

文档类别

杭州海康机器人股份有限公司

文档编号

---

厂内物流机器人控制系统 RCS-2000 V3.3  
对外任务接口文档

## 版权声明

本文档由杭州海康机器人股份有限公司开发，其版权受中华人民共和国版权法保护。海康机器人拥有本文的全部版权，未经本公司许可，任何单位及个人不得对本文中的任何部分进行转印、影印或复印。

## 信息反馈

海康机器人尽最大的努力保证本手册的准确性和完整性。如果您在使用中发现问题，希望及时将情况反馈给我们以完善产品，我们将非常感谢您的支持。

### 总公司联系方式

公司总机：0571-88967998

技术支持电话：0571-86611880（工作日9:30-17:30）

传真：0571-88805843

地址：浙江省杭州市滨江区丹枫路399号2号楼B楼304室

邮编：310051

公司E-mail：hikrobot@hikrobotics.com

公司网站：www.hikrobotics.com

## 目录

1 协议概述 .....	6
2 常用接口* .....	9
2.1 调度系统提供的接口 .....	9
2.1.1 生成任务单* .....	9
2.1.2 继续执行任务* .....	15
2.1.3 取消任务* .....	17
2.2 上层平台提供的接口 .....	20
2.2.1 任务执行通知* .....	20
3 可选接口 .....	23
3.1 调度系统提供的接口 .....	23
3.1.1 任务优先级设置 .....	23
3.1.2 货架与位置绑定、解绑 .....	25
3.1.3 货架与物料绑定、解绑 .....	28
3.1.4 位置禁用与启用 .....	29
3.1.5 地图位置信息同步 .....	30
3.1.6 查询货架储位与物料批次关系 .....	33
3.1.7 仓位禁用与启用 .....	35
3.1.8 容器与仓位绑定、解绑 .....	37
3.1.9 查询任务状态 .....	39
3.1.10 查询 AGV 状态 .....	41
3.1.11 停止 AGV .....	44

3.1.12 恢复 AGV .....	46
3.1.13 区域清空/释放.....	48
3.1.14 预调度对外接口 .....	50
3.1.15 清空巷道 .....	53
3.1.16 料箱出库 TPS (CTU+分拨墙) .....	55
3.1.17 料箱回库 TPS (CTU+分拨墙) .....	58
3.1.18 料箱顺序出库 (CTU) .....	61
3.1.19 料箱取放回调 (CTU) .....	66
3.1.20 物料类型与位置绑定、解绑 (移载机器人协议) .....	68
3.2 上层平台提供的接口 .....	70
3.2.1 告警推送通知.....	70
3.2.2 绑定解绑通知.....	72
3.2.3 校验料箱是否可入库 (CTU) .....	74
3.2.4 申请回库仓位 (CTU) .....	76
4 典型调度场景 .....	78
场景一：背货架 AGV 在货架底下待命任务调度场景.....	78
场景二：普通背货架任务调度场景.....	79
场景三：辊筒 AGV 任务调度场景 .....	79
场景四：CTU 输送线预调度+入库.....	81
场景五：CTU 梳齿式工作站入库.....	82
其他 CTU 场景接口使用.....	82
5 接口初步对接接入参示例 .....	83

---

6 附件 .....	86
6.1AGV 常见状态列表.....	86
6.2 常用地图元素类型列表 .....	87
6.3 接口调用 DEMO.....	88
7 更新说明 .....	88

# 1 协议概述

协议统一使用 JSON 格式, 中文使用 `URLEncoder.encode("string", "UTF-8")`; 编号, 获取中文字段时, 使用 `URLDecoder.decode("string", "UTF-8")`;

接口中标\*的为常用接口, 初步对接时, 只需要调通以下接口:

1.生成任务单, 2.继续执行任务, 3.取消任务

如果上层系统需要接收任务的执行状态, 需要提供以下接口, 供调度系统回调:

1.任务执行通知接口

海康调度系统调用上层系统的接口, 获取连接超时时间默认为 30 秒, 数据返回超时时间默认为 60 秒, 超时情况下, 调度系统会返回连接失败。

## 对接约定

- 为接口统一并兼容, 所有的参数都为字符串格式。
- 文本涉及到的 AGV、robot、机器人术语为同一术语, 不要混淆。
- reqCode, 传入的参数与传出的参数一致, 使用 UUID 或其他能够标识唯一即可。
- 为兼容以前版本, 消息上报字段会比列出的字段要多, 上层平台根据业务截取需要的字段。
- 容器: 应用于叉车、CTU 等项目中, 料箱、托盘、载具都可称为容器。

## 请求参数

参数名	是否必填	描述
reqCode	必填	请求编号，每个请求都要一个唯一编号， 同一个请求重复提交， 使用同一编号。由上层系统设定。
reqTime	选填	请求时间戳 格式：“yyyy-MM-dd HH:mm:ss” 。由上层系统设定。
clientCode	选填	客户端编号，如 PDA， HCWMS 等。如果填写，需先在 RCS-2000 系统配置，上层系统调用时进行填写，当多系统调用时，调度系统可以进行调用方区分。
tokenCode	选填	令牌号，由调度系统颁发。如果填写，需先在 RCS-2000 系统配置，上层系统调用时进行填写。

## 返回值定义

参数名	是否必填	描述
reqCode	必填	请求编号返回，形成一一对应
code	必填	返回编号， “0” ：成功， 1~N：失败
message	必填	“0” ：成功 1~N：其他的详细描述
data	选填	返回的数据结构

## 结果码定义

结果码 (code)	描述
"0"	成功
"1"	参数相关的错误
"6"	重复发送, 上层系统不需要重发
"99"	其他未知错误, 调用失败后, 可以重试
"100"	该任务不存在, 上层系统不需重发, 需要人工介入处理

## RCS-2000 基础访问地址

REST 协议:

baseURL	http://IP:PORT/rcms/services/rest/hikRpcService
端口	与 WEB 端口一致, 默认端口: 8182

AGV 状态调用协议:

baseURL	http://IP:PORT/rcms-dps/rest/queryAgvStatus
端口	与 WEB 端口一致, 默认端口: 8182

## 上层系统基础访问地址

REST 协议

baseURL	http://IP:PORT/xxx/agv
---------	------------------------



## 2 常用接口\*

### 2.1 调度系统提供的接口

#### 2.1.1 生成任务单\*

接口名	genAgvSchedulingTask				
功能说明	上层系统平台发送调度请求, RCS 通过请求参数, 生成调度 AGV 任务单。 生成所有车型的搬运任务。				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号, 每个请求都要一个唯一编号, 同一个请求重复提交, 使用同一编号。
	reqTime	String	20	否	请求时间戳 格式: "yyyy-MM-dd HH:mm:ss" 。
	clientCode	String	16	否	客户端编号, 如 PDA, HCWMS 等。
	tokenCode	String	64	否	令牌号, 由调度系统颁发。
	taskTyp	String	16	是	任务类型, 与在 RCS-2000 端配置的主任务类型编号一致。 内置任务类型: 厂内货架搬运: F01

					厂内货架空满交换: F02 辊筒搬运接驳: F03 厂内货架出库 AGV 待命: F04 旋转货架: F05 厂内电梯任务: F06 <b>以下为叉车专用任务类型</b> 高位货架到工作台: F11 工作台到高位货架: F12 巷道到工作台: F13 工作台到巷道: F14 高位货架到工作台(接驳): F15 工作台到高位货架 (接驳): F16 巷道到工作台(接驳): F17 工作台到巷道(接驳): F18 叉车电梯主任务: F20
	ctnrTyp	String	16	否	容器类型 <b>(叉车/CTU 专用)</b> <b>叉车项目必传</b>
	ctnrCode	String	30	否	容器编号 <b>(叉车/CTU 专用)</b>
	ctnrNum	String	2	否	容器数量 (叉车堆叠专用), 默认值 1, 仅记录堆叠的数量不记录堆叠的 每个容器号
	taskMode	String	1	否	任务模式 0-普通 move 1-出库 move 2-入库 move 3-移库 move 说明: 1>任务模板中也可以配置模式, 如 果接口传了则优先使用接口传入的 值 2>出库 move 模式不能被打断, 普

					<p>通 move 和入库 move 可以被打断。 出库 move 执行后必须执行入库 move 或者移库 move, 若后续未带 move 标识, 则系统不会执行, 会将任务挂起。</p> <p>3&gt;支持潜伏、叉车、CTU 场景</p>
	wbCode	String	32	否	<p>工作位, 一般为机台或工作台位置, 与 RCS-2000 端配置的位置名称一致, 工作位名称为字母\数字\或组合, 不超过 32 位。</p>
	positionCodePath	Object[]	列表长度 小于 50	否	<p>位置路径: AGV 关键路径位置集合, 与任务类型中模板配置的位置路径一一对应。待现场地图部署、配置完成后可获取。</p> <p>positionCode: 位置编号, 单个编号不超过 64 位</p> <p>type: 位置类型说明:</p> <p>00 表示: 位置编号</p> <p>01 表示: 物料批次号</p> <p>02 表示: 策略编号 (含多个区域) 如: 第一个区域放不下, 可以放第二个区域</p> <p>03 表示: 货架编号, 通过货架编号找到货架所在位置</p> <p>04 表示: 区域编号, 在区域中查找可用位置</p> <p>05 表示: 仓位编号 <b>(叉车/CTU 专用)</b></p> <p>06 表示: 巷道编号</p> <p>07 表示: 容器编号</p> <p>08 表示: 巷道策略</p> <p>09 表示: 巷道区域</p> <p>10 表示: 巷道仓位</p>

					11 表示: 输送线 (机台) 编号 12 表示: CTU 工作台 (梳齿工作站) 编号 13 表示: 搬运巷道指定货架出库
	podCode	String	16	否	货架编号, 不指定货架可以为空
	podDir	String	4	否	“180”、“0”、“90”、“-90” 分别对应地图的“左”、“右”、“上”、“下”, 不指定方向可以为空; 如果终点不是工作台或工作台未配置操作方向则货架拣货方向 podDir 作为终点方向; 如果工作台配有操作方向, 则系统会根据货架拣货方向 podDir 和工作台操作方向计算货架终点方向
	podTyp	String	16	否	货架类型, 传空时表示随机找个货架 找空货架传参方式如下: -1: 代表不关心货架类型, 找到空货架即可. -2: 代表从工作位获取关联货架类型, 如果未配置, 只找空货架. 货架类型编号: 只找该货架类型的空货架.
	materialLot	String	32	否	物料批次或货架上的物料唯一编码, 生成任务单时, 货架与物料直接绑定时使用. (通过同时传 podCode 和 materialLot 来绑定或通过 wbCode 找到位置上的货架和 materialLot 来绑定) 如果是区域/策略内查找货架, 可以匹配对应物料批次的货架, 支持以

