Design of a Visual Development Model for Embedded System

LV Wei-gong[1], HOU Xiang-mei[2], YANG Shu-hua[3]

School of Computer Science and Technology, Harbin Institute of Technology at Weihai, Weihai, China

Email: 1. lwgswl@sina.com, 2. zhendema_110@yahoo.com.cn

Abstract: According to the features of Embedded Systems, such as simplify, individuation, resource limited, an embedded visual development model based on hardware device is put forward, and a set of development tools and widgets are implemented. The development model, including mappings of three levels from the hardware operation basic unit to the hardware application basic unit, and to the visual application unit, fully embodies the support for hardware, and also represents the essential characteristics of embedded systems. The model, using the existing embedded GUI, compatible with the current development mode, has no hard rules for the operating system. The model is well adapted, and the use of the model is flexible. It is able to reduce the difficulty and workload of embedded visual development.

Keywords: Visual Development; Embedded System; Model; Mapping

一种嵌入式系统可视化开发模型的设计

吕为工[1]，侯向美[2]，杨书华[3]

哈尔滨工业大学（威海）计算机科学与技术学院，威海，中国，264209

Email: 1. lwgswl@sina.com, 2. zhendema_110@yahoo.com.cn

【摘要】本文根据嵌入式系统精简、个性化、资源有限等特点，提出了一种基于设备的嵌入式可视化开发模型，并依据该模型实现了一组开发工具和控件集。开发模型包括从硬件操作基本单元到硬件应用基本单元，再到可视应用单元的三个层次的映射，充分体现了对硬件的支持，表征了嵌入式系统的本质特征。该模型利用了现有嵌入式 GUI，兼容目前的开发方式，对操作系统无要求，使用灵活，适应性强，能够降低嵌入式可视化开发的难度和工作量。

【关键词】可视化开发；嵌入式系统；模型；映射

1 引言

随着嵌入式系统的发展，其硬件性能不断提高，对拥有可视化界面的需求也不断增长，除了数码相机、PDA、手机等传统的嵌入式可视化产品外，越来越多的领域，如工业设备、交通电子等也开始使用可视化的人机接口。庞大的可视化产品需求，对嵌入式可视化开发技术提出了更高的要求。

嵌入式可视化开发不同于基于标准软硬件平台的可视化开发一般模式。在一般模式下，程序员只要熟悉相应的可视化开发环境就可以了，而嵌入式可视化开发[1]要复杂得多，主要体现在以下三个方面：1）嵌入式系统具有个性化的特点，程序员必须进行个性化设备驱动程序的开发，个性化设备驱动程序的开发不仅必须的，而且是最关键最重要的部分。2）嵌入式系统的可视化应用中，通常使用嵌入式 GUI[2]软件(如 Qt/E、MiniGUI 等)来完成可视化系统的开发，这些 GUI 仅支持基本的输入输出设备，对其它硬件设备则没有提供支持。程序员必须自己建立与之对应的可视化元素并实现互动关系。3）对于嵌入式操作系统和嵌入式 GUI 来说，通常也要经过移植，才能正确运行在嵌入式环境中。

从上面的论述可以看到，嵌入式可视化开发的难度和工作量主要由三个方面决定，除了第三方面属于嵌入式操作系统和嵌入式 GUI 设计需要解决的问题范畴，前

基金项目：山东省科技攻关(2005GG3201134). Foundation Item: Project supported by the S&T Program of Shandong Province, China (2005GG3201134).

978-1-935068-23-5 © 2010 SciRes.

两个方面都是由于当前的可视化开发模式不能适应嵌入式系统[3]的特殊性引起的，只有建立与嵌入式相适应的可视化开发模式并提供相应的开发工具，才能从根本上降低嵌入式可视化开发的难度，减少工作量，提高嵌入式可视化开发的效率。

为了有效地表征嵌入式系统的本质特征，我们提出一种嵌入式系统可视化开发模型，该模型底层直接基于硬件设备，上层基于 GUI 可视化控件系统，能有效地支持嵌入式系统的个性化特点，对嵌入式系统可视化开发提供全面地支持。为了验证本模型，本文在 ARM[4]S3C2410 和嵌入式 linux 平台上实现了一个设备驱动程序[5]的框架生成工具、一个属性服务器、一个属性生成工具以及一组 Qt/E[6]可视化控件，并利用它们实现了一个具有可视化应用接口的地毯提花机系统。

