

IAR 嵌入式工作台工具概述

本章概述了IAR 嵌入式工作台®的不同工具。

集成开发环境(IDE)

IDE是个框架,它集中了你构建应用的所有工具: C/C++ 编译器, 汇编器, 链接器, 库工具, 编辑器, 项目管理器以及 IAR C-SPY® 调试器。

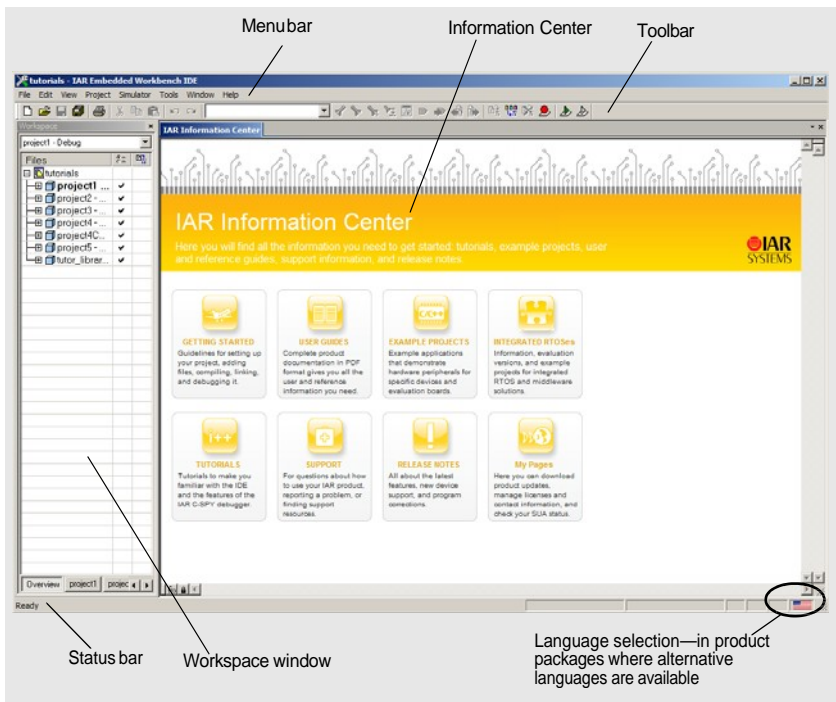
产品软件包的工具链支持特定的微型控制器。同时, IDE包含各种微型控制器的多个工具链。这意味着你如果已为几个微型控制器安装了 IAR 嵌入式工作台, 你可以选择要开发的微型控制器。

注意: 编译器, 汇编器, 链接器以及C-SPY调试器也可以在命令行环境下运行, 如果你想在一个已经建成的项目环境中用它们做外部工具。

启动IDE:

点击Windows**开始**菜单的开始按钮, 或者双击工作台文件名(文件扩展名`eww`), 或者使用安装在你的 IAR 嵌入式工作台 `common\bin` 目录的文件 `IarIdePm.exe`。

集成开发环境主窗口打开后:



当你首次打开 IAR 嵌入式工作台时, IDE主窗口会显示 IAR 信息中心。 这里你可以找到所有入门需要的信息: 教程, 项目示例, 用户指南, 支持信息和版本说明。

配置IDE

有很多方式使你的集成开发环境满足你的偏好和要求:

