



科研课堂学习指南

作者：黄靖宜、万超凡、火振权、陈旭文、王越、韩亮

组织：北京航空航天大学 中法工程师学院

时间：2023 年 2 月

联系方式：暂无



君子博学而日参省乎己，则知明而行无过矣。——荀子

目录

第 1 章 课前准备工作	2
1.1 软件准备	2
1.1.1 Python 环境准备	2
1.1.2 Python 集成开发环境准备	3
1.1.3 安装 robomaster sdk	4
1.1.4 Potato-mini 仿真平台	5
1.1.5 UWB 相关软件准备	6
1.2 仿真	7
1.3 硬件准备	7
1.3.1 与机器人建立连接	8
1.3.1.1 与无人机建立连接	8
1.3.1.2 与无人车建立连接	8

前言

科研课堂是北京航空航天大学中法工程师学院韩亮副教授开设的一门课程，希望能为热爱工程、愿意提升工程和代码能力的同学提供有力的支持。本课程一共持续八周，将使用实验室自研的大规模集群平台进行仿真，并使用大疆的 RoboMaster EP 和 TT 等平台进行开发和算法实现。最后我们将设计一项十分具有挑战性的任务，同学们通过仿真和实物平台进行竞赛，验证自己的学习成果。

希望同学们能在科研课堂中收获友谊和不错的实践能力。我相信同学们能够顺利完美地完成本课程！

第1章 课前准备工作

本次科研课堂将涉及到部分机器人规划控制的内容，需要利用 Python 对机器人进行开发，同时仿真也需要通过 Python 进行环境搭建和算法验证。为了能够顺利完成本课程，同学们在开始进行仿真和实物实验之前需要进行一定的准备工作。

1.1 软件准备

本小节将介绍课程需要使用的相关软件。

1.1.1 Python 环境准备

本课程建议使用 Anaconda 创建一个独立的 Python 环境。

Anaconda 是一个开源的 Python 和 R 语言的发行版本，用于计算科学（数据科学、机器学习、大数据处理和预测分析），Anaconda 致力于简化软件包管理系统和部署。Anaconda 的包使用软件包管理系统 Conda 进行管理。超过 1200 万人使用 Anaconda 发行版本，并且 Anaconda 拥有超过 1400 个适用于 Windows、Linux 和 MacOS 的数据科学软件包。

虚拟环境是一种隔离的 Python 环境，它允许用户在不同的项目中使用不同版本的库和工具，而不会相互干扰。这意味着你可以在一个虚拟环境中安装特定版本的库，并在另一个虚拟环境中安装不同版本的同一库，以便在不同的项目中使用。这种灵活性和隔离性是非常重要的，尤其是当你需要同时开发多个项目或与其他人合作时。

通过 Anaconda 创建虚拟环境非常简单，只需要使用命令行界面输入几行命令即可完成。使用虚拟环境可以帮助您避免因版本不兼容而导致的错误，提高代码的可重用性和可维护性。如果不使用 Anaconda，对本次课程不会有任何影响。但是我们仍然希望同学们在本课程中养成良好的习惯，为以后的开发工作打好基础。

使用 Anaconda 新建一个虚拟环境的步骤如下：

1. 通过<https://www.anaconda.com/>官方网站下载软件(如果无法打开官网,建议通过清华源<https://repo.anaconda.com/archive/>进行下载)。
2. 双击执行文件进行安装，安装过程中，注意安装路径不要有中文。
3. 下载完成后，在 Windows 搜索栏中搜索并打开 Anaconda Powershell Prompt (anaconda)。
4. 输入 `conda create -n keyanketang python=3.7`。

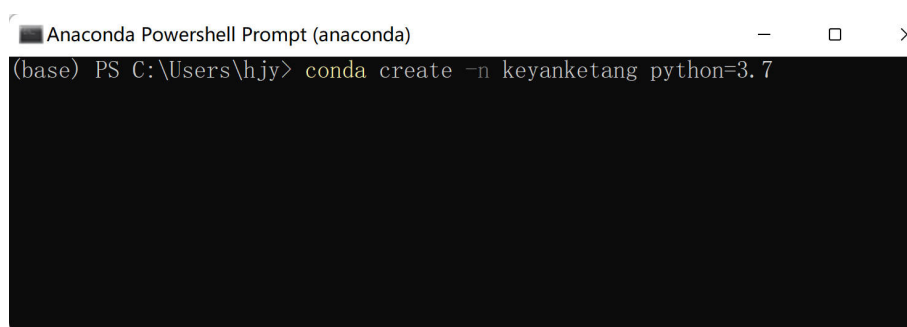


图 1.1: 使用 anaconda 新建虚拟环境

5. 出现如图1.2所示情况，输入 y（不限制大小写）即可完成虚拟环境的创建。
6. 输入 `conda env list` 可以查看自己建立的虚拟环境，如果能够找到 `keyanketang`，说明创建成功。

