ATS 开发文档V2

开发文档变更历史

|  |  |  |
| --- | --- | --- |
| 版本 | 变更时间 | 变更者 |
| V1 | 2014/6/20 | Leo |
| V2 | 2014/10/29 | Leo |

概述:

本文主要介绍ATS测试平台的测试框架、测试测试流程以及测试开发中使用到的类.



目录:

1. 框架介绍
2. 测试流程介绍
3. FlowControl使用的类解析
4. 测试架构



|  |  |
| --- | --- |
| 名称 | 功能描述 |
| GUI | 界面,供用户选择产品,以及显示测试数据 |
| FlowControl | 流程控制,控制产品测试的流程,解析数据 |
| Spec Judge | 规格判定 |
| Algorithm\_Lab | 一些常用的算法 |
| TestModel | 调整以及测试的功能模板 |
| Equipment | 底层的仪器驱动 |
| Data\_access | 与数据库的交互,主要负责数据库的读写 |
| Xml/Database | 存放数据的配置文件或者数据库 |

二、测试流程介绍

整个完整流程如下:

2.1 TestPlan 选择 和 仪器的配置

2.2测试前对仪器以及产品的确认.

2.3测试

2.3.1 数据更新线程

2.3.2 测试线程

2.1 TestPlan 选择 和 仪器的配置



2.2仪器以及产品的确认



2.3测试

2.3.1 数据更新线程



2.3.2 测试线程



1. 类的解析
   1. FlowControl
   2. Environmental\_Control
   3. ConfigInfXml
   4. JudgeSpec
   5. DataIo

3.6 LogQueue

3.1 FlowControll类

.Net Framework4

* 作用

控制测试流程,包含数据解析部分

* 继承层次结构

无

* 命名空间：ATS

程序集：（ATS.exe中）

* 构造函数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| public FlowControll(string DataBaseType) | 根据选择的数据库类型，开启不同的数据库 |

* 属性

|  |  |
| --- | --- |
| 名称 | 说明 |
| public LogQueue<string>QueueInterfaceLog | 界面Log显示 |
| public LogQueue<string> QueueShow | 界面Label 显示 |
| public int ProductionTypeId; | Product 类型ID |
| public int TestPlanId; | 当前TestPlan ID |
| public string ProductionTypeName; | 当前的Product 类型名称 |
| public string CurrentSN; | 当前的产品序号 |
| public string CurrentFwRev | 当前产品的FW版本号 |
| public byte TotalChannel = 0; | 产品通道总数 |
| public byte TotalTempCount = 0; | 产品温度总数 |
| public byte TotalVccCount = 0; | 产品的Vcc总数 |
| public bool TotalResultFlag = true | 当前产品的测试结果 |
| private bool CurrentTestModelResultflag = true | 当前TestModel的测试结果 |
| private bool CurrentConditionResultflag = true | 当前Condition的测试结果 |
| public DataTable DtMyConditionDataTable | 当前TestPlan ID下全部condition信息 |
| private DataTable DtMyDutInf | 存放Dut Manufacatureinf 信息 |
| private bool StopFlag = false | 是否停止测试 |
| private DataTable dtCurrentTestModelTestData | 当前TestModel 测试数据 |
| private DataTable dtCurrentConditionResultData | 当前Condtion 测试数据 |
| private DataTable dtCurrentTestModelList | 当前Condtion 中所有TestModel 信息 |
| private DataTable dtCurrentTestModeParameterlList | 当前TestModel下所有参数信息 |
| private DataRow drCondition | DtMyConditionDataTable 的某行 |
| private DataRow drTestModel | dtCurrentTestModelList的某行 |
| public bool FlagFlowControllEnd = false | Flow Controll 完成测试 |
| public int prosess; | 当前测试进度 |
| public DataTable TotalTestData; | 本次测试全部数据Tabel |
| public bool ReadyforReadflag = false | 是否可以准备被读取 |
| public string StrInterfaceLog | 显示在界面的字段 |
| public string StrSqlLog | 要存放在数据库的字段 |
| public string StrInputLog | TestModel 进入测试前的准备情况字段 |
| public string StrOutputLog | Testmodel完成测试后的输出字段 |
| public EnvironmentalControll pEnvironmentcontroll | 环境控制对象 |
| public string CurrentConditionId | 当前的Condition ID号 |
| public string CurrentLogId | 当前Log信息的Id 号 |
| public string StrCurrentTemp; | 当前温度 |
| public string StrCurrentVcc; | 当前Vcc电压 |
| public string StrCurrentChannel; | 当前通道号 |
| private String StrCurrentConditionName | 当前Condition Name |
| private TestModeEquipmentParameters[]TestModelInputArray | TestModel参数输入列表 |
| private TestModeEquipmentParameters[]TestModelOutputArray | TestModel 参数输出列表 |
| private DataTable DT | 临时DataTable |
| protected DutStruct[] MyDutStructArray | 解析之后的Dut 结构体 |
| public EquipmentList MyEquipList | 仪器对象列表 |
| public EquipmentManage MyEquipmentManage | 仪器对象管理者 |
| public TestModelList MyTestModelList | TestModel 对象列表 |
| public TestModelManage MyTestModelManage | TestModel对象管理者 |
| public SortedList<string, string>EquipmenNameArray; | 仪器名称列表 |
| public DataIO MyDataio | 数据库操作对象 |
| private globalParameters pGlobalParameters | 全局参数 |

