

doi: 10.7690/bgzdh.2014.03.002

基于组件的电子对抗态势生成与显示系统

张阳, 马孝尊, 郭金良, 李晓燕

(中国洛阳电子装备试验中心, 河南 洛阳 471003)

摘要: 针对电子对抗态势的复杂性, 设计一种基于组件的电子对抗态势生成与显示系统。在分析已有电子对抗态势可视化的相关研究成果的基础上, 建立系统功能图, 采用组件化思想开发的态势数据模块与显示模块的模块设计, 得到了模块的组成示意图, 并对模块中的相关组件和关键技术进行分析。仿真结果表明, 该系统能够较大地提高系统的可扩充性和可重用性。

关键词: 电子对抗; 组件; 系统设计

中图分类号: TJ03 **文献标志码:** A

Electronic Countermeasure Situation Make and Display System Based on Modules

Zhang Yang, Ma Xiaozun, Guo Jinliang, Li Xiaoyan

(China Luoyang Electronic Equipment Test Center, Luoyang 471003, China)

Abstract: Aiming at the complex of electronic countermeasure situation, design electronic countermeasure situation generation and display system based on components. Based on analysis of current electronic countermeasure situation visualization, establish system function figure. Adopt situation data module and display module based on component theory, acquire module composition figure, analyze relative component and key technology of module. The simulation result shows that the system can greatly improve system expansibility and reusability.

Keywords: electronic countermeasure; module; system design

0 引言

电子对抗态势是战场敌我双方在电磁空间的状态和趋势, 具有复杂性、不可见性等特点, 因此造成指挥员难以准确、及时地掌握当前态势, 并做出正确决策。电子对抗态势生成与显示系统(以下简称“系统”)以丰富的可视化形式, 结合数学仿真手段以及实际数据, 将电子对抗态势呈现在指挥员眼前, 具有较大的军事应用价值。

目前, 国内外对于电子对抗态势生成与显示相关领域的研究已经很深入, 并开发出了大量的系统。文献[1]提出了一种电子战仿真系统设计方法, 深入研究了仿真控制平台工作流程和电子战态势信息管理平台。文献[2]介绍了一种基于信号级数据流的雷达对抗仿真系统的基本功能和实现途径, 并指出基于 HLA 协议的仿真系统具有鲜明的技术特点, 便于系统扩展和代码重用。

考虑到电子对抗态势的复杂性, 采用组件技术分析并设计相关系统已经成为系统开发的重要趋势之一^[3]。基于组件的系统具有较强的灵活性、可订制性及可扩充性, 能够更好地服务于作战、训练、试验等任务。基于此, 笔者设计一种基于组件的电

子对抗态势生成与显示系统, 为指挥员提供一种准确把握战场电子对抗态势的有力工具。

1 系统功能设计

系统的主要功能有想定设置、电子对抗态势数据生成与计算、可视化显示等, 如图 1 所示。

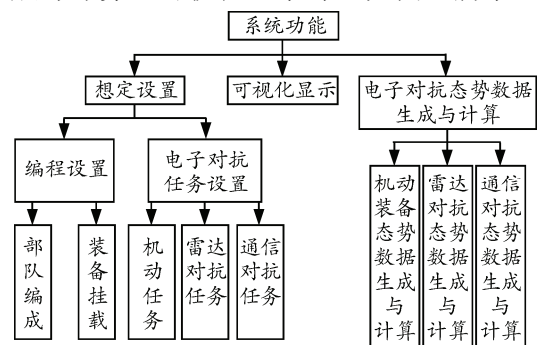


图 1 系统功能

1) 想定设置功能。

该功能包括编成设置和电子对抗任务设置 2 种功能, 支持对红蓝双方编成和电子对抗任务的设置及显示, 将设置好的想定存储在相应的数据库中。

2) 电子对抗态势数据生成与计算功能。

该功能根据想定的设置情况, 结合数据库生成

收稿日期: 2013-10-12; 修回日期: 2013-11-17

作者简介: 张阳(1986—), 男, 河南人, 硕士, 助理研究员, 从事信息对抗仿真与模拟研究。

仿真数据，并实时计算各类电子对抗装备的对抗态势数据。该功能按照对抗手段和作战方式，可以分为机动装备态势数据生成与计算功能、雷达对抗态势数据生成与计算功能与通信对抗态势数据生成与计算功能。