这里针对上述操作进行一些补充和解释：anaconda 安装时间可能会比较长，同学们耐心等待，如果进度条不动怀疑电脑卡住，可以打开任务管理器查看磁盘是否有写入，有的话说明电脑没有卡住；根据大疆的官方说

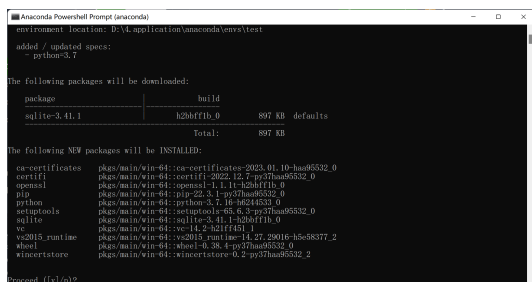


图 1.2: 完成虚拟环境创建

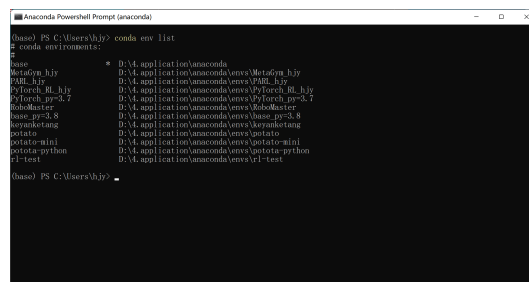


图 1.3: 查看虚拟环境

明手册，本课程全程使用 3.7 版本的 Python；关于 conda 操作的其他细节同学们可以自行上网查询，本课程只需要用到创建虚拟环境。

1.1.2 Python 集成开发环境准备

集成开发环境（Integrated Development Environment, IDE）是一种帮助程序员高效开发软件代码的软件应用程序。它通过将软件编辑、构建、测试和打包等功能结合到一个易于使用的应用程序中，提高了开发人员的工作效率。就像作家使用文本编辑器，会计师使用电子表格一样，软件开发人员使用 IDE 让他们的工作变得更轻松。

本课题组在科研过程中，在需要使用 Python 时，常使用 PyCharm 进行开发和版本控制。PyCharm 是一种 Python IDE，带有一整套可以帮助用户在使用 Python 语言开发时提高其效率的工具，比如调试、语法高亮、项目管理、代码跳转、智能提示、自动完成、单元测试、版本控制。

PyCharm 提供了多种版本：教育版、社区版和专业版。

- 教育版：教育版是免费的，具备社区版的所有功能，除此之外，还提供有一个教学功能，因此它更适合学生。老师可以用它创建教学，学生可以通过他完成教学任务；
- 社区版：就是阉割版的专业版，它也是免费的，如果你并不需要使用专业版才有那些功能，可以选择社区版；
- 专业版：提供所有 PyCharm 的功能，虽然是收费的，但是可以试用一个月。

那么社区版和专业版在功能上有哪些区别呢？同学们可以看下面这个表格。

	PyCharm Professional Edition	PyCharm Community Edition
Intelligent Python editor	✓	✓
Graphical debugger and test runner	✓	✓
Navigation and Refactorings	✓	✓
Code inspections	✓	✓
VCS support	✓	✓
Scientific tools	✓	
Web development	✓	
Python web frameworks	✓	
Python Profiler	✓	
Remote development capabilities	✓	
Database & SQL support	✓	

图 1.4: 不同版本 PyCharm 比较

如果你想使用到专业版，那有什么办法呢？

- 有钱的就是大爷，付费购买；
- 穷人自有穷活法，每次试用一个月，试用期到，卸载干净，再来一次；
- 利用学生与教师的特权，可申请免费使用；
- 若你有开源项目，也可以申请免费使用；
- 用一些非寻常手段（也就是破解）来实现。

