

基于 QT/Embedded 与 Mitab 库的嵌入式电子地图^①

王建安, 李少林, 金 晶

(桂林电子科技大学 机电工程学院, 桂林 541004)

摘 要: 针对当前电子地图地理位置信息在嵌入式导航终端上显示精度不高的问题, 设计一种基于 QT/Embedded 与 Mitab 库的嵌入式 GIS 导航电子地图系统. 该系统使用 Qt 作为图形用户界面, 以 MIF 格式的电子地图作为数据源, 通过 Mitab 库读取矢量电子地图几何数据和属性数据, 然后对地图数据进行坐标转换算法处理, 最终将电子地图准确清晰地显示到 LCD 屏幕上. 在嵌入式 Linux 平台下实验结果表明, 该地图具有缩放、漫游等功能, 实现了嵌入式 GIS 地图更准确的可视化. 该系统比较适合应用于旅游景点导航、城市路径规划, 车辆导航等领域.

关键词: 嵌入式 Linux; 电子地图; QT/Embedded; Mitab 库; 坐标转换

Embedded Electronic Map Based on Qt/Embedded and Mitab Library

WANG Jian-An, LI Shao-Lin, JIN Jing

(School of Mechatronics Engineering, Guilin University of Electronic Technology, Guilin 541004, China)

Abstract: Now, electronic map location information displays on embedded navigation terminal with low accuracy. Based on QT/Embedded and the Mitab Library, an embedded GIS navigation electronic map system is designed. The system uses Qt to develop graphics user interface. Geometry and attribute data of electronic map in MIF format is obtained through Mitab library. The extracted map data is processed using coordinates conversion algorithm. Finally the electronic map accurately and clearly displayed on the LCD screen. The experiment in embedded Linux platform indicates that the electronic map has functions such as zooming and roaming, and visualized more accurately. The system can be applied for the tourist, road planning and vehicle navigation.

Key words: embedded Linux; electronic map; QT/Embedded; Mitab Library; coordinates conversion

近年来, 地理信息系统(geographic information system, GIS)的应用日趋广泛. 同时, 随着嵌入式技术的快速发展, 把 GIS 与嵌入式技术融合在一起, 形成一个嵌入式的地理空间集成平台, 是当前 GIS 研究领域的一个重要趋势. 电子地图结合嵌入式 GIS 技术拓展了基于嵌入式设备的应用的范围, 结合 GPS 定位技术, 可以为现实地理空间的能量运动规律和物质的研究提供方便、准确的管理及空间分析手段. 电子地图地理位置信息在嵌入式终端设备上的准确显示, 是实现 GPS 定位、最优路径规划与智能导航等功能的基础. 当前一些研究方法在进行坐标转换时, 没有将电子地图所采用的 WGS84 坐标系, 转化成我国常用的北京

54 坐标系或西安 80 坐标系, 故而在实际应用时, 使电子地图中物体在嵌入式导航终端设备上显示的地理位置与实际地理位置出现较大偏差, 进而影响 GPS 精确定位、路径规划等功能. 本文重点阐述了坐标转换流程与方法, 所采用的坐标转换算法更符合我国实际情况, 可以实现电子地图位置信息更准确的显示.

1 系统架构

系统由用户应用程序、设备驱动程序、和嵌入式 Linux 操作系统 3 部分组成. 系统利用 MapInfo 的中间转换格式 MIF 数据文件作为地图数据, 通过 Mitab 库读取矢量地图, 并进行坐标转换、投影等相关转换处

① 收稿时间:2014-08-15;收到修改稿时间:2014-11-28

理, 通过 Qt/Embedded 将地图绘制到 LCD 屏幕上, 起到可视化定位导航的目的. 系统构架框图如图 1 所示.

