

1 新增內容

1.1 SysDefine.cs

1.1.1 新增兩個定義

```
public static bool InitialOk; //判斷 Initial 是否完成重  
public static bool Lotend; //結批開關  
public static bool LotendOk; //結批完成  
public static bool ErrorStop; //錯誤停機  
public static bool HMIReady; //HMI 啟動完成  
public static bool HMIClose; //HMI 關閉
```

```
public static byte OEERPFfrom = 0;  
public static string OEERPTtype = string.Empty;  
private static bool mSysRun; //系統運行中  
public static bool SysRun //系統運行中  
{  
    get { return mSysRun; }  
    set
```

```
public static bool InitialOk; //判斷 Initial 是否完成  
public static bool Lotend; //結批開關  
public static bool LotendOk; //結批完成  
public static bool ErrorStop; //錯誤停機  
public static bool HMIReady; //HMI 啟動完成  
public static bool HMIClose; //HMI 關閉
```

```
private static bool mSysRun; //系統運行中  
public static bool SysRun //系統運行中  
{  
    get { return mSysRun; }  
    set
```

1.2 內參設定

1.2.1 內參設定.cs

1.2.1.1 五個新增

1. Using System.IO;

2. 內參設定呼叫: ShowOeeType();

<pre>using System.Drawing; using System.Windows.Forms; using KCSDK; using CommonObj; using System.IO; namespace ProVSimpleProject { public partial class 內參設定 : BaseOptionF { private List<Button> btnlist = new List<Button>(); public 內參設定() : base("Setup") { InitializeComponent(); TopLevel = true; FormBorderStyle = System.Windows.Forms.FormBorderStyleS ControlBox = false; StartPosition = FormStartPosition.CenterScreen; ShowOeeType(); } public override void Initial()</pre>	<pre>using System.Drawing; using System.Windows.Forms; using KCSDK; using CommonObj; namespace ProVSimpleProject { public partial class 內參設定 : BaseOptionF { private List<Button> btnlist = new List<Button>(); public 內參設定() : base("Setup") { InitializeComponent(); TopLevel = true; FormBorderStyle = System.Windows.Forms.FormBorderStyleS ControlBox = false; StartPosition = FormStartPosition.CenterScreen; } public override void Initial()</pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

3. 三個功能函示:

cboxOEEReportSel_SelectedIndexChanged()

ShowOeeType()

dcbOeeDataType_SelectedIndexChanged()

<pre> SYSPara.logDB.WriteLaserPowerCurve(1stPWR); } private void cboxOEEReportSel_SelectedIndexChanged(object sender, EventArgs e) { SYSPara.OEERPFFrom = (byte)dcbOeeSource.SelectedIndex; } private void ShowOeeType() { string ModulePath = Directory.GetCurrentDirectory(); DirectoryInfo di = new DirectoryInfo(ModulePath); bool isProductExist = false; foreach (var file in di.GetFiles()) { if (file.Extension == ".dll") { foreach (string item in dcbOeeDataType.Items) { if (!item.Contains(file.Name.TrimEnd(file.Extension))) continue; else isProductExist = true; } } if (!isProductExist) { string nameProduct = file.Name.TrimEnd(file.Extension); dcbOeeDataType.Items.Add(nameProduct); } } } private void dcbOeeDataType_SelectedIndexChanged(object sender, EventArgs e) { SYSPara.OEERPTType = dcbOeeDataType.Text; } }</pre>	<pre> SYSPara.logDB.WriteLaserPowerCurve(1stPWR); } }</pre>
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------

1.2.2 內參設定.designer.cs

新增紅框部分，可以從新的 SimpleProjecy 複製過來，選擇變更事件要設定。

內參設定

編輯

儲存

取消

結束

應用

系統

IO描述檔

OEE Report

報告型別選擇

資料型別選擇

設備負載時間

22

hrs

錯誤計算設定

報告輸出路徑

D:\

...