* 方法

|  |  |
| --- | --- |
| 名称 | 说明 |
| public FlowControll(string DataBaseType) | 根据选择的数据库类型，选择不同对象 |
| public bool buildConditionList() | 建立ConditionList列表 |
| public DriverInitializeStruct[] GetManufactureChipsetInitialize | 获取Dut Driver初始化结构体数组 |
| public DutStruct[] GetDutManufactureCoefficients() | 获得Dut的Manufacture信息 |
| public bool StartTest() | 进入测试流程 |
| public bool RunTestModel(string StrTestModelName, String StrCurrentTestModelId) | 运行TestModel |
| private void WriteInputLog() | 记录TestModel 输入Log |
| private void WriteOutputLog() | 记录TestModel输出Log |
| public bool SelcetTestModelParameter(string StrCurrentTestModelId) | 完成当前TestModel 输入,输入参数 以及当前TeSTMode DataTable 集合的填充 |
| public string[] ReadProductionTpye() | 获得所有产品类型名称 |
| public string[,] ReadProductionName(int idProductionType) | 获得当前类型下所有产品的名称以及对应ID |
| public bool GetIdFromTpyeTable(string StrProductionTypeName, out int TypeId,out int IMsaId) | 获得当前产品类型获取ID号码以及 MSAid |
| public TestModeEquipmentParameters[] GetCurrentEquipmentInf(string EquipmentID) | 建立当前仪器信息参数列表 |
| public bool GetIdFromProductionNameTable(string StrProductionName, out int ProductionId) | 获得当前产品名称ID号码 |
| public SortedList<string, string> GetEquipmentNameList(int idTestplan) | 建立当前TestPlan下仪器名称列表 |

3.2 EnvironmentalControl类

.Net Framework4

* 作用

切换当前的测试环境

* 继承层次结构

无

* 命名空间：ATS

程序集：（ATS.exe中）

|  |  |
| --- | --- |
| 名称 | 说明 |
| EnvironmentalControll(DUT dut) | 见产品Dut对象传递至环境控制中 |

* 构造函数：
* 属性

|  |  |
| --- | --- |
| 名称 | 说明 |
| private double LastTemp=100 | 上一次的温度，当与上次温度相同的时候，仪器温度达到就不再等待恒定 |
| private Powersupply PS | 需要控制测试板电压的电源对象 |
| private Thermocontroller Xstream | 需要控制温度的热流仪对象 |
| private DUT pDut | Dut 对象 |
| private EquipmentList pEquipmentList | 仪器列表 |
| private String PowersupplyName | 仪器列表中需要控制测试板的电源的名称 |
| private String XstreamName | 仪器列表中需要控温度的热流仪名称 |

* 方法

|  |  |
| --- | --- |
| 名称 | 说明 |
| public bool SelectEquipment(EquipmentList aEquipList) | 讲仪器列表传递给环境控制，并告诉它控制环境的仪器的名字 |
| public bool ConfigEnvironmental(double Temp, double Voltage, byte Channel,String StrAuxAttribles) | 控制当前的温度 电压 和 通道 |
| public bool Dispose() | 释放仪器 |

