BluseLi

我将变得强大，强大到你无法想象； 请记住， From now on, I choose you, not you choose me.

QT学习笔记

For myself

目录

[QT 3](#_Toc34408036)

[环境搭建 3](#_Toc34408037)

[VS2015+QT7.5环境 3](#_Toc34408038)

[Qt SDK下载 3](#_Toc34408039)

[Linux Qt环境搭建 3](#_Toc34408040)

[编码风格 与 编码约定 5](#_Toc34408041)

[窗口 5](#_Toc34408042)

[简单的窗口部件 5](#_Toc34408043)

[QWidget空白窗体 6](#_Toc34408044)

[标准对话框 6](#_Toc34408045)

[信号槽与事件机制 6](#_Toc34408046)

[2D绘图 6](#_Toc34408047)

[UI 6](#_Toc34408048)

[OpenGL开放图形库 9](#_Toc34408049)

[复杂窗口部件 11](#_Toc34408050)

[Qdialog自定义对话框 11](#_Toc34408051)

[Widget窗口 11](#_Toc34408052)

[Textedit 11](#_Toc34408053)

[Listview 11](#_Toc34408054)

[Tableview 11](#_Toc34408055)

[TreeView 11](#_Toc34408056)

[字符操作 11](#_Toc34408057)

[文件操作 12](#_Toc34408058)

[Qt将数据写入文件 12](#_Toc34408059)

[文本文件 12](#_Toc34408060)

[配置文件 12](#_Toc34408061)

[日志文件 13](#_Toc34408062)

[进程编程 14](#_Toc34408063)

[网络编程 14](#_Toc34408064)

[UDP 14](#_Toc34408065)

[TCP/IP 14](#_Toc34408066)

[调用网络接口 14](#_Toc34408067)

[数据库 14](#_Toc34408068)

[SQlite 14](#_Toc34408069)

[Mysql 14](#_Toc34408070)

[外设 16](#_Toc34408071)

[串口 16](#_Toc34408072)

[打印机 16](#_Toc34408073)

[样式表 16](#_Toc34408074)

[QComboBox 16](#_Toc34408075)

[UI 17](#_Toc34408076)

[图表 17](#_Toc34408077)

[汽车表盘 17](#_Toc34408078)

[实例 17](#_Toc34408079)

[飞机大战小游戏 17](#_Toc34408080)

[xxx上位机 17](#_Toc34408081)

[RFID消费系统 17](#_Toc34408082)

[其他 17](#_Toc34408083)

[软件版本 ICON 等 17](#_Toc34408084)

# QT

各种界面库以及功能库（类）：core、gui、network、opengl

工具：IDE工具、界面设计师、帮助文档、国际化工具

许可协议

支持平台

Qt 知识点

PC安装连个版本的Qt 如何设置环境变量

QToolButton 同时显示ICON和文字

QAbstractButton

自定义控件

窗口收缩滚动效果

QSS的常规使用 CSS参考手册 https://www.w3school.com.cn/cssref/index.asp#font

QPushButton样式表 https://blog.csdn.net/caoshangpa/article/details/78549835

QComboBox样式表 https://blog.csdn.net/qq\_40194498/article/details/85067032

QTableWidget和QTableView的区别 https://www.jb51.net/article/182584.htm

QVBoxLayout的用法

主窗口

窗口无边框

窗口半透明

窗口大小固定

窗口圆角

窗口关闭事件

窗口整体拖拽

托盘

停靠托盘

托盘消息

托盘菜单栏

托盘点击事件

托盘Icon闪烁

工程目录结构

单例模式 单例模式，是一种常用的软件设计模式。在它的核心结构中只包含一个被称为单例的特殊类。通过单例模式可以保证系统中，应用该模式的类一个类只有一个实例。即一个类只有一个对象实例。 https://blog.csdn.net/qq21497936/article/details/80046081

图标库使用

https://blog.csdn.net/u011731378/article/details/89575884 https://www.cnblogs.com/pyw0818/p/8052047.html

