

文档类别

杭 州 海 康 机 器 人 技 术 有 限 公 司

文档编号

厂内物流调度控制系统 RCS-2000 V2.5
对外任务接口文档

版权声明

本文档由海康机器人公司开发，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：0571-88075998

技术支持电话：400 700 5998

传真：0571-88805843

地址：中国杭州市滨江区东流路700号

邮编：310051

公司E-mail: market@hikvision.com

公司网站: <http://www.hikvision.com>

目录

1	协议概述.....	4
2	常用接口*.....	6
2.1	调度系统提供的接口.....	6
2.1.1	生成任务单*.....	6
2.1.2	继续执行任务*.....	9
2.1.3	取消任务*.....	12
2.2	上层平台提供的接口.....	14
2.2.1	任务执行通知*.....	14
3	可选接口.....	16
3.1	调度系统提供的接口.....	16
3.1.1	任务优先级设置.....	16
3.1.2	货架与位置绑定、解绑.....	18
3.1.3	货架与物料绑定、解绑.....	20
3.1.4	位置禁用与启用.....	21
3.1.5	地图位置信息同步.....	23
3.1.6	查询货架储位与物料批次关系.....	25
3.1.7	查询任务状态.....	28
3.2	上层平台提供的接口.....	30
3.2.1	数据同步通知.....	30
4	典型调度场景.....	32
5	接口初步对接入参示例.....	35
6	附件.....	36
6.1	调用 DEMO.....	36
7	更新记录.....	36

1 协议概述

REST 协议和 TCP 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。

对接参数基准定义：

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。
- 为解决 TCP 的粘包与拆包问题，结束符统一使用 \$\$

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。由 RCS-2000 告知上层系统。
tokenCode	选填	令牌号，由调度系统颁发。由 RCS-2000 告知上层系统。

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
“99”	其他未知错误

RCS-2000 基础访问地址

REST 协议：

baseURL	http://IP:PORT/cms/services/rest/hikRpcService
端口	与 WEB 端口一致，默认端口：80

TCP 协议：

IP	RCS-2000 的 IP 地址
端口	默认 TCP 端口：6892

上层系统基础访问地址

REST 协议

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

TCP 协议：

IP	上层系统 IP 地址
端口	上层系统 TCP 端口

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。				
接口协议	REST 或 TCP				
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。 由上层系统提供
	reqTime	String	20	否	请求时间戳 格式：“yyyy-MM-dd HH:mm:ss”。 由上层系统提供
	clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。 由 RCS-2000 告知上层系统
	tokenCode	String	64	否	令牌号，由调度系统颁发。 由 RCS-2000 告知上层系统
	interfaceName	String	64		genAgvSchedulingTask TCP 协议必传，REST 协议不用传，传了也不影响
	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
	wbCode	String	32	否	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, 位置名称为字母\数字\或组合, 不超过 32 位。 可以不填
	positionCodePath	Object[]	列表 长小于 50	否	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 positionCode: 位置编号 位置类型说明: 00 表示: 位置编号 01 表示: 物料批次号 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 03 表示: 货架编号, 通过货架编号找到货架所在位置 04 表示: 区域编号, 在区域中查找可用位置
	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
	podTyp	String	16	否	货架类型，找满货架时传空，找空货架时必传 -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 positionCode 找到位置上的货架和 materialLot 来绑定）
	priority	String	32	否	优先级，从（1~5）级，最大优先级最高
	taskCode	String	32	否	任务单号,选填，不填系统自动生成，必须为 32 位 UUID
	agvCode	String	16	否	AGV 编号，填写表示指定某一编号的 AGV 执行该任务
	data	String	2000	否	自定义字段，不超过 2000 个字符
应答	code	String			返回码
	data	String			自定义返回（返回任务单号）
	message	String			返回消息
	reqCode	String			请求编号
备注					

