

基于 LabVIEW 语言的电梯控制系统程序设计

实验者：杨光，合作者：李晓辉，指导老师：俞熹

摘要

LabVIEW is the abbreviation of Laboratory Virtual Instrument Engineering Workbench. It is a product of National Instruments Corporation. In this work, the language is used to design a control system of the lift. It contains light intensity control component, temperature control component, weight measurement module and lift operating system. The program package has been tested to control the virtual instruments produced by Di Yang Company. And it works very well.

关键词

计算机实测与控制，电梯控制系统，LabVIEW 程序设计

I.引言

在日常生活中，我们经常要乘坐电梯。但我们只是知道进入电梯后按一下要到的楼层按钮，就能如愿以偿地到达该层；到达某一楼层后，我们还可以按开门或关门键，来延长或缩短门开的时间。而对于电梯到底是如何被控制的这一问题，大多数人都不会去思考。这涉及的其实就是计算机实测与控制的问题。

本文中，我们用 LabVIEW 语言编写了一套电梯控制系统，它包含电梯内光强控制组件，温度控制组件，质量测量组件，以及最重要的电梯操控系统。这些组件在迪阳公司生产的虚拟仪器系统上做了测试。测试结果表明，控制系统达到了所有预期的功能。在本文程序设计部分，我们将逐一介绍各个组件的设计思路以及编程问题。程序测试部分我们将简单介绍如何利用 LabVIEW 程序以及 A/D 采集卡来进行计算机实测与控制。在附录中，我们还将讨论使用 LabVIEW 语言时需要注意的一些问题，以及 LabVIEW 语言相比于传统的 C, Java 等语言的优缺点。

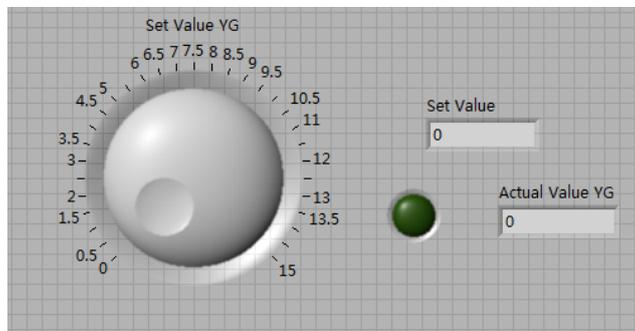
II.程序设计部分：

首先，需要说明的是，本文中我们所设计的各个组件都需要与外部设备进行数据交换。而实现数据交换的函数都是由迪阳公司编写的。它们是：(1)创建设备对象函数 CreateDevice；(2)释放设备对象函数 ReleaseDevice；(3)初始化和启动 A/D 卡函数 InitDeviceAD；(4)停止和释放 A/D 卡函数 ReleaseDeviceAD；(5)从 A/D 卡读取数据函数 ReadDeviceAD；(6)向 A/D 卡输出数据函数 WriteDeviceAD；(7)十六路开关量输入函数 GetDeviceDI；(8)十六路开关量输出函数 GetDeviceDO；(9)初始化 8253 计数器函数 IntDevCounter；(10)读取计数值函数 GetDevCounter。这些函数的端口设置以及具体功能见参考文献 3。下面我们将逐一介绍各个组件的设计思路以及编程问题。

