欢迎

欢迎使用羽刃 OS 系统。

产品简介

羽刃 OS 致力于为企业提供工业生产现场的数字化、透明化、自动化解决方案, 实现高效生产管理。

系统使用

- 羽刃 OS 系统默认使用 80 端口,确保联网后,获得设备 IP 地址。
- 在工控机设备局域网内, 打开浏览器(建议使用 Chrome), 输入羽刃 OS 系统设备的 IP 地址, 即可访问系统。
- 系统初始账户为: admin/123456

文档更新

文档更新时间	更新内容
2023/09/13	增加"羽刃"的相关描述
2023/09/13	修改为符合 4.0 版本; HMI 模版和工艺模快修改
2023/07/25	建立文档

页面概览

首页



展示羽刃 OS 系统当前的运行状态,展示内容说明:

- 今日异常最多:展示今日异常最多的引擎节点
- 今日异常:今日异常次数
- 今日日志:今日存储的日志
- 今日运行次数: 今日总运行的次数
- 引擎总运行次数:引擎总运行次数
- 已安装应用:已安装的应用
- 网络监控: 羽刃 OS 系统硬件设备的网络情况监控
- CPU 使用率: 羽刃 OS 系统硬件设备的 CPU 使用率
- 内存使用率: 羽刃 OS 系统硬件设备的内容使用率

- 存储:设备数据占用数据量,以及羽刃 OS 系统的存储使用情况
- 最近7天异常次数: 展示最近7天的异常趋势
- 今日异常排名:展示今日的异常排名
- 最近7天运行次数:展示最近7天的运行次数
- 今日运行次数:展示今日引擎运行次数

消息中心

	[11]			⊟ 标题/内容	Q. 查看	
↔刷網	Я	盲 删除	仅保留30天内系	统消息		
		操作	日期	标题	内容	
1		详情	2023-07-13 15:37:40.013	测试	{"shiftList":{{"date":"2023-07-13";start_date":"2023-07-13 08:00:00.000";end_date":"2023-07-13	
2		详情	2023-07-13 15:37:35.013	测试	{"shiftList";{{"date":"2023-07-13","start_date":"2023-07-13 08:00:00.000","end_date":"2023-07-13	
3		详情	2023-07-13 15:37:30.025	测试	{"shiftList";{{"date":"2023-07-13","start_date":"2023-07-13 08:00:00.000","end_date":"2023-07-13	
4		详情	2023-07-13 15:37:25.014	测试	{"shiftList";{{"date":"2023-07-13","start_date":"2023-07-13 08:00:00.000","end_date":"2023-07-13	
5		详情	2023-07-13 15:37:20.025	测试	{"shiftList";{{"date":"2023-07-13","start_date":"2023-07-13 08:00:00.000","end_date":"2023-07-13	
6		详情	2023-07-13 15:37:15.022	测试	{"shiftList":{{"date":"2023-07-13","start_date":"2023-07-13 08:00:00.000","end_date":"2023-07-13	
7		详情	2023-07-13	测试	{"shiftList":[{"date":"2023-07-13","start_date":"2023-07-13 08:00:00.000","end_date":"2023-07-13	

将你关注的生产现场的状态推送给你,比如设备异常、产量达标等。

通过数据机器人的配置,可以根据你的业务需求完全自定义提示内容和触发条件。

应用包



通过"应用市场"获取应用后,可用的应用即展示在此处。你可以将你关注的应用 内容"标星","标星"后应用入口将添加到左侧的导航栏,方便后续快捷打开。 你也可以通过拖动应用包列表来调整栏目顺序,以便使用。

应用市场

	应用市场	首页	消息中心	全部应用			已注册(🛛 🙆 admin 👻
🤝 后台管理	■ 全部应用						请输入关键字	Q
<mark>⊷</mark> 首页	≣ 功能模块							
▲ 消息中心	🔡 数据运行组件	The 645 445 14		所有	功能模块 数据运行	丁组件 工控协议 连接器		+= + = +
■ 应用包 >	工控协议	切能候状						亘有王部ノ
谷稼动率 >	≣ 连接器	-	报警管理		戸葬町	生产工单	R100.00 T Z	戸森取
🗟 生产管理 >			官理手问以面顶言		1 37.4K	10月10月10月2小头现手间主/	工半皆坯	
🏝 设备管理 >								
🗊 工艺管理 >		s 😍	工艺配方 工艺配方数字化管理和应用		已获取	执行器 → 分部第三方程序调用		已获取
嬰 监控看板 >								
🛒 数据分析 >			监控看板			设备稼动率		
፟⊠数据管理 >		<u>~</u>	拖拉拽快速实现数字化监控		已获取	设备稼动率计算		已获取
◎ 应用市场 >		数据运行组	1件					查看全部 >
			PDF解析			PDF转图片		_
✿ 系统配置 >			提取PDF中的文字		已获取	PDF转换为图片		已获取 🥌
«								