示例	请求	<p>REST: POST http://IP:PORT/cms/services/rest/hikRpcService/genAgvSchedulingTask</p> <p>TCP: 通过 TCP 端口传输以下示例字符串, 需要保留末尾\$\$, 做为分隔符使用.</p> <p>REST 需要去掉末尾\$\$</p> <pre>{ "reqCode": "468513", "reqTime": "", "clientCode": "", "tokenCode": "", "interfaceName": "genAgvSchedulingTask", "taskTyp": "F01", "wbCode": "", "positionCodePath": [{ "positionCode": "p01", "type": "00" }, { "positionCode": "x02", "type": "02" }], "podCode": "100001", "podDir": "0", "priority": "1", "agvCode": "", "taskCode": "", "data": "" }\$\$</pre>
	应答	<pre>{ "code": "0", "data": "F01169C808C317111G", "message": "成功", "reqCode": "468513" }\$\$</pre>

说明:

`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`。

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 或 TCP				
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。 由上层系统提供
	reqTime	String	20	否	请求时间截 格式：“yyyy-MM-dd HH:mm:ss”。 由上层系统提供
	clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。 由 RCS-2000 告知上层系统
	tokenCode	String	64	否	令牌号，由调度系统颁发。 由 RCS-2000 告知上层系统
	interfaceName	String	64		continueTask TCP 协议必传，REST 协议不用传，传了也不影响
	wbCode	String	32	否	工作位，与 RCS-2000 端配置的位置名称一致。
	podCode	String	6	否	货架号，采用货架号触发的方式。
	agvCode	String	16	否	AGV 编号，采用 AGV 编号触发的方式。

	taskCode	String	32	否	任务单号,选填, 不填系统自动生成, 必须为 32 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。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>对象类型定义:</p> <p>00, 代表 nextPositionCode 是一个位置</p> <p>04: 代表 nextPositionCode 是一个区域</p>
	data	String	2000	否	自定义字段, 不超过 2000 个字符
	应答	code	String		
data		String			自定义返回
message		String			返回消息
reqCode		String			请求编号
备注					
示例	请求	REST: POST http://IP:PORT/cms/services/rest/hikRpcService/continueTask TCP: 通过 TCP 端口传输以下示例字符串, 需要保留末尾\$\$, 做为分隔符使用。 REST 需要去掉末尾\$\$ {			

		<pre>"reqCode": "123", "reqTime": "", "clientCode": "", "tokenCode": "", "interfaceName": "continueTask", "wbCode": "", "podCode": "", "agvCode": "", "taskCode": "123456", "taskSeq": "", "nextPositionCode": {"positionCode": "p02", "type": "00"}, "data": "" }\$\$</pre>
	应答	<pre>{ "code": "0", "data": "", "message": "成功", "reqCode": "123" }\$\$</pre>

说明：

`wbCode`、`agvCode`、`taskCode` 和 `podCode` 四个必须填一个，优先级从高到低依次为：`wbCode`、`taskCode`、`agvCode`、`podCode`，都传了优先使用 `wbCode`，以确定任务单编号。待现场地图部署、配置完成后可获取。

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，会把货架直接放在路上，任务取消后，AGV 为空闲状态。				
接口协议	REST 或 TCP				
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。由上层系统提供

	reqTime	String	20	否	请求时间截 格式：“yyyy-MM-dd HH:mm:ss”。由上层系统提供
	clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。 由 RCS-2000 告知上层系统
	tokenCode	String	64	否	令牌号，由调度系统颁发。由 RCS-2000 告知上层系统
	interfaceName	String	64		cancelTask TCP 协议必传，REST 协议不用传， 传了也不影响
	agvCode	String	16	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			返回码
	data	String			自定义返回
	message	String			返回消息
	reqCode	String			请求编号
备注	taskCode 与 agvCode 其中一个必填				
示例	请求	REST: POST http://IP:PORT/cms/services/rest/hikRpcService/cancelTask TCP: 通过 TCP 端口传输以下示例字符串，需要保留末尾\$\$，做为分隔符使用。 REST 需要去掉末尾\$\$ { "reqCode": "1541954B96B1112", "reqTime": "", "clientCode": "", "tokenCode": "", "interfaceName": "cancelTask", "agvCode": "", "taskCode": "123456" }\$\$			
	应答	{ "code": "0", "data": "",			