項目	名稱	統計方式
----	----	------

新增統計項目

刪除統計項目

1.3 HistoryLogger.cs

1.3.1 5 個新增

1.3.1.1 System.Json

C:\...stanley\Desktop\ProVSimpleProject\ProVSimpleProject\FormSet\HistoryLogger.cs	C:\...d9b3e66dc7add93402116190cf748\ProVSimpleProject\FormSet\HistoryLogger.cs
<pre>using System; using System.Collections.Generic; using System.ComponentModel; using System.Data; using System.Drawing; using System.Linq; using System.Text; using System.Windows.Forms; using ProVLib; using System.IO; using System.Runtime.Serialization.Formatters.Binary; using System.Runtime.InteropServices; using System.Diagnostics; using Microsoft.Win32; using Newtonsoft.Json; namespace ProVSimpleProject</pre>	<pre>using System; using System.Collections.Generic; using System.ComponentModel; using System.Data; using System.Drawing; using System.Linq; using System.Text; using System.Windows.Forms; using ProVLib; using System.IO; using System.Runtime.Serialization.Formatters.Binary; using System.Runtime.InteropServices; using System.Diagnostics; using Microsoft.Win32; namespace ProVSimpleProject</pre>

1.3.1.2 資料庫儲存 SPC 類

1. oriMSG

2. 判別為 SPC，資料庫加入內容

Left Side Code	Right Side Code
<pre>List<String> MsgList = new List<string>(); private string oriMSG = string.Empty; private void AddMsg(string mType, string mMsg) { if (TypeList.IndexOf(mType) == -1) TypeList.Add(mType); //+ By Max 20180424 mMsg = GetDate() + '\t' + GetTime() + '\t' + mType lock (locker) { MsgList.Add(mMsg); } //+ By Max 20200204 For LogAnalyzer try { //+ By Stanley 20250326 For LogAnalyzer if (mType == "SPC") { oriMSG = mMsg; int idST = oriMSG.IndexOf("ST="); oriMSG = oriMSG.Substring(idST, oriMSG.Len mMsg = String.Format("{0}\t{1}\t{2}\t{3}\t" DateTime.Now.ToString("yyyy/MM/dd"), DateT } SYSPara.logDB.LogSayDb(mMsg); } }</pre>	<pre>List<String> MsgList = new List<string>(); private void AddMsg(string mType, string mMsg) { if (TypeList.IndexOf(mType) == -1) TypeList.Add(mType); //+ By Max 20180424 mMsg = GetDate() + '\t' + GetTime() + '\t' + mType lock (locker) { MsgList.Add(mMsg); } //+ By Max 20200204 For LogAnalyzer try { SYSPara.logDB.LogSayDb(mMsg); } }</pre>

1.3.1.3 Log 分析點擊事件修改

2025年 4月14日 ▾	日 ▾
2025年 4月14日 ▾	LOG分析

```
private void cycleWidget1_AfterSummary(object DataList)
{
    //ReportF frm = new ReportF();
    //frm.Initial(DataList);
    //frm.ShowDialog();

    //+ By Max, v4.1.0.26, OEEPerformance
    try
    {
        OEE oee = new OEE();
        oee.Array = DataList;
        oee.OEERPTType = SYSPara.OEERPTType;
        oee.OEEFrom = SYSPara.OEERPFFrom;

        string json = JsonConvert.SerializeObject(oee);

        ProcessStartInfo startInfo = new ProcessStartInfo
        {
            FileName = Directory.GetCurrentDirectory() + "\\OEEperformance.exe",
            Arguments = json
        };
        Process.Start(startInfo);
    }
    catch (Exception e)
    {
        MessageBox.Show(e.Message);
    }
}
```

1.3.1.4 OEE 類別

加入 Class OEE

```
String str = ex.Message;
}
}
}
public class OEE
{
    private object obj = null;
    public object Array
    {
        get { return obj; }
        set { obj = value; }
    }

    private string type = string.Empty;
    public string OEERPTType
    {
        get { return type; }
        set { type = value; }
    }

    private short oeeFrom = 0;
    public short OEEFrom
    {
        get { return oeeFrom; }
        set { oeeFrom = value; }
    }
}
```

2 新增 DLL

2.1 新增 5 個 dll,修改一個 dll

位置=> D:\ProVSimpleProject\ProVSimpleProject\bin\ (同專案 exe 位置)

