## 英创主板 JAVA 应用开发简介

#### 英创公司

#### 2017年9月

随着互联网技术的快速发展, Java 语言以其简单可靠,安全性高,可移植性强等特点 得到极大普及,长期占据编程语言排行榜第一的位置。开发者使用 Java 语言开发应用程序, 可以达到"一次编译,到处运行"的目的。

所谓"一次编译"是指对于开发者而言,只需要使用 Java 编译工具对 Java 源文件(.java 文件)进行一次编译,生成字节码文件(.class 文件);而"到处运行"是指字节码.class 文件可以运行在所有装有 Java 运行环境(JRE - Java Runtime Environment)的设备上运行。只要设备上装有对应操作系统的 JRE,就能运行所有.class 文件,无需再次编译。



#### 图 1. Java 语言编译执行示意图

为了实现对 Java 应用程序的支持,英创在 Linux 的文件系统中移植了针对嵌入式 Linux 的 JRE。JRE 的运行需要占用的内存超过 100MB,英创的 ESM335x、ESM6800 和 ESM6802 三种产品线的主板可满足 Java 虚拟对硬件环境的要求。另一方面,ESM335x 和 ESM6800 受 NandFlash 容量限制,预装了 JRE 的 ESM335x 和 ESM6800 主板,将不再支持 QT。

鉴于英创主板主要应用于工业控制和通信管理领域,所以英创提供的 Java 硬件支持包 主要包括对以太网口、串口、CAN、GPIO、SPI、I2C 等工控接口的支持。目前暂时还不支 持对显示接口(/dev/fb0)的访问。英创将提供访问硬件所需的名为 libemtronixhardware.so 的动态库和相关接口 Java 类源文件给客户。libemtronixhardware.so 完全符合 JNI 规范, 用户加载之后,即可通过英创提供的 java 类,访问硬件接口。为了让用户能够快速熟悉英 创主板上的 Java 程序开发,英创提供了完整的开发环境和应用示例,用户可以按如下步骤 配置学习 Java 应用程序开发。在此过程中如果遇到问题,请先看 7、常见问题 Q&A。如果 不能解决问题,请通过电话或者邮件联系我们。

### 1、在 PC 上安装 Java 开发工具包(JDK)

要进行 Java 应用程序的开发工作,必须要在 PC 机上安装 Java 软件开发工具包 JDK。 用 户 可 以 直 接 通 过 以 下 链 接 到 oracle 官 网 下 载 对 应 操 作 系 统 的 JDK。 http://www.oracle.com/technetwork/java/javase/downloads/index.html



Java SE Development Kit 8u144 You must accept the Oracle Binary Code License Agreement for Java SE to download this software.							
Accept Licen	se Agreement	Decline License Agreement					
Product / File Description	File Size	Download					
Linux ARM 32 Hard Float ABI	77.89 MB	jdk-8u144-linux-arm32-vfp-hflt.tar.gz					
Linux ARM 64 Hard Float ABI	74.83 MB	jdk-8u144-linux-arm64-vfp-hflt.tar.gz					
Linux x86	164.65 MB	jdk-8u144-linux-i586.rpm					
Linux x86	179.44 MB	jdk-8u144-linux-i586.tar.gz					
Linux x64	162.1 MB	jdk-8u144-linux-x64.rpm					
Linux x64	176.92 MB	jdk-8u144-linux-x64.tar.gz					
Mac OS X	226.6 MB	jdk-8u144-macosx-x64.dmg					
Solaris SPARC 64-bit	139.87 MB	jdk-8u144-solaris-sparcv9.tar.Z					
Solaris SPARC 64-bit	99.18 MB	jdk-8u144-solaris-sparcv9.tar.gz					
Solaris x64	140.51 MB	jdk-8u144-solaris-x64.tar.Z					
Solaris x64	96.99 MB	jdk-8u144-solaris-x64.tar.gz					
Windows x86	190.94 MB	jdk-8u144-windows-i586.exe					
Windows x64	197.78 MB	idk-8u144-windows-x64.exe					

