**车联网演示系统功能**

**测试报告**

2017年3月

**目录**

[一、介绍 3](#_Toc478741811)

[1.1 系统简介 3](#_Toc478741812)

[1.2 编写目的 3](#_Toc478741813)

[1.3 术语及缩写词 3](#_Toc478741814)

[1.4 参考资料 3](#_Toc478741815)

[二、测试概要 4](#_Toc478741816)

[2.1 测试环境及平台 4](#_Toc478741817)

[2.2 测试功能单元 5](#_Toc478741818)

[2.3 测试软、硬件环境及配置 5](#_Toc478741819)

[三、测试方案与流程 5](#_Toc478741820)

[3.1 基础功能的测试方案及流程 5](#_Toc478741821)

[3.2 碰撞避免的测试方案及流程方案及流程 6](#_Toc478741822)

[3.3 视频传输功能的测试方案及流程 7](#_Toc478741823)

[3.4 智能泊车的测试方案及流程 8](#_Toc478741824)

[四、测试结果及分析 9](#_Toc478741825)

[4.1 基础功能测试结果 9](#_Toc478741826)

[4.2 碰撞避免功能测试结果 11](#_Toc478741827)

[4.3 视频传输功能测试结果 11](#_Toc478741828)

[4.4 智能泊车功能测试结果 12](#_Toc478741829)

[4.5测试结果数据详情 13](#_Toc478741830)

[五、测试结论及建议 13](#_Toc478741831)

#### 一、介绍

##### 1.1 系统简介

西安电子科技大学先进交通技术实验室车联网演示系统的功能为实现车联网平台开发和系统演示，即在有交通事故出现的道路上实现事故信息的预警广播转发、直接疏导或引导车辆避让事故点、防止交通拥塞、提高交通效率的场景演示功能，达到理论与实践相结合的目的。

##### 1.2 编写目的

本测试目的为验证先进交通实验室演示系统功能的实现和总结演示系统的问题所在，从而指导下一步的系统开发任务，控制产品的质量，更好地完成系统的功能、稳定性和可靠性。

本文档是为硬件开发人员、软件开发人员和软件维护人员及相关科研人员制定的。

##### 1.3 术语及缩写词

|  |  |
| --- | --- |
| **缩写、术语** | **意义** |
| 演示系统、本系统 | 先进交通技术实验室车联网演示系统 |
| 服务器后台 | 车联网演示系统服务器与后台系统 |
| 智能小车、小车 | 搭载Arduino控制器的智能小车 |

