【整理】关于Toolchain,cross toolchain,cross compiler

Embedded crifan 6年前 (2012-08-02) 744浏览 0评论

什么是工具链

首先，要解释明白什么是工具链的话，得先简单介绍一下软件开发。

整个软件开发的过程，可以简单解释为，写出软件源码，用编译器将其编译为目标文件，然后用链接器将目标文件以及相关的库函数链接到一起，最终生成可执行文件，即可以在目标系统（CPU）上运行的程序。

当然，程序不可能一次性就完全正确，所以往往还涉及到程序的调试，也就要用到调试器。

很明显，在整个软件开发过程中，除了你自己写的源码之外，所涉及的东西有：

1. 编译器

将源代码文件编译为目标文件:

如果是C，C++等高级语言，那么是通过编译器将源代码编译为目标文件的。

如果是汇编代码，则用的是汇编器将汇编代码编译为目标文件的。

详情可参考：【整理】编译器和汇编器的区别 + 汇编语言和高级语言的区别

2. 库文件

你在写程序时候，就用include所包含的那些标准的C语言的库函数，这些C语言的库函数的全部的集合，就叫做C标准库函数，是个对应的库，普通桌面系统一般为glibc，嵌入式系统一般为uClibc或者其他的，比如eglibc等。

详情可参考：【整理】uclibc,eglibc,glibc之间的区别和联系

3. 链接器

将（源代码编译后所生成的）目标文件，和所依赖的库函数（glibc/uClibc/eglibc/...）等，合并（链接）为可执行程序。

4. 调试器

帮助你调试程序，以便找到程序错误的位置和原因。

工具链，toolchain，指的就是上述软件开发过程中所涉及的一堆的开发工具的组合。

即toolchain包括了：

1. Binutils

上面已经说了，可能需要用到汇编器assembler，需要用到链接器linker等，

关于这两个工具，目前用的最多的是GNU组织的Binutils，即Binary Utilities，二进制工具集，其包含了最核心的两个工具：

A. as，GNU的汇编器

B. ld，GNU的链接器

还包含一些其他的，和二进制处理，分析等相关的工具：

addr2line	Converts addresses into filenames and line numbers
ar	A utility for creating, modifying and extracting from archives
c++filt	Filter to demangle encoded C++ symbols
dlltool	Creates files for building and using DLLs
gold	A new, faster, ELF only linker, still in beta test
gprof	Displays profiling information
nlmconv	Converts object code into an NLM
nm	Lists symbols from object files
objcopy	Copys and translates object files
objdump	Displays information from object files
ranlib	Generates an index to the contents of an archive
readelf	Displays information from any ELF format object file
size	Lists the section sizes of an object or archive file

strings	Lists printable strings from files
strip	Discards symbols
windmc	A Windows compatible message compiler
windres	A compiler for Windows resource files

