Motion Studio 教程 4 人机界面开发 - 为教程 3 程序加上人机界面

说明

本教程目的是说明如何使用 MS HMI.Net 控件在微软 Visual Studio 里快速实现人机界面. 本教程将设计下图左的界面来与教程 3 程序互动,来实现运行 P1P2 与停止功能.开发过程大概需 6 分钟. 废话不多说,开始进入正题! (Note: 此 HMI 界面调适时需要同时运行教程 3 程序)



MS HMI.Net 简介

HMI.NET 是界面控件,透过简单配置设定来与 Motion Runtime 的数据交互。通过使用这些封装的控件,上位机 开发者可以快速地完成界面布局与组态。MS HMI.NET 控件所关联 Motion Studio 状态信息,全部基于标签 Tag, 即在 MSHMI 内部的 TagServer 标签 服务所提供的标签状态信息。标签服务内部调用 MSConnection 驱动进 行下位数据的访问。整体的系统架 构如下图所示。(详细请参考 MS HMI.Net 手册)



工具/原料

- Motion Studio v1.9.5.1
- Motion Runtime v1.9.5.1
- Visual Studio 2015 并安装 MS HMI.Net v1.9.5.1

主步骤

步骤 1 - 规划人机界面 步骤 2 - 将教程 3 项目的 VR 加入 Modbus 位址 步骤 3 - 以 HMI.Net 模板创建 Visual Studio C#专案 步骤 4 - 使用 HMI.Net 控件设计画面

- 步骤 5 修改 HMI.Net 元件 Tag 属性, 建立连结关系
- 步骤 6-调适/运行

步骤1-规划人机界面

(1). 在开始前,先说明 HMI 画面每个元件对应的 Tag. 设计的画面中共有: 2个按钮, 1个状态与 2个坐标数值.

- 1. 系统状态: 将对应到教程 3 中的 VR(0)
- 2. 2个按钮: 将对应到教程 3 中的 VR(3)与 VR(4)虚拟按钮(启动, 停止).
- 3. X,Y坐标数值: 与1,2不同, 在 Motion Runtime 已內建各轴坐标数值(*). 因此我们要做的是建立连结到 Runtime 系统内的 X,Y 坐标 Tag, 就可以显示坐标值.

X 坐标:Axis0-> DPOS

Y 坐标:Axis1-> DPOS

所有的关系都厘清后,我们就可以进行下一步骤!

(*)Motion Runtime 内建的 Tag 说明请参考 MAS Modbus 使用手册



步骤 2 - 将教程 3 项目的 5 个 VR 加入 Modbus 位址

因为我们需要将 VR 变量与 HMI 建立连结,因此需要将 VR 加上 Modbus 位址,并下载到控制器

- 1. 打开 Motion Studio 并开启教程 3 项目,
- 2. 开启 VR 表工具, 将项目原有的 5 个 VR 加入 Modbus 位址, VR(0): 40001- VR(4):40009 (如下图),
- 3. 点击上方的 [下载到控制器] 💷 按钮, 将 Tag 设定档下载到控制器

(注意: 一定要填入 Modbus 位址, 并下载后, 步骤 5 才能看到 VR(0)-VR(4) Tag 描述)

VR表					Ψ×
0 % 0	<	🖾 🖸	🖾 🖾		
名称	当前值	1 <u>2</u> *	初始	Modbus	数据类型
范围[0-1			Ν.		
VR(0)	0	MODE	0	40001	BIT 16 U 🝷
VR(1)	0	RunStep	0	40003	BIT 16 U 🝷
VR(2)	0	StopStep	10	40005	BIT 16 U 🝷
VR(3)	0	RunButton	0	40007	BIT 16 U 🝷
VR(4)	0	StopButton	0	40009	BIT 16 U 🝷
VR(5)	0		0		BIT 16 U 🝷
VR(6)	0		0		BIT 16 U
VR(7)	0		0		BIT 16 U 🝷
UD (O)					DIT 46 U

步骤 3 - 创建 Visual Studio C#专案

(1). 启动 Visual Studio 后, 点选新增项目, 我们可以看到在 Visual C#下有 MS HMI Application for C Sharp. 请选择此模板, 输入名称后, 建立新项目



(2). 确定后可以看到新项目已经建立. 此时编辑视窗是空白. 先别急, 这时再点击[Form1.cs], 然后编辑视窗就会出现 Form1 视窗. 并会再左下角看到 msTagServer1 组件.

