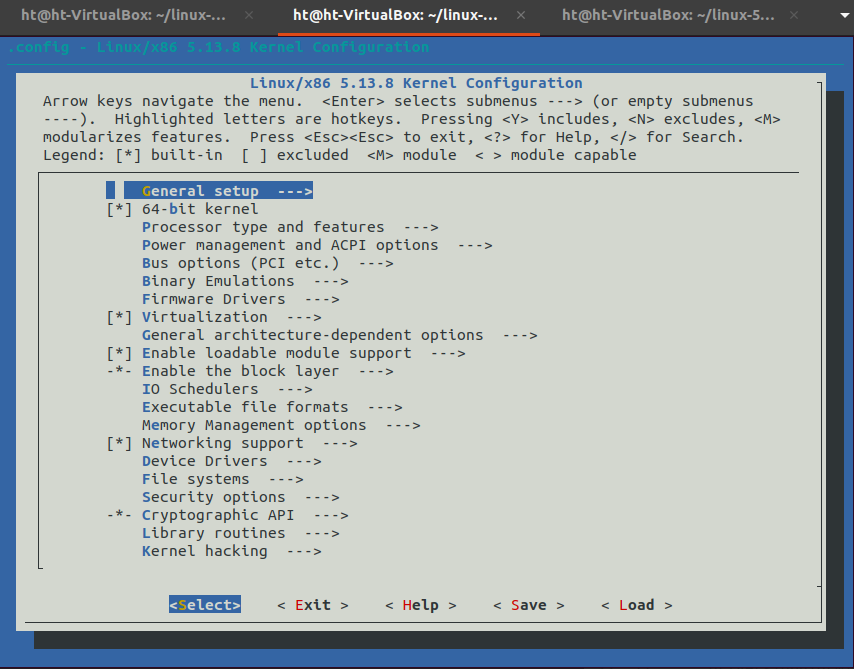
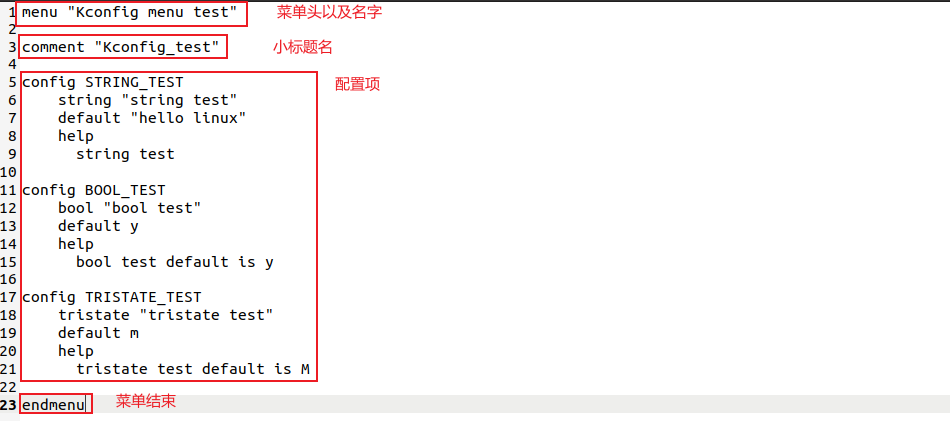
在linux源码下使用make menuconfig，就可进入下图所示的界面，可视化勾选配置项是否编译，编译进内核或者模块。



