

上海交通大学

学生实习报告

实习题目：AB 杯全国大学生自动化系统应用技术大赛

实习单位：上海交通大学罗克韦尔实验室

实习时间：2019/7/1 至 2019/7/31

学院(系)：电子信息与电气工程学院

专 业：自动化

学生姓名：陈思哲 学号：516021910038

2019 年 7 月 31 日

一、 实习单位

上海交通大学罗克韦尔实验室。

二、 实习目的

1. 了解 PTC 物联网公司的 ThingWorx Foundation / Connectivity 的前端制作方法。
2. 学习 Micro820 系列 PLC 和编程软件 CCW 及其应用。
3. 掌握 Studio5000 控制 CompactLogix 控制系统的方法。
4. 掌握 ThingWorx 前端，Micro820，CompactLogix 直接的通讯实现。

三、 实习任务

初赛实习任务

- (1) 使用参数技术软件公司（PTC）的 ThingWorx 工业物联网技术平台
- (2) 使用罗克韦尔自动化公司的 Micro800 系列 PLC 和其它自动化产品
- (3) 选择能够体现“安全”、“环保”、“节能”和“增效”可持续发展的工业应用场景
- (4) 选择使用其它的现场设备和被控对象
- (5) 提交作品的完整方案（pdf 格式），作品的 5 分钟演示视频（mp4 格式）

决赛实习任务（根据回忆记录）

- (1) 使用 ThingWorx Industrial Connectivity 软件连接 CompactLogix 控制器，并通过该软件监测 CompactLogix 控制器中的标签值
- (2) 使用 ThingWorx 平台监测 CompactLogix 控制器中的标签值
- (3) 使用 ThingWorx 平台监测 E+H 温度计所测量的实时温度值
在 ThingWorx 平台上监测和控制集成运动控制系统和同步皮带系统。
- (4) 控制同步皮带的运行和停止，并监测显示同步皮带的状态（运行或停止）
- (5) 控制同步皮带的正转和反转，并监测显示同步皮带的运行方向（正转或反转）
- (6) 改变同步皮带的运行速度，并监测显示同步皮带的实际运行速度
- (7) 监测显示同步皮带实际运行速度的历史记录曲线

四、 实习内容

初赛实习内容

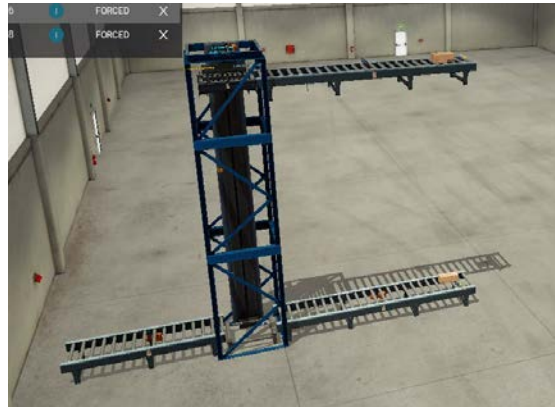
微信扫码查看视频 Demo →



项目综述

项目目标

我们意在利用 Connected Components Workbench 在 Micro800 上设计 PLC 控制器，模拟控制 FactoryIO 中的虚拟工厂。工作环境由上下两条水平传送带和一个升降电梯组成。两个物件从左侧进入系统，利用三个传感器的反馈信息，我们将第一个物件传送到下方传送带末端；第二个物件传送到上方传送带末端。在箱子的传送过程中，利用 PTC ThingWorx 实时显示箱子距离终点的相对坐标。在箱子运动状态发生变化时，向操作者发送邮件。



项目具体要求

因为本项目涉及的软件平台较多, 包括 Connected Components Workbench、FactoryIO、ThingWorx、Micro800 硬件设备, 所以 CCW 与 Micro800、CCW 与 FactoryIO、ThingWorx 与 Micro800、ThingWorx 与邮箱的通讯, 都是我们需要解决的问题。

