龙芯中科
LOONGSON TECHNOLOGY

人机交互应用解析

提纲

CONTENTS

一、Qt环境实现

龙芯中科
LOONGSON TECHNOLOGY

GUI编程

图形用户界面（Graphical User Interface，简称 GUI，又称图形用户接口）是指采用图形方式显示的计算机操作用户界面。

图形用户界面是一种人与计算机通信的界面显示格式，允许用户使用鼠标等输入设备操纵屏幕上的图标或菜单选项，以选择命令、调用文件、启动程序或执行其它一些日常任务。与通过键盘输入文本或字符命令来完成例行任务的字符界面相比，图形用户界面有许多优点。图形用户界面由窗口、下拉菜单、对话框及其相应的控制机制构成，在各种新式应用程序中都是标准化的，即相同的操作总是以同样的方式来完成，在图形用户界面，用户看到和操作的都是图形对象，应用的是计算机图形学的技术。

一、Qt环境实现

龙芯中科 LOONGSON TECHNOLOGY

PyQT 是什么

pyqt5 是一套 Python 绑定 Digia QT5 应用的框架。它可用于 Python 2 和 3。本教程使用Python 3。Qt 库是最强大的 GUI 库之一。pyqt5 做为 Python 的一个模块,它有 620 多个类和 6000 个函数和方法。

模块	内容描述
QtCore	包含了核心的非 GUI 功能。此模块用于处理时间、文件和目录、各种数据类型、流、URL、MIME 类型、线程或进程。
QtGui	包含类窗口系统集成、事件处理、二维图形、基本成像、字体和文本。
QtWidgets	模块包含创造经典桌面风格的用户界面提供了一套 UI 元素的类。
QtMultimedia	包含的类来处理多媒体内容和 API 来访问相机和收音机的功能。
QtBluetooth	模块包含了查找和连接蓝牙的类。
QtNetwork	包含了网络编程的类,这些工具能让 TCP/IP 和 UDP 开发变得更加方便和可靠。
QtPositioning	包含了定位的类,可以使用卫星、WiFi 甚至文本。
Enginio	包含了通过客户端进入和管理 Qt Cloud 的类。
QtWebSockets	包含了 WebSocket 协议的类。
QtXml	包含了处理 xml 的类,提供了 SAX 和 DOM API 的工具。

一、Qt环境实现

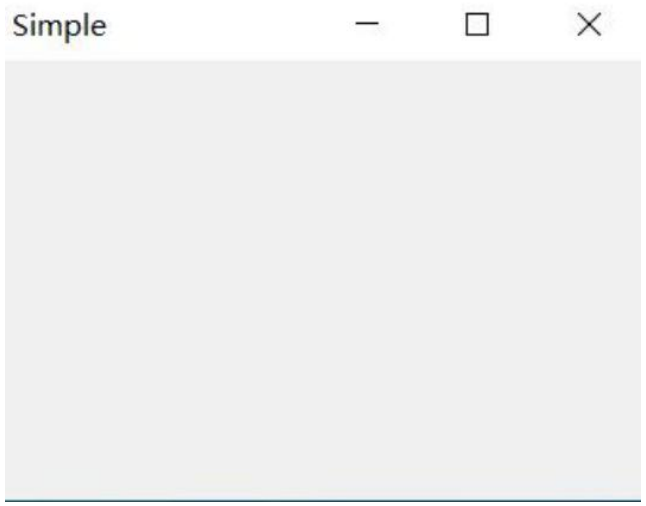
龙芯中科 LOONGSON TECHNOLOGY

开发环境的安装

安装环境依赖
sudo apt-get install qt-sdk
sudo apt-get install qttools5-dev-tools

QT 桌面应用

PyQt5 是一种高级的语言，右边的代码能显示一个小窗口，底层已经实现了窗口的基本功能。




```
import sys
# 基本控件位于 pyqt5.qtwidgets 模块中。
from PyQt5.QtWidgets import QApplication, QWidget
if __name__ == '__main__':
# 每一 pyqt5 应用程序必须创建一个应用程序对象。 sys.argv 参数是一个列表,从命令行输入参数。
app = QApplication(sys.argv)
# QWidget 部件是 pyqt5 所有用户界面对象的基类。 QWidget 提供默认构造函数。默认构造函数没有父类。
w = QWidget()
# resize() 方法调整窗口的大小。这离是 250px 宽 150px 高
w.resize(250, 150)
# move() 方法移动窗口在屏幕上的位置到 x = 300 , y = 300 坐标。
w.move(300, 300)
# 设置窗口的标题
w.setWindowTitle('Simple')
# 显示在屏幕上
w.show()
# 系统 exit() 方法确保应用程序干净的退出
# exec_() 方法有下划线。因为执行是一个 Python 关键词。因此, exec_() 代替
sys.exit(app.exec_())
```

一、Qt环境实现

龙芯中科 LOONGSON TECHNOLOGY

QtDesigner 的使用

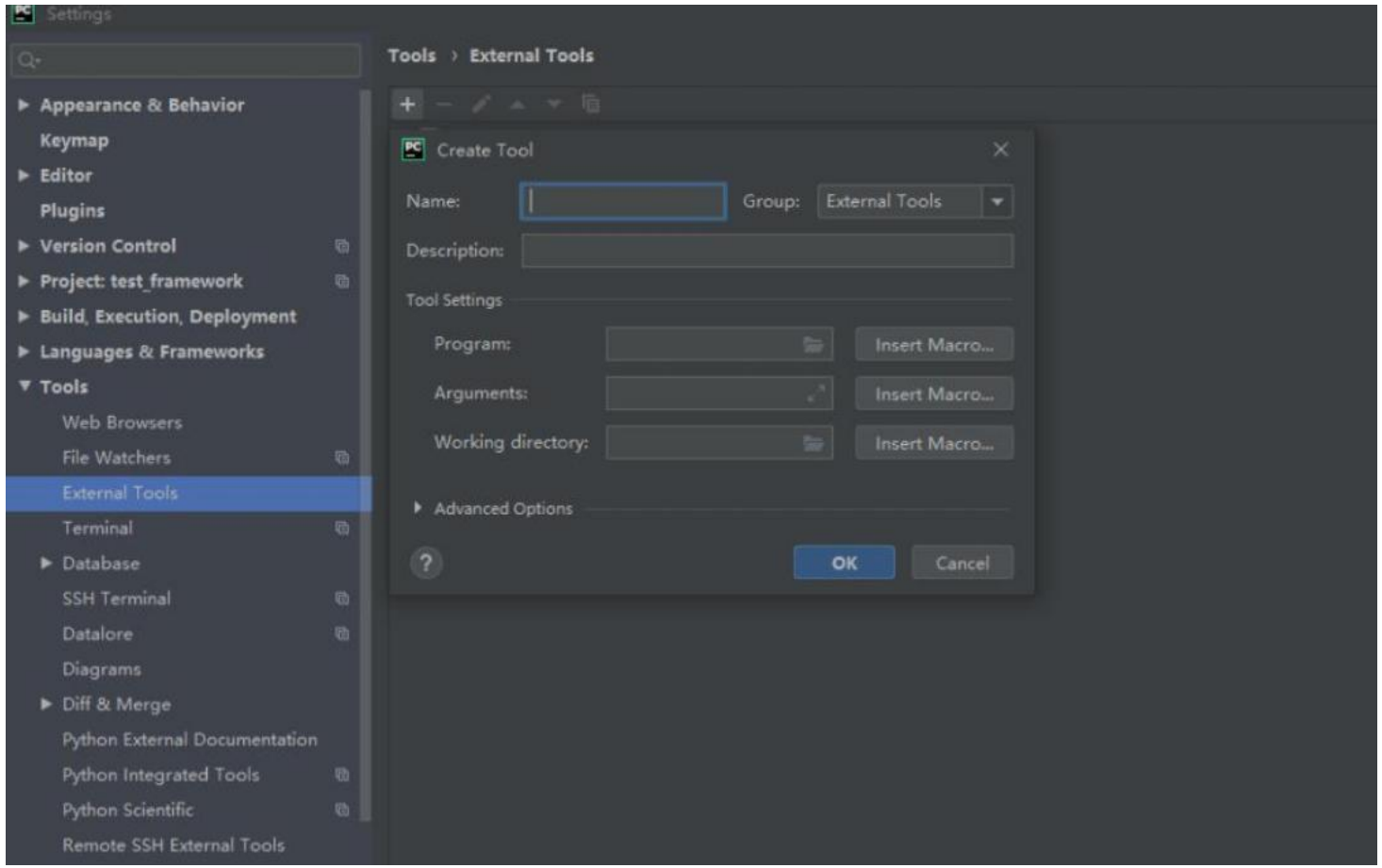
Qt Designer 是 PyQt 程序 UI 界面的实现工具,Qt Designer 工具使用简单,可以通过拖拽和点击完成复杂界面设计,并且设计完成的.ui 程序可以转换成.py 文件供 python 程序调用。

pip 安装 QT

在线安装
pip3 install PyQt5
pip3 install PyQt5-tools
pip3 install paramiko
pip3 install pyinstaller

