DistanceSensor

+sensor(sensor)
+offset(int)
+robotDiameter(robotDiameter)
+angle(sensorAngle)
+tileSize(tileSize)
+maxDetect(float)
+detectionLimit(detectionLimit)
+val(sensor)

+__init__(self, sensor, sensorAngle, robotDiameter, tileSize, timeStep, detectionLimit=1): set
+getDistance(self): get
+__getAngle(self, globalRotation): get
+getGlobalDetection(self, globalRotation, robotPos): get

aStarNode

+ parent(aStarNode)
+position(list, int)
+g(int)
+h(int)
+f(int)

+__eq__(self, other): get
+__init__(self, parent=None, position=None)

ColourSensor

+distance(self.distance = distancefromCenter)
+sensor(sensor)
+r(int)
+g(int)
+b(int)

+__init__(self, sensor, distancefromCenter, timeStep): set
+getPosition(self, robotGlobalPosition, robotGlobalRotation): get
+__update(self): set
+__isTrap(self): get
+__isSwamp(self): get
+__isCheckpoint(self): get
+__isNormal(self): get
+getTileType(self): get

Gyroscope

+sensor(gyro)
+oldTime(float)
+index(index)

+__init__(self, gyro, index, timeStep): set
+def update(self, time, currentRotation): get

Wheel

+__init__(self, wheel, maxVelocity): set
+move(self, ratio): set

HeatSensor

+sensor(sensor)
+threshold(threshold)

+__init__(self, sensor, thershold, timeStep): set
+isClose(self): get

Camer000

+camera(camera)
+height
+ width
+tileRanges(tileRanges)
+classifyThresh(int)

+__init__(self, camera, tileRanges, timeStep): set
+getImg(self): get
+getVictimImagesAndPositions(self): get
+getVictimRange(self, pos, img): get
+getVictimRange(self, pos, img): get

Gps

+gps(gps)
+multiplainer(coordsMultiplier)

+__init__(self, gps,timeStep, coordsMultiplier=0): set
+getPosition(self): get

StateManager

+state(initialState)
+state(newState)

+__init__(self, initialState): set
+changeState(self, newState):set
+checkState(self, state): get

Emitter

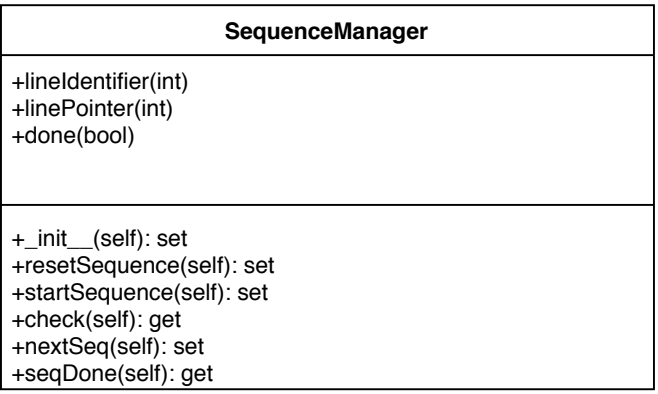
+emitter(emitter)
+divisor(coordsDivisor)

