根据设计的UI界面，设计将最终接收到的数字信号可以通过折线的形式展示环境的温湿度变化。初步设计的代码如下：

设置基本的坐标X，Y轴，调用折线图类，

/\*该类折线图类\*/

#include <QApplication>

#include <QButtonGroup>

#include <QIcon>

#include <QPushButton>

#include <math.h>

#include <stdio.h>

static const int DataInterval = 250;

RealtimeDemo::RealtimeDemo(QWidget \*parent) :

QDialog(parent)

{

setFixedSize(1030, 470);

setWindowTitle("Simple Realtime Chart");

// 左侧界面

QFrame \*frame = new QFrame(this);

frame->setGeometry(4, 4, 150, 466);//设置位置和大小前两位是XY的位置，后两位是高和宽

frame->setFrameShape(QFrame::StyledPanel);

//通过setGeometry setFrameShape两个函数设置表头的主要参数

(new QLabel("Temp", frame))->setGeometry(6, 200, 48, 21);

m\_ValueA = new QLabel(frame);

m\_ValueA->setGeometry(55, 200, 59, 21);

m\_ValueA->setFrameShape(QFrame::StyledPanel);

(new QLabel("HUM", frame))->setGeometry(6, 223, 48, 21);

m\_ValueB = new QLabel(frame);

m\_ValueB->setGeometry(55, 223, 59, 21);

m\_ValueB->setFrameShape(QFrame::StyledPanel);

(new QLabel("", frame))->setGeometry(6, 246, 48, 21);

m\_ValueC = new QLabel(frame);

m\_ValueC->setGeometry(55, 246, 59, 21);

m\_ValueC->setFrameShape(QFrame::StyledPanel);

m\_ValueC->hide();

// 设置表的查看器

m\_ChartViewer = new QChartViewer(this);

m\_ChartViewer->setGeometry(165, 8, 600, 270);

connect(m\_ChartViewer, SIGNAL(viewPortChanged()), SLOT(drawChart()));

//X轴获取系统当前时间

m\_nextDataTime = QDateTime::currentDateTime();

// 设置触发频率

QTimer \*dataRateTimer = new QTimer(this);

dataRateTimer->start(DataInterval);

connect(dataRateTimer, SIGNAL(timeout()), SLOT(getData()));

m\_ChartUpdateTimer = new QTimer(this);

m\_ChartUpdateTimer->start(250);//设置固定的速率

connect(m\_ChartUpdateTimer, SIGNAL(timeout()), SLOT(updateChart()));

runPB->click();

}

//将新数据放到数据组

static void shiftData(double \*data, int len, double newValue)

{

memmove(data, data + 1, sizeof(\*data) \* (len - 1));

data[len - 1] = newValue;

}

//通过获取当前时间将新数据转换成实时数据

void RealtimeDemo::getData()

{

QDateTime now = QDateTime::currentDateTime();

{

double currentTime = Chart::chartTime2(m\_nextDataTime.toTime\_t()) + m\_nextDataTime.time().msec() / 1000.0;

p = currentTime \* 4;//X轴函数

shiftData(m\_dataSeriesA, sampleSize, dataA);

shiftData(m\_dataSeriesB, sampleSize, dataB);

shiftData(m\_dataSeriesC, sampleSize, dataC);

shiftData(m\_timeStamps, sampleSize, currentTime);

m\_nextDataTime = m\_nextDataTime.addMSecs(DataInterval);

}

while (m\_nextDataTime < now);

m\_ValueA->setText(QString::number(m\_dataSeriesA[sampleSize - 1], 'f', 2));

m\_ValueB->setText(QString::number(m\_dataSeriesB[sampleSize - 1], 'f', 2));

m\_ValueC->setText(QString::number(m\_dataSeriesC[sampleSize - 1], 'f', 2));

}