

数字微流控生物芯片模拟界面 说明文档

程序框架

block 类：代表芯片上各个网格

三个布尔成员变量, 代表是否为入口、出口或障碍, 一个 `vector<int>` 与一个 `vector<QColor>`, 表示在某一时刻被液滴沾染的颜色

成员函数 `addTained(int,QColor)` 用于增加某一时刻沾染的颜色; 成员函数 `wipe (int)` 用于消除某一时刻及之前沾染的颜色

block	bool in				
	bool out				
	Bool isBarrier				
	vector<int>times	times[0]	times[1]
	vector<QColor>colors	colors[0]	colors[1]

drop 类：代表液滴（不包括清洗液滴）

两个 `int` 成员代表开始出现的时间和消失的时间, 一个 `vector<int>` 与一个 `vecor<QPoint>` 表示每次发生位置改变的时间与对应位置, 两个布尔成员表示是否发生分裂、合并, 三个 `QPoint` 成员表示分裂与合并的目标位置, 一个 `QColor` 成员表示液滴颜色 (初始化时随机设定)

函数 `pos (int)` 返回一个 `QPoint*` 对象, 表示某时刻液滴位置, 若该时刻液滴未出现或已消失则返回空指针; 函数 `setEnd(int)` 设置消失的时刻, 函数 `setPath(int,QPoint&)` 设置液滴某时刻的位置, 函数 `setSplit(QPoint&,QPoint&)` 和 `setMerge(QPoint&)` 分别设置分裂、合并的目标位置

drop	bool splitting	int startTime	QColor color	
	bool merging			
	QPoint splitingTarget1			
	QPoint splitingTarget2	int endTime		
	QPoint mergingTarget			
	vector<int>times			
	vector<QPoint>path			

state 类：记录清洗时对应时刻的状态并计算清洗液滴经过目标点的路径

四个 `int` 成员, 分别代表对应的时刻、清洗液滴最大步数 (默认为 1000), 行数、列数, 一个 `int` 二维数组代表清洗液滴到达某点的步数 (无法到达设为 -1)

函数 `setBarrier(int x,int y)` 将某点设为不可到达 (否则违反约束条件), `addDangerZone(int x,int y)` 将某点及其周围 8 个点设为不可到达, `reset()` 将二维数组中所有可到达点的值设为最大, `bs(QPoint,QPoint)` 计算两点之间的路径并返回一个 `vector<QPoint>`, 无路径则返回空 `vector`,

getPath(QPoint)和 getPath(vector<QPoint>target)计算清洗液滴需要经过某个点或某些点的路径并返回一个 vector<QPoint>

state	int t
	int mx
	int X
	int Y
	int p[20][20]

WashDrop 类：代表清洗液滴

一个 int 成员代表液滴出现时间，一个 QColor 成员代表液滴颜色（默认为 QColor(0, 128, 255)），一个 vector<QPoint>记录清洗液滴的移动路径

WashDrop	int t	QColor color
	vector<QPoint>path	

GBK 类：用于字符串编码转换

静态成员函数 FromUnicode(const QString&)将 QString 转换为 string，ToUnicode(conststring&)则反之

InitDialog：QDialog 的派生类，作为初始化的弹出对话框

两个 int 成员代表读取的行数、列数，两个 QString 代表读取的入口与出口

InitDialog	int x	int y
	QString ins	QStiring outs

信号 getParameters(int,int,QString,QString)将获取的参数传递给主界面，槽函数 OnOK(),OnCancel()分别与界面的确定与取消按键连接，OnOK()函数会发送 getParameters()信号



Chip: QMainWindow 的派生类，作为主界面

一个 vector<Drop>代表所有(非清洗)液滴，一个 vector<QPoint>[1200]代表各个时刻需要清洗的网格，一个 WashDrop* 作为当前的清洗液滴，一个 block**作为包含所有网格的二维数组，两个 int 代表行数、列数，两个 int 代表正常时刻、清洗时清洗液滴自身的时刻（清洗时相当于正常时间暂停），两个布尔值代表是否已初始化、是否需要清洗，另有一些控制绘图的数据成员

Chip	vector<Drop>drops	int t
	vector<QPoint>[1200]	int tWash
	block** blocks	bool initialized
	WashDrop* curWash	bool wash
	

