

Dec 05, 17 18:52	Makefile	Page 1/2
PRG	=lab5_48	
OBJS	=lab5_48.o lm73_functions.o twi_master.o uart_functions_m48.o	
SRCS	=lab5_48.c lm73_functions.c twi_master.c uart_functions_m48.c	
#MCU_TARGET	= atmega128	
MCU_TARGET	= atmega48	
#PROGRAMMER_TARGET	= m128	
PROGRAMMER_TARGET	= m48	
#agressive optimization		
OPTIMIZE	= -O2 # options are 1, 2, 3, s	
#optimize for small code		
#OPTIMIZE	= -Os # options are 1, 2, 3, s	
#OPTIMIZE	= -ls # options are 1, 2, 3, s	
F_CPU	= 8000000UL	
DEFS	=	
LIBS	=	
CC	= avr-gcc	
# Override is only needed by avr-lib build system.		
override CFLAGS	= -g -Wall \$(OPTIMIZE) -mmcu=\$(MCU_TARGET) \$(DEFS) -DF_CPU=\$(F_CPU)	
override LDFLAGS	= -Wl,-Map,\$(PRG).map	
OBJCOPY	= avr-objcopy	
OBJDUMP	= avr-objdump	
all: \$(PRG).elf lst text eeprom		
\$(PRG).elf: \$(OBJS)		
\$(CC) \$(CFLAGS) \$(LDFLAGS) -o \$@ \$^ \$(LIBS)		
#prevent confusion with any file named "clean"		
# "-" prevents erroring out with file not found		
.PHONY : clean		
clean:		
-rm -rf \$(PRG).o \$(PRG).elf		
-rm -rf \$(PRG).lst \$(PRG).map		
-rm -rf \$(PRG).srec \$(PRG)*.bin \$(PRG).hex		
-rm -rf \$(PRG)_eeprom.srec \$(PRG)_eeprom*.bin \$(PRG)_eeprom.hex		
-rm -rf *.d *.o *.map *.lst *.eeprom* *.elf *.hex *.bin *.srec		
all_clean:		
rm -rf *.o *.elf *.lst *.map *.srec *.bin *.hex		
#setup for usb programmer		
#hacked the permissions for stinkin' Redhat box		
program: \$(PRG).hex		
#chmod 644 \$(PRG).hex		
#avrdude -p \$(PROGRAMMER_TARGET) -c usbasp -e -U flash:w:\$(PRG).hex		
#avrdude -p \$(PROGRAMMER_TARGET) -c osuisp2 -e -U flash:w:\$(PRG).hex		
sudo avrdude -c usbasp -p m48 -e -U flash:w:\$(PRG).hex -v		
lst: \$(PRG).lst		
%.lst: %.elf		
\$(OBJDUMP) -h -S \$< > \$@		
#put dependencies on header files in here		

Dec 05, 17 18:52	Makefile	Page 2/2
#there must be a better way to do this!		
#lm73_functions.o thermo2.o : lm73_functions.h		
#lcd_functions.o thermo2.o : lcd_functions.h		
#thermo2.o : twi_master.h		
#Here is the pattern rule to generate a file of dependencies (i.e., a makefile)		
#called 'name.d' from a C source file +called 'name.c':		
#%.d: %.c		
# @set -e; rm -f \$@; \		
# \$(CC) -MM \$(CFLAGS) \$< > \$@.\$\$\$\$; \		
# sed 's,\(\$*\)\.o[:]*,\1.o \$@ : ,g' < \$@.\$\$\$\$ > \$@; \		
# rm -f \$@.\$\$\$\$		
#include the dependencies from the other makefiles		
#include \$(SRCS:.c=.d)		
text: hex bin srec		
hex: \$(PRG).hex		
bin: \$(PRG).bin		
srec: \$(PRG).srec		
%.hex: %.elf		
\$(OBJCOPY) -j .text -j .data -O ihex \$< \$@		
%.srec: %.elf		
\$(OBJCOPY) -j .text -j .data -O srec \$< \$@		
%.bin: %.elf		
\$(OBJCOPY) -j .text -j .data -O binary \$< \$@		
# Rules for building the .eeprom rom images		
eeprom: ehex ebin esrec		
ehex: \$(PRG)_eeprom.hex		
ebin: \$(PRG)_eeprom.bin		
esrec: \$(PRG)_eeprom.srec		
_eeprom.hex: %.elf		
\$(OBJCOPY) -j .eeprom --change-section-lma .eeprom=0 -O ihex \$< \$@ \		
{ echo empty \$@ not generated; exit 0; }		
_eeprom.srec: %.elf		
\$(OBJCOPY) -j .eeprom --change-section-lma .eeprom=0 -O srec \$< \$@ \		
{ echo empty \$@ not generated; exit 0; }		
_eeprom.bin: %.elf		
\$(OBJCOPY) -j .eeprom --change-section-lma .eeprom=0 -O binary \$< \$@ \		
{ echo empty \$@ not generated; exit 0; }		