

## 关于 Keil C166 的使用

单片机开发除了必要的硬件同样也离不开软件，我们写的程序要转化成 CPU 所能执行的机器码有两种方法：一是手工汇编，二是机器汇编。机器汇编是通过汇编软件将源程序编程机器码。Keil 软件是目前最流行的开发单片机的软件工具，Keil 编译器提供了包括 C 编译器，宏汇编，连接器，库管理和一个功能强大的仿真调试器等在内的完整的开发方案。通过一个集成开发环境（uVision）将这些部分组合在一起。

KEIL 软件有支持 8 位单片机的 Keil C 51 系列和支持 16 位单片机的 Keil C 166 系列。在项目开发过程中并不是仅有一个源程序就足够了，还要为项目选择 CPU 确定编译，汇编，连接的参数，指定调试的方式，有一些项目还会由多个文件组成。为管理和使用方便，Keil 引入了工程（Project）概念。将这些参数和所需要的文件都加在一个工程中，只能对工程进行编译和连接等操作。

### 一、工程的详细设置：

以下针对在使用 XC164CS 评估板过程中在 Keil C166 环境下的一些设置谈一下。首先点击 Project 窗口中的 Target1 Project->Option for Target1 “target 1”即出现对工程设置的对话框。菜单如下图 1：

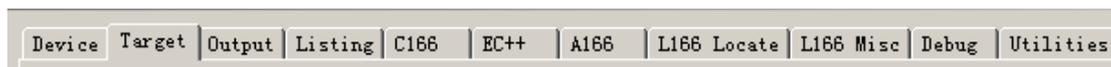


图 1

以下针对各个标签详细说明：

#### Device

选择所使用的 CPU（即所选用的芯片）。KEIL 支持很多种 CPU，当选中一款芯片以后右侧窗口还会有相应的芯片介绍。此处选择 Infineon XC164CS 系列芯片作为 CPU。

