| | |
|---|---|
| IHubF | Inertia tensor for front hub, must be symmetric and +ve. |
| IWheelF | Front wheel moment of inertia about its axle. |
| mHubR | Unsprung mass of a rear corner. |
| IHubR | Inertia tensor for rear hub, must be symmetric and +ve. |
| IWheelR | Rear wheel moment of inertia about its axle. |
| mHubF | Unsprung mass of a front corner. |
| IHubF | Inertia tensor for front hub, must be symmetric and +ve. |
| xTrackF | Horizontal distance between the front wheel centres. |
| xTrackR | Horizontal distance between the rear wheel centres. |
| rWeightBalF | Proportion of total weight born by the front axle. |
| rRideF | Position at which hRideF is measured (in body coordinates). |
| rUndertrayFront | Position of the front plank contact point. |
| rUndertrayMid | Position of the mid plank contact point. |
| rUndertrayRear | Position of the rear plank contact point. |
| | |
| kUndertrayFront | Stiffness of the front plank contact point. |
| kUndertrayMid | Stiffness of the mid plank contact point. |
| kUndertrayRear | Stiffness of the rear plank contact point. |
| muChassis | The coefficient of friction between the chassis and the ground. |
| kVerticalSuspensionComplianceF | Stiffness between front hubs and chassis when suspension is locked. |
| kVerticalSuspensionComplianceR | Stiffness between rear hubs and chassis when suspension is locked. |
| bFlexJointLockedF | Option to lock the axis for the front chassis body flex joint. |
| bFlexJointLockedR | Option to lock the axis for the rear chassis body flex joint. |
| rRackRatio | Steering rack movement per rotation of the steering wheel. |
| zCoG | Centre of gravity height (remember, z is +ve downwards!). |
| bSteeringCompliance | Option to treat steering as compliant. |

## Asymmetric Options

- Crossweight, and lateral weight distribution (defined under carRunningMass) can be defined.
- With the asymmetric add-on, `hRideFSetupL` or `hRideRSetupL` (with the corresponding reference point `rRideFL` or `rRideRL`) can be used to define the left-hand ride height separate from the right-hand side. 3 points can be used to define a plane; therefore we can define either [`hRideFSetup`, `hRideFSetupL`, `hRideRSetup`] or [`hRideFSetup`, `hRideRSetup`, `hRideRSetupL`].