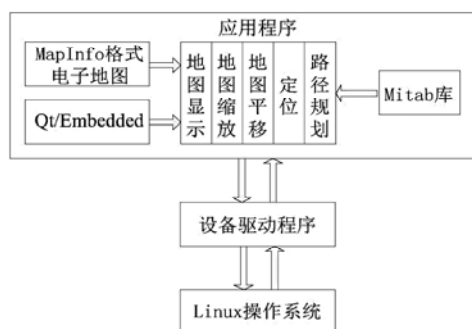


图 1 系统架构

电子地图数据源采用 MapInfo 地图的 MIF 格式, 利用基于 C++ 的开放源码 Mitab 库提供的核心函数对 MIF 文件字段内容进行读写操作. Qt/Embedded 实现图形用户界面设计, 代码设计简单方便, 功能丰富.

本系统选择嵌入式 Linux 操作系统, 它遵循 GPL 条款发布, 成本低, 内核稳定而精悍, 运行所需资源少, 方便裁剪定制, 支持广泛的硬件平台.

设备驱动程序是一种可以使计算机和设备通信的特殊程序, 相当于硬件的接口, 操作系统通过这个接口, 控制硬件设备的工作.

用户应用程序实现电子地图显示、漫游、缩放等功能. 更高级的用户应用程序, 可以实现定位导航和动态路径规则等.

2 电子地图数据的提取

电子地图在导航系统中起基础性作用, 地图的实时显示、漫游、缩放以及路径规划等功能都依赖于电子地图.

2.1 电子地图格式

目前主流的电子地图都是 MapInfo 格式, 由桌面地理信息系统工具 MapInfo 中制作完成. 在嵌入式 Linux 中并没有专门的软件或插件直接支持 MapInfo 格式的电子地图的操作.

MAPINFO 公司并没有对外公布原始的以矢量形式存储的数据格式, 即 TAB 格式. 但其提供地图交换数据格式 (MIF, map info interchange for mat) 供开发人员作二次开发. 该格式空间数据以 ASCII 格式存储, 易生成且可编辑, 但数据不具有拓扑结构.

当用户在 MapInfo 软件中将一张地图以 MIF 格式

转出来后, MapInfo 软件会同时在用户指定的保存目录下生成两个文件 (* .mif 和 * .mid). 其中 * .mif 文件保存了该 MapInfo 地图的结构及所有空间对象的空间信息 (如: 每个点对象的符号样式、点位坐标; 每个线对象的线样式、节点数据、节点坐标; 区域对象的填充模式、每个区域包含的子区域个数及每个区域的节点数等). 而 * .mid 文件则顺序保存了每个空间对象的所有属性信息. 这两个文件都为文本性质的文件, 用户可以通过相应的文件读写方法实现对文件内容的读写^[1].

2.2 Mitab 库

Mitab 库是一个开放源码的 C++ 库, 提供了丰富的接口函数和 C++ 类, 支持 MIF 文件读写、空间对象几何图形和属性数据读取、设置投影方式等, 使用抽象数据模型来解析它所支持的空间对象的数据结构. 在程序设计时调用这个函数库中的相关函数, 可以对 MapInfo 的 TAB 格式文件和 MIF/MID 格式文件进行读写操作^[2]. Mitab 的基本结构是由 Mitab Package、CPL library 和 OGR library 共 3 个部分组成. 其中, Mitab Package 是操作 MapInfo 文件的主要接口, CPL Library 和 OGR Library 是 Mitab 的两个支持库, CPL (common portability library) 是一个可移植的公用库, 封装了通用的地理数据结构和对不同平台的库操作; OGR (OpenGIS simple features reference) 是用来封装简单地理特征 (如点、线、面等) 数据结构和操作的 C++ 库.

2.3 地图数据提取流程

地图中有很多图形对象, MIF 文件把这些图形对象分为点 (point)、直线 (line)、折线 (polyline)、区域 (region)、圆弧 (arc)、文本 (text)、矩形 (rectangle)、圆角矩形 (rounded rectangle)、椭圆 (ellipse) 九类. 利用 Mitab 库按图层读取地图数据 MIF 文件, 具体读取步骤如下^[3]:

1) 调用 IMapInfoFile 类的 SmartOpen() 函数, 打开包含图层数据的 .MIF 格式文件. 打开成功则返回 mitab_handle 类型的数据, 该数据保存了整个 mif 文件的所有数据. 如果打开失败返回 NULL.