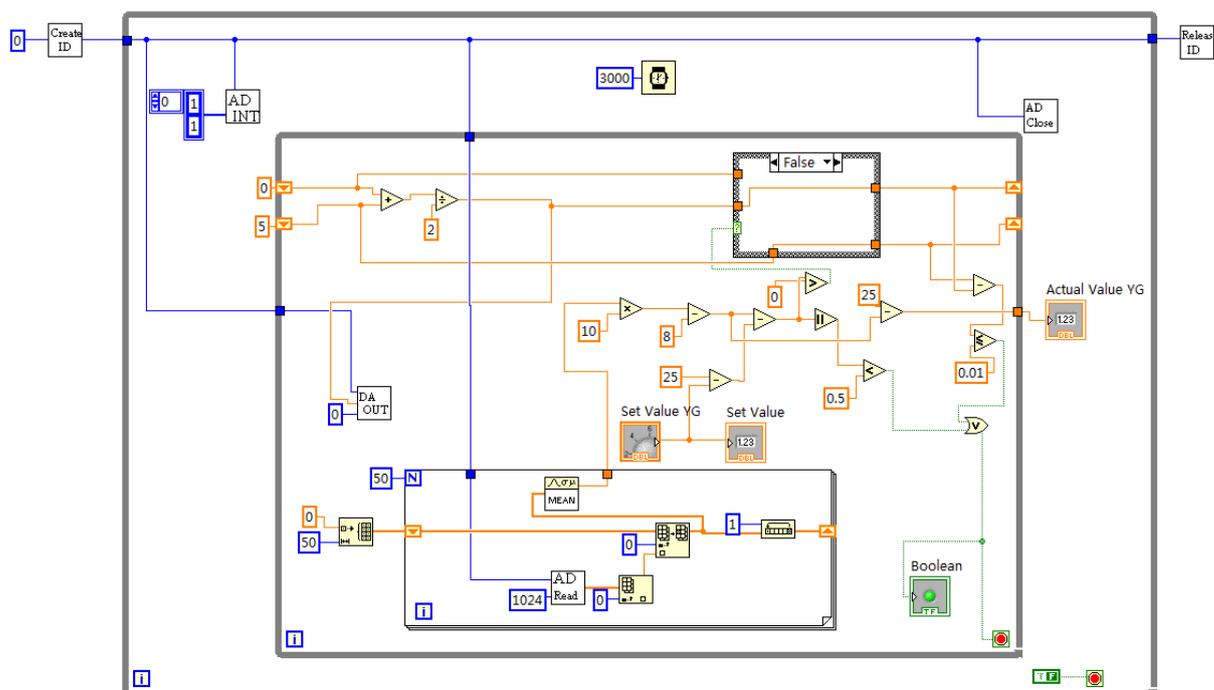
光强控制组件：电梯内的光强度由光强控制组件控制。前面板如图 (1) 所示，通过这一组件，电梯内的实际光强将被控制在输入的指定光强处。也就是说，当自然光光强提高时，电梯内灯上加的电压将降低，从而灯的亮度减弱；特别是环境光强很强时，电梯内灯将关闭。这是节能的体现——我们可以通过多种方法将自然光引入到电梯中，那么当白天自然光强时，由于光强组件的控制，电梯内灯灭，而到夜晚时灯打开。

组件的后台程序见图 (2)。由于灯发出的光强随其两端电压单调增加，因此我们可以采

用取区间中值逐步逼近的算法来实现所要的功能，而函数 WriteDeviceAD 输出的数据范围为 [0,5]，区间初始端点即可设为 0 和 5。在编程过程中还需要注意的一个问题是，由于电子噪声的存在，对于一个固定电压值，测得的光强值将有一定的波动。当波动幅度较大时，所用的算法就变得没有效率，有时还会给出错误的结果。那么我们就应该对测得的光强再做统计处理，用得到的均值来进行后续计算，这就是图（2）中 for 循环的作用。



图（1）：光强控制组件前台面板



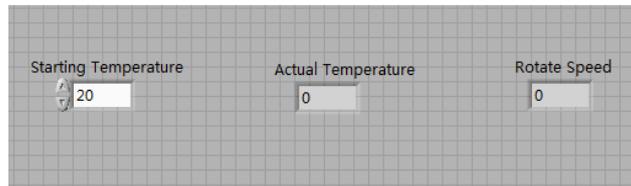
图（2）：光强控制组件后台程序

温度控制组件：对于一部电梯，要控制其中的温度很简单，装一部空调就解决问题了，但空调中必然要涉及到温度控制组件——当测得的实际温度超过设定温度时空调打开；而实际温度降到设定温度后，空调停止工作。这一部分就是要编写这样的组件，只不过是将空调换为了风扇¹。组件的前台见图（3），首先输入一个起始温度，当实际温度高于此温度时，风扇打开，且实际温度越高，风扇转速越快，并在前台显示风扇转速。