		<pre>"message": "成功", "reqCode": "1541954B96B1112" }\$\$</pre>
--	--	--

说明：

`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。取消任务单后可释放对应的 AGV。

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				
接口协议	REST 或 TCP				
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 由上层系统提供
	reqTime	String	20	是	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss” 由上层系统提供
	interfaceName	String	64		agvCallback TCP 协议必传，REST 协议不用传，传了也不影响
	cooX	String	8	否	地码 X 坐标(mm)

	cooY	String	8	否	地码 Y 坐标(mm)
	currentPositionCode	String	32	否	当前位置编号
	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，唯一标识
	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
	podCode	String	16	否	货架编号
	podDir	String	4	是	"180","0","90","-90" 分别对应地图的"左","右","上","下"
	robotCode	String	16	是	AGV 编号（同 agvCode）
	taskCode	String	32	是	当前任务单号
	wbCode	String	32	是	工作位，与 RCS-2000 端配置的位置名称一致
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	REST: POST http://IP:PORT/xxx/agv/agvCallbackService/agvCallback TCP: 通过 TCP 端口传输以下示例字符串，需要保留末尾\$\$，做为分隔符使用。			

		<p>REST 需要去掉末尾\$\$</p> <pre>{ "cooX":3000, "cooY":21999, "currentPositionCode":"p02", "data": "", "mapCode": "AA", "mapDataCode": "002069AA015172", "method": "end", "podCode": "100001", "podDir": "", "reqCode": "1541954B96B1112", "reqTime": "2019-04-03 10:08:06", "robotCode": "6001", "taskCode": "test169E0F39740116Q", "wbCode": "p02" "" }</pre>
	应答	<pre>{ "code": "0", "message": "成功", "reqCode": "1541954B96B1112", "data": "" }</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(1~5 级), 值越大, 优先级越高。
接口协议	REST 或 TCP
提供方	RCS-2000
调用方	上层系统

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。 由上层系统提供
	reqTime		String	20	否	请求时间截 格式:“yyyy-MM-dd HH:mm:ss”。 由上层系统提供
	clientCode		String	16	否	客户端编号,如 PDA,HCWMS 等。 由 RCS-2000 告知上层系统
	tokenCode		String	64	否	令牌号,由调度系统颁发。 由 RCS-2000 告知上层系统
	interfaceName		String	64		setTaskPriority TCP 协议必传,REST 协议不用传,传了也不影响
	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(1~5)级,最大优先级最高
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	
备注						
示例	请求		REST: POST http://IP:PORT/cms/services/rest/hikRpcService/setTaskPriority TCP: 通过 TCP 端口传输以下示例字符串,需要保留末尾\$\$,做为分隔符使用。 REST 需要去掉末尾\$\$ { "reqCode": "1234567", "reqTime": "", "clientCode": "",			

		<pre>"tokenCode":""," "interfaceName":"setTaskPriority", "priorities": [{ "priority": "1", "taskCode": "1232" }, { "priority": "2", "taskCode": "3214" }] }\$\$</pre>
	应答	<pre>{ "code": "0", "data": "", "message": "成功", "reqCode": "1234567" }\$\$</pre>

说明：

设置优先级仅适用于还未派 AGV 执行的任务，设置优先级后，可根据优先级分配 AGV 执行。若任务已分配 AGV 开始执行，则设置任务优先级无效。

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，系统可以通过货架找到对应位置。				
接口协议	REST 或 TCP				
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。 由上层系统提供
	reqTime	String	20	否	请求时间戳，格式：“yyyy-MM-dd

					HH:mm:ss”。由上层系统提供
	clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。 由 RCS-2000 告知上层系统
	tokenCode	String	64	否	令牌号，由调度系统颁发。由 RCS-2000 告知上层系统
	interfaceName	String	64		bindPodAndBerth TCP 协议必传，REST 协议不用传， 传了也不影响
	podCode	String	6	是	货架编号
	positionCode	String	32	是	位置编号
	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 http://IP:PORT/cms/services/rest/hikRpcService/bindPodAndBerth TCP: 通过 TCP 端口传输以下示例字符串，需要保留末尾\$\$，做为分隔符使用。 REST 需要去掉末尾\$\$ <pre>{ "reqCode": "12345678", "reqTime": "", "clientCode": "", "tokenCode": "", "interfaceName": "bindPodAndBerth", "podCode": "100001", "positionCode": "p05", "indBind": "1" }</pre> \$\$\$			
	应答	<pre>{ "code": "0", "data": "", </pre>			

