Class Diagram

October 19, 2019

Ishan PatelNakul PatelSuyash YeotikarUID: 116169103UID: 116334877UID: 116188456

Мар		AckermanKinematicModel
-mapBounds:Point -currCarCoords:Point3f -startCoords:Point3f -destinationCoords:Point3f -currMapImage:Mat -currCoords:Point3f -length:double -width:double		 - wheelBase : double - axleWidth : double - steeringAngle : double - carVelocity : double - carState : Point3f
+SetStartCoordinates(inputCoordinates):ret +SetDestinationCoordinates(inputCoordinates):ret bool +UpdateRobotLocation(inputCoordinates):ret +GetRobotCoordinates():returnType Point3f +InitializeMap(Point, length, width):returnType +CheckReachedDestination():returnType bool +CheckValidCoordinates(): returnType bool +DisplayMapImage():returnType bool	eturnType bool f pe bool pool	+ getWheelBase(): double + getAxleWidth(): double + getCarVelocity(): double + getCarState(): Point3f + setWheelBase(double I): bool + setAxleWidth(double w): bool + setSteeringAngleAndCarVelocity(Eigen::Vector2d controllerOutput): bool + setCarState(Point3f state): bool + calcAckermanParameters(): Point3f + checkAngleConstraints(): bool
	PID	
	- kp : Eigen::Matrix <double, 2,="" 3=""> - ki : Eigen::Matrix<double, 2,="" 3=""> - kd : Eigen::Matrix<double, 2,="" 3=""> - lastError : Eigen::Vector3d - errorSum : Eigen::Vector3d</double,></double,></double,>	
· · ·	+ getKp(): Eigen::Matrix <double, 2,="" 3=""> + getKi(): Eigen::Matrix<double, 2,="" 3=""> + getKd(): Eigen::Matrix<double, 2,="" 3=""> + setKp(kpIn: Eigen::Matrix<double, 2,="" 3="">): b + setKi(kiIn: Eigen::Matrix<double, 2,="" 3="">): book + setKd(kdIn: Eigen::Matrix<double, 2,="" 3="">): b + getControllerOutput(targetState: Eigen::Vec Eigen::Vector3d): Eigen::Vector2d + resetErrors(): void</double,></double,></double,></double,></double,></double,>	ool Control of the co