解决通讯问题后, 接下来就是控制层面的实现。虚拟工厂中可以控制的器件有三个传送带的开关和电梯的开关。输入是三个传感器的箱子遮挡信息。我们需要设计一个合适的控制策略, 并用 PLC Ladder Diagram 编程实现。

完成了核心控制逻辑, 我们还要把控制的效果在 PTC ThingWorx 中可视化, 并且将工厂当前的情况 (即箱子距离目标位置的相对坐标) 实时显示出来, 并在关键节点发送提醒邮件到操作者指定邮箱。最后我们还要录制视频, 制作 PPT。

项目环境配置

Windows 10

Connected Components Workbench 12

FactoryIO 2.3.5

JDK x64 8

ThingWorx Industrial Connectivity 8.4

项目方案

设备连接与通讯

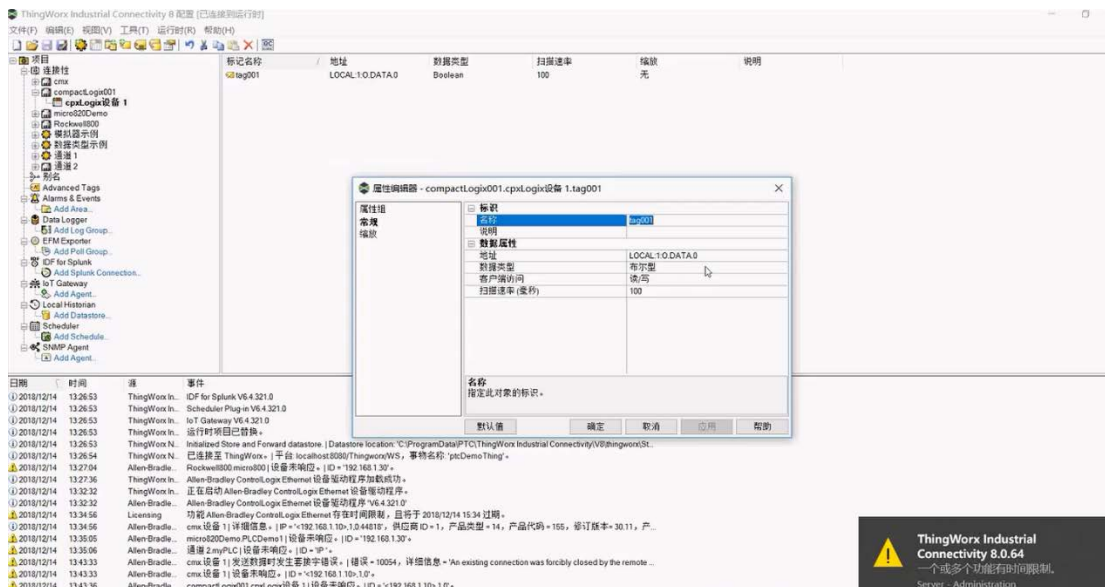
CCW 与 Micro800 连接上，我们使用的是 Ethernet。首先利用网线，在路由器上获得 PLC 的 Wifi 名称，然后使用电脑连接上此 Wifi。在 CCW 中选择 PLC 型号为 Micro800，然后连接设备。这样 CCW 编程的梯形图，即可通过 PLC 控制平台上的红绿灯。