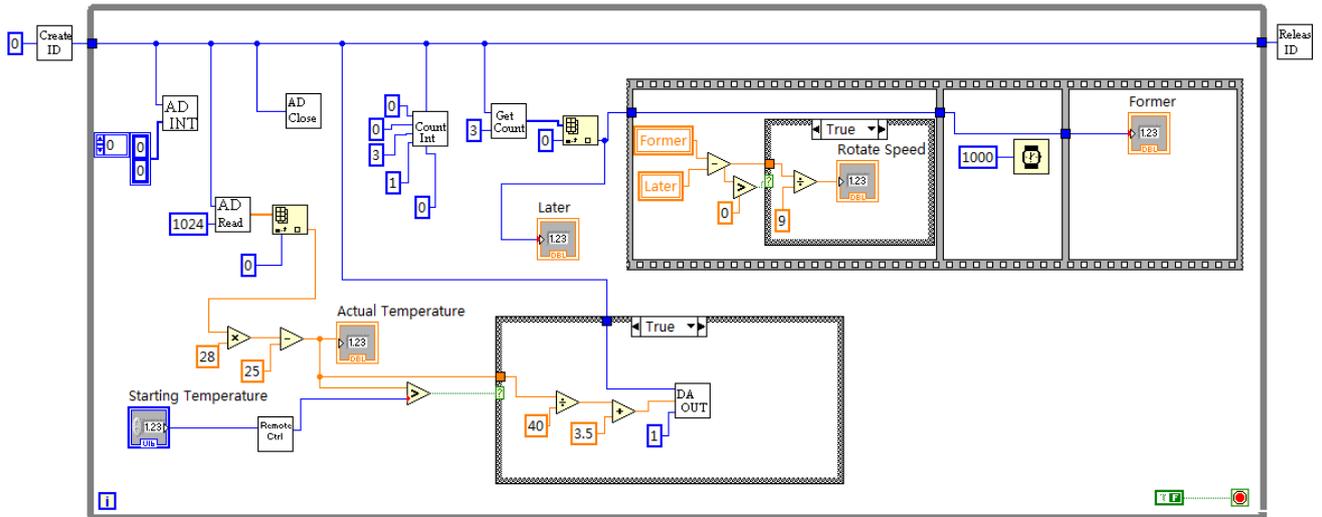
图（4）为组件的后台程序。它主要包括温度测量部分，风扇电压调控部分，以及风扇测速部分。由于 8253 计数器是倒着计数的，并且计数值达到 0 时，再计一数将返回计数器最大值。因此在风扇测速部分的程序编写中，我们需要用前一秒计数值减去后一秒的数值，而且只输出大于 0 的差值。还要注意的，风扇有 9 片扇叶，它转一圈，计数器要计 9 个数，

¹ 这样做是为了方便程序测试，因为迪阳公司生产的虚拟仪器系统里只提供了一个风扇而非空调。

因此以上的差值还应再除以 9 才能得到实际的转速值。另外，图（4）左下角还涉及到一个红外传输的子程序，我们输入的起始温度值将通过红外传输给风扇电压调控部分。这就实现了无线传输功能。红外传输的后台程序参见附录第一部分。