		<pre> "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，系统可以通过物料或批次找到对应货架。				
接口协议	REST 或 TCP				
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。 由上层系统提供
	reqTime	String	20	否	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”。 由上层系统提供
	clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。 由 RCS-2000 告知上层系统
	tokenCode	String	64	否	令牌号，由调度系统颁发。 由 RCS-2000 告知上层系统
	interfaceName	String	64		bindPodAndMat TCP 协议必传，REST 协议不用传，传了也不影响
	podCode	String	16	是	货架编号
	materialLot	String	32	否	物料批次
	indBind	String	1	是	"1"：绑定，"0"：解绑

					解绑时，物料批次可以为空
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	<p>REST: POST http://IP:PORT/cms/services/rest/hikRpcService/bindPodAndMat</p> <p>TCP: 通过 TCP 端口传输以下示例字符串, 需要保留末尾\$\$, 做为分隔符使用.</p> <p>REST 需要去掉末尾\$\$</p> <pre>{ "reqCode": "1541954B96B1112", "reqTime": "", "clientId": "", "tokenCode": "", "interfaceName": "bindPodAndMat", "podCode": "100001", "materialLot": "123", "indBind": "1" }</pre> }}\$			
	应答	<pre>{ "code": "0", "data": "", "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 或 TCP
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。 由上层系统提供
	reqTime	String	20	否	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”。 由上层系统提供
	clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。 由 RCS-2000 告知上层系统
	tokenCode	String	64	否	令牌号，由调度系统颁发。 由 RCS-2000 告知上层系统
	interfaceName	String	64		lockPosition TCP 协议必传，REST 协议不用传，传了也不影响
	positionCode	String	32	是	位置编号
	indBind	String	1	是	"1"：启用， "0"：禁用
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	<p>REST: POST http://IP:PORT/cms/services/rest/hikRpcService/lockPosition</p> <p>TCP: 通过 TCP 端口传输以下示例字符串，需要保留末尾\$\$，做为分隔符使用。</p> <p>REST 需要去掉末尾\$\$</p> <pre>{ "reqCode": "1541954B96B1112", "reqTime": "", "clientCode": "", "tokenCode": "", "interfaceName": "lockPosition",</pre>			

		<pre>"positionCode": "p02", "indBind": "1" }\$\$</pre>
	应答	<pre>{ "code": "0", "data": "", "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 或 TCP				
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。由上层系统提供
	reqTime	String	20	否	请求时间戳 格式：“yyyy-MM-dd HH:mm:ss”。由上层系统提供
	clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。由 RCS-2000 告知上层系统
	tokenCode	String	64	否	令牌号,由调度系统颁发。由 RCS-2000 告知上层系统
	interfaceName	String	64		syncMapDatas TCP 协议必传，REST 协议不用传，传了也不影响

	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			返回码
	data [列表]	cooX	String	8	是	地码 X 坐标(mm)
		cooY	String	8	是	地码 Y 坐标(mm)
		dataTyp	String	2	是	地图元素类型，11-充电桩，10-工作台，1-储位，20-缓冲区
		direction	String	8	否	工作台方向 “180”、“0”、“90”、“-90” 分别代表“左”、“右”、“上”、“下” 工作台方向即人在面向货架拣货时的方向
		mapCode	String	16	是	地图编号
		mapDataCode	String	32	是	地码编号，唯一标识
		userCallCode	String	32	是	呼叫站点，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。
		berthType	String	2	否	储位类型，1-外层储位，2-内层储位，3-普通储位
	message		String			返回消息
	reqCode		String			请求编号
备注						
示例	请求	REST: POST http://IP:PORT/cms/services/rest/hikRpcService/syncMapDatas				