自定义任务栏

PNG图片iCCP警告

文本编辑控件QsciScintilla

QWidget 不能使用exec

## 环境搭建

### VS2015+QT7.5环境

<https://blog.csdn.net/liang19890820/article/details/53931813>

“toAscii”: 不是“QString”的成员

toAscii 替换为 toLatin1

### Qt SDK下载

<https://download.qt.io/official_releases/qt/5.13/5.13.1/>

### Linux Qt环境搭建

<https://www.jianshu.com/p/7dad1a4483bd>

下载源码

通过共享目录将源码拷进虚拟机

虚拟机设置共享文件夹 怎么访问

如果cd /mnt 下面为内容 则安装[安装vmtools](https://blog.csdn.net/james199209/article/details/54378982)

sudo apt-get install open-vm-tools

mkdir /mnt/hgfs

vmhgfs-fuse .host:/ /mnt/hgfs

Cd /mnt/hgfs 下面就是你虚拟机与宿主主机的共享目录

Sudo ./安装文件 弹出安装界面

断网安装就不会 让你设置代理商啥的

Qt安装位置 opt/ 默认

**sudo apt-get install gcc g++** 安装gcc g++ 编译器

**sudo apt-get install libqt4-dev** 不然编译时会出现错误“**cannot find -lgl**”；

**udo apt-get install build-essential** 编译工具，它可以使得我们的程序知道头文件和库函数放在哪个位置；

如何运行Qt

**cd /opt/Qt5.7.0/Tools/QtCreator/bin/ ./qtcreator OK**

**安装Mysql5.7**

<https://www.cnblogs.com/duanrantao/p/8988116.html>

**官网下载安装包 linux通用**

**将源码拷贝到linux中，Windows与虚拟机中的共享文件夹不行**

**tar -zxvf 压缩包**

**再改个名字 mysql**

**创建mysql用户组及用户**

**groupadd mysql**

**useradd -r -g mysql mysql**

**进入到mysql目录，执行添加MySQL配置的操作**

**cp support-files/my-medium.cnf /etc/my.cnf 或：**

**cp support-files/my-default.cnf /etc/my.cnf**

以上文件都没有

cp support-files/my-medium.cnf /etc/my.cnf 或：

cp support-files/my-default.cnf /etc/my.cnf 重新百度换一种方式安装

<https://www.cnblogs.com/fanshudada/p/9781794.html>

前面相同 添加如下类容

Mysql目录安装位置：/usr/local/mysql  
数据库保存位置：/data/mysql  
日志保存位置：/data/log/mysql

改变目录属有者

cd /usr/local/mysql

chown -R mysql .  
chgrp -R mysql .

chown -R mysql /data/mysql

注意有个点 . 不允许操作前面加sudo

**配置参数**# bin/mysqld --initialize --user=mysql --basedir=/usr/local/mysql --datadir=/data/mysql

Can`t open shared object file 很遗憾

一般而言，有很多so档会在/usr/local/lib这个目录下，所以在/etc/ld.so.conf中加入/usr/local/lib这一行，可以解决此问题。

将/etc/ld.so.conf存档后，还要执行/etc/ldconfig 来更新一下才会生效。

无法修改只读文件 先把文件权限改成666 再修改

chattr -i 文件    让只读文件可编辑

chattr +i 文件   让文件只读

不行

Apt-get install libaio1 libaio-dev

然后OK 有个临时密码 （jUpqfDy1gq:

然后发现没有my-default.cnf

网上找一个

Linux 安装搜狗输入法

dpkg -i 搜狗安装板.deb

sudo [apt-get](https://www.baidu.com/s?wd=apt-get&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao) install -f 搜索并补全依赖关系 重启一下OK