功能应用、工控协议、连接器、数据运行组件等等,我们都将打包到"应用市场" 中,你根据需要获取即可使用。

- 功能类应用,获取后将在"应用包"中查看到对应的入口。
- 工控协议、连接器等,获取后你即可在"数据机器人"中,与你的数据库或设备进行连接。
- 数据运行组件为"数据机器人"中,云雀提供的数据处理、计算类组件。

数据机器人

	数据机器人	全部应用 数据运行	指件 连接器 功能模块 数据机器人 ≥ 更多	已注册 🕜 🦲 admin 👻
🎔 后台管理	■ 数据机器人	组件库 组件库搜索 Q 三	④ Mes对接 … 海绵切割 … pdf解析测试… 绑定设备数3 … 过程参数 … ∀ 更多	合 已锁定 ① 导入
🛃 首页	≣ 运行日志	Trigger		● ½ [] @ 問
阜 消息中心	謳 异常日志	Feature	>> (0) 日志打印 →	
■ 应用包 >	≣ 持久寄存器(\$dbs)	Process PRM		
🔒 设备稼动率 >	讄 协议配置管理	Algorithm Dification		
🕏 生产管理 >		▶ 🗋 ICA.	MRQ7 is delivered	
名 设备管理 >		CNC data mining		
工艺管理 >		AGV robot	○ 第□写入	
🖷 监控看板 >		 Detwork 	γ	
🔄 数据分析 >		Staff finch		
🖾 数据管理 >		Archive Document process		->> Postgresql
		Monitor board	获取Mes需要> ③ 判断 \$\Delta_224111 ● ● 请求MES11	Bind device
🕅 应用市场 >		Equipment Mgr Alarm Mgr	uyruxent 20224111 ->> ◇ 解析返回的数	data
		 Deperation rate of equ 	O & Manual Co-	
痛系統配置 >		🕨 🗋 Data Mgr		「「「」」「利助走出版」
		Prod work order	◎ 日志 ◎ 异常	i 7 ^

数据机器人为羽刃 OS 的核心模块,通过使用云雀提供的功能组件和工控协议,通过模块化的配置,即可以完成本系统与设备的连接、生产数据的采集计算、产线业务流程的数字化。

系统配置

	系统配置	全部应用 数据机器人 排班设置 设备总览 系统配置 > 更多 已注册 (🔉 🙆 admin 👻
■物联平台	■ 系統配置	× 92	
🛂 首页	譜 用户管理	* 1617	
🌲 消息中心	讄 角色	配置管理平台	
■ 应用包 >	应用授权	1.平台图标 仅能上传png/pg/pgg 2.平会数 照到4个中文专指10个(例字上在文) 不予給入任何符号成立株	
🐖 数据分析 >		2.1日日時間間の「モステリ」の「(8.ティスス)」、1.4(人口)の5.34110	
😫 生产管理 >			
🖳 监控看板 🔷			
ቓ 能源管理 →		平台名称 : 物联平台	
🔓 设备稼动率 >		已启用:	
∽ 数据管理 >		边缘计算授权码	
② 过程参数 >		授权码有效期: 永久授权	
🗊 工艺管理 >			
⑦ 应用市场 >			
数据机器人 >			•
▲ 系統配置 >			8
«			

可以设置 logo 或平台名称、创建账号、角色权限管理等操作。

数据机器人

数据机器人

数据机器人为羽刃 OS 的核心模块,通过使用云雀提供的功能组件和工控协议, 通过模块化的配置,即可以完成本系统与设备的连接、生产数据的采集计算、产 线业务流程的数字化。