					下值: 0: 区域/策略内查找空货架 1: 区域/策略内查找满货架 其他值: 查找指定物料批次对应的货架 巷道任务通过该字段传入特征值
	materialType	String	32	否	物料类型, 仅移栽机器人协议专用必填, 其它车型任务不填
	priority	String	32	否	优先级, 从 (1~127) 级, 最大优先级最高。为空时, 采用任务模板的优先级。
	taskCode	String	64	否	任务单号, 选填, 不填系统自动生成, UUID 小于等于 64 位
	agvCode	String	5	否	AGV 编号, 填写表示指定某一编号的 AGV 执行该任务
	groupId	String	16	否	组编号 CTU 场景下用于按组出库, 同组任务优先拼车。如业务需要任务组间或组内按顺序出库, 则需调用料箱顺序出库 (CTU) 接口。 潜伏式场景下, 通过组号来管理顺序出库的顺序, 组号小的优先出库。
	agvTyp	String	4	否	设备类型
	positionSelStrategy	String	2	否	区域/策略中挑选货架以及根据物料批次挑选货架时的先进先出规则, 支持以下 4 个值: 1: 按照货架到达储位的时间顺序, 先进先出

					2: 按照货架到达储位的时间顺序, 先进后出 9: 按照货架绑定物料批次的时间顺序, 先进先出 10: 按照货架绑定物料批次的时间顺序, 先进后出
	data	String	2000	否	自定义字段.JSON 格式
应答	code	String	6	是	返回码
	message	String	64	是	返回消息
	reqCode	String	64	是	请求编号
	data	String	2000	否	自定义返回 (返回任务单号)
备注	<b>wbCode 和 positionCodePath 至少填写其中一项, 以确定任务中的位置信息。若任务中需要指定多个位置信息, 如起点和终点信息等, 请使用 positionCodePath。</b>				
示例	请求	REST: POST URL: <a href="http://IP:PORT/rcms/services/rest/hikRpcService/genAgvSchedulingTask">http://IP:PORT/rcms/services/rest/hikRpcService/genAgvSchedulingTask</a> <pre> {   "reqCode": "468513",   "reqTime": "",   "clientCode": "",   "tokenCode": "",   "taskTyp": "F01",   "ctnrTyp": "",   "ctnrCode": "",   "ctnrNum": "",   "taskMode": "",   "wbCode": "",   "positionCodePath": [{     "positionCode": "p01",     "type": "00"   },   {     "positionCode": "x02",     "type": "02"   } ] </pre>			

		<pre>] "podCode": "100001", "podDir": "0", "podTyp": "", "materialLot": "", "materialType": "", "priority": "1", "agvCode": "", "taskCode": "", "groupId": "", "agvTyp": "", "positionSelStrategy ": "", "data": "" }</pre>
	应答	<pre>{ "code": "0" , "data": "F01169C808C317111G", "message": "成功", "reqCode": "468513" }</pre>

### 2.1.2 继续执行任务\*

接口名	continueTask				
功能说明	上层系统平台发送继续调度请求，RCS-2000 获取 AGV 下一个动作，继续执行。				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号。

	reqTime	String	20	否	请求时间戳 格式: “yyyy-MM-dd HH:mm:ss” 。
	clientCode	String	16	否	客户端编号, 如 PDA, HCWMS 等。
	tokenCode	String	64	否	令牌号, 由调度系统颁发。
	wbCode	String	32	否	工作位, 与 RCS-2000 端配置的位置名称一致。
	podCode	String	16	否	货架号, 采用货架号触发的方式。
	agvCode	String	5	否	AGV 编号, 采用 AGV 编号触发的方式。
	taskCode	String	64	否	任务单号,选填, 不填系统自动生成, 必须为 64 位 UUID
	taskSeq	String	32	否	下一个子任务的序列, 指定第几个子任务开始执行, 校验子任务执行是否正确。不填默认执行下一个子任务。
	nextPositionCode	Object	40	否	<p>下一个位置信息, 在任务类型中配置外部设置时需要传入, 否则不需要设置。待现场地图部署、配置完成后可获取</p> <p>positionCode:位置编号</p> <p>type:对象类型定义:</p> <p>00:代表 nextPositionCode 是一个位置</p>



					02:代表 nextPositionCode 是一个策略
应答	code	String	6	是	返回码
	message	String	64	是	返回消息
	reqCode	String	64	是	请求编号
备注	taskCode、agvCode、wbCode 和 podCode 四个只填一个，填哪个需要与任务模板配置的触发类型一致，优先推荐 taskCode				
示例	请求	REST: POST <a href="http://IP:PORT/rcms/services/rest/hikRpcService/continueTask">http://IP:PORT/rcms/services/rest/hikRpcService/continueTask</a> <pre>{   "reqCode": "123",   "reqTime": "",   "clientCode": "",   "tokenCode": "",   "wbCode": "",   "podCode": "",   "agvCode": "",   "taskCode": "123456",   "taskSeq": "",   "nextPositionCode": {     "positionCode": "p02",     "type": "00"   } }</pre>			
	应答	<pre>{   "code": "0",   "message": "成功",   "reqCode": "123" }</pre>			

### 2.1.3 取消任务\*

接口名	cancelTask
-----	------------

功能说明	通过正在执行的任务编号,取消该任务,不再执行, AGV 如果背着货架, 取消类型为 0 时, 会把货架直接放在路上, AGV 为空闲状态, 需要人工介入发指令搬走或拖走货架; 取消类型为 1 时, 称为软取消, AGV 仍然背着货架, 通过货架所属库区或回库区域执行回库指令, 如果回库区域没有位置, 返回错误信息, 取消不成功。CTU 取到料箱后取消, 取消类型为 0 时, 会把料箱留在 CTU 上, 需要人工取走; 取消类型为 1 时, CTU 将料箱搬回指定的区域的空仓位, 若区域没位置, 返回错误信息, 取消不成功。CTU 只能软取消回库区, 支持双伸位场景, 不支持软取消回输送线等工作位。  叉车、辊筒车不支持软取消				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号, 每个请求都要一个唯一编号, 同一个请求重复提交, 使用同一编号。
	reqTime	String	20	否	请 求 时 间 截 格 式 : “yyyy-MM-dd HH:mm:ss” 。
	clientCode	String	16	否	客户端编号, 如 PDA, HCWMS 等。
	tokenCode	String	64	否	令 牌 号, 由调度系统颁发。由 RCS-2000 告知上层系统
	forceCancel	String	16	否	取消类型 0 表示: 取消后货架直接放地上 1 表示: AGV 仍然背着货架, 根据

					回库区域执行回库指令，只有潜伏车和 CTU 车支持。 若不传值，默认的取消模式为 0
	matterArea	String	16	否	forcecancel=1 时有意义， 回库区域编号， 如果为空，采用货架配置的库区。
	agvCode	String	5	否	取消该 AGV 正在执行的任务单
	taskCode	String	64	否	任务单编号，取消该任务单
应答	code	String	6	是	返回码
	message	String	64	是	返回消息
	reqCode	String	64	是	请求编号
备注	taskCode 和 agvCode 选一项填写，优先级从高到低依次为：agvCode、taskCode，都传了优先使用 agvCode，以确定需要取消哪个任务单。				
示例	请求	REST: POST <a href="http://IP:PORT/rcms/services/rest/hikRpcService/cancelTask">http://IP:PORT/rcms/services/rest/hikRpcService/cancelTask</a> <pre>{   "reqCode": "1541954B96B1112",   "reqTime": "",   "clientId": "",   "tokenCode": "",   "forceCancel": "1",   "matterArea": "abc",   "agvCode": "",   "taskCode": "123456" }</pre>			
	应答	<pre>{   "code": "0",   "message": "成功",   "reqCode": "1541954B96B1112" }</pre>			

## 2.2 上层平台提供的接口

### 2.2.1 任务执行通知\*

接口名	agvCallback				
功能说明	AGV 执行回调的方法，包含任务开始，走出储位，任务完成及任务取消。 任务取消通知需要取消任务单才会上报，取消子任务不会上报。其他动作通知是单个任务组或子任务相应动作完成后进行上报。 <b>注意：是否通知需要在任务模板配置。</b>				
接口协议	REST				
提供方	上层平台				
调用方	RCS-2000				
备注	为兼容以前版本，消息字段会比列出的字段要多，上层平台根据业务截取需要的字段。				
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号
	reqTime	String	20	是	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”
	cooX	Number	8	否	地码 X 坐标(mm)：任务完成时有值
	cooY	Number	8	否	地码 Y 坐标(mm)：任务完成时有值
	currentPositionCode	String	32	是	当前位置编号 任务开始：该位置为任务起点

					走出储位：该位置为任务起点 任务单取消：该位置为工作位编号 任务结束：该位置为任务终点 取放申请：取放料箱的点
	data	String	2000	否	自定义字段，不超过 2000 个字符
	mapCode	String	16	否	地图编号
	mapDataCode	String	32	否	地码编号：任务完成时有值
	stgBinCode	String	32	否	仓位编号：叉车与 CTU 任务时有值
	method	String	16	是	方法名，可使用任务类型做为方法名 由 RCS-2000 任务模板配置后并告知上层系统 默认使用方式： start：任务开始 outbin：走出储位 end：任务结束 cancel：任务单取消 apply：CTU 料箱取放申请
	podCode	String	16	否	货架编号：背货架时有值
	podDir	String	4	否	"180" , " 0" , " 90" , " -90" 分 别对应地图的 " 左" , " 右" , " 上" , " 下" : 任务完成时有值
	materialLot	String	32	否	物料编号
	materialType	String	32	否	物料类型，仅移栽机器人协议专用 必填，其它车型任务不填

	robotCode	String	5	是	AGV 编号 (同 agvCode )
	taskCode	String	64	是	当前任务单号
	wbCode	String	32	否	工作位, 与 RCS-2000 端配置的位置名称一致。任务完成时有值, 与生成任务单接口中的 wbCode 一致。
	ctnrCode	String	30	否	容器编号
	ctnrType	String	2	否	容器类型
	roadWayCode	String	16	否	巷道编号
	seq	String	2	否	巷道内顺序号 巷道尾是 0, 到巷道头依次递增 1
	eqpCode	String	32	否	设备编号, 如梳齿式工作站、输送线等, 一般使用于 CTU 场景。系统根据仓位定位到关联的设备编号。
应答	code	String	6	是	返回码
	message	String	64	是	返回消息
	reqCode	String	64	是	请求编号
示例	请求	REST: POST <a href="http://IP:PORT/xxx/agv/agvCallbackService/agvCallback">http://IP:PORT/xxx/agv/agvCallbackService/agvCallback</a> { "reqCode": "1541954B96B1112", }			

		<pre>"reqTime": "2019-04-03 10:08:06", "cooX": "3000", "cooY": "21999", "currentPositionCode": "p02", "data": "", "mapCode": "AA", "mapDataCode": "002069AA015172", "stgBinCode": "", "method": "end", "podCode": "100001", "podDir": "", "robotCode": "6001", "taskCode": "test169E0F39740116Q", "wbCode": "p02", "materialLot": "", "materialType": "", "ctnrCode": "", "ctnrType": "", "roadWayCode": "", "seq": "", "eqpCode": "" }</pre>
	应答	<pre>{   "code": "0",   "message": "成功",   "reqCode": "1541954B96B1112" }</pre>

