

.NET Conf China
2022

.NET 和数控系统

孔令磊
维宏股份-首席架构师



- 
1. 数控系统的组成
 2. 数控系统体系结构
 3. .NET关键技术应用



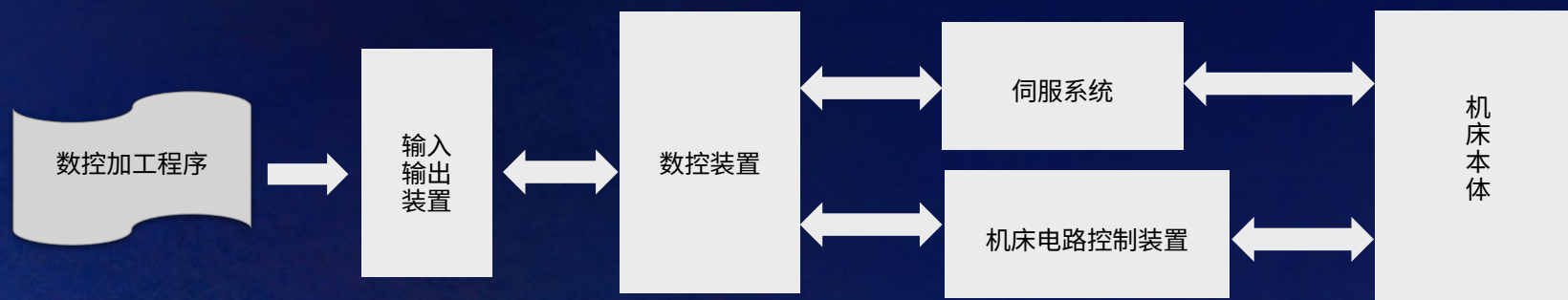
数控系统的组成



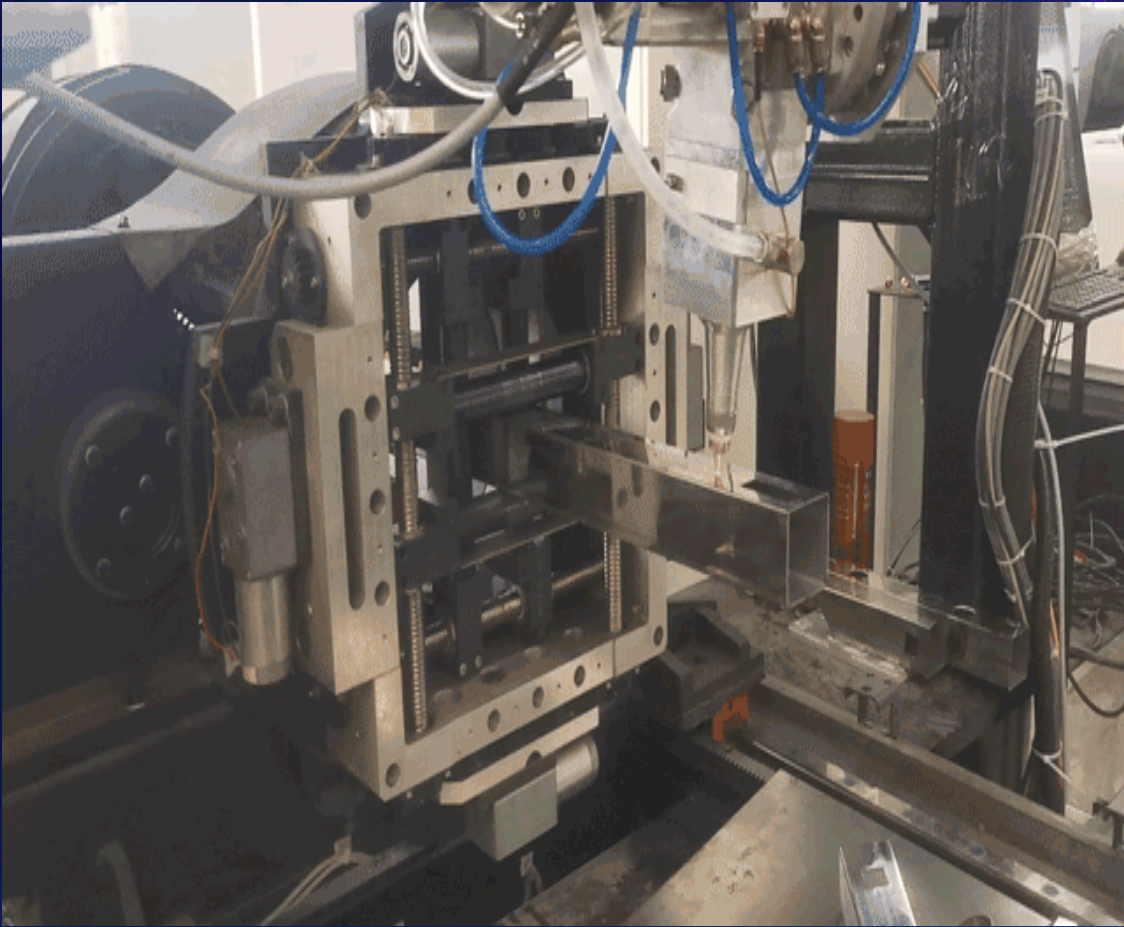
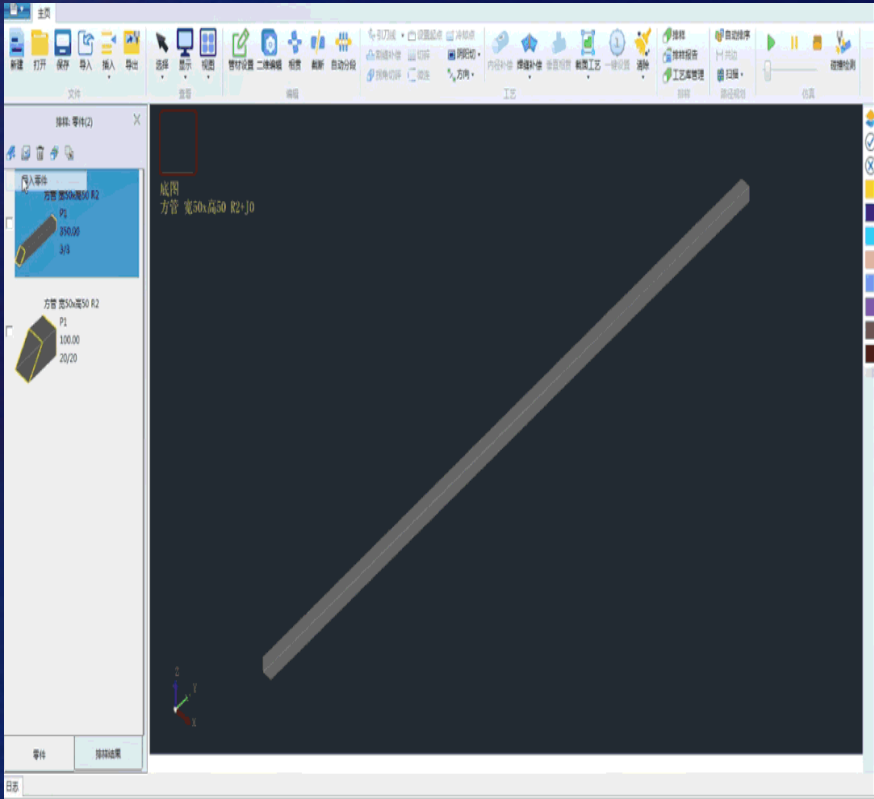
数控系统是数字控制系统的简称，根据计算机存储器中存储的**控制程序**，执行特定的**数值控制**，并配有接口电路和**伺服驱动装置**的专用计算机系统。