##### 1.4 参考资料

参考资料：

《车联网核心技术可视化关键问题研究》

《智能交通演示系统平台搭建\_控制系统技术方案》

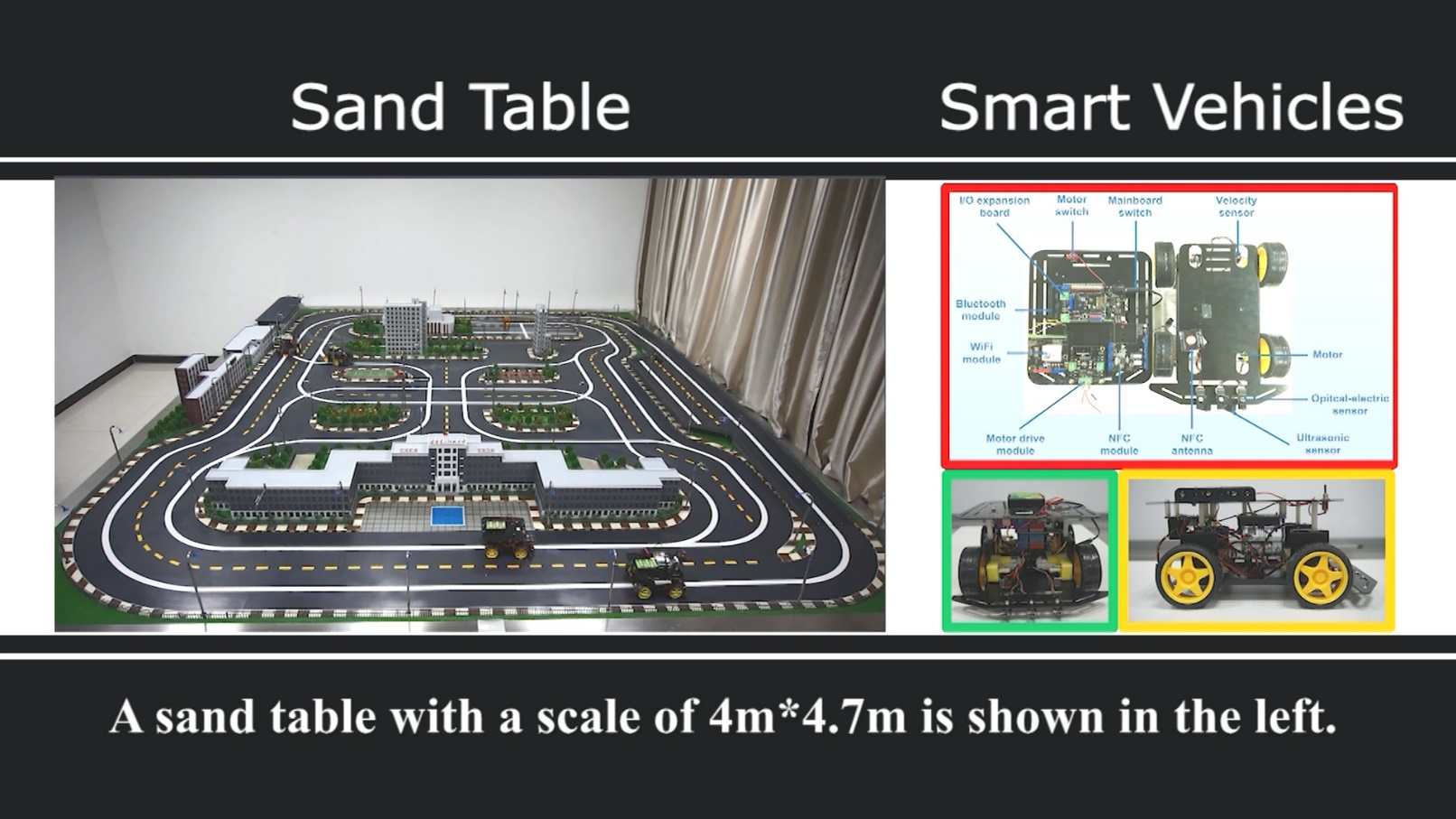
《智能交通演示系统平台搭建\_演示沙盘制作指标要求》

《车联网演示系统的视频拍摄剧本\_修订版本三》

#### 二、测试概要

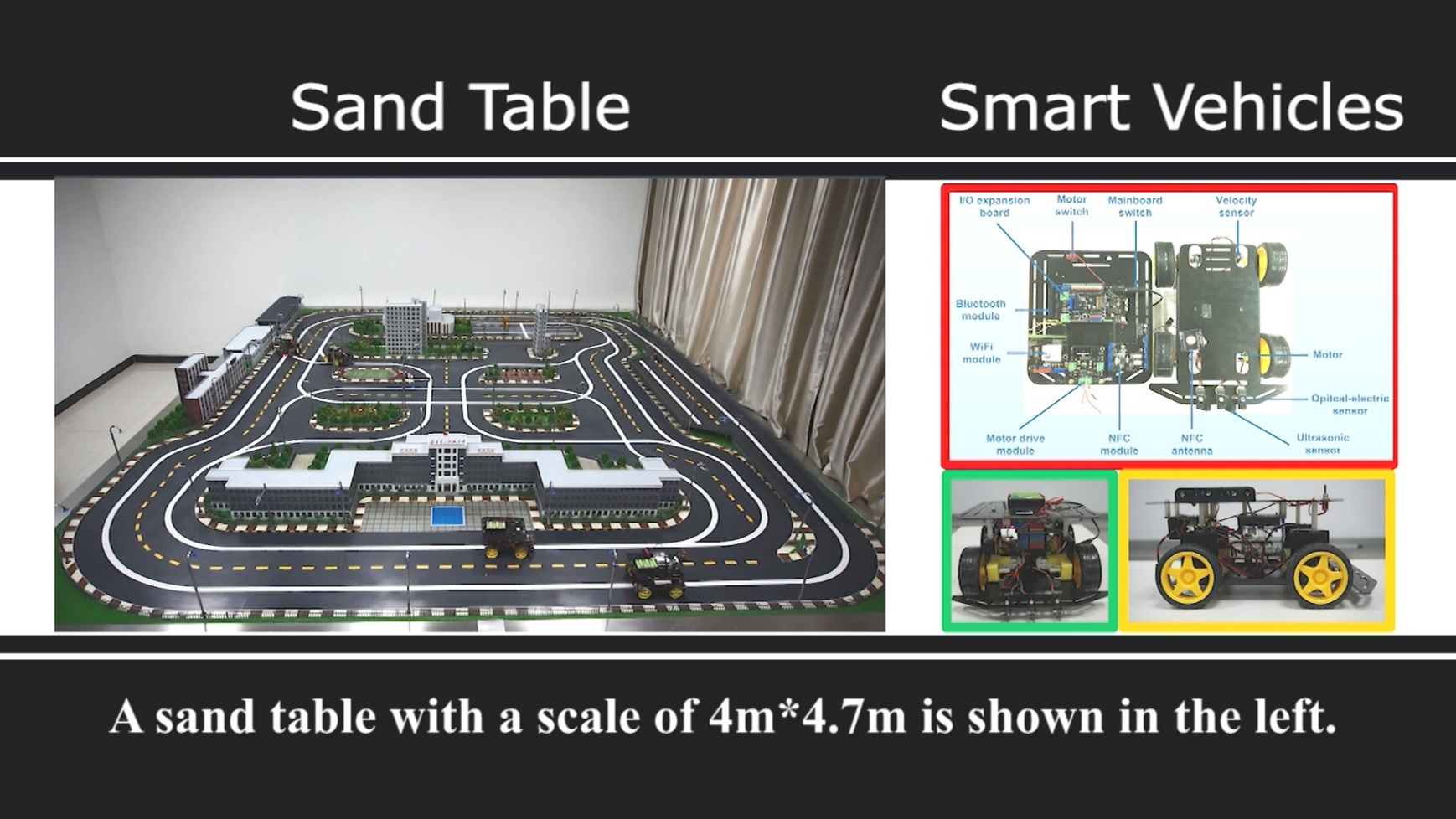
##### 2.1 测试环境及平台

演示系统测试的地点位于西安电子科技大学先进交通实验室，演示场景为以西安电子科技大学为模型的微缩沙盘，如图1所示。



1. 微缩沙盘

车载平台则为搭载Arduino控制器的智能小车，Arduino是一款便捷灵活、方便上手的开源电子原型平台，包含硬件（各种型号的Arduino板）和软件（Arduino IDE），车载平台如图2所示。



1. 车载平台

演示功能测试的环境为一个混合复杂的通信环境，在一个35平方米左右的室内空间充斥着若干WiFi接入点发出的电磁信号，以及周围移动终端设备的蓝牙模块发出的电磁信号，室内温度15~20℃。

##### 2.2 测试功能单元

测试的功能单元主要为车联网演示系统的若干典型应用场景：精确定位，碰撞避免，视频传输与智能泊车等实际演示场景下的系统性能指标，以及循迹功能、超声波检测等基础功能的可用性与稳定性。

##### 2.3 测试软、硬件环境及配置

|  |  |  |
| --- | --- | --- |
| **项目** | **测试环境及配置** | **备注** |
| 本地客户端 | 4G内存、500G硬盘存储空间 | 无 |
| 服务器 | 8G内存、500G硬盘存储空间 | 无 |
| PC操作系统 | Windows7或Windows10、Linux系统Debian型 | 无 |
| 软件需求 | DOS控制台 | 系统自带 |
| PC调试工具 | 通信连接性测试软件、安卓ADB、Tomcat等 | 无 |
| 设备调试工具 | Arduino调试平台 | 无 |
| 智能移动端 | 小米4C：2G RAM、ROM 16G | 无 |
| 移动端系统 | MIUI 6（基于Android 5.1） | 无 |
| 内网转发 | Oray内网穿透3.3.014073版本 | 无 |

#### 三、测试方案与流程

##### 3.1 基础功能的测试方案及流程

(1) NFC定位功能

定位功能使用的是NFC模块，通过在沙盘底下布置相应的NFC卡片，智能小车底盘安装相应的接收器，当小车经过沙盘底下贴有NFC卡片的路面时，小车的接收器即开始扫描该处的NFC卡片信息。