### 编码风格 与 编码约定

<https://wiki.qt.io/Qt_Coding_Style>

<https://wiki.qt.io/Coding_Conventions>

## 窗口

### 简单的窗口部件

QLabel

显示文字 对齐方式 自动换行 设置行高 省略 垂直显示

富文本 html显示

显示图像

显示动画

显示数字

显示超链接

QLCDNumber

不方便获取显示内容 先存储再显示

QAbstractButton

### QWidget空白窗体

### 标准对话框

### 信号槽与事件机制

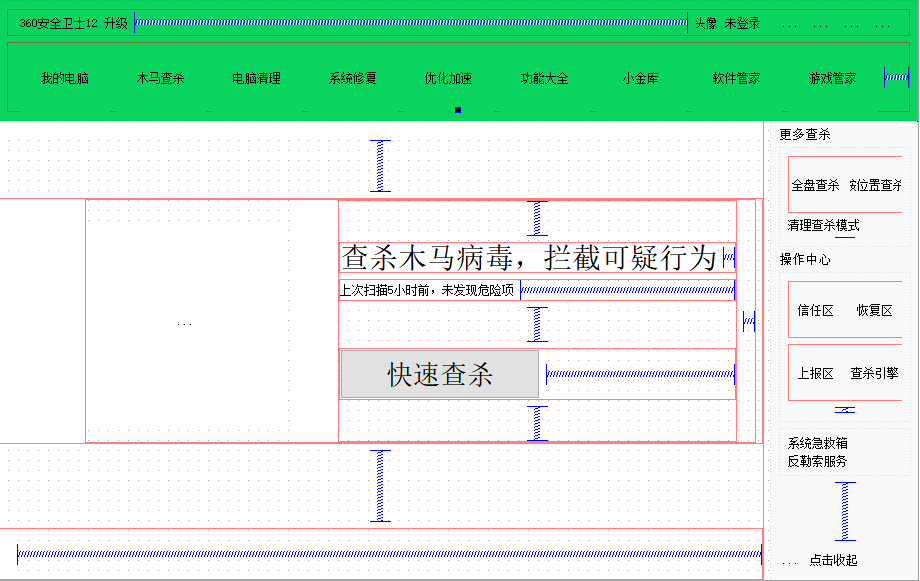
### 2D绘图

### UI

仿360

使用QT的界面设计器编辑直接编辑，UI界面

修改背景窗口背景颜色即可。





整个窗口可移动

void MainWindow::***mousePressEvent***(QMouseEvent \*event)

{

//当鼠标左键点击时.

if (event->button() == Qt::LeftButton)

{

m\_move = true;

//记录鼠标的世界坐标.

m\_startPoint = event->globalPos();

//记录窗体的世界坐标.

m\_windowPoint = this->frameGeometry().topLeft();

}

}

void MainWindow::***mouseMoveEvent***(QMouseEvent \*event)

{

if (event->buttons() & Qt::LeftButton)

{

//移动中的鼠标位置相对于初始位置的相对位置.

QPoint relativePos = event->globalPos() - m\_startPoint;

//然后移动窗体即可.

this->move(m\_windowPoint + relativePos );

}

}

void MainWindow::***mouseReleaseEvent***(QMouseEvent \*event)

{

if (event->button() == Qt::LeftButton)

{

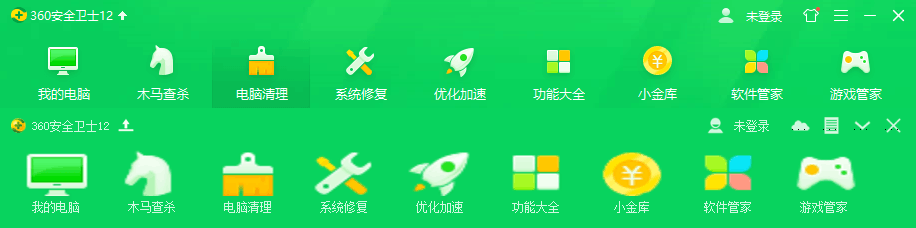
//改变移动状态.

m\_move = false;

}

}

然后找素材，，直接上手PS， 哈哈图片抓得有点粗糙。



用QSS实现

Label 鼠标移动上去改变样式

QLabel {

border-image: url(:/resource/image/light/9-av-upload.png);

}

QLabel:hover{

border-image: url(:/resource/image/dark/9-av-upload.png);

}

QToolButton鼠标悬浮，按下，点击，抬起效果

QToolButton{

border-image: url(:/resource/image/light/6-social-person.png);

border-style: flat;