数控系统的一般工作过程是由**输入、决策、输出**三个环节组成：

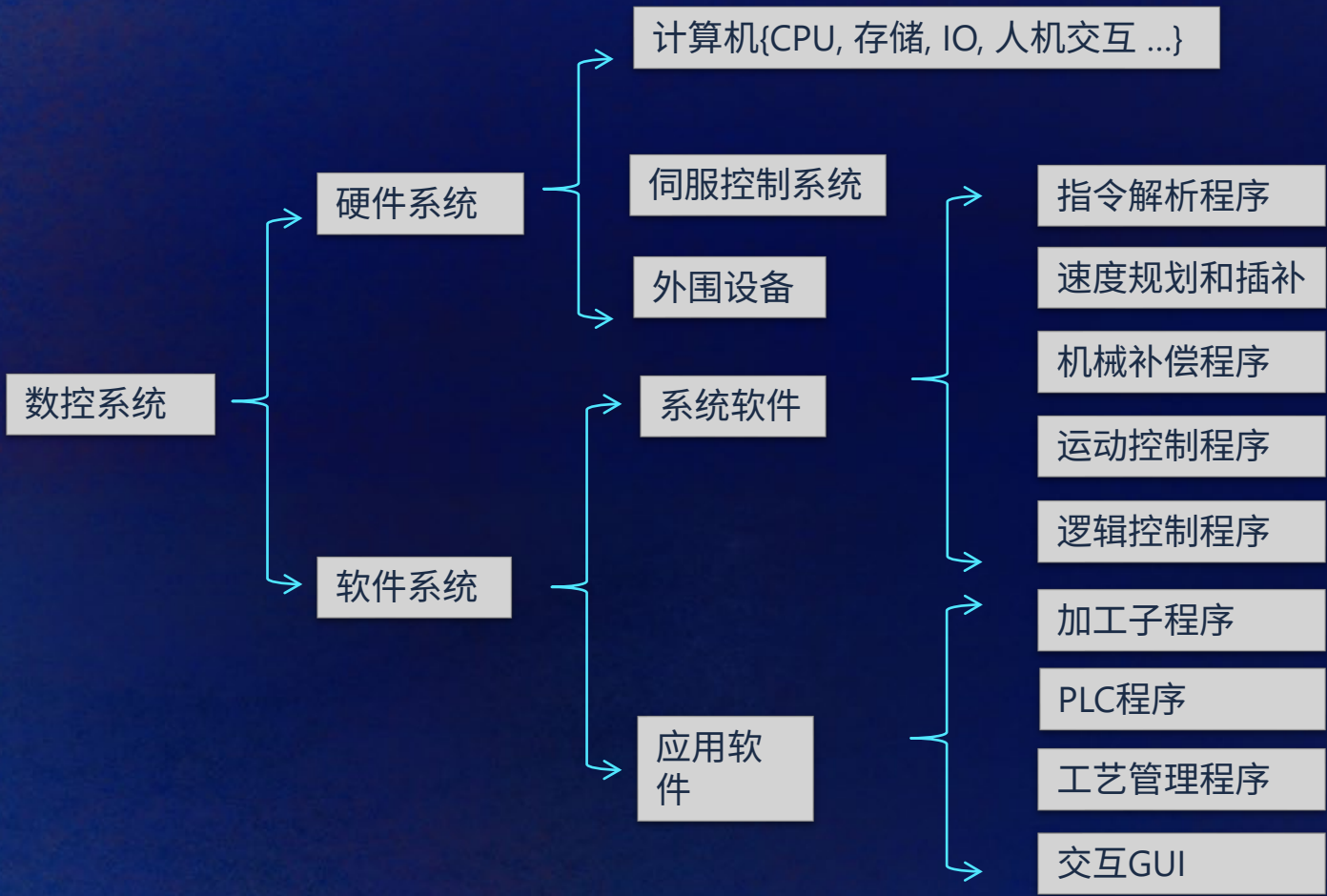
- 首先：采用某种**编程语言**描述一个生产过程，并以**数字的形式**存入**数字控制装置**。
- 其次：通过**数据计算和处理**，数控装置快速识别出该程序所描述的生产过程，并将程序分解为一系列实际的**控制信号**。
- 最后：数控装置输出**实际控制信号**，是相应的**执行部件**产生相应的动作。