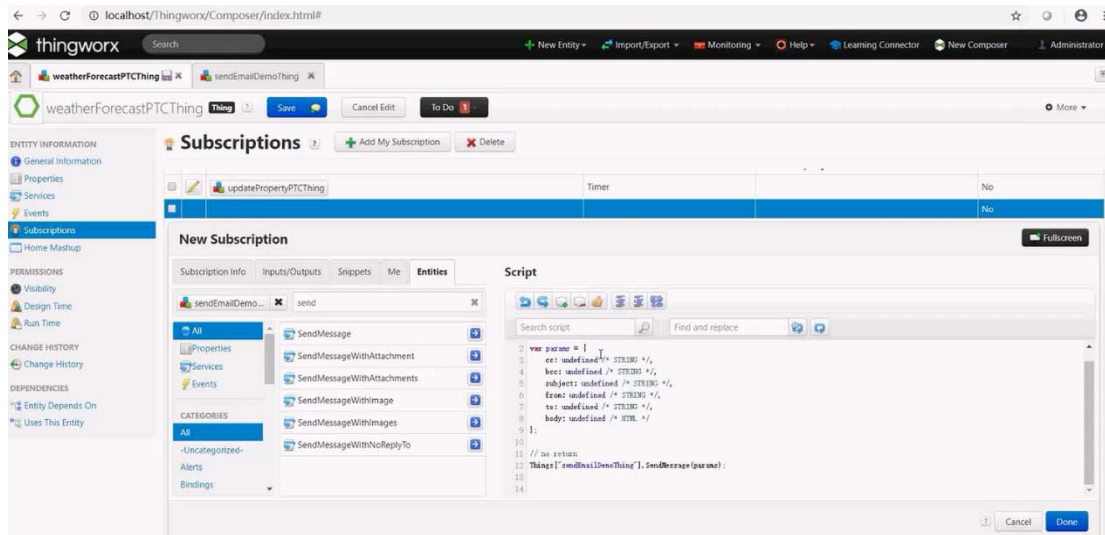
CCW 与 FactoryIO 的通讯上，我们在 FactoryIO 上定义好对应的输入输出变量的名称，并定义 Micro800 上的管脚。点击 connect，CCW 中的程序运行后，即通过 Micro800 控制 FactoryIO 虚拟工厂中的器件。



ThingWorx 与 Micro800 连接上,我们在 PTC 中新建一个通道,在此通道中新建一个 PLC 设备,选择对应的型号和 IP 地址,最后设置一个 tag 为变量的标记。这样 PTC 就可以获得 PLC 中的变量值。

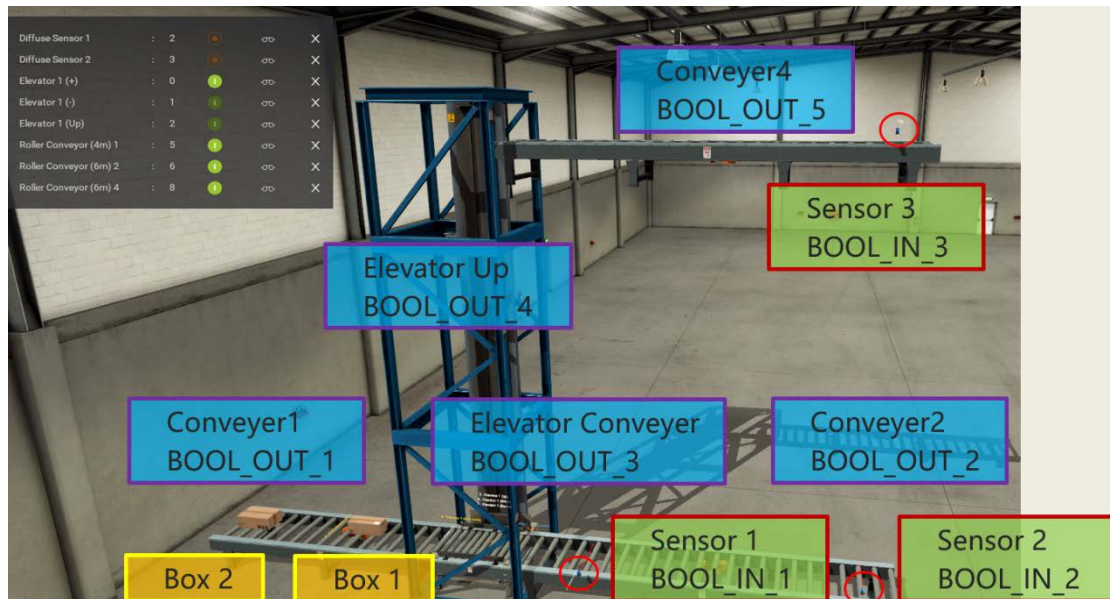


ThingWorx 与操作者邮箱通讯上,我们在 ThingWorx 建立一个 Thing 对象,导入 Mail Service 的 ThingTemplate,在 Configuration 中配制发件人邮箱地址及其密码。然后添加 Subscriptions,设置触发事件,导入之前建立的 Thing,设置收件人邮箱以及邮件内容。



