DEMONSTRATION SUBMISSION

11/27/2017

Jake Andrus - 25%

Anirudh Chinepalli - 25%

Michael Galvan - 25%

Mohit Doshi - 25%

**FASTEST TIME**

Time to pick up – 0:27.83

Time to drop off - 1:29.18

Total Time – 1:57.01

**SOURCE CODE**

%MANUAL CONTROLS INSTRUCTIONS

% Automatic/Manual Toggle

% a - Switches to Automatic.

% m - Switches to Manual Control.

%

% DRIVING

% Uparrow ---- Moves Forwards.

% DownArrow -- Moves Backwards.

% LeftArrow -- Turns Left.

% RightArrow - Turns Right.

% Backspace -- Stops ALL Wheel Movement.

%

% CLAW

% o - Stops ALL claw movement

% i - Closes claw

% p - Opens claw

%

% Customizable Variables

auto\_Speed\_Forward = 80;

auto\_Speed\_Backwards = -80;

auto\_Speed\_TurnSpeed = 80;

manual\_Speed\_Forward = 40;

manual\_Speed\_Backwards = -40;

manual\_Speed\_TurnSpeed = 40;

claw\_Speed\_Open = 40;

claw\_Speed\_Close = -40;

turnCounter = 0;

global key

InitKeyboard();

state = 0;

%state = -1;

timerVal = 0;

timerVal2 = 0;

timerVal\_Ultra = 0;

timerVal\_TurnTime = 0;

turnTime = 1.3;

noturnTime = .60;

timerStarted\_Ultra = false;

turnBuffer = 1.5; %Time used for automatic turning

turnTimer = 0;

openClaw = true;

isClawMoving = false;

wallflag = false;

wallFlagLeft = false;

%Bools for passenger Pick Up

manualControl = false;

blueFound = false; %Pickup is Blue

greenFound = false; %Dropoff is Green

while 1

pause(0.1);

%-----------AUTOMATIC CONTROL--------------------------------

if(manualControl == false)

switch state

%TEST CASE

case -1

%distance = brick.UltrasonicDist(1); % Get distance to the nearest object.

%display(distance); % Print distance.

color = brick.ColorColor(4);

display(color);

%--Auto\_Forward Movement Until Sensors are Triggered

case 0

%disp(0);

% if(wallflag == true) %Moves Car Away from wall

% brick.MoveMotor('A', 80);

% brick.MoveMotor('D', 80/1.2);

% else

% if(wallflag == true) %Moves Car Towards wall

% brick.MoveMotor('A', 80/1.2);

% brick.MoveMotor('D', 80);

% else

brick.MoveMotor('AD', 80); %If this doesn't work for just go forward

% end

% end

%--Auto\_ULTRASONIC SENSOR---------------------------------------

if(turnCounter == 1 && turnBuffer >= 1)

turnTimer = tic; % starts timer

turnBuffer = 1.0;

end

if(turnTimer >= .5 && turnCounter == 1)

turnCounter = 0;

turnBuffer = 1.5;

end

distance = brick.UltrasonicDist(1); % Get distance to the nearest object.

if(distance > 45) %

if(timerStarted\_Ultra == false)

timerStarted\_Ultra = true;

timerVal\_Ultra = tic; %starts timer

end

if(timerVal\_Ultra >= turnBuffer)

disp("No Right wall detected Turning Right");

timerVal\_TurnTime = tic; %starts timer

state = 4;

end

% else

% timerStarted\_Ultra = false;

% timerVal = tic; %Reset Timer

% if(distance <= 15) %if too close to wall

% wallflag = true;

% wallFlagLeft = false;

% else %if too far away from wall

% wallFlagLeft = true;

% wallflag = false;

% end

end

%--Auto\_COLOR SENSOR---------------------------------------------------

color = brick.ColorColor(4);

if(color == 5) % Color Red is Detected

if(distance > 45)

turnBuffer = 1.0;%if stopped and red reduce turn buffer time.

end

disp("Red Detected: Switching to Manual Control");

timerVal = tic;

state = 3;

end

if(color == 2 && blueFound == false) % Color Blue is Detected

blueFound = true;

brick.StopAllMotors();

disp("Blue Detected: Switching to Manual Control");

manualControl = true;

end

if(color == 3 && blueFound == true) % Color Green is Detected

brick.StopAllMotors();

disp("Green Detected: Switching to Manual Control");

manualControl = true;

end

%--Auto\_TOUCH SENSOR-----------------------------------------

if (brick.TouchPressed(2)) % Front touch Sensor Hit

disp('Front Touch Sensor Hit. Moving to Case 6');

state = 1;

timerVal = tic; % Start Timer.

