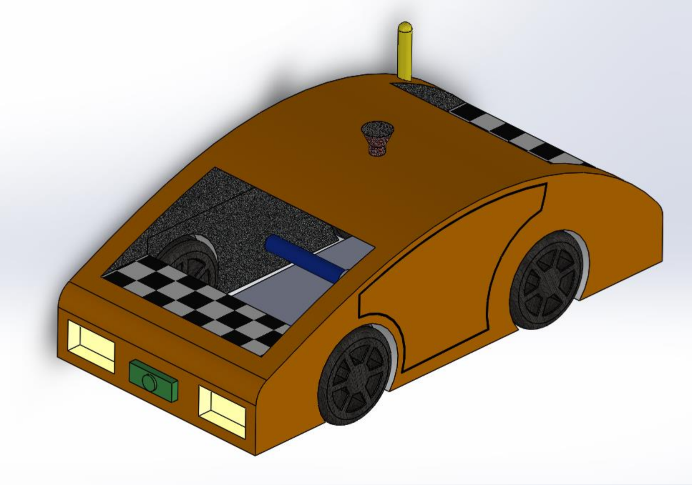
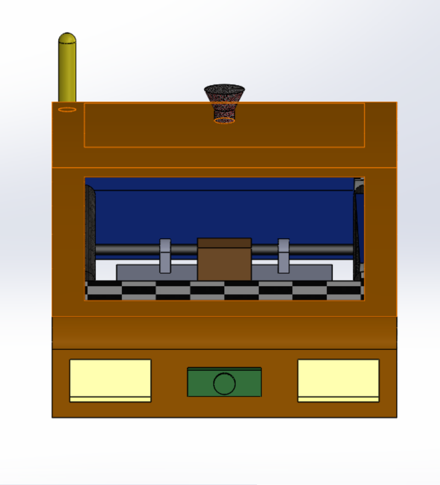
# Vehicle Dynamics Project presentation Group 8

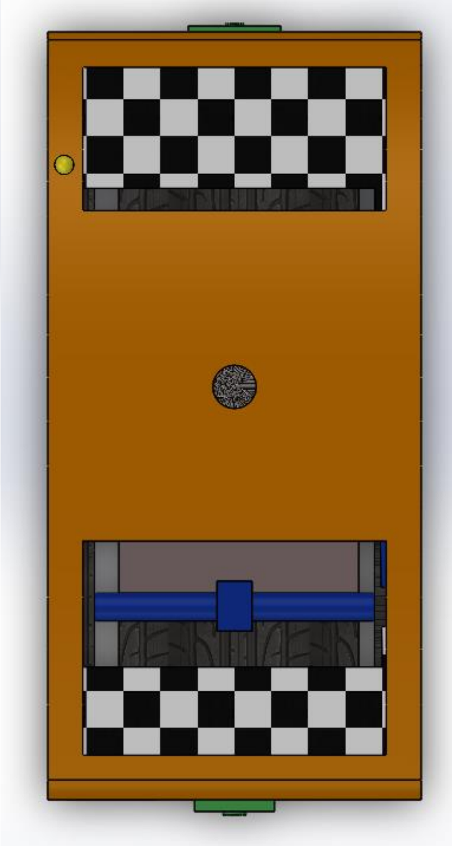
## Autonomous Drift Cornering with Mixed Open-loop and Closed-loop Control

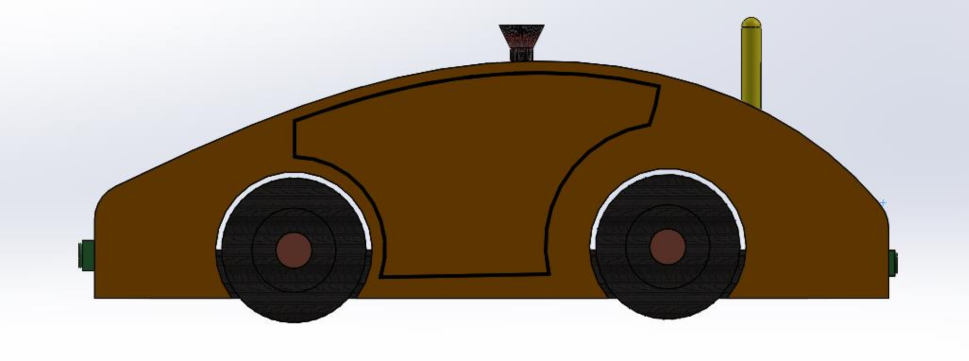
R.Prathima-Me15btech11032
B.Sunidhi-Me15btech11005
Rahul Alam-Me15btech11030
Aman Chaurasia-Me15btech11003