数控系统的组成



数控系统的组成

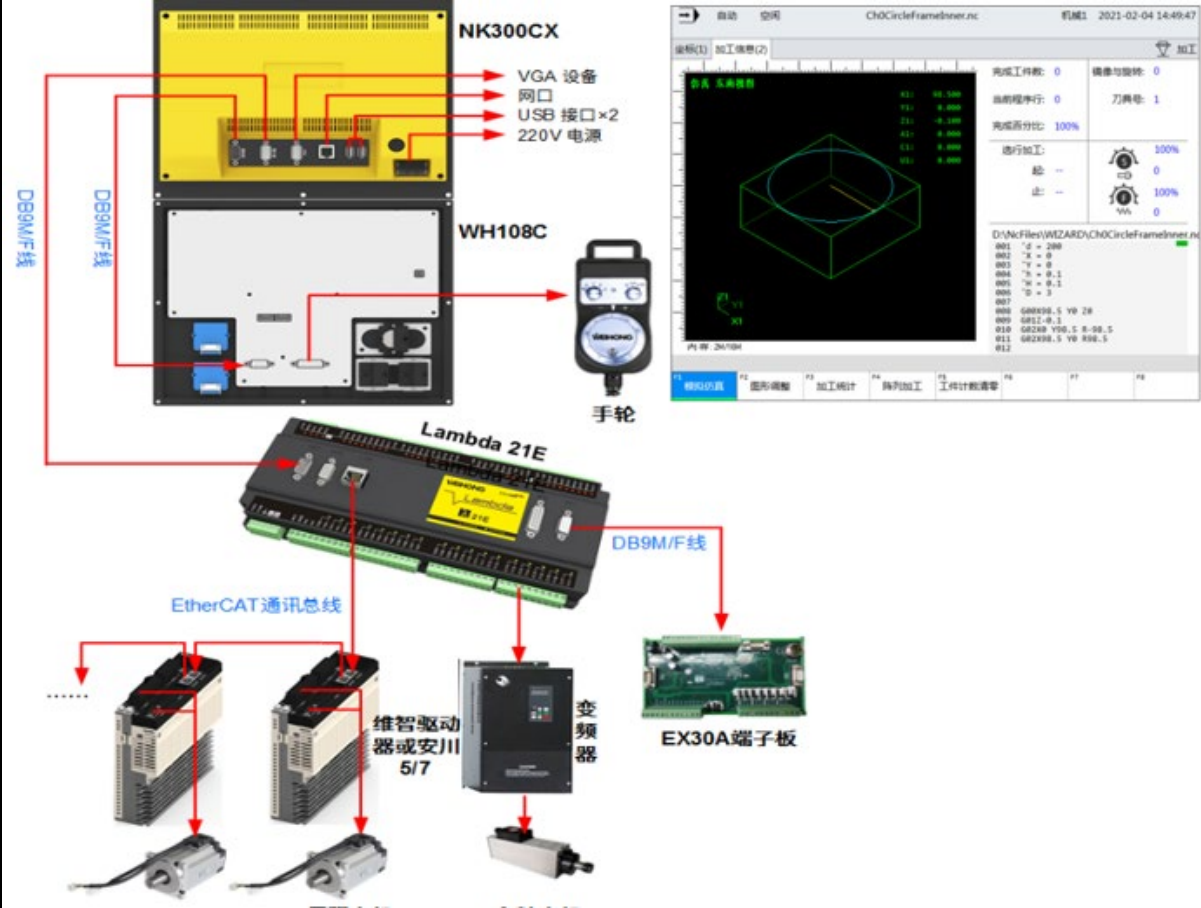


数控系统组成

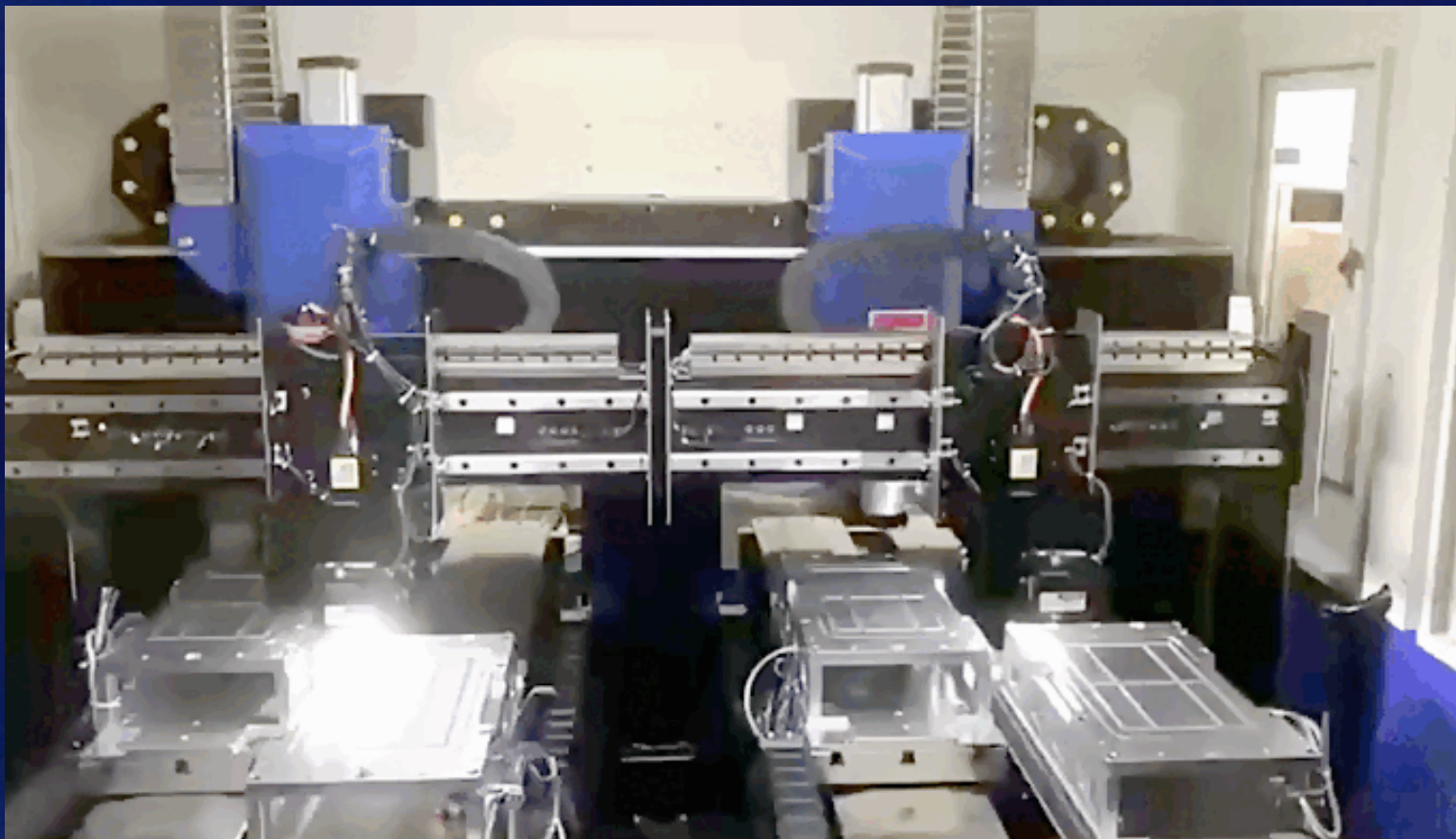
数控系统:硬件是基础，软件是灵魂！

当数控系统采用**计算机数控装置**时，该数控系统称为计算机数控系统（CNC）。计算机数控系统由软件和硬件共同完成数控功能。

基于PC的数控系统提供了丰富的软硬件资源，可以支持绚丽的GUI显示；可以集成CAD/CAM、CCD 等一些上下游辅助软件；可以通过多通道机制由一台数控装置模拟多台独立的控制器。



数控系统的组成



体系结构-开放式数控平台



Phoenix 数控平台是维宏股份完全自主知识产权的新一代开放式数控系统开发平台。该平台从 2016 年开始产品化应用，目前已经应用到家装、雕刻、3C、金属切割、激光切割等多个行业。

特征：

1. 可替换：各核心功能模块化， 标准化发布(nuget)， 可替换。
2. 可移植： 对硬件逻辑抽象， windows/linux 跨平台移植。
3. 伸缩能力： 接口依赖于契约,按机型集成所需模块，即插即用。
4. 扩展能力： GUI高度重定义、PLC二次开发、子程序开发。
5. 互操作能力： 支持Phoenix、MII和EtherCAT现场总线协议，通过内部网关代理支持OPCUA等各种IT互联协议。



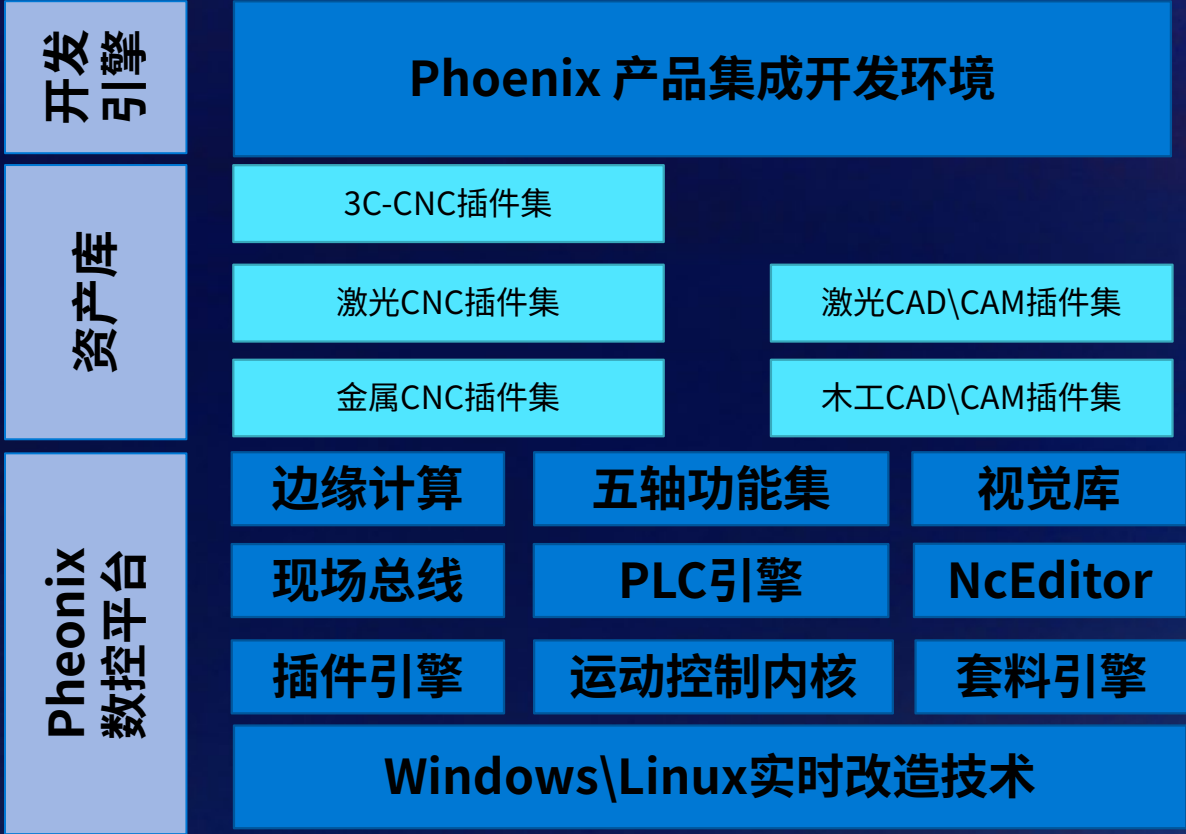
体系结构-产品群架构图



体系结构-业务架构



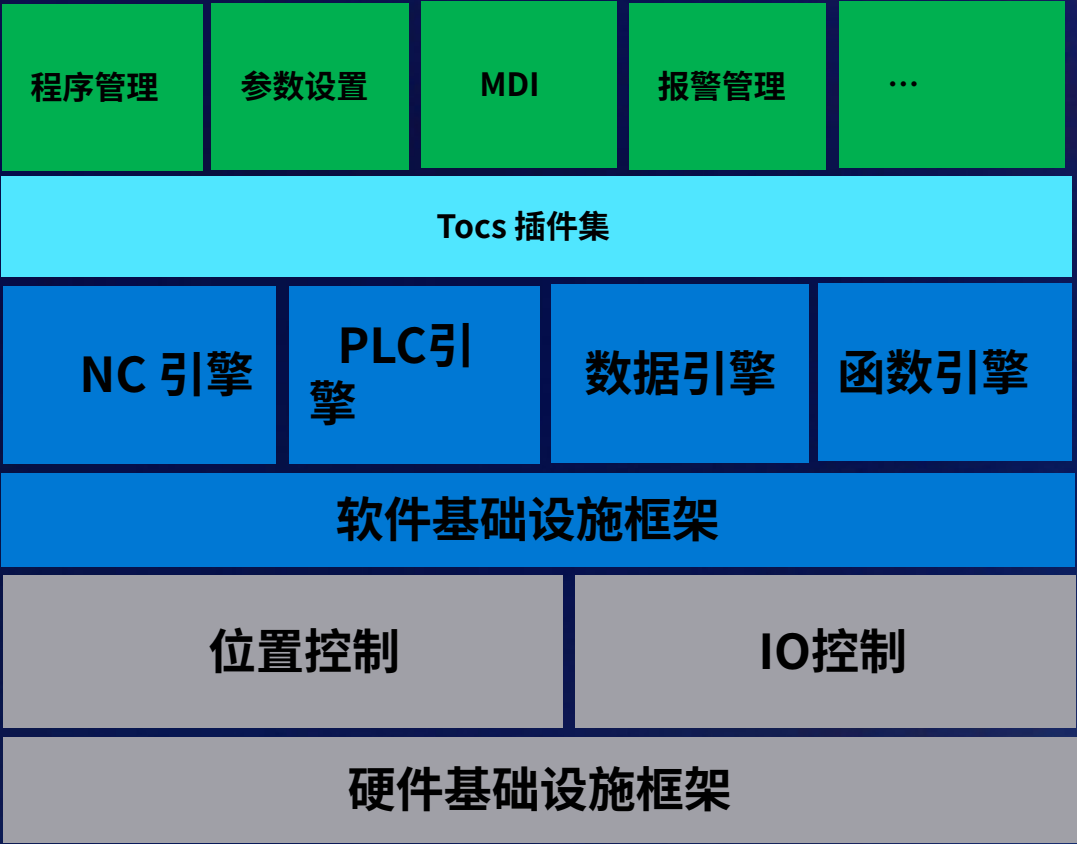
- 数控平台为各行业产品提供基础支持，通过 Toc 的开发满足各行业各场景的终端产品。
- Toc 插件系统让 OEM 产品的定制输出以资产的形式保存下来。OEM的Toc资产支持加密保护，具备交易的属性。
- 未来，OEM资产需要交易的平台，这是IT的生态了。