BouncyCastle.Crypto.dll

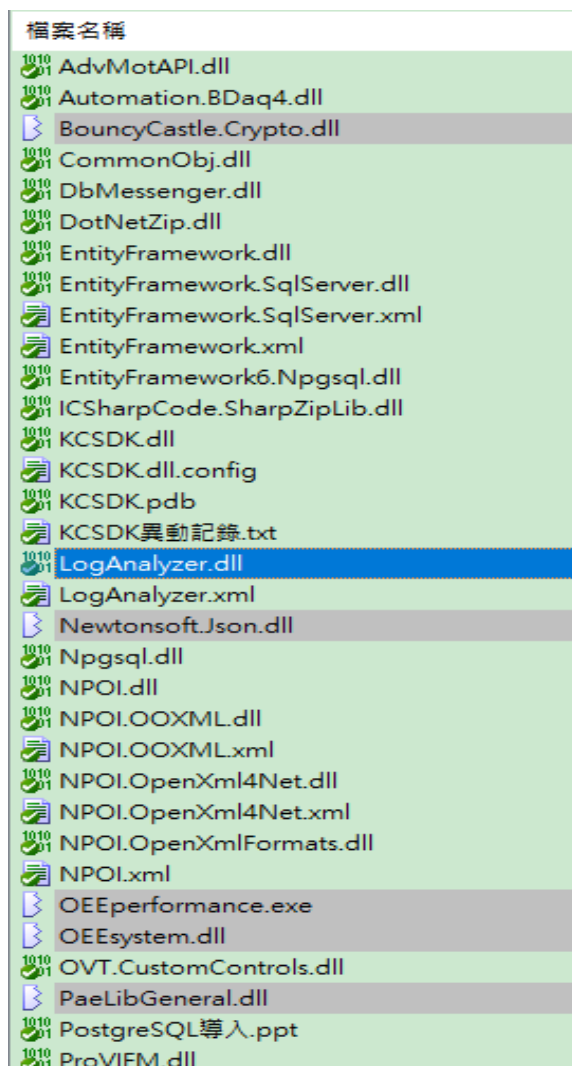
LogAnalyzer.dll (覆蓋)

Newtonsoft.Json.dll

OEESystem.dll

PaeLibGeneral.dll

Ude.NetStandard.dll (判斷 Log 編碼為 UTF8 / Big5)

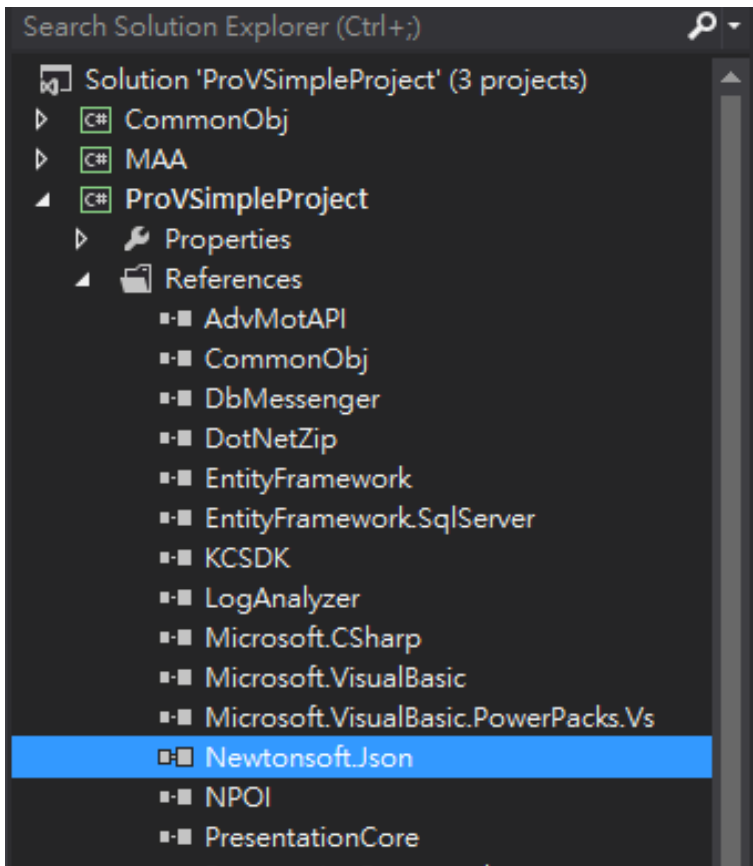


2.2 新增 OEEperformance.exe

一樣放在 D:\ProVSimpleProject\ProVSimpleProject\bin\資料夾內(如上圖)

