

---

## 目录

前言.....	4
1 Autobot 机器人的结构设计.....	5
2 Autobot 机器人的硬件设计.....	6
2.1 硬件结构概述.....	6
2.2 主要控制系统选型.....	6
2.2.1 上位机主控选型.....	6
2.2.2 下位机主控选型.....	8
2.3 电机驱动电路设计.....	9
2.4 传感器选型.....	10
2.4.1 惯性测量单元.....	10
2.4.2 激光雷达.....	11
2.4.3 深度相机.....	12
2.4.4 雷达与测距模块.....	13
2.4.5 电源电路设计.....	13
2.5 本章小结.....	13
3 Autobot 机器人的软件设计.....	14
3.1 机器人下位机软件设计.....	14
3.1.1 基于 FreeRTOS 操作系统的程序框架设计.....	14
3.1.2 电机控制程序设计.....	15
3.1.3 基于 DMA 的 ADC 数据读取程序设计.....	15
3.1.4 基于 VCP 的模拟串口通讯设计.....	15

---

3.1.5	基于 roserial 的下位机 ROS 通讯设计 .....	15
3.2	机器人上位机软件设计 .....	15
3.2.1	基于 roserial 的上位机 ROS 通讯设计 .....	15
3.2.2	机器人实体模型的 TF 分析 .....	15
3.2.3	基于 RIVZ 的机器人传感器数据显示 .....	15
3.2.4	基于话题通讯的机器控制程序设计 .....	15
3.3	本章小结 .....	15
4	人机交互系统设计 .....	16
4.1	基于 PyQt5 的控制软件设计 .....	16
4.1.1	控制界面功能总览 .....	16
4.1.2	基于 matplotlib 的传感器数据图形化显示设计 .....	16
4.2	基于 stm32 的网络遥控设计 .....	16
4.3	遥控设计概述 .....	16
4.3.1	OLED 的 GUI 界面设计 .....	16
4.3.2	局域网内 stm32 与 ROS 系统的通信系统设计 .....	16
4.4	基于 Web 的网页控制界面设计 .....	16
4.5	本章小结 .....	16
5	SLAM 算法分析与实践 .....	17
5.1	SLAM 算法概述 .....	17
5.2	激光 SLAM .....	17
5.2.1	Gmapping .....	17
5.2.2	Cartographer .....	17
5.3	视觉 SLAM .....	17

---

6	人体姿态估计系统设计 .....	18
6.1	人体姿态估计概述.....	18
6.2	OpenPose、AlphaPose、Adafuse 算法对比 .....	18
6.2.1	OpenPose 原理与实验 .....	18
6.2.2	AlphaPose 原理与实验 .....	18
6.2.3	Adafuse 原理与实验 .....	18
6.2.4	对比分析 .....	18
6.3	Autobot 机器人多人人体姿态估计实验.....	18
7	总结与展望 .....	19
	参考文献.....	20

---

## 前言

---

## 1 Autobot 机器人的结构设计

## 2 Autobot 机器人的硬件设计

### 2.1 硬件结构概述

### 2.2 主要控制系统选型

#### 2.2.1 上位机主控选型

上位机作为机器人的大脑，承担机器人最复杂的计算任务，如建图、导航、人体姿态估计等。所以对上位机型号的选择尤为重要，它决定能否自如的完成机器人的计算任务。另外，在机器人设计时不仅要考虑当前功能的需要，还要留有一定计算冗余，便于后期增强神经网络算法深度，用于实现更复杂的功能。常用上位机主控有树莓派、JetsonNano、JetsonTX2、miniPC 等，下表选择了其中具有代表性的几款产品进行对比：

表 2-1 上位机主控对比

Tab.2-1 Comparison of host computer controls

	Raspberry-Pi-4	Jetson TX2	Jetson Nano	Intel-i5- 6300HQ miniPC
CPU	1.5GHz 四核 64 位 ARM Cortex-A72 CPU	HMP Dual Denver 2/2 MB L2 +Quad ARM® A57/2 MB L2	四 核  ARM Cortex A57	2.3GHz 四 核  Intel-i5 CPU 最大睿频频率： 3.2GHz 缓存：6MB Intel Smart Cache 散热设计功耗 (TDP): 45W
GPU	Broadcom	NVIA	128	NviaN16P-

