**三层电梯状态机课程设计报告**

**15069130005 王涵**

1. **状态机图**

D

S4

D

S2

S1

S5

S3

S9

S6

S7

S8

**二、各个状态函数的描述**

该三层电梯状态机共有5个状态：Idle、MovingUp、MovingDown、DoorOpen、DoorClosing。

Idle状态为电梯静止状态。在该状态下，电梯门处于关闭状态，电梯没有任何上下运动或开关门动作，但是会进行静态检测，等待下一步的操作。根据静态检测结果，此状态可能转换到其他四种状态。如果电梯停在2层或3层，10s内无操作，电梯将自动返回1层。

MovingUp状态为电梯向上运动状态。在该状态下，电梯在向上运动过程中，在此过程中，将进行动态监测，判断将要停止的楼层，到达目标楼层后转换为DoorOpen状态。

MovingDown状态为电梯向下运动状态。在该状态下，电梯在向下运动过程中，其他同MovingUp状态。

DoorOpen状态为电梯开门状态。在此状态下，电梯门处于正在打开或打开的状态。在开门完成2秒内，若无手动开关门操作，将自动关门。该状态会在关门灯亮时或到达2秒时转换为DoorClosing状态。

DoorClosing状态为电梯关门状态。在此状态下，电梯门处于正在关闭状态。该状态会在检测到障碍物时转换为开门状态，或者在开门灯亮起的时候转换为开门状态，在关门完成后转换为Idle状态，。

该电梯一共有5类事件：

E1: 门内开门按钮(OpenDoorLight)

E2: 门内关门按钮(CloseDoorLight)

E3: 门内楼层按钮(PanelFloorLight)

E4: 门外up呼叫按钮(Call Light)

E5: 门外down呼叫按钮(Call Light)

电梯在不同状态下，某种事件发生时，会伴随着不同的状态转换：

静止状态：StateIdle函数。检测到不同楼层的E3、E4、E5事件发生时，会转换为移动状态（向上或向下），即S1、S2；检测到同楼层的E4、E5事件发生时，会转换为开门状态，即S3；同楼层的E3无效。检测到E1事件发生时，转换到开门状态，即S3。因为改状态下，电梯门是关闭的，所以E2事件无效。电梯无动作则保持静止状态，即S4；在2、3层时，10秒内未检测到任何动作，将自动返回1层。

运动状态：MovingUp/MovingDown函数。检测E3、E4、E5事件，可能会改变目标楼层。检测与目标楼层距离，如果达到误差以内，就停止运动，并转换到开门状态，即S5、S6。如果没有到达目标楼层，就保持当前运动状态，即D。E1、E2事件在改状态下无效。

开门状态：DoorOpen函数。检测到E2事件，转换为关门状态，即S7。E1事件在此状态下无效。

关门状态：DoorClosing函数。在关门结束后自动转换为静止状态。检测到E1事件，转换为开门状态，即S8。E2事件在此状态下无效。同样的如果检测到障碍物转换为开门状态。

1. **状态机代码**

#include "stdafx.h"

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include "ElevatorLib.h"

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Idle状态，电梯停止在某楼层，门是关闭的，处于静止状态，等待相关事件的发生，从而转换到下一个状态。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void StateIdle(int \*state)

{

int Floor,PresentFloor;

bool up;

Floor = IdleWhatFloorToGoTo(&up);

PresentFloor = GetNearestFloor();

if (Floor > 0 && up)

{

SetMotorPower(1);

\*state = MovingUp;

}

if (Floor > 0 && !up)

{

SetMotorPower(-1);

\*state = MovingDown;

}

if (GetOpenDoorLight())/\*检测门内开门按钮\*/

{

SetDoor(PresentFloor, true);/\*开门\*/

SetOpenDoorLight(false);/\*消费开门按钮\*/

\*state = DoorOpen;

}

if (GetCallLight(PresentFloor, up))/\*检测门外up按钮\*/

{

SetDoor(PresentFloor, true);/\*开门\*/

SetCallLight(PresentFloor, up, false);/\*消费按钮\*/

\*state = DoorOpen;

}

if (GetCallLight(PresentFloor, !up))/\*检测门外down按钮\*/

{

SetDoor(PresentFloor, true);/\*开门\*/

SetCallLight(PresentFloor, !up, false);/\*消费按钮\*/

\*state = DoorOpen;

}

if (GetCloseDoorLight())/\*检测门内关门按钮\*/

{

SetCloseDoorLight(false);

\*state = DoorClosing;

return;

}

}

/\*MovingUp状态\*/

void StateMovingUp(int \*state)

{

SetMotorPower(1); /\*向上移动\*/

int Floor;

Floor = GoingUpToFloor();

if (fabs(GetFloor() - Floor) < Lib\_FloorTolerance)/\*到达目标楼层，停止，开门\*/

{

SetMotorPower(0);

SetDoor(Floor, true);

\*state = DoorOpen;

SetCallLight(Floor, true, false);/\*消费门外up按钮\*/

SetPanelFloorLight(Floor, false);/\*消费门内楼层按钮灯\*/

}

if (fabs(GetFloor() - 3) < Lib\_FloorTolerance)/\*到达3楼\*/

{

SetMotorPower(0);

SetDoor(Floor, true);

\*state = DoorOpen;

SetCallLight(3, false, false);/\*消费门外down按钮\*/

SetPanelFloorLight(3, false);/\*消费门内按钮\*/

}

if (GetOpenDoorLight()) /\*灭掉开关门灯\*/

SetOpenDoorLight(false);

if (GetCloseDoorLight())

SetCloseDoorLight(false);

}

void StateMovingDown(int \*state)

