

Scripts-Arithmetic



WECON Technology Co., Ltd.

Website: <http://www.we-con.com.cn/en>

Technical Support: liux@we-con.com.cn

Skype: "fewkkj" or "Jason.chen842"

Phone: 86-591-87868869

QQ Group: 465230233

Technical forum: <http://wecon.freeforums.net/>

1. General

Scripts could meet many special requirements. This demo shows the trigger mode scripts and how to calculate 16-bit integer, 32-bit integer and 32-bit floating in scripts.

2. Trigger mode scripts

In this mode, scripts are controlled by bit address; there are five controlling modes, as FIG 1 shows, users could select it according to their requirements.

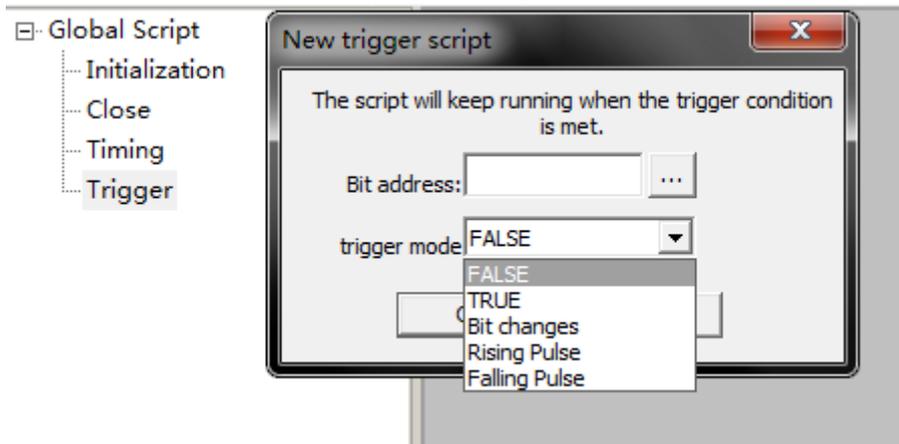


FIG 1

3. 16-bit integer

This is the simplest scripts, address can be used in scripts directly, and there is no need to set any variable. The scripts as FIG2 shows, its working mechanism is, when HDW100.0 has rising pluses, the scripts working.

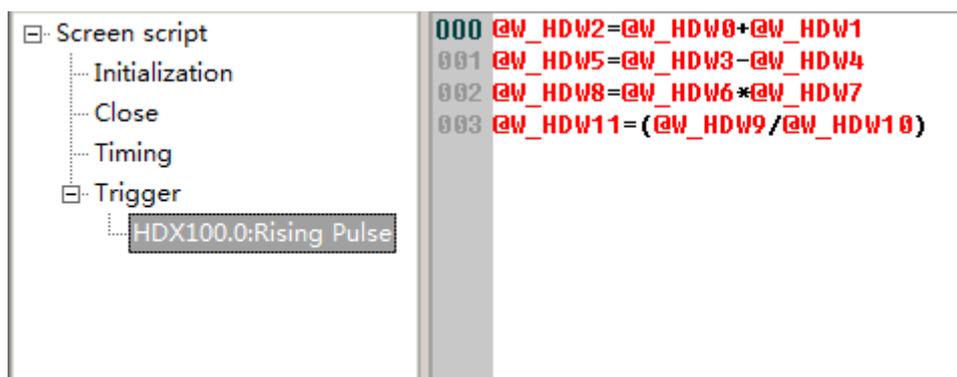


FIG 2

4. 32-bit integer

The scripts can't support 32-bit integer directly, if users want to use 32-bit integer in scripts, please set some variables, the 32-bit integer as FIG 3 shows.

```

000 a1=@w_HDW200+@w_HDW201<<16 'first 32-bit integer
001 a2=@w_HDW202+@w_HDW203<<16 'second 32-bit integer
002 a3=a1+a2 'calculation
003 @w_HDW204=a3 'Results
004 @w_HDW205=a3>>16 'Results
005
006 b1=@w_HDW206+@w_HDW207<<16
007 b2=@w_HDW208+@w_HDW209<<16
008 b3=b1-b2
009 @w_HDW210=b3
010 @w_HDW211=b3>>16
011
012 c1=@w_HDW212+@w_HDW213<<16
013 c2=@w_HDW214+@w_HDW215<<16
014 c3=c1*c2
015 @w_HDW216=c3
016 @w_HDW217=c3>>16
017
018 d1=@w_HDW218+@w_HDW219<<16
019 d2=@w_HDW220+@w_HDW221<<16
020 d3=d1/d2
021 @w_HDW222=d3
022 @w_HDW223=d3>>16
    
```

FIG 3

5. 32-bit floating

If users want to use floating in scripts, there are script functions for floating format.

