

## Segger Jscope波形软件在AT32 MCU的使用

## 前言

这篇应用笔记描述了J-Scope波形软件简单应用。

在实际应用中，我们需要将数据以波形的形式显示。但使用串口上位机、MDK的逻辑分析仪功能等方式都比较复杂。而Segger推出的J-Scope，硬件连接仅需要J-Link，简单易用。

*注：本应用笔记对应的代码是基于雅特力提供的V2.x.x 板级支持包（BSP）而开发，对于其他版本BSP，需要注意使用上的区别。*

支持型号列表：

支持型号	AT32 全系列
------	----------

## 目录

<b>1</b>	<b>J-Scope 简介</b> .....	<b>7</b>
1.1	什么是 J-Scope.....	7
1.2	软/硬件版本 .....	7
1.3	支持的设备 .....	7
1.4	不同版本 JLINK 速度对比 .....	7
<b>2</b>	<b>J-Scope 环境准备</b> .....	<b>9</b>
2.1	J-Scope 安装 .....	9
2.2	安装 AT32 相关 package.....	13
2.2.1	安装 ArteryTek.AT32F403A_407_DFP.2.x.x.pack.....	13
2.2.2	安装 Segger_AT32MCU_AddOn.exe.....	14
<b>3</b>	<b>HSS (High Speed Sampling) 模式</b> .....	<b>18</b>
3.1	HSS 模式简介.....	18
3.2	使用方法.....	18
3.2.1	生成可执行文件 .....	18
3.2.2	新建工程 .....	18
3.2.3	配置工程 .....	19
3.2.4	选择要查看的变量.....	21
3.2.5	开始运行 .....	21
<b>4</b>	<b>RTT (Real Time Transfer) 模式</b> .....	<b>23</b>
4.1	RTT 模式简介 .....	23
4.2	使用方法.....	23
4.2.1	移植 RTT 组件 .....	23
4.2.2	打开 J-Scope, 新建工程.....	24
4.2.3	配置工程 .....	25
4.2.4	开始运行 .....	26
<b>5</b>	<b>J-Scope 软件介绍</b> .....	<b>28</b>

5.1	界面介绍.....	28
5.2	数据导出.....	28
5.3	快捷键.....	30
<b>6</b>	<b>例程.....</b>	<b>31</b>
6.1	HSS 模式.....	31
6.2	RTT 模式.....	31
<b>7</b>	<b>版本历史.....</b>	<b>33</b>

## 表目录

表 1. 数据格式 .....	23
表 2. 文档版本历史 .....	33

## 图目录

图 1. HSS 模式支持的设备 .....	7
图 2. RTT 模式支持的设备 .....	7
图 3. J-Link 采样速度.....	8
图 4. JScope 安装 点击 NEXT.....	9
图 5. JScope 安装 点击 I Agree.....	10
图 6. JScope 安装 点击 NEXT.....	10
图 7. JScope 安装 点击 Browse 选择安装路径.....	11
图 8. JScope 安装选择安装路径.....	12
图 9. JScope 安装 点击 Install.....	12
图 10. JScope 安装 点击 Finish.....	13
图 11. AT32 package 安装 点击 NEXT .....	14
图 12. AT32 package 安装 点击 Finish .....	14
图 13. AT32 package 安装 点击 Browse .....	15
图 14. AT32 package 安装 选择安装路径.....	15
图 15. AT32 package 安装 点击 Start.....	16
图 16. AT32 package 安装 安装成功 .....	16
图 17. AT32 package 安装 安装失败 .....	17
图 18. 新建工程（HSS） .....	18
图 19. 配置工程（HSS） .....	19
图 20. 目标器件选择(HSS).....	20
图 21. 接口、速度选择(HSS).....	20
图 22. 选择可执行文件（HSS） .....	21
图 23. 选择要查看的变量（HSS） .....	21
图 24. 开始运行（HSS） .....	22
图 25. 显示的波形（HSS） .....	22
图 26. RTT 组件(RTT).....	23
图 27. 将 RTT 组件添加到 MDK 工程(RTT).....	23
图 28. 新建工程（RTT） .....	24
图 29. 配置工程（RTT） .....	25