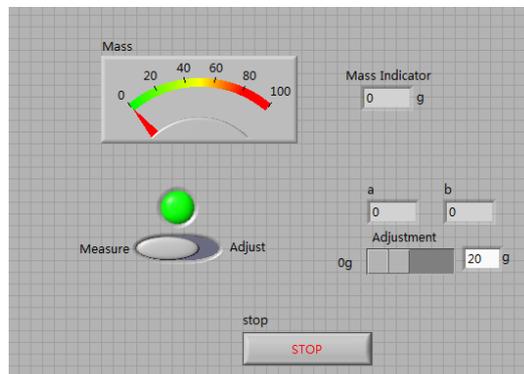


图（3）：温度控制组件前面板



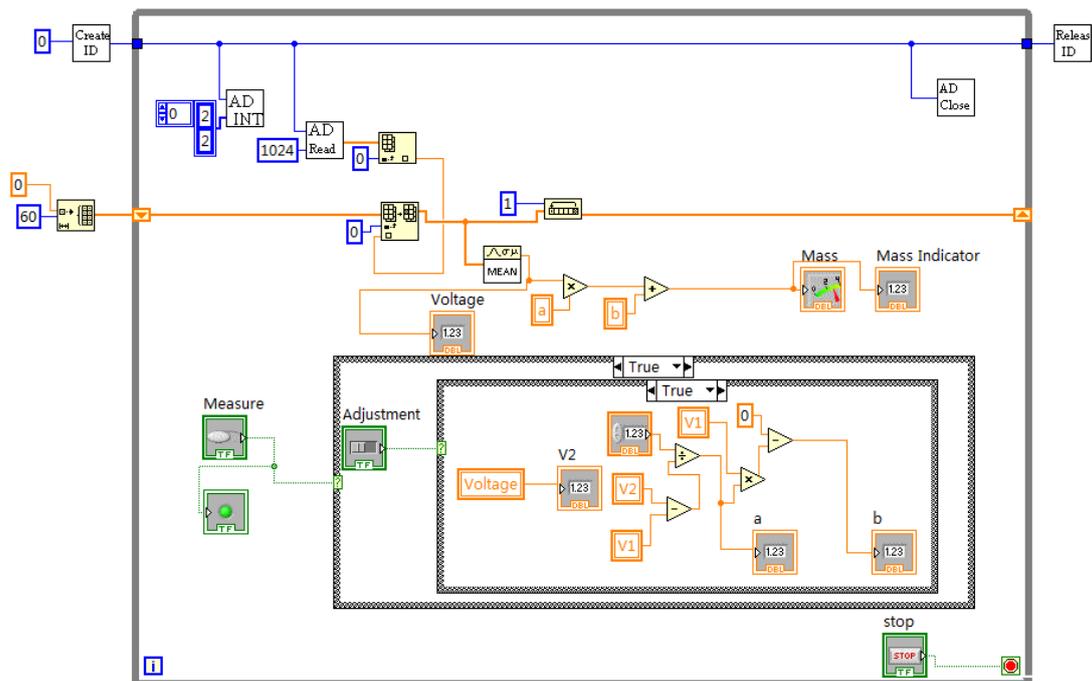
图（4）：温度控制组件后台程序

质量测量组件：这一组件的功能是实时的测量电梯内的承重。其前后台面板如图（5）（6）所示。由于硬件设备输入到此组件中的是电压值，因此需要将其转变为实际质量值。两者间的关系为 $M = a * V + b$ 。那么我们就需要通过校准来确定 a 和 b 的值。校准程序参见图（6）的下半部分。如图（5）所示，当我们要校准时，先将 Measure/Adjust 开关拨到 Adjust 档，此时开关上方的灯将变为红色；然后将 Adjustment 开关置于 0g 档，并在测重设备上不放任何东西；再将 Adjustment 开关置于 20g 档，并相应地在测重设备上放标准 20g 的砝码¹，当 a 和 b 稳定时即可将 Measure/Adjust 开关拨回到 Measure 档，此时绿灯亮，就可以进行实际测量了。



图（5）：质量测量组件前面板

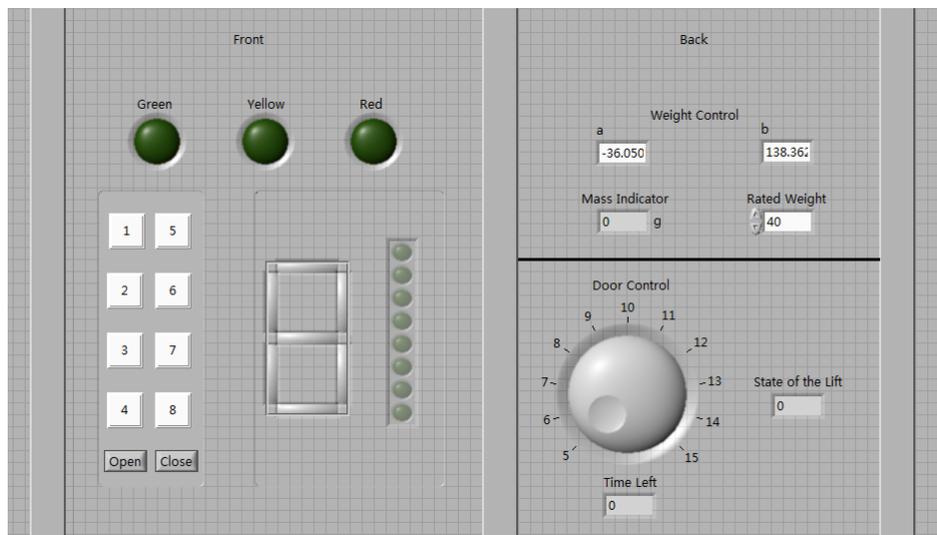
¹ 这里设为 20g 是为了方便之后在虚拟仪器上的测试。对于实际电梯，当然不能用 20g 进行校准，此时只要校准值与电梯内的实际承重值一致即可。