PyCharm 配置

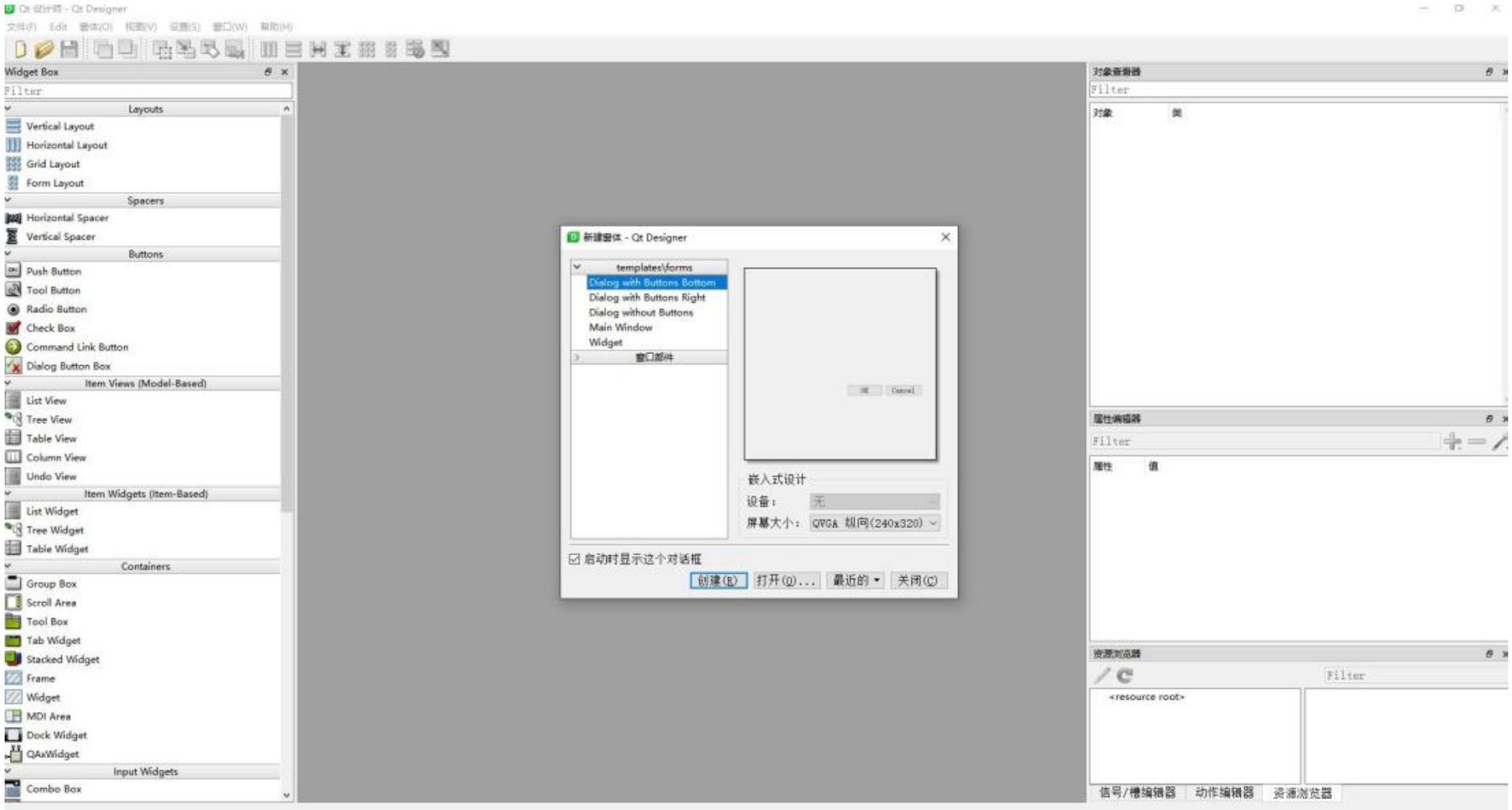
安装好 pyqt5 和 pyqt5-tool 后,还需要对 pycharm 进行配置,添加扩展工具(External Tools)



一、Qt环境实现

龙芯中科 LOONGSON TECHNOLOGY

Qt Designer主界面

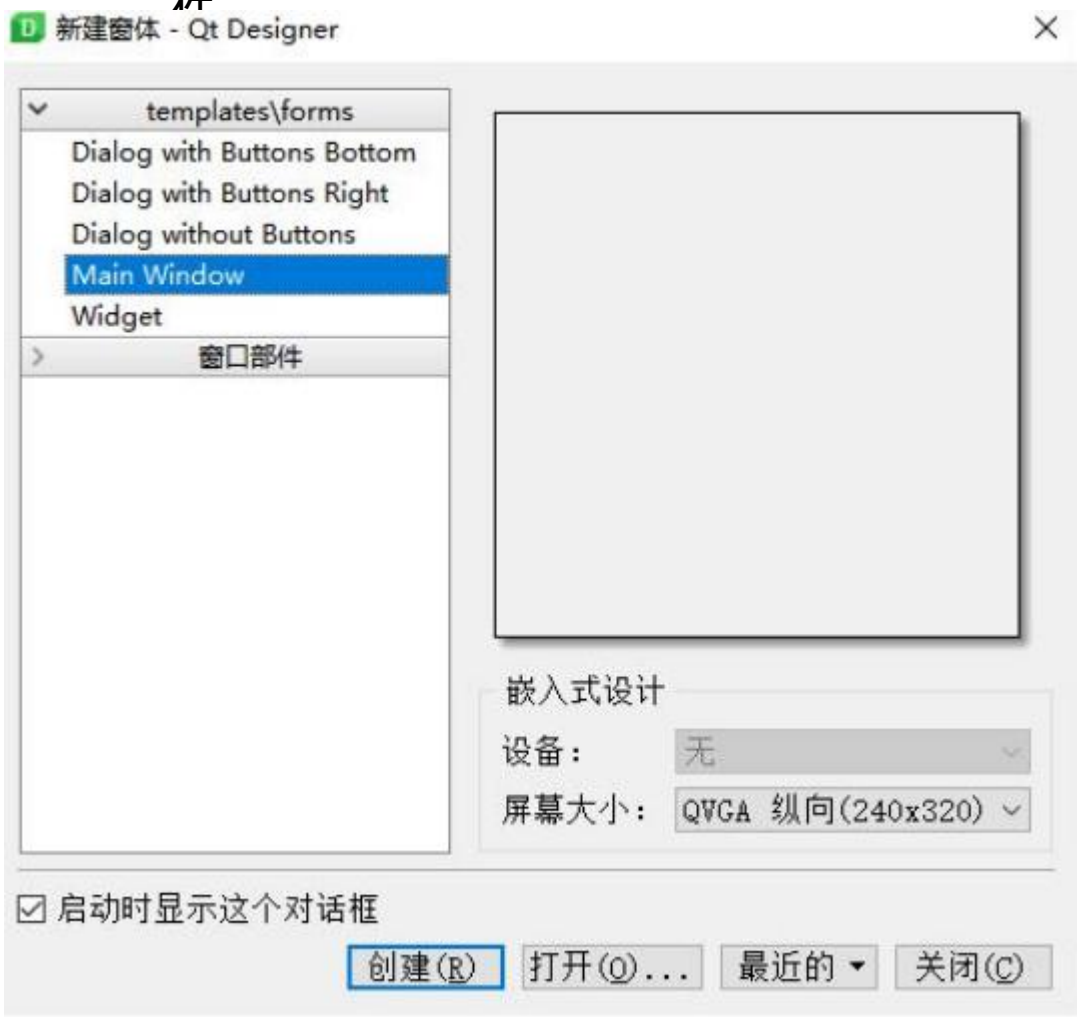


二、Qt应用开发

龙芯中科 LOONGSON TECHNOLOGY

简单应用程序

新建一个 Main Window，在上面放置 1 个 Label 控件和 2 个 pushButton 控件，点击保存为 test.ui 文件



点击第一个按键,在属性编译器上将:
objectName 属性修改为 add_Button (程序调用名)
text 属性修改为 增加(显示的文字)
点击第二个按键,在属性编译器上将:
objectName 属性修改为 dec_Button (程序调用名)
text 属性修改为 减少(显示的文字)

