

ROS机械臂开发：从入门到实战

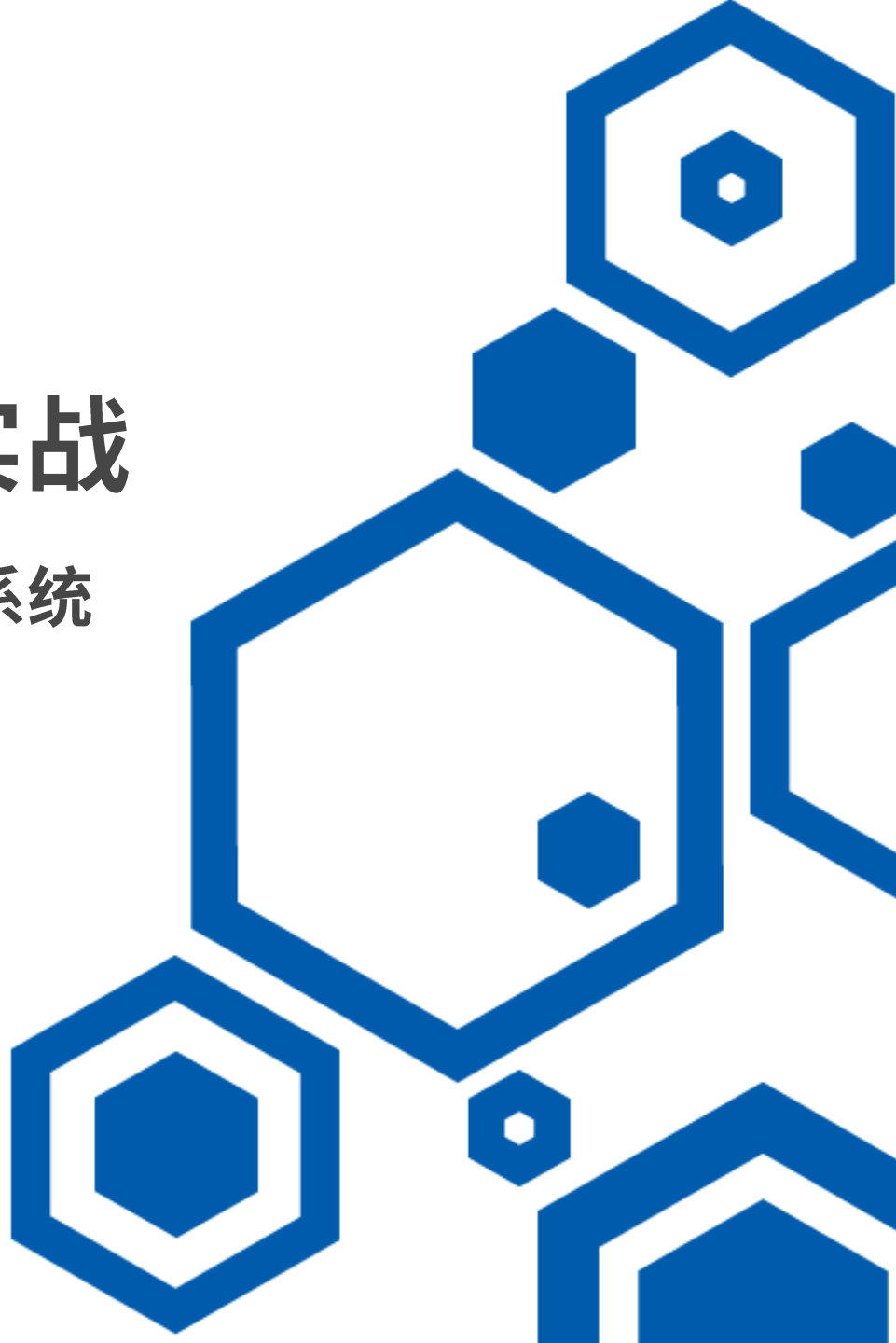
—— 第11讲：基于ROS设计一款机械臂控制系统



主讲人 胡春旭



机器人博客“古月居”博主
《ROS机器人开发实践》作者
武汉精锋微控科技有限公司 联合创始人
华中科技大学 自动化学院 硕士



1. ROS控制系统设计方法

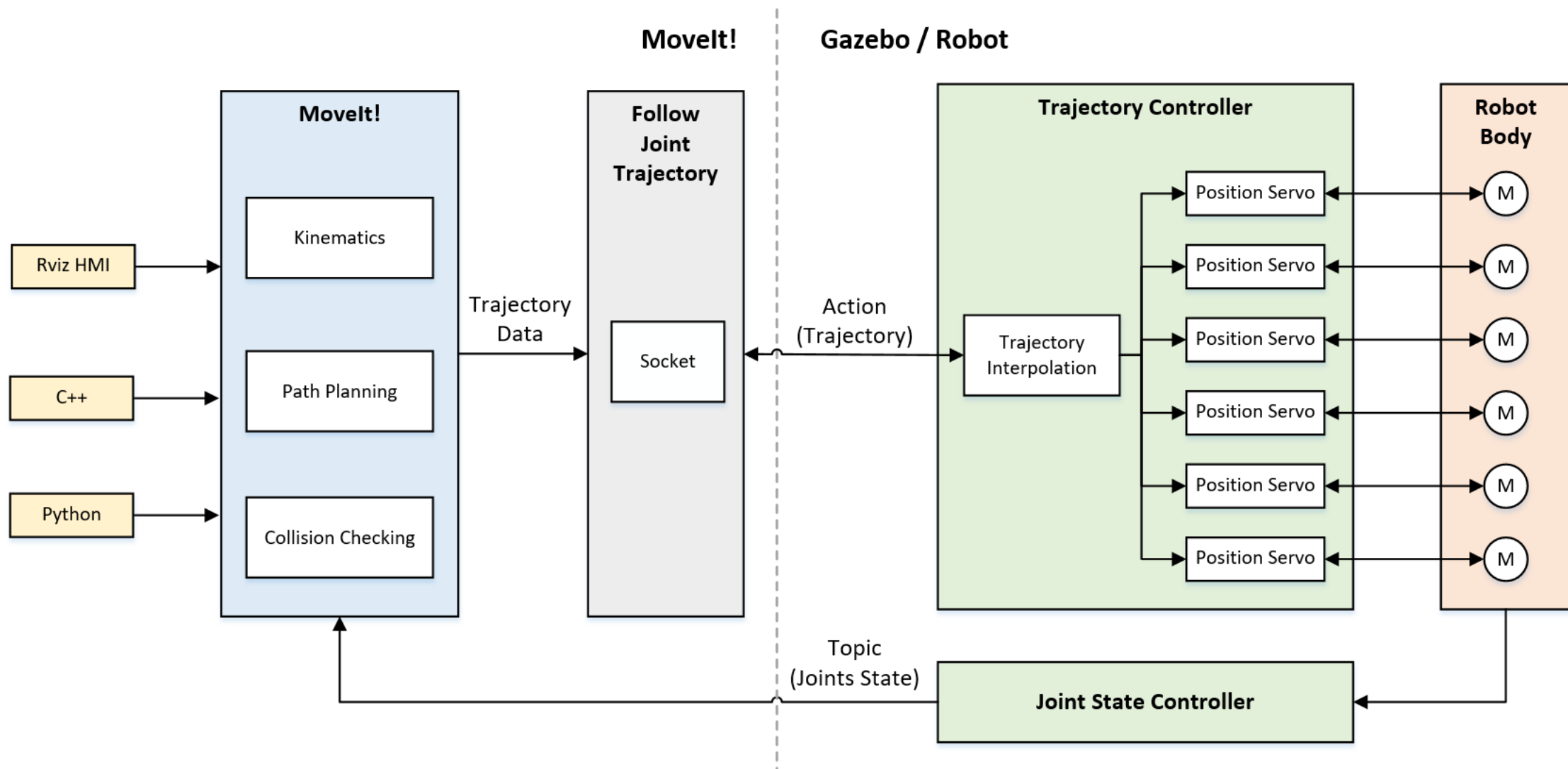
2. PROBOT Anno控制系统案例分析



1. ROS控制系统设计方法



1. ROS控制系统设计方法



ROS机械臂控制框架



1. ROS控制系统设计方法

➤ 控制器管理器

提供一种通用的接口来管理不同的控制器。

➤ 控制器

读取硬件状态，发布控制命令，完成每个 joint 的控制。

➤ 硬件资源

为上下两层提供硬件资源的接口。

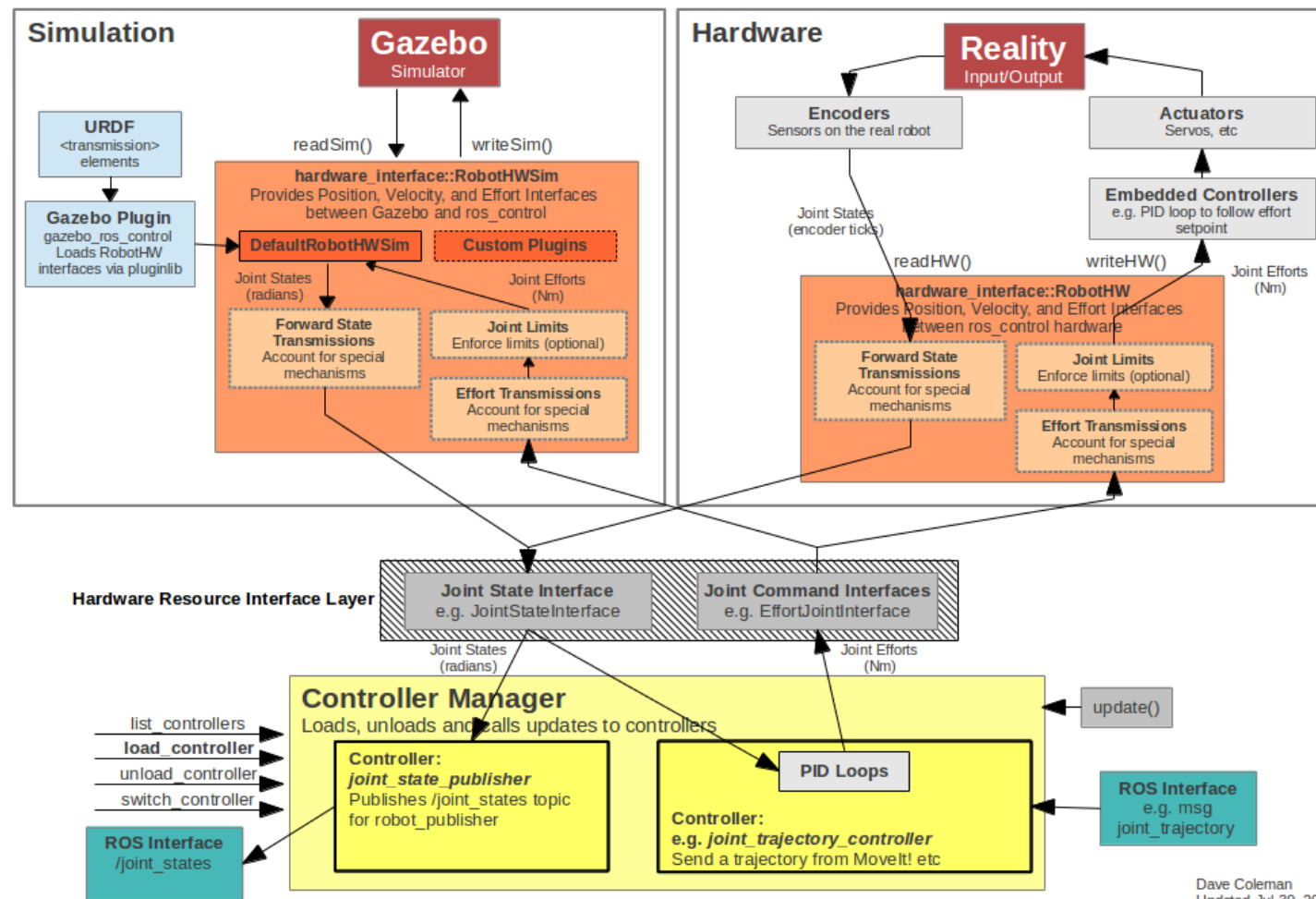
➤ 机器人硬件抽象

机器人硬件抽象和硬件资源直接打交道，通过 write 和 read 方法完成硬件操作。

➤ 真实机器人

执行接收到的命令。

GAZEBO + ROS + ros_control



Dave Coleman
Updated Jul 30, 2013

http://wiki.ros.org/ros_control