图（6）：质量测量组件后台程序

电梯操作系统：这是模拟电梯系统最重要的部分。其前面板如图（7）所示。这一系统要实现的功能是：1）调用质量测量组件实时测量电梯内承载的质量，当它高于设置的额定质量时，则在电梯门打开时，红灯亮并报警，直至承载质量低于额定值；2）乘客可以选择1~8层对应的按钮，按下后按钮灯亮，这就像实际电梯里的一样；3）当电梯开始运行时，绿灯亮，front板右边的数字灯将实时显示所在的楼层，而数字灯旁的一列小灯将给出现在电梯是上行还是下行的指示；4）当到达指定的一个楼层后，电梯门开，绿灯和此楼层的按钮灯灭，黄灯开始闪烁，闪烁时间为门开的时间，此时乘客可以按开门 Open 键以及关门 Close 键，来延长或缩短门开的时间，这也是和实际电梯一致的。

这部分的后台程序较为复杂，但需要注意的问题并不多。具体的后台程序介绍参见附录第二部分。



图（7）：电梯操作系统前面板

III.程序测试部分:

以上四个组件在迪阳公司生产的虚拟仪器系统上做了测试。此系统如图（8）所示。它包括相应的温度测量，光强检测与控制，红绿灯系统，红外传输，模拟电梯，电机调速与控制，电子秤设计等 7 个虚拟仪器。

我们把图（9）所示的 A/D 转换卡与电脑的 USB 口相连，再将 A/D 卡与虚拟仪器系统相连。这样即可通过上面所编写的 LabVIEW 程序对虚拟仪器系统进行实测与控制，并在此过程中测试程序的所有功能。测试结果表明我们的电梯控制系统达到了所有预期的功能。



图（8）迪阳虚拟仪器系统



图（9）A/D 转换卡

IV.结论:

本文中，我们利用 LabVIEW 语言编写了一套电梯控制系统，它包括光强控制组件，温度控制组件，质量测量组件，以及电梯操控系统。我们在虚拟仪器系统上做了相应的测试，测试结果表明这套系统达到了所有预期的功能。之后，这套系统稍作修改后即可应用到实际的电梯系统中。

参考文献:

1. LabVIEW Help, 2005–2008 National Instruments Corporation.
2. LabVIEW 计算机实测与控制, Fudan Physics Teaching Lab, http://phylab.fudan.edu.cn/doku.php?id=exp:labview_sensor.
3. U18 数据采集卡软件开发使用说明书, 软件版本 4.1, 北京迪阳公司。
4. U18 数据采集卡硬件使用说明书, 北京迪阳公司。
5. 模拟电梯工作 (步进电机), Fudan Physics Teaching Lab, <http://phylab.fudan.edu.cn/doku.php?id=exp:stepmotor-labview>.

附录:

I.红外传输的后台程序:

这段程序先将要传输的整数值转换为布尔数组，再依次传输数组中的各个布尔值，最后再把接收到的布尔数组转换成整数并输出。这里输出与输入的布尔数所在的通道值由硬件设备具体决定。