二、Qt应用开发

龙芯中科 LOONGSON TECHNOLOGY

界面与业务逻辑

采用界面与业务逻辑相分离来实现。
也就是通过创建主程序调用界面文件方式实现。这有 2 个好处。第 1 就是实现逻辑清晰。第 2 就是后续如果界面需要变更,只需要重新界面文件即可。新建 main.py 文件程序,调用 test.py 文件显示界面。

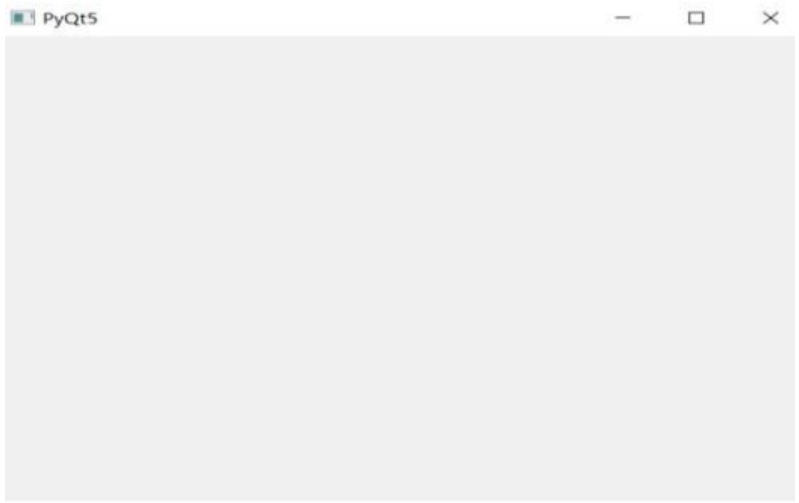

```
#main.py
import sys
# PyQt5 中使用的基本控件都在 PyQt5.QtWidgets 模块中
from PyQt5.QtWidgets
import QApplication, QMainWindow
# 导入 designer 工具生成的 login 模块
from test import Ui_MainWindow
class MyMainForm(QMainWindow, Ui_MainWindow):
def __init__(self, parent=None):
super(MyMainForm, self).__init__(parent)
self.setupUi(self)
if __name__ == "__main__":
# 固定的,PyQt5 程序都需要 QApplication 对象。sys.argv 是命令行参数列表,确保程序可以双击运行
app = QApplication(sys.argv)
# 初始化
myWin = MyMainForm()
# 将窗口控件显示在屏幕上
myWin.show()
# 程序运行,sys.exit 方法确保程序完整退出。
sys.exit(app.exec_())
```

二、Qt应用开发

龙芯中科 LOONGSON TECHNOLOGY

2.1 创建主窗口

使用 setGeometry(left,top,width,height)
方法设置窗口大小。
使用 setWindowTitle(标题)设置窗口标题。
使用 show() 来显示窗口。



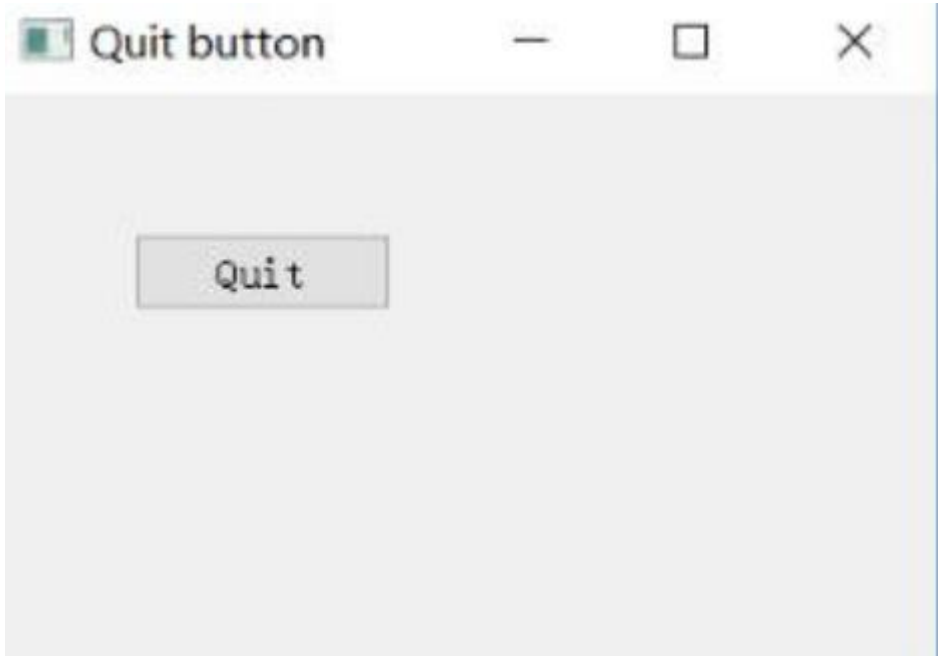

```
# 使用代码创建 PyQt5 窗口
import sys
from PyQt5.QtWidgets import QApplication, QWidget
from PyQt5.QtGui import QIcon
class App(QWidget):
def __init__(self):
super().__init__()
self.title = 'PyQt5'
self.left = 10
self.top = 10
self.width = 640
self.height = 480
self.initUI()
def initUI(self):
self.setWindowTitle(self.title)
self.setGeometry(self.left, self.top, self.width, self.height)
self.show()
if __name__ == '__main__':
app = QApplication(sys.argv)
ex = App()
sys.exit(app.exec_())
```

二、Qt应用开发

龙芯中科 LOONGSON TECHNOLOGY

2.2 退出应用程序

关闭一个窗口可以点击标题栏上的 X。在下面的例子中,我们将展示我们如何通过编程来关闭窗口。



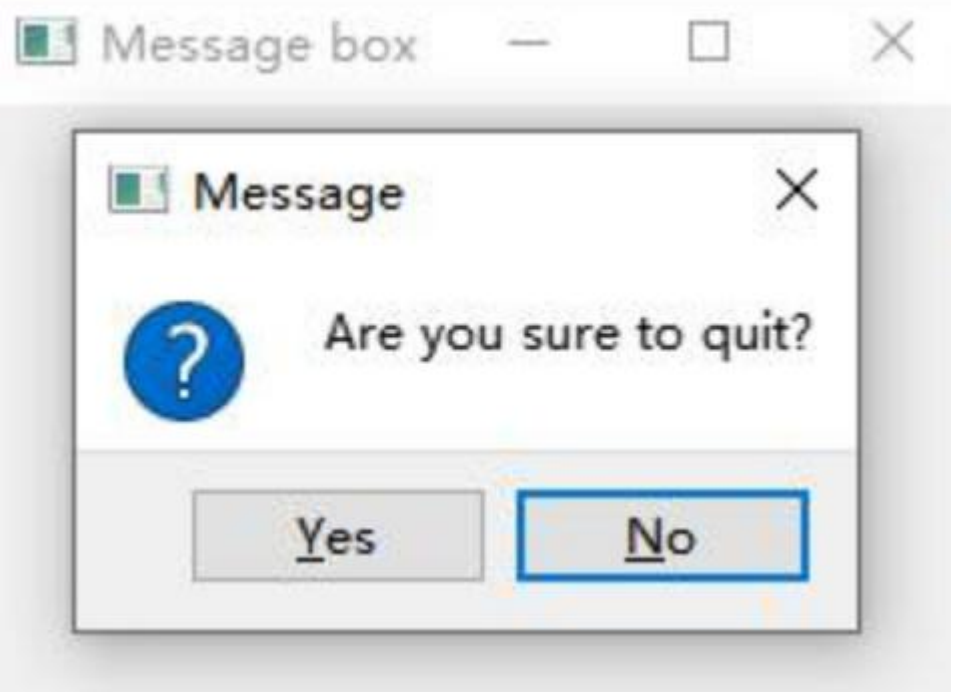

```
import sys
from PyQt5.QtWidgets import QWidget, QPushButton,
QApplication
from PyQt5.QtCore import QCoreApplication
class Example(QWidget):
def __init__(self):
super().__init__()
self.initUI()
def initUI(self):
qbtn = QPushButton('Quit', self)
qbtn.clicked.connect(QCoreApplication.instance().quit)
qbtn.resize(qbtn.sizeHint())
qbtn.move(50, 50)
self.setGeometry(300, 300, 250, 150)
self.setWindowTitle('Quit button')
self.show()
if __name__ == '__main__':
app = QApplication(sys.argv)
ex = Example()
sys.exit(app.exec_())
```