至此,我们完成了所有的设备连接,通讯,测试工作。只要 PLC 控制逻辑正确, Micro800 就能正确执行,将值传给 FactoryIO 和 ThingWorx。FactoryIO 中的模拟工厂按预期进行控制, ThingWorx 中将控制的过程展现出来,并实时反馈目标的坐标,在合适的时机向指定邮箱发送邮件。

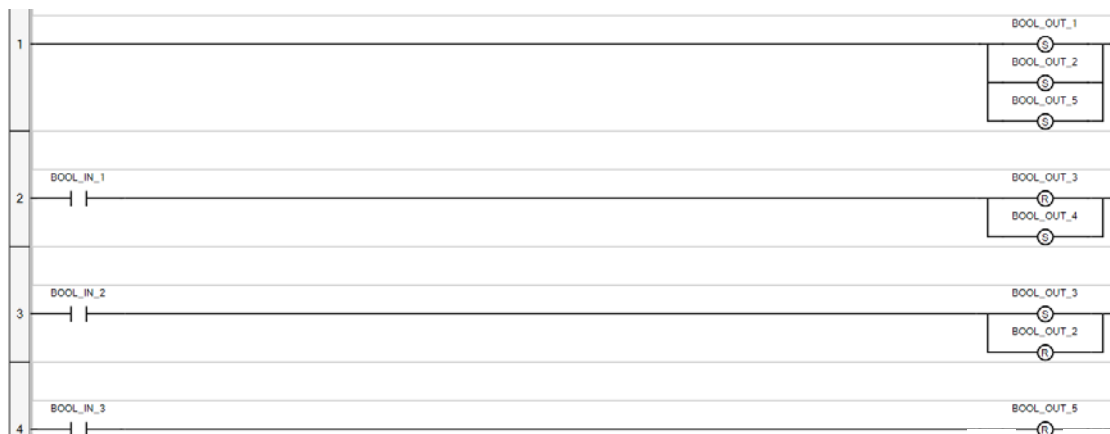
PLC 控制逻辑



在上一部分的描述中，我们在 FactoryIO 中定义了 Micro800 管脚的连接，在虚拟工厂中的器件对应关系如上图所示，以下介绍我们的控制逻辑。首先设置初始变量 Elevator Conveyer 为 1 和 Elevator Up 为 0。

	Name	Alias	Data Type	Dimension	Initial Value
	BOOL_OUT_1		BOOL		
	BOOL_OUT_2		BOOL		

三个传送带保持运转模式。适当设置 Sensor 1 的位置，使之与电梯的距离和电梯传送带的长度相当。当 Sensor 1 检测到 Box 1 时，Box 2 刚好在电梯上，设置电梯向上并且电梯传送带不运转。当 Sensor 2 检测到 Box 1 时，Box 2 已经被电梯传送到顶端，这时停止 Conveyer 2 防止 Box 1 掉落，同时启动 Elevator Conveyer，Box 2 进入 Conveyer 3。当 Sensor 3 检测到 Box 2 时，停止 Conveyer 3，防止 Box 2 掉落。具体的梯形图如下。



控制效果可视化

首先在 ThingWorx 里创建一个应用，然后创建一个 entity 作为实体模版，将其添加到一个新的项目之中。在新项目中，从 Thing Template 中添加 Thing，为其添加四个属性，分别为两个箱子相对于终点的 x, z 坐标。