页面说明

~	机器人	工艺配方 排班设	工艺配方 排班设置 设备可动率 消息中心 > 更多 已注册 《				
ፇ robot	譜 数据机器人	相件席 組件序指索 Q 三	引擎tab ① 数据瓶入 … 设备管理 … 【生产工单 … 】 Mia测试 … 】大屏看板_副 … ⇒	夏多 土导入			
🔛 首页	譜 运行日志	▶ □ 競发 组件目录	Q 开启运行:	Q Q X [] @ #			
↓ 消息中心	譜 异常日志	▶ 🗀 功能	运行开关				
■ 应用包 >	☷ 持久寄存器(\$dbs)	 ▶ □ 过程参数 ▶ □ 算法 					
嬰 监控看板 >		▶ 🗀 通知					
		▶ 🗀 工控协议					
● 报警管理 →		▶ □ AGV机器人		C HADDED A			
会生产管理 >		 ▶ □ 云连接 ▶ □ 网络 					
🙇 设备管理 🚿		▶ □ 数据库		人 炭送至塩控着 人			
11 工艺管理 >		▶ □ 文件处理 ▶ □ 收拾重新		1 12			
♥ 过程参数 >		► □ 设备管理					
🟟 应用市场		▶ □ 报警管理					
◎ 机器人 >>		▶ 🗀 设备稼动率					
		▶ □ 数据管理		9			
44 糸筑配直 >		▶ □ 生产工单	通行日志 日本				
*		▶ □ 工艺管理	V HW I WITH	a 1 //			

区域说明

- 顶部 tab: 目前已经创建的引擎
- 左侧:组件目录,组件拖动出来后即形成节点
- 中间:数据流规则主区域
- 右侧:显示日志/异常操作
- 下方: 打印的日志以及组件的异常提示

• 开启运行:开关开启后,当前引擎才会运行

创建引擎

在目录区域点击"加号",即可创建一个新的引擎。各个引擎的数据之间互相独立。



业务节点

从左侧的组件库中,将需要的节点拖到中间的画板中。然后通过连线将各个节点 连接起来,现成数据业务流程。



节点详情

双击节点即可打开节点的详情界面。

基础属性配置

		✓ 完成	取消
基础属性	节点配置 ⑦ 帮助手册		
基本信息			
开启运行:			
节点类型:	Euromap63读		
名称:	Euromap63读		
备注:			
			0/200
节点输出			
节点输出:	覆盖上一节点 ▼ 节点返回值不为{}对象时,则自动挂载\$msg.value中		
异常处理			
异常时,自动打	[印异常至控制台,以及保存异常信息至异常日志中		
处理方式:	终止流程		v

所有节点的基础属性配置都是一样的:

- 开启运行:默认开启,关闭后该节点以及后续节点都不会运行
- 节点类型: 节点的类型
- 节点名称:默认等于节点类型,用户可修改
- 备注: 节点的备注
- 节点输出: 节点输出的操作, 节点输出支持三种方式:

1、覆盖上一节点:本节点的数据会直接覆盖上一节点的数据

2、合并数据:当前节点的数据会和上一节点的数据进行合并,当要合并的数据为 list,或者单一数据时(int 值/字符串值等),默认数据的 key 为 value

3、挂在到\$msg 节点: 当前节点的数据会放在\$msg 节点下, 挂载的变量由用户 自定义

- 异常处理:当节点异常时,如何处理,目前支持二种方式:
 - 1、终止流程:终止当前节点以及后续流程
 - 2、忽略异常并继续执行: 忽略当前异常并继续往下执行

节点配置

各个节点的【节点配置】有所不同,具体请查看节点详情中的【帮助手册】tab。

数据流程规则

将节点连接起来后,每一个节点产生的数据通过【\$msg.data_name】方式发送至下一个节点,下一个节点即可对获取都到数据进行处理。然后再发送至再下一个节点。

用例:



如上图中,通过【Euromap63读】节点,读取到设备的状态数据后,会将数据 发送至【判断】节点,根据设备的数据,判断是否需要报工,若是则报工。 或者是将读取到的设备状态数据,发送到【函数】组件,【函数】组件可以将数 据进行清洗处理后,将处理的结果数据发送但【发送至监控看板】组件。 小 tips:

可以通过【日志打印】组件,来查看某个节点的输出的数据的具体值哦

表达式

羽刃 OS 系统中,很多组件可以使用数据表达式,由用户以低代码的形式,自定 义数据处理的算法。数据处理表达式使用 JavaScript 语言,无需编译,保存即 生效。