体系结构-组件图



- 三层结构：硬件和抽象、内核层和应用层。
- 内核系统要以可靠性和确定性为主，以插件的形式扩充业务控制能力。
- 应用层面向操作人员，包括交互界面、工艺管理、日志报警等功能。

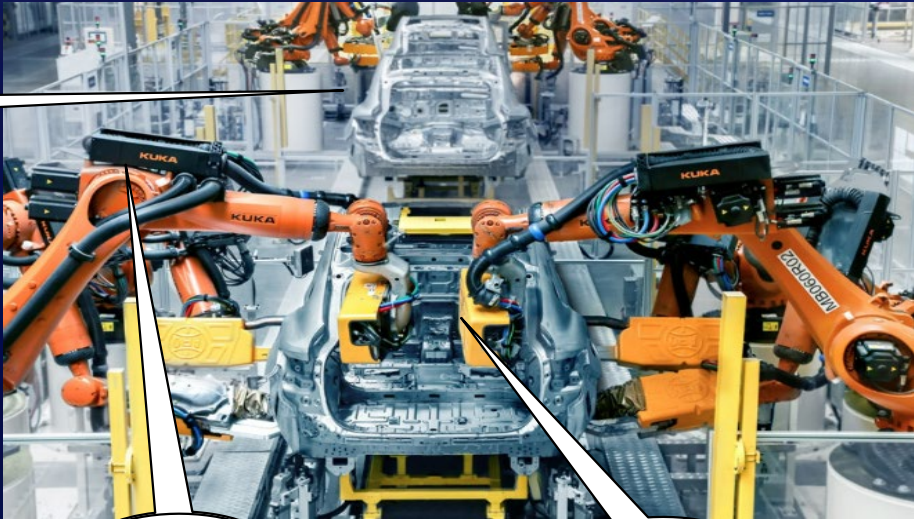


体系结构-NC引擎



- 串联多级 **生产者/消费者** 流水线设计模型。
- 协程作业
`for (可切换条件) write(f(read(part)));`
- 各级work独立成可替换dll.

