

常见 ARM 编译器简介

ARM 应用软件的开发工具根据功能的不同，分别有编译软件、汇编软件、链接软件、调试软件、嵌入式实时操作系统、函数库、评估板、JTAG 仿真器、在线仿真器等，目前世界上约有四十多家公司提供以上不同类别的产品。

用户选用 ARM 处理器开发嵌入式系统时，选择合适的开发工具可以加快开发进度，节省开发成本。因此一套含有编辑软件、编译软件、汇编软件、链接软件、调试软件、工程管理以及函数库的集成开发环境（IDE）一般来说是必不可少的，至于嵌入式实时操作系统、评估板等其他开发工具则可以根据应用软件规模和开发计划选用。

使用集成开发环境开发基于 ARM 的应用软件，包括编辑、编译、汇编、链接等工作全部在 PC 机上即可完成，调试工作则需要配合其他的模块或产品方可完成。

（一）SDT

ARM SDT 的英文全称是 ARM Software Development Kit，是 ARM 公司(为方便用户在 ARM 芯片上进行应用软件开发而推出的一整套集成开发工具。ARM SDT 经过 ARM 公司逐年的维护和更新，目前的最新版本是 2.5.2，但从版本 2.5.1 开始，ARM 公司宣布推出一套新的集成开发工具 ARM ADS 1.0，取 ARM SDT 而代之，今后将不会再看到 ARM SDT 的新版本。ARM SDT 由于价格适中，同时经过长期的推广和普及，目前拥有最广泛的 ARM 软件开发用户群体，也被相当多的 ARM 公司的第三方开发工具合作伙伴集成在自己的产品中，比如美国 EPI 公司的 JEENI 仿真器。ARM SDT（以下关于 ARM SDT 的描述均是以版本 2.50 为对象）可在 Windows95、98、NT 以及 Solaris 2.5/2.6、HP-UX 10 上运行，支持最高到 ARM9(含 ARM9)的所有 ARM 处理器芯片的开发，包括 StrongARM。ARM SDT 包括一套完整的应用软件开发工具：
*armcc ARM 的 C 编译器，具有优化功能，兼容于 ANSI C。
*tcc THUMB 的 C 编译器，同样具有优化功能，兼容于 ANSI C。
*armasm 支持 ARM 和 THUMB 的汇编器。
*armlink ARM 连接器，连接一个和多个目标文件，最终生成 ELF 格式的可执行映像文件。
*armsd ARM 和 THUMB 的符号调试器。
以上工具为命令行开发工具，均被集成在 SDT 的两个 Windows 开发工具 ADW 和 APM 中，用户无需直接使用命令行工具。
*APM Application Project Manager，ARM 工程管理器，完全图形界面，负责管理源文件，完成编辑、编译、链接并最终生成可执行映像文件等功能。

（二）ADS

ADS 是 ARM 公司的集成开发环境软件，他的功能非常强大。他的前身是 SDT，SDT 是 ARM 公司几年前的开发环境软件，目前 SDT 早已经不再升级。ADS 包括了四个模块分别是：SIMULATOR；C 编译器；实时调试器；应用函数库。

ADS 的编译器调试器较 SDT 都有了非常大的改观，ADS1.2 提供完整的 WINDOWS 界面开发环境。C 编译器效率极高，支持 c 以及 c++，使工程师可以很方便的使用 C 语言进行开发。提供软件模拟仿真功能，使没有 Emulators 的学习者也能够熟悉 ARM 的指令系统。配合 FFT-ICE 使用，ADS1.2 提供强大的实时调试跟踪功能，片内运行情况尽在掌握。ADS1.2 需要硬件支持才能发挥强大功能。目前支持的硬件调试器有 Multi-ICE 以及兼容 Multi-ICE 的调试工具如 FFT-ICE。而简易下载电缆不能支持 ADS1.2

（二）REALVIEW DEVELOPER SUITE

RealView Developer Suite 工具是 ARM 公司是推出的新一代 ARM 集成开发工具。支持所有 ARM 系列核，并与众多第三方实时操作系统及工具商合作简化开发流程。开发工具包含以下组件：

- 完全优化的 ISO C/C++编译器
- C++ 标准模板库
- 强大的宏编译器
- 支持代码和数据复杂存储器布局的连接器
- 可选 GUI 调试器
- 基于命令行的符号调试器(armsd)
- 指令集仿真器
- 生成无格式二进制工具、Intel 32 位和 Motorola 32 位 ROM 映像代码的指令集模拟工具
- 库创建工具
- 内容丰富的在线文档