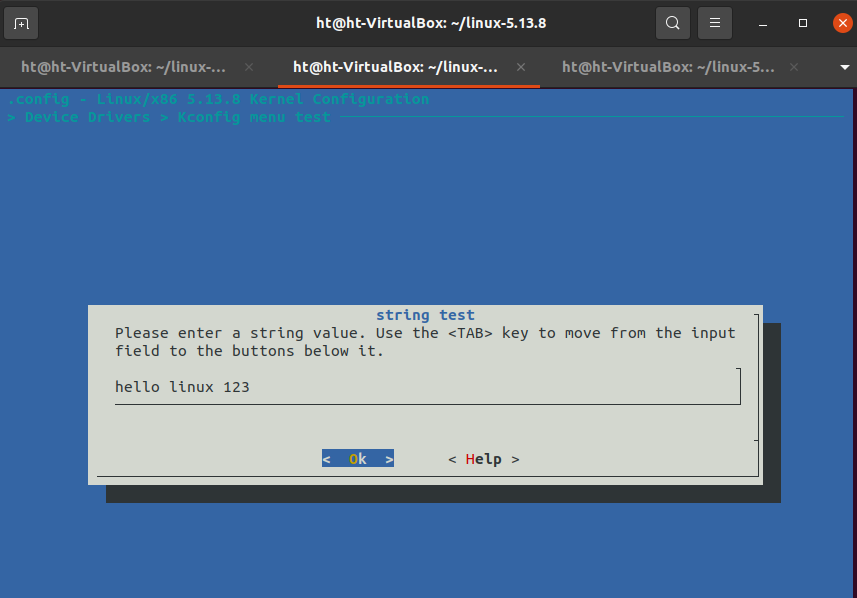
1.1自建Kconfig

对其中的配置项进行增删改查在对应的Kconfig文件中。下图为基础的Kconfig文件。



其中每个配置项以conmfig开头，下一行为配置项类型，简单描述，下一行为默认值。help下一行在图形化可查看。

String类型，选中后可以修改值



最主要使用的为tristate，有三个值y，m，n,分别为编译进内核，编译为模块，不编译。

还有其他的依赖和反依赖可在需要时在网上查阅。

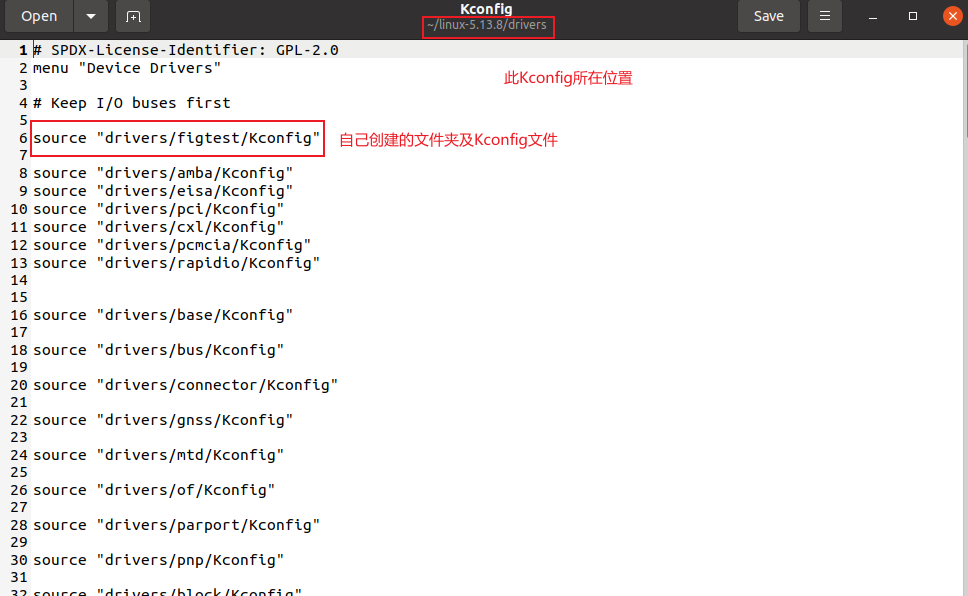
查看配置项的help:



实际效果为：



从图中可见此菜单路径在device drivers下面，在drivers文件夹下的Kconfig文件中引入自建文件夹及Kconfig即可

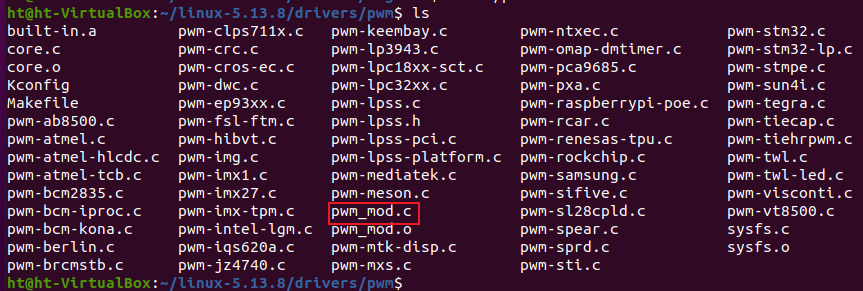


此时menuconfig中添加的配置项是无效的，没有对应.c文件。

1.2添加配置项对应.c文件

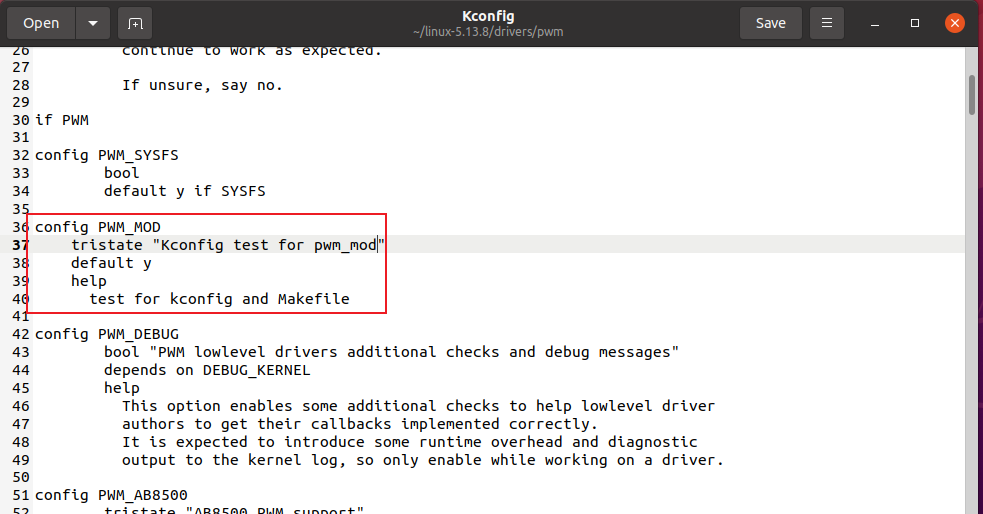
一般只是需要在对应驱动的大文件夹下添加额外的驱动文件，所以一般在已有的文件上进行修改。下面将添加一个pwm的驱动在drivers/pwm文件夹下。