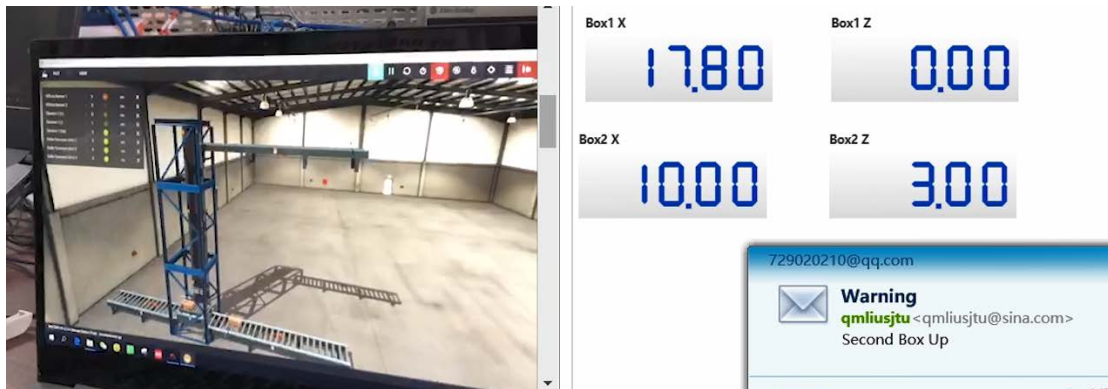
接着，添加一个 Service，不停从 PLC 获取变量值，将其显示在界面中。我们在 PTC 中新建一个通道，在此通道中新建一个 PLC 设备，选择对应的型号和 IP 地址，最后设置一个 tag 为变量的标记。这样 PTC 就可以获得 PLC 中的变量值。

这里需要创建一个 Timer Thing，设置时间间隔为 1s，也就是每 1s 触发一次计时器中断。在主程序的 Thing 中，订阅 Timer Thing，设置执行上面定义的服务。

为了记录历史数据，我们需要创建一个 Value Stream，将其插入主程序的 Thing 中。当任意一个箱子，变化的坐标发生改变时（比如从 x 坐标变化变成 z 坐标变化），触发发送邮件事件。

ThingWorx 与操作者邮箱通讯上，我们在 ThingWorx 建立一个 Thing，导入 Mail Service 的 ThingTemplate，在 Configuration 中配制发件人邮箱地址及其密码。然后添加 Subscriptions，设置触发事件，导入之前建立的 Thing，设置收件人邮箱以及邮件内容。

在界面设置上，创建一个 Mashup，设置合适的布局 layout。左侧通过虚拟工厂软件的画面插件将工业现场展现在主界面中。右侧放置四个 LED 显示屏，显示四个坐标值，并启动自动刷新功能。



至此，我们成功利用 CCW 在 Micro800 上设计 PLC 控制器，模拟控制 FactoryIO 中的虚拟工厂。利用传感器反馈信息，将两个箱子分别传送到上下两个传送带末端。在箱子的传送过程中，在 PTC ThingWorx 界面上实时显示箱子距离终点的相对坐标。在箱子运动方向发生变化时，向操作者发送邮件提示。具体演示效果见展示视频。

复赛实习内容（根据回忆记录操作细节）

1. 使用 Connectivity 通过 Studio 5000 连接 CompactLogix 控制器，并监测控制器标签值。
 - (1) 在设备箱中，将右上角的开关闸向上拉，红色开关打开，红色按钮向上旋开。连接以太网网线到电脑，配制以太网 IP=192.168.1.xxx。打开 Rslinx Classic 检测设备的 IP 地址，2198-H008-ERS 是伺服驱动器（可右键查看版本），1769-L36ERM 是 CompactLogix，PanelView 为无用的显示屏，剩下一个设备为 Micro820 PLC。
 - (2) 在 Studio 5000 中，创建一个 Project，设备为 1769-L36ERM，Revision=32，Security