（三）RealView MDK

RealView MDK 开发工具源自德国 Keil 公司，被全球超过 10 万的嵌入式开发工程师验证和使用，是 ARM 公司目前最新推出的针对各种嵌入式处理器的软件开发工具。RealView MDK 集成了业内最领先的技术，包括µVision3 集成开发环境与 RealView 编译器。支持 ARM7、ARM9 和最新的 Cortex-M3 核处理器，自动配置启动代码，集成 Flash 烧写模块，强大的 Simulation 设备模拟，性能分析等功能，与 ARM 之前的工具包 ADS 等相比，RealView 编译器的最新版本可将性能改善超过 20%。RealView MDK 的 RealView 编译器与 ADS 1.2 比较：代码密度：比 ADS 1.2 编译的代码尺寸小 10%；

（四）GNU

GNU 是“GNU’s Not Unix”的递归缩写。Stallman 宣布 GNU 应当发音为 Guh-NOO 以避免与 new 这个单词混淆（注：Gnu 在英文中原意为非洲牛羚，发音与 new 相同）。UNIX 是一种广泛使用的商业操作系统的名称。由于 GNU 将要实现 UNIX 系统的接口标准，因此 GNU 计划可以分别开发不同的操作系统部件。GNU 计划采用了部分当时已经可自由使用的软件，例如 TeX 排版系统和 X Window 视窗系统等。不过 GNU 计划也开发了大批其他的自由软件。1985 年 Richard Stallman 又创立了自由软件基金会（Free Software Foundation）来为 GNU 计划提供技术、法律以及财政支持。尽管 GNU 计划大部分时候是由个人自愿无偿贡献，但 FSF 有时还是会聘请程序员帮助编写。当 GNU 计划开始逐渐获得成功时，一些商业公司开始介入开发和技术支持。当中最著名的就是之后被 Red Hat 兼并的 Cygnus Solutions。到了 1990 年，GNU 计划已经开发出的软件包括了一个功能强大的文字编辑器 Emacs，C 语言编译器 GCC，以及大部分 UNIX 系统的程序库和工具。唯一依然没有完成的重要组件就是操作系统的内核(称为 HURD)。1991 年 Linus Torvalds 编写出了与 UNIX 兼容的 Linux 操作系统内核并在 GPL 条款下发布。Linux 之后在网上广泛流传，许多程序员参与了开发与修改。1992 年 Linux 与其他 GNU 软件结合，完全自由的操作系统正式诞生。该操作系统往往被称为“GNU/Linux”或简称 Linux。（GNU 工具还被广泛地移植到 Windows 和 Mac OS 上。

（五） IAR EWARM

Embedded Workbench for ARM 是 IAR Systems 公司为 ARM 微处理器开发的一个集成开发环境(下面简称 IAR EWARM)。比较其他的 ARM 开发环境, IAR EWARM 具有入门容易、使用方便和代码紧凑等特点。 IAR Systems 公司目前推出的最新版本是 IAR Embedded Workbench for ARM version 4.30。这里提供的是 32k 代码限制、但没有时间限制的 Kickstart 版。 EWARM 中包含一个全软件的模拟程序(simulator)。用户不需要任何硬件支持就可以模拟各种 ARM 内核、外部设备甚至中断的软件运行环境。从中可以了解和评估 IAR EWARM 的功能和使用方法。

IAR EWARM 的主要特点如下:

- 高度优化的 IAR ARM C/C++ Compiler
- IAR ARM Assembler
- 一个通用的 IAR XLINK Linker
- IAR XAR 和 XLIB 建库程序和 IAR DLIB C/C++运行库
- 功能强大的编辑器
- 项目管理器
- 命令行实用程序
- IAR C-SPY 调试器(先进的高级语言调试器)

（六） KEIL ARM-MDKARM

Keil uVision 调试器可以帮助用户准确地调试 ARM 器件的片内外围功能(I2C、 CAN、 UART、SPI、中断、I/O 口、A/D 转换器、D/A 转换器和 PWM 模块等功能)。ULINK USB-JTAG 转换器将 PC 机的 USB 端口与用户的目标硬件相连(通过 JTAG 或 OCD), 使用户可在目标硬件上调试代码。通过使用 Keil uVision IDE/调试器和 ULINK USB-JTAG 转换器, 用户可以很方便地编辑、下载和在实际的目标硬件上测试嵌入的程序。

- 支持 Philips、Samsung、Atmel、Analog Devices、Sharp、ST 等众多厂商 ARM M7 内核的 ARM 微控制器。
- 高效工程管理的 uVision3 集成开发环境
- Project/Target/Group/File 的重叠管理模式, 并可逐级设置;
- 高度智能彩色语法显示;
- 支持编辑状态的断点设置,并在仿真状态下有效。
- 高速 ARM 指令/外设模拟器
- 高效模拟算法缩短大型软件的模拟时间;
- 软件模拟进程中允许建立外部输入信号;
- 独特的工具窗口, 可快速查看寄存器和方便配置外设;
- 支持 C 调试描述语言, 可建立与实际硬件高度吻合的仿真平台;
- 支持简单/条件/逻辑表达式/存储区读写/地址范围等断点。
- 多种流行编译工具选择
- Keil 高效率 C 编译器;
- ARM 公司的 ADS/RealView 编译器;
- GNU GCC 编译器;
- 后续厂商的编译器。