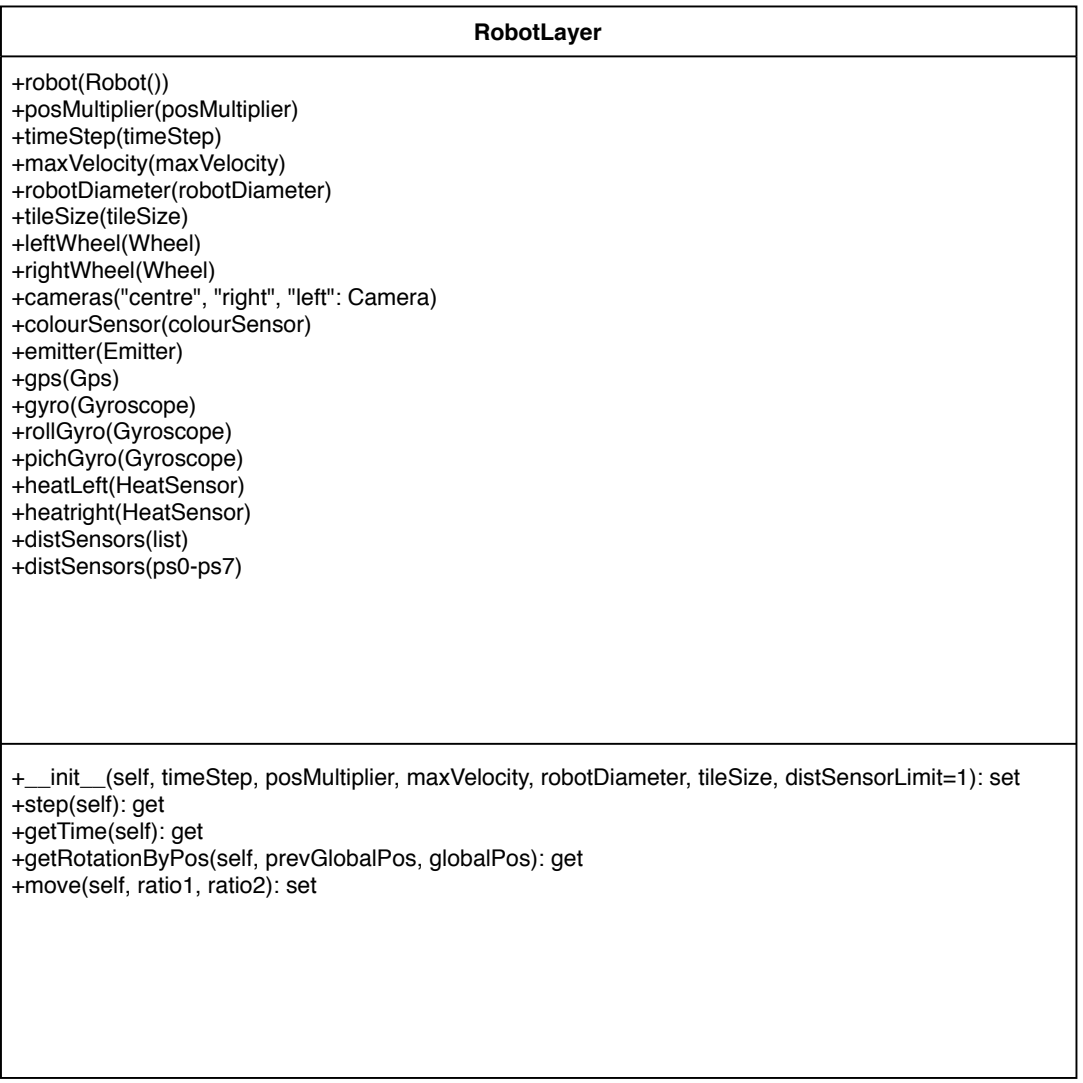
+__init__(self, emmitter, coordsDivisor=0): set
+sendMessage(self,pos, identifier): set

NodeGrid

+ grid(list)
+ center(int)
+ tileSize(int)
+offsets(int)
+orientations(dic)
+nodeColors(dic)
+colorNames(dic)

+__init__(self, x, y, tileSize, nodeTypeDict, offsets=[0,0])
+astar(self, start, end): get
+bfs(self, start, objectives, limit="undefined"): seget
+printMap(self): get
+getMat(self): get
+setPosition(self, position, val, orientation="center"): set
+getPosition(self, position, orientation="center"): get
+getTileNode(self, pos): get
+getPosfromTileNode(self, tileNode): get
+changeValue(self, pos, val, orientation="center"): set
+getValue(self, pos, orientation="center"): get
+getTile(self, position): get
+getOrientationInTile(self, inputPos): get


SequenceManager
+lineIdentifier(int)
+linePointer(int)
+done(bool)
+_init__(self): set
+resetSequence(self): set
+startSequence(self): set
+check(self): get
+nextSeq(self): set
+seqDone(self): get


RobotLayer
+robot(Robot())
+posMultiplier(posMultiplier)
+timeStep(timeStep)
+maxVelocity(maxVelocity)
+robotDiameter(robotDiameter)
+tileSize(tileSize)
+leftWheel(Wheel)
+rightWheel(Wheel)
+cameras("centre", "right", "left": Camera)
+colourSensor(colourSensor)
+emitter(Emitter)
+gps(Gps)
+gyro(Gyroscope)
+rollGyro(Gyroscope)
+pichGyro(Gyroscope)
+heatLeft(HeatSensor)
+heatright(HeatSensor)
+distSensors(list)
+distSensors(ps0-ps7)
+__init__(self, timeStep, posMultiplier, maxVelocity, robotDiameter, tileSize, distSensorLimit=1): set
+step(self): get
+getTime(self): get
+getRotationByPos(self, prevGlobalPos, globalPos): get
+move(self, ratio1, ratio2): set