}

QToolButton:hover{

border-image: url(:/resource/image/dark/6-social-person.png);

}

QToolButton:pressed{

border-image: url(:/resource/image/light/6-social-person.png);

border-style: flat;

}

QToolButton:checked{

border-image: url(:/resource/image/light/6-social-person.png);

}

QToolButton:release{

border-image: url(:/resource/image/light/6-social-person.png);

}

QToolButton 上面显示图片，下面显示文字 文字颜色

QToolButton

{

border-image:url(:/resource/image/my360/360\_Computer.png)32 0 0 0;

border-width:48px 0px 0px 0px;

color: rgb(206, 246, 223);

}

QToolButton:hover

{

border-image:url(:/resource/image/my360/360\_Computer.png)32 0 0 0;

border-width:48px 0px 0px 0px;

color: rgb(206, 246, 223);

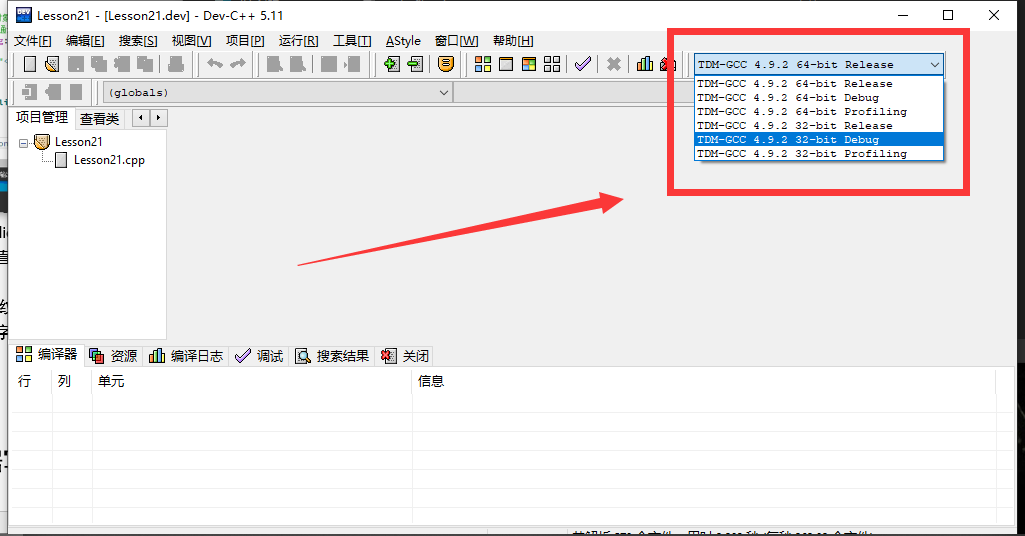
}



### OpenGL开放图形库

DevC++ 版本源码

更改编译器



BOOL 🡪 bool FALSE 🡪 false TRUE 🡪 true

int InitGL(GLvoid)；

报错

[Error] '<anonymous>' has incomplete type

[Error] invalid use of 'GLvoid {aka void}'

解决办法 直接去掉GLvoid 或替换为void

glaux.dll 加载纹理

salflibc.dll 加载字体

//#include <iostream.h>

#include <iostream>

Lesson30

ostream &write(ostream &out) const;

istream &read(istream &in);

报错

[Error] 'ostream' does not name a type

[Error] 'istream' does not name a type

解决

using std::ostream;

using std::istream;

[Error] extra qualification 'TMatrix33::' on member 'inverse' [-fpermissive]

## 复杂窗口部件

### Qdialog自定义对话框

### Widget窗口

### Textedit

显示好看的效果

### Listview

### Tableview

### TreeView

## 字符操作

分割字符串

<https://blog.csdn.net/rgbmarco/article/details/61930518>

字符串中查找另一个字符串

Str.contains("GGA", Qt::CaseSensitive)；

字符串截取某个字符

字符串转时间

<https://blog.csdn.net/u011731378/article/details/79650240>

QString 转 char

QString str;

str.toStdString().c\_str();