#### Target

窗口设置如下：

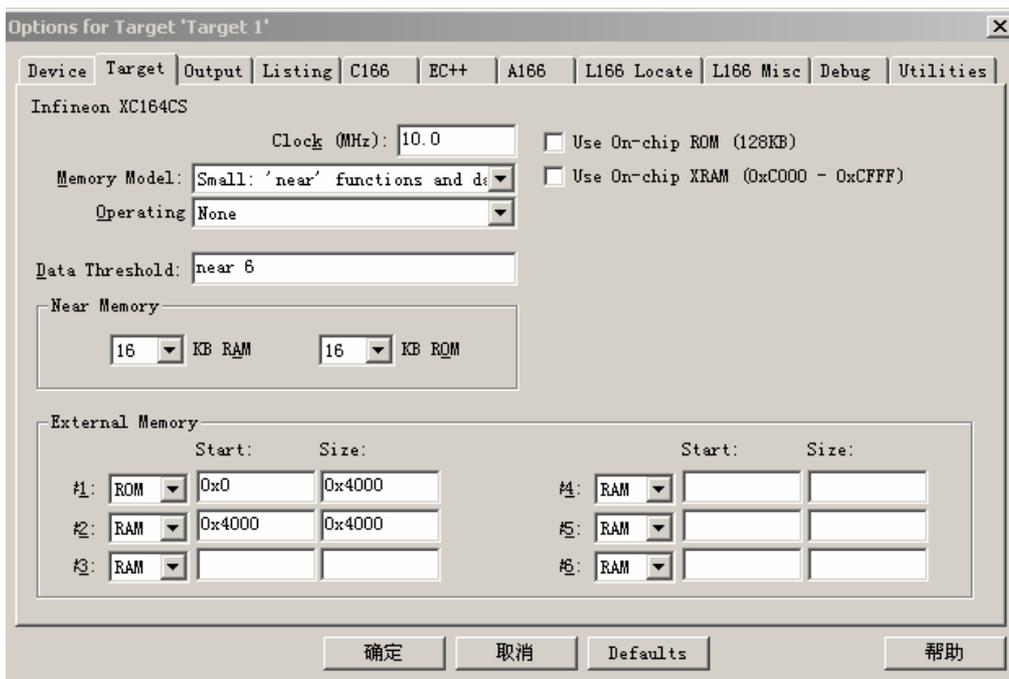


图 2

这里可以设置时钟频率，片内和片外资源的选择及地址的设置。

其中 Memory Model 用于设置 RAM 使用情况，KEIL C 166 编译器可支持 7 种存储类型。

- TINY CPU 处于非分段工作方式下，可产生高效的 16 位线性地址，并把代码和数据限制在 64KB 种。不能使用 far, huge, xhuge 存储类型。
- SMALL 使用分段 CPU 方式，同样产生高效的代码，但代码和数据不再限制再 64KB 中，用户可通过 far, huge, xhuge 引用变量和函数
- COMPACT 一般用于代码少而数据多的场合
- HCOMPACT 一般用于代码多而数据少的场合
- MEDIUM 所有的函数调用默认为 far 调用，一般用于代码多而数据少的场合
- LARGE 所有的函数调用默认为 far 调用，一般用于代码和数据多的场合
- HLARGE 所有的函数调用默认为 far 调用，一般用于代码和数据多的场合，不适合于 C166 系列 CPU

在仿真过程中如果使用片内 FLASH，则选中 Use On-chip ROM

在仿真过程中如果使用片外 RAM，则取消 Use On-chip ROM 复选框并设置 ROM 和 RAM 空间起始地址及大小。注意片外 RAM 起始地址为 0x0000。此例中设置 ROM 起始地址为 0x0000 大小为 2K，设置 RAM 起始地址为 0x4000 大小为 2K。

### Output

此页面有多个选择项，其中 Create Hex file 用于生成可执行代码文件（可以用编程器写入单片机芯片的 HEX 格式文件，文件的扩展名为 HEX），默认情况下该项未被选中，如果要写片做硬件试验，就必须选中该项。其他的取默认设置即可以。

### Listing

该标签页用于调整生成的列表文件选项。该页用于对列表文件的内容和形式进行细致的调节，其中比较常用的选项是“C Compile Listing”“Assamble Code”项，选中该项可以在列表文件中生成 C 语言源程序所对应的汇编代码。

### C166

该标签用于对 KEIL C 166 编译器的编译过程进行控制，其中比较常用的是“Code Optimization”组，level 设置优先等级，在对源程序进行编译时可以对代码进行优化，系统默认为第六级，一般不用修改，如果编译程序时出错可以尝试降低优化等级。Emphasis 选择编译优化方式，第一项为代码项优化（生成代码量小），第二项为速度优化（最终生成代码速度快），第三项为缺省，默认为速度优化，可根据需要更改。

EC++ A166 L166 Locate L166 Misc 是对编译环境的一些设置，可按默认选项

### Debug

该标签是对仿真的一些设置。选择 Use Simulator 则应用软件进行仿真，这里可对启动类型，总线形式，时钟信号及片选信号等进行设置。

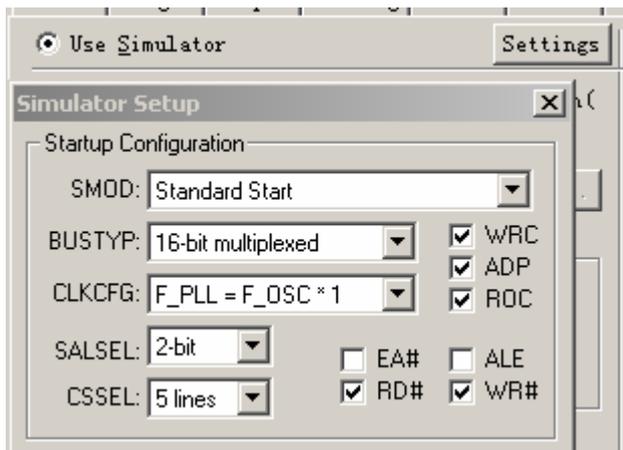


图 3

如果想实现硬件仿真则选择 Use OCDS Driver for XC16x ,点击 Settings 按钮弹出相应设置窗口。Port 默认为 LPT(0x378) ,配置均可按默认设置。