## 3 可选接口

### 3.1 调度系统提供的接口

#### 3.1.1 任务优先级设置

接口名	setTaskPriority
-----	-----------------

功能说明	设置任务优先级(1~127 级),值越大,优先级越高。					
	优先级生效仅在系统中 AGV 数量不足, 存在多个优先级不同的任务时候, 会按照优先级的先后顺序分配 AGV 执行。					
	如果设置优先级失败, 则全部失败。					
接口协议	REST					
提供方	RCS-2000					
调用方	上层系统					
请求参数	参数名		数据类型	最大长度	是否必填	备注
	reqCode		String	32	是	请求编号, 每个请求都要一个唯一编号, 同一个请求重复提交, 使用同一编号。
	reqTime		String	20	否	请求时间戳 格式: "yyyy-MM-dd HH:mm:ss" 。
	clientCode		String	16	否	客户端编号, 如 PDA, HCWMS 等。
	tokenCode		String	64	否	令牌号,由调度系统颁发。
	priorities[列表]	taskCode	String	64	是	必填, 正在执行的任务单编号
		priority	String	32	是	必填, 优先级, 从 (1~127) 级, 最大优先级最高
应答	code		String	6	是	返回码
	message		String	64	是	返回消息
	reqCode		String	64	是	请求编号



备注	设置优先级适用于还未派 AGV 执行的任务，设置优先级后，可根据优先级分配 AGV 执行。任务分配 AGV 后，任务优先级可对路径规划产生影响。当路径产生冲突时，优先级高的任务优先使用道路。	
示例	请求	REST: POST <a href="http://IP:PORT/rcms/services/rest/hikRpcService/setTaskPriority">http://IP:PORT/rcms/services/rest/hikRpcService/setTaskPriority</a> <pre>{   "reqCode": "1234567",   "reqTime": "",   "clientCode": "",   "tokenCode": "",   "priorities": [{     "priority": "1",     "taskCode": "1232"   },   {     "priority": "2",     "taskCode": "3214"   } ]</pre>
	应答	<pre>{   "code": "0" ,   "message": "成功",   "reqCode": "1234567" }</pre>

3.1.2 货架与位置绑定、解绑

接口名	bindPodAndBerth
功能说明	货架与储位的关系绑定和解绑，绑定后系统可以通过货架找到对应位置，货架编号记录在点位上，解绑时货架编号从点位上清空
接口协议	REST

提供方	RCS-2000				
调用方	上层系统				
备注	解绑时，货架与位置都要传，用于检验，避免误操作。				
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号。
	reqTime	String	20	否	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”。
	clientCode	String	16	否	客户端编号，如 PDA, HCWMS 等。
	tokenCode	String	64	否	令牌号，由调度系统颁发。
	podCode	String	16	是	货架编号
	positionCode	String	32	是	位置编号，地图位置的别名，能任意命名(字母+数字)，但要唯一，由 RCS-2000 界面配置。
	podDir	String	6	否	货架方向 “0”：横向，“1”：纵向 横向：货架长与 X 轴一致； 纵向：货架宽与 X 轴一致。 优先使用地图点位中配置的值，点位若配置了横向或竖向，该参数不

					生效，点位若配置了全向，按该参数操作。不传默认为横向。
	characterValue	String	64	否	特征值，搬运巷道绑定货架可用，特征值可以是物料种类、批次等信息
	indBind	String	1	是	"1": 绑定, "0": 解绑
应答	code	String	6	是	返回码
	message	String	64	是	返回消息
	reqCode	String	64	是	请求编号
示例	请求	REST: POST URL: <a href="http://IP:PORT/rcms/services/rest/hikRpcService/bindPodAndBerth">http://IP:PORT/rcms/services/rest/hikRpcService/bindPodAndBerth</a> <pre>{   "reqCode": "12345678",   "reqTime": "",   "clientCode": "",   "tokenCode": "",   "podCode": "100001",   "positionCode": "p05",   "podDir": "0",   "characterValue": "0",   "indBind": "1" }</pre>			
	应答	<pre>{   "code": "0",   "message": "成功",   "reqCode": "12345678" }</pre>			

### 3.1.3 货架与物料绑定、解绑

接口名	bindPodAndMat				
功能说明	货架与物料批次的关系绑定, 系统可以通过物料批次找到对应货架。				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
备注	解绑时, 货架与物料都要传, 用于检验, 避免误操作。				
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号, 每个请求都要一个唯一编号, 同一个请求重复提交, 使用同一编号。
	reqTime	String	20	否	请求时间戳, 格式: "yyyy-MM-dd HH:mm:ss" 。
	clientCode	String	16	否	客户端编号, 如 PDA, HCWMS 等。
	tokenCode	String	64	否	令牌号, 由调度系统颁发。
	podCode	String	16	是	货架编号
	materialLot	String	32	是	物料批次
	indBind	String	1	是	"1": 绑定, "0": 解绑
应答	code	String	6	是	返回码
	message	String	64	是	返回消息
	reqCode	String	64	是	请求编号

示例	请求	REST: POST <a href="http://IP:PORT/rcms/services/rest/hikRpcService/bindPodAndMat">http://IP:PORT/rcms/services/rest/hikRpcService/bindPodAndMat</a> <pre>{   "reqCode": "1541954B96B1112",   "reqTime": "",   "clientCode": "",   "tokenCode": "",   "podCode": "100001",   "materialLot": "123",   "indBind": "1" }</pre>
	应答	<pre>{   "code": "0",   "message": "成功",   "reqCode": "1541954B96B1112" }</pre>

### 3.1.4 位置禁用与启用

接口名	lockPosition				
功能说明	位置禁用与启用, 位置禁用后, 从区域中寻找位置时, 不能被找到。				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
备注					
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号, 每个请求都要一个唯一编号, 同一个请求重复提交, 使用同一编号。
	reqTime	String	20	否	请求时间戳, 格式: "yyyy-MM-dd"

					HH:mm:ss" 。
	clientCode	String	16	否	客户端编号, 如 PDA, HCWMS 等。
	tokenCode	String	64	否	令牌号, 由调度系统颁发。
	positionCode	String	32	是	位置编号, 地图位置的别名, 能任意命名(字母+数字), 但要唯一, 由 RCS-2000 界面配置。
	indBind	String	1	是	"1": 启用, "0": 禁用
应答	code	String	6	是	返回码
	message	String	64	是	返回消息
	reqCode	String	64	是	请求编号
示例	请求	REST: POST <a href="http://IP:PORT/rcms/services/rest/hikRpcService/lockPosition">http://IP:PORT/rcms/services/rest/hikRpcService/lockPosition</a> <pre>{   "reqCode": "1541954B96B1112",   "reqTime": "",   "clientCode": "",   "tokenCode": "",   "positionCode": "p02",   "indBind": "1" }</pre>			
	应答	<pre>{   "code": "0",   "message": "成功",   "reqCode": "1541954B96B1112" }</pre>			

### 3.1.5 地图位置信息同步

接口名	syncMapDatas
-----	--------------

功能说明	全量同步地码数据					
接口协议	REST					
提供方	RCS-2000					
调用方	上层系统					
请求参数	参数名		数据类型	最大长度	是否必填	备注
	reqCode		String	32	是	请求编号，每个请求都要一个唯一编号， 同一个请求重复提交， 使用同一编号。
	reqTime		String	20	否	请求时间戳 格式：“yyyy-MM-dd HH:mm:ss” 。
	clientCode		String	16	否	客户端编号，如 PDA，HCWMS 等。
	tokenCode		String	64	否	令牌号,由调度系统颁发。
	mapDataCode		String	32	否	地码编号，唯一标识, 代表只获取这个位置
	mapShortName		String	32	是	地图简称, 同步该地图的位置信息
	dataTyp		String	6	否	地图元素类型, 为空时, 代表同步该地图全部地码
	应答	code		String	6	是
message		String	64	是	返回消息	
reqCode		String	64	是	请求编号	
data		cooX	String	8	是	地码 X 坐标(mm)
		cooY	String	8	是	地码 Y 坐标(mm)

	[列表]	dataTyp	String	2	是	地图元素类型,常用类型: 1-储位 10-工作台 11-充电桩 20-缓冲区, 55-巷道存储区
		direction	String	8	否	工      作      台      方      向 "180" ," 0" ," 90" ," -90" 分别 代表" 左" ," 右" ," 上" ," 下" 工作台方向为工作人员面向货架拣货 的方向
		mapCode	String	16	是	地图编号
		mapDataCode	String	32	是	地码编号, 唯一标识
		positionCode	String	32	是	位置编号, 地图位置的别名, 能任意 命名(字母+数字), 但要唯一, 由 RCS-2000 界面配置。
		podCode	String	16	否	货架编号, 若储位上有货架则有值。
		berthType	String	2	否	储位类型, 1-外层储位, 2-内层储位, 3-普通储位; 位置是储位类型时必填。
备注	为兼容以前版本, 返回字段会比列出的字段要多, 上层平台根据业务截取需要的 字段。					
示例	请求	REST: POST <a href="http://IP:PORT/rcms/services/rest/hikRpcService/syncMapDatas">http://IP:PORT/rcms/services/rest/hikRpcService/syncMapDatas</a> { "reqCode": "1541954B96B1112", "reqTime": "", "clientCode": "", "tokenCode": "", "mapDataCode": "xxxxxx", "mapShortName": "xxxxxx", "dataTyp": "" }				



	应答	<pre>{   "code": "0",   "message": "成功",   "reqCode": "1541954B96B1112",   "data": [{     "berthType": "3",     "cooX": "17000.0",     "cooY": "18000.0",     "dataTyp": "1",     "direction": "0",     "mapCode": "AA",     "mapDataCode": "011724AA012414",     "podCode": "100001",     "positionCode": "011724AA012414"   },   {     "berthType": "3",     "cooX": "11000.0",     "cooY": "21999.0",     "dataTyp": "10",     "direction": "0",     "mapCode": "AA",     "mapDataCode": "007586AA015172",     "podCode": "100002",     "positionCode": "104"   } ]</pre>
--	----	--

3.1.6 查询货架储位与物料批次关系

接口名	queryPodBerthAndMat
功能说明	查询货架\储位与物料批次绑定关系
接口协议	REST
提供方	RCS-2000
调用方	上层系统
备注	货架,位置,批次,地图简称不能同时为空

		参数名	数据类型	最大长度	是否必填	备注
请求参数		reqCode	String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号。
		reqTime	String	20	否	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”。
		clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。
		tokenCode	String	64	否	令牌号，由调度系统颁发。
		podCode	String	16	否	货架编号
		materialLot	String	32	否	物料批次
		positionCode	String	16	否	位置编号，地图位置的别名，能任意命名(字母+数字)，但要唯一，由 RCS-2000 界面配置。
		areaCode	String	16	否	区域编号
		mapShortName	String	16	否	地图简称
		code	String	6	是	返回码
应答		message	String	64	是	返回消息
		reqCode	String	64	是	请求编号
	data	areaCode	String	16	否	区域编号
		materialLot	String	64	否	物料批次
		podCode	String	16	是	货架编号

		mapDataCode	String	32	是	地码编号, 唯一标识
		positionCode	String	32	是	位置编号, 地图位置的别名, 能任意命名(字母+数字), 但要唯一, 由RCS-2000 界面配置。
示例	请求	REST: POST URL: <a href="http://IP:PORT/rcms/services/rest/hikRpcService/queryPodBerthAndMat">http://IP:PORT/rcms/services/rest/hikRpcService/queryPodBerthAndMat</a> { "reqCode": "1541954B96B1110", "reqTime": "", "clientCode": "", "tokenCode": "", "podCode": "", "materialLot": "", "positionCode": "", "areaCode": "", "mapShortName": "test" }				
	应答	{ "code": "0", "message": "成功", "reqCode": "1541954B96B1110", "data": [{ "areaCode": "", "materialLot ": "", "podCode": "100001", "mapDataCode": "P02", "positionCode": "P02" }, { "areaCode": "", "materialLot ": "", "podCode": "100002", "mapDataCode": "P03", "positionCode": "P03" } ] }				