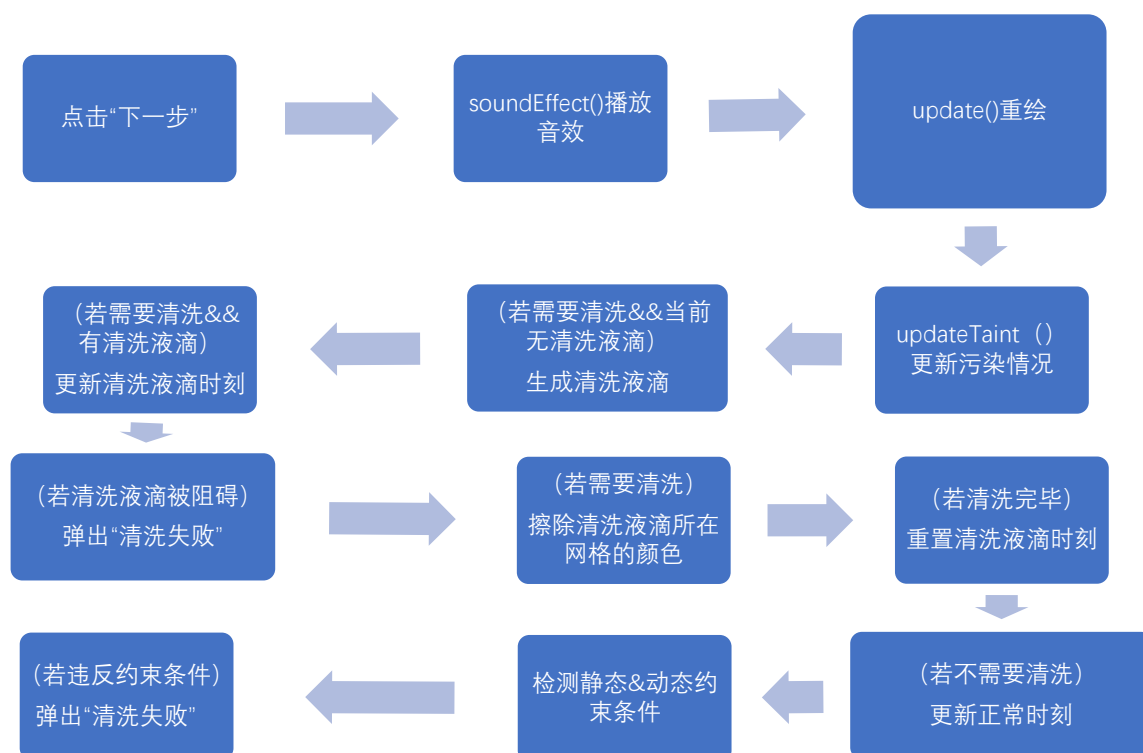
槽函数:

OnInit()与界面的 actionInit 动作相连, 用于生成 InitDialog 对象, 在此对象执行完毕时对行数、列数、入口、出口进行初始化

OnWash(bool)与界面的 wash 选择按钮相连, 点击“清洗”时触发, 设置是否需要清洗

OnRead()与界面的读取 (read) 按钮相连, 点击“读取”时触发, 根据输入的文件名读取文件, 若文件不存在则报错, 根据读取内容生成 Drop 对象并设置其路径, 同时更新网格的污染情况。若需要清洗则调用 setWash () 函数, 当某一时刻有网格被重复污染时将其加入此时刻对应的需要清洗网格的 vector 内, 若发现有无法清洗的网格则更新清洗失败的时间 (默认为 12000)

OnNextStep()与界面的下一步 (nextStep) 按钮相连, 点击“下一步”时触发, 若正常时刻已到达终点则返回。调用 soundEffect()函数, 根据情况选择是否播放移动、分裂、合并的音效, 调用 update()进行重绘, 调用 updateTaint()函数对每个网格更新污染情况。若当前时刻有需要清洗的网格且当前清洗液滴为空则生成一个 state 对象,并调用其 getPath () 函数获取清洗液滴路径, 从而生成一个清洗液滴, 若当前清洗液滴不为空则令清洗液滴的时刻加一, 若清洗液滴的路径被障碍阻挡则弹出“清洗失败!”消息框并返回, 否则对清洗液滴所在的网格调用 wipe(int)擦除此前沾染的颜色, 若清洗液滴已清洗完毕则重置清洗液滴的时刻。若不需要清洗或者当前时刻没有需要清洗的液滴则令正常时刻加一。若到达此前设置的清洗失败的时刻则弹出“清洗失败!”消息框并返回。调用 staticCheck()和 dynamicCheck()分别检测静态和动态约束条件, 若某一条件不符合则弹出“违反约束条件!”对话框