2 可视化模型的整体设计

本文提出的一种嵌入式系统可视化开发模型从底层硬件设备开始，把可视化开发自底向上地抽象成三个映射：硬件基本操作单元到硬件基本应用单元的映射、硬件基本应用单元到可视应用属性的映射以及可视应用属性到可视应用单元的映射，如图 1 所示。

2.1 硬件基本操作单元层

对于任意一个嵌入式设备 D，本文假定：

$D = \{D_1, D_2, \ldots, D_n\}$

其中 $D_1, D_2, \ldots, D_n$ 为有独立软硬件开发环境的子设备，这些子设备是独立的嵌入式系统开发实体，子设备的各组成部分不再具有独立软硬件开发环境，对这样的子设备 D_i，可以描述为：

$$D_i = \{D_{i1}, D_{i2}, \ldots, D_{im}\}$$

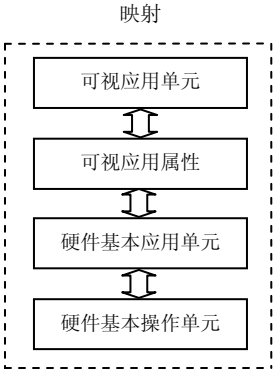

Figure 1. a visual development model for embedded system

图 1. 一种嵌入式系统可视化开发模型

其中 $D_{i1}, D_{i2}, \ldots, D_{im}$ 为硬件基本操作单元，是从操作角度分解出的小粒度的硬件单元，具有如下特性：

1）可操作性。只需要描述子设备中的可操作硬件，对于子设备中的不可操作部分，如某嵌入式系统中的不可控电源，从可视化开发角度是透明的，所以不需要进行描述。

2）独立性。操作独立的硬件应作为独立的整体被描述，因为进一步的细化对操作的执行不会产生影响，所以这种细化毫无意义。

3）实时要求。对不能打断的强实时要求的操作为保证操作的有效，必须作为独立的单元对待。

4）简单性。在满足上述要求的条件下，尽量保证硬件基本操作单元的粒度最小化，复杂的操作逻辑应该在更上层实现。

这样，一个嵌入式设备被简单描述为多个子设备的集合，而子设备被描述为多个硬件基本操作单元的集合。

2.2 硬件基本应用单元

在 1.1 节，我们对嵌入式系统中的硬件按着其操作进行了分解，我们关心的是操作本身，而不是这个操作在系统中的使用，例如，一个输出用开关量，在硬件操作单元层被关注的是开关状态如何被控制，而不是这种转换的外部含义。在硬件基本应用单元层则不同，如果前面的输出用开关量影响的是某个阀门的开关状态，那么应用单元就应该是这个阀门的说明和其状态描述。

本文把一个子设备 D_i 的硬件基本应用单元层看作一个集合 A_i：

$A_i = \{A_{i1}, A_{i2}, \ldots, A_{it}\}$

其中 $A_{i1}, A_{i2}, \ldots, A_{it}$ 为硬件基本应用单元，对于任意一个 A_i，抽象为一个四元组为：

$A_i = <$ name, sort, number, type$>$

其中，name 为硬件基本应用单元的名字；sort 为种类，如 bit(位类型，如开关量就应定义为位)，U8，U16 等；number 为数量，当应用需要以一组数据的形式进行描述时使用，一般情况为 1；type 为应用类型，包括输入和输出两种。

硬件基本操作单元层和硬件基本应用单元之间的映射关系，实际上是 D_i 到 A_i 之间的映射，如图 2 所示。

对于任意的 $D_i = \{D_{i1}, D_{i2}, \ldots, D_{im}\}$ 和 $A_i = \{A_{i1},$

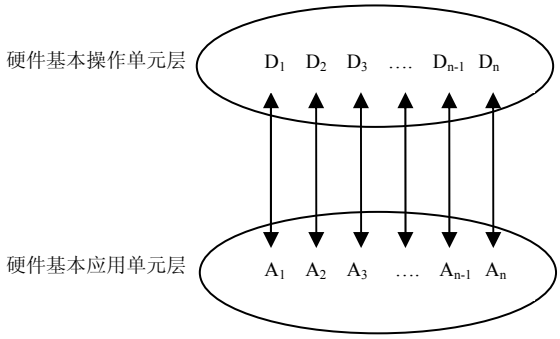

Figure 2. mapping between levels of basic hardware operation unit and basic application unit