1. ROS控制系统设计方法

```
#include <ros/ros.h>
#include <my_robot/my_robot.h>
#include <controller_manager/controller_manager.h>

int main(int argc, char **argv)
{
    // Setup
    ros::init(argc, argv, "my_robot");

    MyRobot::MyRobot robot;
    controller_manager::ControllerManager cm(&robot);

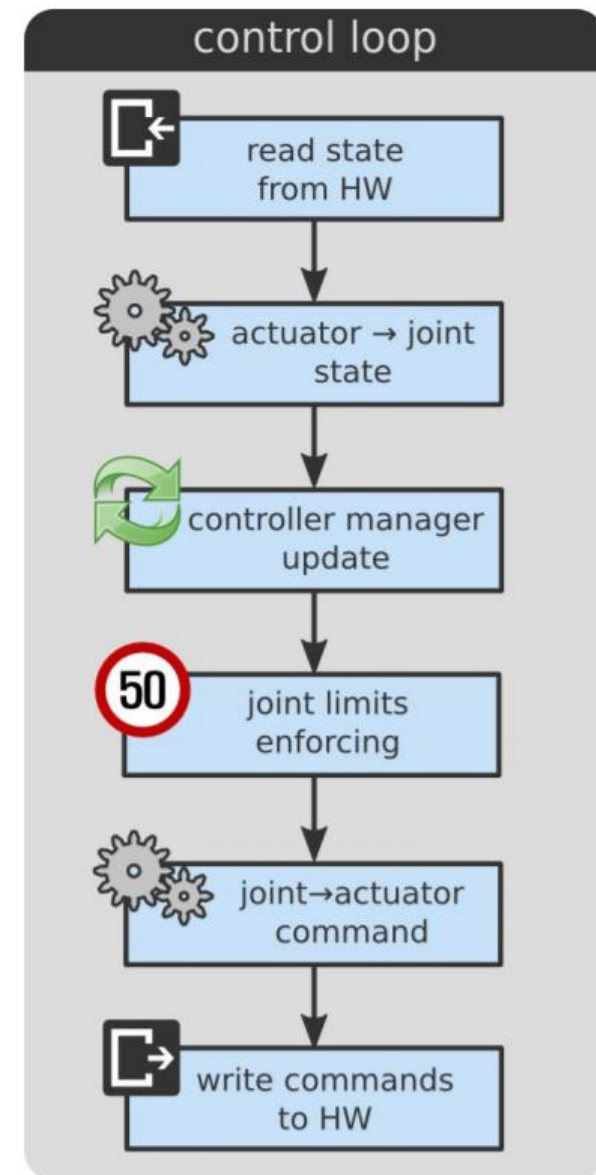
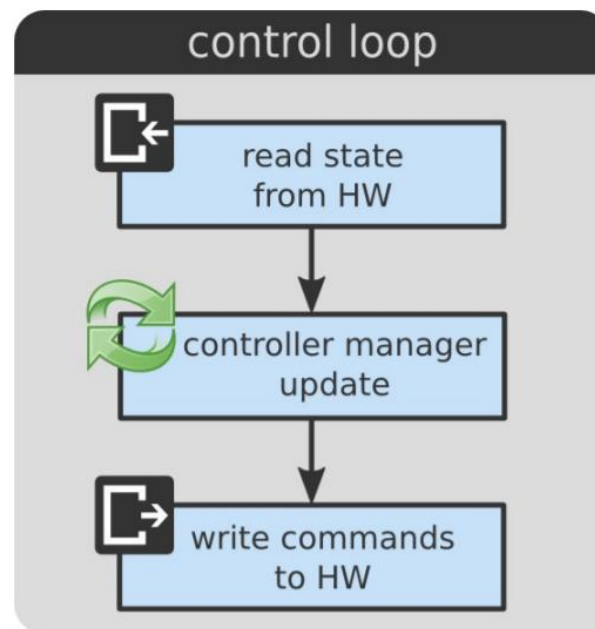
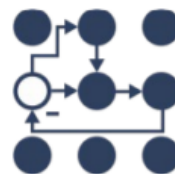
    ros::AsyncSpinner spinner(1);
    spinner.start();

    // Control loop
    ros::Time prev_time = ros::Time::now();
    ros::Rate rate(10.0);

    while (ros::ok())
    {