下载后请按提示完成安装。

# 2、安装 eclipse 集成开发环境

英创提供了 eclipse-java-oxygen -SR1-win32-x86\_64.zip,用户只需要解压到磁盘即可。

## 3、导入英创示例程序到 eclipse

英创公司为用户快速熟悉英创主板的 Java 应用程序开发提供了所有接口的示例程序。 用户只需要将示例程序导入到 eclipse 中即可进行学习。

打开 eclipse 软件后,首先,点击左上角 "File" --> "Import..."。

🛢 Ja	ava - Eclipse					
File	) Edit Source	e Refactor	Navigate	Search	Project	Run V
	New Open File				Alt+Sh	ift+N ▶
	Close Close All				C Ctrl+Sh	trl+W ift+W
	Save Save As				(	Ctrl+S
¢	Save All Revert				Ctrl+Sl	hift+S
	Move					
Ľ	Rename					F2
\$	Refresh Convert Line I	Delimiters To				F5
Ð	Print				C	Ctrl+P
	Switch Works Restart	pace				۲
2	Import					
4	Export					
	Properties				Alt+	Enter

Import	
Select Create new projects from an archive file or directory.	Ľ
<u>S</u> elect an import source:	
type filter text	
<ul> <li>▲ General</li> <li>④ Archive File</li> <li>☆ Existing Projects into Workspace</li> <li>④ File System</li> <li>■ Preferences</li> <li>&gt; ▷ CVS</li> <li>&gt; ⊘ Git</li> <li>&gt; ⊘ Install</li> <li>&gt; ⊘ Maven</li> <li>&gt; ⊘ Run/Debug</li> <li>&gt; Tasks</li> <li>&gt; ∑ Team</li> </ul>	
(?) < <u>B</u> ack <u>Next &gt;</u> <u>Finish</u>	Cancel

在弹出的窗口中选择 "General" >> "Existing Projects into Workspace",点击 "Next"。

Import		
Import Project Select a direct	ts tory to search for existing Eclipse projects.	
Select roo <u>t</u> ○ Select <u>a</u> rchi	directory: we file:	]
<u>P</u> rojects:		1
3	刘览文件夹	
	Select root directory of the projects to import	]
	<ul> <li>▷ interpretende interpretende</li></ul>	
	Imetadata	
Options Searc <u>h</u>	▷ bin ▷ src	
Working s	ESM6802_Android	
🗆 Add pr	⊳ <u>iii</u> e eta728	
W <u>o</u> rking	ftproot 2	
	文件夹 (F): testEMHardwarelib	
	新建文件夹 (M) 确定 取消	
?	< <u>B</u> ack <u>N</u> ext > <u>F</u> inish Cancel	]

在弹出的窗口中点击"Browse",选择英创示例程序 testEMHardwarelib 所在目录,点击确定。最后点击"finish"即可。

# 4、在 eclipse 中编辑编译 Java 示例程序

成功到导入 testEMHardwarelib 示例程序后, 在 eclipse 的 Package Explorer 可以看到

项目的目录结构。其中 src 目录下是项目的源代码, JRE System Library 是安装 JDK 时安装的 Java 类库。 src 目录下分了两个源码包: com.emtronix.main 是测试程序, emtronix.hardware 是英创针对英创主板的硬件接口封装的功能接口类及相关数据结构。

a 📂 testEMHardwarelib

🔺 / 🖽 src

- 🔺 🌐 com.emtronix.main
  - I2CKeypad.java
  - I testCANMain.java
  - ItestGPIOMain.java
  - I testI2CMain.java
  - I testIRQMain.java
  - I testRTCMain.java
  - I testSerialMain.java
  - I testSPIMain.java
- 🔺 🌐 emtronix.hardware
  - D CAN.java
  - D CanFilter.java
  - D CanFrame.java
  - I GPIO.java
  - 👂 🚺 I2C.java
  - I2CParam.java
  - IRQ.java
  - I RTC.java
  - I Serial.java
  - I SPI.java
  - I ToolsFunc.java
- ▲ Marcel JRE System Library [jre1.8.0\_111]
  - resources.jar C:\Program Files\Java\jre1.8.0\_111\lib
  - rt.jar C:\Program Files\Java\jre1.8.0\_111\lib
  - isse.jar C:\Program Files\Java\jre1.8.0\_111\lib

为了让 Java 应用程序能够访问英创主板的硬件接口,英创在文件系统中提供了动态链接库 libemtronixhardware.so,Java 应用程序只要加载这个动态库,就能够使用其中的类,调用类的公共方法去访问对应硬件接口。如下图所示,用户可以直接双击 com.emtronix.main 包中的 testCANMain.java 源码,进行编辑,在源文件中加载 emtronixhardware 库后,调用 emtronix.hardware 包中的 CAN 功能类,即可使用 CAN 类的 Open、SendFrame、RecvFrame、Close 等接口完成 CAN 通信的应用开发。

testCANMain.java %

import emtronix.hardware.CAN;
public class testCANMain {
 static {
 System.loadLibrary("emtronixhardware");
 }
 public static boolean exitflag;
 public static CAN can0;