二、Qt应用开发

龙芯中科 LOONGSON TECHNOLOGY

2.3 消息框

我们显示一个消息框,两个按钮:“是”和“不是”。第一个字符串出现在 titlebar。第二个字符串消息对话框中显示的文本。第三个参数指定按钮的组合出现在对话框中。最后一个参数是默认按钮,这个是默认的按钮焦点。



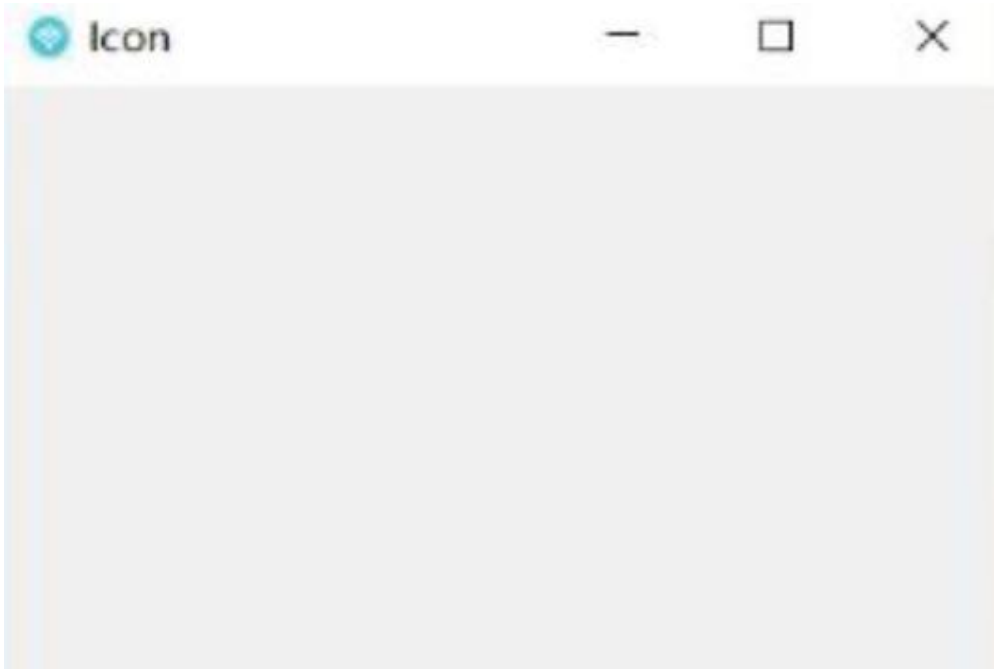

```
class Example(QWidget):
def __init__(self):
super().__init__()
self.initUI()
def initUI(self):
self.setGeometry(300, 300, 250, 150)
self.setWindowTitle('Message box')
self.show()
# 关闭窗口时，触发 QCloseEvent 。重写 closeEvent() 事件处理程序。
def closeEvent(self, event):
reply = QMessageBox.question(self, 'Message',
"Are you sure to quit?", QMessageBox.Yes |
QMessageBox.No, QMessageBox.No)
#单击 Yes 按钮,关闭小部件并终止应用程序。否则忽略关闭事件。
if reply == QMessageBox.Yes:
event.accept()
else:
event.ignore()
```

二、Qt应用开发

龙芯中科 LOONGSON TECHNOLOGY

2.4 设置窗口和应用程序图标

应用程序图标是一个小的图像,通常在标题栏的左上角显示。




```
import sys
from PyQt5.QtWidgets import QApplication, QWidget
from PyQt5.QtGui import QIcon
class Example(QWidget):
def __init__(self):
super().__init__()
self.initUI() # 界面绘制交给 InitUi 方法
def initUI(self):
# 设置窗口的位置和大小
self.setGeometry(300, 300, 300, 220)
# 设置窗口的标题
self.setWindowTitle('Icon')
# 设置窗口的图标,引用当前目录下的 web.png 图片
self.setWindowIcon(QIcon('data/icon.jpg'))
# 显示窗口
self.show()
if __name__ == '__main__':
# 创建应用程序和对象
app = QApplication(sys.argv)
ex = Example()
sys.exit(app.exec_())
```

二、Qt应用开发

龙芯中科 LOONGSON TECHNOLOGY

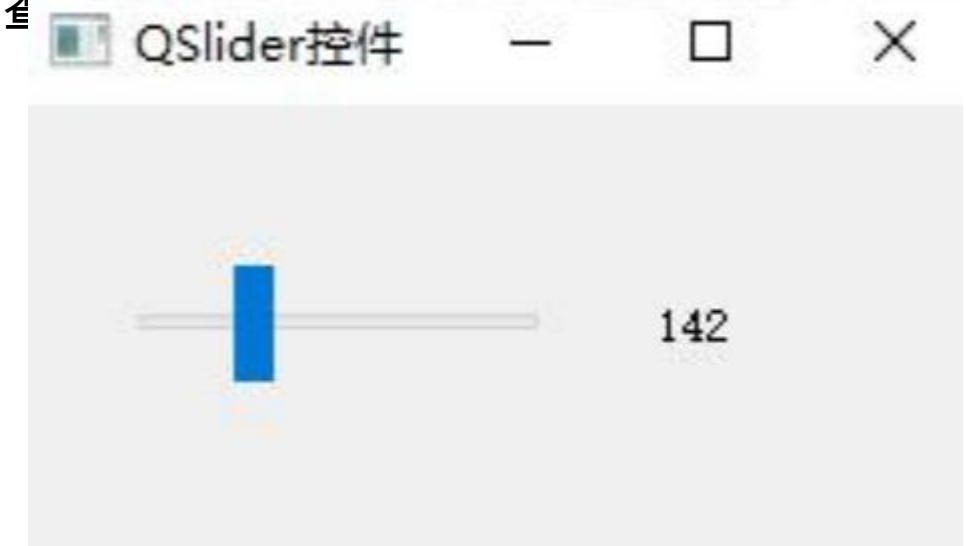
2.5 控件添加提示信息

显示一个提示语,在鼠标移动到按键上时,显示提示。


```
class Example(QWidget):
def __init__(self):
super().__init__()
self.initUI()
def initUI(self):
#这种静态的方法设置一个用于显示工具提示的字体。我们使用 10px 滑体字体。
QToolTip.setFont(QFont('SansSerif', 10))
#创建一个提示,我们称之为 settooltip()方法。我们可以使用丰富的文本格式
self.setToolTip('This is a <b>QWidget</b> widget')
#创建一个 PushButton 并为他设置一个 tooltip
btn = QPushButton('Button', self)
btn.setToolTip('This is a <b>QPushButton</b> widget')
#btn.sizeHint()显示默认尺寸
btn.resize(btn.sizeHint())
#移动窗口的位置
btn.move(50, 50)
self.setGeometry(300, 300, 300, 200)
self.setWindowTitle('Tooltips')
self.show()
```