# CAR MODEL









# A clip to show drift

# Drifting definition

- Slip angle:The side slip angle β measures the angle between vehicle's direction of travel and longitudinal velocity vector $\beta = \tan^{-1}\frac{U_y}{U_x}.$

- Drifting:It is controlled oversteering while driving the vehicle through a corner where often the front wheels are pointing in the opposite direction to the turn

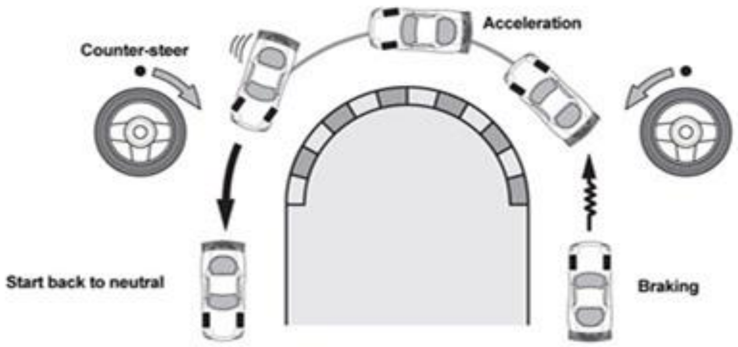



Image source

# Why Mixed open loop and closed loop control?

- Expert drivers have the skill to perform high side slip maneuvers, like drifting, during racing events to minimize lap time
- Current driver assist controls systems like ABS and ESC try to prevent drifting conditions from ever arising. These are costly and critical to use
- Need to design controllers
  - respond quickly to unintentional drift
  - avoid collisions
  - minimize lap time for racing applications

# System Model

Six state - Bicycle model:

- Longitudinal velocity Ux (body-fixed frame)
- Lateral velocity Uy (body-fixed frame)
- Yaw rate r
- Longitudinal position X (ground-fixed frame)
- Lateral position Y (ground-fixed frame)
- Yaw angle ψ.

The input vectors:

- steering angle δ
- rear longitudinal force FxR.