刘 HMI for P1P2 - Microsoft Visual Studio 🛛 💙 快速图	(Ctrl+Q) 👂 🗕 🗖 🗙
檔案(F) 編輯(E) 檢視(V) 專案(P) 建置(B) 偵錯(D) 小組(M) 格式(O) 工具(T) 測試(S) 分析(N) 視窗(W) 說明(H) 登入 🎴
🕺 O マ O 🏠 マ 🏠 💾 🔐 ヴ マ 🤇 マ Debug 🝷 Any CPU 🛛 🝷 開始 マ 声 🚽 🕆 🃁 =	÷====================================
<mark>Form1.cs[設計] 中 ×</mark>	方案總管 ▼ ♀ ×
海 端 Form1	◎ ◎ ☆ ☵ - ఀ • ≒ ♂ ₽ ₪ "
	授辱万案總管 (Ctrl+;)
	☑ 方案 'HMI for P1P2' (1 專案)
	 Properties
	▶ ■■ 參考
	P app.config
	. ▷ Form1.cs
	MSHMI.ico
	v c. Program.cs
	方案總管 Team Explorer
	· 屬性 · ₽ ×
	Form1 System.Windows.Forms.Form
	🔠 🖫 🗲 🔎
	FormBorderStyle Sizable
	RightToLeft No
	Text Form1
	UseWaitCursor False 🔻
	Text
😡 msTagServer1	與控制項關聯的文字。
	↑ 新增至原始檔控制 🔺

这时候点击 [工具箱](上图左侧), 会出现 MSHMI 控件在工具箱里(下图)

Note: 如果未出现 MSHMI 控件, 请参考本文最后 - 工具箱没有出现 HMI.Net 控制处理方式

工具箱 ▼ 🕂 ×	Form1.cs [设计]*
搜索工具箱 🖉 🗸	
▷ 所有 Windows 窗体	🖷 Form1 🗖 🗖 🔀
▷ 公共控件	
▷ 容器	
▷ 菜单和工具栏	
▷ 数据	
▷ 组件	
▷打印	
▷ 对话框	
▷ WPF 互操作性	
MSHMI	
指针	
AxHomeMode	
AxJogControl	
MS3DPath	
MSAlarmConfig	
MSAlarmView	
MSDataTable	
MSDataTableList	
MSOperationLogCon	
MSOperationLogView	
MSParameter	msTagServer1
MSTagSenver	

步骤 4 - 使用 HMI.Net 控件设计画面

(1). 使用 HMI.Net 控件设计画面

这时候可以开始设计画面,我们依照下图,

- 1. 从工具箱拉入 3 个 TagNumTextBox 与 2 个 TagButton.
- 2. 并将 TagNumTextBox, TagButton 的字形大小改为 16, 方便观看.