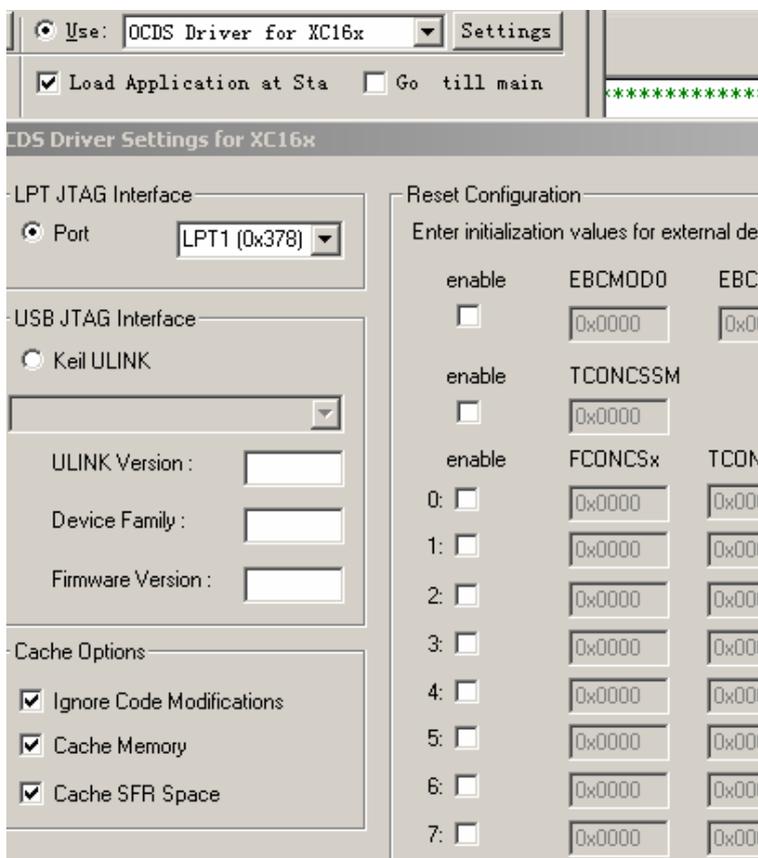


图 4

**Utilities**

当使用片内 FLASH 进行仿真时，需要选择 Use Target Driver for Flash Programming 下拉菜单中的 OCDS Driver for XC16x ,在进行 Configure Flash Download 时可设置 FLASH 下载功能设置。下图为在使用 XC164CS 评估板进行片内 FLASH 仿真时的 FLASH 设置。

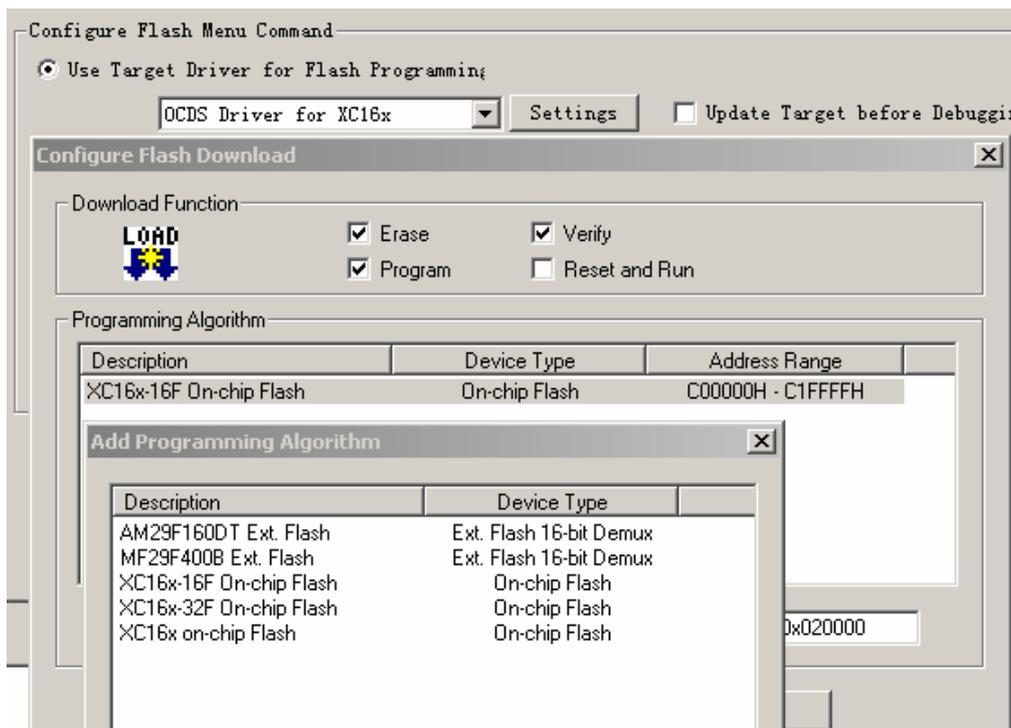


图 5

## 二．仿真

以上即完成了工程的相关设置，接下来可以进行编译，连接。选择菜单 Project→Build target 或单击  图标对当前工程进行连接。编译过程中的信息将出现在输出窗口中的 Build 页，如果源程序中有语法错误，会有错误报告出现，单击该行会有相应的错误报告出现。编译成功后提示获得\*.hex 文件，该文件可被编译器读入并写入芯片中，同时还产生了一些其他相关文件可用于 Keil 的仿真与调试。