		<p>TCP: 通过 TCP 端口传输以下示例字符串, 需要保留末尾\$\$, 做为分隔符使用. REST 需要去掉末尾\$\$</p> <pre>{ "reqCode": "1541954B96B1112", "reqTime": "", "clientCode": "", "tokenCode": "", "interfaceName": "syncMapDatas", "mapDataCode": "xxxxxx", "mapShortName": "xxxxxx", "dataTyp": "" }\$\$</pre>
	应答	<pre>{ "code": "0", "data": [{ "berthType": "3", "cooX": "17000.0", "cooY": "18000.0", "dataTyp": "1", "direction": "0", "mapCode": "AA", "mapDataCode": "011724AA012414", "userCallCode": "011724AA012414" }, { "cooX": "11000.0", "cooY": "21999.0", "dataTyp": "10", "direction": "0", "mapCode": "AA", "mapDataCode": "007586AA015172", "userCallCode": "104" }, ], "message": "成功", "reqCode": "1541954B96B1112" }\$\$</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 或 TCP				
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。 由上层系统提供
	reqTime	String	20	否	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”。 由上层系统提供
	clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。 由 RCS-2000 告知上层系统
	tokenCode	String	64	否	令牌号，由调度系统颁发。 由 RCS-2000 告知上层系统
	interfaceName	String	64		queryPodBerthAndMat TCP 协议必传，REST 协议不用传，传了也不影响
	podCode	String	16	否	货架编号
	materialLot	String	32	否	物料批次
	positionCode	String	16	否	位置编号
	areaCode	String	16	否	区域编号
	mapShortName	String	16	是	地图简称
应答	code		String		返回码
	data	areaCode	String		区域编号

		materialLot	String	物料批次
		podCode	String	货架编号
		posCode	String	位置呼叫号
		positionCode	String	位置编号
	message		String	返回消息
	reqCode		String	请求编号
示例	请求	<p>REST: POST http://IP:PORT/cms/services/rest/hikRpcService/queryPodBerthAndMat</p> <p>TCP: 通过 TCP 端口传输以下示例字符串, 需要保留末尾\$\$, 做为分隔符使用.</p> <p>REST 需要去掉末尾\$\$</p> <pre>{ "reqCode": "1541954B96B1110", "reqTime": "", "clientCode": "", "tokenCode": "", "interfaceName": "queryPodBerthAndMat", "podCode": "", "materialLot": "", "positionCode": "", "areaCode": "", "mapShortName": "test" }\$\$</pre>		
	应答	<pre>{ "code": "0", "data": [{ "areaCode": "", "materialLot": "", "podCode": "100001", "posCode": "P02", "positionCode": "P02" }, { "areaCode": "", "materialLot": "", "podCode": "100002", "posCode": "P03", "positionCode": "P03" }] }</pre>		

		<pre> }, }, "message": "成功", "reqCode": "1541954B96B1110" }\$\$ </pre>
--	--	--

3.1.7 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 或 TCP				
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。 由上层系统提供
	reqTime	String	20	否	请求时间戳，格式：“yyyy-MM-dd HH:mm:ss”。 由上层系统提供
	clientCode	String	16	否	客户端编号，如 PDA，HCWMS 等。 由 RCS-2000 告知上层系统
	tokenCode	String	64	否	令牌号，由调度系统颁发。 由 RCS-2000 告知上层系统
	interfaceName	String	64		queryTaskStatus TCP 协议必传，REST 协议不用传，传了也不影响
	taskCodes	String[]	64	否	任务编号数组 任务编号数组与 AGV 编号至少传其

						中之一
	agvCode		String	16	否	AGV 编号 任务编号数组与 AGV 编号至少传其中之一
应答	code		String		返回码	
	data [列表]	taskCode	String		任务编号	
		taskTyp	String		任务类型	
		taskStatus	String		任务状态：0-发送异常，1-已创建，2-正在执行，3-正在发送，4-正在取消，5-取消完成，6-正在重发，9-已结束，10-被打断，（0、1、2、5、9 常用）	
	message		String		返回消息	
	reqCode		String		请求编号	
示例	请求	REST: POST http://IP:PORT/cms/services/rest/hikRpcService/queryTaskStatus TCP: 通过 TCP 端口传输以下示例字符串，需要保留末尾\$\$，做为分隔符使用。 REST 需要去掉末尾\$\$ { "reqCode": "1541954B96B1110", "reqTime": "", "clientCode": "", "tokenCode": "", "interfaceName": "queryTaskStatus", "agvCode": "", "taskCodes": ["123", "234"] }\$\$				
	应答	{ "code": "0", "data": [{ "taskCode": "234", "taskStatus": "正在执行", "taskTyp": "厂内货架搬运" },]				

