

2. 系统框架设计及主要功能

本设计主要实现的功能集成于下面三个模块：

1. 分类训练模块：主要实现的是对电池三类故障的分类学习，该模块下有训练数据导入、BP 神经网络训练、分类性能计算等。
2. 状态监测模块：可以实现六节电池的实时特征指标输出和波形输出。
3. 故障诊断模块：与状态监测模块协调，实现实时故障诊断。

本系统程序框架如图 1 所示。本程序从实现上分层两个大部分，一是分类训练模式，基于 LabVIEW 的顺序结构模块，依次实现数据导入、数据标准化、模型训练和性能评价几个过程，其中必要的计算环节用 MATLAB 实现；二是状态监测+故障诊断环节，该部分设计了一六路电池通道和随机数生成通道，实现单节电池的输入，并同样基于 MATLAB 实现了故障诊断。整体程序放置在一个大 while 循环之中以实现程序不间断运行，检测和诊断部分放置于一个 For 循环之中，将运行直至电池数据读取完毕。

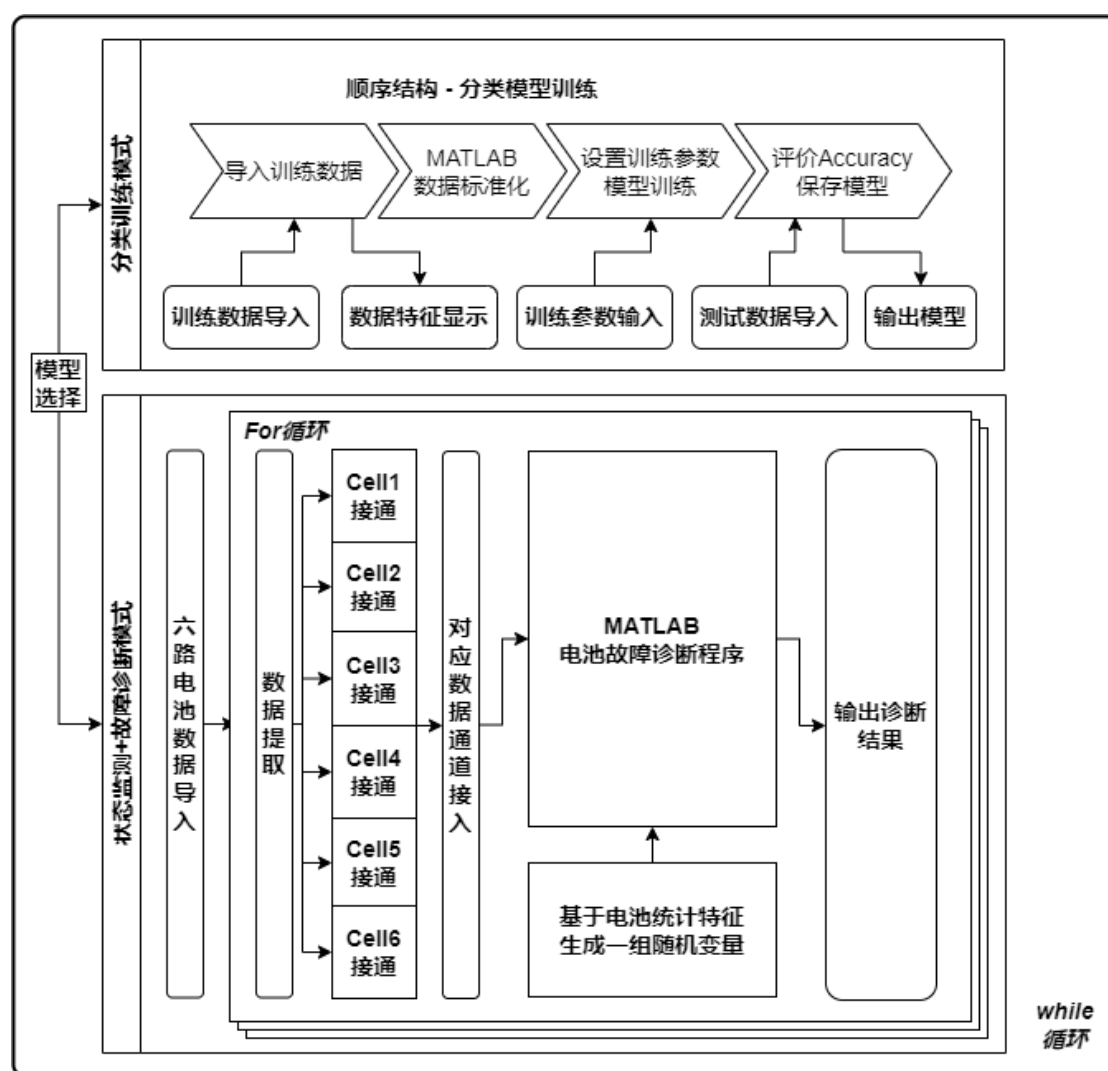


图 1 电池管理系统框架设计

系统的整体流程如图 2 所示。

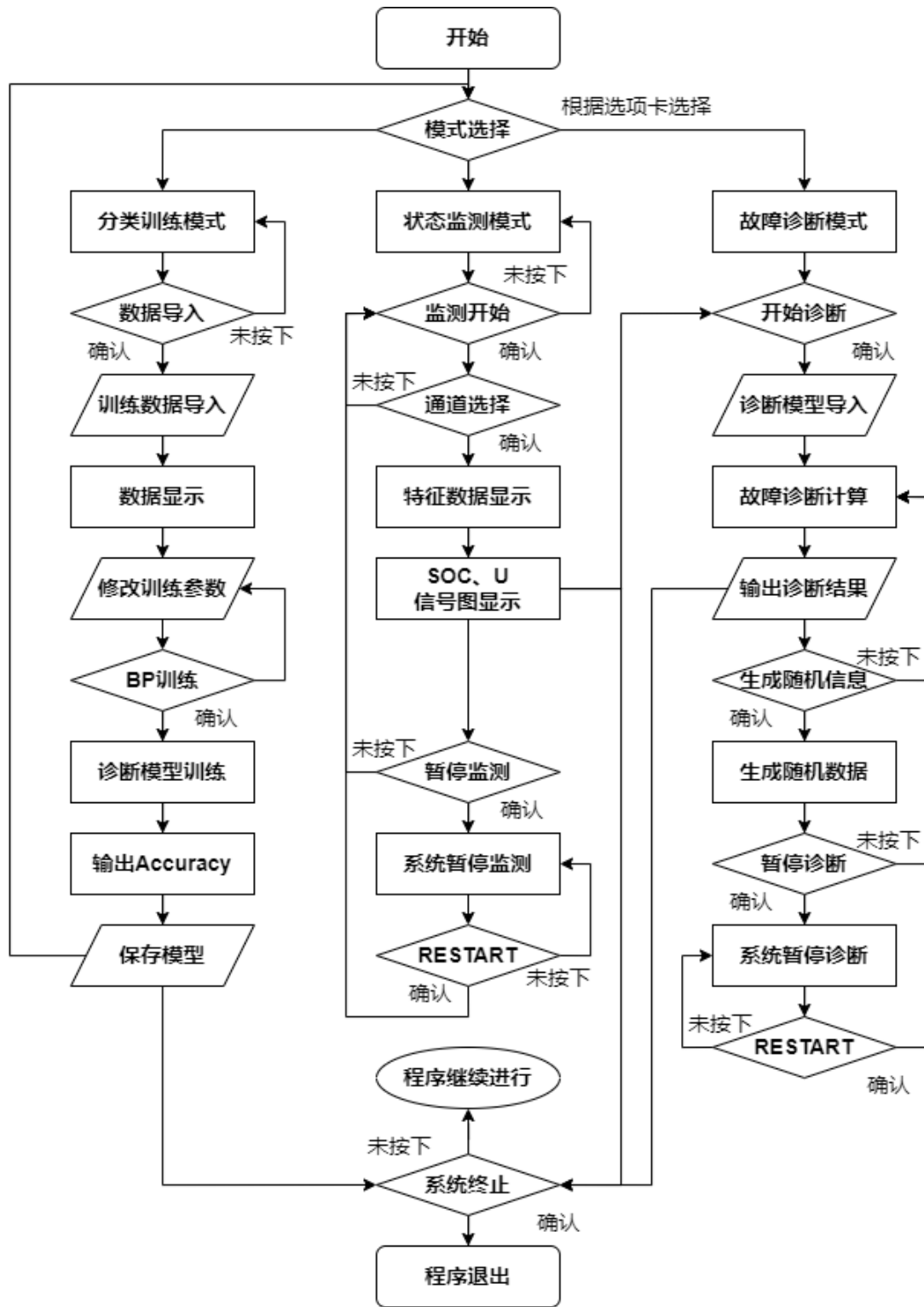


图 2 整体系统程序框图

3. 运行监测界面设计及实现

3.1 运行监测界面设计

本设计的测界面要实现的功能包括六节电池的实时特征监测,同时能通过选择开关来实现单节电池信号的采集或停止切换,并对用户关心的信号进行波形图的显示。本设计的运行监测界面如图 3 所示。

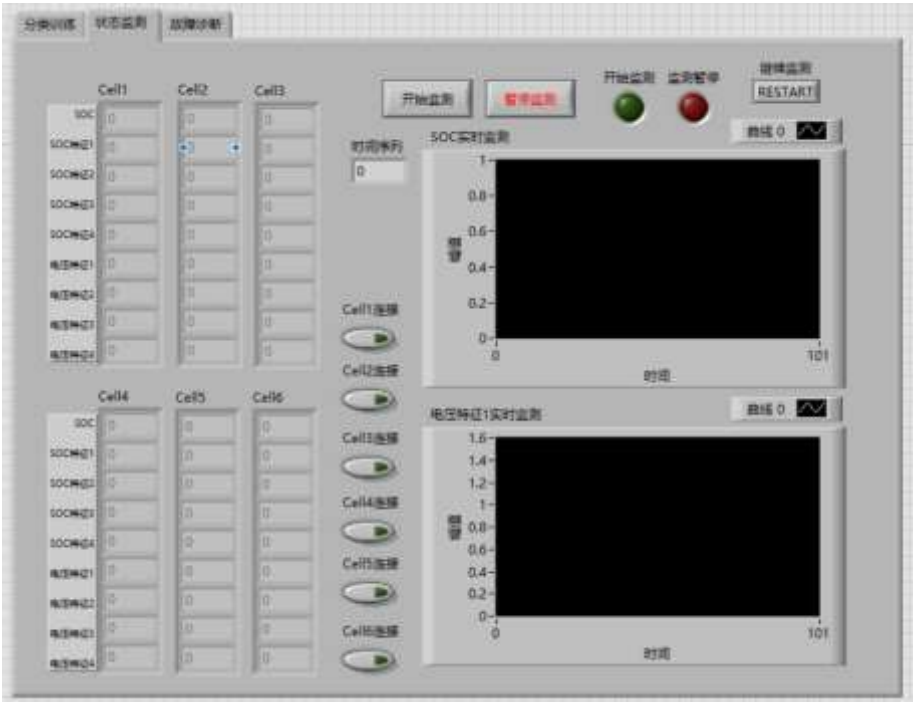


图 3 运行监测界面

本监测界面通过数值显示空间实时显示电池数据的信号特征值,并将 SOC 数据和电压特征 1 的数据通过波形图显示控件进行展示,通过六个布尔逻辑输入组建实现六路电池信号的输入状态切换,当前运行的时序数以一个数值显示控件在前面板的展示。

