

uniapp 安卓七华称重UTS原生插件

目 录

使用说明	1
使用方法	1.1
更新日志	1.2
插件方法	2
初始化	2.1
归零	2.2
强制归零	2.3
扣重	2.4
获取重量	2.5
获取扣重值	2.6
获取秤重库版本号	2.7
获取AD版本号	2.8
获取内码	2.9
设置秤重模式	2.10
设置小数点位数	2.11
设置第一量程	2.12
设置第二量程	2.13
设置通讯协议	2.14
设置零点追踪	2.15
设置分度值1	2.16
设置分度值2	2.17
设置半自动归零范围	2.18
设置开机归零范围	2.19
标定零点	2.20
标定重量	2.21
设置预扣重	2.22
获取稳定状态	2.23
获取扣重状态	2.24
获取零点状态	2.25
获取小数点位数	2.26
设置回调	2.27
释放称对象	2.28

使用方法

介绍

安卓七华称重UTS原生插件是七华称重SDK开发的插件，插件已包含SDK的全部功能，欢迎使用

联系作者

关注微信公众号可联系作者



插件地址

<https://ext.dcloud.net.cn/plugin?id=22355>

用法

在需要使用插件的页面加载以下代码

```
import * as module from "@uni_modules/leven-uts-qhScale"
```

页面内容

```
<template>
  <view style="display: flex; flex-direction: column; padding: 16px;">
    <view style="margin-bottom: 10px;">
      <u-button type="primary" text="初始化" @click="init"></u-button>
```

```

</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="归零" @click="zero"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="强制归零" @click="forceZero"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="扣重" @click="tare"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="获取重量" @click="getWeight"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="获取扣重值" @click="getTare"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="获取秤重库版本号" @click="getVersion"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="获取AD版本号" @click="getADVersion"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="获取内码" @click="getISN"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="设置秤重模式" @click="setModel"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="设置小数点位数" @click="setDecimalNumber"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="设置第一量程" @click="setFirstRange"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="设置最二量程" @click="setMaxRangeCao"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="设置通讯协议" @click="setProtocol"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="设置零点追踪" @click="setZeroTracking"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="设置分度值1" @click="setDiv1"></u-button>
</view>
<view style="margin-bottom: 10px;">
  <u-button type="primary" text="设置分度值2" @click="setDiv2"></u-button>
</view>
<view style="margin-bottom: 10px;">

```

```

    <u-button type="primary" text="设置半自动归零范围" @click="setManualZeroRange">
    </view>
    <view style="margin-bottom: 10px;">
      <u-button type="primary" text="设置开机归零范围" @click="setAutoZeroRange"></u-button>
    </view>
    <view style="margin-bottom: 10px;">
      <u-button type="primary" text="标定零点" @click="setZeroPoint"></u-button>
    </view>
    <view style="margin-bottom: 10px;">
      <u-button type="primary" text="标定重量" @click="setCalibration"></u-button>
    </view>
    <view style="margin-bottom: 10px;">
      <u-button type="primary" text="设置预扣重" @click="preTareWeight"></u-button>
    </view>
    <view style="margin-bottom: 10px;">
      <u-button type="primary" text="获取稳定状态" @click="isStableStatus"></u-button>
    </view>
    <view style="margin-bottom: 10px;">
      <u-button type="primary" text="获取扣重状态" @click="isTaredStatus"></u-button>
    </view>
    <view style="margin-bottom: 10px;">
      <u-button type="primary" text="获取零点状态" @click="isZeroStatus"></u-button>
    </view>
    <view style="margin-bottom: 10px;">
      <u-button type="primary" text="获取小数点位数" @click="getDecimalCount"></u-button>
    </view>
    <view style="margin-bottom: 10px;">
      <u-button type="primary" text="设置回调" @click="setCallback"></u-button>
    </view>
    <view style="margin-bottom: 10px;">
      <u-button type="primary" text="释放称对象" @click="deleteScale"></u-button>
    </view>
  </view>
</template>

<script>
import * as module from "@uni_modules/leven-uts-qhScale"
export default {
  data() {
    return {}
  },
  onLoad() {

  },
  methods: {
    //设置回调
    deleteScale() {
      module.deleteScale(res => {
        console.log(JSON.stringify(res))
      })
    }
  }
}