2.6 QSlider 控件(滑块控件)

QSlider 很少有自己的函数,大部分功能在 QAbstractSlider 中。最有用的函数是 setValue(),用来设置滑块的当前值; triggerAction()来模拟点击的效果(对快捷键有用), setSingleStep()、setPageStep()用来设置步长,setMinimum()和 setMaximum()用于定义滚动条的范围。QSlider提供了一些方法来控制刻度标记。可以使用 setTickPosition()来表示刻度标记的位置,使用 setTickInterval()来指定刻度的间隔;当前设置的刻度位置和间隔可以分别使用 tickPosition()和 tickInterval()函数来查询。



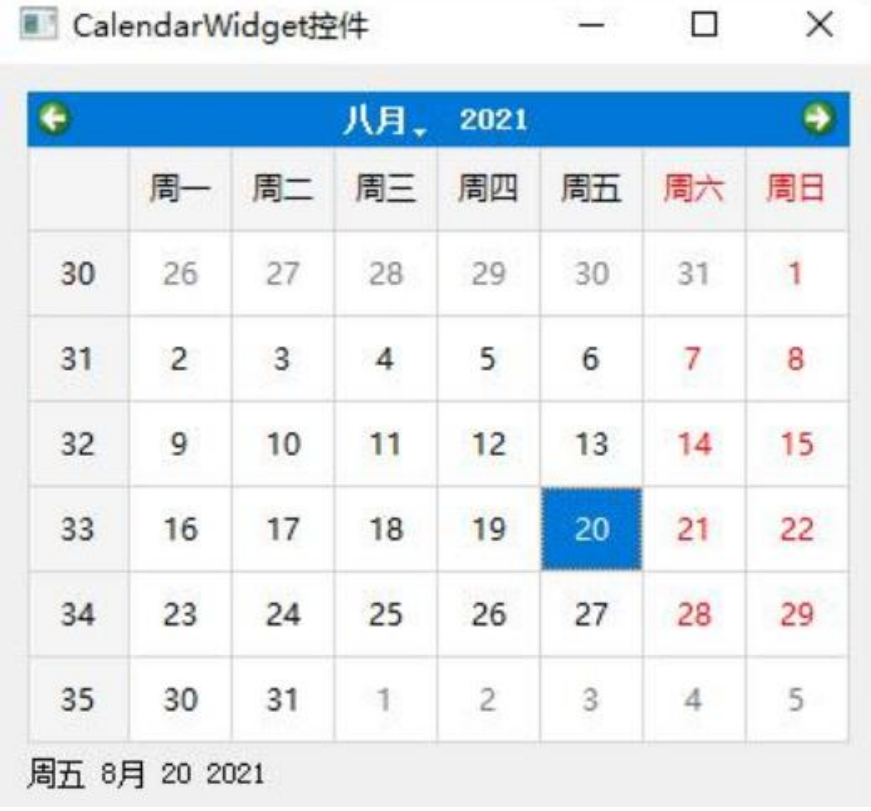

```
class Slider(QWidget):
def __init__(self):
super().__init__()
self.initUI()
def initUI(self):
sld = QSlider(Qt.Horizontal, self)
sld.setMinimum(10)
sld.setMaximum(500)
sld.setGeometry(30, 40, 100, 30)
sld.valueChanged[int].connect(self.changevalue)
self.label = QLabel(self)
self.label.setGeometry(160, 40, 80, 30)
self.setGeometry(300, 300, 280, 170)
self.setWindowTitle('QSlider 控件')
self.show()
def changevalue(self, value):
# 在当前位置显示计划快的位置
self.label.setText(str(value))
if __name__ == '__main__':
app = QApplication(sys.argv)
ex = Slider()
sys.exit(app.exec_())
```

二、Qt应用开发

龙芯中科 LOONGSON TECHNOLOGY

2.6 QCalendarWidget 控件(显示日历控件)

QCalendarWidget 控件适用于显示日历的控件,可以按年月显示日历,通过 setGridVisible 方法可以设置是否在日期中显示网格,通过绑定 clicked 事件,可处理单击日历某一天的动作。



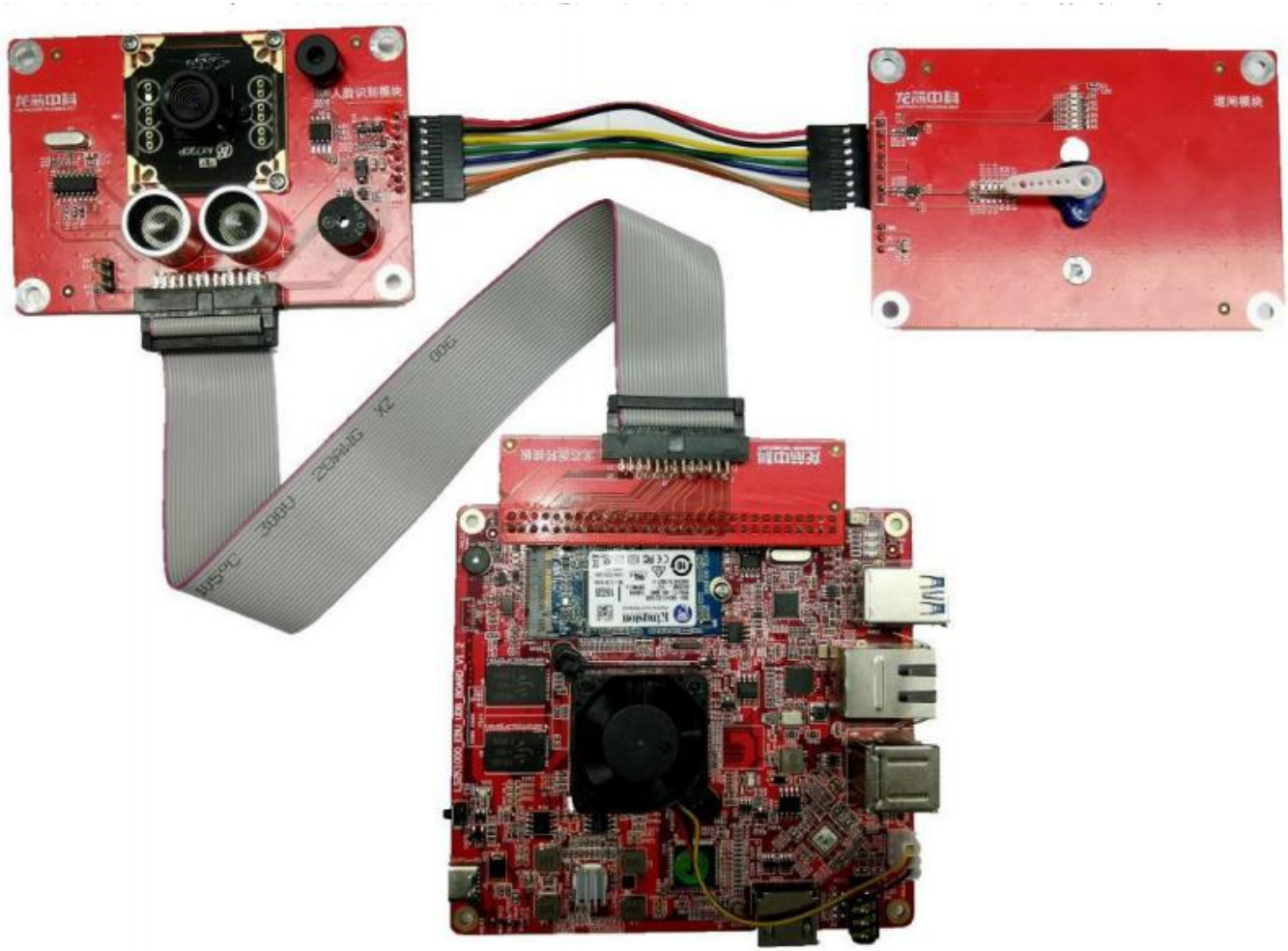

```
from PyQt5.QtCore import Qt
class CalendarWidget(QWidget):
def __init__(self):
super().__init__()
self.initUI()
def initUI(self):
vbox = QVBoxLayout(self)
cal =QCalendarWidget(self)
cal.setGridVisible(True)
cal.clicked[QDate].connect(self.showDate)
vbox.addWidget(cal)
self.lbl = QLabel(self)
date = cal.selectedDate()
self.lbl.setText(date.toString())
vbox.addWidget(self.lbl)
self.setLayout(vbox)
self.setGeometry(300,300,350,300)
self.setWindowTitle('CalendarWidget 控件')
self.show()
def showDate(self,date):
self.lbl.setText(date.toString())
```