在运行控制上,本模块设置有开始监测、暂停监测和继续监测按键。当按下开始监测按键时,系统开始采集电池信息;在采集进行中,按下暂停监测按键则系统暂停采集电池信息,直到继续监测按键按下,恢复采集状态。

3.2 运行监测功能实现

监测功能的实现较简单,首先需要建立电池数据集。本设计基于 10000 条的电池数据,进行一定的处理后作为六节锂电池的数据信息,其建立方法如下:

Step1: 采用随机过采样的方法,扩充了“无任何故障”的样本数;

Step2: 制定模型训练数据集 input 和 output,分别为模型训练的特征变量输入和分类标签输入, input 只采用前九列变量;

Step3: 从扩充的样本集(无故障:有任一故障 $\approx 1:1$)中随机采样抽取了 6×1000 条不完全相同的数据集,作为 6 个单体电池的变量输入。上述数据中单体电池的特征变量、模型训练的输入特征变量和分类标签均采用 csv 格式,以逗号分隔,命名为 cellx.csv, x 为电池序号。

当开始监测按键被按下时，进入状态监测功能，LabVIEW 根据设定的文件地址，读取六节单体电池的 csv 文件，并以双精度二维数组的形式输入至下一个环节。

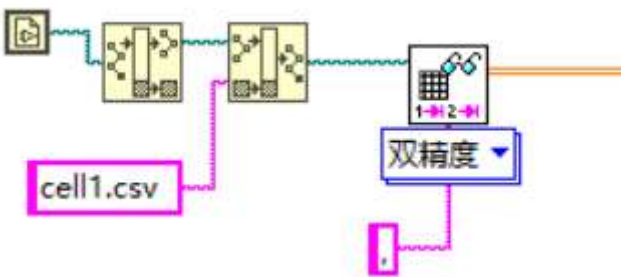


图 4 电池数据读取

六路电池信号均为 1000 条，所以电池状态监测环节设置为一循环总长度为 999 的 For 循环（初始时间为 0）。每一路电池信号进入 For 循环之后先经过两个索引数组单元，第一个索引数组基于当前循环的时间序列 i ，输出第 i 时刻的电池信息；第二个索引数组单元取出每一时刻下的第 1 个特征变量（SOC）和第 6 个特征变量（电压特征 1）。通道选择开关可以控制某节电池信号是否采集，其通过一个布尔输入控件和一个条件结构实现，当布尔输入为真，则电池数据可以直接通过；当布尔输入为假，则电池数据以全零替代。每一个电池信号对应一组索引单元、一个选择条件结构及一个数值显示控件，数值显示控件能实时显示当前的电池特征变量值，同时六组电池信号中提取出的 SOC 值和电压特征 1 通过一个簇合成再输入到两个显示波形图显示控件上。

