**Ubuntu下用arm-none-eabi-gcc编译STM32F10x**

对于[Ubuntu](http://www.linuxidc.com/topicnews.aspx?tid=2) 14.04（我的是14.10），官方仓库里就有适用的交叉编译器

apt-get install gcc-arm-none-eabi

对于较低的版本，可以使用https://launchpad.net/gcc-arm-embedded/提供的二进制压缩包，

以及https://launchpad.net/~terry.guo/+archive/ubuntu/gcc-arm-embedded提供的软件源

add-apt-repository ppa:terry.guo/gcc-arm-embedded  
apt-get update  
apt-get install gcc-arm-none-eabi

以下是项目的makefile，CMSIS中核心支持使用2.00版本，硬件支持使用3.5版本，外设驱动为3.5版

makefile使用find找到所有.c和.s文件，根据自动依赖进行编译。编译出来的文件，根据stm32\_f103ze\_gcc.ld生成.bin和.hex文件

#######################################

# binaries

#######################################

BINPATH = /home/byhu/work/byhu/gcc-arm-none-eabi\_byhu/bin

######################################

# target

######################################

TARGET = stm32\_onLinux\_demo

########################################################################

export CC = $(BINPATH)/arm-none-eabi-gcc

export AS = $(BINPATH)/arm-none-eabi-as

export LD = $(BINPATH)/arm-none-eabi-ld

export OBJCOPY = $(BINPATH)/arm-none-eabi-objcopy

TOP=$(shell pwd)

INC\_FLAGS= -I $(TOP)/CORE \

-I $(TOP)/FWLIB/inc \

-I $(TOP)/HARDWARE \

-I $(TOP)/SYSTEM \

-I $(TOP)/USER

export CFLAGS= -W -Wall -g -mcpu=cortex-m3 -mthumb -D STM32F10X\_HD -D USE\_STDPERIPH\_DRIVER $(INC\_FLAGS)

ASFLAGS= -W -Wall -g -Wall -mcpu=cortex-m3 -mthumb

########################################################################

C\_SRC=$(shell find ./ -name '\*.c')

C\_OBJ=$(C\_SRC:%.c=%.o)

C\_DEP=$(C\_SRC:%.c=%.cdep)

ASM\_SRC=$(shell find ./ -name '\*.s')

ASM\_OBJ=$(ASM\_SRC:%.s=%.o)

ASM\_DEP=$(ASM\_SRC:%.s=%.adep)

########################################################################

.PHONY: all clean

all:$(C\_DEP) $(ASM\_DEP) $(C\_OBJ) $(ASM\_OBJ)

$(LD) $(C\_OBJ) $(ASM\_OBJ) -T stm32\_f103ze\_gcc.ld -o $(TARGET).elf

$(OBJCOPY) $(TARGET).elf $(TARGET).bin -Obinary

$(OBJCOPY) $(TARGET).elf $(TARGET).hex -Oihex

###################################

%.cdep:%.c

$(CC) -MM $< > $@ $(CFLAGS)

sinclude $(C\_DEP)

$(C\_OBJ):%.o:%.c

$(CC) -c $< -o $@ $(CFLAGS)

####################################

%.adep:%.s

$(CC) -MM $< > $@ $(ASFLAGS)

sinclude $(ASM\_DEP)

$(ASM\_OBJ):%.o:%.s

$(AS) -c $@ -o $@ $(ASFLAGS)

####################################

clean:

@for i in $(shell find ./ -name '\*.o');do if [ -e $${i} ];then rm $${i};fi;done

@for i in $(shell find ./ -name '\*.cdep');do if [ -e $${i} ];then rm $${i};fi;done

@for i in $(shell find ./ -name '\*.adep');do if [ -e $${i} ];then rm $${i};fi;done

stm32\_f103ze\_gcc.ld的内容

\_estack = 0x20000400;

MEMORY

{

FLASH\_ON\_CHIP (rx) : ORIGIN = 0x08000000, LENGTH = 512K

SRAM\_ON\_CHIP (rwx) : ORIGIN = 0x20000000, LENGTH = 64K

}

SECTIONS

{

.text : {

KEEP(\*(.isr\_vector))

\*(.text\*)

\*(.rodata\*)

\_etext = .;

} > FLASH\_ON\_CHIP

\_sidata = .;

/\* .data : AT(ADDR(.text) + SIZEOF(.text)) {\*/

.data : AT(\_sidata) {

\_sdata = .;

\*(vtable)

\*(.data\*)

\_edata = .;

} > SRAM\_ON\_CHIP

.bss : {

\_sbss = .;

\*(.bss\*)

\*(COMMON)

\_ebss = .;

} > SRAM\_ON\_CHIP

}

 向上查找makefile并执行的shell脚本，可以用在geany的编译快捷键中

cd $1;while [ ! -e ./Makefile ] ; do cd ..; path=`pwd` ; if [ "$path" = "/" ] ; then break; fi;done;if [ -e ./Makefile ] ;then make $2;fi