3.1.7 仓位禁用与启用

接口名	blockStgBin					
功能说明	该接口用做批量禁用启用仓位，要么全部更新成功，要么全部更新失败。 生成货架初始化的仓位都是已启用状态的。					
接口协议	REST					
提供方	RCS-2000					
调用方	上层系统					
备注	上层管理仓位时需保证与 RCS-2000 的仓位状态保持一致。					
请求参数	参数名		数据类型	最大长度	是否必填	备注
	reqCode		String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号。
	reqTime		String	20	否	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”。
	clientCode		String	16	否	客户端编号，如 PDA，HCWMS 等。
	tokenCode		String	64	否	令牌号，由调度系统颁发。
	data	stgBinCode	String	32	是	仓位编号
		action	String	2	是	操作类型：0.启用仓位；1.启用仓位
应答	code		String	6	是	返回码
	message		String	64	是	返回消息
	reqCode		String	64	是	请求编号
示例	请求	REST: POST URL: <a href="http://IP:PORT/rcms/services/rest/hikRpcService/blockStgBin">http://IP:PORT/rcms/services/rest/hikRpcService/blockStgBin</a>				

		<pre>{   "reqCode": "1541954B96B1110",   "reqTime": "",   "clientCode": "",   "tokenCode": "",   "data": [{     "stgBunCode": "",     "action": ""   }]{     "stgBunCode": "",     "action": ""   } }</pre>
	应答	<pre>{   "code": "0",   "message": "成功",   "reqCode": "1541954B96B1110", }</pre>

### 3.1.8 容器与仓位绑定、解绑

接口名	bindCtnrAndBin				
功能说明	仓位与容器的关系绑定, 容器类型编号写入仓位表。				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
备注	解绑时, 容器类型与仓位号都要传, 用于检验, 避免误操作。				
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号, 每个请求都要一个唯一

					编号，同一个请求重复提交，使用同一编号。 <b>由上层系统提供</b>
	reqTime	String	20	否	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”。 <b>由上层系统提供</b>
	clientCode	String	16	否	客户端编号，如 PDA, HCWMS 等。 <b>由 RCS-2000 告知上层系统</b>
	tokenCode	String	64	否	令牌号，由调度系统颁发。 <b>由 RCS-2000 告知上层系统</b>
	ctnrCode	String	30	否	容器编号
	ctnrTyp	String	16	是	容器类型
	stgBinCode	String	32	否	仓位编号， <b>与仓位绑定解绑必填</b>
	binName	String	32	否	客仓编号， <b>对应多个仓位编号时报错</b>
	characterValue	String	64	否	特征值，叉车巷道绑定容器可用
	positionCode	String	32	否	地图数据编号， <b>用于虚拟货架仓位的绑定解绑</b> 。当绑定时，从该点位上的虚拟货架找一个空仓位进行绑定，顺序是从下至上搜索。当解绑时根据容器类型和容器号进行匹配，若结果有多条则从优先从上部的仓位进行解绑。
	indBind	String	1	是	"1"：绑定，"0"：解绑

应答	code	String	返回码
	data	String	自定义返回
	message	String	返回消息
	reqCode	String	请求编号
示例	请求	REST: POST <a href="http://IP:PORT/rcms/services/rest/hikRpcService/bindCtnrAndBin">http://IP:PORT/rcms/services/rest/hikRpcService/bindCtnrAndBin</a> { "reqCode": "12345678", "reqTime": "", "clientCode": "", "tokenCode": "", "ctnrCode": "", "ctnrTyp": "C1", "stgBinCode": "p05", "positionCode": "p05", "indBind": "1" }	
	应答	{ "code": "0", "data": "", "message": "成功", "reqCode": "12345678" }	

### 3.1.9 查询任务状态

接口名	queryTaskStatus
功能说明	通过任务编号查询任务当前执行状态，支持批量查询。
接口协议	REST
提供方	RCS-2000
调用方	上层系统

备注		参数错误或者任务单未找到返回均为成功。				
请求参数	参数名		数据类型	最大长度	是否必填	备注
	reqCode		String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号。
	reqTime		String	20	否	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”。
	clientCode		String	16	否	客户端编号，如 PDA，HCWMS 等。
	tokenCode		String	64	否	令牌号,由调度系统颁发。
	taskCodes		String[]	64	否	任务单编号数组 任务单编号数组与 AGV 编号只能传其中之一，批量查询只能使用任务单编号数组
	agvCode		String	5	否	AGV 编号 任务编号数组与 AGV 编号只能传其中之一
	应答	code		String	6	是
data [列表]		taskCode	String	64	是	任务单编号
		taskTyp	String	16	是	任务类型
		taskStatus	String	2	是	任务状态：1-已创建，2-正在执行，5-取消完成，9-已结束，10-被打断
		agvCode	String	16	否	AGV 编号，任务分配车后有值
message		String	64	是	返回消息	
reqCode		String	64	是	请求编号	
示例	请求		REST: POST <a href="http://IP:PORT/rcms/services/rest/hikRpcService/queryTaskStatus">http://IP:PORT/rcms/services/rest/hikRpcService/queryTaskStatus</a>			



		<pre>{   "reqCode": "1541954B96B1110",   "reqTime": "",   "clientCode": "",   "tokenCode": "",   "agvCode": "",   "taskCodes": ["123", "234"] }</pre>
	应答	<pre>{   "code": "0",   "message": "成功",   "reqCode": "1541954B96B1110",   "data": [{     "taskCode": "234",     "taskStatus": "2",     "agvCode": "",     "taskTyp": "F01"   },   {     "taskCode": "123",     "taskStatus": "9",     "agvCode": "",     "taskTyp": "F01"   } ] }</pre>

3.1.10 查询 AGV 状态

接口名	queryAgvStatus
功能说明	通过请求参数, 查询 AGV 状态信息, 包括电池电量。
接口协议	REST
提供方	RCS-2000
调用方	上层系统

备注	调用频次： 100 车以下： 5 秒， 100~200 车： 10 秒， 200 ~300 车： 15 秒 调用路径： http://IP:8182/rcms-dps/rest/queryAgvStatus					
请求参数	参数名		数据类型	最大长度	是否必填	备注
	reqCode		String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号。
	reqTime		String	20	否	请求时间戳, 格式: “yyyy-MM-dd HH:mm:ss” 。
	clientCode		String	16	否	客户端编号, 如 PDA, HCWMS 等。
	tokenCode		String	64	否	令牌号,由调度系统颁发。
	mapCode		String	32	是	地图编号
应答	code		String	6	是	返回码
	data	robotCode	String	5	是	机器人编号
		robotDir	String	4	是	机器人方向 (范围 -180~360 度)
		robotIp	String	64	否	机器人 IP
		battery	String	4	是	机器人电量, 范围: 0-100
		posX	String	8	是	机器人 x 坐标,单位:毫米
		posY	String	8	是	机器人 y 坐标,单位:毫米
		mapCode	String	32	是	机器人所在地图
		speed	String	6	是	机器人当前速度, 单位: mm/s
		status	String	6	是	机器人状态 <b>AGV 常见状态编号和描述 见附件 6.1</b>
		exclType	String	1	是	是否已被排除, 被排除后不接受新

						任务（1-排除，0-正常）
		stop	String	1	是	是否暂停 0-否 1-是
		podCode	String	16	否	背货架的编号
		podDir	String	16	否	背货架的方向
		path	String[]	300	否	执行路径,单位是毫米, 格式 x 轴,y 轴,方向 示例: [ "[x,y,dir]" , "[x,y,dir]" , "[x,y,dir]" ]
	message		String	64	是	返回消息
	reqCode		String	64	是	请求编号
示例	请求	REST: POST http://IP:8182/rcms-dps/rest/queryAgvStatus { "reqCode": "1541954B96B1112", "reqTime": "", "clientId": "", "tokenCode": "", "mapCode": "" }				
	应答	{ "code": "0", "message": "成功", "reqCode": "1541954B96B1112", "data": [{ "robotCode": "1001", "robotDir": "180", "robotIp": "", "battery": "80", "posX": "1.0", "posY": "2.0", "mapCode": "", "speed": "", "status": "1", "exclType": "0", "stop": "1", "podCode": "200001", "podDir": "90", "path": [				

		<pre>"[10000,20000,90]", "[20000,30000,-90]", "[20000,30000,180]", "[30000,40000,0]"  ], }, {   "robotCode": "1001",   "robotDir": "180",   "robotIp": "",   "battery": "80",   "posX": "1.0",   "posY": "2.0",   "mapCode": "",   "speed": "",   "status": "1",   "exclType": "0",   "stop": "1",   "podCode": "200001",   "podDir": "90",   "path": [] }] }</pre>
--	--	---

### 3.1.11 停止 AGV

接口名	stopRobot				
功能说明	停止指定 AGV 或全部 AGV				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
备注					
请求参数	参数名	数据类型	最大长度	是否必填	备注

	reqCode	String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号。
	reqTime	String	20	否	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”。
	clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。
	tokenCode	String	64	否	令牌号，由调度系统颁发。
	robotCount	String	64	否	停止的机器人数量，-1 表示所有机器人
	mapShortName	String	32	否	AGV 所在地图的简称， 注：robotCount 填 -1 的话，mapShortName 必填，代表停止该地图中的全部 AGV
	robots	String[]	16	否	具体机器人编号列表
应答	code	String	6	是	返回码
	message	String	64	是	返回消息
	reqCode	String	64	是	请求编号
示例	请求	REST: POST http://IP:PORT/rcms/services/rest/hikRpcService/stopRobot { "reqCode": "1541954B96B1112", "reqTime": "", "clientCode": "", "tokenCode": "", "robotCount": "2", "mapShortName": "", "robots": [ "1001", "1002" ]			

		<pre>] }</pre>
	应答	<pre>{   "code": "0" ,   "message": "成功",   "reqCode": "1541954B96B1112" }</pre>

### 3.1.12 恢复 AGV

接口名	resumeRobot				
功能说明	恢复 AGV, 恢复后继续执行未完成任务				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
备注					
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号, 每个请求都要一个唯一编号, 同一个请求重复提交, 使用同一编号。
	reqTime	String	20	否	请求时间戳, 格式: "yyyy-MM-dd HH:mm:ss" 。
	clientCode	String	16	否	客户端编号, 如 PDA, HCWMS 等。
	tokenCode	String	64	否	令牌号, 由调度系统颁发。
	robotCount	String	64	否	恢复的机器人数量, -1 表示所有机器

					人
	mapShortName	String	32	否	AGV 所在地图的简称, 注: robotCount 填 -1 的话, mapShortName 必填,代表恢复该 地图中的全部 AGV
	robots	String[]	16	否	具体机器人编号列表
应答	code	String	6	是	返回码
	message	String	64	是	返回消息
	reqCode	String	64	是	请求编号
示例	请求	REST: POST http://IP:PORT/rcms/services/rest/hikRpcService/resumeRobot { "reqCode": "1541954B96B1112", "reqTime": "", "clientId": "", "tokenCode": "", "robotCount": "2", "mapShortName": "", "robots": [ "1001", "1002" ] }			
	应答	{ "code": "0", "message": "成功", "reqCode": "1541954B96B1112" }			