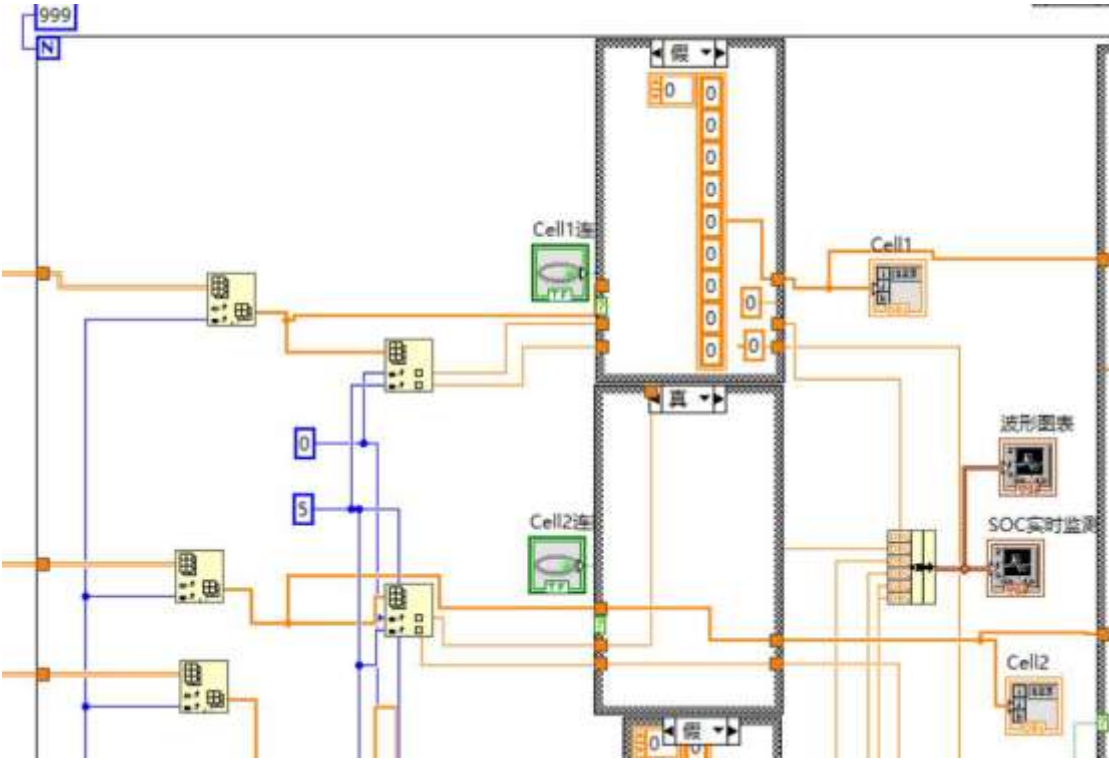


图 5 状态监测程序框图（后面板）

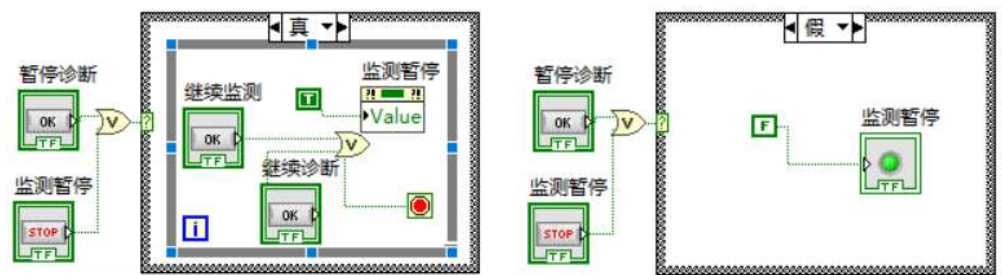


图 6 系统暂停与启动程序

本状态监测模式和故障诊断模式共用一组程序暂停-重启程序，在状态监测界面中有一组暂停监测-继续监测按钮，同样在故障诊断界面中有一组暂停诊断-继续诊断按钮，两个暂停按钮和两个继续按钮各通过一个或门连接，只要按下任意一个就可以使总输出为真。当输入为真时，执行一个空 while 循环，暂停监测的指示灯亮起，直到继续的总输出为真结束该循环。上述所有按钮都是按下时触发，松起时复原，所以当继续按键按下时，系统可以跳出循环，条件结构的状态为假，监测暂停的指示灯熄灭。故障诊断模块中亦有一组暂停-继续按键，与上述实现的逻辑相同。