有一种邮箱，叫做教育邮箱，这东西在这个互联网的世界有很大的优惠及特权，在 Jetbrain 这里，如果你有教育邮箱 (.edu.cn 后缀的邮箱)，你可以免费使用专业版的 PyCharm。如果没有教育邮箱，你只要能提供能证明你的学生或者老师身份的证明，比如学生证、教师证等，Jetbrain 也可以让你免费一定期限（申请地址：<https://www.jetbrains.com/student/>），学生证有效期是一年，每年都要复审一次，老师的话就是长期的了。

PyCharm 专业版下载步骤如下：

1. 打开<https://www.jetbrains.com/student/>申请学生资格。
2. 在网页上点击 apply 进行学生资格认证，注意用北航邮箱进行申请。
3. 进入<https://www.jetbrains.com/pycharm/download/#section=windows>官网下载专业版 PyCharm。
4. 下载过程中注意勾选 Add "bin" folder to the PATH。

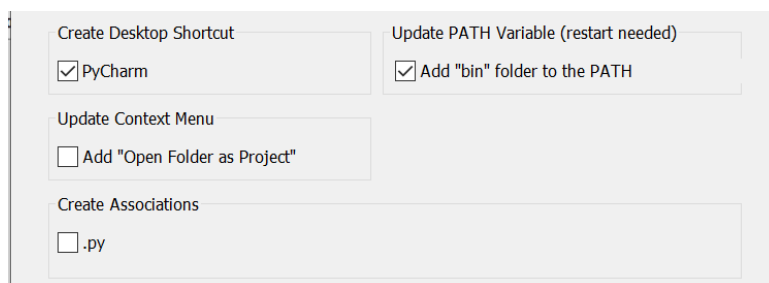


图 1.5: 安装 PyCharm

5. 成功后即可用 jetbrains 账号登录 Pycharm 专业版。

1.1.3 安装 robomaster sdk

如图所示，先点击右下角，Add New Interpreter -> Add Local Interpreter，进行 PyCharm 解释器的配置，在其中找到刚才创建好的新虚拟环境（在此命名是 keyanketang）。

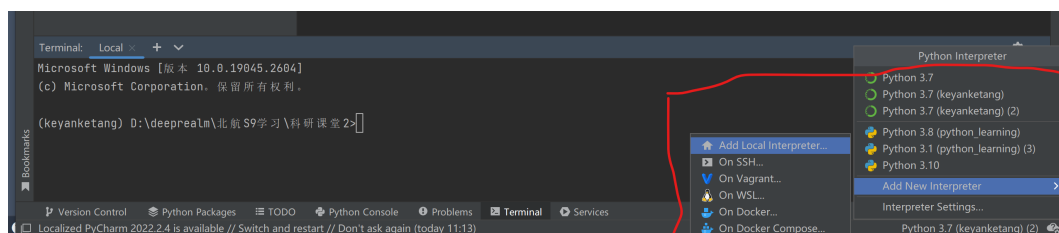


图 1.6: 在 PyCharm 中使用虚拟环境-步骤 1

如果没有自动匹配创建好的新虚拟环境，就要手动寻找路径。如图所示，首先点击 interpreter 栏尾的...，在出现的窗口中选择文件路径.../conda/envs/keyanketang/python.exe，... 是最开始创建虚拟环境所在的路径。

选中后点击 OK，至此已经成功在 PyCharm 中使用了之前配置好的 Python3.7 虚拟环境。

在 PyCharm 中，首先点击 terminal 打开终端，当命令行头部出现虚拟环境名（keyanketang）时，即为成功，接下来，输入 pip install robomaster 完成包的安装。如果打开终端的时候出现报错或者命令行头部未出现虚拟环境名（keyanketang），建议按照一下操作进行设置：