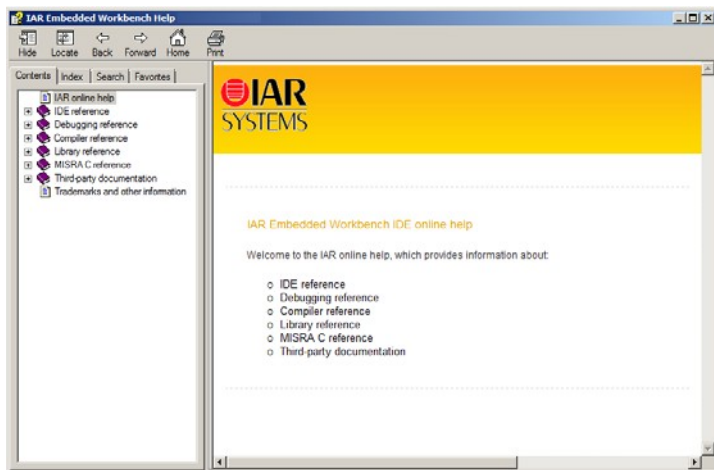
- 组织窗口 — 你可以**停泊**窗口在指定的地方, 并把它们组织在标签组里。你还可以让窗口**浮动**, 也就是让它总是在其他窗口的上面。如果你改变浮动窗口的大小和位置, 其他当前打开的窗口不受影响。状态条, 位于主窗口的底部, 包含如何安排窗口等有用的帮助信息。

- 用外部的工具扩展工具链，例如一个版本修订控制系统或你选择的编辑器。你还可以添加IAR visualSTATE到工具链里,这意味着你可以在IDE里将状态机编制图直接加进你的项目里。
- 从工具菜单调用外部工具
- 定制你自己的IDE, 例如可以用命令:
 - 配置编辑器
 - 改变命令字体
 - 改变绑定的键值
 - 使用自己选择的外部的编辑器
 - 配置项目构建指令,命令
 - 配置在信息窗口输出的数量。

查看在线帮助系统:

选择 **Help>Content** 也可以在IDE的窗口或者对话框中点击并且按下F1键。

在线帮助系统显示出如下:



在这里你可以找到上下文相关的帮助信息,例如:

- IDE 和 C-SPY
- 编译器
- 库
- MISRA-C.

IAR C/C++ 编译器

编程语言

大多数产品软件包有两种高级编程语言供你用 IAR C/C++ 编译器:

C, 最广泛使用在嵌入式系统工业中的高级编程语言。遵循以下这些标准就可以搭建独立的应用:

- 标准C — 也就是著名的C99。此后,本指南中,该标准被称为标准C。
- C89 — 也就是著名的C94, C90, C89和ANSI C。当启用MISRA C时需要这个标准。

C++ (取决于你的产品软件包)。这些标准任何一个都可以用:

- 标准C++ — 可以用于支持不同等级的异常和运行时类型信息(取决于你的产品软件包)。
- 嵌入式 C++ (EC++), C++程序设计标准的子集。它是工业联盟, 嵌入式C++技术委员会, 定义的。
- IAR扩展的嵌入式C++,具有附加功能,例如支持全模板,多重继承(取决于你的产品软件包), 支持命名空间, 新的转型操作符,以及标准模板库(STL)。

MISRA-C

MISRA-C是一套规则, 配套用于安全攸关系统。制定MISRA-C规则意味着强制执行C编程语言的ISO标准中的更严格的安全措施。

根据你的产品软件包, 对MISRA-C:2004和MISRA-C:1998都可能支持。

编译程序扩展

编译器提供了C和C++语言的标准功能,以及大量的扩展:

C 语言扩展 可分成三组:

- 嵌入式系统编程扩展 — 尤其是为你正在使用的特定的微控制器的

高效嵌入式编程量身定制的扩展，具有代表性的是满足内存局限或者声明诸如中断这样特殊的功能类型。

- 对标准C的放宽——也就是，放宽一些轻微的标准C的问题，还有一些有用但次要的语法扩展。

Pragma 指令 是一种C标准定义的机制，用于供应商指定的扩展，以保证源码总是可移植的。预定义的指令控制着编译器的行为，例如怎样分配内存，是否允许扩展的关键字和它是否输出报警消息。

pragma 指令在编译器里始终是可用的。

预处理功能，例如：

- 预定义的预处理器符号让你查看编译时间环境，例如编译的时间和不同的编译器设置。
- 用户定义预处理器符号，由编译器选项或者在IDE中定义，除了 #define 指令。

访问微型控制器的底层特征 是必需的。编译器支持几种方法来干这件事：内置函数——提供直接访问低层处理器的操作——混合C和汇编语言模块以及内联汇编语言。你要仔细选择用那种方法。

