
121:     ${CC} ${CFLAGS} ${DEFINES} ${DEBUG} $.c -o $@
122:
123:
124: # pdfs - generate PDFs for each source file #
125: ifneq (${strip $(wildcard /usr/bin/enscript)},)
126: ifneq (${strip $(wildcard /usr/bin/ps2pdf)},)
127: pdfs: ./pdf/ ${addprefix ./pdf/,${addsuffix .pdf,${PDFS}}}
128:
129: ./pdf/:
130:     mkdir ./pdf/
131:
132: ./pdf/makefile.pdf: ./makefile
133:     enscript ${ENSCRIPT} -Emakefile $< -p - | ps2pdf - $@
134:
135: ./pdf/%.pdf: %
136:     enscript ${ENSCRIPT} -Ec $< -p - | ps2pdf - $@
137:
138: ./pdf/%.pdf:
139:     @echo "**** Cannot make $@ - '$*' does not exist ****"
140: else
141: pdfs:
142:     @echo "WARNING: ps2pdf is not installed - cannot generate PDF files"
143: endif
144: else
145: pdfs:
146:     @echo "WARNING: enscript is not installed - cannot generate PDF files"
147: endif
```