2. compiler

编译器，用于将高级语言，编译为目标代码。

目前都是用的GNU的编译器GCC，支持众多语言：C, C++, Java, Ada, Fortran, Objective-C等。

3. C library

C库函数，实现了POSIX API，用于用户程序调用，实现相应功能。

目前存在很多C库，常见的有：

glibc，uClibc，eglibc，其三者关系参见：【整理】uclibc,eglibc,glibc之间的区别和联系

其他还有：

BSD libc，Microsoft C Run-time Library，dietlibc，Newlib，klibc，musl等等。

4. Debugger

用于调试程序。最常用的是GDB（GNU Debugger）。

什么是cross toolchain/cross compiler

嵌入式系统开发，也属于软件开发，和上述流程类似，也需要用到上述的toolchain。

只是由于嵌入式系统，相对于普通桌面系统，比较特殊。

特殊在于：

普通桌面系统，为了得到可以运行的程序，直接用对应的编译器编译对应的源码，然后链接生成可执行程序，即可。

此时，桌面系统中，所涉及的：

1. 硬件系统一般为x86的cpu。

2. 软件系统一般为普通的操作系统，包括windows和linux类的，

而所谓的嵌入式系统中：

1.硬件系统不同，硬件资源有限

一硬件一般为特定种类的cpu，比如arm的，mips等。

所以桌面系统中的程序，是包含了x86的指令集的程序，无法在在arm，mips等嵌入式系统中运行的

2.软件系统资源往往也很有限

一般嵌入式系统中，有的没有操作系统，有的有操作系统，也有很多种不同类型的操作系统，比如uCOS，嵌入式Linux等。

而且相关的软件方面的资源也有限制。

总之，想要让程序在嵌入式系统中运行，首先要保证编译出来的程序，可以在相应的硬件cpu下面运行，其次还要充分利用有限的软硬件资源，实现特定的功能。

因此，想要程序在嵌入式系统中运行的话，也是需要上述的toolchain所包含的各种工具，但是有些部分，需要特定针对嵌入式的：

（1）交叉编译器：

用于实现，在普通PC下面，把源码编译成可执行程序，该可执行程序，是包含了目标CPU，比如arm、mips等指令，可以在目标cpu上运行；

（2）嵌入式C运行库：

相对于普通桌面系统所用的glibc，需要采用更加小巧，空间占用少的uClibc或eglibc等运行库。

因此，才有所谓的cross toolchain/cross compiler的出现，目的在于将程序编译出来，可以在嵌入式系统中，满足性能，稳定性，资源占用等等方面的要求，正常运行。

而由于cross toolchain和普通PC系统中的toolchain，其核心部分在于编译器不同，所以cross toolchain/又常常被称为cross compiler，交叉编译器。

如何建立交叉编译器或嵌入式开发环境

由于嵌入式开发的特殊性，比如对于嵌入式Linux来说，整个开发环境，其实包含了除了交叉编译之之外的其他部分吗，比如uboot，rootfs（root filesystem）等内容，而这些内容，也是需要自己建立好，才能真正地继续开始开发具体的功能。

即，整个嵌入式开发环境，核心是交叉编译器，但不仅仅只是编译器。

交叉编译器

下面先单独对于建立交叉编译器的方式，做个简单介绍：

1.直接下载别人已经编译好的交叉编译器

- 从CodeSourcery 下载所需的编译器
- 你买开发板的时候，一般对应开发板的方案提供商会提供你整个交叉编译器：比如TQ2440的天嵌公司，会在bbs上提供交叉编译器的下载的，

2.自己编译交叉编译器

（1）手动自己下载源码自己编译

（2）利用已有工具去手动编译

crosstool

注：

对于编译交叉编译器，其中会涉及很多版本依赖，版本不兼容的问题。

后来有人总结了一些结果，供参考：

Crosstool build results

嵌入式环境搭建系统

其实目前也已经有很多系统，可以帮助我们搭建整个嵌入式开发环境：

（1）Buildroot

buildroot，是个不错的系统，用来建立整套开发环境，包括交叉编译器，uboot，kernel，root fs等。

关于buildroot方面的资料：

buildroot制作编译环境全过程解决

buildroot制作编译环境全过程解决 － 2

buildroot-2009.11编译全过程

Buildroot – 让交叉编译更轻松

（2）ELDK

（3）OpenEmbedded

（4）其他的，比如

- Crosstool-NG
- Crossdev (Gentoo)
- Crossdev/tsrpm (Timesys)

参考资料

1.Toolchains

2. GNU Binutils

3. C standard library

4. 嵌入式Linux下常用的交叉编译方法

继续浏览有关 cross compiler cross toolchain toolchain 交叉编译器 工具链 的文章

与本文相关的文章

【问题解答】AT91SAM9G45使用什么编译链?

【记录】在Cygwin上用Buildroot为xscale建立交叉工具链

【记录】Cygwin下安装crosstool-ng后去为arm920t制作交叉编译器

【记录】Cygwin下用crosstool-ng为arm制作交叉编译器

【整理】arm的交叉工具链（交叉编译器）下载

【整理】交叉编译和现存的交叉编译工具

【记录】在Ubuntu下用crosstool-ng编译xscale的交叉工具链

【记录】Ubuntu下用crosstool-ng为xscale建立交叉编译器arm-xscale-linux-gnueabi-gcc

嵌入式开发之交叉编译器

【记录】在Cygwin下编译gcc-3.4.5-glibc-2.3.6的arm-xscale-linux-gnu交叉编译器

【整理】天嵌的开发板TQ2440相关资料

【整理】嵌入式领域中常见的名词和概念的解释

发表我的评论

写点什么...

表情

提交评论