		<pre> { "taskCode": "123", "taskStatus": "已结束", "taskTyp": "厂内货架搬运" }], "message": "成功", "reqCode": "1541954B96B1110" }\$\$ </pre>
--	--	---

3.2 上层平台提供的接口

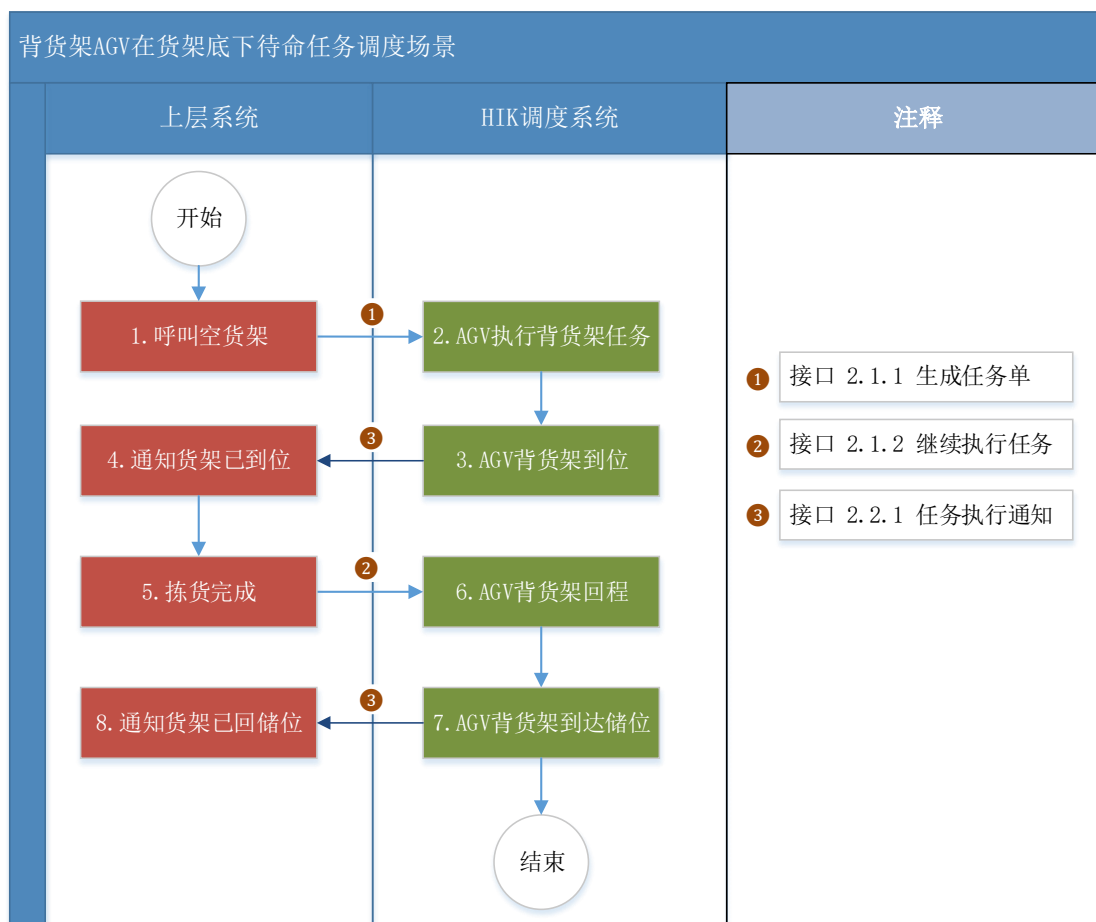
3.2.1 数据同步通知

接口名	syncNotify				
功能说明	数据同步通知接口				
接口协议	REST 或 TCP				
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	是	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。
	interfaceName	String	64		syncNotify