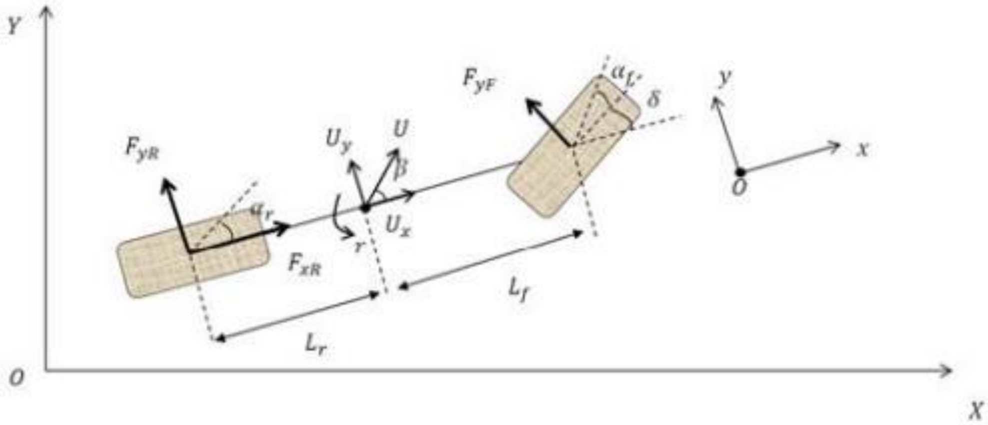


Fig. 1. Vehicle model schematic

# Governing Differential equations

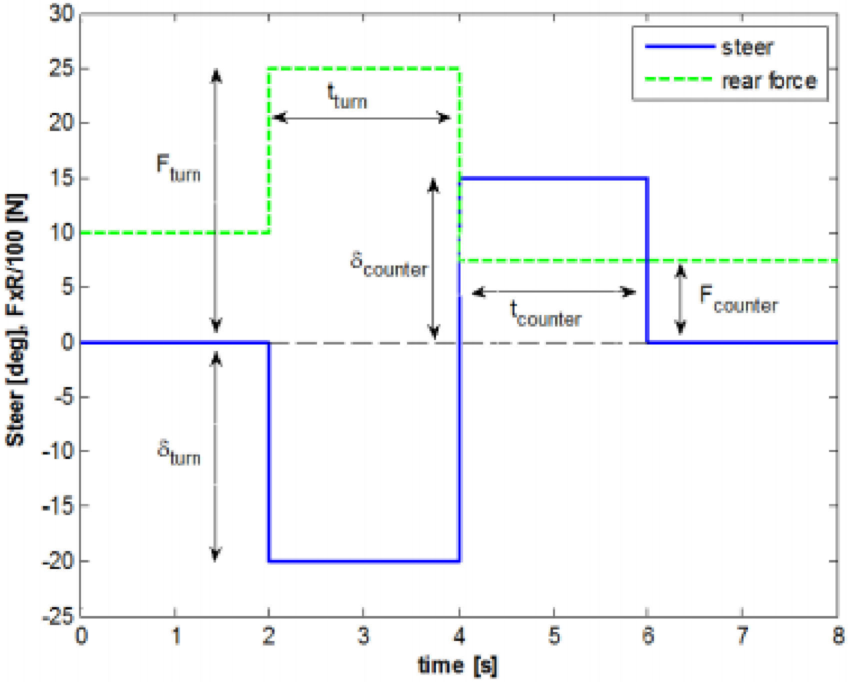
$$\dot{U}_x = \frac{1}{m}\sum F_x = \frac{1}{m}F_{xR}$$
$$\dot{U}_y = \frac{1}{m}\sum F_y - U_x r = \frac{1}{m}(F_{yF} + F_{yR}) - U_x r$$
$$\dot{r} = \frac{1}{I_z}\sum M_z = \frac{1}{I_z}(L_f F_{yF} - L_r F_{yR})$$
$$\dot{X} = U_x \cos\psi - U_y \sin\psi$$
$$\dot{Y} = U_x \sin\psi + U_y \cos\psi$$
$$\dot{\psi} = r$$

$$F_{yF} = C_{\alpha f}\alpha_f$$
$$F_{yR} = C_{\alpha r}\alpha_r$$

$$\alpha_f = \delta - \frac{U_y + L_f r}{U_x}$$
$$\alpha_r = \frac{L_r r - U_y}{U_x}.$$

- Fyf and Fyr are lateral forces on the front and rear wheels
- For the tires, we use a linear model to estimate the lateral force
- Cαi is the cornering stiffness of front or rear wheel
- αi is the side slip angle of wheels (i = f, r).

# Input sequence Steering angle and Rear Force




- t-turn, time duration of Turning in phase;
- δturn, steering angle of Turning in phase;
- Fturn, rear force of Turning in phase;
- tcounter, time duration of Counter-steering phase;
- δcounter, steering angle of Counter-steering phase;
- Fcounter, rear force of Counter-steering phase.