	VeoCore IV@500MHz	Pascal™, 256 CUDA 核心	Maxwell cores	GX(GTX960M), VRAM 4G GDDR5
内存	2GB/4GB/8GB 可选	8GB 128bit LPDDR4	4GB 64bit LPDDR4	8GB DDR4- 2133
USB	2 x USB2.0 2 x USB3.0	1 x USB2.0 1 x USB3.0	1 x USB2.0 4 x USB3.0	2 x USB2.0 2 x USB3.0
WiFi	802.11b/g/n/ac 2.4GHz/5GHz 双频	802.11 2x2 ac/BT Ready	M.2 Key E 扩展 WiFi	802.11bgn
功耗	15W	≤7.5W	5-10W	65W(UMA model) 120W (Nvia GPU model)

如上表所述，树莓派虽功耗较低，但其整体性能有限，对于同时部署 SLAM 算法与人体姿态估计的神经网络算法来说，略显不足。Jetson 系列是 NVIA 公司推出的嵌入式人工智能超级计算平台，在思科的电视电话会议系统、法拉赫的工厂自动化等都采用了 Jetson 系列计算平台。JetsonNano 的四核 A57CPU 性能较为一般，与 JetsonTX2 的 Denver 核心要差不少。JetsonTX2 的 CPU 相较于 Intel Core 系列的 I5-3XXXU 计算性能大致相当，所以相较于 i5-6300HQ 的 miniPC 还要差一些。

本人在使用 JetsonTX2 时，ARM 架构下的部分功能包与工具稍有不足。在计算平台使用神经网络进行实验时，会用到较多不同深度学习框架，如 Tensorflow、Pytorch 等，其对系统环境的依赖有严格的要求，所以我们在实验时会使用 Anaconda 创建多个不同虚拟环境，用于支撑不同算法的运行。但 Anaconda 并不支持 ARM 架构，有着类似功能的 miniconda、Archiconda 等对程序功能包

---

的支持却不如 Anaconda。所以在实际使用中，Intel i5 的 miniPC 更为便利。

所以考虑主控性能与实际使用，miniPC 应为首选。但其额外突出的功耗却是十分值得考虑的问题。对于普通的实验平台，整体框架较小，驱动能力与承载能力有限，所以更加青睐高性能低功耗的计算平台。但本机器人的设计为实用平台，而非实验平台，其驱动能力与承载能力足以支撑 miniPC 的功耗。

综上所述，本机器人的主控型号选择，确定为搭载 GTX960M 的 Intel-i5-6300HQ miniPC。

### 2.2.2 下位机主控选型

下位机主要用于控制电机、读取传感器数据、与上位机通讯等功能。由于 SLAM 算法中的导航算法大多基于概率模型，加上计算速率与传输速率的影响，仅仅依靠上位机很难实现实时性特别高的功能。尽管机器人设计的目标为低速移动机器人，出于安全考虑，机器人也要有多重安全防护措施。所以上位机 SLAM 算法的避障处理作为第一重防护，下位机直接读取的测距模块或雷达的距离信息作为第二重防护，为避免电信号受磁场、短路、断路等影响，在机器人机身布置防撞条作为机器人的第三重防护。所以下位机要尽量准确、快速的读取传感器信息，灵活、精准的实现通讯功能。

较为常用的下位机主控有 STC51、MSP430、Arduino、STM32、飞思卡尔等。MSP430 系列资料较少、飞思卡尔价格较高、STC51 与 Arduino 的性能较 STM32 相差较多，ST 公司推出的 STM32CubeMX、CubeMonitor、X-CUBE-AI 等为 STM32 的开发提供了极大便利。所以下位机主控确定在 STM32 单片机系列。