```

```
,
//设置回调
setCallback() {
  module.setCallback(res => {
    console.log(JSON.stringify(res))
  })
},
//获取小数点位数
getDecimalCount() {
  module.getDecimalCount(res => {
    console.log(JSON.stringify(res))
  })
},
//获取扣重状态
isZeroStatus() {
  module.isZeroStatus(res => {
    console.log(JSON.stringify(res))
  })
},
//获取扣重状态
isTaredStatus() {
  module.isTaredStatus(res => {
    console.log(JSON.stringify(res))
  })
},
//获取稳定状态
isStableStatus() {
  module.isStableStatus(res => {
    console.log(JSON.stringify(res))
  })
},
//设置预扣重
preTareWeight() {
  module.preTareWeight({
    //预扣重的重量
    preTareValue: "5"
  }, res => {
    console.log(JSON.stringify(res))
  })
},
//标定重量
setCalibration() {
  module.setCalibration({
    //标定的重量，必须为字符串数字
    caliValue: "5"
  }, res => {
    console.log(JSON.stringify(res))
  })
},
//标定零点
```

```

setZeroPoint() {
  module.setZeroPoint(res => {
    console.log(JSON.stringify(res))
  })
},
//设置开机归零范围
setAutoZeroRange() {
  module.setAutoZeroRange({
    // 0 : 0%
    // 1 : 2%
    // 2 : 4%
    // 3 : 10%
    // 4 : 20%
    // 5 : 50%
    // 6 : 100%
    number: 0
  }, res => {
    console.log(JSON.stringify(res))
  })
},
//设置半自动归零范围
setManualZeroRange() {
  module.setManualZeroRange({
    // 0 : 0%
    // 1 : 2%
    // 2 : 4%
    // 3 : 10%
    // 4 : 20%
    // 5 : 50%
    // 6 : 100%
    number: 0
  }, res => {
    console.log(JSON.stringify(res))
  })
},
//设置分度值2
setDiv2() {
  module.setDiv2({
    // 0 : 1
    // 1 : 2
    // 2 : 5
    // 3 : 10
    // 4 : 20
    // 5 : 50
    // 6 : 100
    number: 0
  }, res => {
    console.log(JSON.stringify(res))
  })
},

```

```

//设置分度值1
setDiv1() {
  module.setDiv1({
    // 0 : 1
    // 1 : 2
    // 2 : 5
    // 3 : 10
    // 4 : 20
    // 5 : 50
    // 6 : 100
    number: 0
  }, res => {
    console.log(JSON.stringify(res))
  })
},
//设置零点追踪
setZeroTracking() {
  module.setZeroTracking({
    // 0 : off
    // 1 : 0.5d
    // 2 : 1d
    // 3 : 2d
    // 4 : 4d
    number: 0
  }, res => {
    console.log(JSON.stringify(res))
  })
},
//设置通讯协议
setProtocol() {
  module.setProtocol({
    //0 : 七华(A 版)
    // 1 : 顶尖 0S2
    // 2 : 大华 RS232-1 连续式
    // 3 : 大华 RS232-1 TZQ 连续式
    //4 : 大华问答式
    // 5 : 迪宝
    number: 0
  }, res => {
    console.log(JSON.stringify(res))
  })
},
//设置最二量程
setMaxRangeCao() {
  module.setMaxRangeCao({
    //整数重量
    weight: 20
  }, res => {
    console.log(JSON.stringify(res))
  })
}

```

```
},
//设置第一量程
setFirstRange() {
  module.setFirstRange({
    //整数重量
    weight: 10
  }, res => {
    console.log(JSON.stringify(res))
  })
},
//设置称重模式
setDecimalNumber() {
  module.setDecimalNumber({
    //小数点位数, 0 : 0
    //      1 : 0.0
    //      2 : 0.00
    //      3 : 0.000
    //      4 : 0.0000
    number: 0
  }, res => {
    console.log(JSON.stringify(res))
  })
},
//设置称重模式
setModel() {
  module.setModel({
    //模式, 0 : 单精度, 1 : 双分度值
    model: 0
  }, res => {
    console.log(JSON.stringify(res))
  })
},
//获取内码
getISN() {
  module.getISN(res => {
    console.log(JSON.stringify(res))
  })
},
//获取AD版本号
getADVersion() {
  module.getADVersion(res => {
    console.log(JSON.stringify(res))
  })
},
//获取称重库版本号
getVersion() {
  module.getVersion(res => {
    console.log(JSON.stringify(res))
  })
},
```

```

//获取扣重值
getTare() {
  module.getTare(res => {
    console.log(JSON.stringify(res))
  })
},
//获取重量
getWeight() {
  module.getWeight(res => {
    console.log(JSON.stringify(res))
  })
},
//扣重
tare() {
  module.tare(res => {
    console.log(JSON.stringify(res))
  })
},
//强制归零
forceZero() {
  module.forceZero(res => {
    console.log(JSON.stringify(res))
  })
},
//归零
zero() {
  module.zero(res => {
    console.log(JSON.stringify(res))
  })
},
//初始化
init() {
  module.init({
    //端口号, 默认: ttyS7
    port: "ttyS7",
    //波特率, 默认: 9600
    baudRate: 9600
  }, res => {
    console.log(JSON.stringify(res))
  })
}
}
}
</script>

<style>
</style>