OnLastStep()与界面的上一步 (lastStep) 按钮相连, 点击“上一步”时触发, 若正常时刻已到达起始点则返回, 否则令正常时刻减一, 调用 update()函数

OnStart()与界面的全部播放(start)按钮相连, 点击“全部播放”时触发, 将 OnNextStep()与一个时间间隔为 1 秒的 QTimer 相连, 每秒更新一次

OnReset()与界面的复位(reset)按钮相连, 点击“复位”时触发, 将正常时刻重置为起点并调用 update()

OnHalt()与界面的暂停(pause)按钮相连, 点击“暂停”时触发, 停止与 OnNextStep()相连的 QTimer 的计时

事件函数:

paintEvent(QPaintEvent*)若未初始化则返回, 调用 drawBlocks()画出网格, 调用

drawInOut()画出液滴入口、出口, 调用 drawDrops()画出 (非清洗)液滴,

调用 drawTaints()画出网格的污染情况, 调用 drawBarriers()将设为障碍的网格画成黑色,

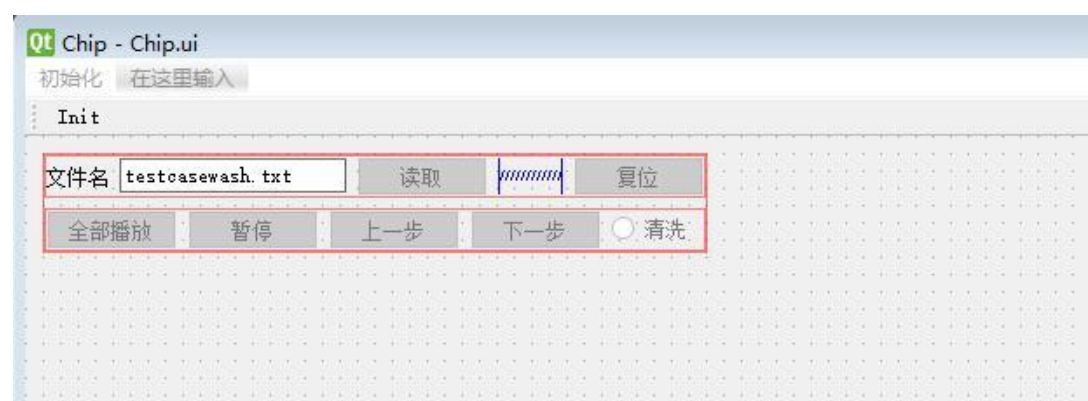
若已到达最终时刻则调用 drawNumbers()画出每个网格的污染次数, 若需要清洗则调用