IAR 汇编器

IAR 汇编器是种重定位的宏汇编器，具有一套针对你使用的微型控制器的多用途的指令和表达式运算符。汇编器以内置C语言预处理器并支持条件汇编为特色。

汇编器将符号化的汇编语言助记符翻译成可执行的机器代码。要编写高效的汇编程序应用，你应该熟悉所用的微型控制器的架构和指令集。

即使你不想完全用汇编语言编写应用程序，有时你会发现必须用汇编写一部分代码，例如，用到微型控制器内部的机理，需要精确的时间和特殊的指令序列的时候。

IAR 链接器及其相关工具

基于你的产品软件包，IAR 嵌入式工作台要么自带 XLINK 链接器要么自带 ILINK 链接器。

它们链接小的, 单个文件的, 纯汇编的应用程序和链接大的, 重定位的, 多模块的, C/C++或混合C/C++及汇编的应用程序, 样样都合适。

两个链接器都使用*配置文件*, 你可以用它详细说明你的目标系统的内存映射的代码和数据区域的单独的位置, 让你完全管理代码和数据的放置。

链接之前, 链接器对全部输入文件中的所有符号执行完整的依赖解析, 不受输入次序 (除了库外) 的影响。还会检查编译器所有模块设置的一致性并确保使用了C或C++运行时库的正确版本和变体。

链接器只会自动加载这样一些库模块 — 用户库和标准C或C++库的变体— 它们是你正在链接的应用程序确实需要的。更准确的说, 只有实际使用的库模块的函数会加载。

IAR ILINK 链接器 将一个或多个可重定位的目标文件与一个或多个目标库中选出的部件组合生成一个可执行的映像。

最终ILINK生成的输出是一个绝对目标文件, 包含ELF (含DWARF调试信息) 格式的可执行映射。文件可以下载到C-SPY或者其他支持ELF/DWARF格式的调试器, 或转换成任意合适的格式后编程写入EPROM。

含有各种处理ELF文件的工具, 例如, 解压缩软件, ELF查看器和格式变换器。

基于你的产品软件包, ILINK还可以为许多应用程序计算其使用堆栈的最大值。

IAR XLINK 链接器 将IAR系统编译器或者汇编器产生的一个或多个可重定位的目标文件组合, 为你正在使用的微控制器生成机器码。XLINK可以生成超过30种以上的工业标准的加载器格式, 还不包括C-SPY调试器使用的UBROF格式。