3.3 ConfigInfXml类

* 作用

切换当前的测试环境

* 继承层次结构

ATS.XmlIO

* 命名空间：ATS

程序集：（ATS.exe中）

|  |  |
| --- | --- |
| 名称 | 说明 |
| public ConfigXmlIO(string StrPath) | 根据路径读取Xml文件 |

* 构造函数：
* 属性

|  |  |
| --- | --- |
| 名称 | 描述 |
| protected XDocument doc | 文件对象 |
| public string filePath | 路径 |
| protected XElement XmlLoadElment | 元素对象 |
| private XElement NodeDataSource | 光源的节点元素 |
| private XElement NodeEquipmentOffset | 仪器参数的节点元素 |

* 方法

|  |  |
| --- | --- |
| virtual public bool BuildXml() | 创建XML 文件 |
| public XElement CreatNode(string NodeName,string XAttributeName, string XAttributeValue) | 添加Node节点 |
| public XElement GetNode(string NodeName, string XAttributeName, string XAttributeValue) | 获得Node对象 |
| public bool AddElement(XElement Node, XElement New\_Element) | 添加Node节点中的元素 |
| public bool ModifyElement(XElement Node, string ElementName, string ElementValue) | 修改节点中的元素 |
| public string ReadElementValue(XElement Myxelemnt, string StrElementName) | 读取节点中的元素的值 |
| public void Dispose() | 释放对象 |

3.4 JudgeSpec类

* 作用

规格判定

* 继承层次结构

无

* 命名空间：ATS

程序集：（ATS.exe中）

* 构造函数：

无

* 属性

无

* 方法

|  |  |
| --- | --- |
| 名称 | 说明 |
| public void Judge(DataTable Indt, TestModeEquipmentParameters[] pOutPutArray, out DataTable Outdt, out bool Breakflag) | 完成规格判定，并填写并返回当前TestModel的DataTable,且返回 |

3.5 DataIO类

* 作用

数据库操作

* 继承层次结构

无

* 命名空间：ATS

程序集：（ATS.exe中）

* 构造函数：

无

* 属性

无

* 方法

|  |  |
| --- | --- |
| 名称 | 说明 |
| public virtual bool OpenDatabase(bool Swith) | 开启或者关闭数据库（true :开启；false:关闭） |
| public virtual DataTable GetDataTable(string sqlCmd, string StrTableName) | 获取当前的DataTable 信息(SqlCmd 为筛选条件) |
| public virtual bool WriterLog(string strConditionID, string StrInf, string aSerialNo, string StartTime, string EndTime, out String strid) | 将信息写入LogRecord 并返回当前Log信息的Id号 |
| public virtual bool WriteResult(string StrInterfaceLogid, DataTable ResultDataTabel) | 将测试结果数据写入数据库中 |
| public virtual DataTable GetDataTable(string sqlCmd, string strTableName) | 根据对应的查询条件返回datatable； |
| public virtual bool UpdateDT() | 更新datatable |
| public virtual bool UpdateDataTable(string sqlCmd, DataTable dt) | 根据查询条件对datatable有更改的部分进行更新到数据库 |
| public virtual long GetPID(string sqlCmd) | 根据查询条件返回PID |
| public virtual bool BlnISExistTable(string tabName) | 查询当前数据库中是否存在指定的表名称 |
| public virtual string[] GetCurrTablesName(string serverName, string dbName, string userName, string pwd) | 根据指定的连接返回该database中所有表名称 |
| public virtual bool WriterSN(int TestPlanID, string StrSN,string StrFwRev, out int snid) | 填写产品序号 Fw版本号到数据库 |

3.6 LogQueue<T>类

* 作用

管理Log 信息队列

* 继承层次结构

System.Collections.Generic. Queue<T>

ATS.Log.LogQueue<T>

* 命名空间：ATS

程序集：（ATS.exe中）

* 构造函数：

无

* 属性

无

* 方法

|  |  |
| --- | --- |
| 名称 | 说明 |
| Enqueue(T obj) | 讲数据存放在列表中，存放的过程中会锁定当前列表 |
| T Dequeue() | 从列表中读取数据，读取的过程中会锁定当前列表 |