        const ros::Time time = ros::Time::now();
        const ros::Duration period = time - prev_time;

        robot.read();
        cm.update(time, period);
        robot.write();

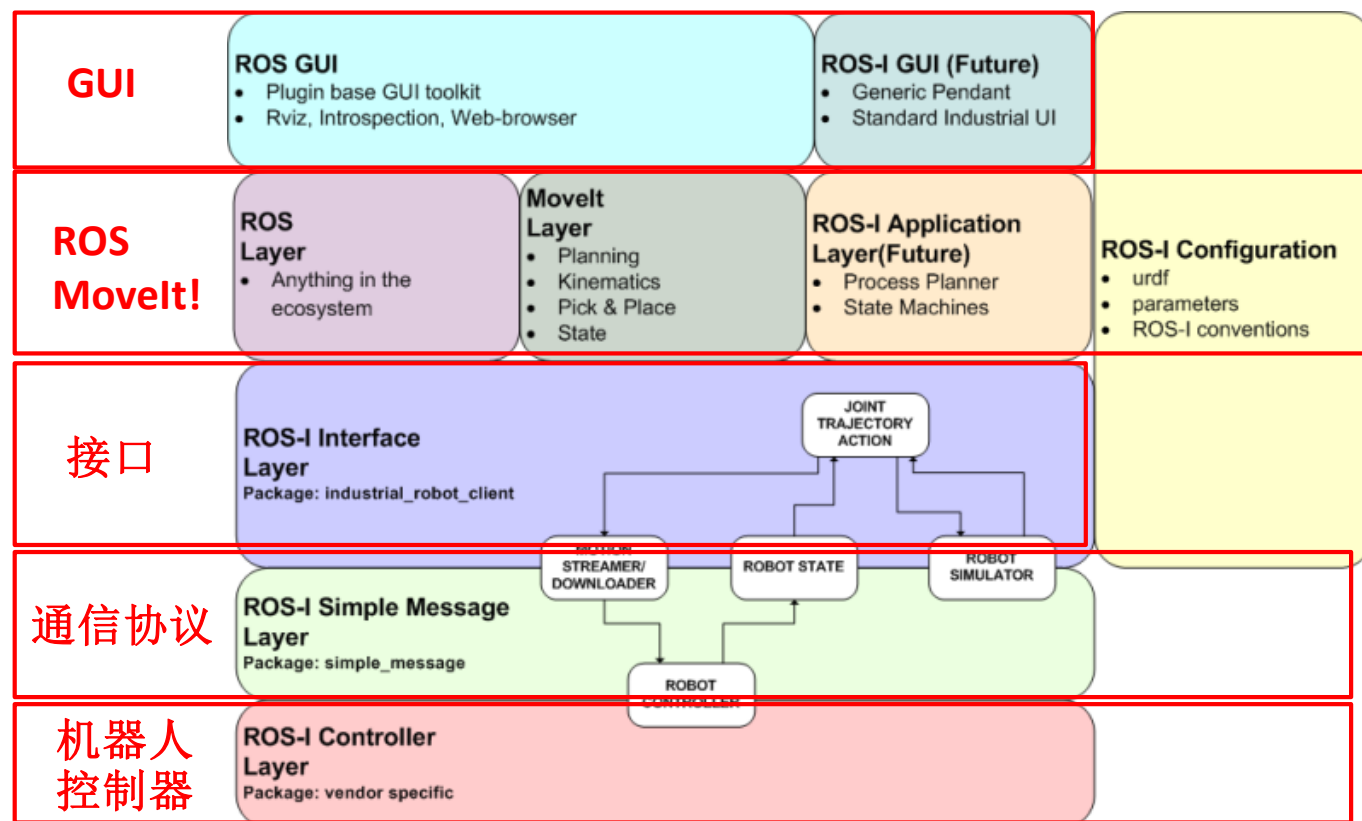
        rate.sleep();
    }
    return 0;
}
```





1. ROS控制系统设计方法

1. GUI: ROS中现在已有的UI工具和专门针对工业机器人通用的UI工具（规划中）；
2. ROS Layer: ROS基础框架，提供核心通信机制；
3. MoveIt! Layer: 为工业机器人提供规划、运动学等核心功能的解决方案；
4. ROS-I Application Layer: 处理工业生产的具体应用，目前也是针对未来的规划；
5. ROS-I Interface Layer: 接口层，工业机器人的客户端，通过 simple message协议与机器人的控制器通信；
6. ROS-I Simple Message Layer: 通信层，定义了通信的协议，打包和解析通信数据；
7. ROS-I Controller Layer: 机器人厂商开发的工业机器人控制器。



