

```

class AirSimClientActor() extends Actor {
  ...

  def startedReceive(client: Client): Receive = {
    case GetMultirotorState(vehicleName) =>
      sender ! client.callApply(
        "getMultirotorState", Array(vehicleName)
      ).toString

    case GetCameraImage(vehName, camName, imageType) =>
      val image = client.callApply(
        "simGetImage", Array[AnyRef](camName, imageType, vehName)
      ).toString
      sender ! (if (isValidImage(image)) Some(image) else None)

    ...
  }
}

val airSimClientRef = system.actorOf(Props[AirSimClientActor])

val getStateFlow =
  Flow[String]
    .map(name => GetMultirotorState(name))
    .ask[String](parallelism = 4)(airSimClientRef)

```