end

if (brick.TouchPressed(3)) % Side touch Sensor Hit

disp('Side Touch Sensor Hit. Moving to Case 6');

state = 6;

timerVal2 = tic;

else

wallflagLeft = false;

end

%-Auto\_CASE-1\_\_Bumped, Reverse for a time.

case 1

brick.MoveMotor('AD', auto\_Speed\_Backwards);

if(toc(timerVal) > 1.5) % State Transition after 1 seconds

state = 2;

timerVal = tic; %Reset Timer

turnBuffer = 1.5; %resets turn buffers

end

%-Auto\_CASE-2\_\_Done Reversing, Turn right 90 degrees.

case 2

disp(2);

brick.MoveMotor('A', auto\_Speed\_TurnSpeed);%Left turn

brick.StopMotor('D');

if(toc(timerVal) > 1.075) %Stat Transition after 1 seconds

state = 0;

end

%-Auto-CASE-3\_\_Light Sensor red, Stop for a time

case 3

disp(3);

brick.StopMotor('AD');

if(toc(timerVal)> 1.0) % Sate Transition afer 1 seconds

state = 0;

timerVal = tic; % Reset Timer

end

%-Auto-CASE-4\_\_

case 4

%this is where the car turns 90 degrees

brick.MoveMotor('D', auto\_Speed\_TurnSpeed);%Right turn

brick.StopMotor('A');

if(toc(timerVal\_TurnTime) >= turnTime)

disp("Back TO ZERO state");

turnCounter = turnCounter + 1;

state = 5;

timerVal\_TurnTime = tic; % Reset Timer

end

case 5

%handle double turns

brick.MoveMotor('AD', auto\_Speed\_Forward);%double right turn

if(toc(timerVal\_TurnTime) >= noturnTime)

state = 0;

timerVal\_TurnTime = tic; %reset timer

end

case 6

%brick.MoveMotor('D', 80);

%brick.MoveMotor('A', 80/1.2);

brick.MoveMotor('D', 20);

brick.MoveMotor('A', -40);

if(toc(timerVal2) > 1.5) % State Transition after 1 seconds

state = 0;

timerVal2 = tic; %Reset Timer

end

end

% Enters into Manual Control

if(key == 'm')

manualControl = true;

brick.StopMotor('AD');

timerVal = tic;

end

end

%------------------MANUAL CONTROl-------------------------------

if(manualControl == true)

switch key

case 'uparrow' %Move Foreward

%disp(mUpArrow);

brick.MoveMotor('AD', manual\_Speed\_Forward);

case 'downarrow'%Move Backwards

brick.MoveMotor('AD', manual\_Speed\_Backwards);

case 'leftarrow'

brick.MoveMotor('A', manual\_Speed\_TurnSpeed);%Left turn

brick.StopMotor('D');

case 'rightarrow'

brick.MoveMotor('D', manual\_Speed\_TurnSpeed);%Right turn

brick.StopMotor('A');

case 'backspace' %Stops all movement

brick.StopMotor('AD');

case 'o' %Stops Claw Motor and resets

brick.StopMotor('C');

openClaw = false;

isClawMoving = false;

timerVal= tic;

case 'i'

brick.MoveMotor('C', claw\_Speed\_Close);

case 'p'

brick.MoveMotor('C', claw\_Speed\_Open);

case 0 % No key is being pressed

%Handles Claw Logic

if(toc(timerVal) > .5 && openClaw == true && isClawMoving == true) %Stat Transition after 0.5 seconds -- Closes Claw

brick.StopMotor('C');

openClaw = false;

isClawMoving = false;

timerVal = tic;

end

if(toc(timerVal) > .5 && openClaw == true && isClawMoving == true) %Stat Transition after 0.5 seconds -- Closes Claw

brick.StopMotor('C');

openClaw = false;

isClawMoving = false;

timerVal = tic;

end

case 'a' %Switches back to automatic control

manualControl = false;

end

end

if(key == 'q')%Quit the program

brick.StopAllMotors();

break;

end

end

CloseKeyboard();

**INVENTORY STATUS**

Inventory all clear