图 30. 目标器件选择 (RTT) .....	26
图 31. 接口/速度选择 (RTT) .....	26
图 32. 开始运行 (RTT) .....	27
图 33. 显示的波形 (RTT) .....	27
图 34. J-Scope 界面介绍 .....	28
图 35. 停止运行 .....	28
图 36. 导出数据 .....	29
图 37. 保存数据文件 .....	29
图 38. 查看数据文件 .....	29
图 39. 快捷键 .....	30
图 40. 应用程序 HSS .....	31
图 41. 应用程序 RTT .....	32

# 1 J-Scope 简介

## 1.1 什么是 J-Scope

J-Scope 是 Segger 推出的一款免费软件，用于 MCU 运行时，实时显示数据的波形。

J-Scope 分为 HSS 和 RTT 两种模式：

- 1) HSS 模式直接使用 J-Scope 加载 MDK 或 IAR 的可执行文件即可，操作简单但采样速度较慢；
- 2) RTT 模式只需要在用户程序里添加 Segger 的 RTT 组件，操作稍麻烦但是采样速度更快更实用。

## 1.2 软/硬件版本

硬件-JLINK：常见的 JLINK BASE 需要 V9 及以上版本；而 JLINK PRO 或 JLINK ULTRA 使用 V4 及以上版本即可。

软件-MDK：MDK4 或 MDK5 皆可。本专题配套的例程需使用 MDK5。

## 1.3 支持的设备

- 1) J-Scope在HSS（High-Speed-Sampling）模式下支持的内核如下：

图 1. HSS 模式支持的设备

Core	HSS
ARM7, ARM9, ARM11	<input type="radio"/>
Cortex-M0	<input checked="" type="checkbox"/>
Cortex-M1	<input checked="" type="checkbox"/>
Cortex-M3	<input checked="" type="checkbox"/>
Cortex-M4	<input checked="" type="checkbox"/>
Cortex-M7	<input checked="" type="checkbox"/>
Cortex-A, Cortex-R	<input type="radio"/>
RX100	<input checked="" type="checkbox"/>
RX200	<input checked="" type="checkbox"/>
RX600	<input checked="" type="checkbox"/>
PIC32	<input type="radio"/>

- 2) J-Scope在RTT（Real-Time Transfer）模式下支持的内核如下：

图 2. RTT 模式支持的设备

Core	RTT
Cortex-M0	<input checked="" type="checkbox"/>
Cortex-M0+	<input checked="" type="checkbox"/>
Cortex-M1	<input checked="" type="checkbox"/>
Cortex-M3	<input checked="" type="checkbox"/>
Cortex-M4	<input checked="" type="checkbox"/>
Cortex-M7	<input checked="" type="checkbox"/>
RX100	<input checked="" type="checkbox"/>
RX200	<input checked="" type="checkbox"/>
RX600	<input checked="" type="checkbox"/>

## 1.4 不同版本 JLINK 速度对比

截图中的数据主要对于HSS模式说的，而RTT模式没有这些限制，基本JLINK速度支持到多大，RTT就可以达到多大。

对于JLINK BASE来说，基本都是截图里面的Other选项，也就是说在HSS模式下支持10个变量采集，采样速度1KHz。

图 3. J-Link 采样速度

J-Link Model	Max. no. of variables	Max. sampling frequency
J-Link PRO (V4 or later)	100	Unlimited <sup>1</sup>
J-Link ULTRA- (V4 or later)	100	Unlimited <sup>1</sup>
Other models	10	1 kHz <sup>2</sup>