{

SetMotorPower(-1); /\*向下移动\*/

int Floor;

Floor = GoingDownToFloor();

if (fabs(GetFloor() - Floor) < Lib\_FloorTolerance)/\*到达目标楼层，停止，开门\*/

{

SetMotorPower(0);

SetDoor(Floor, true);

\*state = DoorOpen;

SetCallLight(Floor, true, false);/\*消费门外down按钮\*/

SetPanelFloorLight(Floor, false);/\*消费门内按钮\*/

}

if (fabs(GetFloor() - 1) < Lib\_FloorTolerance)/\*到达1层\*/

{

SetMotorPower(0);

SetDoor(Floor, true);

\*state = DoorOpen;

SetCallLight(1, false, false);/\*消费门外up按钮\*/

SetPanelFloorLight(1, false);/\*消费门内当前楼层的按钮\*/

}

if (GetOpenDoorLight()) /\*灭掉开关门灯\*/

SetOpenDoorLight(false);

if (GetCloseDoorLight())

SetCloseDoorLight(false);

}

/\*DoorOpen状态\*/

void StateDoorOpen(int \*state)

{

int PresentFloor;

PresentFloor = GetNearestFloor();

if (GetCloseDoorLight())/\*检测门内关门按钮\*/

{

SetDoor(PresentFloor, false);/\*关门\*/

SetCloseDoorLight(false);/\*消费关门按钮\*/

\*state = DoorClosing;/\*转入关门状态\*/

}

if (IsDoorOpen(PresentFloor))/\*检测门的状态，自动关门\*/

{

SetDoor(PresentFloor, false);

\*state = DoorClosing;

}

if (GetOpenDoorLight())

SetOpenDoorLight(false);

}

/\*DoorClosing状态\*/

void StateDoorClosing(int \*state)

{

int PresentFloor;

PresentFloor = GetNearestFloor();

if (GetOpenDoorLight()) /\*检查门内开关按钮\*/

{

SetDoor(PresentFloor, true);

SetCloseDoorLight(false); /\*消费开门灯\*/

\*state = DoorOpen; /\*开门\*/

}

if (GetCloseDoorLight()) /\*检查关门灯，正在关门，无动作，消费关门灯\*/

SetCloseDoorLight(false);

if (IsBeamBroken()) /\*检测是否有障碍物\*/

{

SetDoor(PresentFloor, true);/\*有障碍物，开门\*/

\*state = DoorOpen;

}

if (IsDoorClosed(PresentFloor))

\*state = Idle;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* 主控循环： 状态机，每隔一定时间(如，100ms)被调用一次，采集系统的运行状态

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void main\_control(int \*state)

{

if (IsElevatorRunning()) // 仿真正在运行

{

switch (\*state)

{

case Idle:

if (GetNearestFloor() != 1)

{

AutoTo1Floor();// Idle状态，一定时间无动作，自动到一楼

}

StateIdle(state);

break;

case MovingUp:

CancelTo1Floor(); // 其它状态，取消自动到一楼

StateMovingUp(state);

break;

case MovingDown:

CancelTo1Floor();

StateMovingDown(state);

break;

case DoorOpen:

CancelTo1Floor();

StateDoorOpen(state);

break;

case DoorClosing:

CancelTo1Floor();

StateDoorClosing(state);

break;

default:

printf("没有这种状态!!!\n");

}

}

}

1. **运行测试描述实现的功能及测试结果**

1、电梯停于1F或2F时，按3F向下呼叫按钮；电梯上升到3F停止，开门/关门。

2、电梯停于2F或3F时，按1F向上呼叫按钮；电梯下降到1F停止，开门/关门。

3、电梯停于1F时，按2F向上呼叫按钮；电梯上升到2F停止，开门/关门。

4、电梯停于3F时，按2F向下呼叫按钮；电梯下降到2F停止，开门/关门。

5、电梯停于1F，2F和3F均有按钮呼叫；电梯先上升到2F，开门/关门，然后上升到3F停止，开门/关门。

6、电梯停于3F，2F和1F均有按钮呼叫；电梯先下降到2F，开门/关门，然后下降到1F停止，开门/关门。

7、电梯上升途中或下降途中，任何反方向按钮呼叫均无效。

8、电梯停于1F，按3F向下呼叫按钮，然后立即按2F向下呼叫按钮；电梯上升到3F停止，开门/关门，然后下降到2F停止，开门/关门。

9、电梯停于2F，按门内楼层按钮3，然后门内楼层按钮1；电梯上手到3F停止，开门关门，然后下降到1F，开门/关门。

10、电梯停于1F，按门内楼层按钮3，当电梯上升在1F到2F中间以下，按2F向上呼叫按钮；电梯先上升到2F，开门/关门；然后再到3F，开门/关门。

11、电梯停于1F，按门内楼层按钮3，当电梯上升在1F到2F中间以上，按2F向上呼叫按钮；电梯先上升到3F，开门/关门；然后再到2F，开门/关门。

12、电梯停于1F，按2F向下呼叫按钮和向上呼叫按钮以及3F的向下呼叫按钮；电梯上升到2F停止，开门/关门，2F的向上呼叫按钮灯关闭，开门/关门， 2F的向下呼叫按钮灯关闭，然后上升到3F停止，开门/关门，3F的向下呼叫按钮关闭。

13、电梯在2F以上，10s无动作，自动降到1楼。

14、所有停止，开门/关门后，对应楼层的同方向门外呼叫按钮灯（最高楼向下呼叫按钮，最底层向上呼叫按钮）和门内楼层按钮灯关闭。

15、运动状态，开关门按钮失效。正在开门，开门按钮失效；正在关门，关门按钮失效