**TwinCAT HMI Echart实现显示24小时数据**

|  |  |
| --- | --- |
|  | 作者：范小军  职务：风电部 技术工程师  公司：BECKHOFF中国  邮箱：x.fan@beckhoff.com.cn  日期：2022-12-26 |
| **摘 要：**  显示24小时数据的波形图，两种方式进行显示，plc代码中提供单个的数据，在js代码中进行数据处理，这样显示的数据在HMI界面加载的数据就是从0开始然后再加载，第二种方式是通过plc代码中的数组去实现。 | |
| **附 件：**   |  |  |  | | --- | --- | --- | | 序 号 | 文件名 | 备注 | | 1 | Echart显示24小时数据.zip | 例程汇总 | | 2 | TcHmiProject\_Linechart2.zip | 2.2节例程 | | 3 | TcHmiProject\_Linechart3.zip | 2.3节例程 | |  |  |  | | |
| **历史版本：**   |  |  |  | | --- | --- | --- | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | | |
| **免责声明：**  我们已对本文档描述的内容做测试。但是差错在所难免，无法保证绝对正确并完全满足您的使用需求。本文档的内容可能随时更新，如有改动，恕不事先通知，也欢迎您提出改进建议。 | |
| **参考信息：** | |

目 录

[1. 方式一：通过js处理数据然后显示 3](#_Toc126913250)

[2. 方式二：通过plc代码实现 6](#_Toc126913251)

[2.1. PLC代码处理 6](#_Toc126913252)

[2.2. Js代码通过计时器实现 7](#_Toc126913253)

[2.3. Js代码通过watch实现 9](#_Toc126913254)

# 方式一：通过js处理数据然后显示

通过js语言实现，将plc采集的数据通过数组的移位进行处理，首先移位然后再进行数据的压栈操作实习数据的动态显示。

var chartDom = document.getElementById('TcHmiContainer');

var myChart = echarts.init(chartDom);

var option;

var PLCvalue2;

var PLCvalue1;

var Xname = [];

var data = [];

var data1 = [];

var counter = 0;

var dataname;

for (var i = 0; i < 1440; i++) { //初始化x轴的数据，可以根据实际情况设置1440数值的大小设置x轴坐标显示的。

var d = new Date();

Xname[i] = d.getHours() + ':' + d.getMinutes();

data[i] = 0;//sy;ValMath.random() \* 21;

data1[i] = 0;

}

var symbol = new TcHmi.Symbol('%s%PLC1.MAIN.Power1%/s%');

var destroySymbol = symbol.watch(function (data) {

if (data.error === TcHmi.Errors.NONE) {

// Handle result value...

PLCvalue1 = data.value;

// console.log(value);

} else {

// Handle error...

}

// Stop watch inline

// data.destroy(); // Call the destroy function inline to stop the watch and free resources.

});

var symbol = new TcHmi.Symbol('%s%PLC1.MAIN.Power2%/s%');

var destroySymbol = symbol.watch(function (data) {

if (data.error === TcHmi.Errors.NONE) {

// Handle result value...

PLCvalue2 = data.value;

// console.log(value);

} else {

// Handle error...

}

// Stop watch inline

// data.destroy(); // Call the destroy function inline to stop the watch and free resources.

});

option = {

title: {

text: '功率曲线'

},

tooltip: {

trigger: 'axis'

},

legend: {

data: ['功率1', '功率2']

},

grid: {

left: '3%',

right: '4%',

bottom: '3%',

containLabel: true

},

//toolbox: {

// feature: {

// saveAsImage: {}

// }

//},

dataZoom: [

{

show: true,

realtime: true,

start: 80,

end: 100

},

{

type: 'inside',

realtime: true,

start: 65,

end: 85

}

],

xAxis: {

type: 'category',

boundaryGap: false,

data: Xname

},

yAxis: {

type: 'value'

},

series: [

{

name: '功率1',

type: 'line',

//stack: 'Total',

data: data,

smooth: true

},

{

name: '功率2',

type: 'line',

//stack: 'Total',

data: data1,

smooth: true,

}

]

};

intervalId = setInterval(function () {

var d = new Date();

Xname.shift();

Xname.push(d.getHours() + ':' + d.getMinutes());

data.shift();

data.push(PLCvalue1);

data1.shift();

data1.push(PLCvalue2);

myChart.setOption({

series: [

{

data: data

},

{

data: data1

}

]