3) 可视化显示功能。

该功能用来调用并组合所需的可视化显示组件，实现对于电子对抗态势数据的可视化显示。

2 基于组件的系统模块设计

系统主要分为运行模块、支持模块以及接口模块，如图 2 所示。

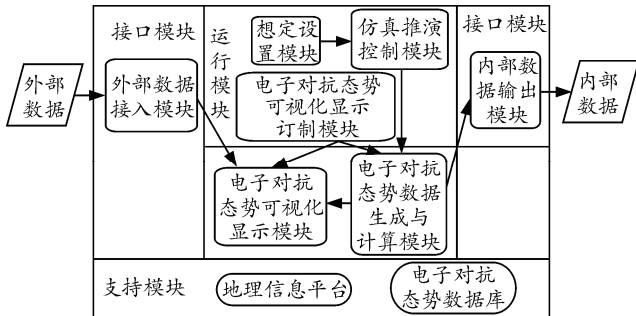


图 2 系统模块

1) 支持模块。

该模块包括地理信息平台与电子对抗态势数据库(简称“数据库”)。

2) 运行模块。

该模块包括想定设置模块、仿真推演控制模块、电子对抗态势数据生成与计算模块(简称“态势数据模块”)、电子对抗态势可视化显示模块(简称“显示模块”)、电子对抗态势可视化显示定制模块(简称“定制模块”)。

3) 接口模块。

该模块包括态势数据模块和显示模块均采用组件化思想开发，便于定制模块的管理和调用，笔者将主要分析和研究这 2 类模块。

外部数据接入模块与内部数据输出模块。

2.1 态势数据模块设计

态势数据模块用于解析想定设置模块设置的战情，生成并计算机动装备态势数据、雷达对抗态势数据、通信对抗态势数据。采用组件化、模块化方式设计并开发态势数据模块。在进行模块设计时，尽可能将其分割为可独立运行、关联度较低的组件；在进行程序编码时，尽可能将各组件封装成*.dll^[4]形式，便于系统的快速组装和调用，如图 3。

态势数据组件模块^[5]包括由机动装备态势数据组件、雷达对抗态势数据组件、通信对抗态势数据组件组成的中层组件，以及飞机、导弹、车辆、雷

达、通信等态势数据组件组成的底层组件。每个底层组件都采用*.dll 形式封装，用户只需按照数据接口定义的格式，输入相关仿真参数，就能得到相应的态势输出数据，如图 4 所示。

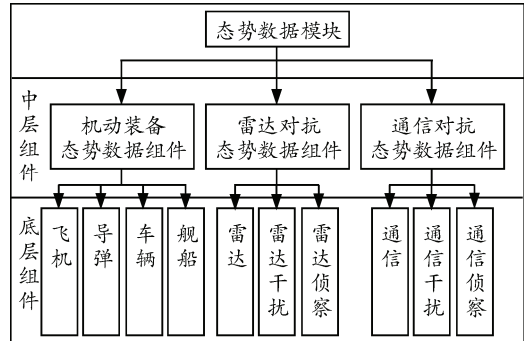


图 3 态势数据模块组件构成示意图

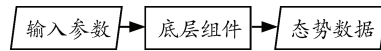


图 4 底层组件数据流程

每个底层组件相对于用户而言都类似于“黑箱”，用户不需要关心组件的具体实现方式以及仿真算法，便于系统按照用户的需求进行组装。

2.2 显示模块设计

显示模块用于接收并解析态势数据模块发送的态势数据，将其转化为显示模块规定的的数据形式，调用相应的可视化组件，进行可视化显示。显示模块同样采用组件化、模块化方式设计并开发，并尽可能将各个组件封装成*.dll 形式，如图 5 所示。

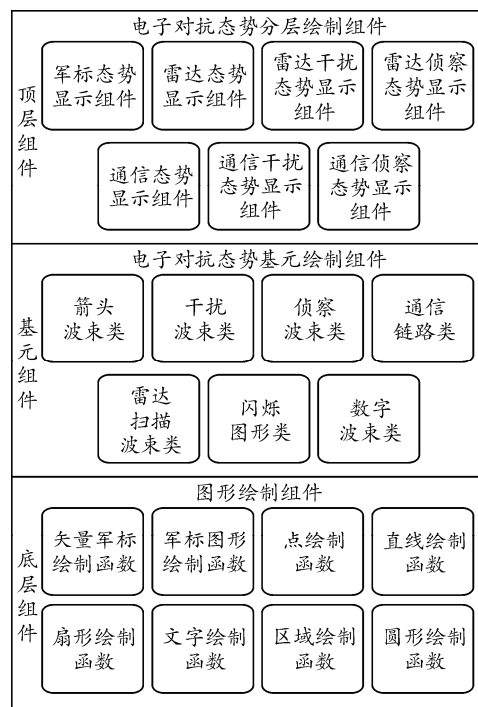


图 5 显示模块组件构成示意图

显示模块由底层组件、基元组件以及顶层组件组成。

1) 底层组件。

底层组件采用 GDI+与 GDI 图形接口函数开发,并结合 MapInfo 地理信息系统,实现在 MapInfo 的图形绘制。底层组件主要由矢量军标绘制函数、军标图形绘制函数、点绘制函数等组成。