3. 并再拉入 3 个 Label 元件,也同时将 Label 的字形大小改为 16,方便观看

工具箱 ▼	ч ×	Form1.cs [设计]* 🖕 🗙		- [解决方案资源管理器	器 🗕 🗕 구 🖓	×
搜索工具箱	ρ-				G O 🟠 🛗 -	'o • 与 🖒 🖻	**
▷ 所有 Windows 窗体 ┛ 公共控件		Porm1		J.		管理器(Ctrl+;)	-
指针				ь	☑ 解决方案"HM	II for P1P2"(1 个项目)	
Button	- 11	p		н	▲ C# HMI for P	P1P2	11
CheckBox	- 11	label1			Proper	rties	
CheckedListBox	- 11				▶■引用		
ComboBox	- 11	1-1-10			♥」 app.cc 磁 山 亡 安 恣 酒 答 理 9	DNTIG 界 同时次适答理职	
DateTimePicker		label2			附次刀朱贝际官庄(11 四队负尿管理器	
Label		label3			属性	▼ ₽ >	×
LinkLabel		140015			label1 System.Wir	ndows.Forms.Label	
ListBox					🗄 💱 🖓 🗲 🎾		
ListView					AccessibleRole	Default	•
MaskedTextBox				E	Appearance	_	
MonthCalendar		tagButton1	tagButton2		BackColor	Control	11
Notifylcon					BorderStyle	None	
NumericUpDown					ElatStyle	Standard	
PictureBox					T Font	新細明體, 16,125pt	
ProgressBar				_	ForeColor	ControlText	
RadioButton		msTagServer1			Image	(none)	
RichTextRox				_	ImageAlign	MiddleCenter	-

- 4. 并将 3 个 Label, 与 2 个 TagButton 的 Text 属性, 分别改成我们期望的名称
 - "Label1" → "系统模式", "Label2" → "X", "Label3" → "Y"
 - "tagButton1" → "运行 P1P2", "tagButton2" → "停止"



步骤 5 - 修改 HMI.Net 元件 Tag 属性, 建立连结关系

(1). 设定 系统模式 tagNumTextBox1 元件的 Tag, 建立连结关系

- 1. 用鼠标选择 系统模式 tagNumTextBox1, 此时会出现对应的属性视窗
- 2. 选择用类别来排列属性
- 3. 找到 MAS 类别
- 4. 用鼠标点选 MasTagGroup



5. 在弹出的标签选择对话框中,选择范围[0-10],再选择 MODE,点选[确定]完成设定。 Note:这些 Tag 就是我们在步骤 2 所导出的内容.



(2). 设定 X, Y 等 2 个元件的 Tag, 建立连结关系

同上一步骤,用鼠标选择,X or Y 元件,并用鼠标点选 MasTagGroup,并分别设定

X:选择 Axis0->DPOS Tag

Y:选择Axis1->DPOS Tag

Note: DPOS 是理论位置



(3). 设定[停止按钮] 与[运行 P1P2 按钮] 的 Tag, 建立连结关系

1. 同上一步骤,用鼠标选择,[停止按钮]与[运行 P1P2 按钮]元件,并用鼠标点选 MasTagGroup1

Form1.cs [设计]* + × 🔹	解决方案资源管理器 ▼♀♀×
Form1	 ○ ○ 습 급 → 1 ○ · 与 ○ 副 2g素解決方案资源管理器(Ctrl+;)
系統模式	
X Y	属性 ▼ ₽ × tagButton1 Advantech.MSHMI.TagButto • 副型↓ ① ゲ ル
· 運行P1P2 停止	 MAS ▲ EnableWriteLog False TipBindingTag MAS_Tag1 MAS Security SecurityEnable False
	SecurityLevel Level0 SecurityType Operator MAS_Tag1 MasTagGroup1
Dim msTagServer1	Masiagivamei MouseDown1Log MouseDownValu 1

并分别设定[停止按钮] 与[运行 P1P2 按钮]元件
 [停止按钮]:选择范围[0-10]->StopButton Tag
 [运行 P1P2 按钮]:选择范围[0-10]->RunButton Tag

ب	Select Tag
Form1	■HMI内部Int变量 ●HMI内部String变量
	• AXISO
系統模式	
	±AXIS3
X	■ Robot
	t Robot0
Y	
	€ DO
	Motion Command
· 揮行P1P2 · 停止	三范围[0-10]
	MODE
	-RunStep
	StopStep
	KunButton
e.	StopButton

(4). 我们已经完成所有步骤, 可以编译运行了!

是的,不需要编写代码,就可以完成画面开发!您可以点选 VS 上的启动按钮运行 HMI 程序.

步骤 6-调适/运行

(1).运行 Visual Studio HMI 程序我们可以看到如下画面. 这时候点击 [运行 P1P2], [停止] 按钮会没有任何 反应, 请不用担心, 这是因为 Motion Studio 程序尚未运行.