Authority=No Protection。

- (3) 在 Studio 5000 中, 左侧最上方 Controller 设置, Date/Time-Enable Time Synchronization。可以在 Controller-Controller Tags 里查看全局变量 (main 为局部变量)。设置一个 Float_SJTU 的 REAL 型全局变量, 并且任意给该标签设定一个初始值。
- (4) 在 Studio 5000 中, 第一次下载需要设置, Communications-Who Active-CompactLogix-Download, 以后直接 download 就行。
- (5) 在 Connectivity 中, 右键连接性, 创建通道, 类型为 Allen-Bradley ControlLogix Ethernet (其他默认), 添加设备 CompactLogix 5300, ID= CompactLogix 5300 的 IP,1,0
- (6) 在 Connectivity 中, 在对应设备下新建 tag, 地址为 Studio5000 中的变量名 Float_SJTU, 设置数据类型, 点击“OPC Quick Client”界面监测该标签。

2. 使用 ThingWorx 平台监测 CompactLogix 控制器中的标签值。

- (1) 在 ThingWorx 里, 新建 Application Key, 设置 User name reference=Administrator, Expiration time 很长。
- (2) 在 Thingworx 里, 新建 Thing, 名称为 communication, 模版为 IndustrialGateway。
- (3) 在 Connectivity 里, 项目-属性-Thingworx, 按如下图配制。连接成功后 Thingworx 里有 connected 的符号。



- (4) 在 Thingworx 里, 点开 communication-discover, 将所有变量选中, create entity, 选择 remote thing, 创建导入变量的 entity get_value。
- (5) 在 Thingworx 里, 创建一个 mashup, 添加 LCD, 右侧 Data 新建 get_value 的 getProperties, mashup loaded。
- (6) 在 Thingworx 里, 将 getProperties 需要的 value 箭头指向 LCD 显示屏, 选择 Data。

3. 使用 ThingWorx 平台监测 E+H 温度计所测量的实时温度值。

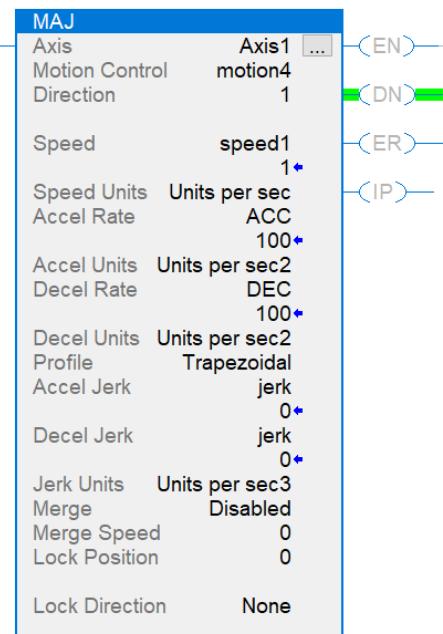
- (1) 在 CCW 中, 创建一个项目, 选择设备为 Micro820 2080-LC20-20QBB, 在设备上添加模拟模块 2080-IF2, 注意每次下载时不可以更新以太网设置。
- (2) 在 CCW 中, 左侧程序右键添加梯形图, 设置 L_IO_P1_AI_00->ANY_TO_REA->/262.14->-100=temperature。
- (3) 在 Studio5000 中, 设置两台 PLC 通讯。MSG 模块变量 t 的 t.en 为反向开关的触发变量。

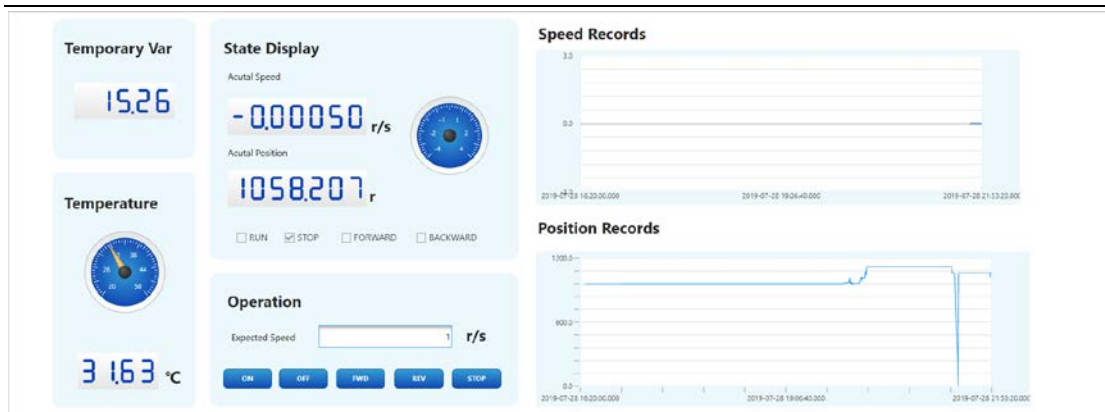
配制 MSG 模块 Configuration-Message Type=CIP Data Table Read, Source Element=CCW 中全局变量 TEMP, Destination Element=Studio 5000 全局变量 temperature。Communication-Path=2, Micro820 IP。