综合考虑前文说明的下位机主控要求，将主控型号的锁定在主流 MCU 和高性能 MCU 之间。下面选择两类中具有代表性的两款同配置单片机进行对比。



表 2-1 单片机参数对比

Tab.2-1 Comparison of host computer controls

单片机型号	STM32F103VGT6	STM32F407VGT6
Frequency(MHz)	72	168
Core	Cortex-M3	Cortex-M4
I2C	2	3
CAN	1	2
USB OTG_HS/FS	0	1 + 1
Ethernet	0	1
U(S)ART	3 + 2	4 + 2

如上表所述, STM32F4 系列主控频率更高, 处理性能更强。我们常用的 OLED、IMU 等大多要用到 I2C 或 USART 通信, 所以从通信接口看, STM32F4 系列更胜一筹。STM32 与 ROS 间使用串口通信时, 需要借用 USB-TTL 电路实现, 本人在魔方机器人的设计中使用 USB-TTL 模块通信时, 常遇到信号不稳定与信号乱码情况, 所以在本机器人的设计中使用 VCP 虚拟串口实现, 这依赖于单片机的 USB OTG\_HS/USB OTG\_FS。为使机器人配置更加灵活, 机器人上位机与下位机还可使用 Ethernet 通信, 所以选定 STM32F4 系列单片机。

在下位机的程序设计中, 传感器数据读取、OLED 显示、通讯、APP 等互有交流, 但运行互不影响, 所以使用嵌入式操作系统将这几个功能分别置于不同线程运行, 能更大效率利用单片机性能。使用实时操作系统依赖于较大 FLASH。

综上, 考虑运行性能、FLASH 大小、读取传感器占用引脚等因素, 选定 LQFP100 封装的 STM32F407VGT6 作为下位机主控。

## 2.3 电机驱动电路设计

---

## 2.4 传感器选型

### 2.4.1 惯性测量单元

惯性测量单元，即 IMU，是机器人获取当前位姿和运动状态的传感器。惯性测量数据与双目视觉图像进行紧耦合优化，能显著提高系统的鲁棒性[1]；与激光雷达点云数据的融合能有效提高对物体的检测能力[2]。

本人在武术擂台机器人的设计中，发现短时间内使用六轴传感器能保持一定精度，当机器人长时间运行时，其 Z 轴误差较大。这是因为在六轴传感器中，Z 轴角度是由角速度积分计算得到的，存在累计误差，当机器人运行时间过长时，累计误差会产生一定影响。九轴传感器相较于六轴传感器多了三轴的磁场数据，其角度信息与磁强计的磁场数据进行融合，能保持一定的稳定性。但九轴传感器容易受到外部磁场影响，除地理位置的特殊磁场，机器人的电机、电磁手指、扬声器等都会对九轴传感器产生影响，所以在机器人的结构设计时要注意减小机器人其他系统的磁场干扰。



图 2-1 惯性测量单元-JY901

**Fig.2-1 Inertial measurement unit-JY901**

本机器人的惯性测量单元选取维特智能 JY901。JY901 内置 MPU9250 芯片，

能读取三轴加速度、三轴角速度、三轴磁场等原始数据，在 JY901 模块内还内置一个 Cortex-M0 内核处理器，使用高动态卡尔曼滤波融合算法对 MPU9250 的原始数据进行预处理，可通过串口或 IIC 输出更加平滑稳定的数据。JY901 还可链接符合 NMEA-0183 的 GPS 模块，组成惯性组合导航单元。由于 GPS 卫星信号穿透能力差，在室内环境中不适用，所以本机器人的设计过程不包含 GPS 数据的解算，但在电路设计时为 GPS 数据的读取留出了接口，便于机器人功能的扩展。

2.4.2 激光雷达

激光雷达是机器人建图与导航实现的核心传感器，2D 的激光雷达能够获取统一平面内的景深信息，得到二维点云图。本人在大学四年的学习过程中共使用过三款激光雷达，分别为：思岚 A2、思岚 S1 和杉川 Delta 3A，现将其参数列举如下：