HMI for P1P2 - Microsoft Vis	ual Studio				X
文件(F) 编辑(E) 视图(V) 项目	∃(P) 生成(B) 调词	t(D) 团队(M)	格式(O)	工具(T)	测试(S) 分
🛛 🗢 🗢 🛛 🎌 🕶 🔛 📸 🖊	🤈 🗸 🦿 🚽 Debug	 Any CPU 		启动 🔹 🍃	• • =
	1	Ļ			
	🔜 Form1	- 0	×		
	1. 65 H	0			
	<i>杀</i> 须 保 式				
	Х	0			
	Y	0			
	運行P1P2	停止			

(2). 请用 Motion Studio 开启教程 3 项目, 并点选 [编译] → [运行],

ProjectDefine.bi	ProjectDefine.bas	Task1.bas		→ ×
8 % 6 6 ♠	🔺 🗐 🖪 🔯 () वांस	ā · 📳 🕰 🤇	🖸 🔿 🔘 🜻 🖉 🔳	2

(3). 这时候在点击 [运行 P1P2] 按钮,可以发现 HMI 可以正常动作了!
 同时可以看到模式栏位会由 0(Ready) 变化成 1(Run).
 移动过程中,如果按下[停止]按钮,会由 1(Run) 变化成 2(Stop),再变化成 0(Ready)



Note -工具箱没有出现 HMI.Net 的处理方式

1. 建立选项卡: 在工具箱中按下鼠标右键, 选择[添加选项卡], 并输入 MSHMI 名称. 在建立的 MSHMI 选项卡上, 点击鼠标右键, 并选择 [选择项]



2. 导入控件到工具箱

选择 [选择项]后会出现 [选择工具箱] 视窗, 这时请选择[浏览], 会跳出选择档案视窗, 这时请到 C:\advantech\MSHMI\Library 下, 并选择 AdvMSHMI.dll



这时候可以看到元件已经导入到列表中,请再点选 [确定] 后,我们可以看到 控件已经出现再工具箱中.

莈	E择工具	箱项			?	×	工具箱	▼ ₽ 3	×
Γ	.NET F	ramework 组件 COM 组	组件 WPF 组件 通用 Windows 组件				搜索工具箱	P	*
							▷ WPF 互操作性		•
	_	名称 ▲	命名空间	程序集名称		^	MSHMI		
		AxHomeMode	Advantech.MSHMI	AdvMSHMI			指针		
	-	AxJogControl	Advantech.MSHMI	AdvMSHMI			AvillemeMede		
		Backgroundworker	System.Componentiviodei	System			AxHomewode		
		BehaviorEditorPart	System.Web.UI.WebControls.WebParts	System.Web			AxJogControl		
	✓	BindingNavigator	System.Windows.Forms	System.Windows.Forms			MS3DPath		
		BindingSource	System.Windows.Forms	System.Windows.Forms			MSAlarmConfi	q	
		BulletedList	System.Web.UI.WebControls	System.Web			MSAlarmView	5	
		Button	System. Web.UI. WebControis	System.web			MCDataTable		
		Button	Misroach Visual Pasia Compatibility V/PC	System. windows.Forms			IVISDataTable		
		ButtonArray	Microsoft.VisualBasic.Compatibility.VB6	MICrosoft. VisualBasic.Co	mpatibility		MSDataTableL	ist	
	<	Calendar	System.web.UI.MobileControls	System.web.wobile	>		MSOperationL	.ogCon	
	筛洗	器(F):			清除(C)	MSOperationL	.ogView	
		lomeMode					MSParameter		
		语言: Invariant Land	guage (Invariant Country)		浏览(B)		MSTagServer		
	•	版本: 1.3.1.6					MSWorkOrder		
							Path		
							PathGraphViev	N	
				明正		(K)	PathGrid		

更多信息可到 Motion Studio 官网: http://mas.advantech.com.cn