基础属性	节点配置	② 帮助手册	
🗹 使用表达	式		
判断表达式配	置表达式 代码	马调试 内存寄存器(\$db))

表达式配置弹窗, 左侧为云雀提供的已封装好的方法。双击可以使用。用户也可

以在右侧的输入框中, 编辑自己的函数体。

🦳 开启数据处理 🚺 🖾 新增函数	代码格式化 测试执行 代码调试 內存審存語(\$db)	
▶ □ 系统变量	1	
▼ 🗋 通用方法		
10进制转2进制		
2进制转10进制		
10进制转16进制		
16进制转10进制		
10进制转二进制数组		
布尔值是否相等		
字符串相等		
字符串相等(忽略大小写)		
▶ 🗋 数据转换		
▶ 🗋 内存寄存器		
▶ □ 持久寄存器		
▶ □ 自定义函数		

自定义函数

云雀允许用户创建自己的函数库,将常用的方法封装起来,后续可以直接调用, 避免重复编写。



第一步: 输入函数的基础信息设置: 函数名、所需参数等。

新建自定义函数

基本	信息	函数体
	_	
显示	示名称:	可以是中文或英文
Ē	函数名:	需要符合JS命名规范
参数	效列表:	多个参数使用;;隔开
函数	数说明:	

第二步: 输入函数体内容。

新建自定义的	i数	
基本信息	函数体	
代码格式化		
1		
	确认 取消	

第三步:确认保存后即可在函数列表中看到新创建的自定义函数。



某些节点可以手动执行(比如【定时触发】【手动触发】),点击执行按钮则直接执行一次该节点。

٩		开启运行: 🌑	6 = Q	. ♥. ≭ [] @ #
② 定时触发	2			
② 定时触发	2			

节点运行状态

节点的下方,会显示当前节点的最后运行时间和运行次数

Q		开启运行: 🌑		⊕ :::
	,->>● ▲ ModbusTCP 读	> З 判断		3
🕘 定时触发 🥵	-			
○ <u>18:02:49</u> .37 最后运行时间	·→> 2 数据注入 ⓒ 18:02:49. <u>37</u> 运	> [●] 日志打印 行次数 [○] 18:02:49.37	·->> ⑦ 西门子S7写	3

节点异常状态

节点上发显示报警图标时,代表节点运行出现异常

	→ ModbusTCP 读 》	3
⊙ 18:03:26.74	 → 2 数据注入 → 2 数据注入 → 18:03:26.74 → 18:03:26.74 	}

日志&异常

组件库 组件库搜索 Q 三	① 大屏看板_	动 … 【机床数排	据采集 … 【 工位监控	… MIA_大屏看 …	·	… ※ 更多
 ▶		Q		开启运行	ī: C	
 · 手动触发 · ③ · 初始化触发 · □ · □ · □						->0
 □ 切能 ◎ 数据注入 	母 日志 │ ①	异常				
② 格式化	2023-05-17 18:15:18	日志打印	0			
 	2023-05-17 18:15:19	日志打印	0			

数据机器人下方将展示日志和异常:

- 日志: 当某个节点与【日志打印】组件连接后, 这里将展示这个节点输出的数据值。
- 异常:当某个节点发生异常时,这里将会自动显示异常。

运行日志

Sec. 1	机器人	Mia_t	est Co	-	异常日志 持久寄	存器(\$ 运行日志	∀ 更多	已注册 🕜 🦲 admin 👻						
🔻 robot	譜 数据机器人	运行	□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□											
😫 首页	III 运行日志	2.13												
阜 消息中心	謳 异常日志	运行	疗规则名	称/节点名	弥内容 Q 查询									
■ 应用包 >	譜 持久寄存器(\$dbs)	8	删除					¢ ()						
學 监控看板 →				操作	运行规则名称	节点名称	时间	日志信息						
፼ 数据分析 >		1		详情	test001	日志打印	2023-05-11 11:41:00.020	{"value1":12,"value2":53}						
		2		详情	test001	日志打印1121	2023-05-11 11:41:00.020	0						
▲ 设备管理 >		3		详情	test001	日志打印	2023-05-11 11:40:50.010	{"value1":12,"value2":53}						
IIII 工艺管理 >		4		详情	test001	日志打印1121	2023-05-11 11:40:50.010	0						
 ② 过程参数 > 向 应用市场 		5		详情	test001	日志打印1121	2023-05-11 11:40:40.024	0						
◎ 机器人 >		6		详情	test001	日志打印	2023-05-11 11:40:40.024	{"value1":12,"value2":53}						
✿系统配置 >						. k < 1 . #	6,118,851页 > > 15	∽ 1-15 共91,782,759条						