图 2. 硬件基本操作单元层和硬件基本应用单元层之间的映射

A_{i2}, …, A_{it}}，A_{ij}(j=1, ..., t)可能与一个或多个 D_{ik}(k=1, ..., m)关联，即某个基于硬件的应用可能涉及一个或多个硬件的操作，在 A_{ij} 到 D_{ik} 的映射中，在描述出这种操作逻辑，有时这种操作逻辑很简单，甚至可以直接映射，有时则很复杂，甚至需要引入中断或某些底层协议。

硬件基本应用单元层，处于应用的底层，应具有以下特点：1）对于上层应用来说，它屏蔽了硬件使用的细节。2）提供最基本的硬件应用方式，是最小粒度的应用单元，充分保证上层应用的灵活性。3）实时性能。最强实时要求已经在硬件操作单元层完成，本层要保证次一级的硬件实时性能，以使更高一级应用不必再考虑硬件使用的实时问题。

2.3 可视化控件层

在本文的可视化开发模型中，可视化控件完成了可视应用属性到可视应用单元的映射，二者之间的逻辑关系体现在控件内部代码中，对于用户来说，可视化控件相当于一个可视应用单元，对于可视化应用程序员，则使用可视应用属性及其内部的方法操作控件，用户使用控件时，可视界面的变化是其内部逻辑关系和程序员控件操作的外在表现。

可视化控件是支持可视化编程的基础，无论是大型的可视化开发环境，还是嵌入式 GUI，都把可视化控件作为可视化编程的基本组成部分。通过可视化控件，可视化编程转变为对可视化控件的操作，而控件本身又是可重用的，使用已有可视化控件和生成专用可视化控件，能够适应可视化开发的需求。

假定 C 为能够使用的可视化控件集合：

$C=\{C_1,C_2,...,C_k\}$

其中 $C_1,C_2,...,C_k$ 为可视化控件，对于任意一个控件 C_i，其包含的可视应用属性为 V_i：

$V_i=\{V_{i1},V_{i2},...,V_{is}\}$

其中 V_{ij}(j=1,...,s)为控件 V_i 可视应用属性。

要使可视化应用和底层的硬件设备关联，还需要建立硬件基本应用单元和可视应用属性之间的映射，这个映射体现了普通应用到可视化应用的过渡。

任意一个 V_{ij} 可以与同一个或多个硬件基本应用单元 D_{pq} 相关联，但不是所有的 V_{ij} 都需要做这种关联，例如，一个温度计形式的可视化控件，包含一个能影响温度高低显示的可视应用属性——温度值，只要把它同代表温度传感器的硬件基本应用单元关联起来，就能实现实际温度显示功能，而控件中除了温度值以外的属性不必与底层关联。

同理，也不是所有的硬件基本应用单元都必须与可视应用属性关联，一些硬件的基本应用完全可能不需要可视化，此时这种关联是没有必要的，程序员完全可以直接使用硬件基本应用单元。

3 可视化开发模型的实现

经过前面的分析，以下提供各层次的实现方案，如图 3 所示。

3.1 驱动程序框架生成

这部分主要功能是实现硬件基本操作单元和硬件基本应用单元的映射，这个映射是由驱动程序完成的，驱动程序框架生成工具为驱动程序的生成提供支持。

硬件基本操作单元可以支持三类基本的硬件，第一类为 CPU 的 GPIO 引脚，第二类是简单硬件，一般指针利用非可编程接口芯片操作的硬件，第三类是复

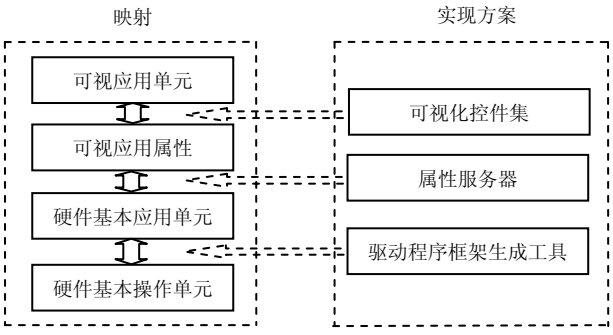

Figure 3. Realization scheme for the visual development model

图 3. 可视化开发模型的实现方案

978-1-935068-23-5 © 2010 SciRes.