## 3.1.13 区域清空/释放

接口名	blockArea				
功能说明	封锁/解封指定区域				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
备注	区域在 RCS-2000 中配置；封锁后，区域内 AGV 停止；即将进入区域的 AGV 绕路行走				
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号。
	reqTime	String	20	否	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”。
	clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。
	tokenCode	String	64	否	令牌号，由调度系统颁发。
	matterArea	String	16	是	被封锁或解封的区域编号
	indBind	String	1	是	"1"：封锁，"0"：解封
	pause	String	1	否	完成清空区域后是否暂停该地图所有 AGV 0-不暂停（默认值） 1-暂停



	controlMod	String	2	否	调度模式：0-调度到区域外(默认) 1-调度到区域外暂住区 2-调度到指定区域，此时需要指定区域 -1-原先的区域锁定模式（即简单锁定区域，只封锁区域，车不能进入该区域）
	targetArea	String	16	否	指定区域的编码，目标区域 当 controlMod=2 时，需要传该值
	noticeThird	String	2	否	是否在封锁完成时上报第三方 0-不上报（默认） 1-上报
应答	code	String	6	是	返回码
	message	String	64	是	返回消息
	reqCode	String	64	是	请求编号
示例	请求	REST: POST http://IP:PORT/rcms/services/rest/hikRpcService/blockArea <pre>{   "reqCode": "1541954B96B1112",   "reqTime": "",   "clientCode": "",   "tokenCode": "",   "matterArea": "2",   "indBind": "1",   "pause": "0",   "controlMod": "2",   "targetArea": "HF002" }</pre>			
	应答	<pre>{   "code": "0",   "message": "成功", }</pre>			

		"reqCode": "1541954B96B1112"
--	--	------------------------------

### 3.1.14 预调度对外接口

接口名	genPreScheduleTask				
功能说明	上层系统通过预测任务到达的时间，提前下发预调度指令。由调度系统提前派车来到预调度点位，等待该点位的真实任务下发，以便直接匹配执行。				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号。 <b>由上层系统提供</b>
	reqTime	String	20	否	请求时间戳 格式：“yyyy-MM-dd HH:mm:ss”。 <b>由上层系统提供</b>
	clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。 <b>由 RCS-2000 告知上层系统</b>

	tokenCode	String	64	否	令 牌 号， 由 调 度 系 统 颁 发。 由 <b>RCS-2000 告知上层系统</b>
	positionCode	String	32	是	任务点位呼叫号
	nextTask	String	32	是	预调度时间 (s) 表示真实任务预计多久后生成，传-1 清空点位全部的预调度任务
	agvTyp	String	256	是	AGV 类型，预调度需要指定车型
	priority	String	4	否	优先级
	useableLayers	String	2	否	需求空仓位数 针对 ctu 使用 对接输送线时只需有一个仓位可用的 ctu 就可调度 对接梳齿式工作站时需要调度与该空 仓位数一致的全空 ctu
	cacheCount	String	2	否	缓存料箱数 针对 ctu 梳齿式工作站/输送线场景使 用，若需要系统计算预调度任务数量 时需传。系统根据缓存料箱数与需求

					空仓位数计算需要的预调度任务数。
	update	Number	2	否	是否更新，默认为 0  0：不更新之前的预调度任务，新增一个预调度任务  1：更新预调度任务，根据缓存料箱数与需求空仓位数计算需要的预调度任务数
应答	code	String			返回码
	message	String			返回消息
	reqCode	String			请求编号
	data	String			自定义返回
备注	接口使用示例见第四章 接口初步对接入参示例				
示例	请求	REST: POST <a href="http://IP:PORT/rcms/services/rest/hikRpcService/genPreScheduleTask">http://IP:PORT/rcms/services/rest/hikRpcService/genPreScheduleTask</a>  {  "reqCode": "1541954B96B1112",  "reqTime": "",  "clientCode": "",  "tokenCode": "",  "positionCode": "",  "nextTask": "",			

		<pre>"agvTyp": "",  "priority": "",  "useableLayers": "",  "cacheCount": "",  "update": ""  }</pre>
	应答	<pre>{  "code": "0" ,  "message": "成功",  "reqCode": "1541954B96B1112"  }</pre>

### 3.1.15 清空巷道

接口名	clearRoadWay				
功能说明	直接清空整条巷道的绑定数据，若有任务搬运过程中会进行锁定，无法清空。				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
请求参数	参数名	数据类型	最大长度	是否必填	备注

	reqCode	String	32	是	请求编号，每个请求都要一个唯一编号， 同一个请求重复提交， 使用同一编号。 <b>由上层系统提供</b>
	reqTime	String	20	是	请求时间戳 格式：“yyyy-MM-dd HH:mm:ss” 。 <b>由上层系统提供</b>
	clientCode	String	16	是	客户端编号，如 PDA，HCWMS 等。 <b>由 RCS-2000 告知上层系统</b>
	tokenCode	String	64	是	令 牌 号，由调度系统颁发。 <b>由 RCS-2000 告知上层系统</b>
	roadWayCode	String	16	是	巷道编号
应答	code	String			返回码
	reqCode	String			请求编号
	data	String			自定义返回
备注					
示例	请求	REST: POST <a href="http://IP:PORT/rcms/services/rest/hikRpcService/clearRoadWay">http://IP:PORT/rcms/services/rest/hikRpcService/clearRoadWay</a> {  "reqCode": "1541954B96B1112",  "reqTime": "",			

		<pre>"clientCode": "",  "tokenCode": "",  "roadWayCode": "",  }</pre>
	应答	<pre>{  "code": "0" ,  "message": "成功",  "reqCode": "1541954B96B1112"  }</pre>

### 3.1.16 料箱出库 TPS (CTU+分拨墙)

接口名	getOutPod				
功能说明	料箱出库，适用于分拨墙业务。料箱出库可以打断同料箱号的料箱回库任务。 列表中长度限制 1000。				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
请求参数	参数名	数据类型	最大长度	是否必填	备注

	reqCode		String	32	是	请求编号，每个请求都要一个唯一编号， 同一个请求重复提交， 使用同一编号。 <b>由上层系统提供</b>
	reqTime		String	20	是	请求时间戳 格式：“yyyy-MM-dd HH:mm:ss” 。 <b>由上层系统提供</b>
	clientCode		String	16	是	客户端编号，如 PDA，HCWMS 等。 <b>由 RCS-2000 告知上层系统</b>
	tokenCode		String	64	是	令牌号，由调度系统颁发。 <b>由 RCS-2000 告知上层系统</b>
	taskTyp		String	1	是	传 1 1-CTU 任务
	data	taskCode	String	64	是	任务单号
		ctnrCode	String	30	是	容器编号
		binCode	String	32	否	仓位编号
		wbCode	String	32	是	工作台编号



		agvTyp	String	16	否	机器人类型
		priority	String	3	否	任务优先级 1-127 127 最高, 1 最低
应答	code		String			返回码
	reqCode		String			请求编号
	data		String			自定义返回
备注	/rcms/services/rest/hikTpsService/getOutPod					
示例	请求	REST: POST <a href="http://IP:PORT/rcms/services/rest/hikTpsService/getOutPod">http://IP:PORT/rcms/services/rest/hikTpsService/getOutPod</a>  {  "reqCode": "123",  "reqTime": "",  "clientCode": "",  "tokenCode": "",  "taskTyp": "1",  "data": [{  "taskCode": "",  "ctnrCode": "",  "binCode": "",  "wbCode": "",				

		<pre>"agvTyp": "", "priority": ""  }{{ "taskCode": "", "ctnrCode": "", "binCode": "", "wbCode": "", "agvTyp": "", "priority": ""  }} }</pre>
	应答	<pre>{ "code": "0" , "message": "成功", "reqCode": "123" }</pre>

3.1.17 料箱回库 TPS (CTU+分拨墙)

接口名	returnPod
功能说明	料箱回库，对应料箱出库的返程，用于分拨墙任务。 料箱出库可以打断同料箱号的料箱回库任务。

接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号。 <b>由上层系统提供</b>
	reqTime	String	20	是	请求时间戳 格式：“yyyy-MM-dd HH:mm:ss”。 <b>由上层系统提供</b>
	clientCode	String	16	是	客户端编号，如 PDA，HCWMS 等。 <b>由 RCS-2000 告知上层系统</b>
	tokenCode	String	64	是	令牌号，由调度系统颁发。 <b>由 RCS-2000 告知上层系统</b>
	taskCode	String	64	是	任务单号，当前工作台任务单号，多个时可传一个（初始化入库新生成）
	agvTyp	String	16	否	机器人类型

	returnPodStrategy	String	32	否	回库策略编号(也可以是大区域@小区域的形式，多个组合，有优先级)
	taskTyp	String	2	是	传 5 5-料箱回库
	ctnrCode	String	30	是	料箱编号
	binCode	String	32	是	目标仓位编号
	srcBinCode	String	32	是	起点仓位编号
	wbCode	String	32	否	工作台编号（初始化入库必填）
	应答	code	String		
message		String			返回消息
reqCode		String			请求编号
data		String			自定义返回
备注	/rcms/services/rest/hikTpsService/returnPod				
示例	请求	REST: POST <a href="http://IP:PORT/rcms/services/rest/hikTpsService/returnPod">http://IP:PORT/rcms/services/rest/hikTpsService/returnPod</a>			

		<pre>{      "reqCode": "123",      "reqTime": "",      "clientCode": "",      "tokenCode": "",      "taskTyp": "5",      "taskCode": "",      "returnPodStrategy": ""      "ctnrCode": "",      "binCode": "",      "srcBinCode": ""      "wbCode": "",      "agvTyp": ""  }</pre>
	应答	<pre>{      "code": "0" ,      "message": "成功",      "reqCode": "123",      "data": ""  }</pre>

3.1.18 料箱顺序出库 (CTU)

接口名	genCtuGroupTaskBatch				
功能说明	<p>CTU 顺序出库任务场景使用，控制任务按照顺序下发执行。</p> <p>顺序出库不支持终点多工位与双深位。</p> <p>CTU 按组出库场景是调用 2.1.1 生成任务单接口，传递组号 (groupId)。</p>				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号。 <b>由上层系统提供</b>
	reqTime	String	20	否	请求时间戳 格式: "yyyy-MM-dd HH:mm:ss"。 <b>由上层系统提供</b>
	clientCode	String	16	否	客户端编号, 如 PDA, HCWMS 等。 <b>由 RCS-2000 告知上层系统</b>
	tokenCode	String	64	否	令牌号，由调度系统颁发。 <b>由 RCS-2000 告知上层系统</b>

	taskTyp		String	16	是	任务类型，默认用 B10
	seqTyp		Number	1	是	顺序出库类型：  1：表示组间组内都有序  2：表示组间有序，组内无序  3：表示组间无序，组内有序
	taskGroups	taskCode	String	64	是	任务单号
		ctnrCode	String	30	是	容器编号
		ctnrTyp	String	32	是	容器类型
		wbCode	String	32	是	工作台编号
		wbTyp	Number	1	是	工作台类型：  0：表示普通工作台  1：表示输送线工作台  2：表示梳齿式工作站  不同的终点工作台对调度系统执行的动作不一样，与输送线对接需要