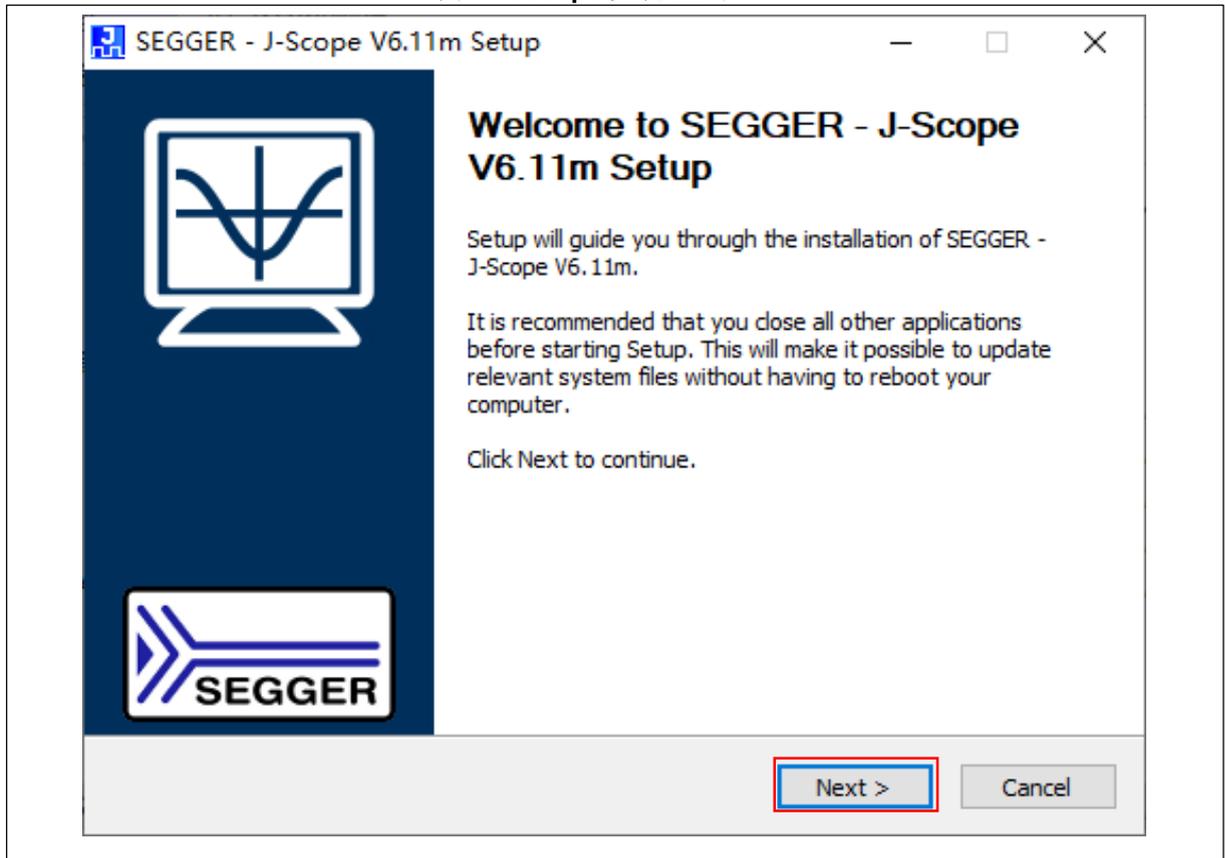
## 2 J-Scope 环境准备

此处以 MDK5 为例，硬件以 AT32F403A 为例。

### 2.1 J-Scope 安装

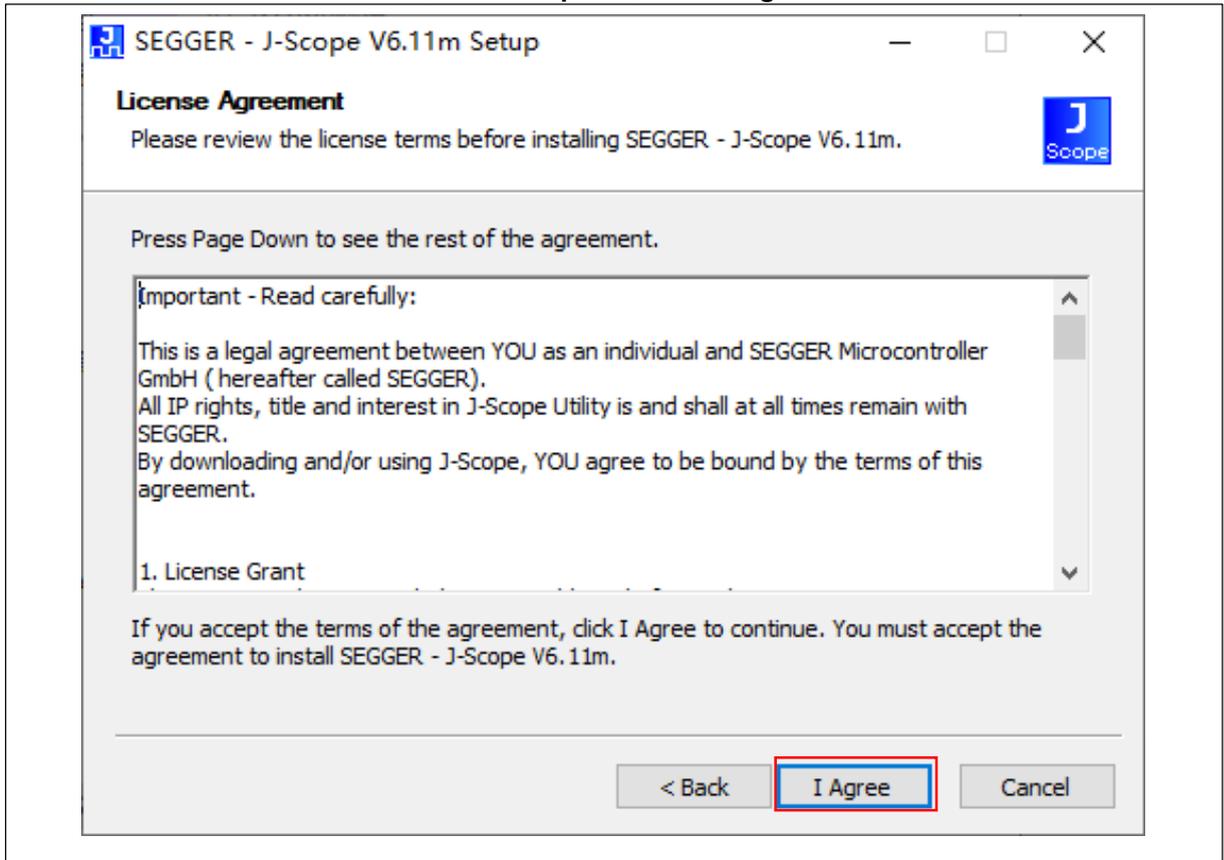
- 1) 解压 xx\AN0044\_Segger\_Jscope\_on\_AT32\_MCU \JScopeTool\Setup\_JScope\_V6xxm.zip;
- 2) 双击 Setup\_JScope\_V6xxm.exe，开始安装。
- 3) 点击点击 NEXT，继续安装。