ROS-I的总体架构



2. PROBOT Anno控制系统案例分析



2. PROBOT Anno控制系统案例分析



PROBOT Anno —— 桌面机械臂至臻之选

PROBOT Anno是一款桌面级、轻量型、可编程机器人，简单易用，配套教学课程，能帮助您提高研究与学习的效率。

有效工作半径

482.3 mm

重复定位精度

0.5 mm

最大末端速度

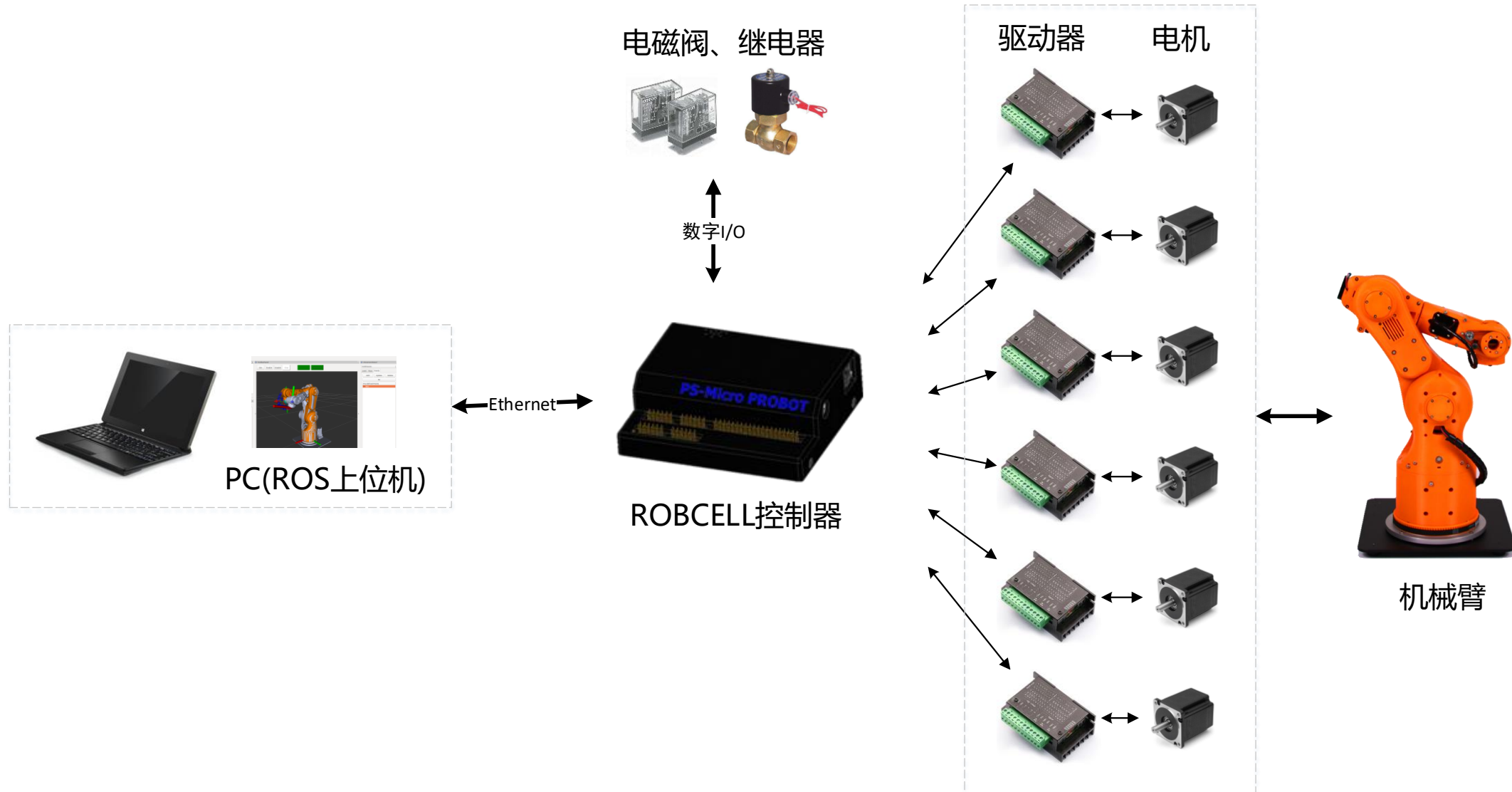
500 mm/s

额定负载

1 kg



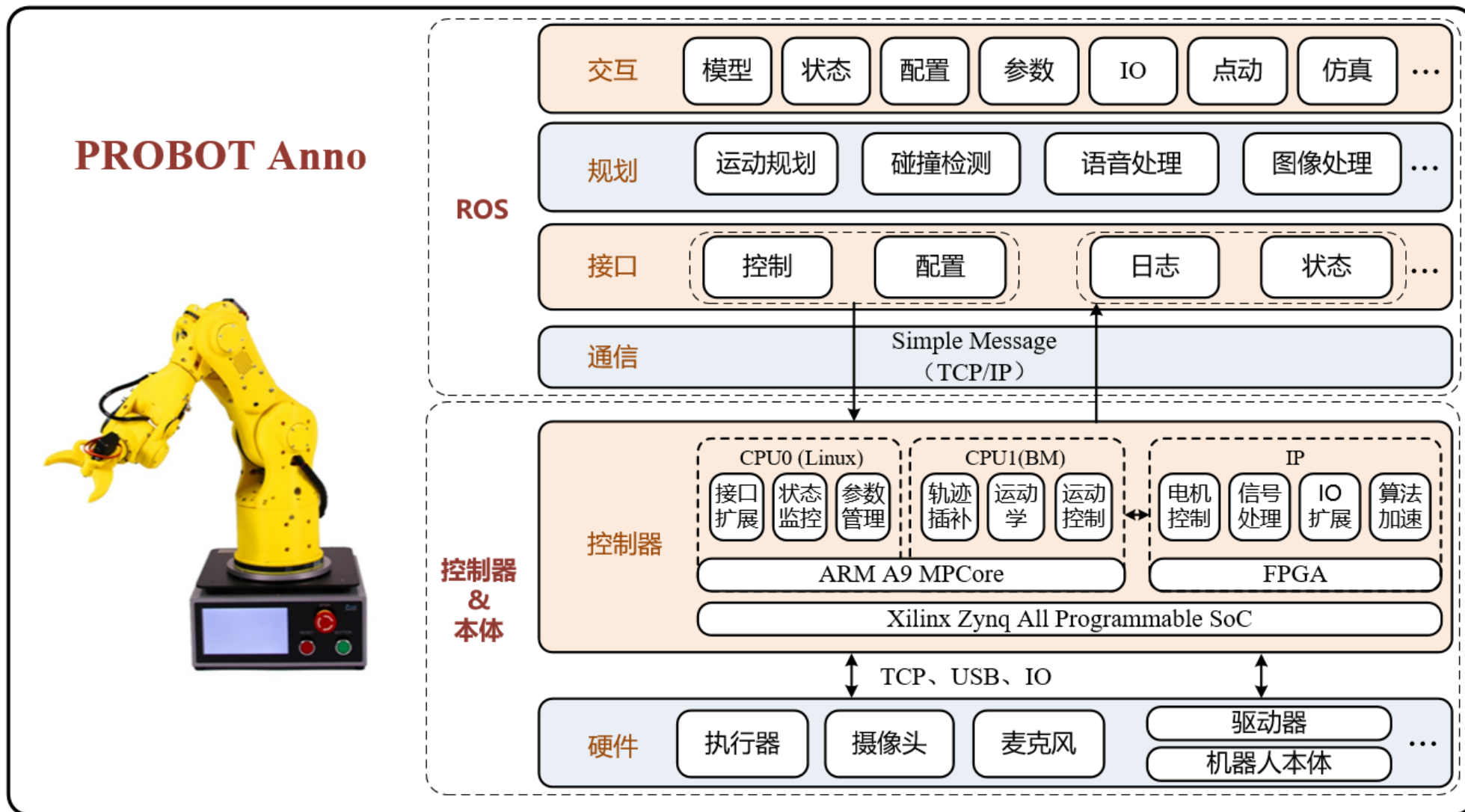
2. PROBOT Anno控制系统案例分析



PROBOT Anno机械臂硬件架构



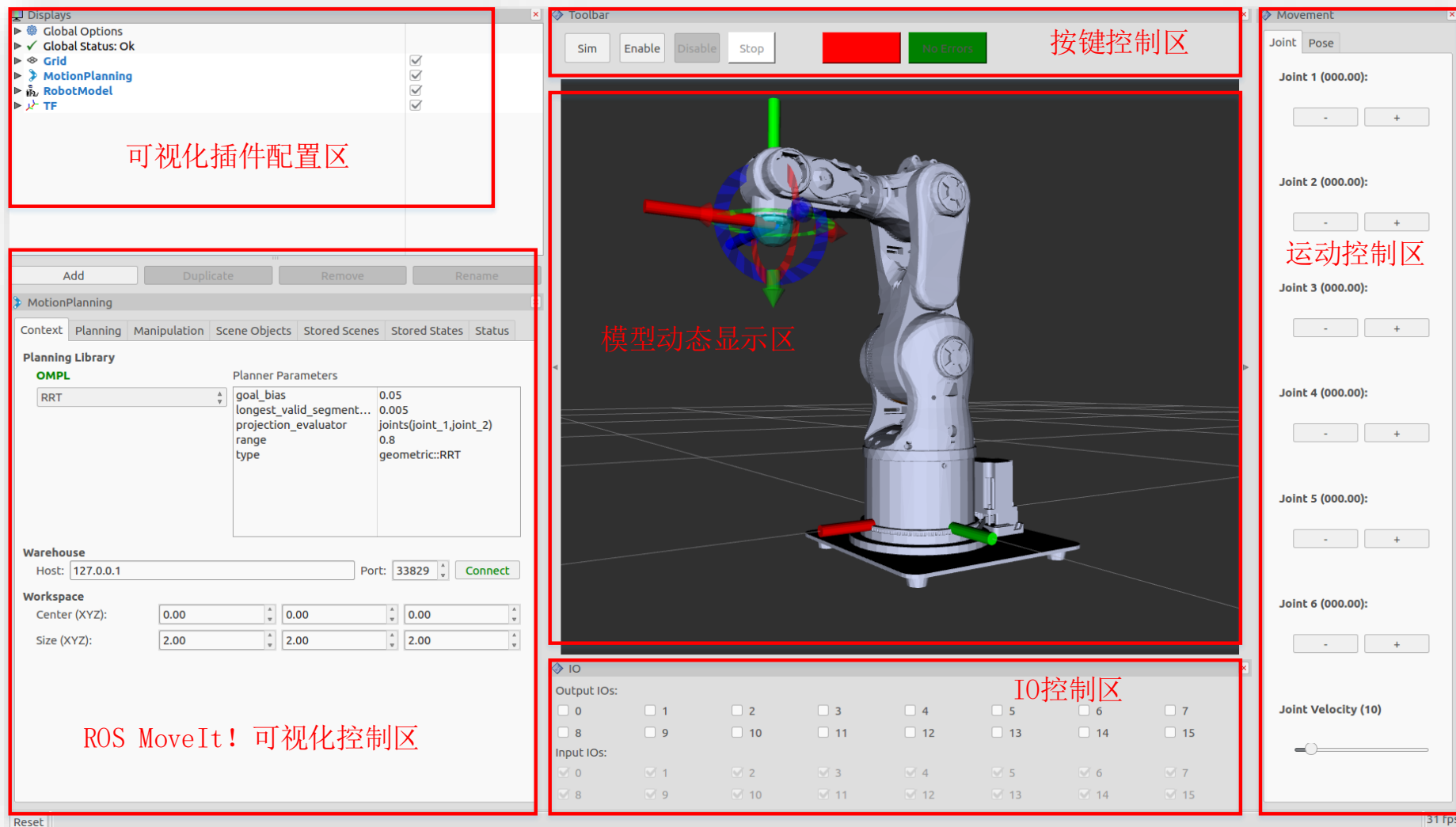
2. PROBOT Anno控制系统案例分析



PROBOT Anno机械臂软件架构



2. PROBOT Anno控制系统案例分析



交互

- 模型显示
- 点动控制
- 状态监控
- 工具插件
- 动态配置
- 在线/离线仿真



2. PROBOT Anno控制系统案例分析

规划



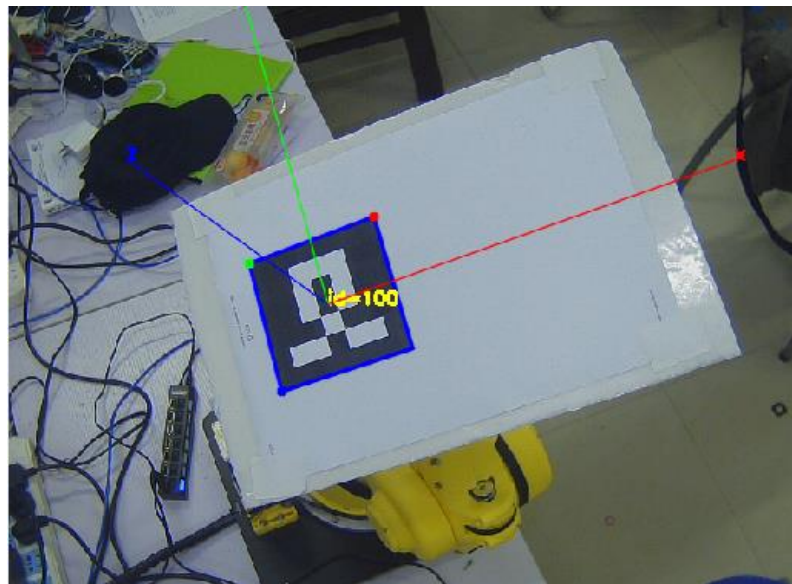
讯飞开放平台

OMPL

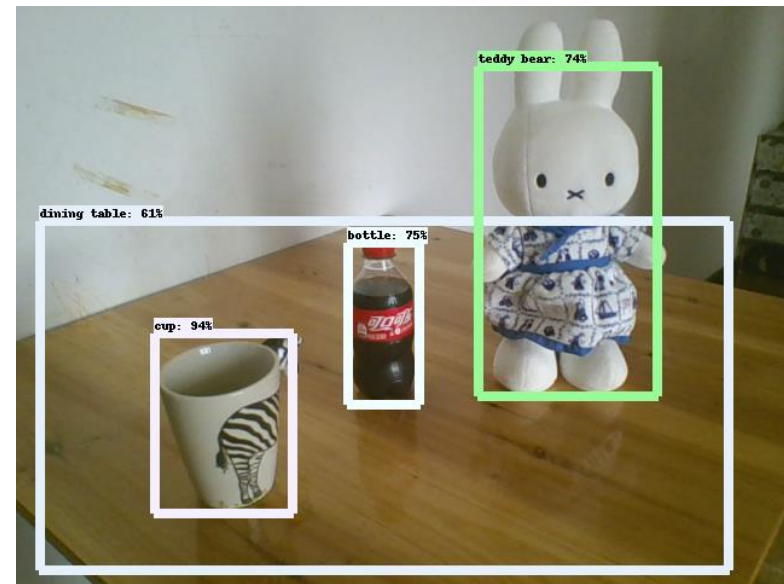
The Open Motion Planning Library



OpenRAVE



手眼标定



机器学习与图像识别



2. PROBOT Anno控制系统案例分析

接口

控制	配置	日志	状态
<ul style="list-style-type: none">● Jogging● Servo Power On● Enable● Start/Pause/stop...	<ul style="list-style-type: none">● Tool Transform● Log Level● Robot Parameters● Servo Parameters...	<ul style="list-style-type: none">● Log_Force● Log_Error● Log_Warn● Log_Info● Log_Debug	<ul style="list-style-type: none">● mode● e_stopped● drives_powered● motion_possible● in_motion● in_error● error_code



通信

数据段 (Simple Message)

前缀

- int LENGTH : 数据头+数据内容的字节长度

数据头

- int MSG_TYPE: 消息类型 (自主定义)
- int COMM_TYPE: 通信类型 (话题/服务)
- int REPLY CODE: 应答代码 (仅服务有效)