QTime转字符

## 文件操作

### Qt将数据写入文件

<https://blog.csdn.net/Touch_Dream/article/details/52839140>

QSTRING QBYTEARRY CHAR\* 的转化

<https://blog.csdn.net/light1028/article/details/7899541>

QT5中文乱码

<https://blog.csdn.net/jh1513/article/details/52331973>

QString::fromLocal8Bit("提示")

Qt数据库相关操作

<https://blog.csdn.net/qq_36038987/article/details/80795626>

### 文本文件

### 配置文件

/\*\*\* 写入密码和锁屏时间到配置文本中 \*\*\*/

//QSettings构造函数的第一个参数是ini文件的路径,第二个参数表示针对ini文件,第三个参数可以缺省

QSettings \*IniWrite = new QSettings("config.ini", QSettings::IniFormat);

IniWrite->setValue("passwd", "123456");

IniWrite->setValue("temperature", "0");

IniWrite->setValue("pressure", "0");

IniWrite->setValue("checktemp", "-15");

//写入完成后删除指针

delete IniWrite;

/\*\*\* 读取配置文件中的密码和锁屏时间 \*\*\*/

//QSettings构造函数的第一个参数是ini文件的路径,第二个参数表示针对ini文件,第三个参数可以缺省

QSettings \*iniRead = new QSettings("config.ini", QSettings::IniFormat);

int calibrateTem = iniRead->value("calibrateTem").toInt();

int calibratePress = iniRead->value("calibratePress").toInt();

int advTemp = iniRead->value("advTemp").toInt();

int prePoint = iniRead->value("prePoint").toDouble();

//读入完成后删除指针

delete iniRead;

### 日志文件

#include <QFile>

#include <QTextStream>

#include <QMutex>

#include <QDateTime>

//日志输出

void **outputMessage**(QtMsgType type, const QMessageLogContext &context, const QString &msg)

{

static QMutex mutex;

mutex.lock();

QString text;

switch(type)

{

case QtDebugMsg:

text = QString("Debug:");

break;

case QtWarningMsg:

text = QString("Warning:");

break;

case QtCriticalMsg:

text = QString("Critical:");

break;

case QtFatalMsg:

text = QString("Fatal:");

}

QString context\_info = QString("File:(%1) Line:(%2)").arg(QString(context.file)).arg(context.line);

QString current\_date\_time = QDateTime::currentDateTime().toString("yyyy-MM-dd hh:mm:ss");

QString current\_date = QString("(%1)").arg(current\_date\_time);

//QString message = QString("%1 %2 %3 %4").arg(current\_date).arg(text).arg(context\_info).arg(msg);

QString message = QString("%1 %2").arg(current\_date).arg(msg);

QDate date=QDate::currentDate();

QString log\_path=LOG\_DIR+QObject::tr("%1.txt").arg(date.toString("yyyy-MM-dd"));

QFile file(log\_path);

file.*open*(QIODevice::WriteOnly | QIODevice::Append);

QTextStream text\_stream(&file);

text\_stream << message << "\r\n";

file.flush();

file.*close*();

mutex.unlock();

}

日志功能

qInstallMessageHandler(outputMessage);

#实现日志输出 保证Release也能正常输出

DEFINES += QT\_MESSAGELOGCONTEXT

### 串口编程

QString("%1").arg(number,2,16,QLatin1Char('0'))

数字转十六进制 补零

## 进程编程

## 网络编程

### UDP

### TCP/IP

### 调用网络接口

## 数据库

### SQlite

### Mysql

完全卸载mysql 亲测有效

<https://www.cnblogs.com/mjhblog/p/10499772.html>

ubuntu彻底卸载mysql并且重新安装

<https://blog.csdn.net/yinjianxiang/article/details/76861719>

<https://blog.csdn.net/Gethin_H/article/details/75090198?utm_source=blogxgwz3>

sudo apt-get install mysql-client mysql-server

Ubuntu 16.04安装Mysql和安装时出现错误解决

<https://blog.csdn.net/gongziwushuang/article/details/96463655>

编译驱动

/opt/Qt5.13.0/5.13.0/Src/qtbase/src/plugins/sqldrivers

正在处理用于 systemd (229-4ubuntu21.27) 的触发器 ...