完成编辑后,需要编译生成后缀为.class 的字节码文件。eclipse 默认是在源文件保存 后自动编译,在项目所在目录的 testEMHardwarelib/bin 目录下生成与 src 下源码包对应的 目录结构和 class 文件。

如果用户不需要自动编译,可以在 Project 页面选择 "Build Automatically" 将前面的钩 去掉。这是之前灰化的 "Build Project" 会变为激活状态,每次保存源文件后,使用 Build Project 就可以编译项目。



### 5、在英创主板上运行 testEMHardwarelib 示例程序

将 testEMHardwarelib/bin 目录下的所有目录全部拷贝到英创主板 esm335x 的 /mnt/nandflash 目录下。进入 /mnt/nandflash 目录,使用 java com.emtronix.main.testCANMain 命令运行 CAN 测试程序。其中,java 命令是 jre 提供的运行 java 程序的指令,com.emtronix.main.testCANMain 是测试类的完整类名,包括包名+ 类名,java 指令需要跟完整的类名作为参数。CAN 测试程序运行如下图所示,同理,要测试其他接口只需要将完整类名换成对应接口即可。

[root@EM335X /]# [root@EM335X /]#cd /mnt/nandflash/ [root@EM335X /mnt/nandflash]#ls emtronix userinfo.txt COII [root@EM335X /mnt/nandflash]#java com.emtronix.main.testCANMain ===CAN test== start testCANMain class args.length:0 canparams:can0:100000 can0 baudrate 1000000 [17513.814004] c\_can\_platform 481d0000.can can0: setting BTR=1c0e BRPE=0000 SOCK\_RAW can sockfd:4 send frame:0 can\_id:00000001 Std. can\_dlc:4 data(HEX): 6A 61 76 61 send frame:1 can\_id:00000064 Std. can\_dlc:5 data(HEX): 68 65 6C 6C 6F send frame:2 can\_id:00000001 Std. can\_dlc:4 data(HEX): 6A 61 76 61

#### 6、使用英创提供的功能类源码包 emtronix.hardware 开发自己的应用程序

使用英创提供的功能类可以快速方便的开发自己的应用程序。首先,用户要新建一个 java 项目, "File" >> "New" >> "Java project", 弹出"New Java Project"窗口,可以 参考下图设置。点击 Finish。

New Java Project	
<b>Create a Java Project</b> Create a Java project in the workspace or in a	n external location.
Project name: testMyApp	
✓ Use <u>d</u> efault location	
Location: D:\01.public\eclipseJava\testMyAp	Browse
JRE	
O Use an execution environment JRE:	CDC-1.1/Foundation-1.1
© U <u>s</u> e a project specific JRE:	jre1.8.0_111
Ose default JRE (currently 'jre1.8.0_111')	Configure JREs
Project layout	
Use project folder as root for sources ar	nd class files
Oreate separate folders for sources and	class files <u>Configure default</u>
Working sets	
Add projec <u>t</u> to working sets	
W <u>o</u> rking sets:	▼ S <u>e</u> lect
(?) < <u>B</u> ack <u>N</u> ex	t > Finish Cancel

然后,右键选中 testEMHardwarelib 项目 src 目录下的 emtronix.hardware 功能类包,选择 Copy。

a 📄 testEMHardwarelib

4 🇀 src			
tom.emtronix.n	nain		
a 🖶 emtronix.hardv	are	1	
D CAN.java		New	•
D CanFilter.jav		Go Into	
D CanFrame.ja		Open in New Window	
I GPIO.java		Open Tupe Hierorchy	54
I2C.java		Open Type Hierarchy	F4
⊳ 🚺 I2CParam.ja		Show In	Alt+Shitt+W ►
IRQ.java	Þ	Сору	Ctrl+C
D RTC.java	Þ	Copy Qualified Name	
D Serial.java	ß	Paste	Ctrl+V
D SPI.java	×	Delete	Delete
I ToolsFunc.ja			
👂 🛋 JRE System Librar	Ð	Remove from Context	Ctrl+Alt+Shift+Down
⊿ 崖 testMyApp		Build Path	+
进 src		Source	Alt+Shift+S ►
👂 🛋 JRE System Librar		Refactor	Alt+Shift+T ►
	2	Import	
	4	Export	