						取放料箱，到梳齿式工作站 CTU 不需要动作
		groupId	Number		是	任务组编号或组间顺序
		sequence	Number		是	组内顺序
应答	code		String			返回码
	reqCode		String			请求编号
	data		String			自定义返回
备注						
示例	请求	REST: POST <a href="http://IP:PORT/rcms/services/rest/hikRpcService/genCtuGroupBatch">http://IP:PORT/rcms/services/rest/hikRpcService/genCtuGroupBatch</a> {  "reqCode": "123",  "reqTime": "",  "clientCode": "",  "tokenCode": "",  "taskTyp": "B10",  "seqTyp": ""  "taskGroups": [{				



		<pre>"taskCode": "",  "ctnrCode": "",  "ctnrTyp": ""  "wbCode": "",  "wbTyp": "",  "groupId": "",  "sequence": ""  }, {  "taskCode": "",  "ctnrCode": "",  "ctnrTyp": ""  "wbCode": "",  "wbTyp": "",  "groupId": "",  "sequence": ""  }  ]  }</pre>
	应答	<pre>{  "code": "0" ,  "message": "成功",</pre>

		"reqCode": "123"
	}	

### 3.1.19 料箱取放回调 (CTU)

接口名	boxApplyPass				
功能说明	任务模板支持配置取放料箱前进行申请，RCS-2000 向上层系统申请后，上层系统可调用此接口通知 RCS-2000 可以取放料箱。				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号。 <b>由上层系统提供</b>
	reqTime	String	20	是	请求时间戳 格式：“yyyy-MM-dd HH:mm:ss”。 <b>由上层系统提供</b>
	clientCode	String	16	是	客户端编号，如 PDA，HCWMS 等。 <b>由 RCS-2000 告知上层系统</b>

	tokenCode	String	64	是	令 牌 号，由调度系统颁发。由 <b>RCS-2000 告知上层系统</b>
	taskCode	String	64	是	任务单号，当前工作台任务单号，多个时可传一个（初始化入库新生成）
	type	String	2	是	1-取申请通过 2-放申请通过
应答	code	String			返回码
	message	String			返回消息
	reqCode	String			请求编号
	data	String			自定义返回
备注					
示例	请求	REST: POST <a href="http://IP:PORT/rcms/services/rest/hikRpcService/boxApplyPass">http://IP:PORT/rcms/services/rest/hikRpcService/boxApplyPass</a>  {  "reqCode": "123",  "reqTime": "",  "clientCode": "",  "tokenCode": "",  "taskCode": "",  "type": ""			

		}
	应答	{  "code": "0" ,  "message": "成功",  "reqCode": "123",  "data": ""  }

### 3.1.20 物料类型与位置绑定、解绑（移载机器人协议）

接口名	bindMaterialTypCodeAndBerth				
功能说明	物料类型和点位的绑定关系(移载机器人新协议用)				
接口协议	REST				
提供方	RCS-2000				
调用方	上层系统				
备注					
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号。
	reqTime	String	20	否	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”。
	clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。

	tokenCode	String	64	否	令牌号, 由调度系统颁发。
	materialType	String	32	是	物料类型编号, 绑定必填, 解绑可不填
	positionCode	String	32	是	位置编号, 地图位置的别名, 能任意命名(字母+数字), 但要唯一, 由RCS-2000 界面配置。
	ctnrCode	String	6	否	容器编号, 非必填, 绑定时填入则点位同时绑定容器编号, 解绑无需填入
	indBind	String	1	是	"1": 绑定, "0": 解绑
应答	code	String	6	是	返回码
	message	String	64	是	返回消息
	reqCode	String	64	是	请求编号
示例	请求	REST: POST URL: http://IP:PORT/rcms/services/rest/hikRpcService/bindMaterialTypCodeAndBerth { "reqCode": "12345678", "reqTime": "", "clientCode": "", "tokenCode": "", "materialType": "1", "positionCode": "p05", "ctnrCode": "111101", "indBind": "1" }			
	应答	{ "code": "0", "message": "成功",           }			

		"reqCode": "12345678"
--	--	-----------------------

## 3.2 上层平台提供的接口

### 3.2.1 告警推送通知

接口名	warnCallback				
功能说明	<p>告警推送回调的方法，调度系统将导致 AGV 停止运行的严重告警推送给上层系统。</p> <p>推送频率：10 秒一次</p> <p>该接口在调度系统的应用注册-业务通知中配置，启用表示通知、禁用表示不通知，通知对象为方法配置中勾选的应用，可选择最少一个、最多三个应用。</p>				
接口协议	REST				
提供方	上层系统				
调用方	RCS-2000				
备注	<p>该接口路径必须为：<a href="http://IP:PORT/service/rest/agvCallbackService/warnCallback">http://IP:PORT/service/rest/agvCallbackService/warnCallback</a></p> <p>其中 <a href="http://IP:PORT/service/rest">http://IP:PORT/service/rest</a> 路径在调度系统的系统配置-应用注册中完成</p>				
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号。
	reqTime	String	20	否	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”。
	clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。

	tokenCode		String	64	否	令牌号,由调度系统颁发。
	data	robotCode	String	5	是	车号
		beginDate	String	64	是	告警开始时间
		warnContent	String	64	是	告警内容
		taskCode	String	64	否	任务号
应答	code		String	6	是	返回码
	message		String	64	是	返回消息
	reqCode		String	64	是	请求编号
示例	请求	REST: POST http://IP:PORT/service/rest/agvCallbackService/warnCallback { "reqCode": "1541954B96B1112", "reqTime": "", "clientId": "", "tokenCode": "", "data": [{ "robotCode": "1001", "beginDate": "2020-04-02 23:12:12", "warnContent": "平台失联", "taskCode": "C002WWQQRR" }], { "robotCode": "1002", "beginDate": "2020-04-02 23:12:12", "warnContent": "导航告警", "taskCode": "C002WWQQRR33" } ] } }				
	应答	{ "code": "0", "message": "成功", "reqCode": "1541954B96B1112" } }				

### 3.2.2 绑定解绑通知

接口名	bindNotify				
功能说明	<p>可在执行绑定货架与储位,绑定货架与物料,绑定仓位与容器后通知上层。</p> <p>该接口在调度系统的应用注册-业务通知中配置, 启用表示通知、禁用表示不通知, 通知对象为方法配置中勾选的应用, 可选择最少一个、最多三个应用。</p>				
接口协议	REST				
提供方	上层平台				
调用方	RCS-2000				
备注	<p>该接口路径必须为: <u>http://IP:PORT/service/rest /bindNotify</u></p> <p>其中 <u>http://IP:PORT/service/rest</u> 路径在调度系统的系统参数中配置, 配置编号为: 10012, 10013, 10014</p> <p>根据系统参数 10026 配置哪种类型绑定解绑需要通知</p>				
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号, 每个请求都要一个唯一编号, 同一个请求重复提交, 使用同一编号
	reqTime	String	20	否	请求时间戳, 格式: "yyyy-MM-dd HH:mm:ss"
	clientCode	String	16	否	客户端编号, 如 PDA, HCWMS 等
	tokenCode	String	64	否	令牌号, 由调度系统颁发。
	method	String	16	是	方法名 bindPodAndBerth : 货架与储位



						绑定解绑 bindPodAndMat：货架与物料绑定解绑 bindCtnrAndBin：仓位与容器绑定解绑
	indBind		String	2	是	1--绑定 0--解绑
	bindParam	List		2000	是	对象封装为 list 对象，包含下列字段
		podCode	String	32	否	货架编号 适用： bindPodAndBerth、 bindPodAndMat
		berthCode	String	32	否	储位编号 适用： bindPodAndBerth
		materialLot	String	32	否	物料批次 适用： bindPodAndMat
		ctnrCode	String	30	否	仓位编号 适用： bindCtnrAndBin
		ctnrType	String	32	否	仓位编号 适用： bindCtnrAndBin
		stgBinCode	String	32	否	仓位编号 适用：

						bindCtnrAndBin
应答	code		String		返回码	
	message		String		返回消息	
	reqCode		String		请求编号	
	data		String		自定义返回	
示例	请求	{ "bindParam": [{ "berthCode": "AB1", "ctnrCode": "", "ctnrType": "", "materialLot": "", "podCode": "111111", "stgBinCode": "" }], "indBind": "0", "method": "bindPodAndBerth", "reqCode": "1541954B96B1112", "reqTime": "2020-09-17 13:48:58" }				
	应答	{ "code": "0", "message": "成功", "reqCode": "1541954B96B1112" }				

### 3.2.3 校验料箱是否可入库 (CTU)

接口名	applyReturnForValid
功能说明	<p>梳齿工作站入库场景使用，在料箱流进梳齿工作站前，向上层系统询问料箱是否可入库。</p> <p>若可以入库，上层返回成功，料箱即流进梳齿工作站；若不能入库，返回错误码，此种情况一般是料箱有问题，料箱不流入梳齿工作站，需要人工处理该料箱。</p> <p>该接口仅是询问能否进入梳齿工作站，料箱的回库仓位通过 3.2.4 申请回库仓位 (CTU) 接口</p>

	询问上层系统				
接口协议	REST				
提供方	上层平台				
调用方	RCS-2000				
备注					
请求参数	参数名	数据类型	最大长度	是否必填	备注
	reqCode	String	32	是	请求编号，每个请求都要一个唯一编号，同一个请求重复提交，使用同一编号
	reqTime	String	20	否	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”
	wbCode	String	32	是	工作台/工作站呼叫号
	ctnrCode	String	30	是	料箱号
应答	code	String			返回码
	message	String			返回消息
	reqCode	String			请求编号
示例	请求	{ "reqCode": "1541954B96B1112", "reqTime": "", "wbCode": "", "ctnrCode": "" }			
	应答	{ "code": "0", "message": "成功", "reqCode": "1541954B96B1112" }			