在对工程成功编译，连接后，按 F5 或点击菜单 Debug Start/Stop Debug Session 或单击  图标即可进入调试状态。

### ➤ 软件仿真

即则按图 3 所示对工程进行设置

### ➤ 硬件仿真

1. 片内 FLASH 仿真（编译通过后要先 DOWNLOAD 到 FLASH 然后在仿真）
  - 如图 2 选中 Use On-chip ROM；
  - 如图 4 选择 Use OCDS Driver for XC16x；
  - 如图 5 所以设置 FLASH（注意起始地址）；
2. 片外 RAM 仿真
  - 如图 2 取消选中 Use On-chip ROM，设置片外 ROM，RAM 起始地址和大小；
  - 如图 4 选择 Use OCDS Driver for XC16x；

### 三. 编译

#### 1. 单步运行与全速执行

下图 6 为编译窗口。在程序调试过程中可根据需要选择全速运行和单步运行。这里需要明确两个概念,单步执行与全速执行。全速执行指一程序执行完后紧接着执行下一程序,中间不间断,这样执行速度很快,可以看到整段程序执行的效果,即最终结果正确与否;如果程序有错则很难确定错误出在哪一行。单步执行指每次执行一程序,执行完后即停止,等待命令执行下一程序,这样可以看到程序执行的结果即是否与所要得到的结果相同,方便发现错误。在整个程序调试过程中这两种方法都要用到。

如果只靠单步运行来查错有时是困难的,比如遇到循环语句或延迟程序时不可能靠重复执行单步运行来跳过循环段,这时可以采取以下几种方法:

- 1) 用鼠标在子程序的最后一行点一下,使光标定位在该行,然后用菜单 Debug->Run to Cursor line 即可全速执行完黄色箭头与光标之间的程序段。
- 2) 在进入子程序后,点击 Debug->Step Out of Current Function 则全速执行完调试光标处的子程序或子函数后指向主程序中的下一行。

#### 2 设置断点

程序调试时,一些程序行需要满足一定的条件才执行(如某变量达到一定值,按键被按下,中断产生等),这些条件往往是异步发生和难以预测的。这时就要使用到程序调试中一个非常重要的方法——断点设置。设置断点的方法很多,可以通过在某程序行首双击即可设置断点或通过菜单 Debug->Insert/Remove Breakpoint 来设置。

KEIL 还提供了多种设置断点的方法,单击菜单 Debug->Breakpoints 将弹出对话框来对断点进行设置。

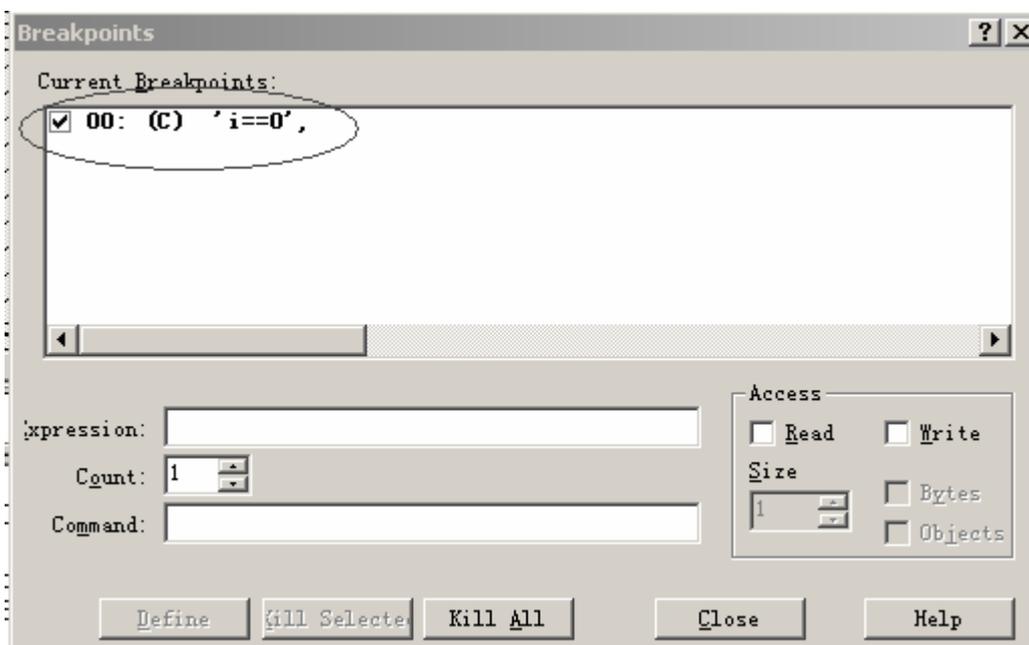


图 6

在 Expression 中添加条件语句,这里可以使用==,<=,>=,!,& (两值按位与),&& (两值与)等运算符。也可键入如“Delay”的字样来设置断点。