杂硬件，一般指通过可编程接口芯片操作的硬件。前两类硬件，由于功能确定，驱动程序可以由驱动程序框架生成工具自动生成，对第三类硬件，驱动程序框架生成工具提供一个开发接口，用户可在基本框架下自行添加处理代码。

在应用单元这个层面，利用驱动程序框架生成工具，用户可以定义硬件基本应用单元，可视化地与已建立的硬件基本操作单元建立关联，简单的映射可自动生成，复杂地映射则提供开发接口。硬件基本应用单元通过驱动程序调用接口与上层发生关联。

驱动程序框架生成工具主要功能包括：硬件基本应用单元设定、硬件基本操作单元设定及开发接口、映射选择及开发接口、中断服务程序选择及开发接口、自定义程序开发接口(头文件及子程序)、系统资源和设备号选择、设备初始化及释放代码开发接口及编译过程显示等。

3.2 可视化控件集

可视化控件集主要包括三类控件：1）操作系统提供驱动程序的设备，例如 USB 等。2）简单设备，是指可以由单一属性决定控件关键外观的设备。完成了包括开关量、模拟量、频率量等类型的代表控件。3）复杂设备，是指那些由多个设备组合而成，控件外观的改变需要多个组合控件属性的共同作用的设备，像提花机、水位控制器等。后两类设备都属于个性化设备，要通过 2.1 节描述的驱动程序框架生成工具生成驱动，第一类设备则使用操作系统提供的设备驱动程序。

可视化控件的生成完成了可视化应用单元和可视化应用属性的映射，为了方便属性服务器使用可视化应用属性，在生成可视化控件的同时，将可视化控件的属性记录进入一个属性的初始化文件。

3.3 属性服务器

在这部分，实现了一个属性服务器生成工具，能建立硬件基本应用单元和可视化应用属性的映射，而由其生成的属性服务器将实际完成这种映射。硬件基本应用单元是由下层驱动程序层提供的有严格命名规范的接口函数，可视化应用属性则来源于上层可视化控件提供的应用属性。硬件应用单元和可视化应用属性之间的映射关系也存在多种可能。生成工具提供简单映射的自动建立，以及复杂映射的开发接口。

通过属性服务器建立的基本应用单元与可视化应用属性之间的映射关系，从实际上将底层驱动程序和上层可视化控件有效的联系在一起，通过可视化控件可以直观的显示硬件的变化，硬件的改变也可以影响可视化控件的属性值。映射关系建立后，形成记录文件。

属性服务器使用驱动程序调用接口传递硬件基本应用单元信息，通过共享存储器与可视化控件传递可视应用属性信息，如图 4 所示。

4 结论

本文构建了一种基于硬件的嵌入式可视化开发模型，并在 ARM S3C2410 和 Linux 环境下实现了基于该模型的驱动程序框架生成工具、属性服务器生成工具和一个 Qt / E 控件集，使用这些工具及控件，验证了该模型的可行性，利用该模型可以降低嵌入式系

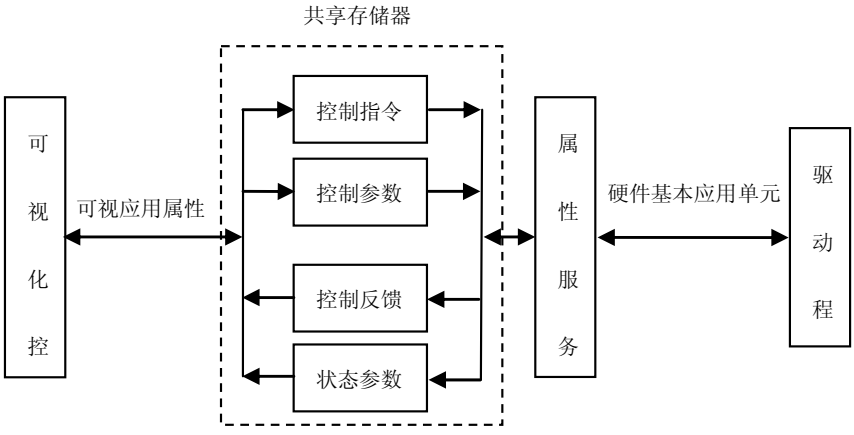

Figure 4. Attribute server is responsible for the transmission of information
图 4. 属性服务器负责信息的传递