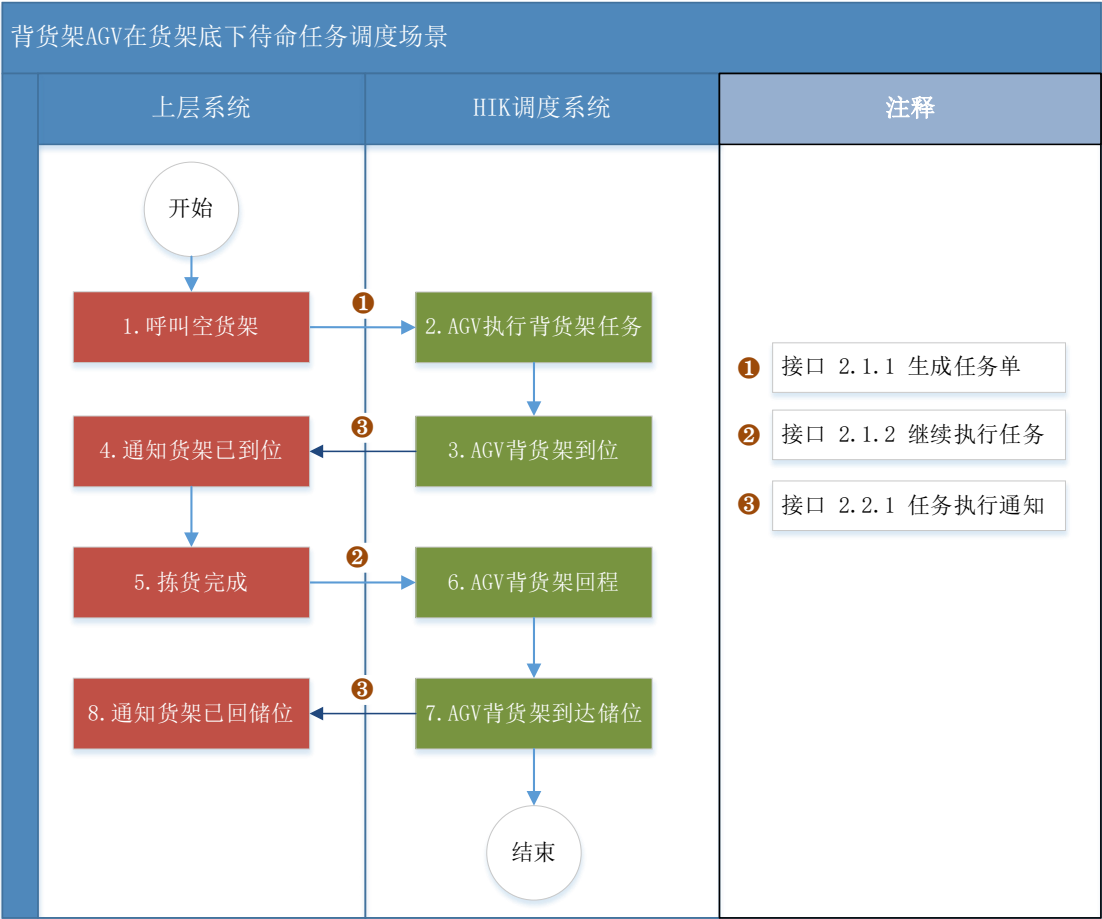
### 3.2.4 申请回库仓位 (CTU)

接口名	applyReturnForBin					
功能说明	梳齿工作站和输送线入库场景使用，向上层系统申请料箱的入库位置  在 B05、B06、B08 任务模板中配置通知（申请回库仓位）与第三方的对接路径  返回数据需按照接口说明中的字段格式返回					
接口协议	REST					
提供方	上层平台					
调用方	RCS-2000					
备注						
请求参数	参数名		数据类型	最大长度	是否必填	备注
	reqCode		String	32	是	请求编号，每个请求都要一个唯一编号， 同一个请求重复提交， 使用同一编号
	reqTime		String	20	否	请求时间戳, 格式: “yyyy-MM-dd HH:mm:ss”
	data	ctnrCode	String	64	是	容器编号
	wbCode		String	32	是	工作台/工作站呼叫号
	groupCode		String	30	是	组号
应答	code		String		返回码	
	message		String		返回消息	
	reqCode		String		请求编号	

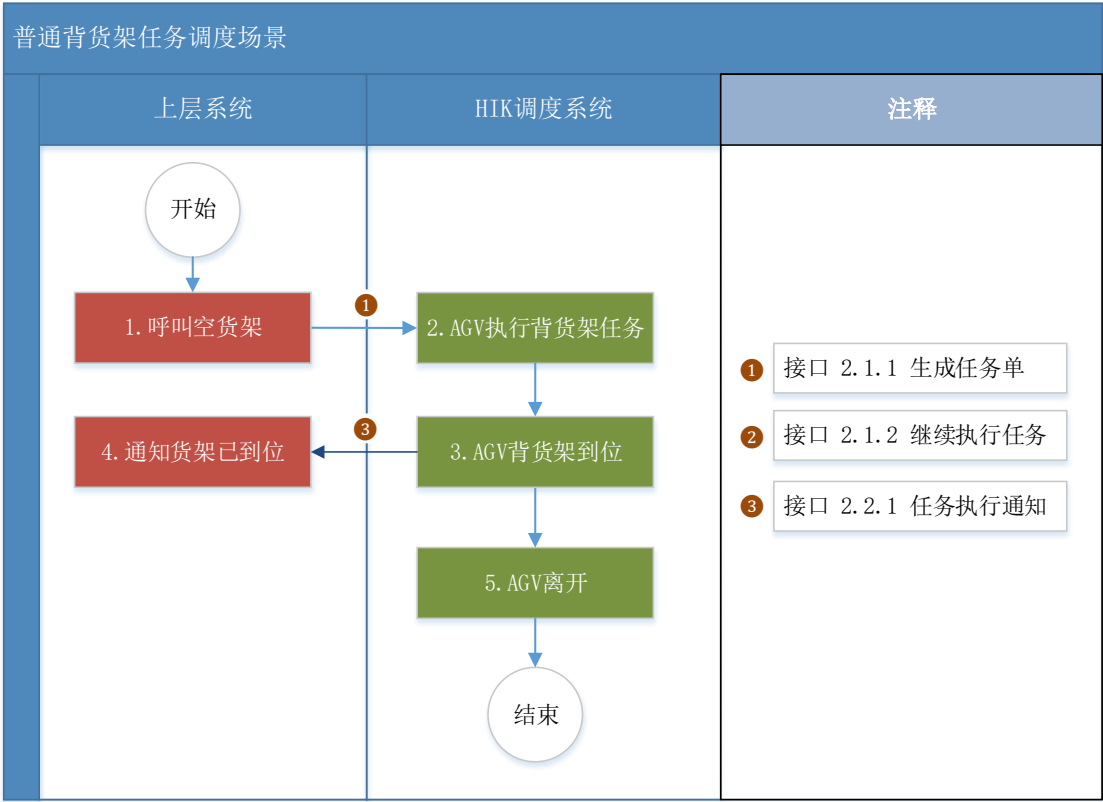
	data	ctnrCode	String	容器编号
		ctnrType	String	容器类型
		binCode	String	仓位编号
示例	请求	{ "reqCode": "1541954B96B1112", "reqTime": "", "data": [{ "ctnrCode": "" }], "wbCode": "", "groupCode": "" }		
	应答	{ "code": "0", "message": "成功", "reqCode": "1541954B96B1112" "data": [{ "ctnrCode": "", "ctnrType": "", "binCode": "" }] }		