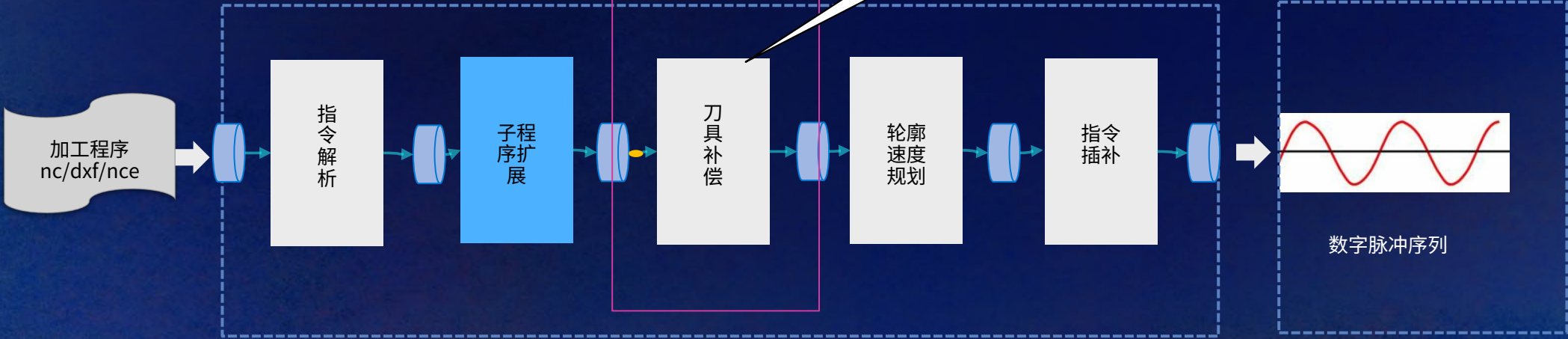
工作台



工人

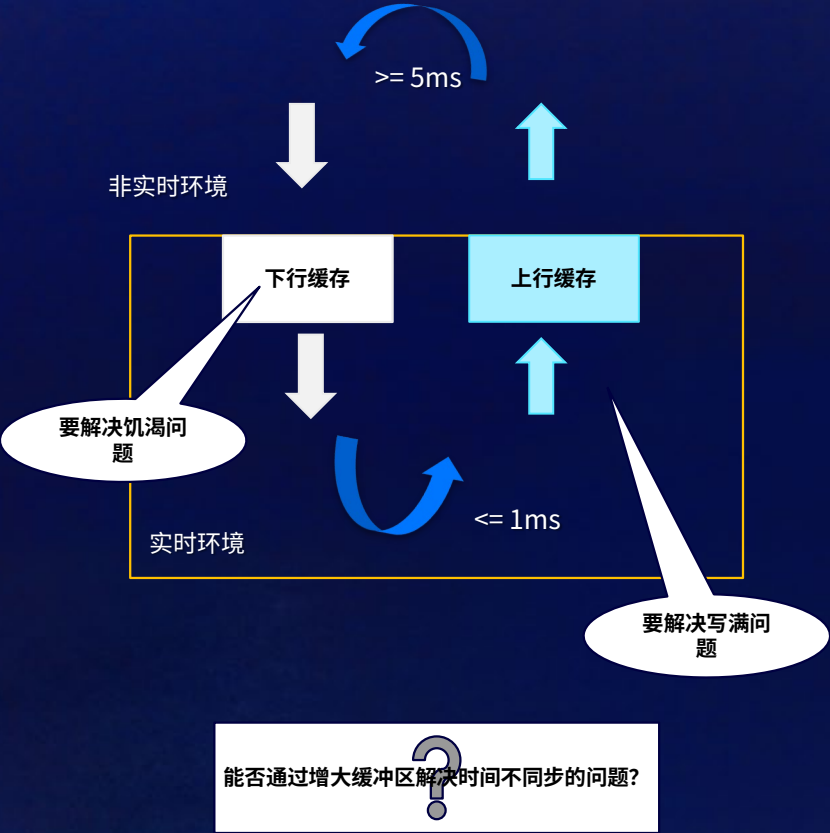
零部件

类比

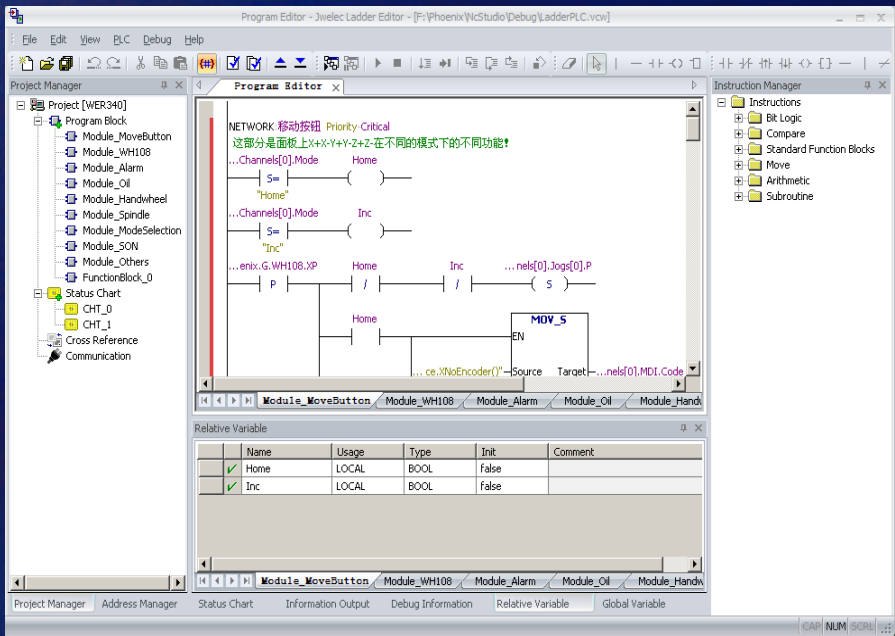


体系结构-NC引擎

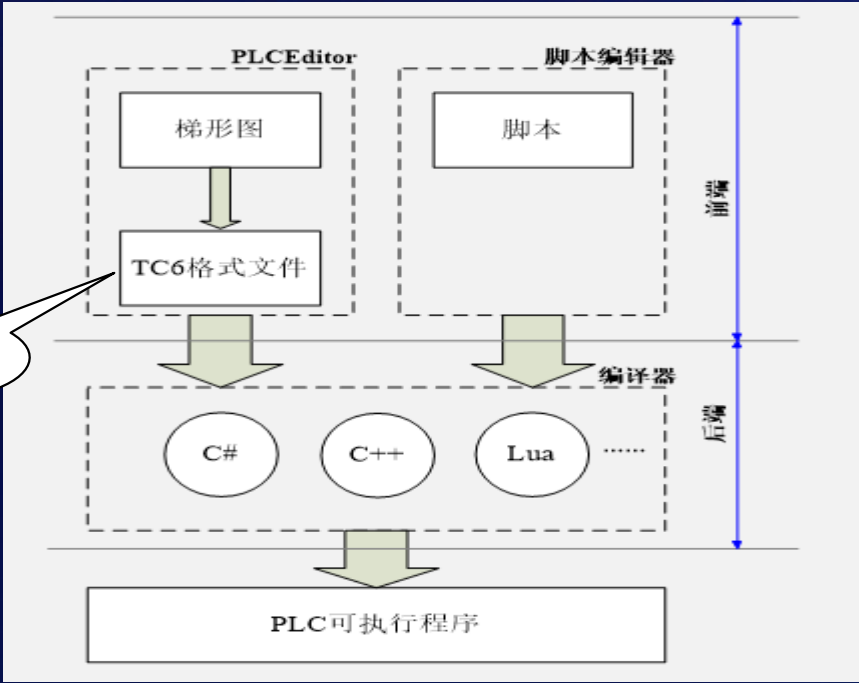
- 解决数据流在非实时环境和实时环境之间的控制问题。
- R3层的高性能定时器。
- R3/R0,上与下的选择问题。



体系结构-PLC引擎



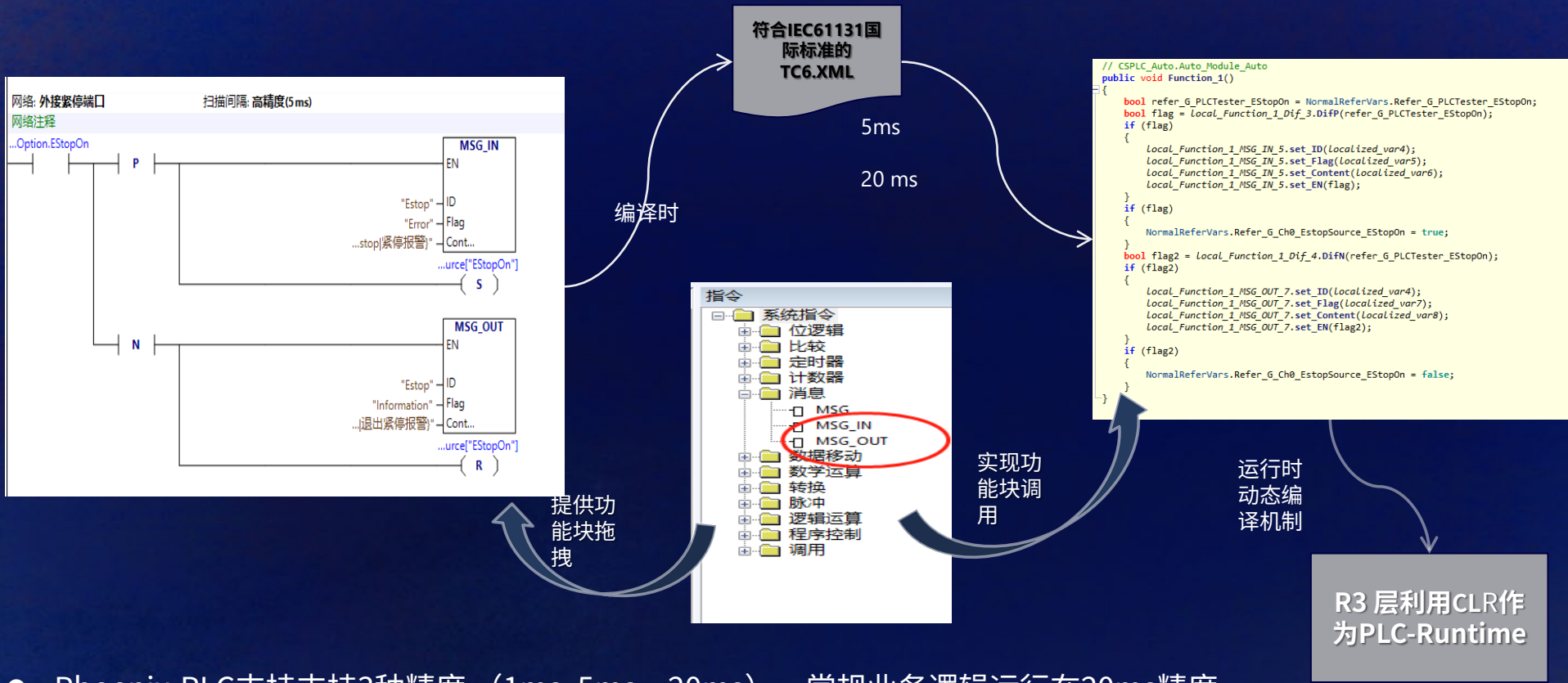
IEC1131-3



PLC作为自动化领域的一种通用技术，作用范围广泛、实用性强。特别是对于现场编程来说有其独特的优势。它是一种工业领域的低代码开发平台（IEC61131 & IEC61499），现场设计人员根据控制系统的工艺控制要求，通过PLC编程语言的编制程序下载到控制器中执行，以对电气工程师比较友好的梯形图语言（LD）语言为主，十分简单易学。