```

更新日志

2025-02-27

首次发布

初始化

方法名

init

用法

- 用法如下：

```
module.init({
  //端口号，默认：ttyS7
  port: "ttyS7",
  //波特率，默认：9600
  baudRate: 9600
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
port	String	否	ttyS7	端口号
baudRate	Integer	否	9600	波特率

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示

初始化

参数名	参数类型	参数描述
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

归零

方法名

zero

用法

- 用法如下：

```
module.zero(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

强制归零

方法名

forceZero

用法

- 用法如下：

```
module.forceZero(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

扣重

方法名

tare

用法

- 用法如下：

```
module.tare(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

获取重量

方法名

getWeight

用法

- 用法如下：

```
module.getWeight(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {
    "weight": 10
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.weight	String	重量值
code	Integer	返回类型，0.成功，其他：失败

获取扣重值

方法名

getTare

用法

- 用法如下：

```
module.getTare(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {
    "tare": 10
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.tare	String	重量值
code	Integer	返回类型，0.成功，其他：失败

获取秤重库版本号

方法名

getVersion

用法

- 用法如下：

```
module.getVersion(res => {  
  console.log(res)  
});
```

- 参数说明

无

回调

- 示例

```
{  
  "data": {  
    "version": "1.0"  
  },  
  "message": "",  
  "code": 0  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.version	String	版本号
code	Integer	返回类型，0.成功，其他：失败

获取AD版本号

方法名

getADVersion

用法

- 用法如下：

```
module.getADVersion(res => {  
  console.log(res)  
});
```

- 参数说明

无

回调

- 示例

```
{  
  "data": {  
    "adVersion": "1.0"  
  },  
  "message": "",  
  "code": 0  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.adVersion	String	版本号
code	Integer	返回类型，0.成功，其他：失败

获取内码

方法名

getISN

用法

- 用法如下：

```
module.getISN(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {
    "isn": "abc"
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.isn	String	内码
code	Integer	返回类型，0.成功，其他：失败

设置秤重模式

方法名

setModel

用法

- 用法如下：

```
module.setModel({
  //模式, 0 : 单精度, 1 : 双分度值
  model: 0
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
model	Integer	是	无	模式, 0: 单精度, 1: 双分度值

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象

设置秤重模式

参数名	参数类型	参数描述
code	Integer	返回类型，0.成功，其他：失败

设置小数点位数

方法名

setDecimalNumber

用法

- 用法如下：

```
module.setDecimalNumber({
  //小数点位数，0 : 0
  //    1 : 0.0
  //    2 : 0.00
  //    3 : 0.000
  //    4 : 0.0000
  number: 0
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
number	Integer	是	无	小数点位数，取值范围：[0,4]，0 : 0，1 : 0.0，2 : 0.00，3 : 0.000，4 : 0.0000

回调

- 示例

```
{
  "data": {},
  "message": "",
}
```

设置小数点位数

```
    "code": 0;  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

设置第一量程

方法名

setFirstRange

用法

- 用法如下：

```
module.setFirstRange({
  //整数重量
  weight: 10
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
weight	Integer	是	无	整数重量

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

设置最二量程

方法名

setMaxRangeCao

用法

- 用法如下：

```
module.setMaxRangeCao({
  //整数重量
  weight: 10
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
weight	Integer	是	无	整数重量

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

设置通讯协议

方法名

setProtocol

用法

- 用法如下：

```
module.setProtocol({
  //0 : 七华(A 版)
  // 1 : 顶尖 OS2
  // 2 : 大华 RS232-1 连续式
  // 3 : 大华 RS232-1 TZQ 连续式
  //4 : 大华问答式
  // 5 : 迪宝
  number: 0
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
number	Integer	是	无	协议，取值范围：[0,5]，0：七华(A 版)，1：顶尖 OS2，2：大华 RS232-1 连续式，3：大华 RS232-1 TZQ 连续式，4：大华问答式，5：迪宝

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

设置零点追踪

方法名

setZeroTracking

用法

- 用法如下：