- 打开 settings->terminal，修改 Shell path 为 cmd.exe(如图所示)。

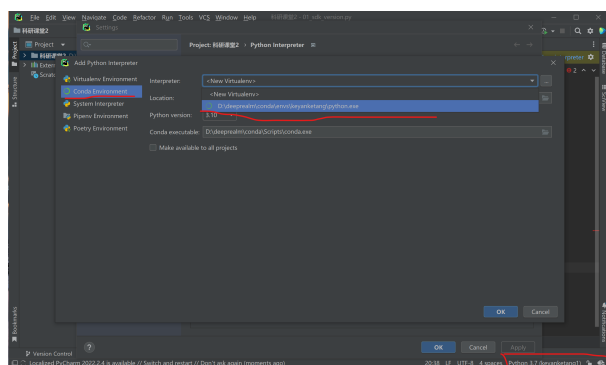


图 1.7: 在 PyCharm 中使用虚拟环境-步骤 2

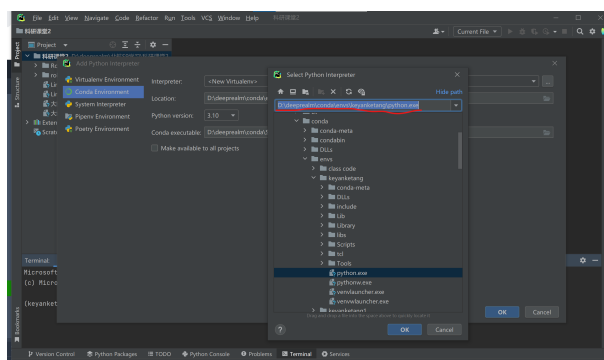


图 1.8: 在 PyCharm 中使用虚拟环境-步骤 3

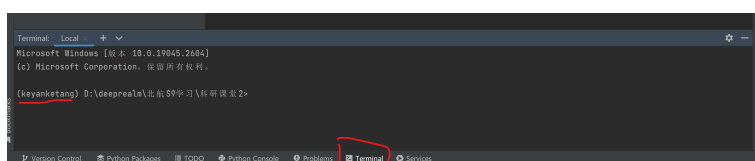


图 1.9: 打开 terminal

注 如果在图1.6中提示“Conda executable is not found”，请参考以下链接解决<https://blog.csdn.net/s1hj/article/details/128759758?spm=1001.2014.3001.5506>。

1.1.4 Potato-mini 仿真平台

Potato 是本题组自研的大规模集群仿真平台，为了照顾到同学们的科研和课程需求，现开发了一个采用仿真平台显示端作为 GUI 的最小示例——potato-mini。同学们可以在 potato-mini 中自行设计开发智能体动力学、添加典型任务环境并进行算法仿真。首先，同学们下载并解压 Potato-2.3.4-win.zip 和 potato-mini-master.zip 文件，分别得到 Potato 的显示端和 Potato-mini 的主体代码。接下来仿真部分的开发主要在 Potato-mini 的主体代码中进行。在 Potato-2.3.4-win 中找到 Potato.exe 并双击，即打开仿真平台显示端。

打开 PyCharm，选择左上角的“文件/file”，点击“打开/open”，打开 potato-mini-master 文件，即可打开 potato-mini 项目。

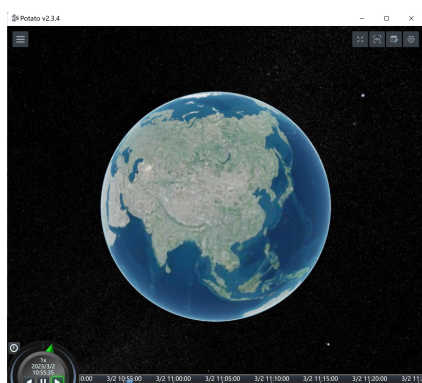


图 1.10: 仿真平台显示端

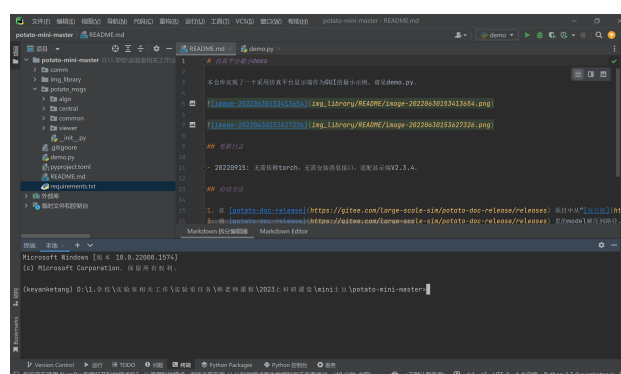


图 1.11: 在 PyCharm 中打开 potato-mini

为了能够正常使用 potato-mini 及其显示端，我们需要安装一些库。所需 Python 库如下所示：

- attrs==21.4.0 ;
- attr==0.3.2;
- numpy==1.21.5;
- orjson==3.8.7;
- torch==1.13.1。