检测过程为一循环过程，采用延时 500ms 的方式来方便结果的观察；同时在前面板上输出当前的时间序列。

3.3 效果展示及问题讨论

状态监测模式工作界面如图 7（全部通道接通）和图 8（部分通道接通）所示：

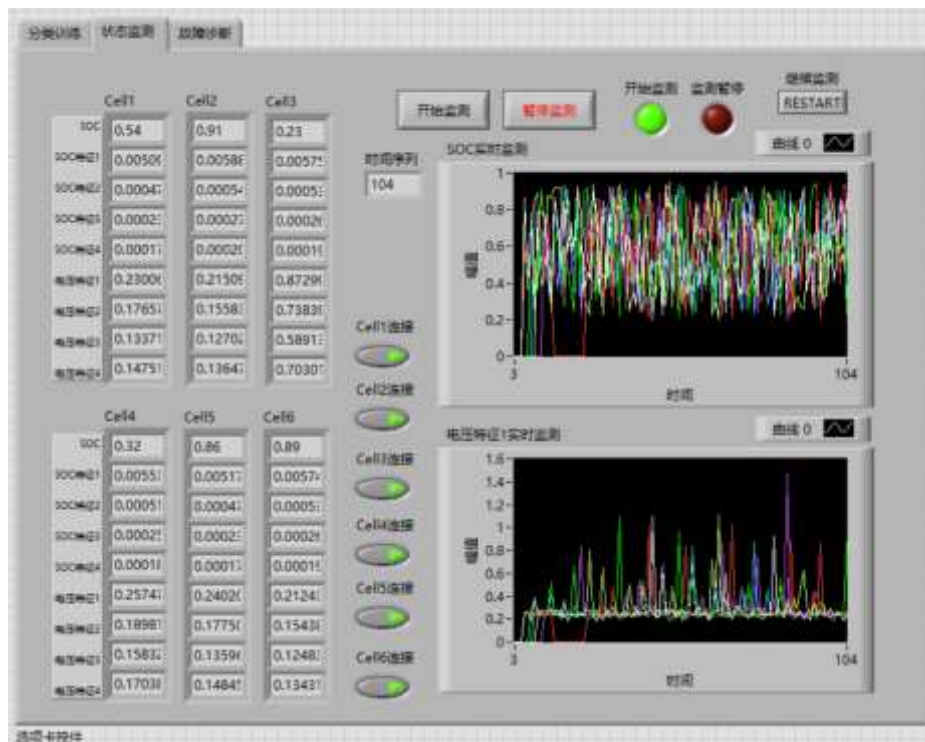


图 7 状态监测模式工作界面-全部通道接通

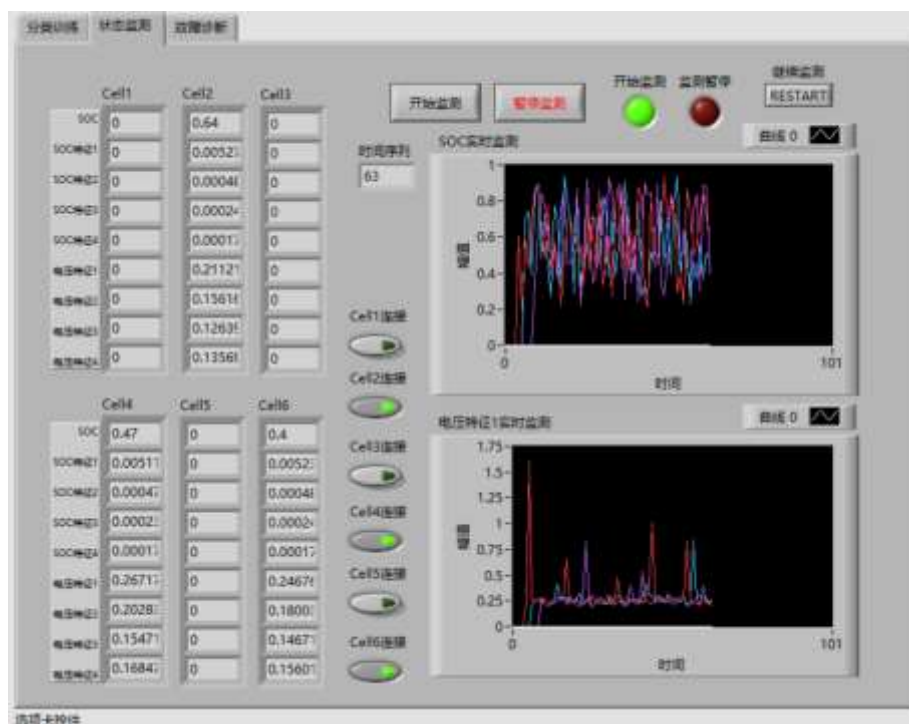


图 8 状态监测模式工作界面-部分通道接通

在设计之时，因为要实现单一通道的数据采集开始和采集终止，在起初是计划当通道连接按键关闭时，直接不输出对应的电池信息，但是该方式会使得后续的簇合成模块少一路输入，导致该处故障报警，所以本设计采用了用一个 9×1 的全 0 数组替代原理的信号输出。

在设计之初是计划用开始采集按键实现暂停状态下的继续采集功能，但是由于开始采集按键置于整个状态监测-故障诊断的条件结构之外，而暂停的 while 循环在条件结构的 For 循环之中，所以该信号起作用的前必须要执行完该 For 循环，显而易见这不可能实现，故就采用了一个重启按键 RESTART 来使采集恢复。

4. 故障诊断界面设计及实现

4.1 故障诊断界面设计

本系统能对 10000 条样本的电池数据进行 BP 神经网络的分类训练，并通过以此生成的模型进行实时的故障诊断。实现本功能需要两个部分，一是分类训练模块，二是故障诊断模块。

分类训练模块的前面板上有路径选择的地址框，以此选择特征变量 input 和分类标签 output 的数据集；按下数据导入按钮后，对应指示灯亮起，同时可以在数组显示控件和波形图上查看对应的信息；当按下 BP 训练按钮时，对应指示灯亮起，同时根据设置的模型参数进行 BP 神经网络分类模型的训练，当训练完毕后，输出整体分类的准确度（Accuracy）和三类电池故障的诊断准确度。

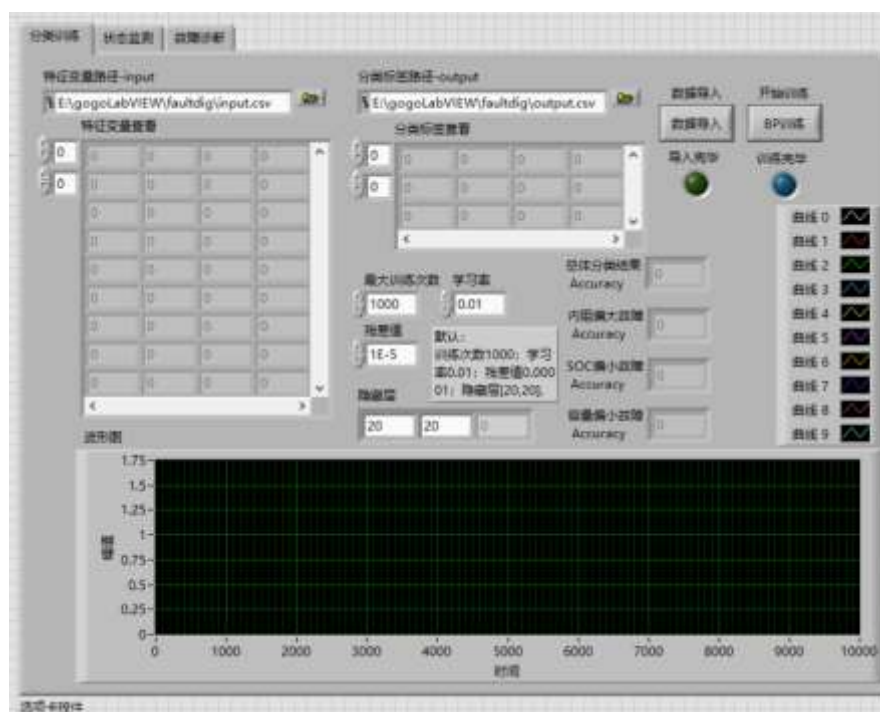


图 9 分类训练界面

故障诊断模块是与状态监测模块同步的，当状态监测功能和状态诊断功能同时启动时，开始对应电池的故障诊断，每一路电池的故障信息在界面左侧的界面上进行展示，右侧的波形图表能显示六路电池的 SOC 信息。当按下随机生成电池信息的按钮时，对应指示灯亮起，同时系统会根据原始数据集的电池统计特征，生成一组随机的电池信息并对该组信息做故障诊断，并在数值显示控件上输出特征变量、诊断结果和 BP 神经网络的实际输出值。