- (4) 在 Connectivity 里, 在对应设备下新建 tag, 地址为 Studio5000 中的变量名 temperature, 设置数据类型, 点击。
- (5) 在 Thingworx 中, 因为已经有了一个 entity, 所以直接在 get_value 里 add 一个 property, 选择来源为 remote bound, add address 里设置关联上面的 tag。
- (6) 在 mashup 中, 添加 gauge/LCD, 右侧 Data 新建, entity 为 get_value 的 getProperties, mashup loaded 打勾。

4~7. 用 Thingworx 同步控制/监视皮带的运行和停止, 正转和反转, 运行速度。监视皮带实际运行速度历史曲线。

- (1) 在 Studio 5000 中, I/O Ethernet 右键 New Module, 选择伺服驱动器设备 2198-H008-ERS, General 里改变其版本和 IP, Power 里设置 Voltage=200-240 VAC, AC Input=Single Phase, Bus=Standalone。
- (2) 在 Studio 5000 中, 右键运动组, 新建 group1, 在 group1 里新建 axis1-AXIS_CIP_DRIVE, axis1 右键设置, General-Associate Module-Module = 刚新建的伺服驱动器设备, Motor-Data Source=Catalog Number, Motor-Change Catalog=VPL_A1001M-P
- (3) 在 Studio 5000 中, RUN-MODE 里测试电机, Axis 右键-Motion Direct Commands。MSO 上电, MSF 掉电, MAJ 正反转, MAS 停止。
- (4) 在 Studio 5000 中, MainRoutine 梯形图设置普通开关, 控制 MSO, 配制与测试程序一致。下载后执行控制, Ctrl+T 改变开关值。
- (5) 在梯形图中, 另外设置为反馈速度赋值的梯形图, 以及用其判定正转反转运行停止的梯形图。注意不动时速度不为 0, 且正转和反转都为运行。
- (6) 在 Connectivity 里, 创建所有变量的 tag, path 为全局变量名, 设置类型。
- (7) 在 Thingworx 里, 导入所有的 tag。新建 mashup, 包含 button, checkbox, LCD, time series, numeric entry, auto refresh 以及合适 label。
- (8) 在得到数据的 thing 里, 新建每个 button 的 service, 在 JavaScript 脚本里, 为各开关对应值先赋 True 再赋 False。
- (9) 在 mashup 里, 右侧添加 getProperties, setProperties 以及上述 services。LCD 关联 getProperties 里的反馈速度, checkbox 关联 getProperties 里的 status, numeric entry 关联 setProperties 的预期速度, button 的 click 关联上述 service。
- (10) 在 mashup 里, Auto refresh 设置关联 LCD, checkbox, time series。
- (11) 在 Thingworx 里, 添加一个 value stream 的 thing 叫 stream, 在得到数据的 thing 里 general-value stream-stream, 并设置反馈速度 property 的 logged 打勾。
- (12) 在 mashup 里, 添加 queryPropertyHistory 的 service, 关联到 time series, 设置 x, y 坐标值。最后重连 Connectivity。





