|  |
| --- |
| 南京长盛仪器 |
| ACW\DCW\IR\GR测试程序文档 |
| 耐压测试仪测试程序说明文档 |
|  |
| **王鑫** |
| **2017/11/5** |

|  |
| --- |
| ACW\DCW\IR\GR测试程序流程说明 |

|  |  |  |
| --- | --- | --- |
| 日期 | 版本 | 编写人 |
| 2017.11.5 | V1.0 | 王鑫 |
|  |  |  |

目录

[2 测试状态机 2](#_Toc497661809)

[2.1 N模式下ACW\DCW\IR\GR测试状态机 2](#_Toc497661810)

[2.2 G模式下ACW\DCW测试状态机 3](#_Toc497661811)

[3 文件结构 4](#_Toc497661812)

[3.1 公共目录下的文件 4](#_Toc497661813)

[3.2 ACW的测试包含了N模式和G模式两种测试流程文件 5](#_Toc497661814)

[3.3 DCW的测试包含了N模式和G模式两种测试流程文件 5](#_Toc497661815)

[3.4 IR的测试流程文件 5](#_Toc497661816)

[3.5 GR的测试流程文件 5](#_Toc497661817)

[4 公共测试数据结构 5](#_Toc497661818)

[5 测试模块接口定义 7](#_Toc497661819)

[5.1 ACW/DCW/GR的测试接口函数流程图 8](#_Toc497661820)

[5.2 IR测试模块接口函数流程图 9](#_Toc497661821)

# 测试状态机

## N模式下ACW\DCW\IR\GR测试状态机



图1是ACW\DCW\IR\GR在N模式下的测试状态转移示意图开始的测试就绪态是用户层对状态机进行的初始状态。在整个测试过程中如果当前步测试发生异常，状态机会暂停，用户层进入到异常处理，处理完成后，如果满足了失败继续的测试条件，用户层就会将状态机切换到失败继续状态下继续测试。ACW\DCW包含了电压上升、正在测试、电压下降、间隔时间几个测试状态。IR包含了电压上升、正在测试、间隔时间几个测试状态。GR包含了正在测试、间隔时间两个测试状态。

当继续测试标记被置1后在用户层会进行相应的操作后继续加载并启动下一步测试。测试完成标记被置1后用户层会结束测试并进行相关操作。

图1

## G模式下ACW\DCW测试状态机



图2是ACW\DCW在G模式下的测试状态转移示意图开始的测试就绪态是用户层对状态机进行的初始状态。在整个测试过程中如果当前步测试时间到了，下一步不为空此时要将状态机切换到再次就绪状态。在用户层要完成对当前步到下一步的切换工作，并更新一下显示信息。更新完成后再次回到状态机的再次就绪态下继续运行。

测试完成标记被置1后用户层会结束测试并进行相关操作。

图2

# 文件结构



文件结构(图3)

图3列出了测试程序文件的结构包含了ACW\DCW\IR\GR的测试状态机的工作流程。

## 公共目录下的文件

|  |  |
| --- | --- |
| 文件 | 功能 |
| cs99xx\_relay\_motion.c和cs99xx\_relay\_motion.h | 提供继电器应用层接口 |
| cs99xx\_vref.c和cs99xx\_vref.h | 提供了设置于ARC,SHORT基准的接口 |
| test\_com.c和test\_com.h | 为用户层提供测试程序接口 |
| test\_tools.c和test\_tools.h | 测试程序使用的工具函数接口 |

表1

## ACW的测试包含了N模式和G模式两种测试流程文件

|  |  |
| --- | --- |
| 文件 | 功能 |
| acw\_test.c和acw\_test.h | 实现了ACW在N模式的代码 |
| acw\_test\_g.c和acw\_test\_g.h | 实现了ACW在G模式的代码 |
| acw\_count.c和acw\_count.h | 实现了对ACW采样数据的校准和异常判断的代码 |
| acw\_relay.c和acw\_relay.h | 实现了在ACW模式下的继电器切换的代码 |

表2

## DCW的测试包含了N模式和G模式两种测试流程文件

|  |  |
| --- | --- |
| 文件 | 功能 |
| dcw\_test.c和dcw\_test.h | 实现了DCW在N模式的代码 |
| dcw\_test\_g.c和dcw\_test\_g.h | 实现了DCW在G模式的代码 |
| dcw\_count.c和dcw\_count.h | 实现了对DCW采样数据的校准和异常判断的代码 |
| dcw\_relay.c和dcw\_relay.h | 实现了在DCW模式下的继电器切换的代码 |

表3

## IR的测试流程文件

|  |  |
| --- | --- |
| 文件 | 功能 |
| ir\_test.c和ir\_test.h | 实现了IR测试模块的代码 |
| ir\_count.c和ir\_count.h | 实现了对IR采样数据的校准和异常判断的代码 |
| ir\_relay.c和ir\_relay.h | 实现了在IR模式下的继电器切换的代码 |
| cs99xx\_ir\_shift\_gear.c和cs99xx\_ir\_shift\_gear.h | 实现了在IR模式下的自动换档的代码 |

表4

## GR的测试流程文件

|  |  |
| --- | --- |
| 文件 | 功能 |
| gr\_test.c和gr\_test.h | 实现了GR测试模块的代码 |
| gr\_count.c和gr\_count.h | 实现了对GR采样数据的校准和异常判断的代码 |
| gr\_relay.c和gr\_relay.h | 实现了在GR模式下的继电器切换的代码 |