三、实战案例：门禁系统开发

龙芯中科 LOONGSON TECHNOLOGY

消息框

使用人脸识别模块和道闸模块模拟真实的门禁系统，能搭建更新用户的人脸数据用于识别，识别成功开启门禁。其他功能还有超声波测距、体温测量、语音播报等。



三、实战案例：门禁系统开发

龙芯中科 LOONGSON TECHNOLOGY

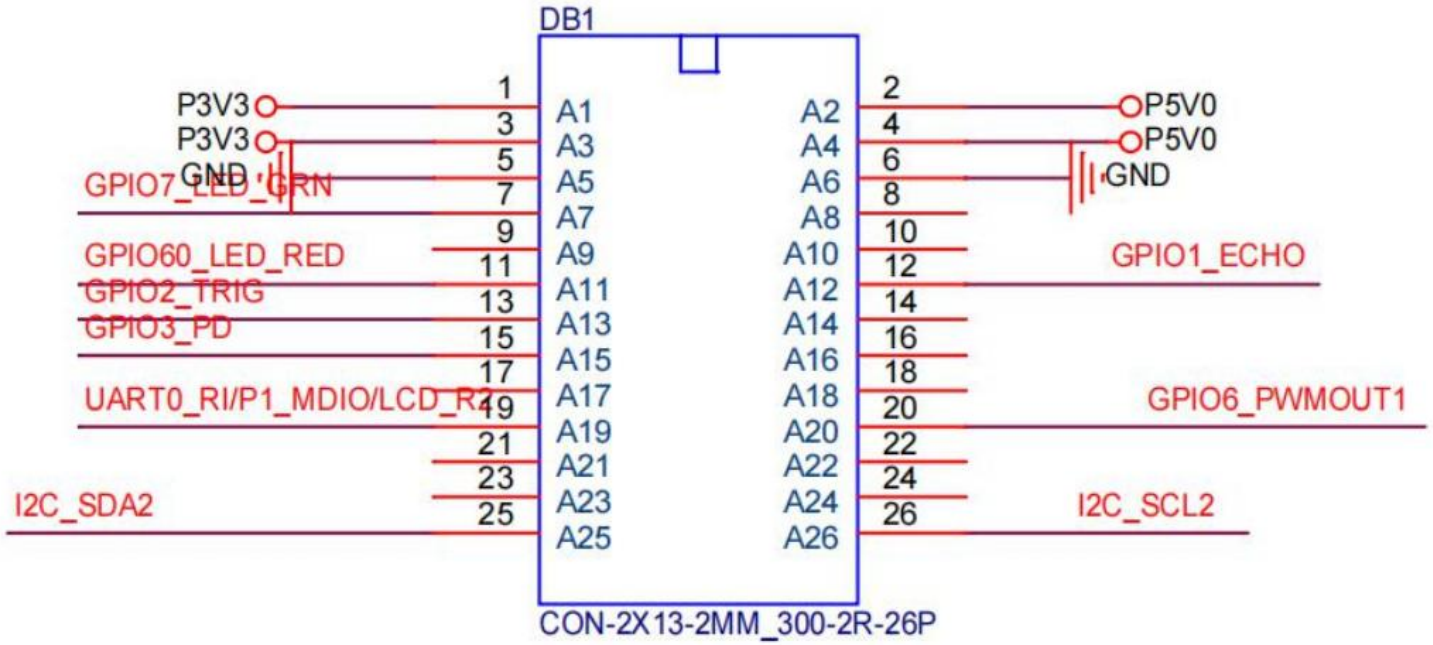
涉及的传感器

使用传感器有:摄像头、红外温度传感器、超声波传感器、舵机。

摄像头AX-8562-V1(38X38)

分辨率 200 万,1280*720 (MJPG 30 fps),视场角 FOV:D54°,像素大小 3μm x 3μm,USB 接口免驱动,焦距可调整,工作电压 5V,使用时直接将其接

pencv 来获取图像。

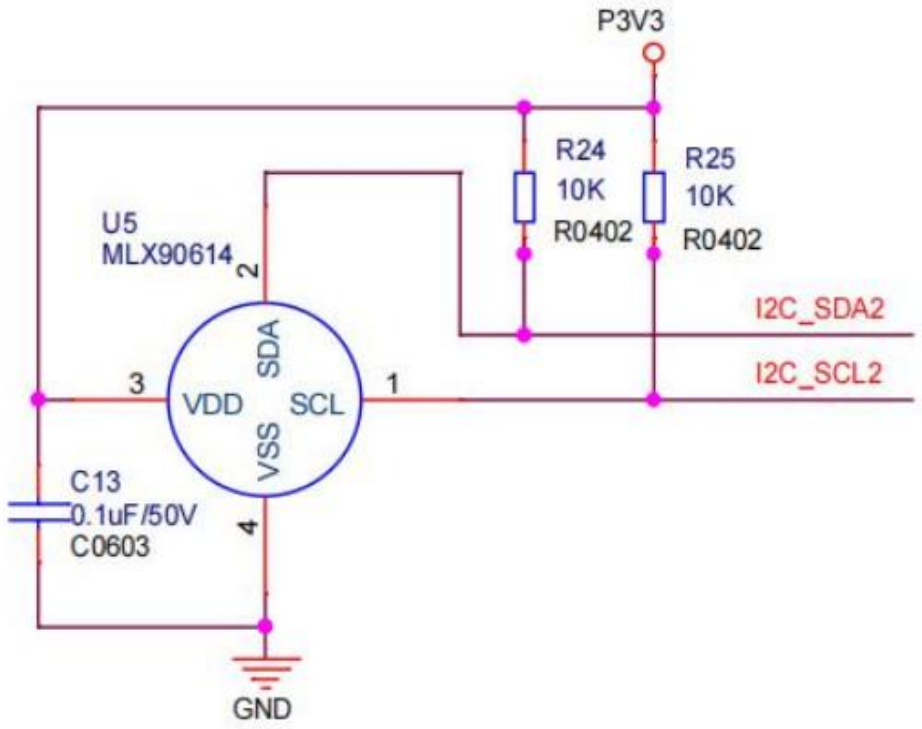


三、实战案例：门禁系统开发

龙芯中科 LOONGSON TECHNOLOGY

红外温度传感器MLX90614

MLX90614出厂校准的温度范围为:环境温度 -40...125°C,物体温度 -70...382.2°C。传感器测量的温度为视场里所有物体温度的平均值。MLX90614 室温下的标准精度为±0.5oC。医疗应用版本的传感器可在人体温度范围内达到±0.1oC 的精度。使用板子上的 I2C 接口来获取温度数据.器件地址: 0x5a 环境温度 Ta 地址:0x06 物体温度 To 地址:0x07