drawWashDrops()画出清洗液滴

mousePressEvent(QMouseEvent*)若按下的不是鼠标右键则返回, 取鼠标点击的位置, 调用

pos2block()函数将其转化为网格坐标, 若在芯片范围之外则返回, 否则对相应的 block 调用

reverseBarrier()函数, 若原本不是障碍则设为障碍, 反之亦然



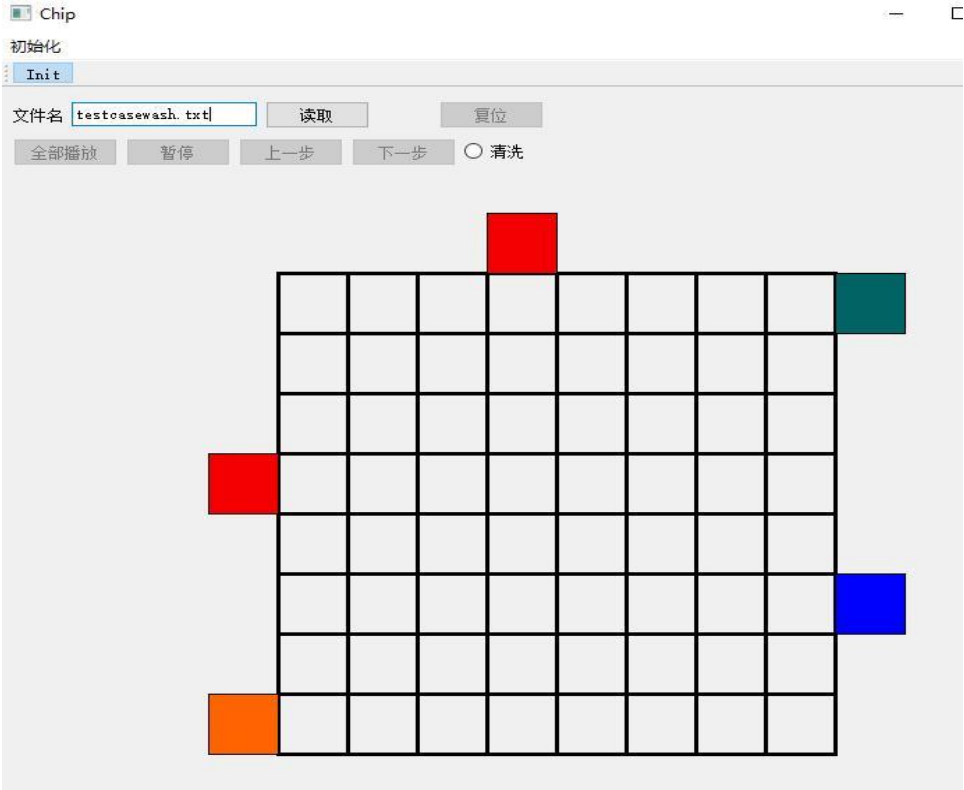
运行过程



点击工具栏的 Init，或点击菜单栏“初始化”再点击下拉菜单中的 Init



在弹出的对话框中输入行数、列数、输入端口、输出端口，点击确定

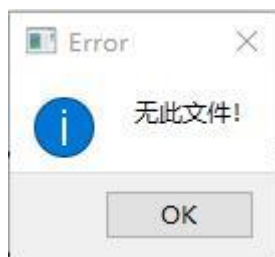


(图中红色为输入端，蓝色为输出端，橙色为清洗液滴输入端，灰绿色为清洗液滴输出端)