});

// myChart.setOption(option);

}, 1000);

图片包含 形状

描述已自动生成

# 方式二：通过plc代码实现

在PLC中将数据进行处理，然后直接再Echart图表中显示。

## PLC代码处理

PROGRAM MAIN

VAR

Power1 :LREAL;

Power2 :LREAL;

bBool AT%Q\* :BOOL;

counter :INT ;

yAxis : ARRAY [0..99] OF REAL;

XAxis : ARRAY [0..99] OF STRING;

myGETSYSTEMTIME: GETSYSTEMTIME;

myFileTime : T\_FILETIME;(\*获取格林威治时间后要加上时区的偏移，部分国家还存在夏令时调整\*)

myFileTime\_ul:T\_LARGE\_INTEGER;

myCurTime :Timestruct;

myCurDT : DT;

ul\_8h:T\_LARGE\_INTEGER:=(dwHighPart:=16#43,dwLowPart:=16#0E234000); (\*8 hours time zone\*)

TimeStamp:STRING(60);

TimeStamp1:STRING(20);

TimewHour:STRING(10);

TimewMinute:STRING(10);

TimewSecond:STRING(10);

ton1:Tc2\_Standard.TON;

END\_VAR

TimewHour:=WORD\_TO\_STRING(myCurTime.wHour); //myCurTime WORD\_TO\_STRING

TimewMinute :=WORD\_TO\_STRING(myCurTime.wMinute);

TimewSecond := WORD\_TO\_STRING(myCurTime.wSecond);

TimeStamp1 :=CONCAT(TimewHour,':');

TimeStamp1:=CONCAT(TimeStamp1,TimewMinute);

TimeStamp1:=CONCAT(TimeStamp1,':');

TimeStamp1:=CONCAT(TimeStamp1,TimewSecond);

counter:=counter+1;

Power1 :=Power1 +0.1;

IF Power1 >100 THEN

Power1 :=0;

END\_IF

Power2 :=Power2 +0.2;

IF Power2 >100 THEN

Power2 :=0;

END\_IF

ton1(IN:=TRUE , PT:=T#1S , Q=> , ET=> ); //实现数据的1s移位1次

IF ton1.Q THEN

XAxis[99]:=TimeStamp1;

yAxis[99] :=Power2;

FOR i:=0 TO 98 DO

yAxis[i] :=yAxis[i+1];

XAxis[i] :=XAxis[i+1];

END\_FOR

ton1(IN:=FALSE , PT:=T#1S , Q=> , ET=> );

END\_IF

## Js代码通过计时器实现

var chartDom = document.getElementById('TcHmiContainer\_10');

var myChart = echarts.init(chartDom);

var option;

var Xname = [];

var data1 = [];

//

/// for (var i = 0; i < 100; i++) {

// var d = new Date();

/// Xname[i] = d.getHours() + ':' + d.getMinutes() + ':' + d.getSeconds();

// data[i] = 0;//sy;ValMath.random() \* 21;

// data1[i] = 0;

/// } %s%PLC1.MAIN.XAxis%/s%

//%s%PLC1.MAIN.aPoints%/s%

var symbol = new TcHmi.Symbol('%s%PLC1.MAIN.XAxis%/s%');

var destroySymbol = symbol.watch(function (data) {

if (data.error === TcHmi.Errors.NONE) {

// Handle result value...

Xname = data.value;

// console.log(value);

} else {

// Handle error...

}

// Stop watch inline

// data.destroy(); // Call the destroy function inline to stop the watch and free resources.

});

var symbol = new TcHmi.Symbol('%s%PLC1.MAIN.yAxis%/s%');

var destroySymbol = symbol.watch(function (data) {

if (data.error === TcHmi.Errors.NONE) {

// Handle result value...

data1 = data.value;

// console.log(value);

} else {

// Handle error...

}

// Stop watch inline

// data.destroy(); // Call the destroy function inline to stop the watch and free resources.