表5

# 公共测试数据结构

测试程序维护了一个公共测试数据结构用来进行数据传递

typedef struct{

uint16\_t sample\_vol\_ad;///<电压采样AD

uint16\_t sample\_cur\_ad;///<电流采样AD

float vol\_value;///<电压显示值

float cur\_value;///<电流显示值

float real\_value;///<真实电流显示值

float res\_value;///<电阻显示值

float cap\_value;///<电容显示值

uint8\_t vol\_segment;///<电压段

uint16\_t output\_da;///<输出DA值

float out\_da\_k;///<电压/电流DA校准k值

float out\_da\_b;///<电压/电流DA校准b值

float vol\_adc\_k;///<电压AD校准k值

float vol\_adc\_b;///<电压AD校准b值

float cur\_adc\_k;///<电流AD校准k值

float cur\_adc\_b;///<电流AD校准b值 //GR使用

float res\_adc\_k;///<电阻AD校准k值

uint8\_t fail\_num;///<异常码

uint8\_t test\_over\_fail;///<测试时间到异常码

uint8\_t test\_status;///<测试状态

uint16\_t test\_time;///<测试时间

uint16\_t dis\_time;///<显示时间

uint8\_t test\_over;///<测试结束

uint8\_t gradation;///<测试阶段

uint8\_t ready\_ok;///<就绪完成标记

uint8\_t cont;///<继续测试标记

uint8\_t vol\_err\_count;///<电压异常计数

uint8\_t low\_err\_count;///<下限报警计数

uint32\_t vol\_rise\_step\_t;///<电压上升/下降阶段时间计数

float vol\_ch\_step;///< 电压缓变步进值

float vol\_ch\_base;///< 电压变化的起始值

float vol\_ch\_target;///< 电压变化的目标

}TEST\_DATA\_STRUCT;

|  |  |  |
| --- | --- | --- |
| 成员变量 | 类型 | 功能 |
| sample\_vol\_ad | uint16 | 电压采样AD |
| sample\_cur\_ad | uint16 | 电流采样AD |
| vol\_value | float | 电压显示值 |
| cur\_value | float | 电流显示值 |
| real\_value | float | 真实电流显示值 |
| res\_value | float | 电阻显示值 |
| cap\_value | float | 电容显示值 |
| vol\_segment | uint8\_t | 电压段(电压分为两段) |
| output\_da | uint16\_t | 输出DA值 |
| out\_da\_k | float | 电压/电流DA校准k值 |
| out\_da\_b | float | 电压/电流DA校准b值 |
| vol\_adc\_k | float | 电压AD校准k值 |
| vol\_adc\_b | float | 电压AD校准b值 |
| cur\_adc\_k | float | 电流AD校准k值 |
| cur\_adc\_b | float | 电流AD校准b值 (GR使用) |
| res\_adc\_k | float | 电阻AD校准k值 |
| fail\_num | uint8\_t | 异常码 |
| test\_over\_fail; | uint8\_t | 测试时间到异常码 |
| test\_status | uint8\_t | 测试状态 |
| test\_time | uint16\_t | 测试时间 |
| dis\_time | uint16\_t | 显示时间 |
| test\_over | uint8\_t | 测试结束 |
| gradation | uint8\_t | 测试阶段 |
| ready\_ok | uint8\_t | 就绪完成标记 |
| cont | uint8\_t | 继续测试标记 |
| vol\_err\_count | uint8\_t | 电压异常计数 |
| low\_err\_count | uint8\_t | 下限报警计数 |
| vol\_rise\_step\_t | uint32\_t | 电压上升/下降阶段时间计数 |
| vol\_ch\_step | float | 电压缓变步进值 |
| vol\_ch\_base | float | 电压变化的起始值 |
| vol\_ch\_target | float | 电压变化的目标 |

表6

# 测试模块接口定义

|  |  |
| --- | --- |
| run\_acw\_test | N模式下ACW对外提供的测试接口 |
| run\_acw\_test\_g | G模式下ACW对外提供的测试接口 |
| run\_dcw\_test | N模式下DCW对外提供的测试接口 |
| run\_dcw\_test\_g | G模式下DCW对外提供的测试接口 |
| run\_ir\_test | IR对外提供的测试接口 |
| run\_gr\_test | GR对外提供的测试接口 |

表7

void run\_xxx\_test(NODE\_STEP \*step, NODE\_STEP \*next\_step, TEST\_DATA\_STRUCT \*test\_data)

每个测试模块对外提供的测试接口形式都是相同的其中第1个参数是当前步的数据结构指针，第二个参数是下一步的数据结构指针(如果没有下一步就传入空值)，第三个参数是测试数据的结构指针(该指针指向系统中唯一一个测试数据结构实体g\_test\_data)

## ACW/DCW/GR的测试接口函数流程图



图4为ACW/DCW/IR测试模块接口函数的流程图该流程图，反应出在测试就绪后才会对进行状态机的状态计算因为，测试就绪状态下会对测试环境进行初始化的准备工作包括开定时器。只有当定时器打开后，时间跑起来，才能进行状态计算。所以要引入一个测试准备就绪标记置1的操作。当采样计算出的数据被判定为异常时测试异常标记会被标记为相应的异常码，异常发生后就不再进行数据计算的异常判断了。该异常将交由用户程序进行处理。异常发生后要将测试就绪标记清0.

图4

## IR测试模块接口函数流程图



图5是IR测试模块接口函数的流程图，其中比ACW/DCW/GR增加了一个自动换档的操作。该操作会在整个测试过程中进行档位查找，如果找到了合适的档位就会稳定下来。如果自动换档关闭也不会进行换档操作，这是由换档操作函数内部实现的。

图5