数据内容

- ByteArray DATA: 由消息类型和通信类型确定的数据内容

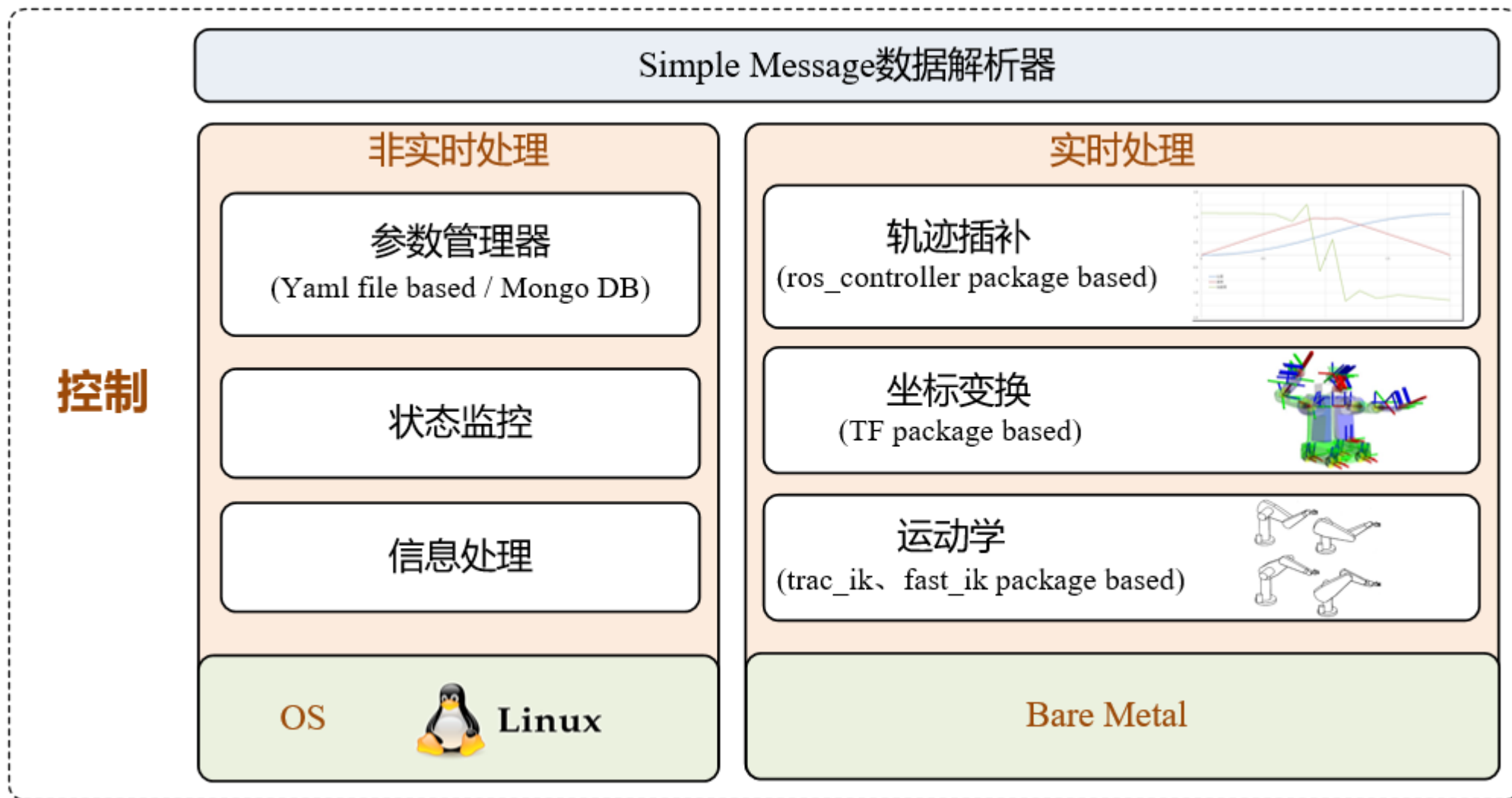
TCP / UDP

Wifi / Ethernet / Localhost



2. PROBOT Anno控制系统案例分析

控制器





2. PROBOT Anno控制系统案例分析

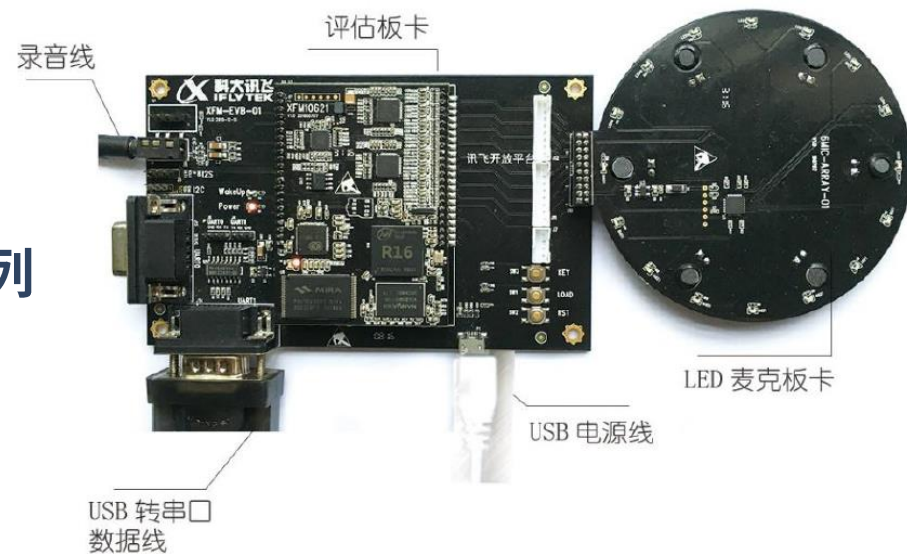


机器人本体



高清摄像头

麦克风阵列



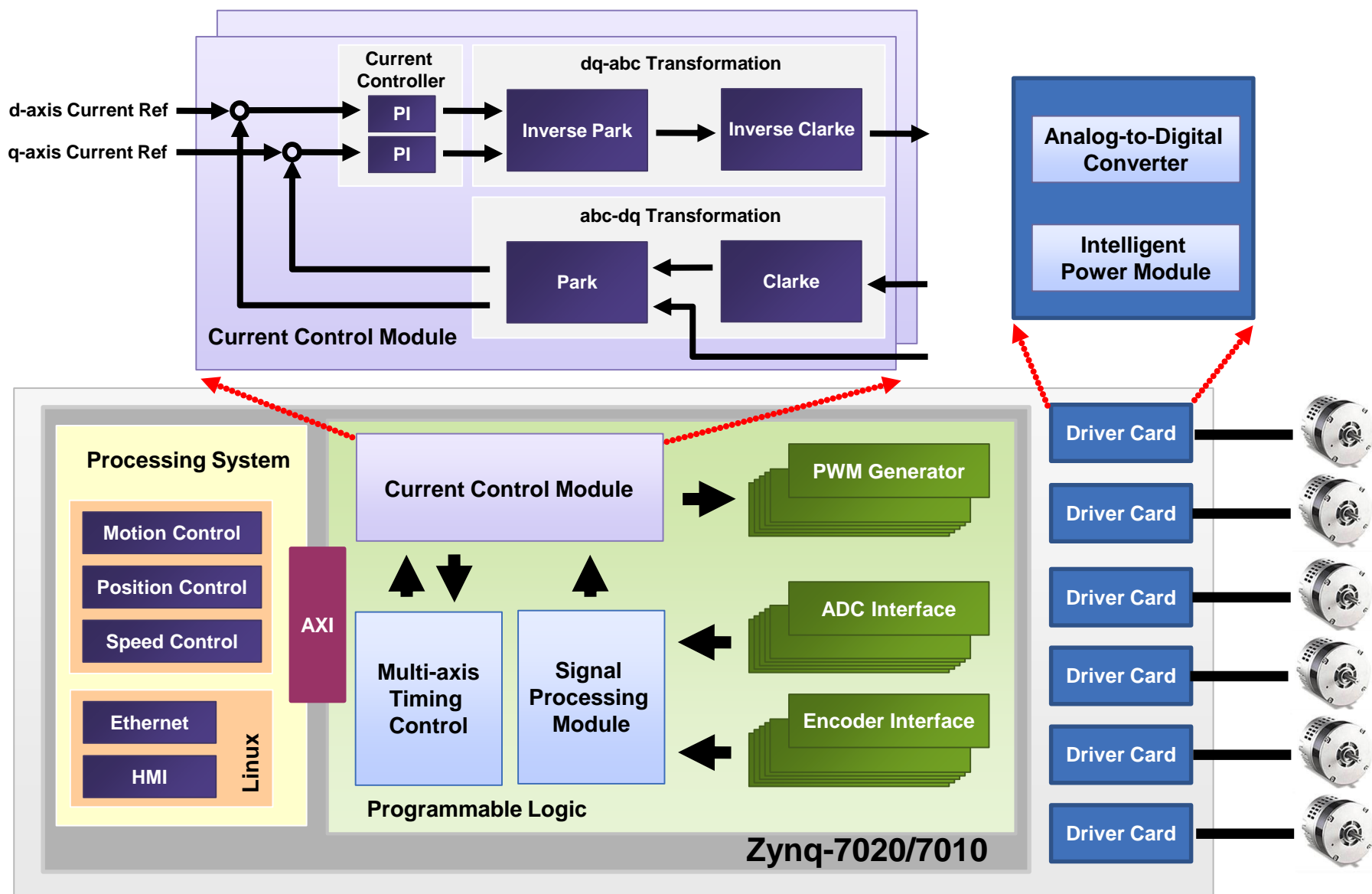


2. PROBOT Anno控制系统案例分析





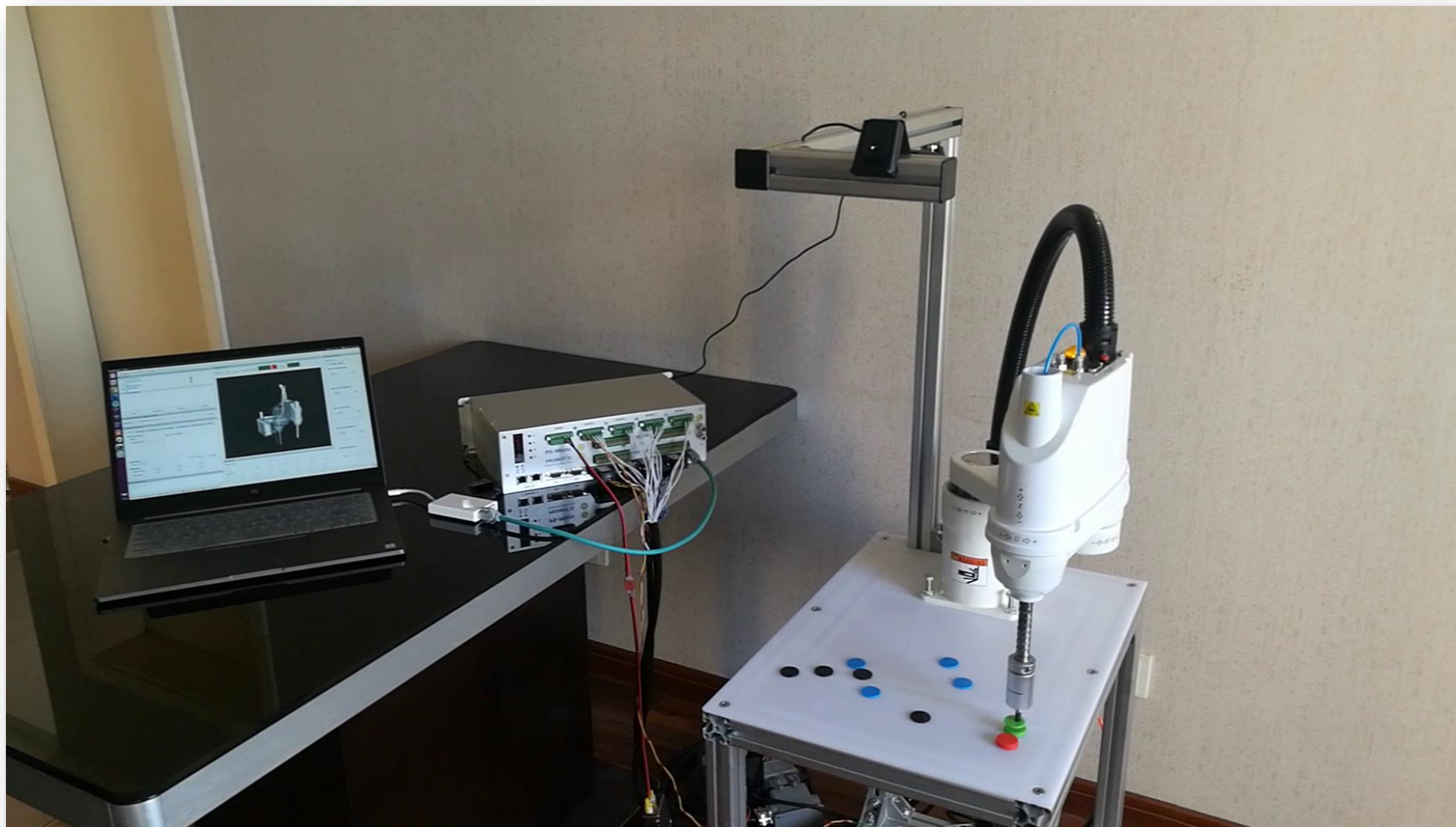