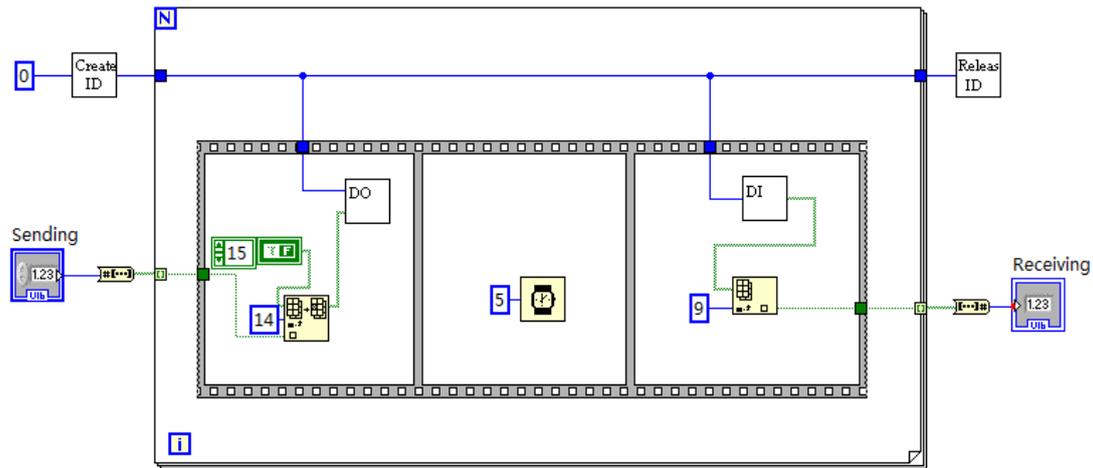


图 (1): 红外传输的后台程序

II. 电梯操控系统的后台程序:

电梯操控系统程序由 4 部分组成, 见图 (2) 到 (5)。图 (2) 部分主要是负责红黄绿三灯和数字灯的显示, 以及四相八拍步进电机的控制¹。图 (3) 是控制电梯运作的主要部分, 它负责电梯承重的实施监控, 楼层按钮信息以及所处楼层信息的读入, 电梯门的控制, 以及电梯运行方向的判断。尤其电梯门控制这部分程序的编写需要一定的技巧, 这段程序参见图中最里层的 while 语句。图 (3) 中最上方涉及到了校准后的质量测量子程序, 实际操作中我们只要将校准的 a 和 b 值输入到此子程序中即可, 它的后台程序见图 (4)。图 (5) 程序控制的是前面板上数字灯旁的一系列小灯, 这列小灯用来显示电梯的运行方向。