表 2-1 激光雷达参数对比			
Tab.2-1 Comparison of lidar parameters			
	思岚 A2	思岚 S1	杉川 Delta 3A
测量半径	0.2m~16m	白色物体：40m 黑色物体：10m	0.25m~16m (反射率 80%)
采样频率	8000Hz	9200Hz	8000Hz
扫描频率	5Hz~15Hz 可调	8~15Hz 可调	5~15Hz 可调
角度分辨率	0.9°	0.313°~0.587° (取决于扫描频率)	0.25°~0.7°
测距分辨率	≤实际距离的 1% (测距≤12m)	3cm	1mm

	≤实际距离的 2% (测距 12m ~ 16m)		
测量精度	实际距离的 1% (≤3 m) 实际距离的 2% (3-5 m) 实际距离的 2.5% (5-16m)	±5cm	<1% (16m)

以上激光雷达参数上思岚 S1 要更胜一筹，在实际使用时思岚 S1 的综合表现也要比其余两个好一点，但本机器人的设计目的为室内环境下的巡检与协作机器人，在本机器人的实际使用时无法发挥出该款激光雷达的突出特点，如 40m 测量半径，较高测量精度等。所以综合考虑价格与实用性能选择杉川雷达的 Delta-3A。

### 2.4.3 深度相机

视觉 SLAM 是通过单目摄像头、双目摄像头、鱼眼摄像头、RGBD 摄像头等作为传感器，将传感器数据处理后传入 SLAM 算法。我们常用的单目摄像头可以通过几帧图像间的信息对比得到画面内的深度信息；双目摄像头的原理与我们人眼的测距原理一致，但是不能看到黑暗处的深度信息，属于被动测量；RGBD 相机能主动发射红外光，根据反射回的反射光得到深度信息，属于主动测量。所以本机器人选用 RGBD 相机。

本人共使用过三款不同的 RGBD 相机，分别为：Kinect V1、Kinect V2 和 Real Sense D435。

Kinect 相机由微软公司生产，其中 KinectV1 是基于结构光原理，KinectV2 是基于 TOF 原理。后者相较于前者有更宽阔的视场角，生成的深度图质量较高。

---

Kinect 的软件工具包里的部分工具性能较强，如人体骨架追踪，能同时进行六个人的骨架追踪。但是在本人的实际使用中，Kinect 相机对 Ubuntu 系统兼容性较差，特别是 KinectV2，在 Ubuntu 操作系统中需要借助 OpenNI2 和 libfreenect2 功能包，其功能有一定限制。

Real SenseD435 是由英特尔公司生产的深度相机，这款相机在 2018 年推出，支持 Windows、Linux、Mac OS 等平台。与 Kinect 相机相比，Real Sense D435 相机性能更强，配置更新。在实际使用中，Real Sense 实现的功能也比较丰富，廖萱等人[3]使用 RealSense 相机实现了基于 RGB-D 传感器的机器人视觉里程计；周雄才[4]实现了一种基于 Real Sense 深度信息的障碍物检测和避障方法；W. Ma 等人[5]提出了一种基于 Real Sense D435 的三维物体点云快速重建方法。

综上，本机器人的深度相机使用 Real Sense D435 深度相机。

#### **2.4.4 雷达与测距模块**

#### **2.4.5 电源电路设计**

### **2.5 本章小结**

---

## 3 Autobot 机器人的软件设计

### 3.1 机器人下位机软件设计

#### 3.1.1 基于 FreeRTOS 操作系统的程序框架设计

在第 2 章内, 经过对比与分析, 选定了单片机 STM32F407VGT6 作为下位机主控。stm32 单片机的常规程序结构为通过定时器中断或外部中断等修改标志位, 主循环内通过标志位的修改执行不同的计算任务。这种程序结构实时性较差, 中断的回调函数内容容易阻塞, 通过人工指定标志位实现的轮询调度系统能在一定程度上缓解程序阻塞, 但整体效率有限。