图 4. JScope 安装 点击 NEXT



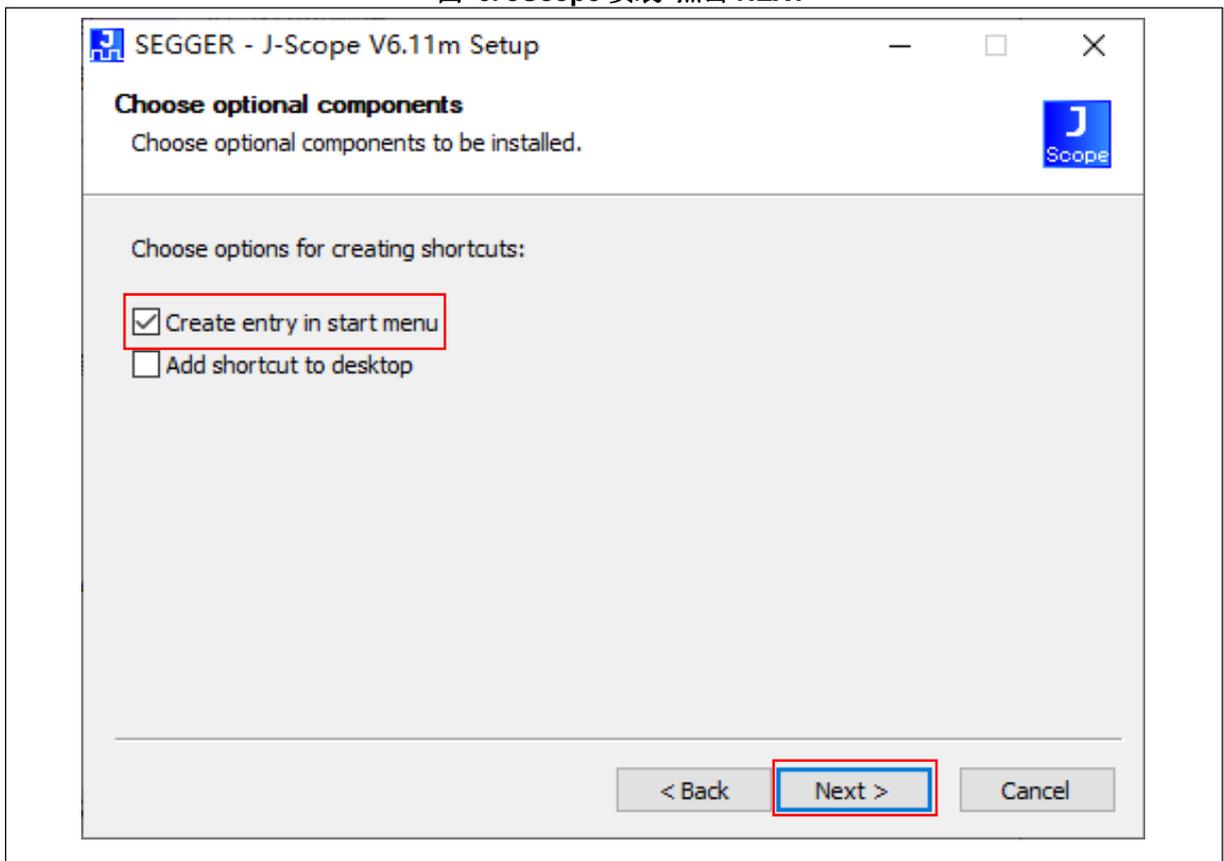
4) 点击 I Agree 同意协议。

图 5. JScope 安装 点击 I Agree



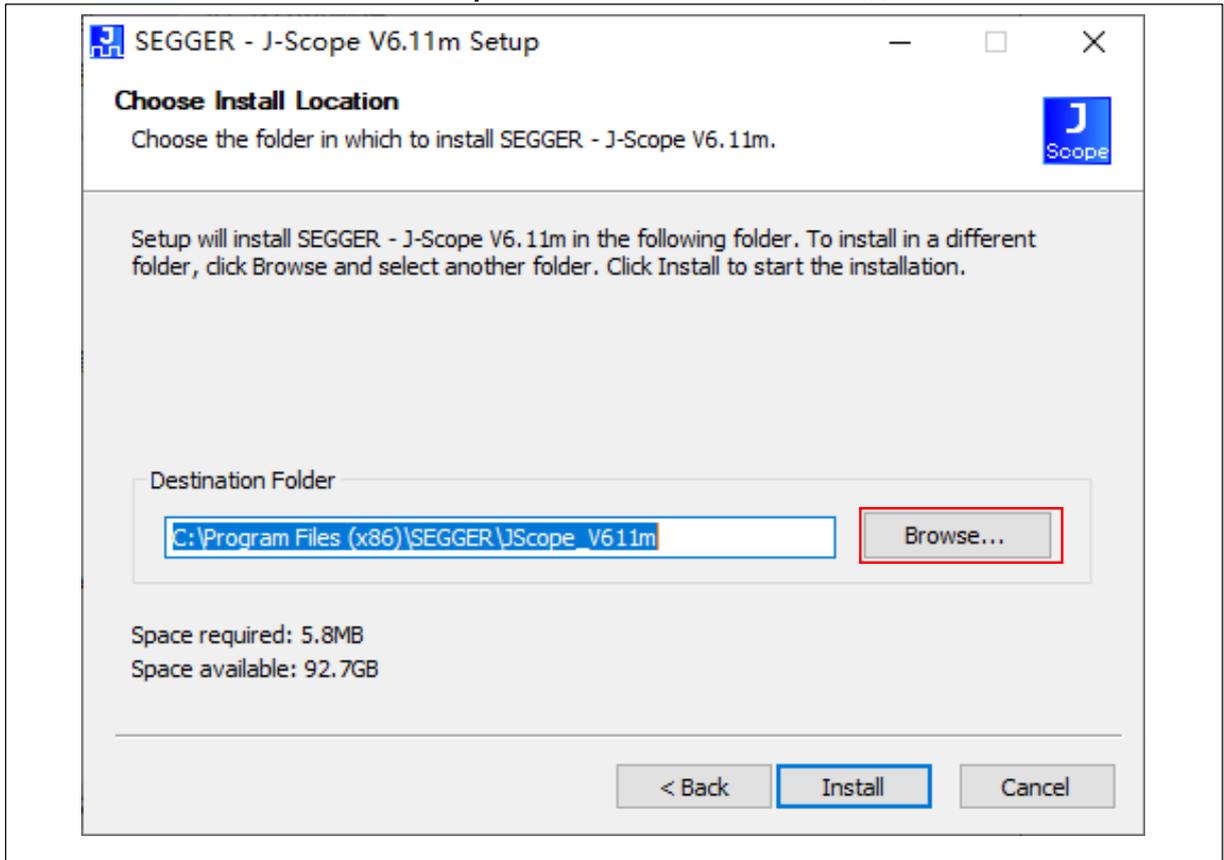
5) 选择快捷方式添加位置，此处选择了添加到开始菜单，然后点击 NEXT 继续安装。