体系结构-PLC引擎

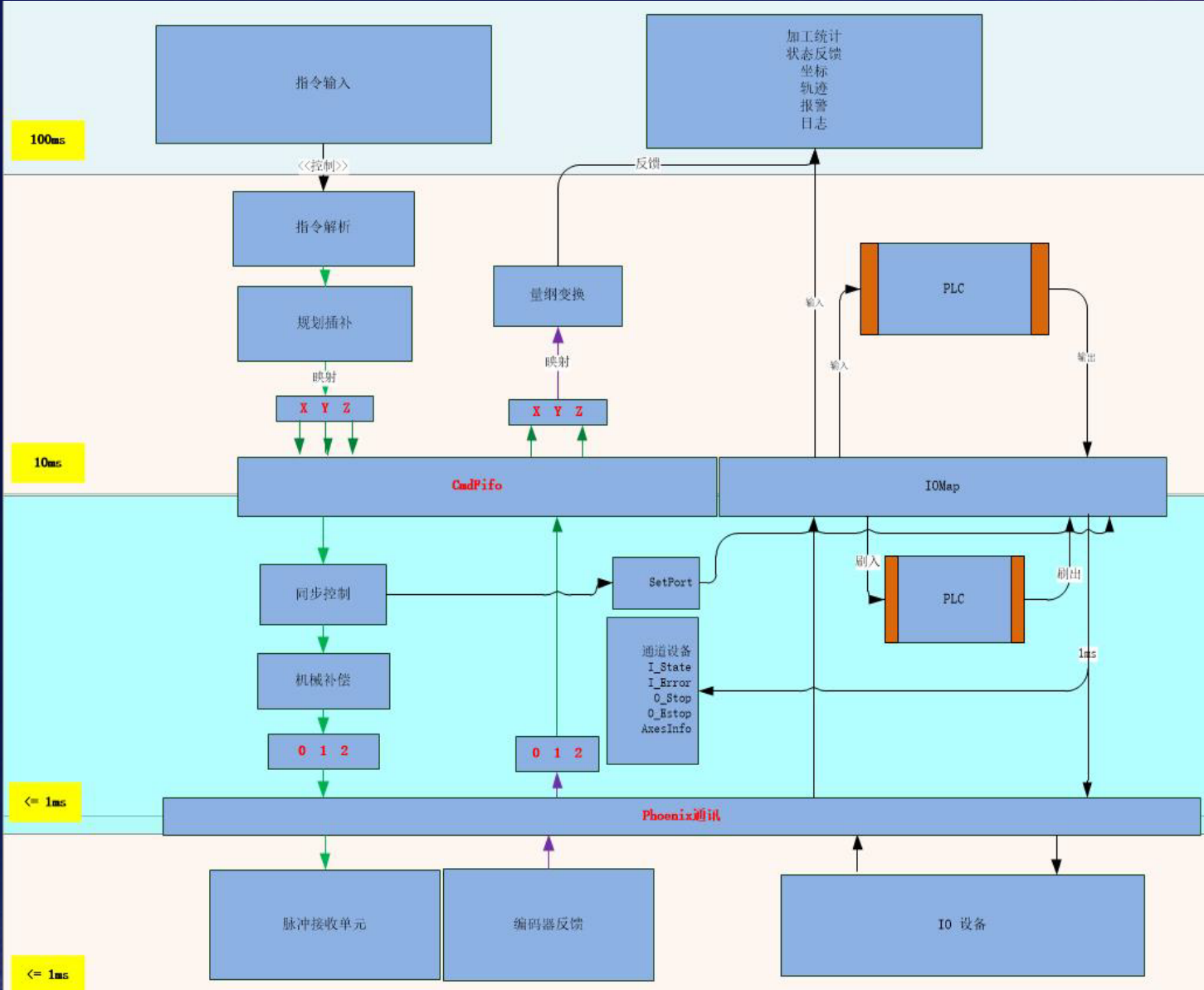


- Phoenix-PLC支持支持3种精度（1ms, 5ms, 20ms），常规业务逻辑运行在20ms精度下。
- 梯形图项目运行时Compile为.NET程序集，在CLR环境中执行，PLC文件可以分发到任何支持CLR的系统中。



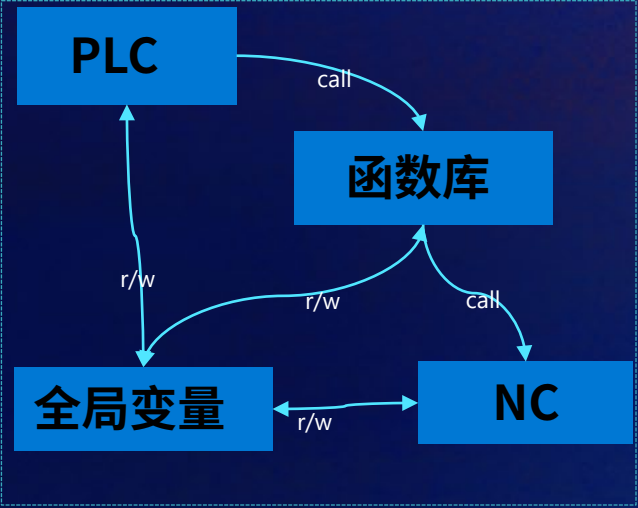
体系结构-控制过程

带有时间和层次的运动控制数据流



体系结构-插件扩展

zua		Tc6.xml		x.dll	
C.NcProg	C.PLC	C.Mod	C.Target	C.Toc.Json	
B.NcProg	B.PLC	B.Mod	B.Target	B.Toc.Json	
A.NcProg	A.PLC	A.Mod	A.Target	A.Toc.Json	
NC 引擎		PLC引擎		数据引擎	函数引擎
				配置管理	