2) 调用 GetLayerDefn() 函数, 获取图层属性信息, 利用 GetFieldCount() 读取图层字段数.

3) 调用 GetNextFeaturesId (-1) 函数获得图层中对象首指针, 该函数返回一个整型值. 判断数据是否完成读取. 若完成, 进入过程 5), 否则进入过程 4).

4) 调用 GetFeatureRef() 函数, 返回 TABFeature 类

对象, 该返回数据包含了当前图形对象的所有信息. 通过 `GetFeatureClass()` 函数获得对象类型值, 根据整型返回值判断对象类型. 把获得的 `TABFeature` 类指针强制类型转换为对应的对象类指针. 调用相应的对象类函数, 如: `Getpoint()`、`GetPolyLine()`、`GetRegion()` 提取属性数据和几何数据到内存. 调用 `mitab_c_get_vertex_x()` 提取经纬度, 调用 `mitab_c_get_vertex_y()` 函数提取纬度, 调用 `mitab_c_get_pen_color()` 函数提取画笔颜色等. 然后指针下移进入过程 4), 遍历所有对象.

5) 对象数据读取完毕, 调用 `Close()` 函数关闭 MIF 格式文件, 释放 `MIFFILE` 类指针.

通过以上过程, 最终可以把电子地图每个图层中所有图形对象的几何图形数据和属性信息都提取出来. 地图数据提取流程如图 2 所示.

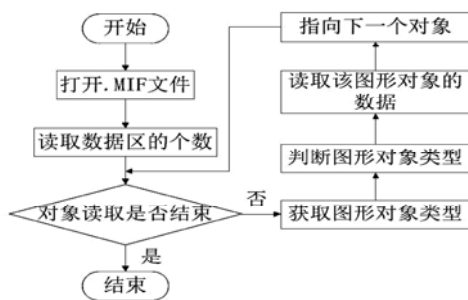


图 2 Mitab 读取 MIF 格式文件流程图

3 电子地图的设计

3.1 QT

Qt 的二维图形引擎是基于 `QPainter` 类的. `QPainter` 既可以绘制几何形状(点、线、矩形、椭圆、弧形、弦形、饼状图、多边形和贝塞尔曲线), 也可以绘制像素映射、图像和文字. 此外, `QPainter` 还支持一些高级特性, 例如反走样、像素混合、渐变填充和矢量路径等. `QPainter` 也支持线性变换, 例如平移、旋转、错切和缩放. 对于用户自定义窗口和绘制一个或者几个项时, 使用 `QPainter` 是理想的. 若需要处理成千上万个项时, `QPainter` 就不那么好用了.

为了解决这个问题, 我们可以使用图形视图框架来进行设计. 在图形视图体系中, 通过 `BSP`(二元空间划分树, `Binary Space Partitioning`)来提供快速的图元查找, 这样就能够实时地显示大场景, 甚至能够显示上百万个图元. 使用视图部件可以使这些图形项可视化,

视图还支持缩放和旋转. 它主要用到的类有 `QGraphicsItem`、`QGraphicsView` 和 `QGraphicsScene` 这三个类. `QGraphicsScene` 提供了图形视图框架中的场景, 相当于一个放置图形项的容器. `QGraphicsView` 类提供一个可视的窗口, 用于显示场景中的项, 一个场景中可以用于多个视图. `QGraphicsItem` 类是场景中各个项的基础类, Qt 内置了很多图形, 比如 `line`、`polygon` 等, 都是继承自 `QGraphicsItem`.

3.2 坐标转换

常用的坐标系有: 大地坐标系、空间直角坐标系、平面直角坐标系. 大地坐标系即经纬度坐标系, 表示地面某一点的位置用 (B, L, H) , 其中 B 为大地纬度, L 为大地经度, H 为大地高. 空间直角坐标系的坐标原点位于参考椭球的中心, 某一点的坐标表示为 (X, Y, Z) , X 轴指向起始子午面与赤道的交点, Y 轴位于赤道面上, 按照右手系与 X 轴垂直, Z 轴指向参考椭球的北极. 平面直角坐标系即通常的笛卡尔直角坐标系, 平面上的点用 (x, y, h) 表示. 由于在进行坐标变换, 高程 H 对结果影响甚微, 所以在进行坐标变换时, 我们一般忽略高程 H .