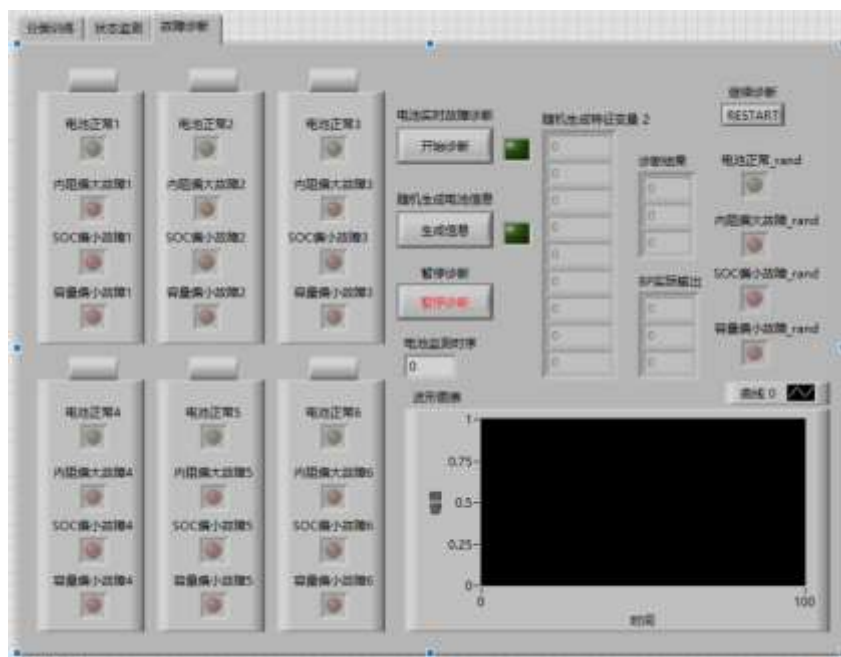


图 10 故障诊断界面

4.2 故障诊断功能实现

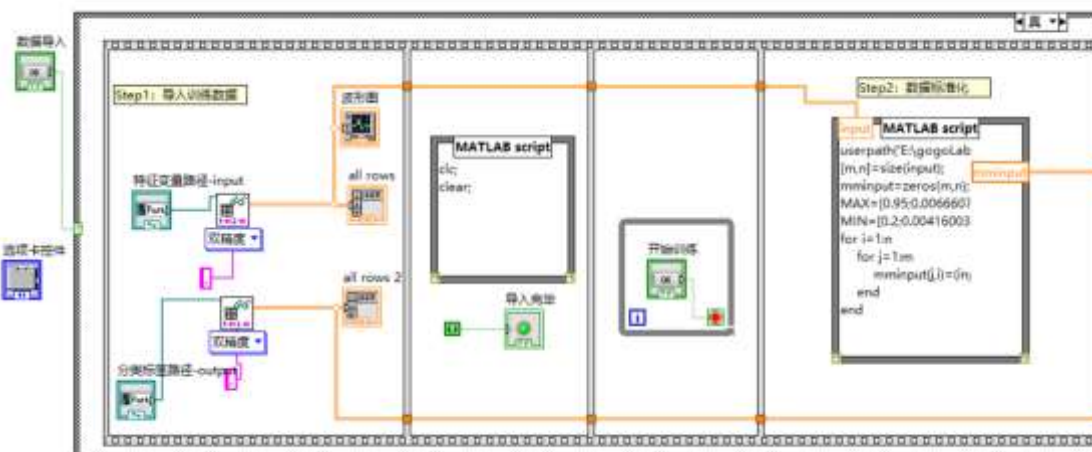


图 11 分类训练程序框图（1）

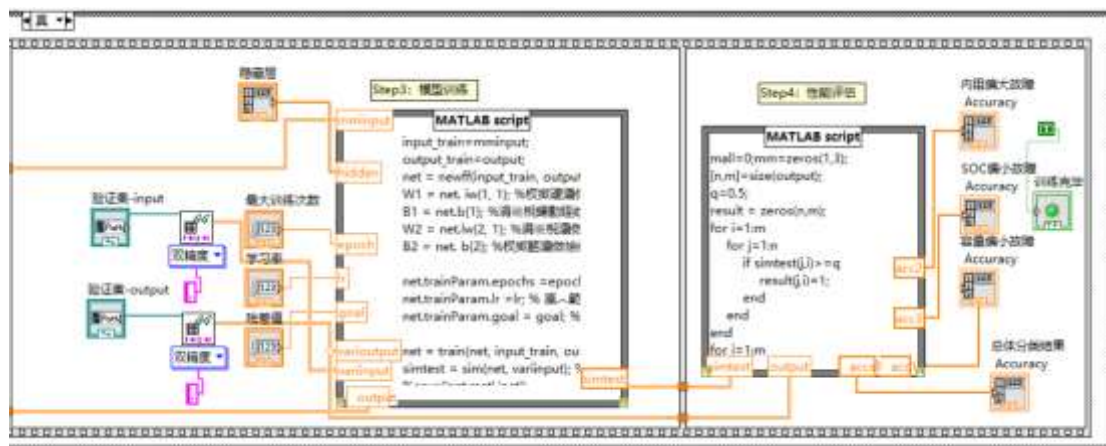


图 12 分类训练程序框图（2）

本设计的分类训练模块整体置于一个条件结构之中，当按下数据导入按钮时程序进入到该条件结构之中。分类训练功能主体为一顺序结构，训练程序由四步组成：

Step1: 导入训练数据，将文件地址下的数据集转换为双精度二维数组，在显示控件上展示，同时开启 MATLAB 命令窗口，等待训练开始信号；

Step2: 数据标准化，为了使得特征变量的特征信息更加明显，提高模型的分类性能，对输入变量进行最值标准化处理；

Step3: 模型训练，调用 BP 神经网络训练程序，并通过前面板上输入的参数进行分类模型训练，模型评价采用验证集 input 和 output 实现；

Step4: 性能评估，通过验证集的分类结果计算总体分类准确度 (Accuracy) 和三个单项故障的分类准确度。