});

option = {

title: {

text: '风速（m/s）'

},

tooltip: {

trigger: 'axis'

},

grid: {

left: '2%',

right: '2%',

bottom: '0%',

containLabel: true

},

xAxis: {

type: 'category',

boundaryGap: false,

data: Xname

},

yAxis: {

min:0, //取0为最小刻度

max: 100, //取100为最大刻度

type: 'value'

},

series: [

{

name: '功率1',

type: 'line',

//stack: 'Total',

data: data1,

smooth: true

},

]

};

intervalId = setInterval(function () {

myChart.setOption({

xAxis: {

data: Xname

},

series: [

{

data: data1

}

]

});

// myChart.setOption(option);

}, 1000);

option && myChart.setOption(option);

## Js代码通过watch实现

var symbol = new TcHmi.Symbol('%s%PLC1.MAIN.yAxis%/s%');

var destroySymbol = symbol.watch(function (data) {

if (data.error === TcHmi.Errors.NONE) {

// Handle result value...

data1 = data.value;

myChart.setOption({

xAxis: {

data: Xname

},

series: [

{

data: data1

}

]

});

// console.log(value);

} else {

// Handle error...

}

// Stop watch inline

// data.destroy(); // Call the destroy function inline to stop the watch and free resources.

});

具体代码如下：

var chartDom = document.getElementById('TcHmiContainer\_10');

var myChart = echarts.init(chartDom);

var option;

var Xname = [];

var data1 = [];

//

/// for (var i = 0; i < 100; i++) {

// var d = new Date();

/// Xname[i] = d.getHours() + ':' + d.getMinutes() + ':' + d.getSeconds();

// data[i] = 0;//sy;ValMath.random() \* 21;

// data1[i] = 0;

/// } %s%PLC1.MAIN.XAxis%/s%

//%s%PLC1.MAIN.aPoints%/s%

var symbol = new TcHmi.Symbol('%s%PLC1.MAIN.XAxis%/s%');

var destroySymbol = symbol.watch(function (data) {

if (data.error === TcHmi.Errors.NONE) {

// Handle result value...

Xname = data.value;

// console.log(value);

} else {

// Handle error...

}

// Stop watch inline

// data.destroy(); // Call the destroy function inline to stop the watch and free resources.

});

option = {

title: {

text: '风速（m/s）'

},

tooltip: {

trigger: 'axis'

},

grid: {

left: '2%',

right: '2%',

bottom: '0%',

containLabel: true

},

xAxis: {

type: 'category',

boundaryGap: false,

data: Xname

},

yAxis: {

min:0, //取0为最小刻度

max: 100, //取100为最大刻度

type: 'value'

},

series: [

{

name: '功率1',

type: 'line',

//stack: 'Total',

data: data1,

smooth: true

},

]

};

var symbol = new TcHmi.Symbol('%s%PLC1.MAIN.yAxis%/s%');

var destroySymbol = symbol.watch(function (data) {

if (data.error === TcHmi.Errors.NONE) {

// Handle result value...

data1 = data.value;

myChart.setOption({

xAxis: {

data: Xname

},

series: [

{

data: data1

}

]

});

// console.log(value);

} else {

// Handle error...

}

// Stop watch inline

// data.destroy(); // Call the destroy function inline to stop the watch and free resources.

});

// intervalId = setInterval(function () {

// myChart.setOption(option);

// }, 1000);

option && myChart.setOption(option);

**上海（ 中国区总部）**

中国上海市静安区汶水路 299 弄 9号（市北智汇园）

电话: 021-66312666

**北京分公司**

北京市西城区新街口北大街 3 号新街高和大厦 407 室

电话: 010-82200036 邮箱: beijing@beckhoff.com.cn

**广州分公司**

广州市天河区珠江新城珠江东路32号利通广场1303室

电话: 020-38010300/1/2 邮箱: guangzhou@beckhoff.com.cn

**成都分公司**

成都市锦江区东御街18号 百扬大厦2305 室

电话: 028-86202581 邮箱: chengdu@beckhoff.com.cn

|  |  |
| --- | --- |
| 请用微信扫描二维码  通过公众号与技术支持交流 | 倍福官方网站：  https://www.beckhoff.com.cn  在线帮助系统：  https://infosys.beckhoff.com/index\_en.htm |
| 倍福虚拟学院：  https://tr.beckhoff.com.cn/ |
| 招贤纳士：job@beckhoff.com.cn  技术支持：support@beckhoff.com.cn  产品维修：service@beckhoff.com.cn  方案咨询：sales@beckhoff.com.cn |
|  |