组件【日志打印】节点,配置日志保存后,日志就会默认存储,该模块即可查看 保存的日志。

日志模块仅保存 30 天内的日志, 超时日志会被自动删除

异常日志

	数据机器人	排班设	置	1	设备总览 系统[配置 运行日志	异常日志	⊗ 更多	已注册 🕜 🕒 admin					
初联十百	数据机器人	异常F	第日志 (仅保留30天内数据)											
会 首页	讀 运行日志													
1 消息中心	■ 异常日志	运行	行規则名称/节点类型/节点名称/2 0 宣看											
■ 应用包 >	讀 持久寄存器(\$dbs)	t f	削除						Ċ					
🗙 数据分析 >	讄 协议配置管理			操作	运行规则名称	节点名称	时间	节点类型	异常					
3 生产管理 >		1		详情	Andy	ModbusTCP读	2023-07-14 13:49:45	ModbusTCP读	Modbus TCP未配置读取点位					
📲 监控看板 >		2		详情	场景-柔性生产	西门子S7读	2023-07-14 13:49:45	西门子S7读	Read timed out					
5 能源管理 >		3		详情	默认规则	倍福Ads读	2023-07-14 13:49:45	倍福Ads读	Cannot assign requested add					
设备稼动率 >		4		详情	演示场景parekr	MC TCP ASCII读	2023-07-14 13:49:45	MC TCP ASCII读	MC协议连接失败:Connection informa					
△数据管理 >		5		详情	场景-柔性生产	设备读数存储	2023-07-14 13:49:45	设备读数存储	绑定设备为空					
♥ 过程参数 >		6		详情	场景-柔性生产	设备报警	2023-07-14 13:49:45	设备报警	绑定设备为空					
1 工艺管理 >		7		详情	场景-柔性生产	西门子S7读	2023-07-14 13:49:44	西门子S7读	Read timed out					
		8		详情	场景-柔性生产	西门子S7读	2023-07-14 13:49:43	西门子S7读	Read timed out					
● 应用市场 >		9		详情	演示场景parekr	西门子S7读	2023-07-14 13:49:40	西门子S7读	Connection reset					
◙ 数据机器人 >		10		详情	场景-柔性生产	西门子S7读	2023-07-14 13:49:40	西门子S7读	Connection reset					
✿系统配置 > 						k < 1	共887,207页 > > 15	~	1-15 共 13,308,099 条					

当节点发生异常时,系统会自动记录。在该模块进行异常数据的追溯。**日志模块仅保存

持久寄存器(\$dbs)

	数据机器人	系统	記置		运行日志 异	常日志 持久	寄存器(\$ 数据机器人	≫ 更多	已注册 🕜	🕒 ad	min 👻	
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□	数据机器人	持久	久寄存器(\$dbs)									
🛃 首页	🏭 运行日志											
▲ 消息中心	譜 异常日志	数技	数据ID/名称 Q 查看									
■ 应用包 >	▮ 持久寄存器(\$dbs)	+	新增数排	8	删除					Ċ	٨	
🛒 数据分析 >	■ 协议配置管理			操作	数据ID	数据名称	数据值		更新时间			
🗟 生产管理 >		1		编辑	Dc231701004FAL9		10.00		2023-07-14 13:52:00.169			
🖳 监控看板 💈		2		编辑	1230502001TGNK		4.00		2023-07-14 13:50:00.016			
♥ 能源管理 >		3		编辑	9230702002FAL9		4.00		2023-07-14 13:50:00.016			
		4		编辑	Dc231701002B1BF		4.00		2023-07-14 13:50:00.016			
□ 设备核动率 >		5		编辑	Dc231701003NHFV		4.00		2023-07-14 13:50:00.015			
⊻ 数据管理 >		6		编辑	Dc231701001FAL9		0.00		2023-05-15 10:30:00.096			
② 过程参数 >		7		编辑	Dc231701001TGNK		0.00		2023-05-15 10:30:00.079			
🗊 工艺管理 >		8		编辑	Dc231701001AXF7		4.00		2023-07-14 13:50:00.015			
应用市场 >		9		编辑	Dc231701001B1BF		0.00		2023-05-15 10:30:00.079			
A 86 102 10 22 1		10		编辑	1230502001FAL9		2871.00		2023-05-10 11:37:05.056			
		11		644 f 13	volue1	8h ti2 1	••		2022 02 02 14:27:40 241		A	
№ 系统配置 > 						K <	1 共1页 > > 15 ¥		1-1	1 共1	1条	