五、 实习收获

在罗克韦尔实验室为期一月的实习是一段难得且有意义的经历,本次实习给我提供了一个学习和提升自己的平台和机会,我觉得收获颇丰。我们使用的硬件设备包括可编程控制器、E+H 温度计、伺服电机驱动器、工业触摸屏等,软件包括 CCW、Studio5000、RSLink Classic 等,均是实际工业现场可能使用到的设备,在实习过程中我感受到了自动化在工业生产中的广泛应用和重要作用,更增强了对自己所学专业的信心,并逐渐明确了自己未来的努力方向。

从软件方面来说,这是我第一次系统地学习梯形图编程。之前在运动控制和计算机控制的课程中虽然接触过一些简单的梯形图逻辑,但是还远远没有达到可以编程解决实际问题的程度。通过这一个月的学习和练习,我觉得梯形图和其他计算机语言一样,也是一种逻辑性强、模块化程度高的编程语言,而且在 CCW 中由于梯形图中把各个对象具体化成直接线圈、设置线圈、上升沿线圈等,把操作指令集成成为一个个模块例如高速计时器、计时器等,这些设计都使梯形图语言更简洁明了,每个梯级逻辑简单易懂,对用户非常友好。

在实习过程中,与队友的合作也是至关重要的。我们既要各自努力练习硬件软件的使用技能,又要默契配合,各自做好分工范围内的工作。遇到问题时,两人需要相互鼓励,攻坚克难,在困难面前必须戒骄戒躁,杜绝互相抱怨,才能配合无间。实习期间,我从队友身上也学习到了很多,从技能到态度,我觉得自己在人际关系和团队合作方面也成长了许多。

回顾一个月的实习生活,从最初阅读软件硬件说明书,一步一步按部就班操作,到脱离资料可以独立控制整套设备,再到可以解决问题、完成任务,我觉得实习的意义不仅仅在于新掌握了多少知识和技能,更在于通过接触实际工业生产过程来亲身体会在校学习的理论知识究竟有何实际意义,以及自己今后可以在什么样的工作岗位上发挥作用。有了这些思考之后,这一个月才算没有虚度。

最后,感谢罗克韦尔公司及其他赞助商为我们提供这样一个良好的平台,感谢交大电子信息与电气工程学院对我的栽培,平日里教授们的讲课,虽没有直接作用于比赛,但他们为我培养起的工程实践的思想,让我受益终身。自动化是一门传统而新兴的学科,我们人工智能方向的同学,需要将最前沿的 AI 技术,结合到传统的控制中,让传统学科继续大放光彩。

专业实习企业导师考核评分表

实习学生姓名		实习单位		企业指导教师	
实习日期		实习部门		团队负责人	
考核项目及说明			对应的毕业 要求指标点	项目分值	考核得分
1、严格遵守公司各项规章制度，无违纪情况，理解并遵守工程职业道德和规范，履行责任			8-3	10	
2、出勤				5	
3、了解相关的方针、政策与法律法规。了解相关工程项目的行业规范与标准			7-1	10	
4、能在规定时间内完成任务，遇到问题迅速反馈，工作质量达到预期效果				15	
5、正确理解工作目标，善于总结、学习，具有一定的自我学习能力			12-2	10	
6、了解本专业的科学技术发展前沿和趋势，了解专业领域的工程最新应用背景			6-1	10	
7、能够综合运用所学知识，对工程问题进行识别、分析、表达，获得有效结论			2-4	10	
8、能够针对具体问题设计解决方案，并考虑社会、健康、安全、法律、文化以及环境等因素			3-3	5	
9、了解自动化工程实践与环境保护的关系，能评价工程实践对社会可持续发展的影响			7-2	5	
10、报告撰写与陈述表达情况			10-3	20	
总 分				100	
实习单位综合意见：					
实习单位人事签章					
考核人员签名				考核日期	

实习报告 评阅人 意见	<p>评阅人（签名）：_____</p> <p>年 月 日</p>
系 主 任 意 见	<p>（实习成绩按等级制（A、B、C、D和F）方式记载）</p> <p>评 定 成 绩：_____</p> <p>系 主 任(签名)：_____</p> <p>年 月 日</p>