```
module.setZeroTracking({
  // 0 : off
  // 1 : 0.5d
  // 2 : 1d
  // 3 : 2d
  // 4 : 4d
  number: 0
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
number	Integer	是	无	协议，取值范围：[0,4]，0：off，1：0.5d，2：1d，3：2d，4：4d

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

设置零点追踪

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

设置分度值1

方法名

setDiv1

用法

- 用法如下：

```
module.setDiv1({
  // 0 : 1
  // 1 : 2
  // 2 : 5
  // 3 : 10
  // 4 : 20
  // 5 : 50
  // 6 : 100
  number: 0
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
number	Integer	是	无	协议，取值范围：[0,6]，0：1，1：2，2：5，3：10，4：20，5：50，6：100

回调

- 示例

```
{
  "data": {},
  "message": "",
}
```

设置分度值1

```
        "code": 0;
    }
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

设置分度值2

方法名

setDiv2

用法

- 用法如下：

```
module.setDiv2({
  // 0 : 1
  // 1 : 2
  // 2 : 5
  // 3 : 10
  // 4 : 20
  // 5 : 50
  // 6 : 100
  number: 0
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
number	Integer	是	无	协议，取值范围：[0,6]，0：1，1：2，2：5，3：10，4：20，5：50，6：100

回调

- 示例

```
{
  "data": {},
  "message": "",
}
```

设置分度值2

```
        "code": 0;
    }
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

设置半自动归零范围

方法名

setManualZeroRange

用法

- 用法如下：

```
module.setManualZeroRange({
  // 0 : 0%
  // 1 : 2%
  // 2 : 4%
  // 3 : 10%
  // 4 : 20%
  // 5 : 50%
  // 6 : 100%
  number: 0
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
number	Integer	是	无	协议，取值范围：[0,6]，0：0%，1：2%，2：4%，3：10%，4：20%，5：50%，6：100%

回调

- 示例

```
{
  "data": {},
  "message": "",
}
```

设置半自动归零范围

```
        "code": 0;
    }
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

设置开机归零范围

方法名

setAutoZeroRange

用法

- 用法如下：

```
module.setAutoZeroRange({
  // 0 : 0%
  // 1 : 2%
  // 2 : 4%
  // 3 : 10%
  // 4 : 20%
  // 5 : 50%
  // 6 : 100%
  number: 0
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
number	Integer	是	无	协议，取值范围：[0,6]，0：0%，1：2%，2：4%，3：10%，4：20%，5：50%，6：100%

回调

- 示例

```
{
  "data": {},
  "message": "",
}
```

设置开机归零范围

```
        "code": 0;
    }
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

标定零点

方法名

setZeroPoint

用法

- 用法如下：

```
module.setZeroPoint(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

标定重量

- 用法如下：

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
caliValue	String	是	无	标定的重量，必须为字符串数字

- 示例

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

设置预扣重

方法名

preTareWeight

用法

- 用法如下：

```
module.preTareWeight({
  //预扣重的重量
  preTareValue: "5"
}, res => {
  console.log(JSON.stringify(res))
})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
preTareValue	String	是	无	预扣重的重量

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

获取稳定状态

方法名

isStableStatus

用法

- 用法如下：

```
module.isStableStatus(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {
    "status": true
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.status	Boolean	稳定状态
code	Integer	返回类型，0.成功，其他：失败

获取扣重状态

方法名

isTaredStatus

用法

- 用法如下：

```
module.isTaredStatus(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {
    "status": true
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.status	Boolean	扣重状态
code	Integer	返回类型，0.成功，其他：失败

获取零点状态

- 用法如下：

- 参数说明

无

- 示例

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.status	Boolean	零点状态
code	Integer	返回类型，0.成功，其他：失败

获取小数点位数

方法名

getDecimalCount

用法

- 用法如下：

```
module.getDecimalCount(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {
    "number": 0
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.number	Integer	小数点位数
code	Integer	返回类型，0.成功，其他：失败

设置回调

方法名

setCallback

用法

- 用法如下：

```
module.setCallback(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {
    "type": "onWeightUpdate",
    "isStable": true,
    "isTared": true,
    "isZero": true,
    "weight": 5.6,
    "tare": 0
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象

参数名	参数类型	参数描述
data.type	String	类型：onWeightUpdate：重量改变， onCalibrationSwitchEvent：校正按钮被按下
data.isStable	Boolean	稳定状态
data.isTared	Boolean	扣重状态
data.weight	float	当前重量
data.tare	float	扣重
code	Integer	返回类型，0.成功，其他：失败

释放称对象

方法名

deleteScale

用法

- 用法如下：

```
module.deleteScale(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败