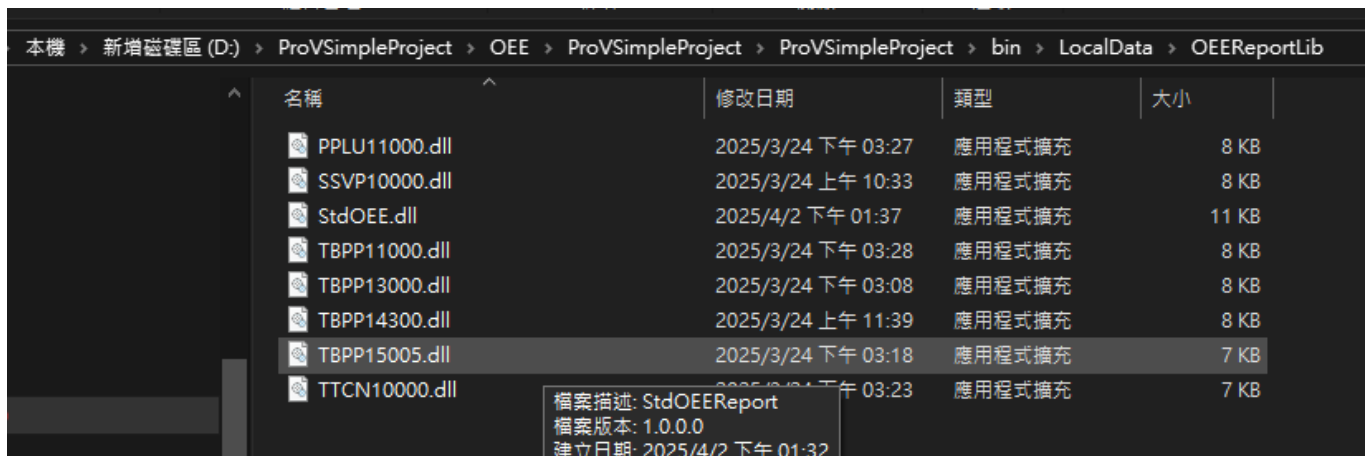
2.3 加入參考

將 Newtonsoft.Json.dll 加入參考



3 加入 OEE library

..\SimpleProject\ProVSimpleProject\bin\LocalData\下新增 OEEReportLib 資料夾，裡面放 OEE 專用 lib 如下圖。除了標準做法(StdOEE.dll)，其餘 dll 是由附上的 cs 製成。



本機 > 新增磁碟區 (D:) > ProVSimpleProject > OEE > ProVSimpleProject > ProVSimpleProject > bin > LocalData > OEEReportLib				
名稱	修改日期	類型	大小	
PPLU11000.dll	2025/3/24 下午 03:27	應用程式擴充	8 KB	
SSVP10000.dll	2025/3/24 上午 10:33	應用程式擴充	8 KB	
StdOEE.dll	2025/4/2 下午 01:37	應用程式擴充	11 KB	
TBPP11000.dll	2025/3/24 下午 03:28	應用程式擴充	8 KB	
TBPP13000.dll	2025/3/24 下午 03:08	應用程式擴充	8 KB	
TBPP14300.dll	2025/3/24 上午 11:39	應用程式擴充	8 KB	
TBPP15005.dll	2025/3/24 下午 03:18	應用程式擴充	7 KB	
TTCN10000.dll	2025/3/24 下午 03:23	應用程式擴充	7 KB	

檔案描述: StdOEEReport
檔案版本: 1.0.0.0
建立日期: 2025/4/2 下午 01:32