3.2.1 基准面的转换(WGS-84 大地坐标转换到西安 80 大地坐标)

对空间点坐标的确定必须要用到坐标系, 而在对地理信息的描述中对坐标系的确定又必须要选择基准面. 然而, 不同国家在不同时期用到的基准面不尽相同. 本文中使用的电子地图的数据源是基于 WGS-84(为 GPS 全球定位系统使用而建立的坐标系统)大地坐标系的, 它是用经纬度表示的球面坐标系统, 而我国常用的坐标系统是北京 54 坐标系统和西安 80 坐标系统. 由于他们采用的基准面不同, WGS-84 坐标系与北京 54 坐标系、西安 80 坐标系相差几十米至一百多米. 由此可见, 必须通过基准面转换, 将 WGS-84 坐标转换成北京 54 坐标或西安 80 坐标, 才能更准确实现位置定位、导航等功能. 大地坐标系之间的转换方法一般采用七参数法或三参数法^[4,5]. 通过基准面的变换, 可将电子地图的 WGS-84 大地坐标转换到西安 80 大地坐标.

3.2.2 投影变换(西安 80 大地坐标转换到西安 80 地心坐标)

地图投影就是为解决由不可展的椭球面描绘到平面上的矛盾, 用数学分析的方法建立地球表面上的点与投影平面上点之间的一一对应关系. 地图数据源使

用的是所处经纬度的地理坐标系,而电子地图在终端上进行绘制和几何操作,使用的是平面直角坐标系,因此需通过投影变换,将地理空间中的实体对象经纬度坐标转换为地图平面坐标。目前,我国大多数GIS均采用了与我国基本比例尺地形图一致的投影系统。常用的投影方式有 $高斯-克吕格投影^{[6]}$ (横轴等角圆柱投影)和 $Lambert$ 投影(正轴等角割圆锥投影)。除1:100万地图采用 $Lambert$ 投影,其他均采用 $高斯-克吕格$ 投影。本文用到的是 $高斯-克吕格$ 投影变换,将西安80大地坐标转换到西安80地心坐标。

3.2.3 坐标映射(窗口坐标转换到视口坐标)

经过地图投影后,原来的经纬度坐标数据就转化成了平面上以米或千米为单位的坐标数据。但是,要把地图绘制在嵌入式设备的屏幕上,还需把平面坐标数据转化为屏幕上以像素为单位的坐标数据,即需要通过映射模式实现从“窗口”(逻辑坐标)到“视口”(设备坐标)的变换。窗口是地图平面坐标空间中的一个矩形区域,决定地图平面空间中的几何模型的哪一部分应该被显示,是基于逻辑坐标的。而视口是设备坐标空间中的一个矩形区域,决定应该绘制在设备表面的何处,是基于设备坐标(像素)的。它们之间的比例决定坐标的缩放。Qt中绘图设备的默认坐标系统的原点在窗口左上角,X轴水平向右为正,Y轴垂直向下为正。所以,首先要将绘图设备坐标系统的原点(0,0)转化到窗口的左下角。转化后X和Y轴的正方向没有改变,所以窗口中的所有Y坐标现在都是负数。

电子地图源坐标数据转换到嵌入式设备坐标数据的坐标转换流程如图3所示:

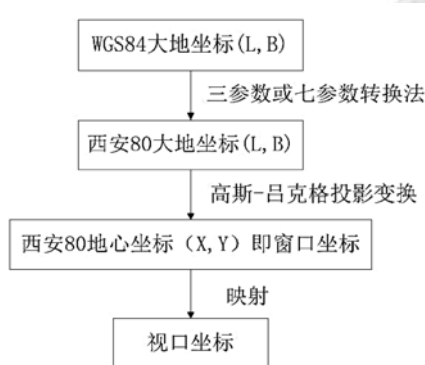


图3 坐标转换流程

3.3 电子地图的绘制

利用 Mitab 库将 MIF 格式电子地图各图层数据提

取出来,提取出的几何图形数据一般为点、线、多边形、椭圆、矩形、文本数据。然后将场景坐标转换成视图坐标和经纬度坐标,获取并设置不同对象的画笔、字体、颜色等信息,根据位置变化和不同缩放比例确定所要显示的区域和图层,实时绘制显示地图。绘图时,可以使用 $QPainter::translate()$ 函数平移坐标系,使用 $QPainter::scale()$ 函数缩放坐标系,设定相应的平移、缩放因子改变地图显示的区域方向和大小,进行地图漫游和缩放操作。由于显示屏的像素有限,初始显示时需对地图的坐标做处理,否则只能显示地图很小的一部分,不便于地图的浏览。处理步骤为:首先,对地图中所有的坐标进行扫描,找出水平方向和垂直方向上的坐标最小值和最大值,计算出地图的初始缩放因子和地图的中心坐标;然后把地图中的坐标转换成屏幕的像素坐标。经过上述过程,在初始化时就可以完整地显示地图。

4 实验结果

首先,在Linux平台下,安装Qt Creator软件,编译Mitab库,搭建好开发环境,通过Qt Creator软件编辑、调试程序。然后,在ARM平台上,移植Qt/Embedded和编译后的Mitab库到S3C6410开发板上,以支持图形的显示和电子地图的读取。最后,在Linux下交叉编译出能够实现所要求功能的应用程序,将交叉编译出的可执行文件移植到ARM开发板中使运行。经实验测试,本文方案能够很好地实现MapInfo格式的电子地图在嵌入式Arm-Linux平台下的显示,并可以对地图进行漫游与缩放操作。与原始的MapInfo地图相比,地图的信息量毫无损失,可以完全复现原始图层。图4是在嵌入式平台上显示的南宁市局部地图。

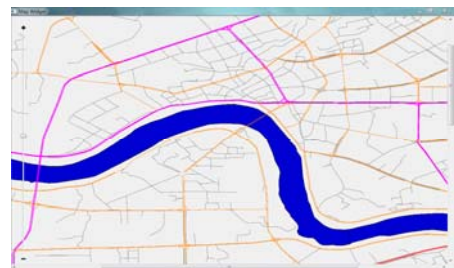


图4 嵌入式平台下的南宁市局部地图

5 结语

实现方案针对嵌入式设备的特点,在硬件环境上,

使用一块 ARM11 为核心的开发板, 搭载嵌入式 Linux 作为操作系统, 以 Qt/Embedded 为开发环境, 实现了将 MapInfo 格式的电子地图显示在嵌入式终端设备上, 并能够实现电子地图显示、漫游和放大等功能基本功能. 用本方案设计的嵌入式电子地图, 与现有的大部分以 WINCE 操作系统和 EVC 为平台的导航电子地图相比, 开发成本更低、制作速度快, 非常适用于旅游景点导游、城市路径规划、车辆导航等方面, 具有一定的使用价值和应用前景.

参考文献

- 1 文广超, 邓寅生, 郭梅荣. 嵌入式电子地图设计中 MIF 文件的读取与转换. 山西建筑, 2007, 33(33): 3.
- 2 马志强, 张德兴, 杨磊. 嵌入式 Linux 平台下地理信息系统. 计算机系统应用, 2011, 20(12): 25-28.
- 3 夏玉杰, 翟艳磊. 基于 Mitab 库的嵌入式 GIS 地图导航设计与实现. 计算机工程与设计, 2011, 32(4): 1304-1305.
- 4 李敏, 沈云中, 刘春. 基于 MapInfo 的电子地图坐标系定义与转换. 测绘工程, 2004, 13(4): 28-31.
- 5 盖森, 熊伟, 刘建忠, 等. 基于 PROJ.4 的空间坐标转换. 测绘工程, 2012, 21(2): 29-31.
- 6 姚俊杰, 沈永增, 房晓菲. 基于 Qt/Embedded 的嵌入式导航电子地图实现. 计算机测量与控制, 2008, 16(4): 570-572.