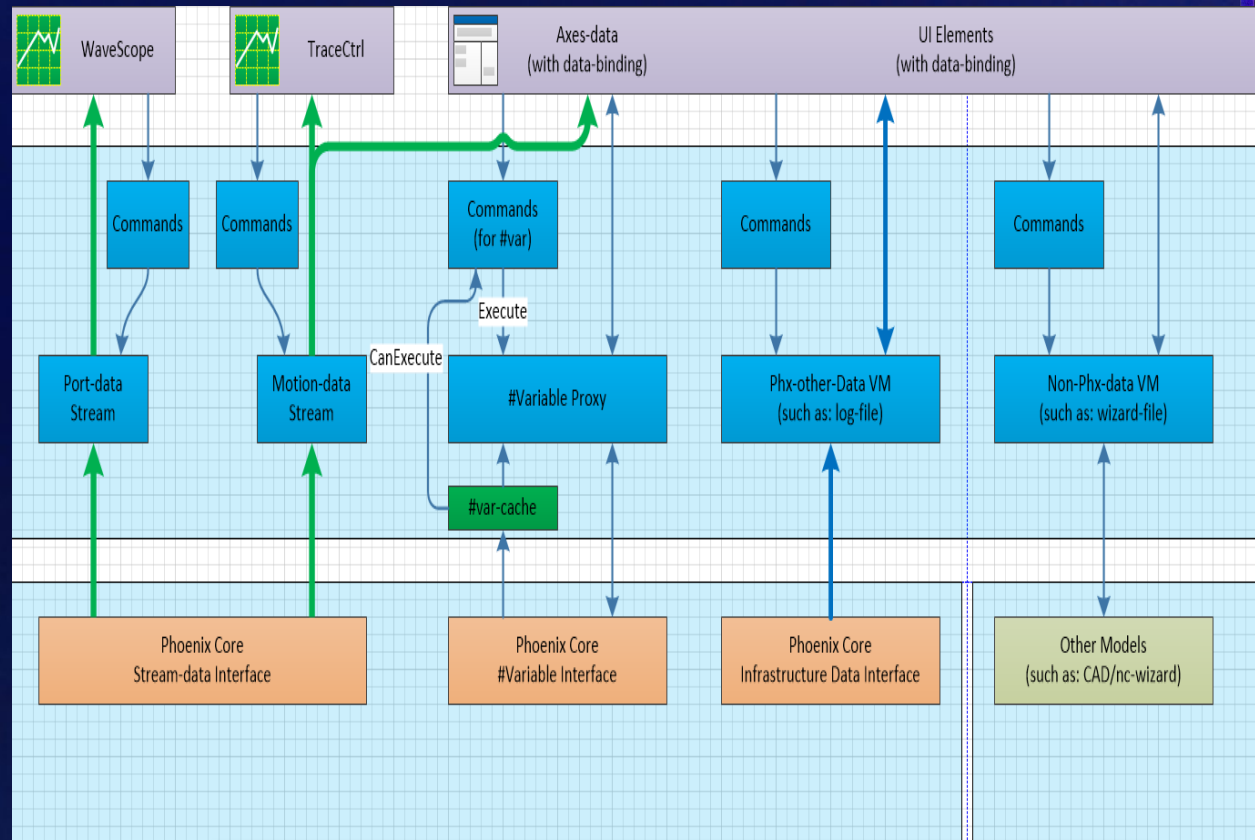


- 一个微内核插件系统(系统插件和业务插件)。
- 业务插件独立发布,重用的资产。
- 一个产品/配置等于一个插件集合。

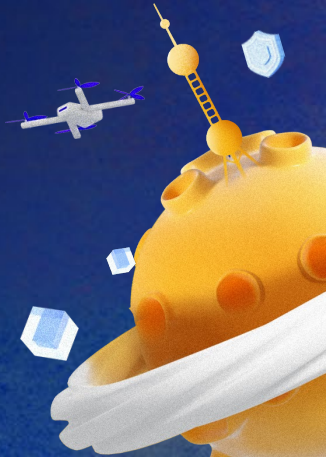
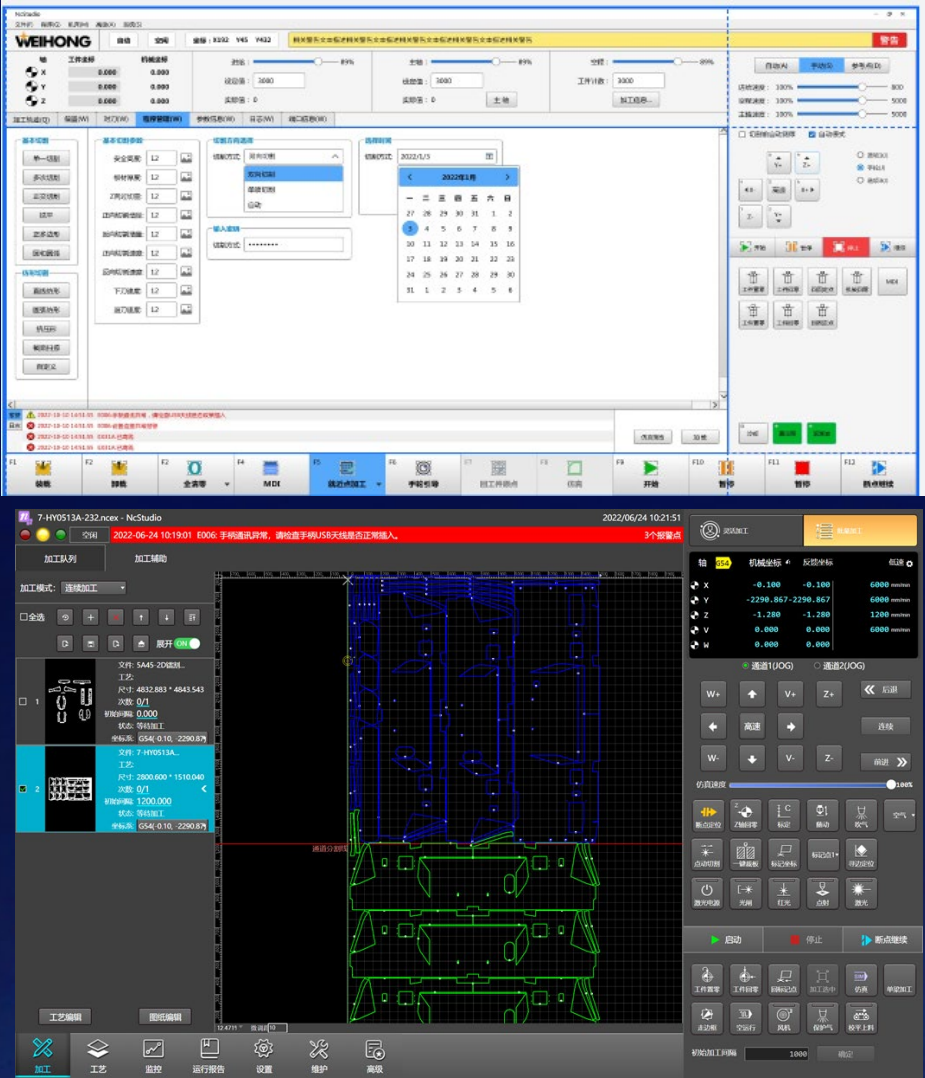
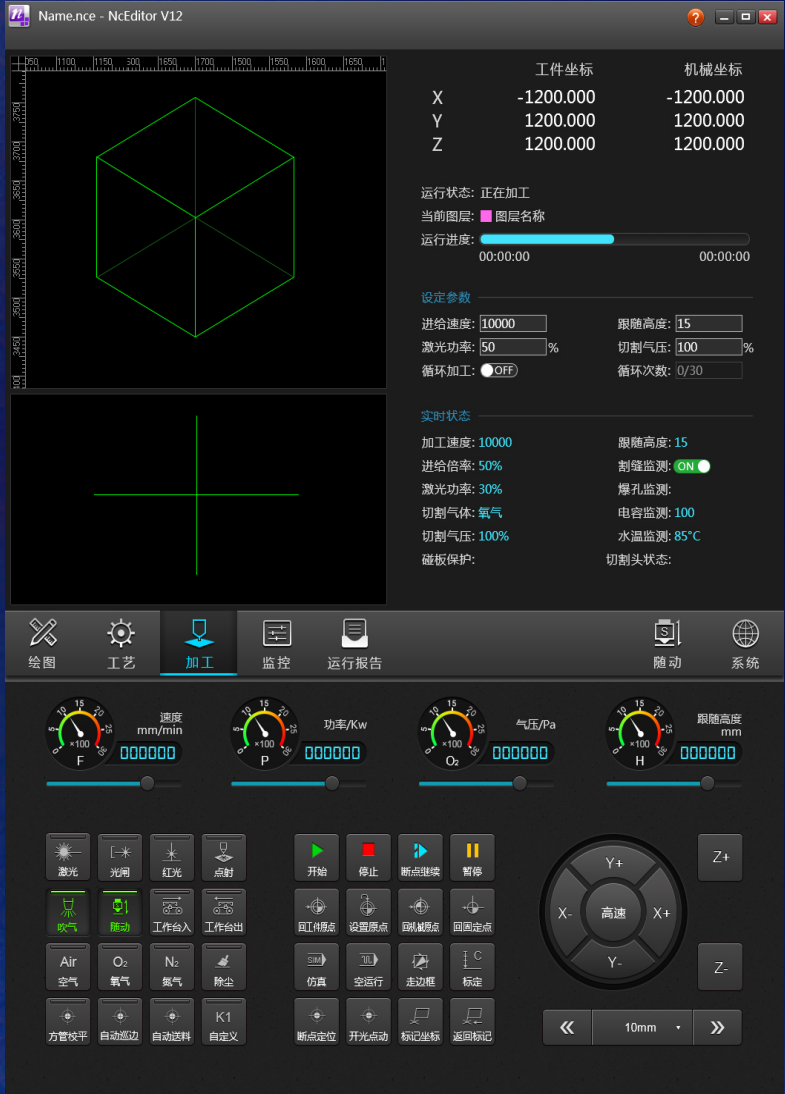
体系结构-GUI框架

数控内核和GUI分离，彻底释放GUI在PC端潜力。

- WPF作为GUI框架。
- 以MVVM作为编程模型
- 以GenericHost作为主机框架
- 通过Prism来管理界面布局
- CommunityToolkit.Mvvm
- HandyControl样式框架
- 设计Brisk工业控件库