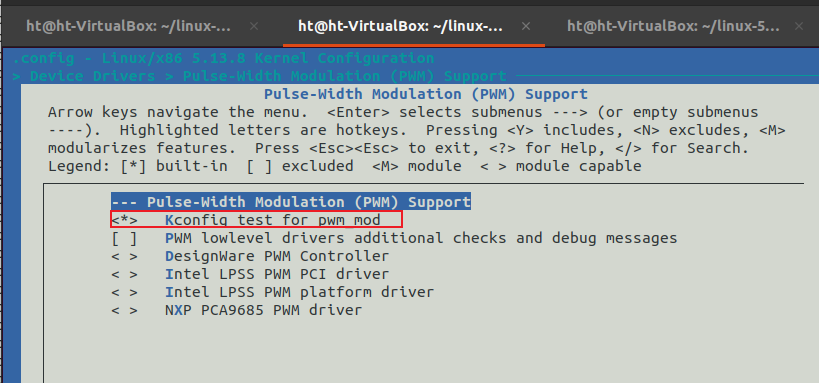
1.2.1，将驱动源码.c文件放在文件夹中。



1.2.2在Kconfig中添加配置项

注：配置项的名称最好为源码文件的大写，便于记忆，添加在文件中的位置对应为menuconfig中的顺序。





1.2.3 在Makefile中将配置项和源码文件连接起来

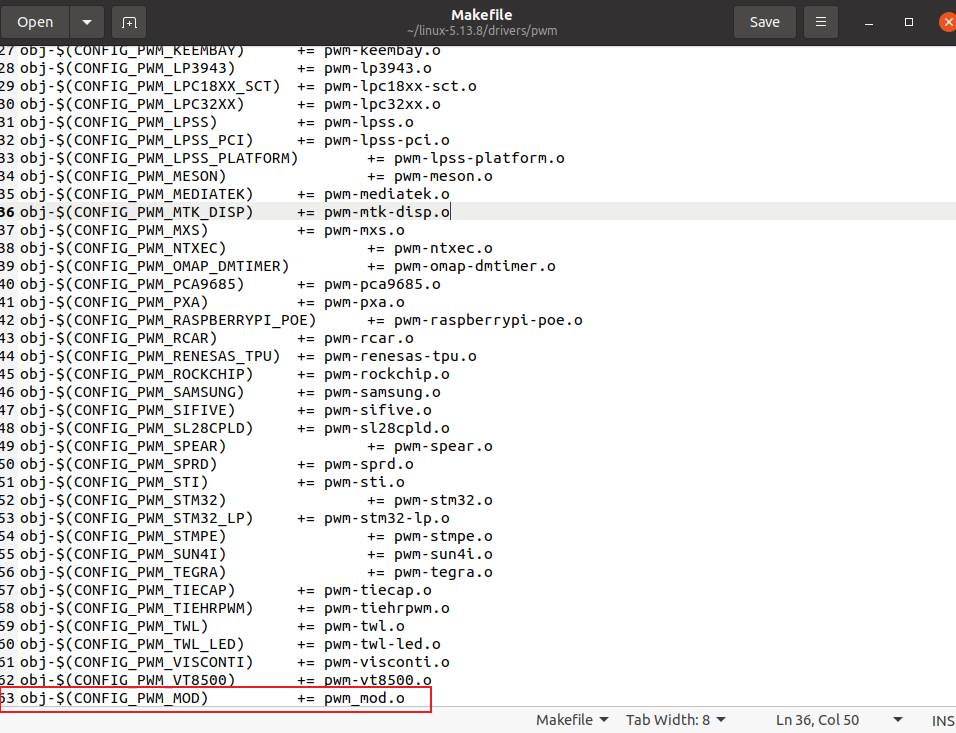
如下图所示，在Makefile文件中，添加了一行：

obj-$(CONFIG\_PWM\_MOD) += pwm\_mod.o

在自行添加时只需修改前半括号中为CONFIG\_ + 配置项名称，不能有误差。

将后半修改为源码名称.o文件（如pwm\_mod.c --> pwm\_mod.o）。

进行如上修改后，menuconfig中的配置项就跟存在的代码联系起来了。



1.2.3，编译内核

查看编译内核输出可见确实被编译，源码位置确实产生了.o文件。此驱动就被编译到了内核中。

