package remoterobot;

import remoterobot.drivers.Motor;

//import remoterobot.drivers.MotorThread;

//import remoterobot.drivers.ShaftMotor;

import com.pi4j.io.gpio.GpioController;

import com.pi4j.io.gpio.GpioFactory;

import com.pi4j.io.gpio.GpioPinDigitalOutput;

import com.pi4j.io.gpio.PinState;

import com.pi4j.io.gpio.RaspiPin;

//import java.util.ArrayList;

public class RemoteRobot {

public enum MoveDirections {

STOP,

ROTATE\_CW,

ROTATE\_CCW

}

public enum MovementState {

STOPPED,

MOVING

}

public static void main(String[] args) {

Double DEFAULT\_VELOCITY = 20.0;

MovementState state = MovementState.STOPPED;

MoveDirections moveDirection = MoveDirections.STOP;

Double velocity = DEFAULT\_VELOCITY;

//boolean complete = false;

boolean smooth = false;

Double speed = null;

Double distance = 20.0;

GpioController gpio = GpioFactory.getInstance();

GpioPinDigitalOutput enable = gpio.provisionDigitalOutputPin(RaspiPin.GPIO\_01, "Enable", PinState.LOW);

Motor motor = new Motor(RaspiPin.GPIO\_00, RaspiPin.GPIO\_02);

state = MovementState.MOVING;

moveDirection = MoveDirections.ROTATE\_CW;

switch (moveDirection) {

case ROTATE\_CW:

velocity = velocity\*0.8;

speed = 1.0;

break;

case ROTATE\_CCW:

velocity = velocity\*0.8;

speed = -1.0;

break;

}

motor.SetDirection(speed > 0);

if (Math.abs(speed) > 0.0000001) {

motor.move(Math.abs(speed), velocity, distance, smooth,new Motor.MotorAdvCallback() {

@Override

public boolean stop() {

return false;

}

});

}

/\*else {

complete = true;

}\*/

}

}