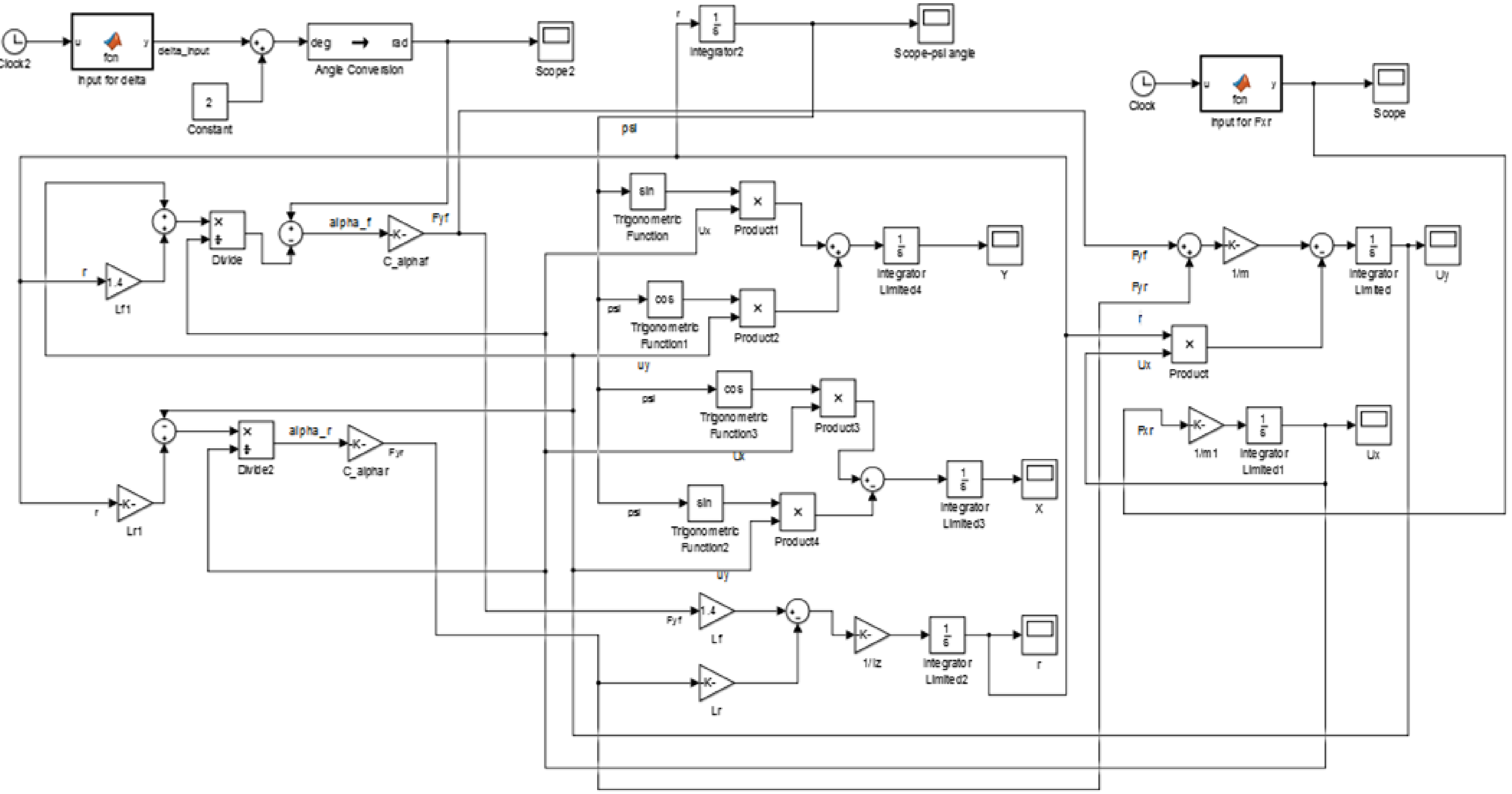
# State Space Form of governing differential equations

$$\dot{\mathbf{z}} = f(\mathbf{z}, \mathbf{u})$$

where $\mathbf{z} = [U_x, U_y, r, X, Y, \psi]$, and $\mathbf{u} = [\delta, F_{xR}]$.