4 典型调度场景

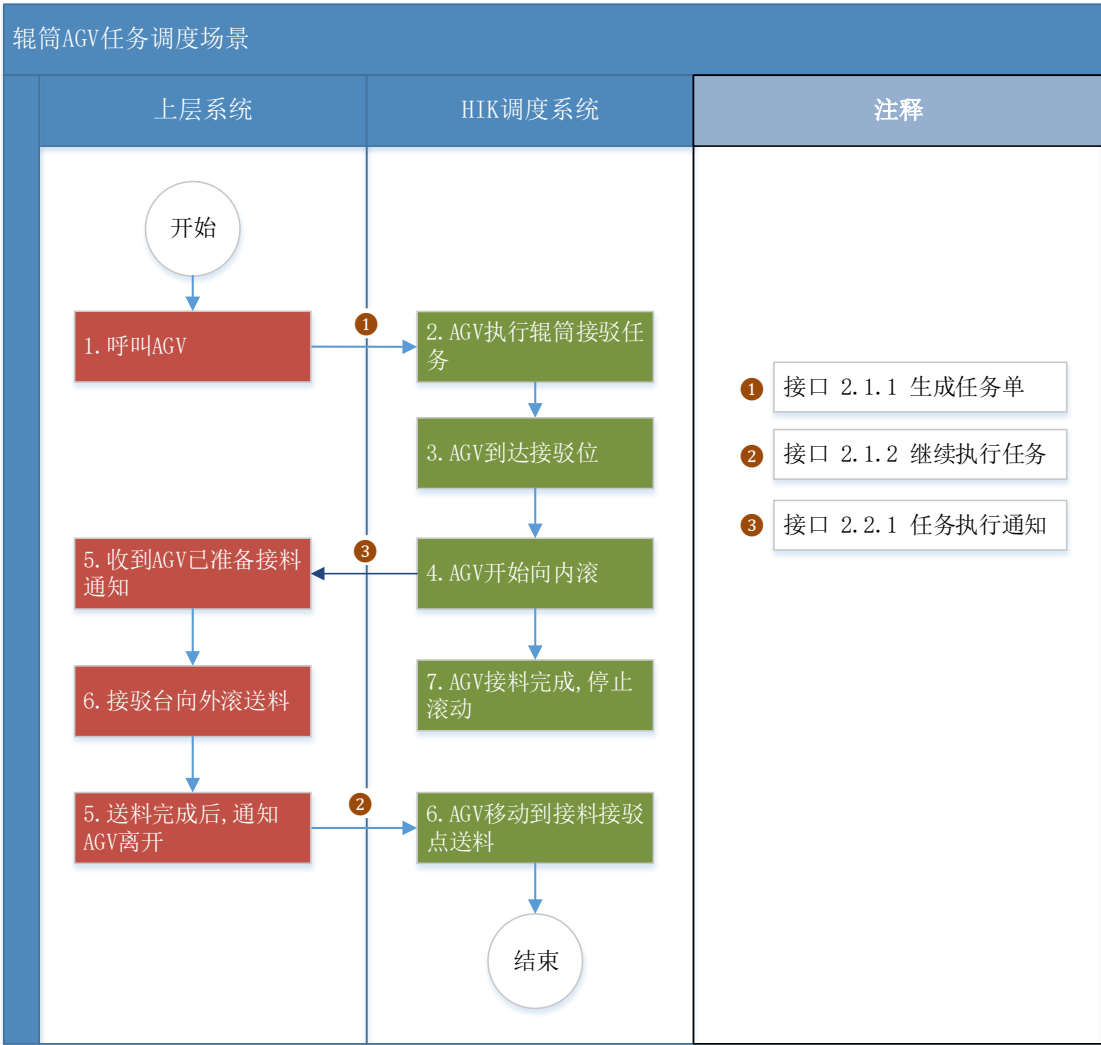
场景一：背货架 AGV 在货架底下待命任务调度场景



场景二：普通背货架任务调度场景

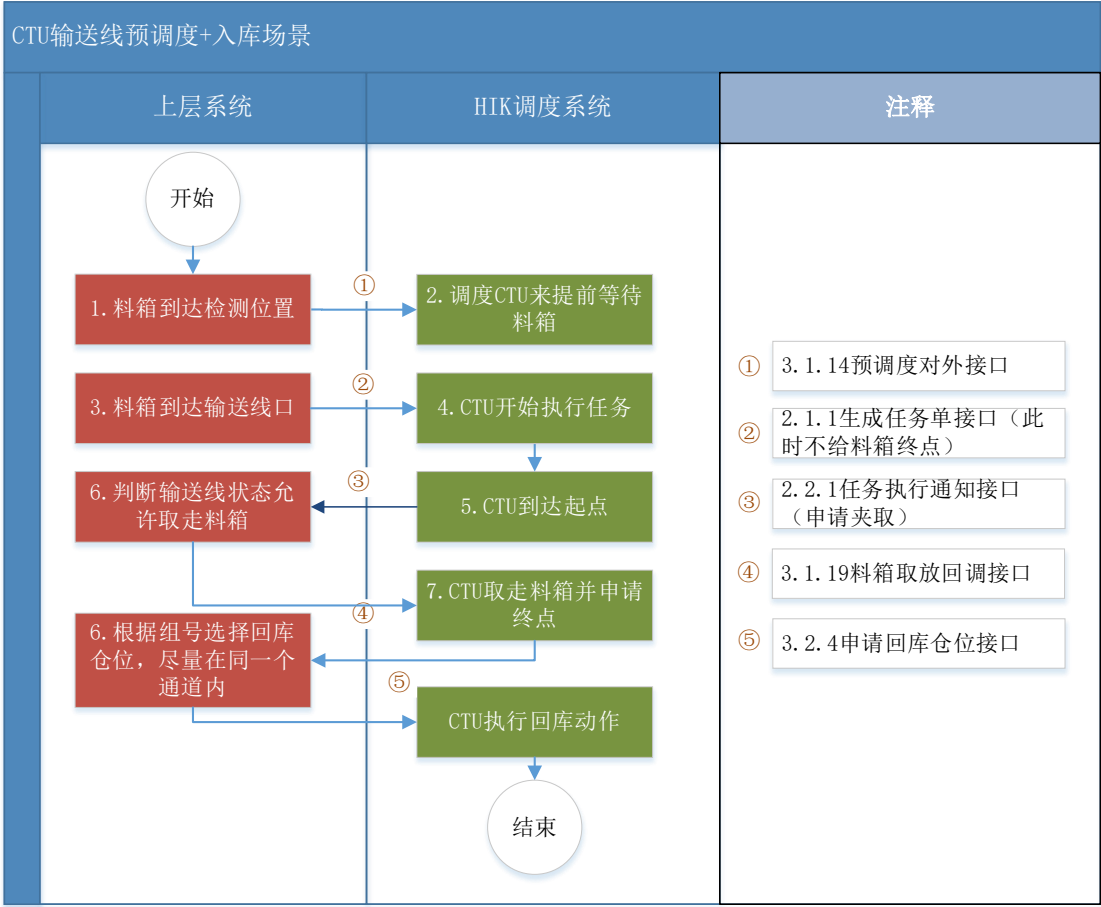


场景三：辊筒 AGV 任务调度场景

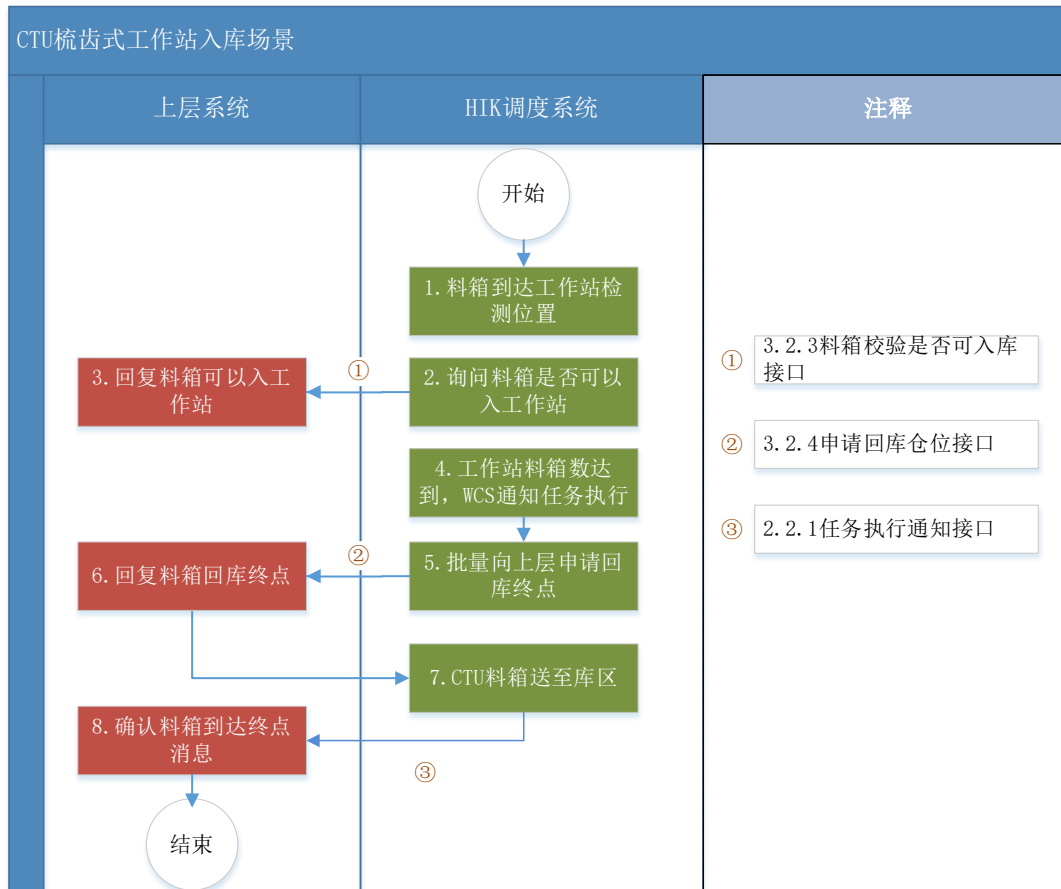




场景四：CTU 输送线预调度+入库



## 场景五：CTU 梳齿式工作站入库



## 其他 CTU 场景接口使用

- 1.分拨墙出入库场景：3.1.16 料箱出库 TPS、3.1.17 料箱回库 TPS
- 2.出库至输送线、梳齿式工作站（单个或按组出库）：2.1.1 生成任务单
- 3.出库至输送线、梳齿式工作站（按顺序出库）：3.1.18 料箱顺序出库
- 4.输送线入库：2.1.1 生成任务单、3.2.4 申请回库仓位（CTU）
- 5.梳齿式工作站入库：3.2.3 校验料箱是否可入库（CTU）、3.2.4 申请回库仓位（CTU）
- 6.料箱取放申请、任务完成通知：2.2.1 任务执行通知

7.上层允许 CTU 取放料箱：3.1.19 料箱取放回调

## 5 接口初步对接入参示例

### 2.1.1 生成任务单接口

url:

http://IP:PORT/rcms/services/rest/hikRpcService/genAgvSchedulingTask

入参:

```
{
  "reqCode": "468513 每次传入必须唯一",
  "taskTyp": "F01",
  "wbCode": "工作位",
  "positionCodePath": [
    {
      "positionCode": "起点位置",
      "type": "00"
    },
    {
      "positionCode": "终点位置",
      "type": "00"
    }
  ],
  "podCode": "货架编号, 默认填-1"
}
```

### 2.1.2 继续执行任务接口

url: http://IP:PORT/rcms/services/rest/hikRpcService/continueTask

入参:

```
{  
  "reqCode": "1231233 每次传入必须唯一",  
  "taskCode": "任务单编号"  
}
```

### 2.2.1 任务执行通知接口

url: 由上层平台提供, 结尾路径建议采用/agvCallbackService/agvCallback

路径示例如下:

http://IP:PORT/xxx/agv/agvCallbackService/agvCallback

入参:

```
任务完成通知参数  
{  
  "reqCode": "",  
  "method": "end",  
  "taskCode": "任务单号",  
  "wbCode": "工作位",  
  "podCode": "货架编号"  
}
```

### 3.1.14 预调度对外接口

url:

http://IP:PORT/rcms/services/rest/hikRpcService/genPreScheduleTask

## 入参:

CTU 输送线料箱入库（每扫到一个料箱下发一个预调度任务）

```
{
  "reqCode": "123",
  "positionCode": "pos1", //预调度的位置或仓位
  "nextTask": "300", //预计真实任务 5 分钟内会下发，若达到时间未下发任务，预调度的车释放
  "agvTyp": "1",
  "priority": "",
  "useableLayers": "1", //单次预调度对应一个料箱，需要一个 CTU 仓位
  "cacheCount": "1", //单次预调度对应一个料箱
  "update": "0" //不更新原有的预调度任务，累加一个新的预调度任务
}
```

CTU 输送线料箱入库（每扫到一个料箱更新一次需要的预调度任务总数）

```
{
  "reqCode": "123",
  "positionCode": "pos1", //预调度的位置或仓位
  "nextTask": "300", //预计真实任务 5 分钟内会下发，若达到时间未下发任务，预调度的车释放
  "agvTyp": "1",
  "priority": "",
  "useableLayers": "1", //单次预调度对应一个料箱，需要一个 CTU 仓位
  "cacheCount": "4", //输送线上当前有 4 个料箱，需要 4 个预调度任务
  "update": "1", //更新预调度任务数，代表需要系统根据缓存料箱数和每个 CTU 对接仓位数，计算需要的预调度任务数。预调度任务数= cacheCount/ useableLayers
}
```

CTU 梳齿工作站料箱入库

```
{
  "reqCode": "123",
  "positionCode": "pos1", //预调度的位置或仓位
  "nextTask": "300", //预计真实任务 5 分钟内会下发，若达到时间未下发任务，预调度的车释放
  "agvTyp": "1",
  "priority": "",
  "useableLayers": "6", //梳齿式工作站的对接仓位数，需要预调度全空的 CTU
  "cacheCount": "12", //梳齿式工作站内缓存着 12 个料箱需要入库
  "update": "1", //更新预调度任务数，代表需要系统根据缓存料箱数和每个 CTU 对接仓位数，计算需要的预调度任务数。预调度任务数= cacheCount/ useableLayers
}
```

## 6 附件

### 6.1 AGV 常见状态列表

状态编号 status	状态描述(中文) statusStr	状态描述(英文) StatusStr
1	任务完成	Task completed
2	任务执行中	Executing task
3	任务异常	Abnormal task
4	任务空闲	Idle task
5	机器人暂停	Robot stopped
6	举升货架状态	Lifting shelf status
7	充电状态	Charging status
8	弧线行走中	Battery arcing in progress
9	充满维护	Fully charged, entering maintenance mode
11	背货未识别	Carried item not recognized
12	货架偏角过大	Excessive shelf angle divergence
13	运动库异常	Motion library exception
14	货码无法识别	Unable to recognise product code
15	货码不匹配	Product code mismatch
16	举升异常	Lift abnormal

17	充电桩异常	Charging post abnormal
18	电量无增加	No increase in current
20	充电指令角度错误	Angle error in charging directive
21	平台下发指令错误	Platform decentralisation directive error
23	外力下放	External force, unloading
24	货架位置偏移	Misaligned shelf
25	小车不在锁定区	Trolley not in designated zone
26	下放重试失败	Decentralisation failed
27	货架摆歪	Uneven shelf
28	举升电池电量太低	Lift battery current too low
29	后退角度偏大	Wide reversing angle
30	未背货架举升	No rack detected
31	区域锁定失败	Failed to lock zone
33	旋转申请暂时失败	Rotation request temporarily failed
34	地图切换点地码未识别	Unable to recognise coordinates to switch maps

6.2 常用地图元素类型列表

类型 dataTyp	类型描述(中文)
1	仓库储位

10	工作区
11	充电区
20	产线缓冲区
55	巷道存储区

## 6.3 接口调用 DEMO

如果上层系统是 C#语言或 JAVA 开发, 采用 REST 协议对接, 通过以下 DEMO 可以快速上手, DEMO 可向产品经理获取.

## 7 更新说明

1. 修改 CTU 梳齿式工作站/输送线入库场景的交互接口, 通过 3.2.4 申请回库仓位 (CTU) 向上层系统进行申请。
2. 生成任务单接口补充机器人类型字段、区域/策略中灵活挑选货架/物料的参数, 增加容器数量字段。
3. V3.3 生成任务单接口、任务执行通知接口增加移载机器人协议字段 materialType, 新增接口物料类型与位置绑定、解绑