数据机器人提供"持久寄存器"功能,用于存放全局变量。在数据机器人中,通过 {\$dbs.data_name},即可以直接引用。

协议配置管理

	数据机器人	运行日	日志	异常日志	持久寄存器(\$	数据机器人	协议配置管理
物联平台	▓ 数据机器人						
首页	▓ 运行日志	工技	空协议	~	连接名称	Q, 查看	
间息中心	▓ 异常日志	+	新建协议	t l			
2.用包 >	讀 持久寄存器(\$dbs)			操作	连接名称	工控协议	IP
≅分析 >	协议配置管理	1		编辑 删除	local_modbus_tcp	Modbus TCP读	127.0.0.1
wiatern s		2		编辑 删除	哦哦哦啦啦啦	GE PLC连接读	192.168.1.1
官理 >		3		编辑 删除	测试	倍福Ads读	192.255.1.1
看板 >		4		编辑 删除	paker-西门子	西门子协议读	127.0.0.1
理 >		5		编辑 删除	local-01	Euromap 63	
家动率 >		6		编辑 删除	FTP连接 -parker	FTP Server	127.0.0.1
理 >		7		编辑 删除	西门子读取	西门子协议读	127.0.0.1
>		8		编辑 删除	三菱本地测试	MC TCP ASCII读	127.0.0.1
! >		9		编辑 删除	本地连接	Modbus TCP读	127.0.0.1
		10		编辑 删除	parker-slave1	Modbus TCP读	127.0.0.1
场 >		11		编辑 删除	西门子	西门子协议读	127.0.0.1
居机器人 >						< < 1	共2页 > > 15 ¥
記置 >							

数据机器人中"工控协议"类的组件,需要进行连接字段配置。各个协议的配置内容有所不同,根据现场的实际情况即可。

函数使用说明

功能介绍

羽刃 OS 内置函数引擎(函数引擎语言为 ES6 标准 JavaScript), 支持用户手动写表达式进行数据处理。



节点配置

 基础属性
 节点配置
 ⑦帮助手册

 函数表达式
 配置表达式
 代码调试 内存寄存器(\$db)

 //从上游节点取设备状态
 ()

 var deviceStatus = \$msg.deviceStatus
 ()

 //将设备状态存入内存寄存器0号点位(ps:内存寄存器支持索引值: 0-499一共500个点位)
 ()

 \$db.setIntValue(0, deviceStatus)
 ()

 //判断上升治
 (关 (false) -> 开(true))

 let deviceRisingEdge = \$db.isRisingEdge(0)
 ()

 //逻辑处理, 将处理结果返回
 ()

 //todo 这里补充具体业务
 ()

 return {
 deviceRisingEdge

 }
 ()

配置表达式

内置系统变量、内存寄存器、持久寄存器、系统方法,还可以使用自定义函数

▶ 新增函数	代码格式化】代码调试】内存寄存器(\$db)	
 ▶ □ 系统变量 ▶ □ 内存寄存器 > □ 持久寄存器 > □ 系统方法 ▶ □ 自定义函数 	1 ///L上游专点取设备状态 2 var deviceStatus = Smsg.deviceStatus 3 //裕论备状态存入内存寄存器包号流位(ps:内存寄存器支持索引值: 0~499—共500个点位) 4 Sdb.setIntValue(0, deviceStatus) 5 //判断上升治(关 (false) ~> 开(true)) 6 let deviceRisingEdge = 36b.isRisingEdge(0) 7 //逻辑处理。将处理是带返回 7 //过程缺处理,将处理是带返回 8 //todo 这里补充具体业务 9 return { 10 deviceRisingEdge 11 }	