- Here, z is the output variables and u is the input variables
- We solve the governing differential equations using open loop and closed loop control in Matlab using Simulink
- Open loop is  non-feedback controller
- A Closed-loop Control System, also known as a *feedback control system* is a control system which uses some portion of the output “back” to the input.
- Closed loop is based on LQR approach in this paper

# Open Loop Control Design



# Closed loop control

- Define the reference trajectory $U^{ref}$ x (i), $U^{ref}$ y (i), $r^{ref}(i)$, $X^{ref}(i)$, $Y^{ref}(i)$, $\psi^{ref}(i)$, $\delta^{ref}(i)$, $F^{ref}_{xR}(i)$
- Linearize vehicle model along the designed reference trajectory
- Define the feedforward-feedback control policy as follows

$$z(i+1) = z(i) + t*f(zref(i),uref(i))$$

$$z(0) = zref(0)$$

$$\Delta z(i) = z(i) - zref(i)$$

$\Delta u(i) = -K(i)\Delta z(i)$ where K is the LQR gain

$$u^{cl}(i) = u^{ref}(i) + \Delta u(i)$$

# Implementation of Closed Loop system in Matlab

```
% Input - from excel sheet
% A, B matrices defined
A = zeros(6,6);
B = zeros(6,2);
uref = [delta,Fx_r];
zref = [Ux,Uy,r,X,Y,psi];
%constants
C_f = 36000;
C_r = 36000;
Lr = 1.65;
Lf = 1.4;
m = 1830;
Iz = 3287;

B(1,2) = 1/m;
B(2,1) = C_f/m;
B(3,1) = (Lf*C_f)/Iz;
%SYS = ss((A-B*K),B,C,D);
%[y,t,x] = lsim(SYS,u,t);
z = zeros(length(time),6);
```

```
u = zeros(length(time),2);
for i = 1:length(time)
    A = Amat(Ux(i),Uy(i),psi(i),r(i));
    C =[B A*B A^2*B];
    Q = C'*C;
    D = zeros(6,2);
    R = eye(2);
    K = lqr(A,B,Q,R);
    z(1,:) = zref(1,:);
    z(i+1,:) = zref(i,:) +
time(i)*eqn(zref(i,1),zref(i,2),zref(i,3),zref(i,4
),zref(i,5),zref(i,6),uref(i,1),uref(i,2));
    u(i+1,:) = uref(i,:) + (z(i,:)-zref(i,:))*K';
end
plot(z(:,4),z(:,5))
xlabel('X');
ylabel('Y');
hold on
plot(X,Y)
```

# Subroutine closed loop

```
function A = Amat(Ux,Uy,psi,r)
    A = zeros(6);
    C_f = 36000;
    C_r = 36000;
    Lr = 1.65;
    Lf = 1.4;
    m = 1830;
    Iz = 3287;

    A(2,1) = ((C_f + C_r)*Uy)/(m*Ux^2) - (Lr*C_r-
Lf*C_f)*r/(m*Ux^2) - r;
    A(2,2) = -(C_f+C_r)/(m*Ux);
```

```
A(2,3) = ((Lr*C_r - Lf*C_f)/m*Ux) - Ux;
    A(3,1) = -(Lr*C_r-Lf*C_f)*Uy/(Iz*Ux^2) +
(Lf^2*C_f + Lr^2*C_r)*r/(Iz*Ux^2);
    A(3,2) = (Lr*C_r - Lf*C_f)/Iz*Ux;
    A(3,3) = - (Lf^2*C_f + Lr^2*C_r)/(Iz*Ux);
    A(4,1) =  cos(psi);
    A(4,2) = -sin(psi);
    A(4,6) = -Ux*sin(psi) -Uy*cos(psi);
    A(5,1) = sin(psi);
    A(5,2) = cos(psi);
    A(5,6) = Ux*cos(psi) - Uy*sin(psi);
    A(6,3) = 1;
end
```

# Sub function

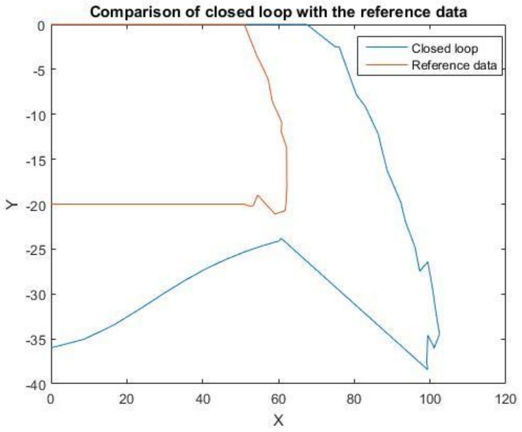
```
function f = eqn(Ux,Uy,r,x,y,psi,delta,Fx_r)

    C_f = 36000;
    C_r = 36000;
    Lr = 1.65;
    Lf = 1.4;
    m = 1830;
    Iz = 3287;

    alpha_f = delta - (Uy+Lf*r)/Ux;
    alpha_r = (Lr*r - Uy)/Ux;
    Fy_f = C_f*alpha_f;
    Fy_r = C_r*alpha_r;
```