统中可视化开发的难度，灵活地构造大小不同功能不同嵌入式可视化系统。该模型有以下特点：

1）以设备为核心的设计，从底层基本硬件操作单元，到顶层的可视应用单元，三层映射关系充分体现了模型对硬件的支持，符合嵌入式具有个性化、专用性的特点，表征了嵌入式系统的本质特征。

应用单元，对应用系统的构建无附加约束。

4）模型实现过程中，对简单的可标准化部分进行了分析，提供自动生成工具，对其余部分也提供了辅助功能，大大降低了难度和工作量。

5）和硬件相关部分集中在第一个映射层，上层的两个映射对硬件操作透明，便于软硬件开发人员的分工。分明的逻辑层次，模型底层规划要求简单、实时，而复杂的逻辑在上两层完全可以实现，也符合嵌入式系统软硬件特点。

6）实现了可视化开发与非可视化开发的无缝链接。非可视化开发只需要利用本模型中的下两层解决方案，而不使用可视化控件集同样可以实现非可视化开发，从而实现了可视化开发与非可视化开发的无缝链接。

要进一步完善该模型，还有很多工作要做：

1）各逻辑层次功能的动态性。尤其是硬件操作单元和应用单元的确定，本文只给出了划分的原则，没有给出明确的数量化标准，这部分工作由开发者自主确定，但却无法保证其方案最为合理。

2）两个工具对映射过程涉及的复杂逻辑，只能给出最简单辅助，对各种特例没有直接支持，这部分工作有待细化，以提供更多的支持，减少开发者的工作量。

2）充分利用了现有嵌入式 GUI。当前嵌入式系统中的可视化开发是以嵌入式 GUI 为核心的，本文的开发模型在顶层使用了可视控件集的方式，用现有 GUI 构造控件集，兼容目前的开发方式。

3）开发模型具有广泛的适应性和灵活性，对操作系统无要求，应用层和 GUI 可以直接使用操作系统的提供的功能，也可以越过控件集直接使用硬件基本

致谢

感谢项目实现过程中的同事；感谢论文撰写过程中提供帮助的实验室老师和同学；感谢论文中参考的参考文献的作者；对于提供论文中隐含的上述提及的支持者以及研究思想和设想的支持者表示感谢。

References (参考文献)

[1] Lv Zhenhua. Research and Implementation of Embedded Visual Development Environment: [D]. Shanghai: East China Normal University, 2006.
吕振华．嵌入式可视化开发环境的研究与实现：[D]．上海：华东师范大学，2006．

[2] Cao Jinnan. Embedded GUI Research and Resources Editor Design:[D].Hangzhou: Zhejiang University,2005.
曹金男．嵌入式 GUI 研究与资源编辑器设计：[D]．杭州：浙江大学，2005．

[3] Sang Nan. Principle and Application of Embedded System Development Technology [M].Beijing: Beijing University of Aeronautics and Astronautics Press, 2002. P117-122.
桑楠．嵌入式系统原理及应用开发技术[M]．北京：北京航空航天大学出版社，2002．P117-122．

[4] Jiang Lidong, Wang Shouwu, Lu Xiaopeng, etc. Principle and Application of Embedded System [M]. Beijing: Machinery Industry Press, 2006. P1-15.
姜立东，王寿武，陆晓鹏，等．嵌入式系统原理与应用 [M]．北京：机械工业出版社，2006．P1-15．

[5] Jonathan Corbet, Alessandro Rubini, Greg Kroah-Hartman. Linux Device Drivers [M].USA: O'Reilly Media, Inc. 2001.

[6] TROLLTECH. Qt for Embedded Linux whitepaper. http: // qt. nokia.com/ .

978-1-935068-23-5 © 2010 SciRes.