嵌入式实时操作系统是能运行在单片机中的轻量操作系统, 一般的嵌入式实时操作系统都具有低功耗、实时性强、低成本等优势。FreeRTOS 系统是一款开源免费的实时操作系统, 具有较强的实时特性与可靠性, 对比自定义的轮询调度系统在效率上有极大的提高。通过 FreeRTOS 的封装, 对任务调度、内存管理、中断管理等程序的执行效率有明显的提高。通过 ST 公司开发的 STM32CubeMX 能更便捷的配置基于 HAL 库的 FreeRTOS 系统。

在上一节说明了机器人下位机需要实现的功能, 根据以上功能, 在 FreeRTOS 系统中定义程序框架图如下:

---

### 3.1.2 电机控制程序设计

### 3.1.3 基于 DMA 的 ADC 数据读取程序设计

### 3.1.4 基于 VCP 的模拟串口通讯设计

### 3.1.5 基于 roserial 的下位机 ROS 通讯设计

## 3.2 机器人上位机软件设计

### 3.2.1 基于 roserial 的上位机 ROS 通讯设计

### 3.2.2 机器人实体模型的 TF 分析

### 3.2.3 基于 RIVZ 的机器人传感器数据显示

### 3.2.4 基于话题通讯的机器控制程序设计

## 3.3 本章小结

---

## 4 人机交互系统设计

### 4.1 基于 PyQt5 的控制软件设计

#### 4.1.1 控制界面功能总览

#### 4.1.2 基于 matplotlib 的传感器数据图形化显示设计

### 4.2 基于 stm32 的网络遥控设计

### 4.3 遥控设计概述

#### 4.3.1 OLED 的 GUI 界面设计

#### 4.3.2 局域网内 stm32 与 ROS 系统的通信系统设计

### 4.4 基于 Web 的网页控制界面设计

### 4.5 本章小结



---

## 5 SLAM 算法分析与实践

### 5.1 SLAM 算法概述

### 5.2 激光 SLAM

#### 5.2.1 Gmapping

#### 5.2.2 Cartographer

### 5.3 视觉 SLAM

---

## 6 人体姿态估计系统设计

### 6.1 人体姿态估计概述

### 6.2 OpenPose、AlphaPose、Adafuse 算法对比

#### 6.2.1 OpenPose 原理与实验

#### 6.2.2 AlphaPose 原理与实验

#### 6.2.3 Adafuse 原理与实验

#### 6.2.4 对比分析

### 6.3 Autobot 机器人多人人体姿态估计实验

---

## 7 总结与展望

---

## 参考文献

- [1] 任金伟, 郑鑫, 李昱辰和朱建科, 《基于新型多传感器融合策略的移动端双目视觉惯性 SLAM 闭环算法研究》, 高技术通讯, 卷 31, 期 07, 页 681–691, 2021.
- [2] 张艳国和李擎, 《基于惯性测量单元的激光雷达点云融合方法》, 系统仿真学报, 卷 30, 期 11, 页 4334–4339, 2018, doi: 10.16182/j.issn1004731x.joss.201811034.
- [3] 廖萱, 陈锐志和李明, 《基于 Realsense 传感器的机器人视觉里程计研究》, 地理空间信息, 卷 18, 期 02, 页 1-4+6, 2020.
- [4] 周雄才, 《基于 RealSense 的深度信息的障碍物检测与避障方法》, 硕士, 重庆邮电大学, 2019. [在线]. 载于: <https://kns.cnki.net/KCMS/detail/detail.aspx?dbcode=CMFD&dbname=CMFD202001&filename=1019643556.nh&v=>
- [5] W. Ma 和 K. Yang, 《A fast Reconstruction Method of 3D Object Point Cloud Based on Realsense D435》, 收入 *2020 39th Chinese Control Conference (CCC)*, Shenyang, China, 7 月 2020, 页 6650–6656. doi: 10.23919/CCC50068.2020.9188665.