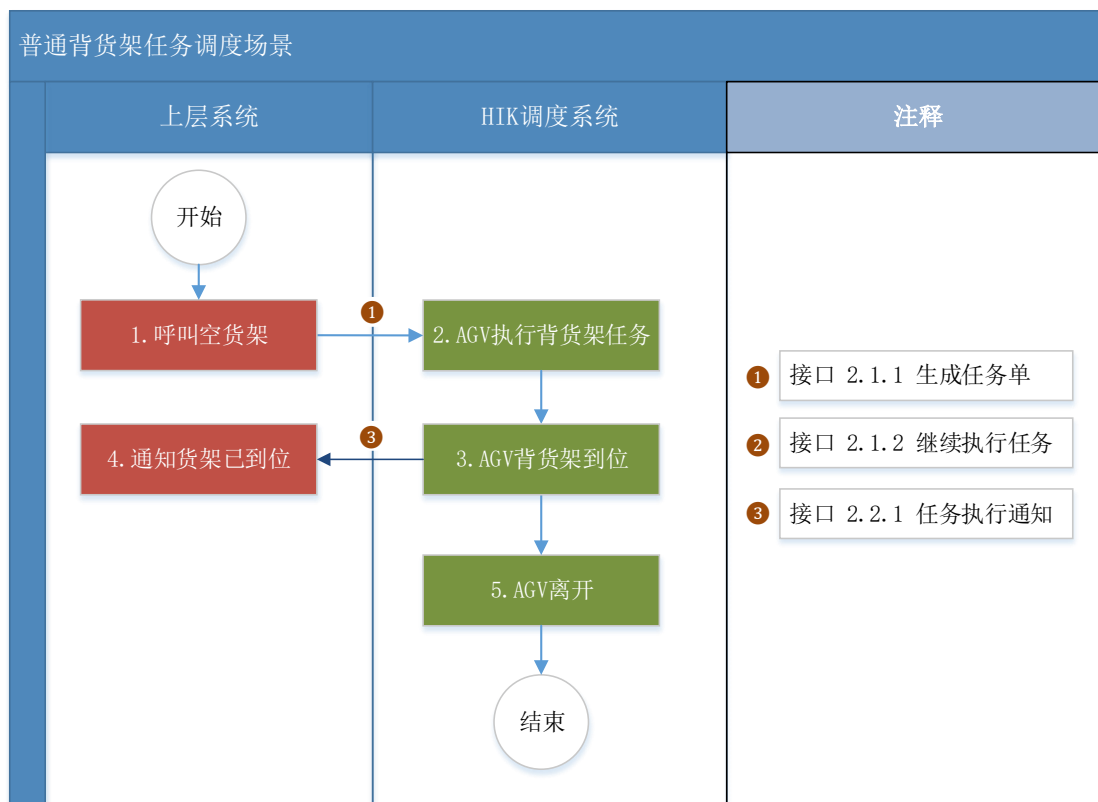
					TCP 协议必传，REST 协议不用传，传了也不影响
	notifyNotify	String	16	是	通知类型： mapPos:地图位置信息同步，收到消息后，调用 HIK 提供的 3.1.5 接口。
	mapShortName	String	32	是	地图简称，需要同步的地图，由 RCS-2000 告知上层系统 不填表示同步全部地图。
	data	String	2000	否	自定义字段，不超过 2000 个字符
应答	code	String			返回码
	message	String			返回消息
	reqCode	String			请求编号
	data	String			自定义返回
示例	请求	REST: POST http://IP:PORT/xxx/agv/agvCallbackService/syncNotify TCP: 通过 TCP 端口传输以下示例字符串，需要保留末尾\$\$，做为分隔符使用。 REST 需要去掉末尾\$\$ <pre>{ "reqCode": "", "reqTime": "", "clientCode": "", "tokenCode": "", "notifyNotify": "", "mapShortName": "", "data": "" }</pre> \$\$			
	应答	<pre>{ "code": "0", "data": "", "message": "成功", "reqCode": "1541954B96B1112" }</pre> \$\$			