系统变量\$msg



\$msg: 是上一个节点输出结果的引用变量

内存寄存器\$db



- 内存寄存器:引擎内置的数据存储对象,可以理解为一个数组,数组内可以存储各种数据类型的值,目前支持:bool,int,string,float,map (Key-Value 结构),array。
- \$db: 是内存寄存器的对象引用变量,通过\$db 对内存寄存器设置或获取不同类型的值。其封装了各种数据类型的 get/set 方法,通过下标索引参数(0...499)获取/设置。\$db.isRisingEdge/\$db.isFallingEdge 用于判断上升/下降沿。上升/下降沿判断逻辑是通过上一个值和当前值变化判断,上升沿:false -> true,下降沿:true -> false。注意这里 true 和 false 不是严格意义上 bool 类型, JavaScript 把 null、undefined、0、NaN 和空字符串''视为 false,其他值一概视为 true

例如:

```
$db.setBoolValue(0,true);//表示设置下标 0 的数据为 bool 类型的值 true;
$db.setBoolValue(0,false);//表示设置下标 0 的数据为 bool 类型的值 false;
$db.isFallingEdge(0); //true
```

```
$db.setIntValue(1,0);//表示设置下标 0 的数据为 int 类型的值 0;
$db.setIntValue(1,1);//表示设置下标 0 的数据为 int 类型的值 1;
$db.isRisingEdge(1); //true
```

Сору

PS: 内存寄存器顾名思义数据存储于内存,系统重启时内存寄存器中所有的数据

都将丢失

1.1.1.1. 持久寄存器\$dbs



- 持久寄存器:引擎内置的数据存储对象,可以理解为一个 map (Key-Value),和内存寄存器一样,可以支持存储 bool,int,string,float,map (Key-Value 结构),array 类型数据。数据都 存放于 SDR 数据库
- \$dbs: 是内存寄存器的对象引用变量,通过\$dbs对持久寄存器设置或获取不同类型的值。其封装了各种数据类型的get/set方法,通过 string 类型 Key 获取设置值
- 总之,内存或持久寄存器给引擎提供了公共数据存储途径,借助寄存器在
 系统内跨引擎通讯。

系统方法



SDR 内置了丰富的系统方法,根据方法名称就能直观理解每个方法的含义,需

要强调的是 Lodash 相关的方法并未完整列出,如果列表中没有找到 Lodash 相关方法,您也可以直接使用。LodashJs 文档

1.1.1.2. 自定义方法

如果有些逻辑可以复用,我们建议您抽取出来封装成单独的方法。我们提供了自定义方法功能。

数据处理	测试		×
🕞 新增方法]	代码格式化] 代码调试] [内存著存题(\$db)]	
▶ 🗋 系统变量	Ē	1	
▶ 🗋 内存寄存	字器		
▶ 🗋 持久寄存	子器		
 二 系统方法 	去		
▼ □ 自定义7	方法		
randoi	m		
		商制认取消	
基本信息	函数体 测试		×
* 显示名称:	random		
* 函数名:	random		
参数列表:	min,max		
函数说明:			
			/



基本信息测试	×
代码格式化 代码调试	
<pre>1 if (min > max) { 2 [min, max] = [max, min] 3 } 4 return min + Math_floor(Math_random(max = min) * (max = min))</pre>	1920 m
	C
76.21	

基本信息 函数体 测试		×
测试数据 JSON格式检查 测试执行	测试结果 JSON格式检查	
1 { 2 min:100,max:1000 3 }	1 913	
	确认取消	

1.1.2.代码调试

业务逻辑写完后,可以使用代码调试功能调试代码。具体的操作如下:

1. 将 debugger 作为单独一行插入到需要调试的代码前

数据处理 测试		×	ζ
G 新增方法	代码格式化 代码调试 内存寄存器(\$db)		
 ○ 系统变量 ○ 内存寄存器 ○ 持久寄存器 ○ 系统方法 ○ 自定义方法 ⑦ andom 	<pre>1 let min = 100, max = 1000; 2 debugger 3 let randomInt = random(100, 1000); 4 5 return { min, max, randomInt }</pre>		-
	論 し		

2. 点击代码调试按钮, 将弹出框中调试链接地址复制下来

代码调试	×
1. 在需要调试的代码行处,增加 debugger 标识 2. 使用Chrome浏览器,在新的标签页中打开如下链接 3. 当数据处理代码运行时,自动弹出调试功能	
<pre>devtools://devtools/bundled/js_app.html?ws=localhost:4242/engine/debug</pre>	