然后选中 testMyApp 项目的 src 目录, 右键选择 Paste 或者直接 "Ctrl+v"。如果粘贴失败, 请重新复制, 再次粘贴, 多试几次。

4	📂 test	MyAp	p	
	🕮 s	sre		
	> 🛋 J		New	+
			Open in New Window	
			Open Type Hierarchy	F4
			Show In	Alt+Shift+W ►
		D	Сору	Ctrl+C
		Þ	Copy Qualified Name	
		Ē	Paste	Ctrl+V
		×	Delete	Delete

接着,在 src 目录下新建一个源码包 mypackage。

▲ 📄 testMyApp	5			L	fra
31	New	•	1	Java Project	
Þ	Go Into		C2	Project	
⊳	Open in New Window		₿	Package	
	Open Type Hierarchy	F4	¢	Class	•

😂 New Java Pao	kage		
Java Package Create a new J	ava package.		
Creates folders	corresponding to packages.		
Source fol <u>d</u> er:	testMyApp/src		Br <u>o</u> wse
Na <u>m</u> e:	mypackage		]
<u>C</u> reate pack	age-info.java		
?		<u> </u>	Cancel

然后在 mypackage 中新建一个 MyApp 类,并勾选 "public static void main"。

a 岸 testEMHardv	vareli	b			package emtronix.ha
🔺 进 src					
▷ 🌐 com.e ⊿ 🌐 emtro		New	•	鬯	Java Project
⊳ J CA ⊳ J Car ⊳ J Car		Open in New Window Open Type Hierarchy Show In	F4 Alt+Shift+W ▶	••••••••••••••••••••••••••••••••••••••	Project Package Class
▷ ]] GP ▷ ]] 12C ▷ ]] 12C ▷ ]] 1RC ▷ ]] 1RC		Copy Copy Qualified Name Paste Delete	Ctrl+C Ctrl+V Delete	6 6 6 6 6 6 7 7 7 7 7 7 7 7 7 7 7 7 7 7	Interface Enum Annotation Source Folder
<ul> <li>▷ D Set</li> <li>▷ D SPI</li> <li>▷ D Too</li> <li>▷ M JRE Syste</li> <li>▲ W testMyApp</li> </ul>	<u>\$</u>	Remove from Context Build Path Source Refactor	Ctrl+Alt+Shift+Down Alt+Shift+S Alt+Shift+T		Java Working Set Folder File Untitled Text File JUnit Test Case
▲ (# src ▲ ⊕ emtro ▷ ID CA	ntro 🕹	Import Export			Task Example
⊳ D Car ⊳ D Car ⊳ D Car	Ŷ	Refresh Assign Working Sets	F5		Other Ctrl+N System.out.
<ul> <li>♪ ] GP</li> <li>♪ ] 12C</li> <li>♪ ] 12C</li> <li>♪ ] 18C</li> <li>♪ ] RT</li> <li>♪ RT</li> <li>♪ SPI</li> <li>♪ To</li> </ul>		Debug As Run As Validate Team Compare With Restore from Local History	> > >		} }
		Properties	Alt+Enter		
⊳ <b>⊒∖</b> JRE System	m Lik	orary [jre1.8.0_111]			

New Java Class		
<b>Java Class</b> Create a new Java	class.	C
Source fol <u>d</u> er:	testMyApp/src	Br <u>o</u> wse
Pac <u>k</u> age:	mypackage	Bro <u>w</u> se
Enclosing type:		Bro <u>w</u> se
Na <u>m</u> e: Modifiers:	MyApp public	]
<u>S</u> uperclass:	java.lang.Object	Brows <u>e</u>
Interfaces:		<u>A</u> dd
Which method stub	os would you like to create? ▼ public static <u>v</u> oid main(String[] args) ■ <u>C</u> onstructors from superclass ▼ Inherited abstract methods	-
Do you want to ad	d comments? (Configure templates and default value <u>her</u> <u>G</u> enerate comments	<u>(e)</u>
?	Finish	Cancel

在类中静态加载 emtronixhardware 动态库,这时已经配置好英创主板的 Java 应用程序开发环境,用户可以按照自己的业务逻辑,遵照 Java 语法,开发 Java 应用程序。



#### 7、常见问题 Q&A

Q1: 导入项目后中文注释乱码

A1:有的项目在 linux 系统下使用的 utf-8 编码,有的项目在 windws 中使用的是 GBK 编码,当导入的项目编码与操作系统编码不同时会出现中文乱码。这时可以在项目属性的 resource 页面选择匹配的编码格式即可。