4 典型调度场景

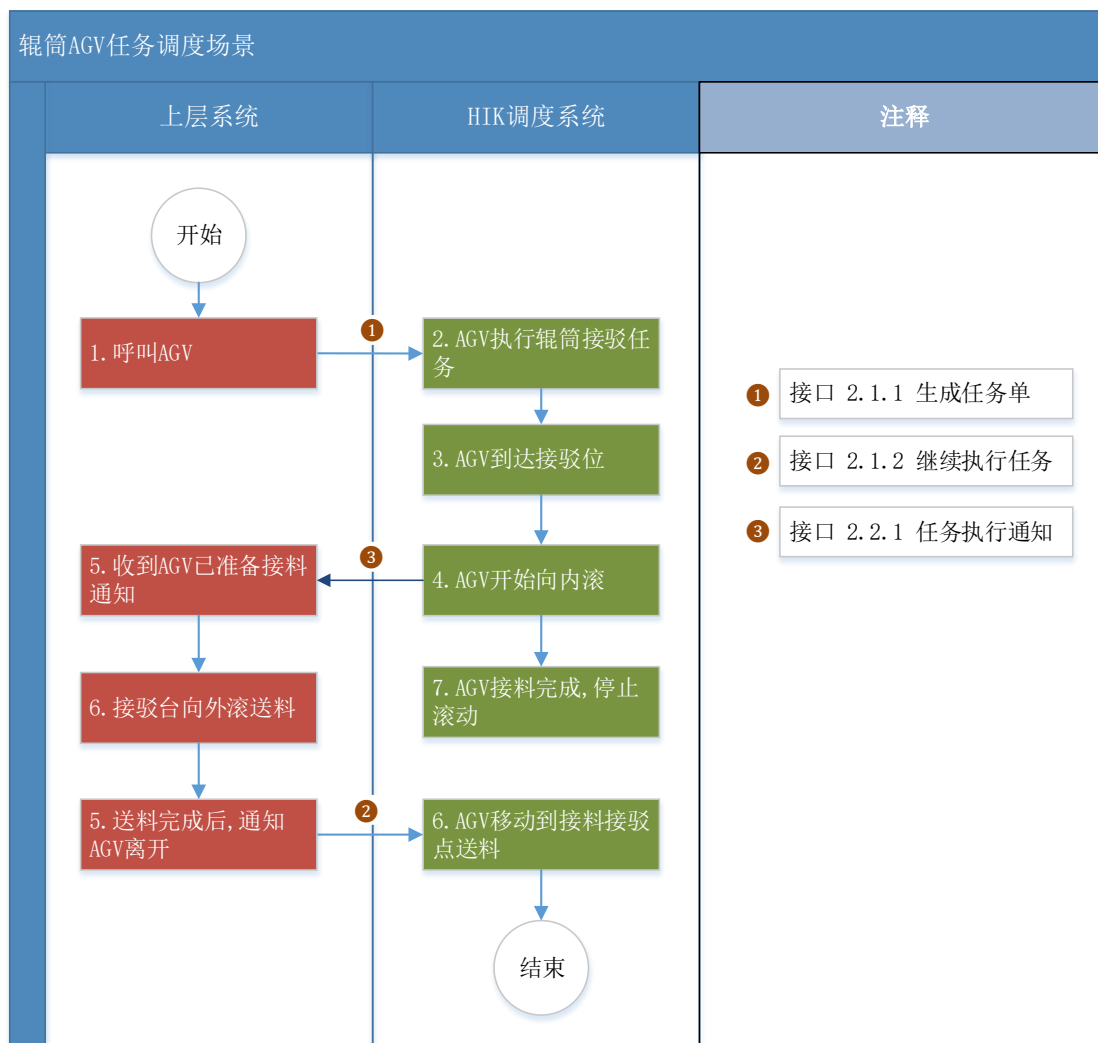
场景一：背货架 AGV 在货架底下待命任务调度场景



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



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/cms/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/cms/services/rest/hikRpcService/continueTask>

入参:

```
{
  "reqCode": "1231233 每次传入必须唯一",
  "wbCode": "工作位"
}
```

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"
}
```

6 附件

6.1 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.



C# demo rest.rar



Javademore_rest协议.rar

7更新记录

更新日期	更新人	更新内容
2017-4-17	张启帆	创建文档, 编写初稿。
2017-6-9	吴华朋	更新为 REST 文档, 并加入调度流程
2017-6-16	张启帆	新加部分接口说明, 细化各字段说明
2017-8-22	吴华朋	加入 TCP 协议, 文档同时支持 REST 与 TCP 两种协议
2018-4-01	吴华朋	更新文档, 加入位置禁用与启用接口, 修改难理解的名称
2018-4-07	吴华朋	将接口分为常用接口与可选接口, 加入调度常用场景, 以及接口初步对接入参示例, 并附上 C#的 DEMO, 上层平台参考
2019-3-29	白银朋	更新接口字段说明、示例、C#demo