在 PyCharm 中最下方点击 terminal 进入终端，确定好 Anaconda 环境为科研课堂环境后，在环境中安装所需的 Python 库。安装方式为 `pip install 库名 == 库所需的版本`。如安装 numpy 为 `pip install numpy==1.21.5`。

1.1.5 UWB 相关软件准备

超宽带（Ultra Wide Band, UWB）技术是一种无线载波通信技术，它不采用正弦载波，而是利用纳秒级的非正弦波窄脉冲传输数据，因此其所占的频谱范围很宽。UWB 技术具有系统复杂度低，发射信号功率谱密度低，对信道衰落不敏感，截获能力低，定位精度高等优点。我们使用深圳空循环科技有限公司所开发的 LinkTrack 产品对无人机无人车进行定位。LinkTrack 是一款基于 UWB 技术的 PNTC（定位、导航、授时与通信）局部定位系统。支持 1 维、2 维、3 维定位，1 维、2 维典型定位精度 10cm，第 3 维典型定位精度 30cm；最高同时支持 200 个标签，或同时支持 120 个基站定位。支持分布式测距与数传，实现不受地域限制的集群编队；支持纯数传模式，带宽高达 3Mbps。

同学们可以在<https://www.nooploop.com/download/>网站下载 LinkTrack 所需软件（NAssistant 上位机）及其驱动（见图1.20右下角）。



图 1.12: UWB 应用以及驱动

1. 运行如图三个程序，下载相应驱动和 NAssistantAPP

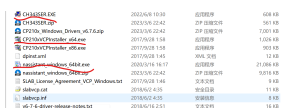


图 1.13: 下载内容

2. 按照场合二位置放置四个基站, 其中 A0A3 代表 x 轴方向, 并且让 A0A1 尽量和 y 轴正方向重合。

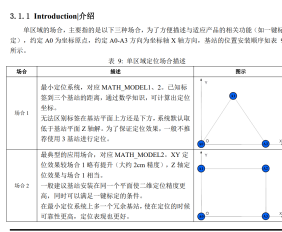


图 1.14: 基站摆放

3. 如图打开 NAssist，依此连接 A0-A3 基站和电脑，如图配置
4. 如图打开 NAssist，连接控制台和电脑，打开 linktrack 并进行一键标定

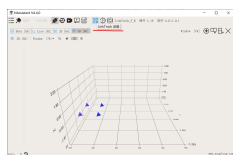


图 1.15: 打开 NAssist

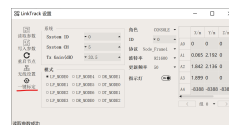


图 1.16: 一键标定

5. 将无人机和 UWB-tag 如图连接 x
6. 在 pycharm 内打开 uwbPos.py，检查 48 行和 54 行的 COM 号和 computerladdr，无误后运行
7. 可能需要安装 pyserial 包：pip install pyserial

1.2 仿真

本节将基于 Potato 仿真平台进行四旋翼无人机的控制仿真,主要使用的文件是 Simulation 文件夹中的 pid_demo.py,该文件是一个反馈控制单个无人机按特定轨迹飞行的示例,其中有四个 TODO 可以点击进行跳转,内容分别为:PID 控制器类的定义、期望轨迹的设置、PID 控制器参数的设置、总仿真时长。部分代码需同学们补充。

具体使用步骤如下

1. 打开 pycharm, 进入代码文件目录, 打开 pid_demo.py
2. 在第一个 TODO 处定义 PID 控制器的更新过程
3. 在第二个 TODO 处定义期望轨迹, 默认期望轨迹是一个圆形
4. 在第三个 TODO 处设定 PID 控制器的参数
5. 在第四个 TODO 处设定仿真总时长, 默认为 30 秒
6. 双击 Potato, 打开仿真平台显示端
7. 运行 pid_demo.py, 观察结果并调参