测试用例的卡片规格为8.5cm x 5.3cm，通过以不同的距离间隔布置卡片，判别智能小车以匀速1.5m/s经过时是否可以稳定读取正确的信息。随后根据精度测定逐渐调小NFC卡片的间距，据此求得最佳的卡片读取精度。

(2) 超声波功能

超声波模块装载于智能小车的车前方，主要用户检测车前的其他车辆或者障碍，为智能小车行进过程中的车辆安全起到一定的保护作用。

测试流程为启动小车的程序，并接入安装好的超声波模块，选定两种场景,一种为小车跟随前车行进的时候测试其追尾过程的制动性；第二种场景为小车行进过程中出现误驾驶冲出道路的情况，判定小车是否能够及时在与障碍物碰撞之前紧急制动。

(3) 蓝牙与WiFi模块

蓝牙模块主要用户车辆与移动设备的通信，传输行进指令及定位的信息，同时还可以做数据传输的上位机，蓝牙使用频段为2.4GHz。WiFi模块的主要功能主要为车与车的通信及车与路设之间的通信。其基础功能测试主要为在一个复杂的磁场环境中，WiFi数据通信传输是否会受到干扰，以及其传输的可靠性与稳定性，测试协议为802.11b/g，重复测试大量数据分组。

蓝牙测试流程为测试若干次不同场景中蓝牙开启之后连接的成功率，以及小车前进时数据传输的数据的丢失率。对于WiFi模块而言，其流程则为小车前进时循环发送数据至相邻车辆及路设，测试其WiFi模块连接的稳定性及数据的分组丢失情况。

(4) 循迹基础功能

对于智能小车而言，其大部分的功能演示都以此为基础，所以该基础功能的可用性与稳定性直接决定了整车的性能，拟采取为小车搭载全部的功能模块，设定其在正常使用时的所需载重能力，分步骤让小车在不同的路况下前进，测试多次试验下其直行及通过路口的场景。

测试流程为小车在所需的模块载重性能的情况下，在不同的路况下的循迹的准确率，如直道、丁字路口、十字路口及坡道等等，每种循迹场景重复测试150~200次，记录其循迹的相关数据。

##### 3.2 碰撞避免的测试方案及流程方案及流程

(1) 丁字路口碰撞避免

在一个丁字路口会车场景中，小车前往目的地的途中它们将同时到达一个丁字路口，两车通过碰撞检测智能地作出决策：直行车辆优先通过，而另外一辆车停下等待直行车辆通过后转弯通过路口。预设两车即将发生碰撞，重复测试100~150次。

(2) 十字路口碰撞避免

在一个十字路口会车场景中，两辆汽车从各自的起点出发，分别开往各自的目的地。该场景包含紧急停车，碰撞避免等功能。在行车途中，两辆车在中间十字路口进行会车。在视距无法判断的场景下，他们通过定位系统实时得到自己的精确位置信息，并通过WiFi模块将位置信息传输给其它车辆，当两辆车的距离小于某一安全距离时，通过碰撞避免算法及时避让，随后重新开始行程。预设两车即将发生碰撞，重复测试100~150次。

##### 3.3 视频传输功能的测试方案及流程

视频传输功能将完善后车在行驶过程中视角受限的缺陷，使后车提前掌握前方非视距内的路况，以便灵活应对相应的状况发生。

视频传输功能实现使用的模块是RT5350F无线传输模块，兼容IEEE 802.11 b/g标准,具有GPIO，SPI，I2C，I2S，PCM，UART及JTAG接口， 集成802.11n 1×1 MAC/基带处理器，如图3所示。视频传输的显示终端为Raspberry Pi 2B，改设备可以插入便携式网卡，且具有一定的终端显示能力，高容量的内存空间及简单易操作的Debian系统，如图4所示。

1. RT5350F无线传输模块
2. Raspberry Pi 2B模块

(1) V2V视频传输

V2V视频传输则是两车在行进过程中进行视频流的传输，其中一辆车为前车，作为视频传输的源点，即视频采集的车辆；另一辆车作为后车，即视频流的接收端。数据通过AP接入的方式传输至第二辆车，测试采用了正常驾驶过程中的工作流程，前车将自身摄像头采集到的影像信息通过WIFI模块向后车进行传输，后车的终端接收到影像资料后实时地将其显示到车载屏幕上，测试终端画面的帧数及数据丢失率，无线终端的系统稳定性，重复测试100~150次，每次传输时间5分钟。在此基础上我们又测试了多跳视频传输，即一辆中间车辆将视频数据转发到后车，中继车辆同样具有视频接收的能力。测试过程重复测试100~150次，每次传输时间5分钟。