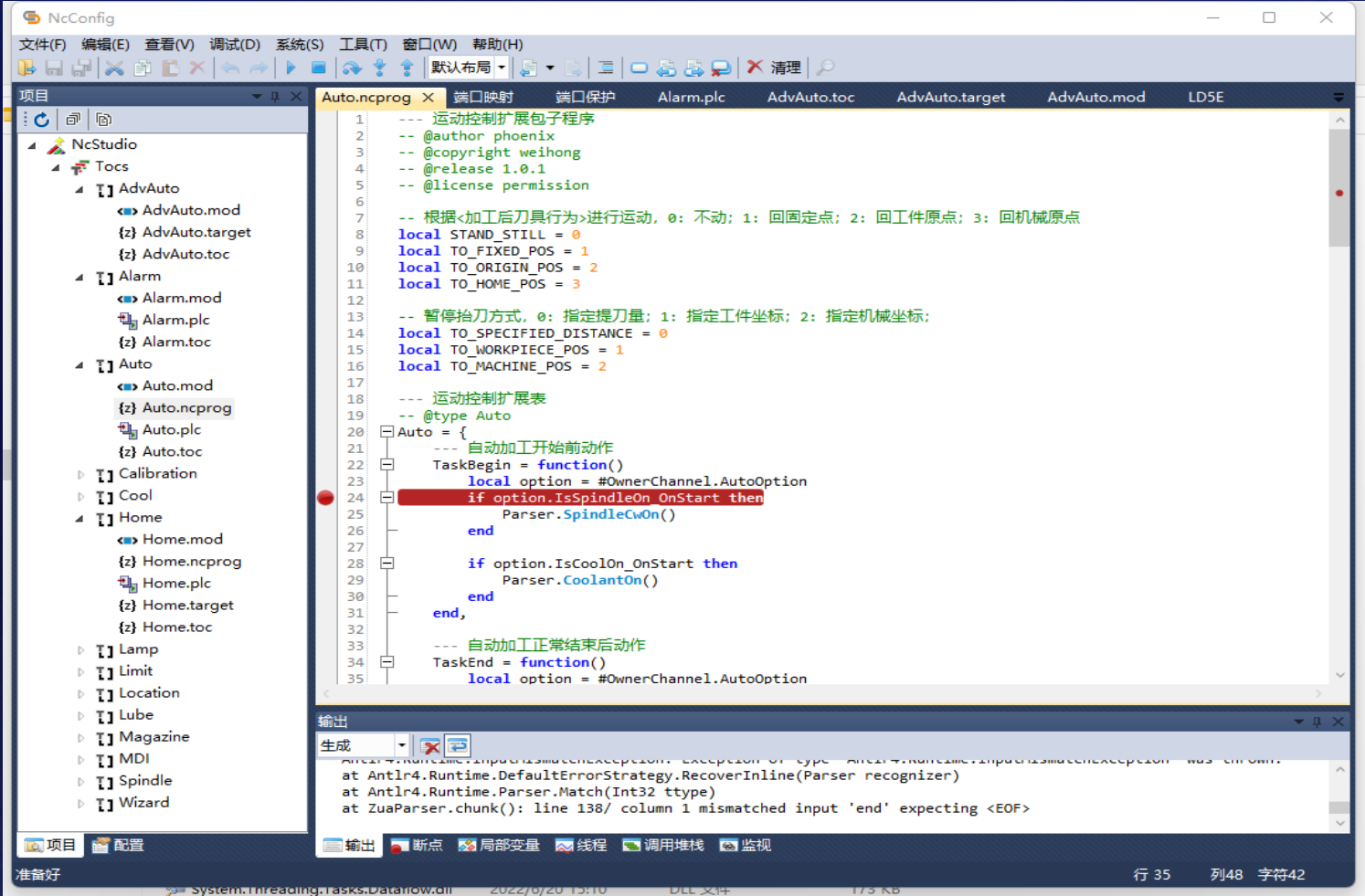
体系结构-GUI框架



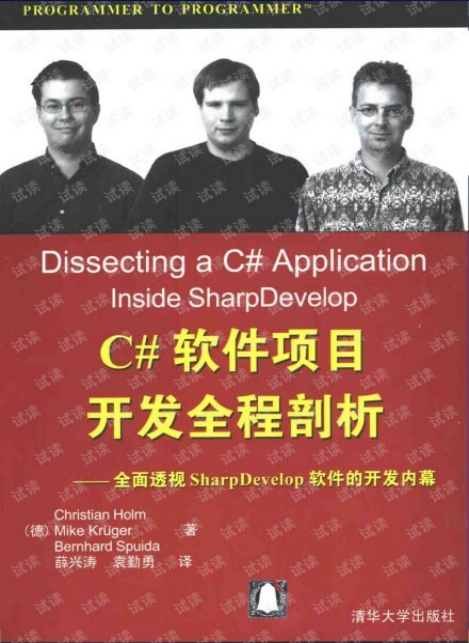
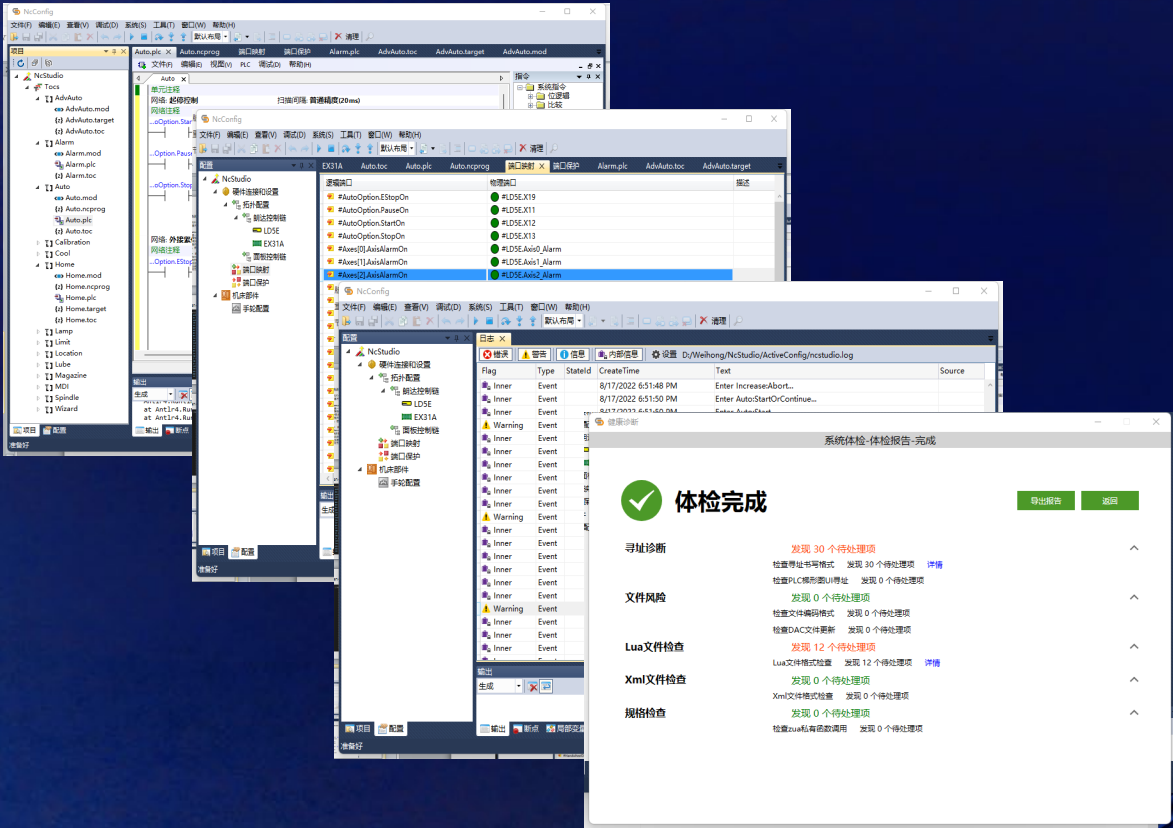
体系结构-集成开发环境



Sharpdevelop
为基础的打造我
们的集成开发、
调试、配置工具。



体系结构-集成开发环境



NET技术

应用



.NET 能做的远不止是上位机!

数控系统是一个跨学科跨语言整合的系统技术，在这个复杂的系统中.NET 可以发挥相当重要的角色。

- .NET是一个集大成者，借助于强大的互操作能力跨语言(C/C++、Lua),跨模块（解析、通讯、算法等等）完美集成。
- 实现即插即用的插件机制，分系统插件和业务插件，支持根据不同的机床型号配置可伸缩的数控系统软件。
- 实现了能够跨模块跨语言数据（信号和参数）共享的寻址系统。
- 实现了用户态的软PLC运行时。
- 支持对GUI无限制的定制能力，快速满足OEM的差异化需求。
- 支持面向数控平台的二次开发集成环境。

.NET Conf China

.NET技术

应用



性能的基础 `span<T>`:

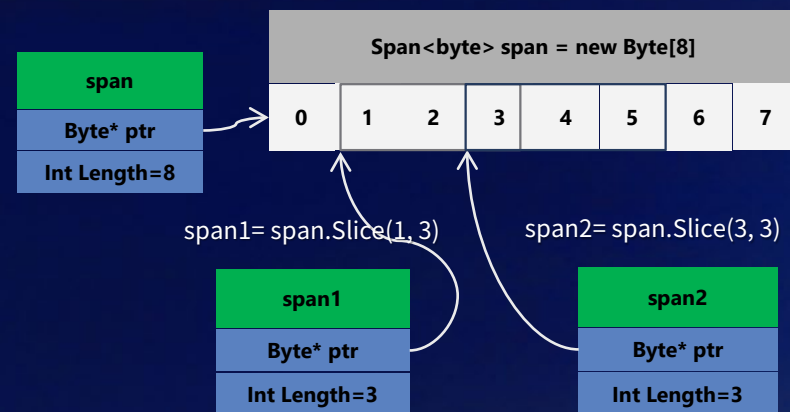
为了提升性能，.NETCore2.1 版本中引入的能够发挥关键作用的新值类型。其可以访问任意相邻的内存区域，不限于托管内存、非托管内存、堆栈内存。且能够确保安全和高性能的特性。

特征:

- 新的值类型。
- 访问任意连续内存区域（托管堆，非托管堆，栈）。
- 确保安全和高性能特性。

优点:

- 值类型低开销或零开销
- 统一了编程体验
- 保持了高性能
- 提供内存和类型安全



.NET技术 --- 应用



优雅又安全的指针

漂亮的改进：

- 看不见指针，但是可以操作内存，同时简化掉unsafe。
- Stackalloc 升级为常规的表达式。更方便使用。
- 更安全。内部指针管理被CLR托管，支持溢出检查。
- 仍旧很快。增加了验证代码，CLR也是努力让性能保持最优。

从.NET Framework时期常常这样操作堆栈内存：

```
unsafe{
    int length = 3;
    int* numbers = stackalloc int[length];
    for (var i = 0; i < length; i++)
        numbers[i] = i;
}
```

现在可以这样操作堆栈了：

```
int length = 3;
Span<int> numbers = stackalloc int[length];
for (var i = 0; i < length; i++)
    numbers[i] = i;
```

竟然还支持表达式！

```
Span<int> numbers = stackalloc[] { 1, 2, 3, 4, 5, 6 };
var ind = numbers.IndexOfAny(stackalloc[] { 2, 4, 6, 8 });
Console.WriteLine(ind); // 输出: 1 且不会触发GC
```

.NET Conf China

.NET技术 --- 应用



工业软件迭代周期长?
不,我们要上更快的.NET7!



交流沟通群

.NET Conf China

Thank you!

