



作者：北岛李工

原创文章，转载请注明出处。

更多实用资料请登录官网：www.founderchip.com

本文是Snap7软件包系列教程的第3篇。

在上一篇文章中，我们介绍了S7通信协议中有三种角色：客户端/服务器/伙伴；有两种通信模式：客户端/服务器（Client/Server）模式和伙伴/伙伴（Partner/Partner）模式，最常见的是客户端/服务器模式。

与S7通信中的角色类似，Snap7软件包中包括三个主要的类^①：

- Snap7Client：S7通信的客户端角色；
- Snap7Server：S7通信的服务器角色；
- Snap7Partner：S7通信的伙伴角色；

^①"类"是面向对象编程中的重要概念，它具有可定义的属性、方法和事件，具有继承性与多态性。类是对象的模板，类的实例就是对象变量。

基于 Snap7

使用C#编程访问西门子PLC系列教程(3)

—Snap7Client(建立连接)

本文及后续几篇文章都将介绍Snap7Client类，本文主要介绍如何使用Snap7Client与S7系列PLC建立连接，包括如下几个主题：

1. S7连接的参数；
2. Snap7Client可连接的对象；
3. S7-1200/1500连接的设置；

1、S7连接的条件

使用S7协议连接S7系列PLC需要明确三个参数：远程IP地址、本地TSAP和远程TSAP；

- 远程IP地址是指要连接的目标PLC的IP地址;
- TSAP是"Transport Services Access Point"的缩写, 中文翻译为"传输服务访问点"。它是传输层与会话层之间的访问接口, 每一个TSAP可以唯一标识一个访问实例。具体来说, 在一个传输连接中可能存在多个进程, TASP可以区分不同进程的数据传输, 与TCP或UDP协议的端口号 (Port)类似, 在一些网络应用中可以使用<IP地址> <端口号>的方式来表示TSAP。TSAP可以是数字或者字符串, 在S7协议中, TSAP是一个16位的字。
- 从Snap7的1.1.0版本开始, 为了支持LOGO! 和S7-200系列PLC, Snap7Client允许在连接PLC之前先调用SetConnectionParams()方法②来设置本地TSAP和远程TSAP。不过在很多时候, 我们使用Snap7Client的ConnectTo方法, 该方法只需要给出远程PLC的IP地址、机架号(Rack)和槽号(Slot);

②"方法"是类的函数, 调用方法时就会执行该函数。更多内容可以查看“面向对象编程”的资料。

Rack和Slot是Snap7Client连接的关键, 不同类型的PLC其Rack和Slot的定义不同, 详见下面的列表:

PLC类型	Rack	Slot	备注
S7-300	0	2	
S7-400/Win AC	不确定	不确定	见硬件组态
S7-1200/1500	0	0	或0, 1
S7-200 SMART	0	1	
CP343	0	0	或硬件组态
CP443	不确定	不确定	见硬件组态

Snap7Client的连接使用"ping"机制。

在尝试与远程PLC建立连接之前, Snap7Client首先ping一下对方的IP地址。如果得到回复, 则发起S7连接请求; 如果超时没有得到回复, 则可以判断对方不在线(未通电或线路故障); 由于ping机制的超时是固定的, 因此可以节省TCP协议不确定的超时时间, 使响应更加快速;