2) 基元组件。

基元组件调用底层组件,实现在 MapInfo 的电子对抗态势基元绘制。基元组件主要由箭头波束类、干扰波束类、侦察波束类等组成。

3) 顶层组件。

顶层组件调用基元组件和底层组件,实现 MapInfo 的电子对抗态势分层绘制。顶层组件主要包括雷达态势显示组件、通信态势显示组件等。

底层组件与基元组件均采用*.dll 形式封装,用户只需按照数据接口定义的格式,输入态势数据模块传递的相应数据,就能得到所需要的显示效果,如图 6 所示。

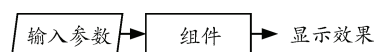


图 6 组件数据流程图

顶层组件、基元组件以及底层组件的调用关系如图 7 所示。

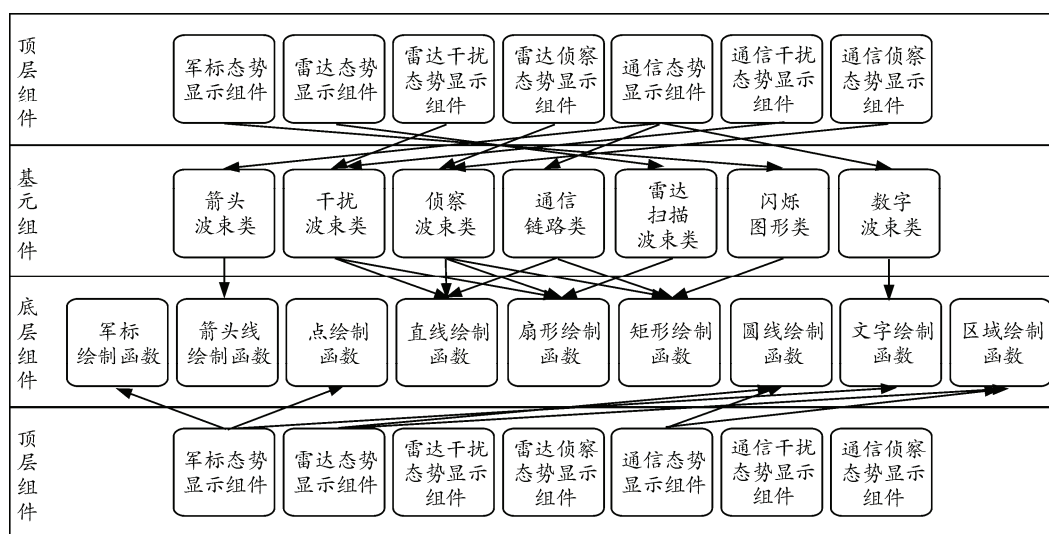


图 7 组件调用关系图

3 系统关键技术分析

笔者采用 VC++6.0 软件开发程序,用嵌入 MapInfo 地理信息系统作为地图,并且采用 GDI+ 图形接口开发底层的显示组件。系统实现的关键技术主要包括地理信息系统分层显示技术、多定时器设置技术等。

3.1 地理信息系统分层显示技术

由于实际战场的电子对抗态势类别多种多样,并且不同指挥员关注的态势不同,因此对于系统显示电子对抗态势进行分层是非常必要的。态势分层显示不仅需要系统设计时考虑好各个模块之间的关系,而且需要地理信息系统的支持。

MapInfo 采用高度结构化的数据去组织地理信息,按照图形对象的叠盖、图元要素在地理位置上的相互关系进行合理分层。MapInfo 图层分层显示^[6]即是对图层进行一定的缩放设置,使得图层各个要

素以图层为单位,按不同的缩放水平来显示。具体设置方法是:点击菜单中的 map 下的 Layer Control,在 Layer Control 对话框中选择所要设置的图层,然后单击 Display,出现 Untitled Display Options 对话框,选中 Display within Zoom Range,依据具体地图各图层的缩放显示需要,指定本图层显示的最大和最小视野。

3.2 多定时器设置技术

定时器设置主要用于显示电子对抗态势中的某些动态效果,如雷达扫描波束、通信链路等,主要应用于基元组件的各个类。由于采用了组件化、模块化的编程方式,并且为了使代码的独立性与可移植性更强,不能采用传统的窗口类设置定时器 WM_TIME 消息响应的方式,只能采用非窗口类设置定时器。