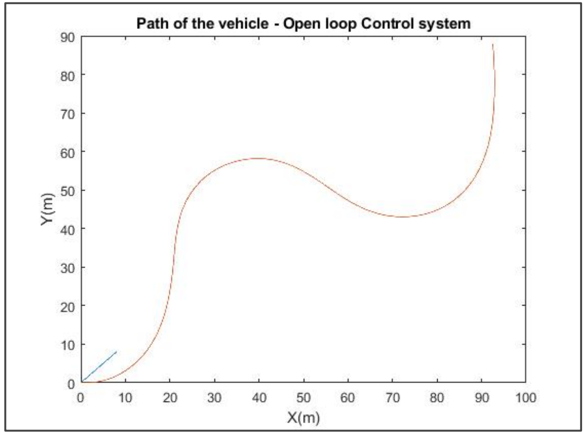
```
 dUx = Fx_r/m;
    dUy = (Fy_f+Fy_r)/m - Ux*r;
    dr = (Lf*Fy_f - Lr*Fy_r)/Iz;
    dX = Ux*cos(psi) - Uy*sin(psi);
    dY = Ux*sin(psi) + Uy*cos(psi);
    dpsi = r;
    f = [dUx,dUy,dr,dX,dY,dpsi];
end
```

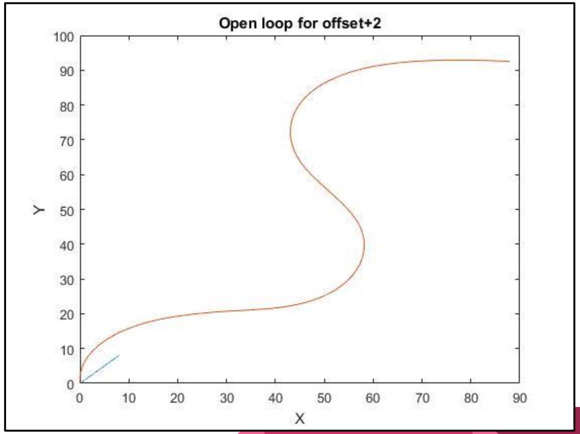
# Vehicle Parameters

| Parameter | Value |
|---|---|
| mass-m(kg) | 1830 |
| Moment of Inertia - Iz(kg.m^2) | 3287 |
| Lf(m) | 1.65 |
| Lr(m) | 1.4 |
| Cαf(N/rad) | 36000 |
| Cαr(N/rad) | 36000 |

# Simulation Results:Comparison of Closed and Open loop system



# Conclusion

- Open Loop Maneuver is erratic during drift cornering
- Closed Loop control is gives better results.
- Mixed open and closed loop controller chooses open loop during straight line path and closed loop control  during drift cornering.

# Future Development

- The use of mix loop, which can give us more optimized results.
- Simulation in Carsim
- The advent of self-driving cars has pushed the boundaries on the safety of automobiles, but most modern self-driving car systems ignore the possibility of a car slipping resulting from inclement weather or driver error .
- Passengers and bystanders would benefit heavily if self-driving cars could handle slipping by learning to drift with the turn rather than against it (by applying the brakes, or turning away, which is the instinctive action), preventing many fatalities

# Autonomous Drift using Machine Learning

# References


1. Project Paper - Autonomous Drift cornering by mixed open and closed loop control
2. https://www.sciencedirect.com/science/article/pii/S0957415818300771
3. Car model reference

4. Drift video

5. Autonomous drift by machine learning video