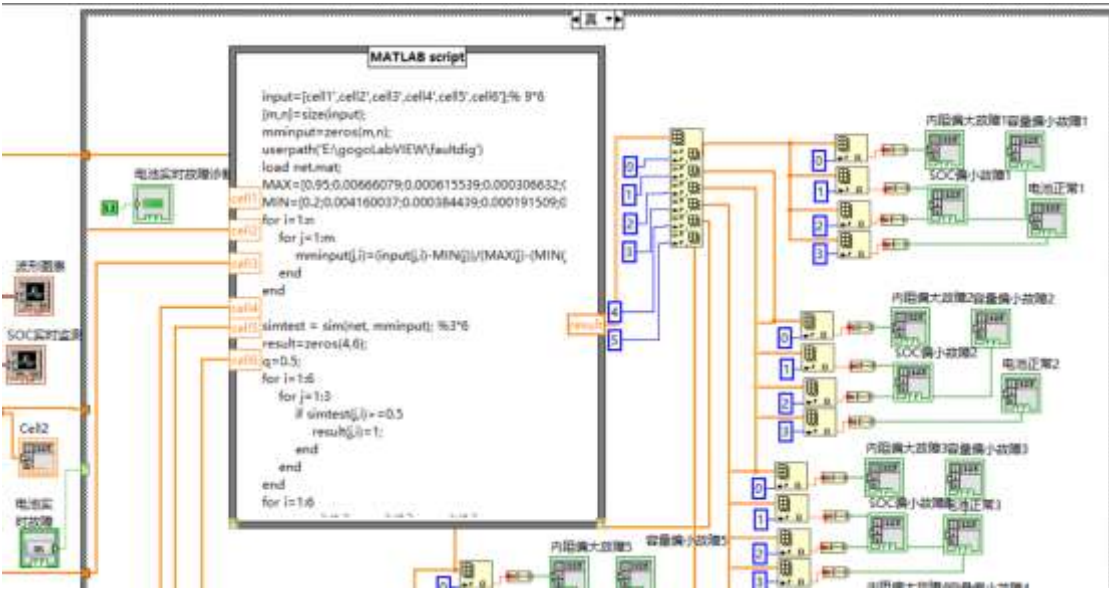


图 13 故障诊断程序框图

当故障诊断功能开启时，进入故障诊断条件结构，其核心为一 MATLAB 脚本框，其功能首先是合成六路电池信号（若未接入用 0 数组代替），再进行数据标准化，与训练的模型在数值上统计特征对齐，并通过分类训练模块保存的神经网络模型进行故障诊断，得到对应的分类结果，最后整理出六路电池的分类结果。分类结果用过两层数组索引抽出单一电池单一故障（或电池正常）状态信息，第一层数组索引的功能是分离出单一电池的故障信息，第二层数组索引的功能是分离出某一电池的单一故障信息，所以的故障信息通过布尔显示控件在前面板上展示。

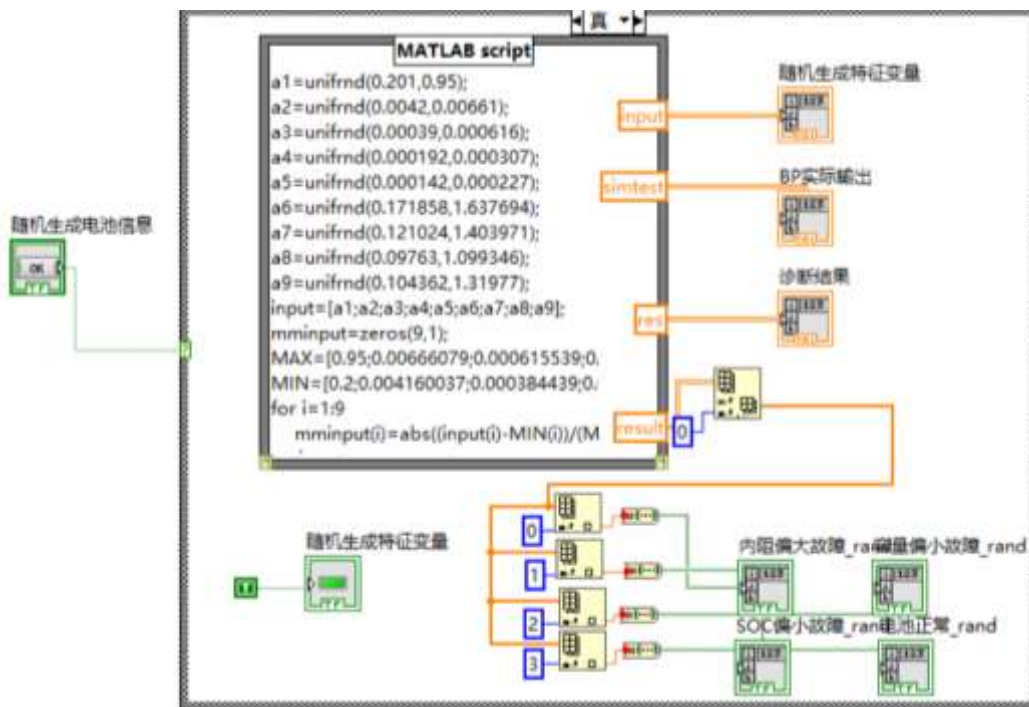


图 14 随机特征生成及诊断程序框图

当电池诊断功能开启时，再选择随机生成电池数据的功能，实现一次随机电池信息的生成和故障诊断。随机电池信息通过 MATLAB 的均匀随机数函数 `unifrnd()` 实现，所有数据均在原始训练数据对应特征下的最值范围内。故障诊断方法同前面程序，同时输出随机生成的特征信息、诊断结果和 BP 输出的实际值。