(2) V2I视频传输

车与路设之间进行视频传输：车辆通过摄像头捕捉在行进过程中前方的视频影像资料。随后，视频数据通过车上的WIFI模块进行传输，由路设进行接收。

测试的流程为小车搭载视频传输模块，在演示沙盘上行进的时候，将道路前方的画面实时的传输到路设端（该路设假定为实验室的一台具有无线网卡的PC），测试视频传输建立的时间，路设端的视频画面的帧数及数据的丢失率。重复测试以上过程150~200次，每次传输时间5分钟。

##### 3.4 智能泊车的测试方案及流程

智能泊车的测试过程分为两部分，一部分为宏观路径规划，即引导车辆从道路中驾驶至停车场区域门口，另一部分为微观路径指导，即指导车辆进入停车场。宏观路径规划主要测试车位搜寻系统-Parking Master，对于微观路径规划则测试一整套智能停车管理系统（IPMS）。

(1) 宏观路径规划

宏观路径规划的功能为：假设司机现在想从主楼到停车场，查找离他最近的停车位。随后打开车载终端上的软件，软件提供了准确车位查询、语音播报导航、查询实时车位数据等功能。此时，司机点击预约停车，选择出发地和目的地，向后台发送路径规划请求。随后，后台服务器接收到该司机的请求，通过执行路径规划算法，为司机规划一条路径，并将此规划后的路径发送给司机。司机收到消息后，按照路径提示前往最近的停车场。

测试流程为对该Parking Master软件进行测试，从应用启动速度，路径规划的准确性，以及应用的稳定性等方面展开测试。测试过程重复200~250次，选定不同的路径需求点测试，以及不同网络情况下（WiFi数据或移动数据）的应用使用情况。

(2) 微观路径规划

在进入停车场门口时，假设司机启动车辆的无人驾驶模式。随后，门禁设施感应到车辆即将进入停车场，后台便为其规划了一条最近的停车路径，车辆根据此路径顺利找到车位进行自动泊车。此时，车位信息由空闲变为占用。当司机想要离开时，IPMS将此信息发送给车辆，车辆便根据之前的路径自动的驶出停车场，这时，车位信息由占用变为空闲，同时前端显示车位占用时长，这将有助于停车场的进行收费。

测试流程为当车辆到达停车场门口时，后台设定相应的停车场进入路径，道闸抬起，小车循迹进入目标车位，测试的数据主要为后台规划路线的准确性，道闸自动检测车辆到来时抬起及放下测试，小车进入车位的成功率及后台数据的准确性。测试过程重复200~250次，选定多种路径进行测试。

#### 四、测试结果及分析

##### 4.1 基础功能测试结果

(1) NFC定位功能

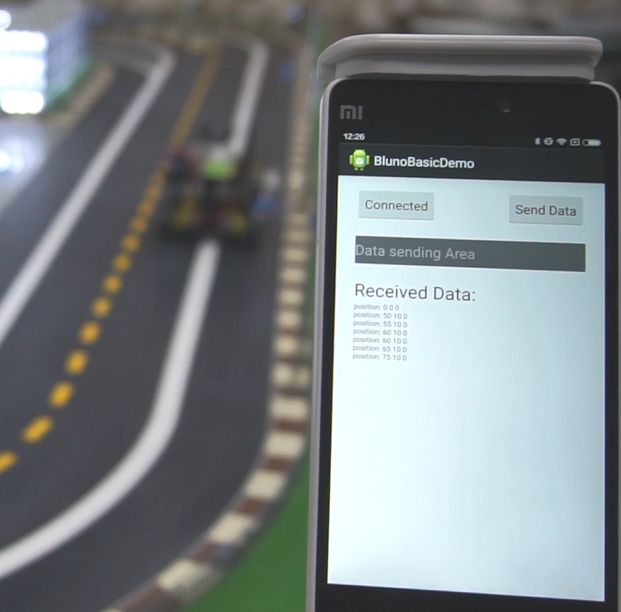
通过测试NFC卡片的放置，测试数据为卡片位于沙盘底下隔板可检测到信息，但在部分路面厚的地方无法读取数据。测试NFC卡片排放距离的结果，得出卡片最佳的读取精度为卡片间隔为5.3cm。经过重复测试200次左右，每次读取一条路面15张连续信息卡片，信息成功读取率可达98%。