5.1 Scripts functions

D2Float

Function

F= D2Float("A1",F)

Description

Convert the designated value to floating then assign to variable.

Parameters

A1: Must begin with address"@";

F: Define the floating by self;

Example

dim F as floating

F=D2Float("@W_HDW10",F)

Float2D("@W_HDW12",F)

'define F as floating

'assign the value of (HDW10) to F in floating

'copy the floating value of F to HDW12 register, use to display result.

Float2D

Function

Float2D (A1,A2);

Description

Copy the floating value to the address.

Parameters

A1: Goal address, the value must be address (e.g. @W_HDW102);

A2: Source data, it can be floating;

Example

```
dim f as floating           'define f as floating
f=1.1                      'assign a designated value to f
Float2D("@W_HDW102",f)    'assign the value f to HDW102
```

Result: HDW102=1.1

5.2 Scripts

The scripts for 32-bit floating as FIG 4 shows, users need to default floating variables.

<ul style="list-style-type: none"> Screen script Initialization Close Timing Trigger <ul style="list-style-type: none"> HDX100.2:Rising Pulse 	<pre> 000 dim a1 as floating 'default floating variable 001 dim a2 as floating 'default floating variable 002 dim a3 as floating 'default floating variable 003 a1=D2float("@W_HDW300",a1) 'a1 reads floating value from HDW300 004 a2=D2float("@W_HDW302",a2) 'a2 reads floating value from HDW302 005 a3=a1+a2 'Calculation 006 Float2D("@W_HDW304",a3) 'a3 writes value to HDW304 007 008 dim b1 as floating 009 dim b2 as floating 010 dim b3 as floating 011 b1=D2float("@W_HDW306",b1) 012 b2=D2float("@W_HDW308",b2) 013 b3=b1-b2 014 Float2D("@W_HDW310",b3) 015 016 dim c1 as floating 017 dim c2 as floating 018 dim c3 as floating 019 c1=D2float("@W_HDW312",c1) 020 c2=D2float("@W_HDW314",c2) 021 c3=c1*c2 022 Float2D("@W_HDW316",c3) 023 024 dim d1 as floating 025 dim d2 as floating 026 dim d3 as floating 027 d1=D2float("@W_HDW318",d1) 028 d2=D2float("@W_HDW320",d2) </pre>
--	---

Script run once when bit HDX100.2 is Rising Pulse

6. Project screens

6.1 16-bit integer

16-bit Integer

69	+	426	=	495
269	-	56	=	213
255	*	26	=	6630
23695	/	63	=	376

Calculation

Note: The default format of script is 16-bit integer, so the the data after the decimal point will be ignored. And if data is more than 16-bit, the result will be wrong.

16-bit integer 32-bit integer 32-bit floating

6.2 32-bit integer

32-bit Integer

69553	+	8555669	=	8625222
2556369855	-	125663369	=	2430706486
15566	*	55663	=	866450258
2366523	/	286	=	8274

Calculation

Note: If customers want to use 32-bit data in scripts, we need to do some sepcial settings. Please refer to screen scripts.

16-bit Integer 32-bit Integer 32-bit floating

6.3 32-bit floating

The screenshot displays the WECON logo and the title "32-bit Floating". It features four rows of arithmetic operations, each with numbers in rectangular boxes and operators between them. A yellow "Calculation" button is positioned to the right of the last row. Below the operations is a red note. At the bottom, a navigation bar contains three buttons: "16-bit integer", "32-bit integer", and "32-bit floating", with the latter being highlighted in green.

3668556.25	+	2669853.25	=	6338409.50
2256695.25	-	3655269.25	=	-1398574.00
266.36	*	256956.30	=	68442872.00
236556.30	/	2669.00	=	88.63

Calculation

Note: If customers want to use 32-bit floating in scripts, we need to do default floating variable firstly. Please refer to screen scripts.

16-bit integer 32-bit integer 32-bit floating