2、Snap7Client可连接的对象

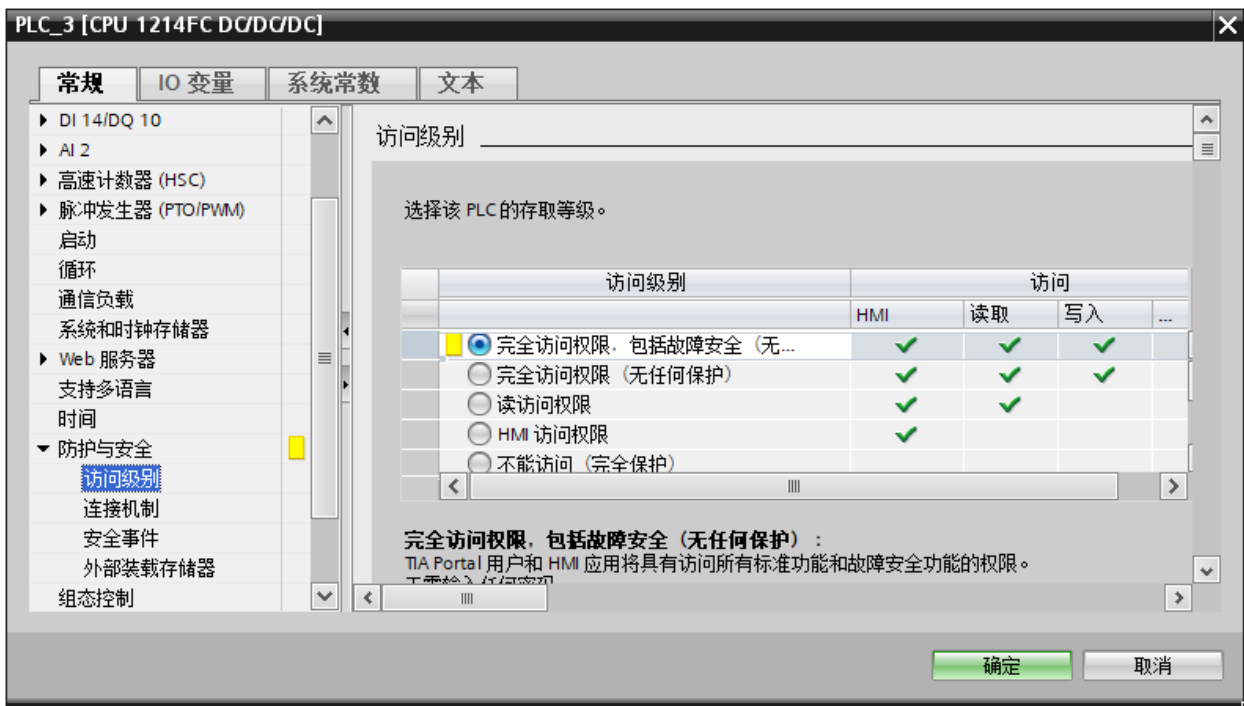
Snap7Client可以连接的对象包括：

- S7-300/400 CPU;
- S7-1200/1500 CPU;
- S7-200/ 200 SMART CPU;
- LOGO! CPU;
- CP343/443;
- WinAC;
- SINAMICS变频器(比如G120系列) ;
- Snap7Server;