(2) 超声波功能

测试通过不同超声波的识别距离进行调整，得出识别距离为20cm左右。超声波模块测试结果为车辆追尾的制动距离为15cm左右，超声波安装的高度为离地面8cm高左右最佳，此时可以及时检测到路边建筑及避免误判的情况。

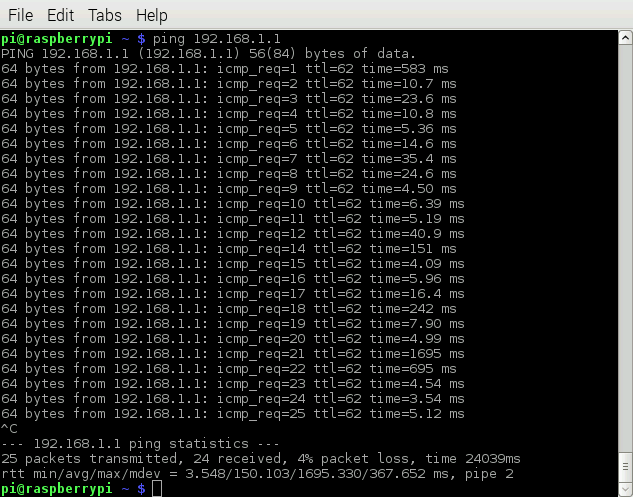
(3) 蓝牙模块与WiFi模块

蓝牙模块的测试结果为传输定位结果的数据，通过多次的测试，读取卡片数据150次左右，蓝牙接收数据的准确率99%，如图5所示。



1. 蓝牙定位数据接收

WiFi模块进行数据通信的结果大致如图6所示。经过约150次的测试，每次测试25个分组，每个分组的大小为64比特，由最后的统计得到分组的丢失率为7%左右，鉴于该数据传输模块作为视频帧数的传输模块，故丢失少量的数据只是影响部分画面帧，不影响整体功能。



1. 数据连通性测试

(4) 循迹基础功能

循迹模块测试结果，经过多次的改良，由起初的3个传感器检测道路改为5个传感器检测道路黑线，200次左右循环跑圈的测试结果表明循迹的功能在直道表现优异，成功率可达99.5%；在丁字分叉路口还存在部分窜线的情况，错误率控制在10%以内。在十字路口表现正常，成功率可达95%；弯道循迹正常，在部分弯道由于转弯半径问题还存在不足，后续继续改良。

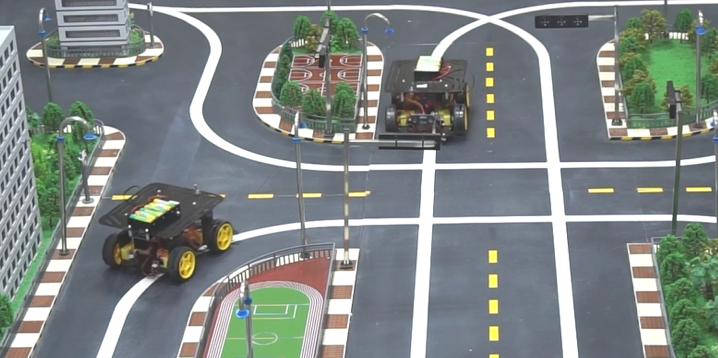
##### 4.2 碰撞避免功能测试结果

(1) 丁字路口碰撞避免

丁字路口的碰撞避免的测试结果为：测试两车相遇的次数200左右，通信检测半径为50cm左右，其中成功制动的次数190次，制动优良的次数186次，丁字路口的碰撞避免成功率可达93%，且后续路线汇集跟车无异常情况。

(2) 十字路口碰撞避免

十字路口两车交汇时，由于不存在最后的并线过程，测试结果表明，重复测试十字路口碰撞过程200次左右，合适的通信检测半径为60cm左右，其中成功制动的次数190次，制动优良的次数188次，丁字路口的碰撞避免成功率可达94%左右，后续小车重新前进不存在异常情况。如图7所示。