（七） **MULTI 2000**

Multi 2000 是美国 Green Hills 软件公司开发的集成开发环境,支持 C/C++/Embedded C++/Ada 95/Fortran 编程语言的开发和调试,可运行于 Windows 平台和 Unix 平台,并支持各类设备的远程调试。 Multi 2000 支持 Green Hills 公司的各类编译器以及其它遵循 EABI 标准的编译器,同时 Multi 2000 支持众多流行的 16 位、32 位和 64 位处理器和 DSP,如 PowerPC、ARM、MIPS、x86、Sparc、TriCore、SH-DSP 等,并支持多处理器调试。 Multi 2000 包含完成一个软件工程所需要的所有工具,这些工具可以单独使用,也可集成第三方系统工具。

（八） **Embest IDE**

Embest IDE 英文全称是 Embest Integrated Development Environment,是深圳市英蓓特信息技术有限公司推出的一套应用于嵌入式软件开发的新一代集成开发环境。 Embest IDE 是一个高度集成的图形界面操作环境,包含编辑器、编译器、汇编器、链接器、调试器等工具,其界面同 Microsoft Visual Studio 类似。 Embest IDE 支持 ARM、Motorola 等多家不同系列的处理器,对于 ARM 系列处理器,目前支持到 ARM9 系列,包括 ARM7、ARM5 等低系列芯片。

（九） **Hitool for ARM**

由 Hitool International Inc 出品,是一种较新的 ARM 嵌入式应用软件开发系统,主要包括 Hitool ARM Debugger、GNU Compiler (内建)、JTAG cable、评估板以及嵌入式实时操作系统 ThreadX 等。其中编译器模块可以替换成 ARM ADS Compiler 或 ARM SDT Compiler。

其主要特点如下:

- 近似 MS Visual Studio 的调试界面风格,可以在 Win98/ME/NT 等多种 Win32 环境下运行;
- 优秀的工程管理器、源代码和二进制代码编辑器、字符串搜索引擎以及调试目标的自由拖放等功能;
- 支持汇编、C 以及 C++源码级调试,不仅可以通过串口和并口进行本地调试,也可以通过 TCP/IP 进行远端调试;
- 集成了 S-Record、Binary 和 Disassembly 格式的内存上下载工具,Flash 编程工具;
- 支持多种常用的 Jtag Cable,具备通过宏和脚本实现的自动化调试功能。

（十） **Mikro PRO for ARM**

Mikro PRO for ARM 是由 mikriElektronika 公司推出的应用于 ARM Cortex M0, M3 和 M4 嵌入式开发的集成开发环境,根据不同的编程语言分为三种编译器:

MikroC PRO for ARM;

MikroBasic PRO for ARM;

MikroPascal PRO for ARM。

它们具有直观的 IDE,具有先进 SSA 优化功能的强大编译器,大量的硬件和软件库,以及辅助你工作的附加工具如 LCD 定制字符工具,GLCD 位图编辑器,七段编辑器,UART 终端,UDP 终端,HID 终端,ASCII 图表,现场注释编辑器,中断助手,先进的统计功能,等等。

主要特点如下：

mikro PRO for ARM 允许你快速开发并展开复杂的应用：

■使用内置的 Code Editor 写你的 C 语言源代码（代码和参数助手，代码折叠，语法突出，自动纠正，代码模板，等等）。

■使用包含的 50 个 mikro PRO for ARM 库显著加速开发：数据采集，存储，显示，转换，通信等。

■在 Code Explorer 中监控你的程序结构，变量和函数。

■生成兼容所有编程器的注释，可读的汇编和标准的 HEX。

■使用集成的 [mikeoICD（在线调试器）](#) 实时调试工具来监控硬件级上的程序的执行。

■检查程序流并用集成的 [软件模拟器](#) 调试可执行逻辑。

■使用 [Single Static Assignment optimization](#) 缩短你的代码尽可能的小。

■获得详细的报告和图表：RAM 和 ROM 映射，代码统计，汇编列表，调用树等。

■[Active Comments](#) 允许你的注释能互动。

■mikro PRO for ARM 提供大量的例程以扩展，开发并用作你的工程的基石。如果你觉得合适你可以全部复制它们，这也是我们在编译器中包含它们的原因。