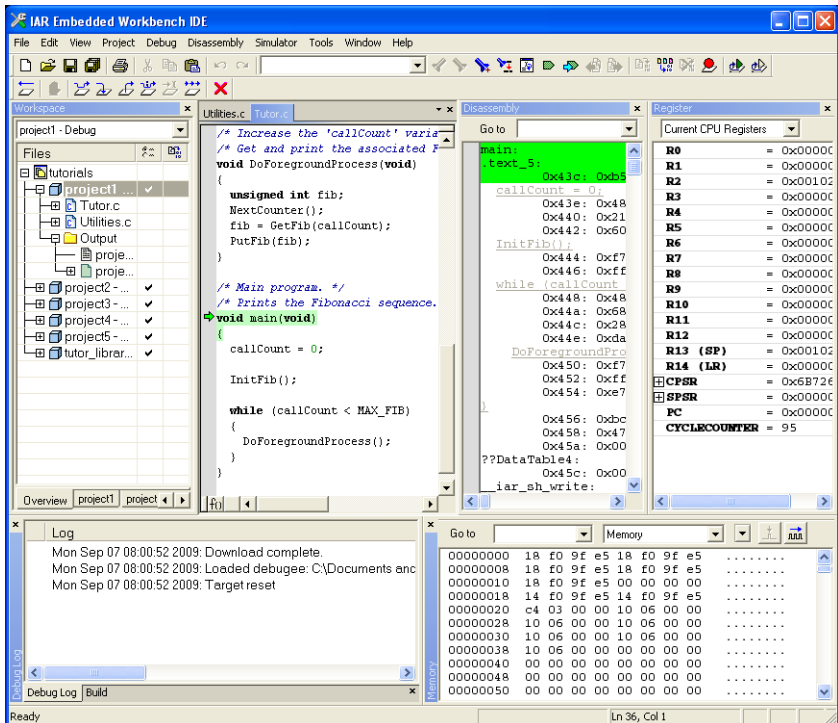
XLINK 还执行横跨所有模块的全部C/C++类型检查。

最终XLINK生成的输出是一个绝对的, 目标可执行的目标文件, 可下载到微控制器或者硬件仿真器。可选择的是, 根据你选择的输出格式, 输出文件可以含有调试信息。

包含了处理库的工具XAR和XLIB。

IAR C-SPY 调试器

IAR C-SPY 调试器是个嵌入式应用开发的高级语言调试器。



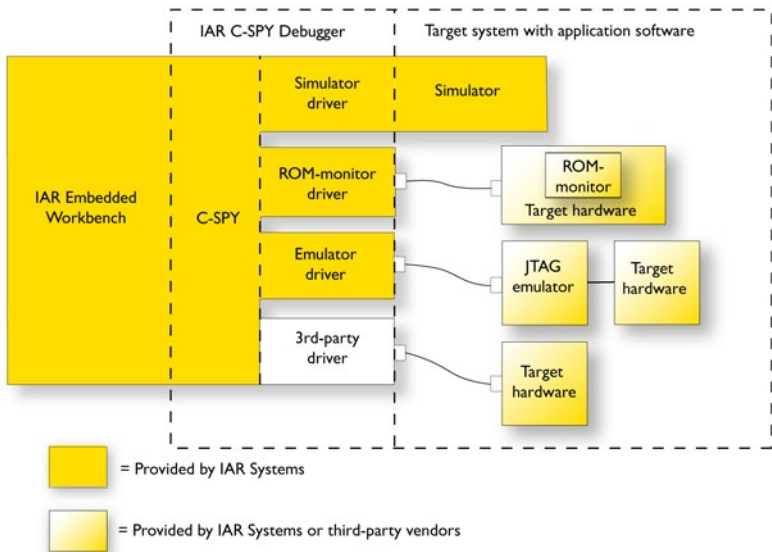
它就是设计成和IAR系统编译器和汇编器一起使用的，与IDE完美集成，供开发和调试之间的无缝切换。它将给你提供如下所述的可能：

- 调试的同时进行编辑。在调试阶段，你可以在用于控制调试的源代码窗口直接改正它。改变会包含在下一次项目重建中。
- 启动调试器之前设置源代码的断点。即使插入了新增的代码，源代码中的断点依然与同一段源代码相关联。

C-SPY 由提供基本的调试功能的通用部件和驱动组成。C-SPY驱动是控制目标系统并与之通信的部件。驱动还提供了用户界面——专门的菜单，

窗口，以及对话框— 对接目标系统提供的功能，如,特殊的断点。

该图展示了 C-SPY 及可能的目标系统的概况：



基于你的产品软件包, C-SPY 可能有仿真驱动和可选的各种硬件调试器系统的仿真驱动可用。

C-SPY 在本指南中有更详细的探讨, 见 *调试*, 第45页。

C-SPY 插件模块

C-SPY 被设计成了模块化的架构。 SDK (软件开发工具包) 能够以插件模块的形式为调试器实现附加功能。这些模块可以集成进入IDE。

插件模块由IAR系统公司提供,也可由第三方供应商供应。这种模块的例子有:

- 代码覆盖, 符号和堆栈插件都完美集成在了IDE中。
- 各种C-SPY调试器驱动都使用某种调试系统。

RTOS 插件模块供实时操作系统的系统感知调试。

- C-SPYLink 是 IAR visualSTATE 和IAR嵌入式工作台之间的桥梁,确保能够,除了普通C级别的符号调试外,在C-SPY中直接调试高级状态机。

有关C-SPY SDK的更多信息,请联系IAR系统公司。