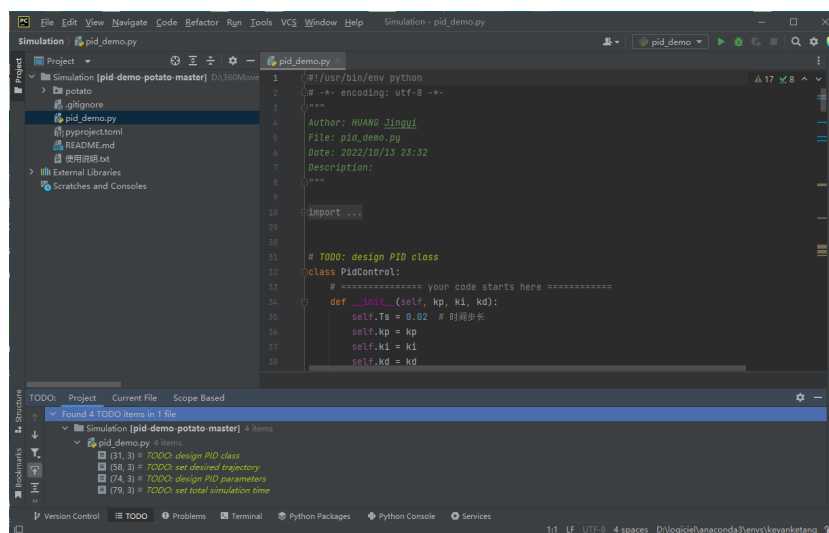


图 1.17: potato 仿真代码

1.3 硬件准备

请先打开 pycharm, 进入上次课的目录, 下载群里发的代码并放在里面, 切换环境到 keyanketang。

1.3.1 与机器人建立连接

请参考网页https://robomaster-dev.readthedocs.io/zh_CN/latest/python_sdk/connection.html。

1.3.1.1 与无人机建立连接

首先完成电脑与无人机的直连

1. 将无人机开机，将 LED 屏侧方开关向下扳，切换至直连模式
2. 在无人机背后查看以“RMTT”开头的 wifi 编号，在电脑上连接该 wifi
3. 启动 cmd，输入 ipconfig，查看自己的 ip 的地址
4. 进入程序 01_ap_conn.py，robomaster.config.LOCAL_IP_STR 设置为本机 ip
5. 按住 Ctrl 点击 config 设置 DEFAULT_CONN_TYPE= “ap”
6. 运行 01_ap_conn.py，显示”Drone SDK Version: 30” 则说明直连没问题

运行 05_takeoff_land.py，无人机正常会起飞而后降落。

接下来使无人机进入实验室的 wifi 组网，需要在直连模式下把 wifi 的账号密码告诉无人机，具体步骤如下

1. 进入 23_set_sat.py 程序，设置 wifi 的名称和密码，运行程序
2. 将 LED 屏侧方开关向上扳切换无人机至组网模式，等待无人机螺旋桨转动即说明进入了组网模式
3. 电脑连接 uavedu，进入路由器后台查看本机和无人机的 ip 地址
4. 进入程序 01_ap_conn.py,robomaster.config.LOCAL_IP_STR 设置为本机 ip,robomaster.config.ROBOT_IP_STR 设置为无人机 ip
5. 按住 Ctrl 点击 config 设置 DEFAULT_CONN_TYPE= “sta”
6. 运行 01_ap_conn.py，显示”Drone SDK Version: 30” 则说明组网没问题

1.3.1.2 与无人车建立连接

接下来与无人车建立联系

1. 长按电池给车开机，将 carconnect 放入创建的项目文件夹



图 1.18: 车开机和准备连接程序

2. 将电脑 wifi 切换至 uavedu，运行 carconnect 程序，出现二维码，放大
3. 切换至组网模式，按红色按钮，按车语音提示扫描二维码



图 1.19: 扫描二维码

4. 控制台出现 connected 即为成功

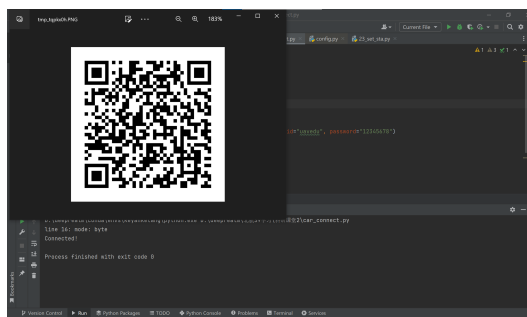


图 1.20: 连接成功