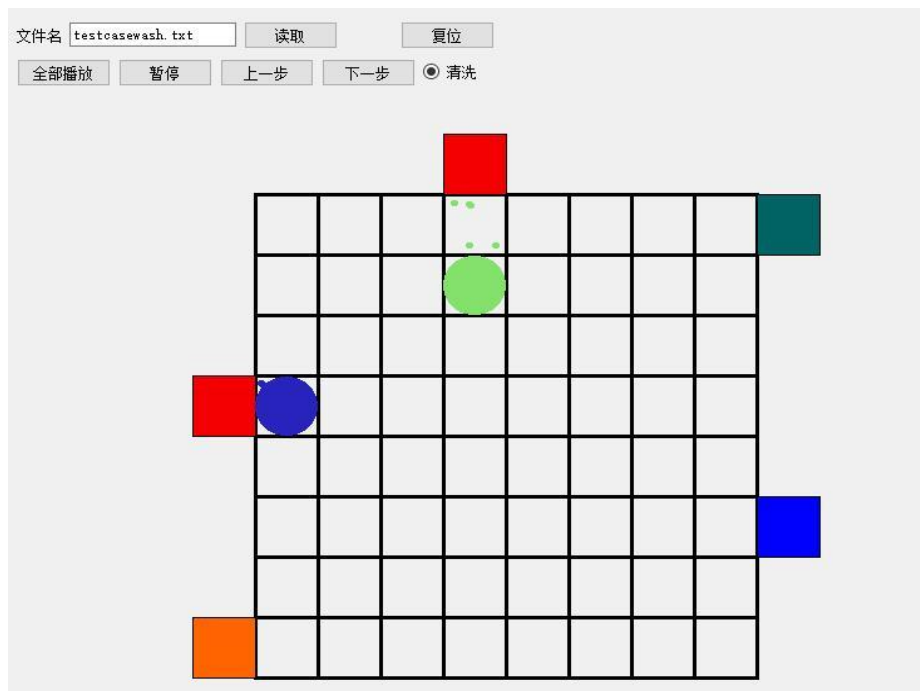
点击“清洗”可设置是否需要清洗



输入文件名，点击“读取”，若没有相应文件则报错

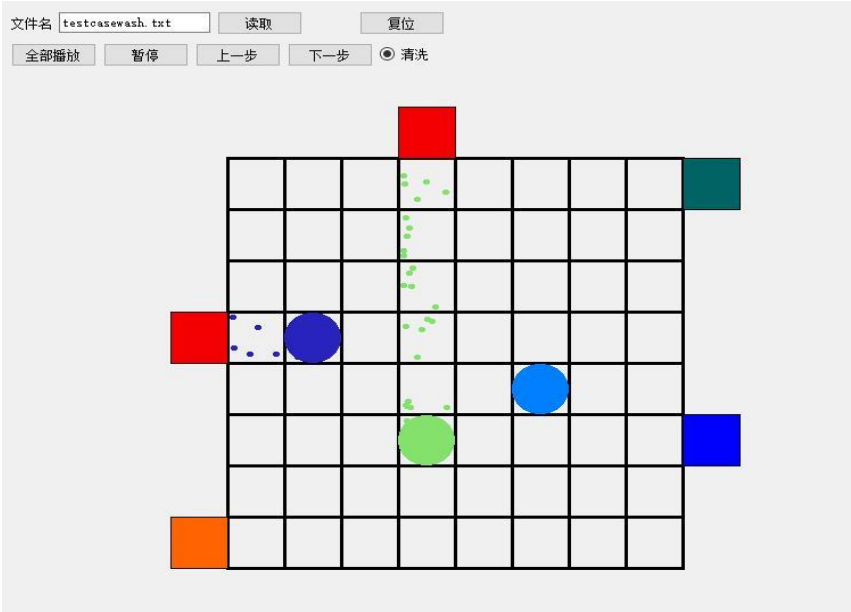


成功读取后点击“全部播放”，则每秒更新一次

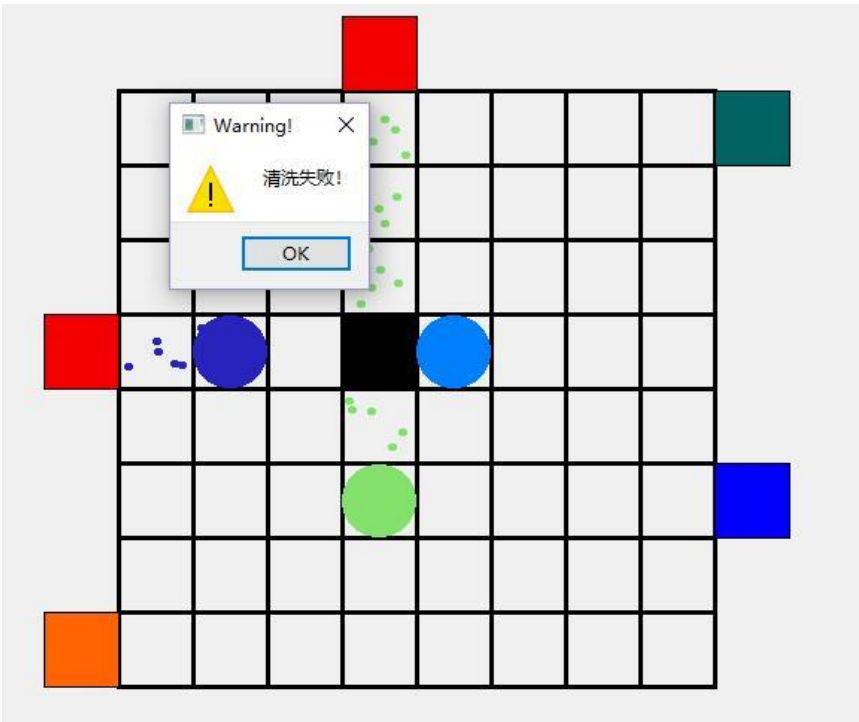


点击“暂停”则停止播放，点击暂停后点击“上一步”则回溯一步，点击“下一步”则前进一步

若设置了清洗，则清洗液滴会在有网格将被二次污染时出现并清洗



若在清洗液滴目的地处设置了障碍则会报告清洗失败



播放到终点时会显示网格的污染次数