1. 十字路口碰撞避免

##### 4.3 视频传输功能测试结果

(1) V2V视频传输

V2V视频传输一跳的传输过程测试结果为：前车发送的视频流在后车通信范围则可马上接收，在该沙盘内通信范围为全沙盘地图，视频一跳的数据丢失率控制在5%以内，最终显示的画面时延为500ms左右（非传输时延），小车显示终端最终的显示测试表明，视频流的帧数刷新率不高，仅为5~10fps左右。画面质量中等，分析可能是由于终端的优化问题以及受限于设备的GPU性能。

后续的多跳视频传输性能没有单跳的高，相对于单跳视频传输性能整体下降10%，推荐降低画面质量及提升设备GPU来提升画面的整体质量。

(2) V2I视频传输

V2I视频传输通过小车将视频传输到路设，画面整体时延较低，为单跳传输。测试结果表明画面帧丢失率在5%以内，实时视频的显示画面帧数为15~20帧，画面质量较高，具有较好的效果，整体测试的成功率在91%。后续的功能开发主要为开发适应本实验室路设视频接收端的软件（借用第三方软件），使其更符合实验室演示的要求。如图8所示。



1. V2I视频传输

##### 4.4 智能泊车功能测试结果

(1) 宏观路径规划

宏观路径规划的测试结果：Parking Master应用软件的基础功能状态良好，运行之后占用手机内存100M左右（约为手机2G内存的5%），应用的外围性能达到良好评级；路径规划的准确度测试总数为150次左右，不同次数的随机选点正确，数据传输正确，成功次数为140次左右，部分路径规划不成功分析原因为服务器端的路径规划数据不完整，存在部分缺省规划。

(2) 微观路径规划

微观路径规划重复测试完整流程160次左右，停车场道闸系统正常，门禁检测小车状态良好，抬起与回置成功率可到99%，无误判行为，说明该部分功能通过要求。智能停车管理系统（IPMS）的测试结果表明可正确完成停车场的占用情况管理，将车场的占用时间及入库路线及时规划，占用计时精度可精确至秒级，测试成功百分比约为90%；远程服务器的数据交互不完善，部分信息无法及时通过因特网更新停车场至Parking Master应用。

##### 4.5测试结果数据详情

测试结果如下所示，其中测试合格率Q为：测试通过次数/该项目总测试次数\*100%，结果评价依高到低分为：通过(Q>90%)、待改进(90%>Q >75%) 、不通过(Q <70%)。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **测试单元** | **类目** | **测试描述** | **测试合格率**Q | **测试结果** |
| 基础功能 | NFC定位 | NFC信息  读取的准确度 | 176/180\*100%=98% | 通过 |
| 超声波功能 | 检测目标  物体的准确性 | 158/150\*100%=95% | 通过 |
| 蓝牙与WiFi | 数据正确  传输与否 | 140/150\*100%=93% | 通过 |
| 循迹功能 | 各类场景  是否成功通过 | 192/200\*100%=96% | 通过 |
| 碰撞避免功能 | 丁字路线  碰撞功能 | 避让成功与否及  回归正常路线与否 | 186/200\*100%=93% | 通过 |
| 十字路口  性能测试 | 避让成功与否及  回归正常路线与否 | 188/200\*100%=94% | 通过 |
| 视频传输功能 | V2I  视频传输 | 传输接收80%以上时延3s以下与否 | 137/150\*100%=91% | 通过 |
| V2V  视频传输 | 传输接收80%以上时延3s以下与否 | 128/150\*100%=85% | 待改进 |
| 自动泊车功能 | Parking Master | 宏观路线指导的  准确性及完善度 | 140/150\*100%=93% | 通过 |
| 智能停车  管理系统 | 微观泊车的  准确性及完善度 | 成功率为90%  完善度80% | 待改进 |
| 整体功能联调 | 整体联调 | 上述模块和功能同时与系统一起运行 | 220/240\*100%=92% | 通过 |

#### 五、测试结论及建议

演示系统的功能联调经过测试，目前功能成功率在92%左右，目标功能基本实现，整体性能满足要求，结论为该车联网演示系统测试通过。测试性能良好的为基础功能演示、碰撞避免功能演示，性能正常的为V2V与V2I的视频传输功能，自动泊车功能的系统架构无潜在缺陷，但存在后续完善工作，如智能停车管理系统（IPMS）。测试建议为车联网演示系统的现阶段功能良好，可以进入下一阶段的项目实现。