海水自动采样集成需求：

不改变之前自动取样的步骤和动作，只是采样点和时间的可配置。

1．自动采样点可配置（例如只在35℃、15℃和0℃执行自动采样，通过软件选择或直接修改配置文件的方式都可以）。

2．之前的执行逻辑是两步：准备采样和开始采样。现在将开始采样从时间上分成两部分：冲洗和取样，冲洗完成的时候软件（控温软件）界面给提示（给了提示后，人员会把取样管放到取样瓶里，取样管未放到取样瓶里之前是放在废液桶里的）。

3．准备采样和开始采样执行的时间可配置（例如原先分别是5分钟和1分钟，现在可根据实际情况随时设定），开始采样执行的时间实际上是冲洗时间+取样时间，因此现在配置时间是三个数字：5分钟+1分钟+1分钟

传感器部件状态需求

class Status

{

public float ZhucaoTemperature = 35.0000f;//主槽设定温度

public float ZhucaoTemperatureReal = 35.0000f;//主槽当前温度

public int ZhucaoPower = 0; //主槽加热功率

public float FucaoTemperature = 34.0000f; //辅槽设定温度

public float FucaoTemperatureReal = 34.0000f; //辅槽当前温度

public int FucaoPower = 0; //辅槽加热功率

public bool Zongdianyuan = false; //总电源开关

public bool Zhucaokongwen = false; //主槽控温开关

public bool Zhucaokuaileng = false; //主槽快冷开关

public bool Fucaokongwen = false; //辅槽控温开关

public bool Fucaokuaileng = false; //辅槽快冷开关

public bool Fucaozhileng = false; //辅槽制冷开关

public bool Fucaoxunhuan = false; //辅槽循环开关

public bool Haishuijin = false; //海水进开关

public bool Haishuichu = false;//海水出开关

public DateTime DataTime = DateTime.Now;//数据采集时间

public double Temperature = double.NaN;//标准温度

public double Conductivity = double.NaN;//标准电导率

public double Salinity = double.NaN;//标准盐度

}

例子：

Status o = new Status() { ZhucaoTemperature = 20.0000f, ZhucaoTemperatureReal = 20.0000f, ZhucaoPower = 10, FucaoTemperature = 20.0000f, FucaoTemperatureReal = 20.0000f, FucaoPower = 10, Zongdianyuan = true, Zhucaokongwen = true, Zhucaokuaileng = true, Fucaokongwen = true, Fucaokuaileng = true, Fucaozhileng = true, Fucaoxunhuan = true, Haishuijin = true, Haishuichu = true, DataTime = DateTime.Now, Temperature = 41, Conductivity = 42, Salinity = 43 };

发送json：txtSendMsg.Text = JsonConvert.SerializeObject(o);

/// <summary>

/// 通信所用的指令

/// </summary>

public enum SocketCmd : int

{

/// <summary> 开始自动控温 </summary>

AutoStart = 0,

/// <summary> 停止 </summary>

Stop,

/// <summary> 测量完成 </summary>

Finished,

/// <summary> 测试 Idx </summary>

TestId,

/// <summary> 设备状态 </summary>

DeviceState,

/// <summary> 传感器部件状态 </summary>

DeviceStatus,

/// <summary> 未知 </summary>

Unknown

}

public class SocketStatusMessage : SocketCmdMessage

{

public SocketStatusMessage() : base(SocketCmd.DeviceStatus) { }

public float ZhucaoTemperature = 35.0000f;//主槽设定温度

public float ZhucaoTemperatureReal = 35.0000f;//主槽当前温度

public int ZhucaoPower = 0; //主槽加热功率

public float FucaoTemperature = 34.0000f; //辅槽设定温度

public float FucaoTemperatureReal = 34.0000f; //辅槽当前温度

public int FucaoPower = 0; //辅槽加热功率

public bool Zongdianyuan = false; //总电源开关

public bool Zhucaozhileng = false; //主槽制冷开关

public bool Zhucaokongwen = false; //主槽控温开关

public bool Fucaokongwen = false; //辅槽控温开关

public bool Fucaozhileng = false; //辅槽制冷开关

public bool Fucaoxunhuan = false; //辅槽循环开关

public bool Fucaokuaileng = false; //辅槽快冷开关

public bool Zhucaokuaileng = false; //主槽快冷开关

public DateTime DataTime = DateTime.Now;//数据采集时间

public double Temperature = double.NaN;//标准温度

public double Conductivity = double.NaN;//标准电导率

public double Salinity = double.NaN;//标准盐度

}