**1.功能及界面架构**

|  |  |
| --- | --- |
| **曲线显示**  **采样设置**  **通信设置**  **数据串口**  **控制串口**  **激光网口**  **当前姿态**  **X坐标**  **Y坐标**  **Z坐标**  **旋转角**  **姿态设置**  **向左移 向右移**  **向前移 向后移**  **向上移 向下移**  **顺旋转 逆旋转**  **采样重置**  **采样**  **重置**  **自动采集**  **扫描轴**  **扫描速度**  **扫描距离**  **扫描步长**  **一键扫描**  **历史记录**  **历史扫描目录**  **历史扫描细节** | 非UI成员  QSerialPort portXYZR;//Used by: pushButtonOpenDataPort\_clickded, portXYZR\_readyRead  QSerialPort portCtrl;//Used by: pushButtonOpenCtrlPort\_clickded, portXYZR\_readyRead, motionCtrl\_event  float realtimeX, lasttimeX, initialX;//Used by: portXYZR\_readyRead, pushButtonAutomatic\_clicked, timerContinousScan\_timeout  float realtimeY, lasttimeY, initialY;//Used by: portXYZR\_readyRead, pushButtonAutomatic\_clicked, timerContinousScan\_timeout  float realtimeZ, lasttimeZ, initialZ;//Used by: portXYZR\_readyRead, pushButtonAutomatic\_clicked, timerContinousScan\_timeout  float realtimeR, lasttimeR, initialR;//Used by: portXYZR\_readyRead, pushButtonAutomatic\_clicked, timerContinousScan\_timeout  long timeid;//Used by: pushButtonSample\_clicked, pushButtonAutomatic\_clicked, timerContinousScan\_timeout  long scanid;//Used by: pushButtonAutomatic\_clicked, timerContinousScan\_timeout  CInterfaceLLT \*m\_pLLT = NULL;//After being initialized by initScanDevice, used by timerContinousScan\_timeout  TScannerType m\_tscanCONTROLType;//After being initialized by initScanDevice, used by timerContinousScan\_timeout  uint m\_uiResolution;//After being initialized by initScanDevice, used by timerContinousScan\_timeout  QTimer \*timerContinousScan = new QTimer(this);  int scanMode;//Used by: pushButtonOpenEthernet\_clickded, tabWidget\_tabBarClicked, pushButtonSample\_clicked, pushButtonAutomatic\_clicked, timerContinousScan\_timeout  Q3DScatter \*scatter3D = new Q3DScatter();  QSqlDatabase db = QSqlDatabase::addDatabase("QSQLITE", "connect\_name\_of\_sqlite");  QSqlTableModel \*tableModelCatalog = new QSqlTableModel(this, db);  QSqlQueryModel \*queryModelDetails = new QSqlQueryModel(this); |

**2.功能详细**

**通信设置**

(1)打开/关闭数据串口：构造函数中找出当前所有可用串口；按钮文本在打开与关闭间切换；打开时设置串口波特率；无其它；

(2)打开/关闭控制串口：构造函数中找出当前所有可用串口；按钮文本在打开与关闭间切换；打开时设置串口波特率；无其它；

(3)打开网口或退出应用：调用initScanDevice函数打开扫描设备并初始化与扫描设备相关的数据成员

初始化数据成员：m\_pLLTm\_tscan、m\_tscanCONTROLType、m\_uiResolution

打开成功或失败：成功或失败都直接设置按钮文本为退出扫描系统，且再次单击的确退出程序，即此功能只能执行一次。

打开成功：**不执行resetArray**(刚启动没有数据)**+打开扫描定时器+设置扫描模式为显示**

**当前姿态**

portXYZR\_readyRead函数实时读取下位机数据填充：realtimeX/Y/Z和lineEditX/Y/ZAxis；无其它

**姿态设置**

**(0)判断控制串口是否已打开**

(1)motionCtrl\_event函数组织数据并置portCtrlParams.hasSendSuccess=false后发送数据，然后进入循环等待

portCtrlParams.hasSendSuccess被置为true。若超时则重新发送，最多重发三次。

(2)portCtrl\_readyRead函数实时读取下位机数据，当收到的命令等于portCtrlParams.cmdSended[1]时

置portCtrlParams.hasSendSuccess=true

**采样重置**

(1)切换到该选项卡，首先调用tabWidget\_tabBarClicked执行：**执行resetArray+打开扫描定时器(若扫描设备已打开)+设置扫描模式为显示**

(2)采样功能：**判断控制串口+数据串口+扫描网口是否已打开**

**不执行resetArray+打开扫描定时器+设置扫描模式为采样**(扫描中断函数采样一次后将设置扫描模式为显示)，

在数据库中插入一条扫描目录(置timeid=time(0))

循环检测当扫描模式为显示时退出

(3)重置功能与姿态设置功能实现相同

**自动采集**

(1)切换到该选项卡，首先调用tabWidget\_tabBarClicked执行：**执行resetArray+终止扫描定时器+设置扫描模式为显示**

(2)一键扫描功能：**判断控制串口+数据串口+扫描网口是否已打开**

**执行resetArray+打开扫描定时器+设置扫描模式为保存**(扫描中断函数检测扫描指定长度后设置模式为显示)

在数据库中插入一条扫描目录(置timeid=time(0)，scanid=0, lasttimeX = initialX = realtimeX)，向下位机发送运动命令

循环检测当扫描模式为显示时退出，且关闭扫描定时器，向下位机发送停止命令

**历史记录**

(1)切换到该选项卡，首先调用tabWidget\_tabBarClicked执行：**执行resetArray+终止扫描定时器+设置扫描模式为显示**，重新加载历史扫描目录数据

(2)单击历史扫描目录的某条记录，提取该记录的timeid为条件执行queryModelDetails->setQuery从而获取该记录的历史扫描细节数据

(3)单击历史扫描细节的某条记录，提取该记录的数据到dataProxy，但要首先执行resetArray清空已经有数据