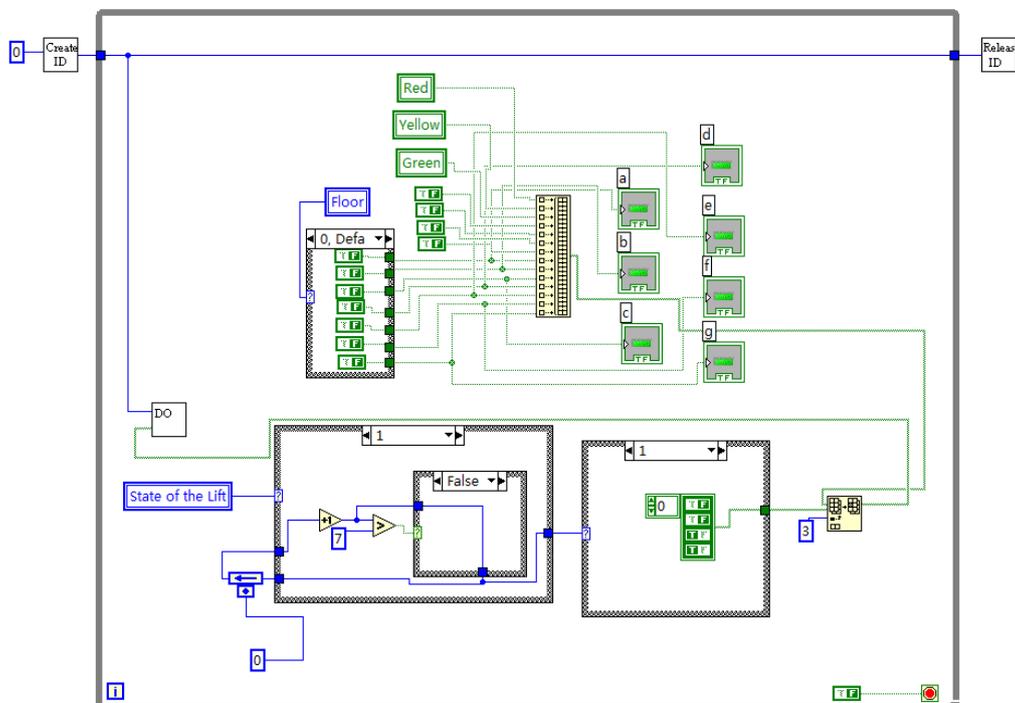
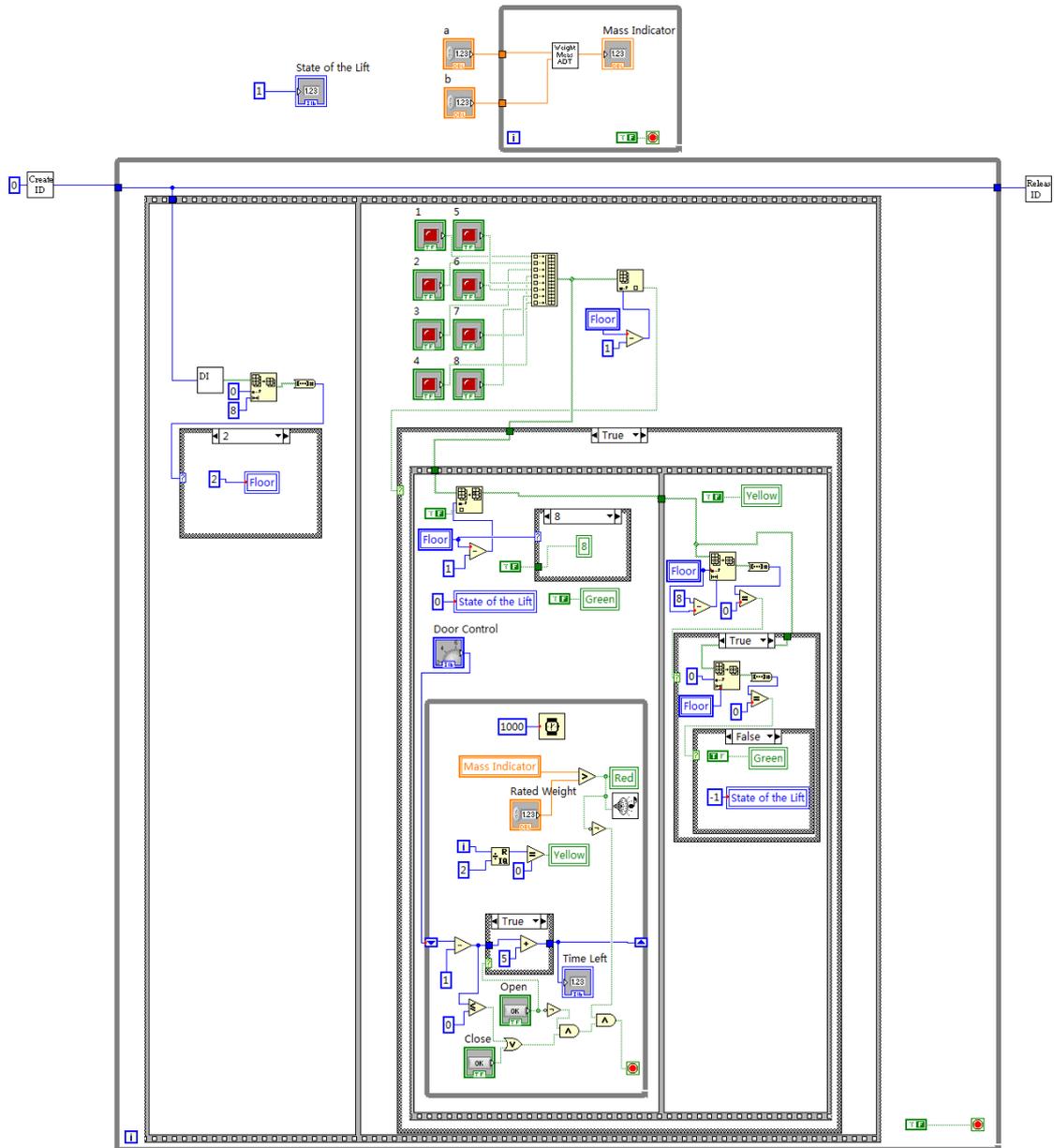
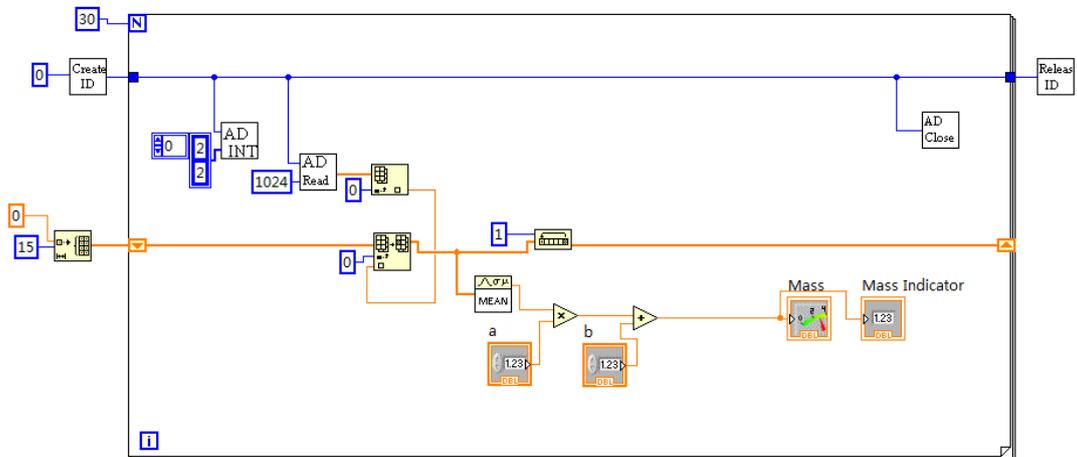