在处理时有错误发生：

sudo apt-get install libmysqlclient15-dev

公社没让我失望过

<https://www.linuxidc.com/Linux/2011-04/34678.htm>

首先必须安装mysql-devel，不装的话会出现找不到-lmysqlclient的问题,这个在[Ubuntu](https://www.linuxidc.com/topicnews.aspx?tid=2)下名字叫libmysqlclient15-dev，于是

sudo apt-get install libmysqlclient15-dev接下来进入QT存放mysql驱动源码的目录，其中$QDIR为你安装的QT的路径

sudo apt-get install libmysqlclient15-dev

E: 无法定位软件包 libmysqlclient15-dev

<https://blog.csdn.net/txl13109187932/article/details/78846914>

sudo apt-get install libmysqld-dev

又重新找了一篇

<https://blog.csdn.net/x85371169/article/details/79855728>

总结一下，mmp搞得我差点崩盘了

<https://blog.csdn.net/x85371169/article/details/79855728>

## 外设

### 串口

### 打印机

## 样式表

<https://blog.csdn.net/qq_40194498/article/details/85067032>

### QComboBox

/\*未下拉时，QComboBox 样式 背景黑色\*/

QComboBox {border:none;background:#000000;color:#ffffff;

padding-left:30px;font-size:16px "SimHei";}

/\*下拉后的整个下拉窗体样式 背景红色\*/

QComboBox QAbstractItemView {background:#ff0000;color:#ffffff;padding-left:30px;}

/\*下拉箭头区域样式\*/

QComboBox*::*drop-down{

width: 40px;

border-left: 1px solid #ffffff;

}

/\*下拉箭头样式\*/

QComboBox*::*down-arrow {image: url(:/xiala.png);height:30px;width:20px;}

//以下三个需要在代码中设置 ui->comboBox->setView(new QListView());

/\*下拉项样式 背景蓝色\*/

QListView*::*item{padding-left:30px;background:0000ff;color:#ffffff;}

/\*下拉项选中样式（效果不明显）背景绿色\*/

QListView*::*item*:*selected{

color:#DCDCDC;

background:#00ff00;

}

/\*下拉项鼠标悬浮样式 背景灰绿色\*/

QListView*::*item*:hover*{

color:#DCDCDC;

background:#3da79d;

}

# UI

## 图表

## 汽车表盘

# 实例

## 飞机大战小游戏

Qt编程

## xxx上位机

外设使用

## RFID消费系统

系统移植

# 其他

## 软件版本 ICON 等

IDI\_ICON1 ICON "image/AppIcon.ico"

#pragma code\_page(65001)

#if defined(UNDER\_CE)

#include <winbase.h>

#else

#include <winver.h>

#endif

VS\_VERSION\_INFO VERSIONINFO

FILEVERSION 1,0,0,10

PRODUCTVERSION 1,0,0,10

FILEFLAGSMASK 0x3fL

#ifdef \_DEBUG

FILEFLAGS VS\_FF\_DEBUG

#else

FILEFLAGS 0x0L

#endif

FILEOS VOS\_\_WINDOWS32

FILETYPE VFT\_DLL

FILESUBTYPE 0x0L

BEGIN

BLOCK "StringFileInfo"

BEGIN

BLOCK "080404b0"

BEGIN

VALUE "CompanyName", "重庆科瑞分析仪器有限公司"

VALUE "FileDescription", "冷滤点检测上位机"

VALUE "FileVersion", "1.0.0.10"

VALUE "ProductVersion", "1.0.0.10"

VALUE "LegalCopyright", "版权所有 (C) 重庆科瑞分析仪器有限公司"

VALUE "LegalTrademarks", "重庆科瑞分析仪器有限公司"

VALUE "OriginalFilename", "CondensationDetector.exe"

VALUE "ProductName", "重庆科瑞分析仪器有限公司"

VALUE "InternalName", "CondensationDetector.exe"

END

END

BLOCK "VarFileInfo"

BEGIN

VALUE "Translation", 0x804, 1200

END

END