### 3. 程序调试时常用窗口

#### 1) 存储器窗口

存储器窗口可以显示任何地址的值，在 Address 栏中添加地址即可查看相应的值。

#### 2) 工程窗口寄存器页

该寄存器页包括了当前的工作寄存器组和系统寄存器，当程序执行到对该寄存器操作时，该寄存器会以反色（蓝底白字）显示

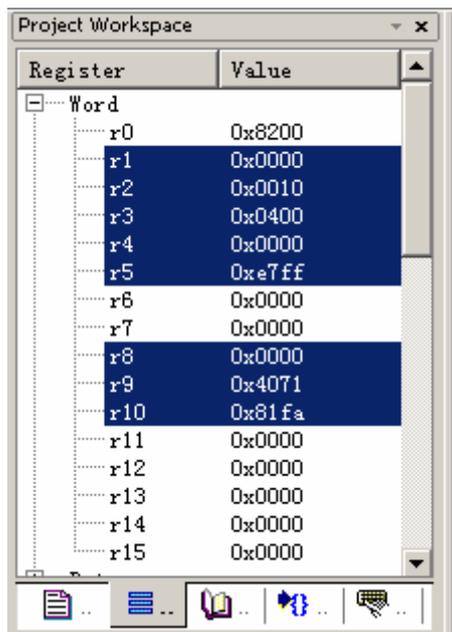


图 7

#### 3) 观察窗口

如果需要观察其他寄存器的值或在高级语言编程时查看变量则要借助于观察窗口了，按 F2 或双击鼠标左键，将需要查看的寄存器或变量输入，即可在程序运行过程中看到其变化过程。

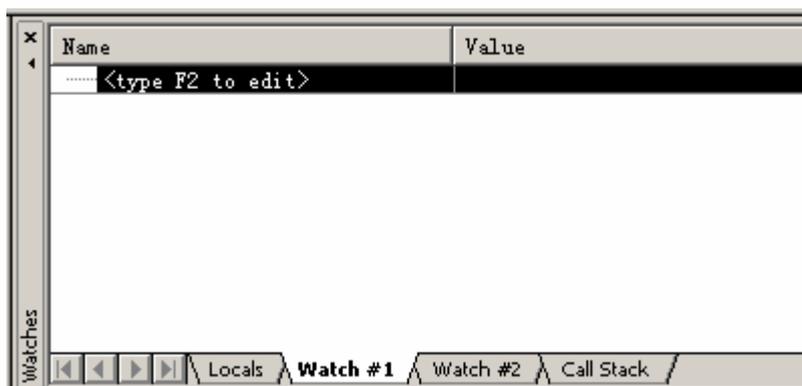


图 8

## 四. 辅助工具

KEIL 提供了一些辅助工具，如外围接口，性能分析，变量来源分析，代码作用分析等。这部分工具并不是用于程序调试的，而是可以帮助我们完成程序的调试和程序性能的分析。

为了能够比较直观的了解单片机的定时器，串口，并行口，中断等外设，KEIL 提供了一些外围接口的对话框，通过 Peripherals 菜单的下拉菜单选择所要查看的外设资源。在编译，连接进入调试后，在全速运行时可以看到相应外设的变化情况。如图 9 为外围接口资源。

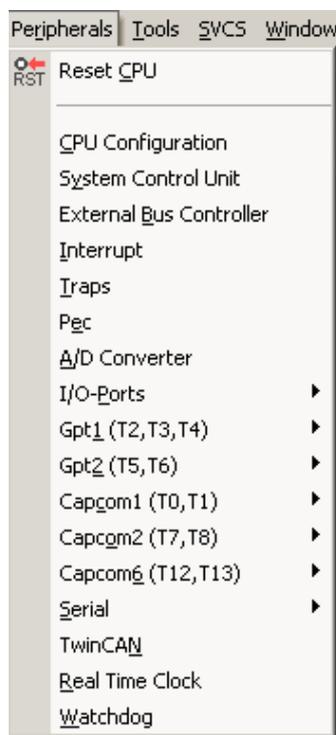


图 9

以上只是我在学习 XC164CS 单片机及使用 Keil C166 的一些体会,希望与大家在论坛上多多交流，共同进步！

作者：尹颖

版权所有：北京乾勤科技发展有限公司