2. PROBOT Anno控制系统案例分析



新一代片上驱控一体系统

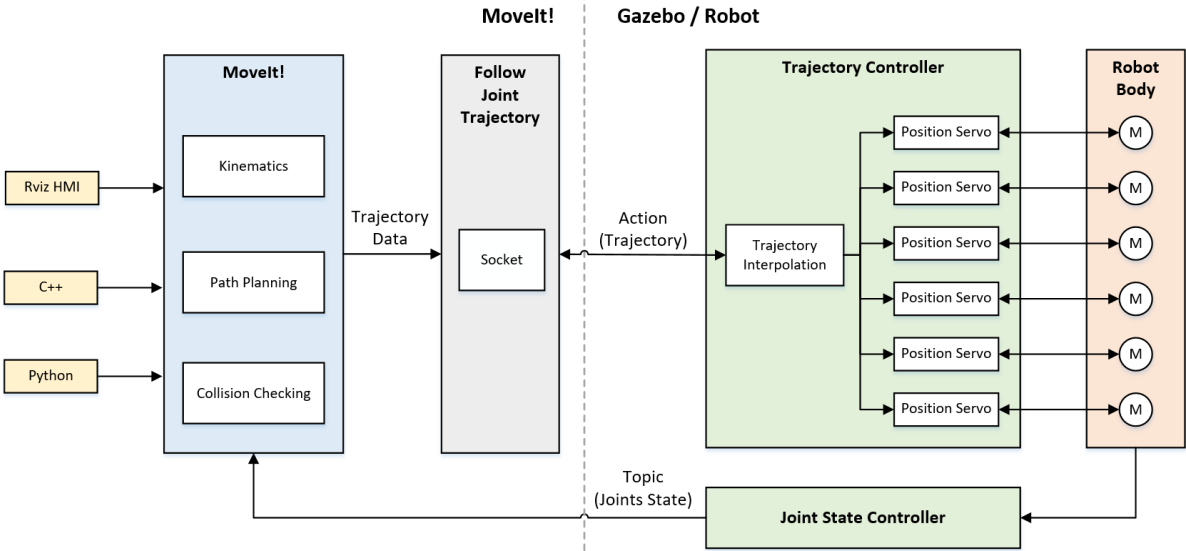


2. PROBOT Anno控制系统案例分析

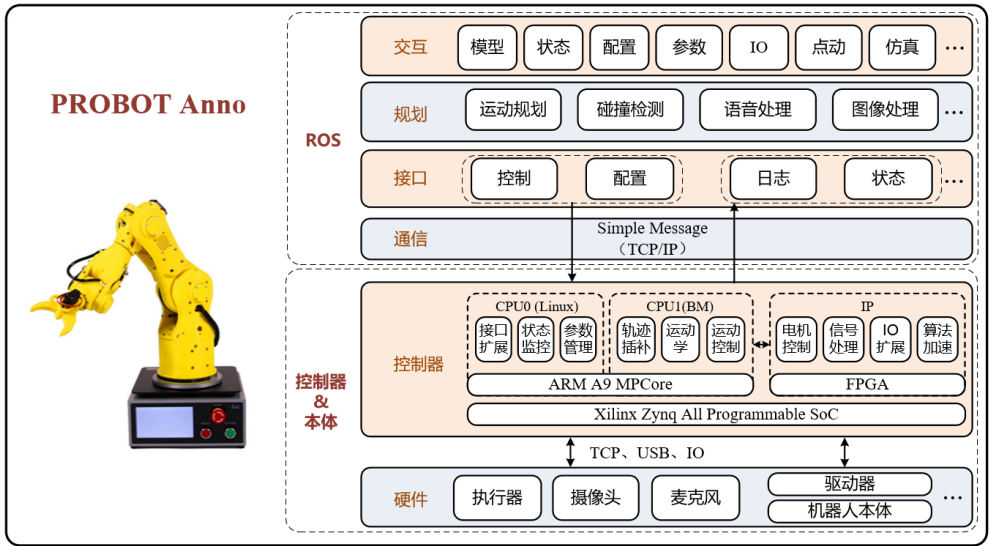




ROS控制系统 设计方法



PROBOT Anno 控制系统案例分析





1. 根据扩展阅读链接，学习ros_control相关的教程；
2. 查阅资料，学习机械臂控制系统的结构和实现方法（**选做**）。



- ROS Control Tutorials

https://github.com/ros-controls/ros_control/wiki

- ros_control: An overview

<https://vimeo.com/107507546>

- PROBOT Anno wiki & Github

http://wiki.ros.org/PROBOT_Anno

https://github.com/ps-micro/PROBOT_Anno

- 功夫手：一款基于ROS的工业机器人

<https://mp.weixin.qq.com/s/iKtKdL1E1Y4WCupFOx8A3w>

- 新一代机器人片上驱控系统

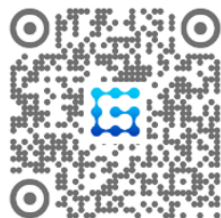
<https://mp.weixin.qq.com/s/eulpdKY5kfp1DeOMsb9BdQ>



Thank You

怕什么真理无穷，进一寸有一寸的欢喜

更多精彩，欢迎关注



 古月居



 古月春旭