4.3 效果展示及问题讨论

分类训练模块的运行结果如图 15（数据导入和模型训练）和图 16（训练完毕后的准确度显示）所示：

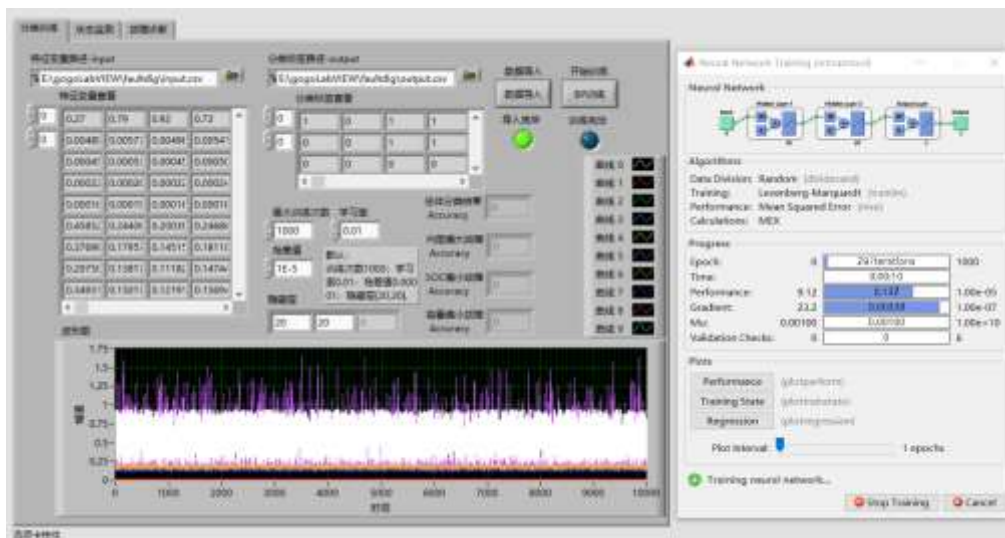


图 15 数据导入和模型训练



图 16 训练准确度显示

在设计分类训练模块时一个突出问题是 MATLAB 脚本的调用，在设计初期由于安装路径有误，LabVIEW 无法找到 MATLAB 地址，后续将 LabVIEW 和 MATLAB 安装在同一个盘符下就解决了该问题。为了简化 MATLAB 调用的过程，在设计时尽可能地将功能用过代码编写，LabVIEW 程序框图的控件仅实现数据的输入和输出展示功能，计算全部通过 MATLAB 实现。

故障诊断模块的运行结果如图 17（故障诊断功能实现）、图 18（故障诊断功能实现-部分通道）和图 19（随机特征生成级故障诊断）所示：

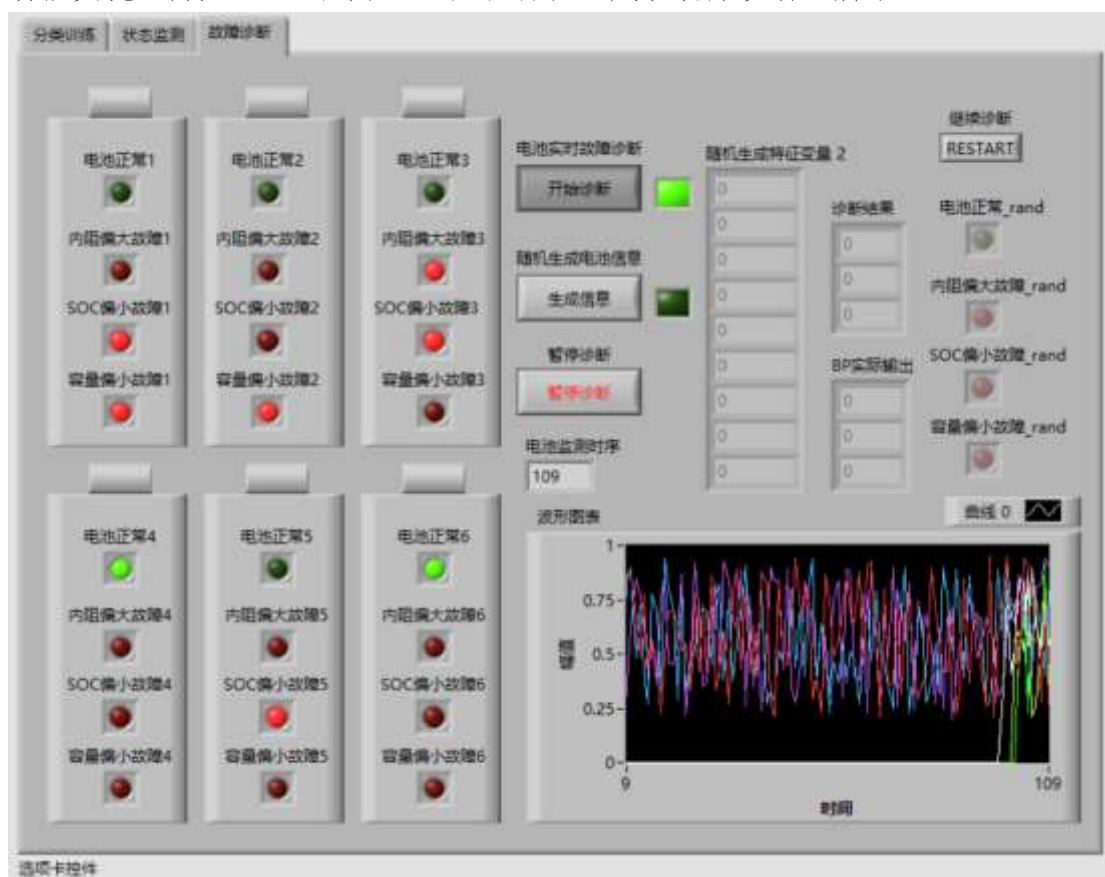


图 17 故障诊断功能实现



图 18 故障诊断功能实现-部分通道



图 19 随机特征生成级故障诊断

在设计本模块时最难的是如何将分类结果转换为前面板可视化的指示灯，最开始计划通过先绘制灯再与输出结果相连接的方式，但是由于输出信号为布尔数组信号，无法将该数组转换为单纯的布尔值，所以无法连线。但是该问题可以通过将分类输出转换为布尔数组后直接创建布尔显示控件实现。在故障诊断程序中必须要注意输入特征变量的标准化，不然会导致诊断结果异常。