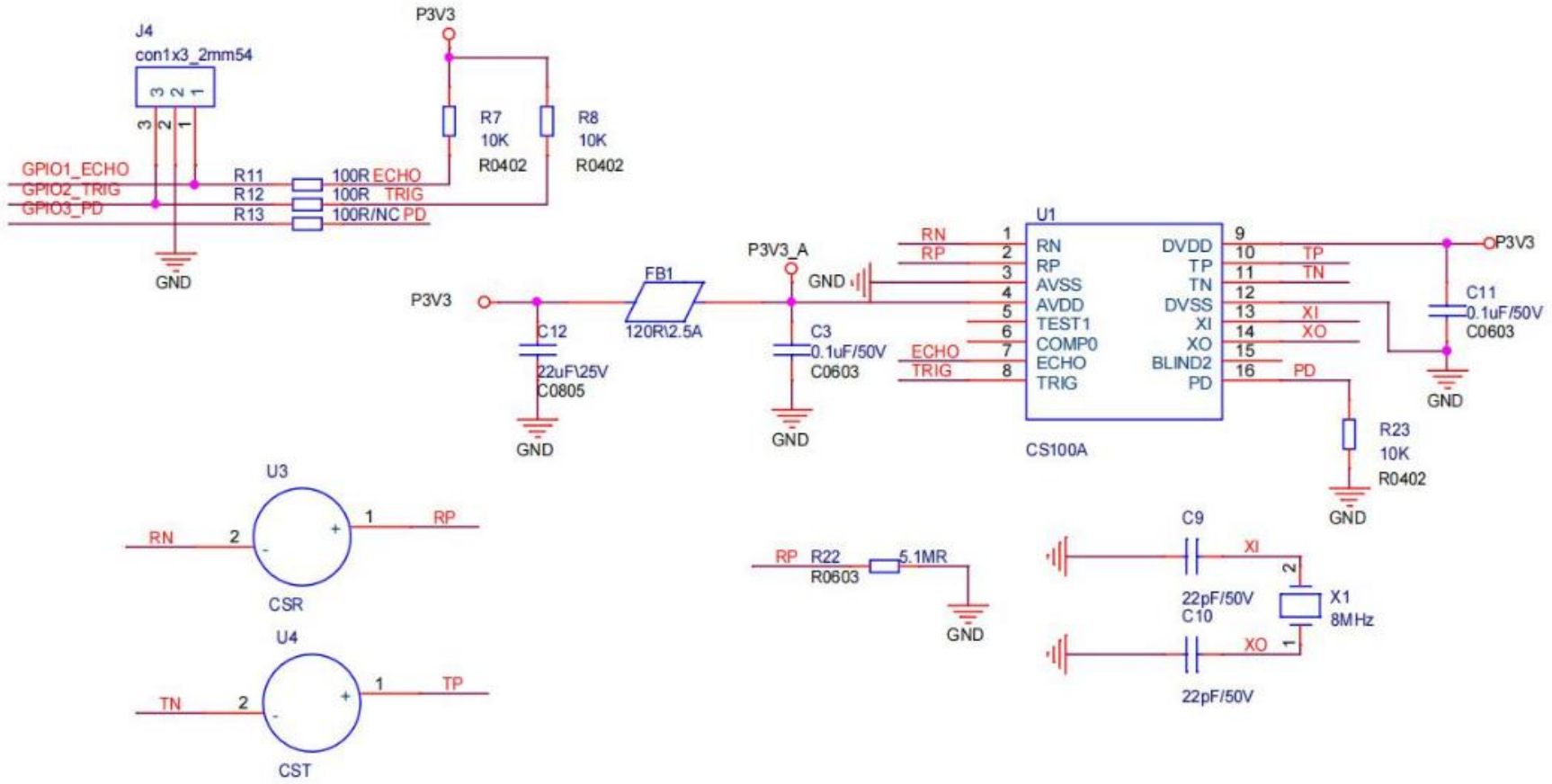
SMBus兼容两线接口[2]						
输入高电平	V_{IH} (Ta, V)	全温度和电压范围	VDD-0.1			V
输入低电平	V_{IL} (Ta, V)	全温度和电压范围			0.6	V
输出高电平	V_{OL}	SDA 引脚为开漏式，全温度和电压范围，Isink = 2mA			0.2	V
SCL 漏电流	I_{SCL}, leak	V_{SCL}=4V, Ta=+85°C			30	uA
SDA 漏电流	I_{SDA}, leak	V_{SDA}=4V, Ta=+85°C			0.3	uA
SCL 电容	C_{SCL}				10	pF
SDA 电容	C_{SDA}				10	pF
从器件地址	SA	出厂默认值		**5A**		hex
唤醒请求	t_{wake}	SDA 低	33			ms
SMBus 请求	t_{REQ}	SCL 低	1.44			ms
超时，低	$T_{imeout,L}$	SCL 低	27		33	**ms**
超时，高	$T_{imeout,H}$	SCL 高	45		55	**us**
确认建立时间	Tsuac(MD)	8-th SCL 下降沿，主控器	0.5		1.5	us
确认保持时间	Thdac(MD)	9-th SCL 下降沿，主控器	1.5		2.5	us
确认建立时间	Tsuac(SD)	8-th SCL 下降沿，从动器	2.5			us
确认保持时间	Thdac(SD)	9-th SCL 下降沿，从动器	1.5			us

三、实战案例：门禁系统开发

龙芯中科 LOONGSON TECHNOLOGY

超声波传感器 CS100A

在 TRIG 管脚输入一个 10US 以上的高电平(一般建议 50US 左右),芯片(TP,TN管脚)便可发出 8 个 40KHZ 的超声波脉冲,然后(RP,RN)检测回波信号。当检测到回波信号后,通过 ECHO 管脚输出。



三、实战案例：门禁系统开发

龙芯中科 LOONGSON TECHNOLOGY

超声波传感器 CS100A

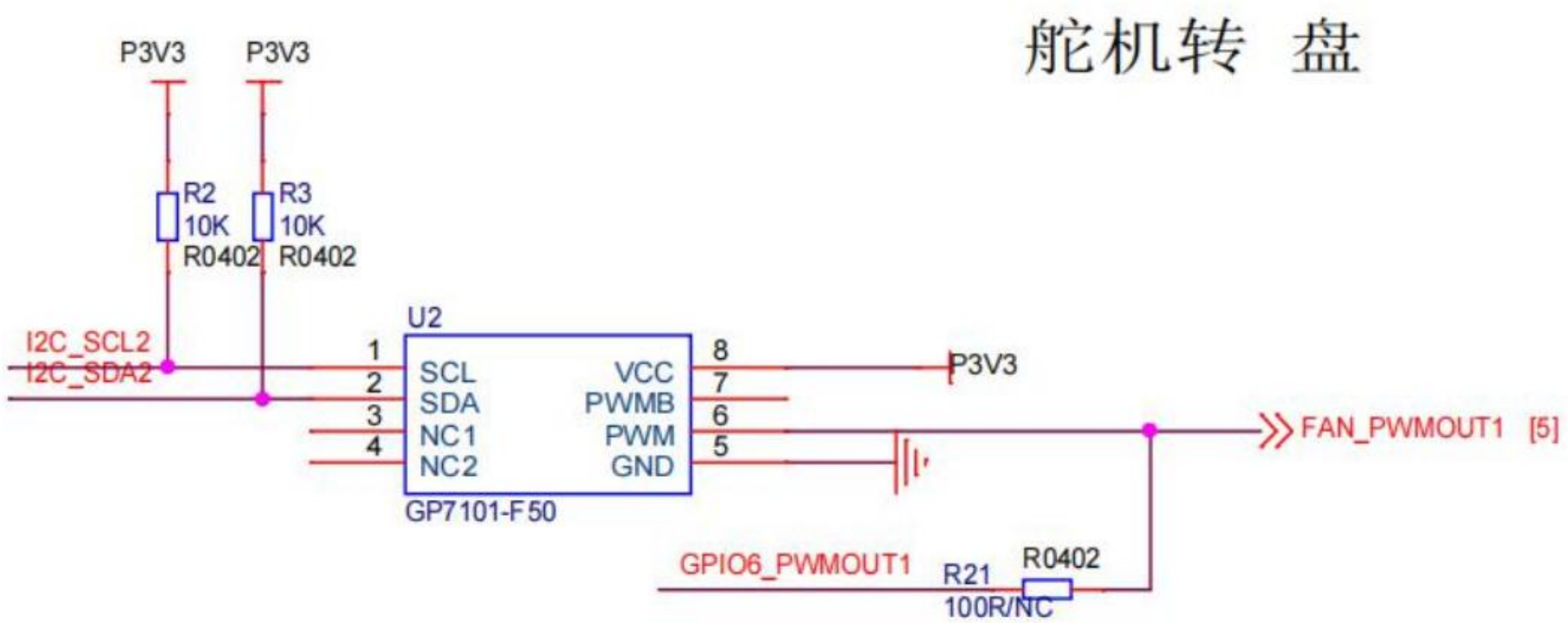
根据 ECHO 管脚输出高电平的持续时间可以计算距离值。当测量距离超过测量范围(约 5.6 米)时,CS100A 仍会通过 ECHO 管脚输出高电平的信号,高电平的宽度约为 33ms。