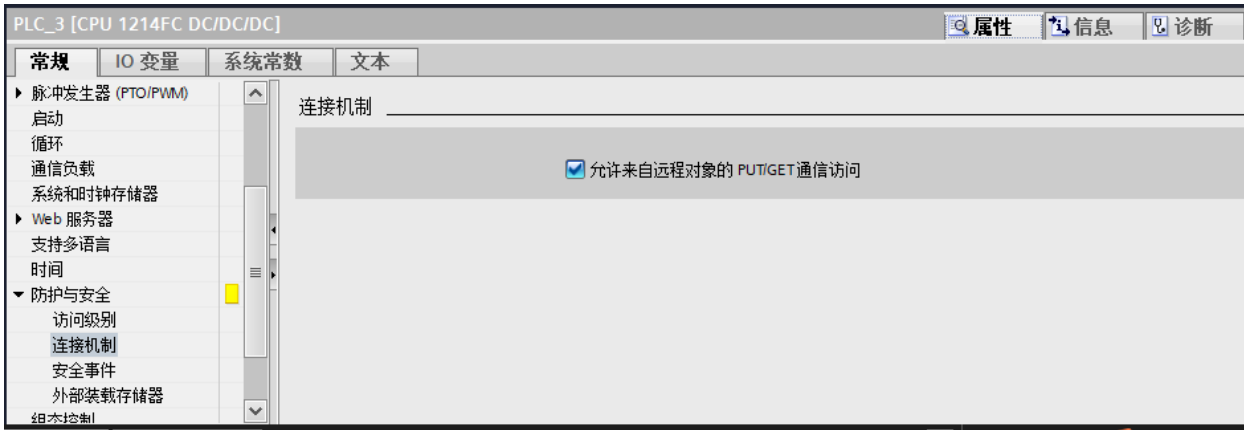
3、S7-1200/1500连接的设置

由于S7-1200/1500系列PLC采用扩展的S7协议，而Snap7Client使用的是"基本" S7协议，因此在连接这类CPU时需要做一些设置，包括：

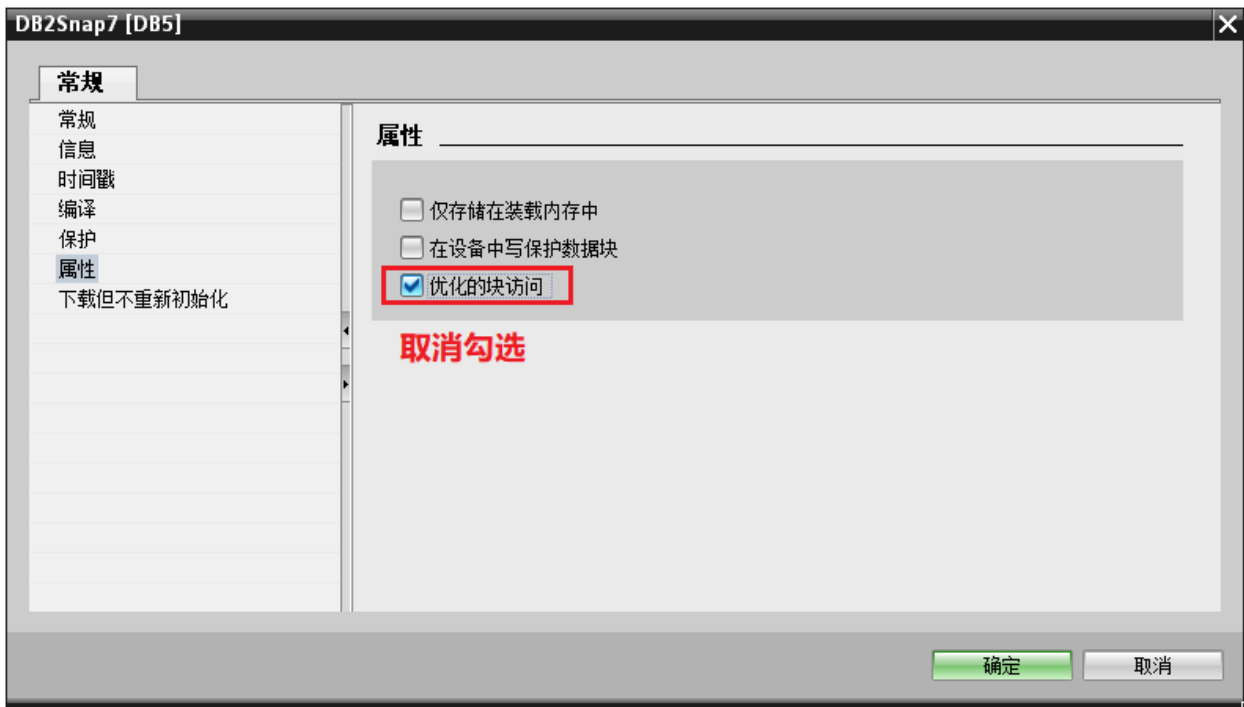
- CPU的访问级别必须为"完全访问权限"，如下图所示：



- CPU的"连接机制"中要勾选"允许来自远程对象的PUT/GET通信访问"，如下图所示：



- 对数据块的访问仅限于全局数据块(Global DB), 并且需要取消勾选"优化的块访问", 如下图所示:



好了, 关于Snap7Client的连接条件及相数设置就先介绍这么多。下一期文章我们介绍Snap7Client的三个方法: ConnectTo()、SetConnectionParams()和 Disconnect()。

我最近开通了微信视频号, 以视频的方式分享一些工控小知识, 欢迎扫描下方二维码观看:



扫码关注我的微信视频号