File Edit Source Refactor Navigate Search	Project Run Window Help	♥ ୬ ♥ 目 日   2 ▼ 원 ▼ ♥ ♥ ▼ → ▼   2	
<ul> <li>Package Explorer 32</li> <li>b testEMHardwarelib</li> <li>c testEMHardwarelib</li> <li>c server and the server and th</li></ul>	Properties for testEMHardw Properties for testEMHardw Resource Builders Java Build Path Java Compiler Java Compiler Java Compiler Java Concation Project References Run/Debug Settings Task Repository Task Tags Validation WikiTest	ItettSerialM	e Defaults Apply OK Cancel
		·····································	

## Q2: 复制粘贴 emtronix.hardware 包到新项目后,编译出错

Syntax error, 'for each' statements are only available if source level is 1.5 or greater;

The method format(String, Object[]) in the type String is not applicable for the arguments (String, int)

⊿ 💕 testMyApp	i			
🔺 🜐 src	1			
4 🌐 emtronix.hardware	Problems X @ Javadoc @ Declaration = Progress			
CAN.java	5 errors () warnings () others			
CanFilter.java	Description			
D CanFrame.java	Description			
I GPIO.java	4 V Errors (5 items)			
I2C.java	😡 Syntax error, 'for each' statements are only available if source level is 1.5 or greater			
I2CParam.java	🙀 Syntax error, 'for each' statements are only available if source level is 1.5 or greater			
IRO.java	😡 Syntax error, 'for each' statements are only available if source level is 1.5 or greater			
RTC.iava	🕼 The method format(String, Object[]) in the type String is not applicable for the arguments (String, int)			
Serial.java	🕼 The method format(String, Object[]) in the type String is not applicable for the arguments (String, int)			
SPLjava				
ToolsFunc.java				

A2: 以上错误是因为 java compiler 设置的兼容版本太低,而 emtronix.hardware 中用 到了部分接口只在 1.5 以上才有。右键项目>>properties>>Java Compiler 可以看到 eclipse 默认的设置是 1.4。

勾选"Enable project specific settings",选择"Compiler compliance level"为1.5或者更高,点击OK,重新编译项目即可。

Properties for testMyApp					
type filter text	Java Compiler	⇔ ◄ ⇔ ◄			
⊳ Resource	Enable project specific settings	Configure Workspace Settings			
Java Build Path	JDK Compliance				
▷ Java Code Style	Use compliance from execution environment on the 'Java Build Path'				
⊿ Java Compiler	Compiler compliance level:	1.4 -			
Annotation Processin Building	✓ Use default compliance settings				
Errors/Warnings	Generated .class files compatibility:	1.2 -			
Javadoc Task Tase	Source compatibility:	1.3 *			
Java Editor	Disallow identifiers called 'assert':	Warning 👻			
Javadoc Location	Disallow identifiers called 'enum':	Warning 👻			
Refactoring History	Classfile Generation				
Run/Debug Settings	✓ Add varia <u>b</u> le attributes to generated class files (used by the debugger)				
Task Repository	Add line number attributes to generated class files (used by the debugger)				
Task Tags	$\checkmark$ Add source <u>fi</u> le name to generated class file (used by the debugger)				
▷ Validation	✓ Pr <u>e</u> serve unused (never read) local variables				
WikiText	☐ Inline finally blo <u>c</u> ks (larger class files, but improved performance)				
<		Restore <u>D</u> efaults Apply			
?		OK Cancel			

Q3: 在英创主板上运行 class 文件, Could not find or load main class

A3:可能是 java 命名跟的参数并不是完整的类名,完整的类名包括了包名和类名,要 在包的上级目录运行 java 命令,跟上完整的类名。例如:

[root@EM335X	/mnt/nandflash	]#ls		
COM	emtronix	userinf	fo. txt	
[root@EM335X	/mnt/nandflash	]#java c	com.emtronix.ma	ain.testCANMain
=====CAN +e	•<†=========			

Q4: build project 在 bin 目录中不再生成 class 文件

A4: Close project, 然后右键项目选择 delete, 不勾选 "delete project contents on disk"。最后重新导入项目, 重新编译。

Delete Resources					
Are you sure you want to remove project 'testEMHardwarelib' from the workspace?					
Delete project contents on disk (cannot be undone)					
Project location:					
D:\01.public\eclipseJava\testEMHardwarelib					
Previe <u>w</u> > OK	Cancel				