3. 在浏览器新 Tab 中打开调试链接地址



4. 将函数节点保存,并触发执行



5. 回到第三步打开的新 Tab,将看到调试断点已经打在 debugger 语句上,

按照浏览器调试 JS 脚步的机制一步步往下调试即可 ●●● ◎ Dev/Dels ← → C © dev/cols/Jdev/cols/bundled/js_app.html?ws=localhost.4242/engine/debug



应用授权

应用授权

应用市场

	应用市场	首页	消息中心	全部应用				已注册	📀 (🕙 admin 👻
🤝 后台管理	計 全部应用							请输入关键字	Q
<mark></mark> 首页	≣ 功能模块								
阜 消息中心	数据运行组件	(al. (at att 14		所有	功能模块 数据运行	厅组件 工 <u>排</u>	空协议 连接器		
ᡱ 应用包 >	工控协议	切能模块							宣有全部 >
🔒 设备稼动率 >	∷ 连接器	1	报警管理				生产工单	No. 607 YEE	一支取
🕏 生产管理 >		10 管理车间设备报警			AX	使用估代码技术实现年间生产		非管理	L 3X.4X
🖀 设备管理 >									
🕼 工艺管理 >		o 📀	工艺配方 工艺配方数字化管理和应用		已获取	 ¥=	执行器 外部第三方程序调用		已获取
學 监控看板 >									
🐖 数据分析 >			监控看板				设备稼动率		
🐱 数据管理 >		~	拖拉拽快速实现数字化监控		已获取	<u>×</u>	设备稼动率计算		已获取
⑦ 应用市场 >		数据运行组	件						查看全部 >
数据机器人 >			PDF解析				PDF转图片		
福 系统配置 >		4	提取PDF中的文字		已获取		PDF转换为图片		已获取 🕒
*									

功能应用、工控协议、连接器、数据运行组件等等,我们都将打包到"应用市场" 中,你根据需要获取授权即可使用。

- 功能类应用,获取后将在"应用包"中查看到对应的入口。
- 工控协议、连接器等,获取后你即可在"数据机器人"中,与你的数据库或设备进行连接。
- 数据运行组件为"数据机器人"中,云雀提供的数据处理、计算类组件。

应用详情

生产工単 使用低代码技术实现车间生产工单管理 已获取	×
应用详情	
应用介绍	
车间生产信息黑盒,无法透明化生产进度? MES已经排产完成,车间生产进度还需人工录入,耗时耗力? 多批次小批量,工艺频繁切换,无法实现柔性生产? 各个工序段生产信息差异较大,无法适配各个车间?	
主要功能和特性	
• 支持低代码拖拉拽,快速匹配各个工序段生产工单	
• 支持手动、全自动报工方式	
 可实现车间柔性生产,车间生产任务自动下发至设备 	

点击应用即可进入应用详情介绍页。可以详细了解应用的能力。

授权方式

若想获取某个应用, 云雀暂时需要用户向支持人员获取应用的"授权文件", 再通过【系统配置】->【应用授权】, 将授权文件添加到系统中, 授权文件上传完成即可获得应用的授权和开放使用。

×	系统配置	î	首页 数据定义 消息中	心 全部应用 .	立用授权	已注册 🕜 🙆 admin 👻			
🐬 后台管理	📰 系统配置		21-68-2-99 +四+17						
🎝 首页	≣ 用户管理		23年1月7月12代	22项计算授权					
▲ 消息中心	≣ 角色	2.应用授权	边缘计算序列号: 7E7D-9019-0160-CA63	-4F74-910B-FB47-97B4 🖺	边缘计算授权有效期: 2023	-08-31			
▋ 应用包 >	■ 应用授权		已授权应用			3.授权人口			
			应用名称	应用类型	授权方式	授权数量			
			用户数量授权	功能模块	按使用量授权	10			
			西门子连接器	工控协议	按使用量授权	10			
			工艺配方	功能模块	永久授权	-			
			执行器	功能模块	永久授权	-			
			设备稼动率	功能模块	永久授权	-			
▲ 应用市场 >			过程参数	功能模块	永久授权	-			
● 数据机器人 >	1.系统配置		文件管理	功能模块	永久授权	-			
✿系統配置 >		置	数据管理	功能模块	永久授权	- 🔒			
 «			3640 // 14°	19-24, 6507, 822 2.44	2. An 415 417				