三、实战案例：门禁系统开发

龙芯中科 LOONGSON TECHNOLOGY

舵机

舵机 FS90 采用 PWM 控制角度,这里使用了 GP7101,I2C 信号转 PWM 芯片来出书脉冲波形。通过 I2C 进行数据输入,线性转换成 0%-100%占空比的 PWM 信号输出。



三、实战案例：门禁系统开发

龙芯中科 LOONGSON TECHNOLOGY

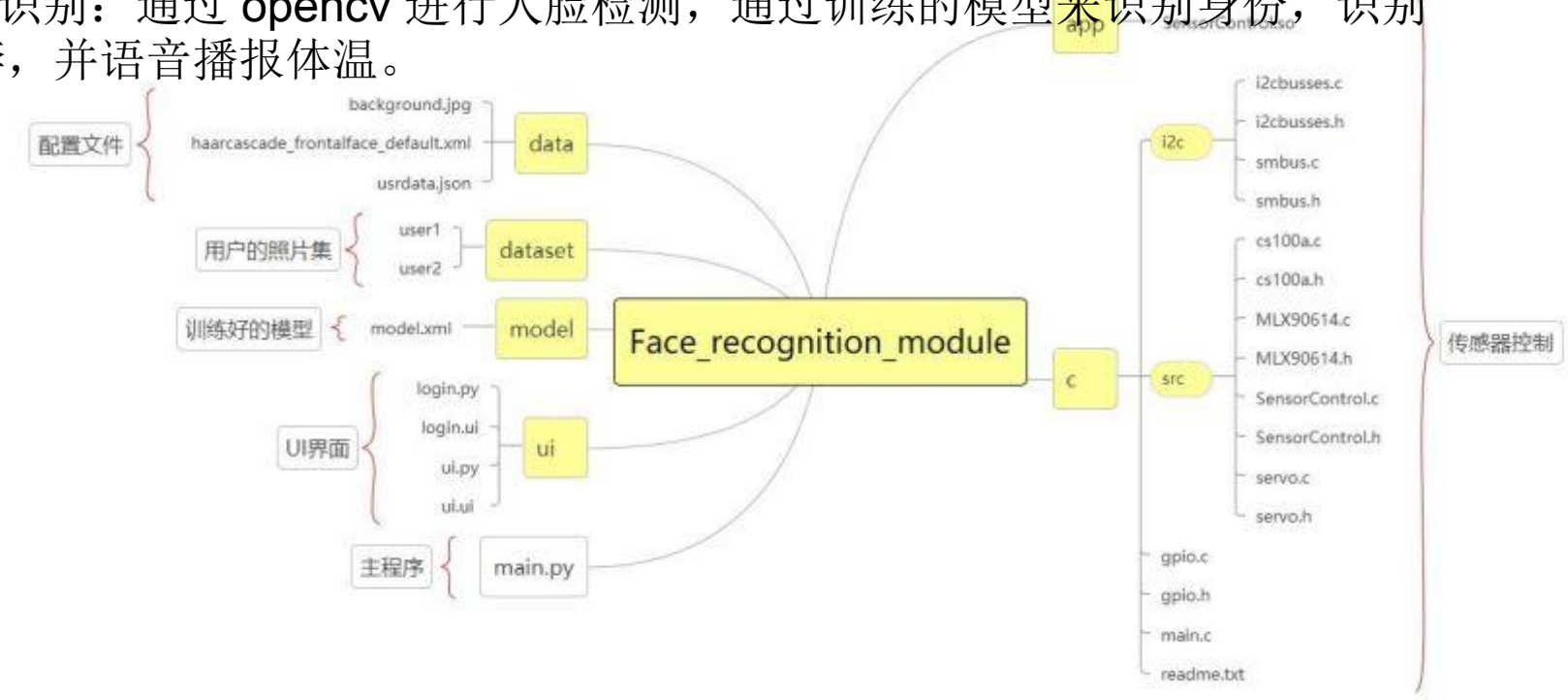
软件部分

软件部分总共分为三个部分：传感器、QT 界面设计、opencv 人脸识别相关程序。

传感器：温度、超声波、舵机控制，将其集成编译成 SO 文件以供主程序调用。

QT 界面：使用 QT 设计一个可视化界面，具有一些简单的用户操作如用户登录、添加 用户、训练模型等按键进行交互。

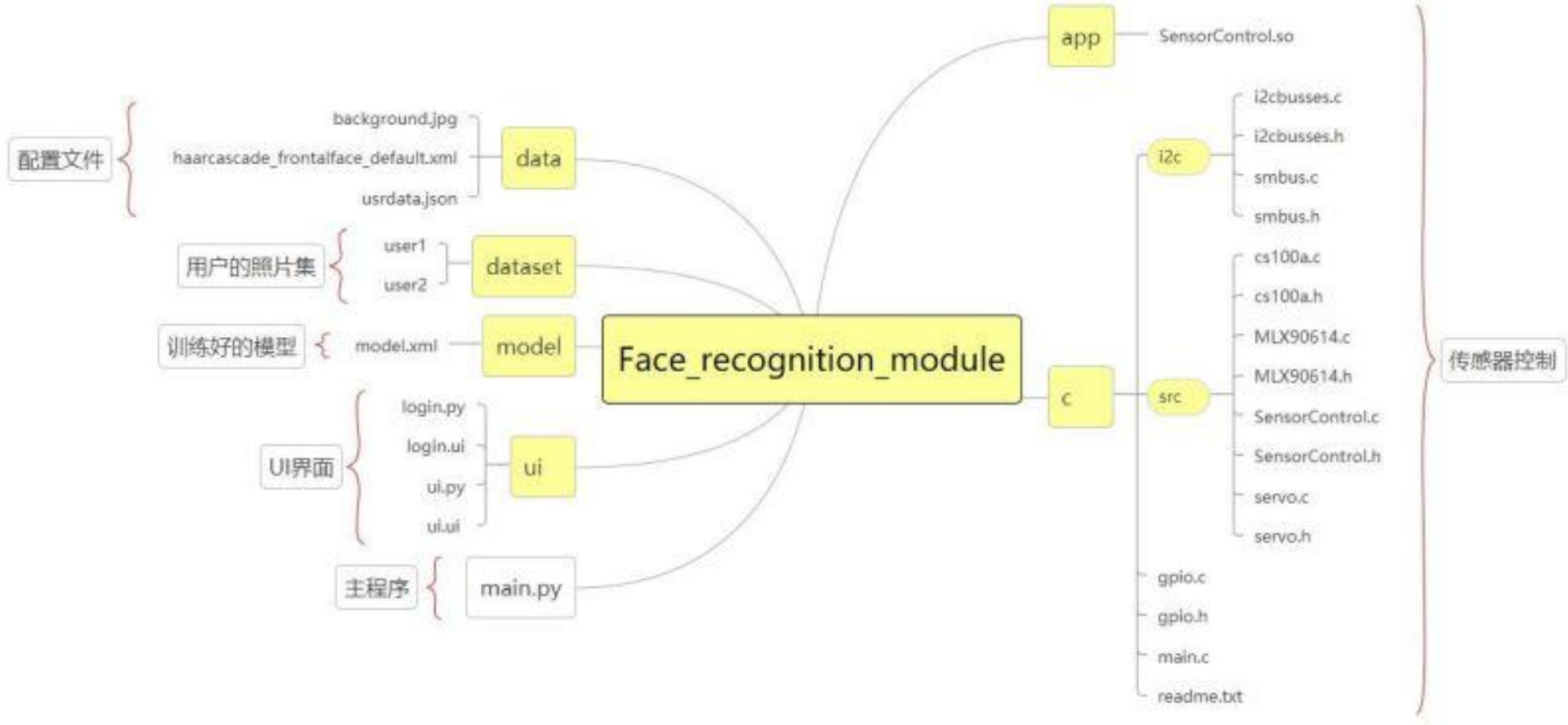
opencv 人脸识别：通过 opencv 进行人脸检测，通过训练的模型来识别身份，识别成功则打开门禁，并语音播报体温。



三、实战案例：门禁系统开发

龙芯中科 LOONGSON TECHNOLOGY

软件部分



四、小任务

龙芯中科
LOONGSON TECHNOLOGY

- 1、Qt应用开发
- 2、门禁系统

感谢观看！