图 (2): 电梯操控系统的后台程序 1

¹这里编写的是四相八拍电机控制程序, 也是为了测试方便。因为虚拟仪器系统采用的就是这种步进电机。对于这种电机的介绍参见参考文献 5。



图（3）：电梯操控系统的后台程序 2



图（4）：电梯操控系统的后台程序 3

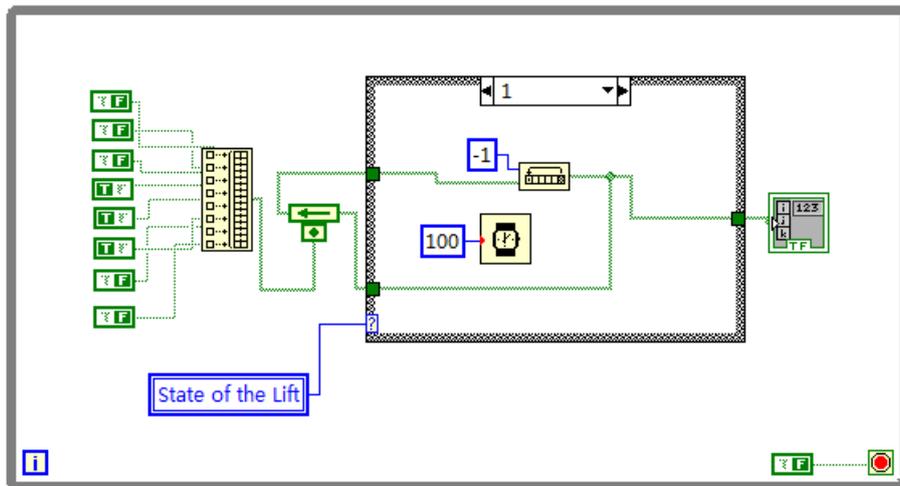


图 (5): 电梯操控系统的后台程序 4

III. 使用 LabVIEW 语言时需注意的问题:

对于懂得 C 或 Java 语言的人,学习 LabVIEW 语言很方便。但刚开始写 LabVIEW 语句时,有一个问题不容易搞清楚,这就是 LabVIEW 语句到底是按什么顺序执行的?在后台窗口中它是从上向下还是从左向右执行呢?这一问题的回答归纳为如下四点:

- (1) 如果几块语句之间无数据交换,那么它们是并行运行的,比如两个循环语句,如果它们之间无任何数据联系,则它们将同时被执行,不分先后。
- (2) 如果几块语句之间有数据交换,那么将先执行处于数据线 Write 端的语句,再执行 Read 端的语句,数据将从 Write 端流向 Read 端。比如两个循环间有数据线连接,循环 1 向循环 2 输出数据,那么将先执行循环 1,当循环 1 执行结束后再把数据输给循环 2,并开始执行循环 2。
- (3) 如果要控制语句的执行顺序,可以使用 Sequence Structure 语句。此时将从 Sequence Structure 语句最左端的框开始一个框一个框地向右执行。
- (4) 对于一个循环结构,只有在循环内所有语句都执行后,才会开始下一轮循环。这和 C 语言等是一致的。

IV. LabVIEW 语言的优缺点:

LabVIEW 语言与传统 C, Java 等语言相比有如下优缺点:

a) 优点:

- 1) 采用可视化编程方式,在做可视化用户界面时非常方便;
- 2) 包含 Highlight 运行方式,可以用此方式很方便地找出程序中的错误;
- 3) 其中的数组处理函数很全面,可以很方便地对数组进行数据操作;
- 4) 在数据运算方面,由于不同的数据流由不同颜色的数据线表示,因此很容易避免数据类型混用的错误。

b) 缺点:

- 1) 运行速度较慢,使得它不适合用来编写大型程序;
- 2) 如果程序较大时,数据线会显得很凌乱,这就使得程序不易被读懂;
- 3) 修改程序时,可能需要移动一部分图标的位置,此时为了避免数据线太乱,只能将图标周围的数据线先剪断,再做移动,这是不大方便的;
- 4) 如果要编写的程序对可视化要求不高,那么用 LabVIEW 语言则不如用 C 这样的语言方便。