图 6. JScope 安装 点击 NEXT



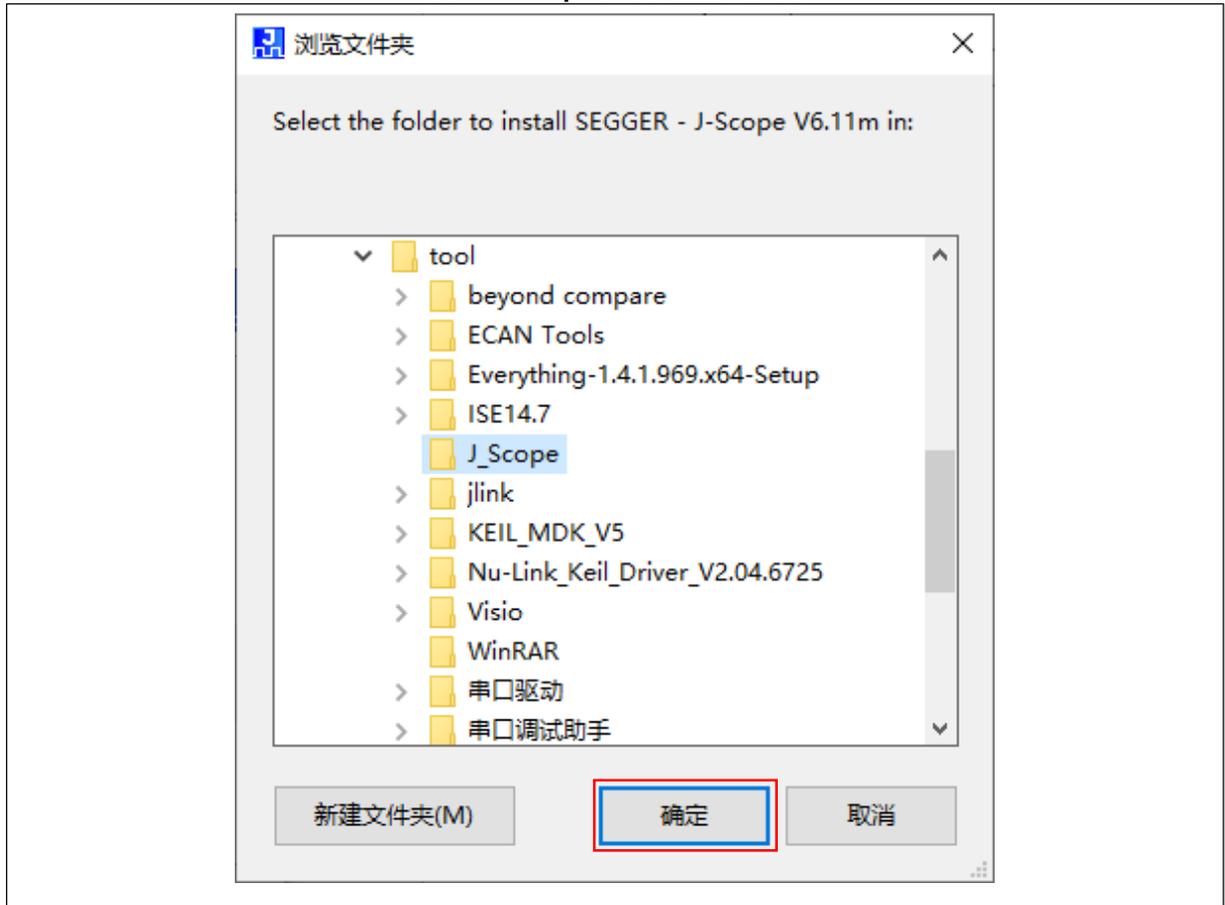
6) 点击 Browse 选择安装路径。

图 7. JScope 安装 点击 Browse 选择安装路径



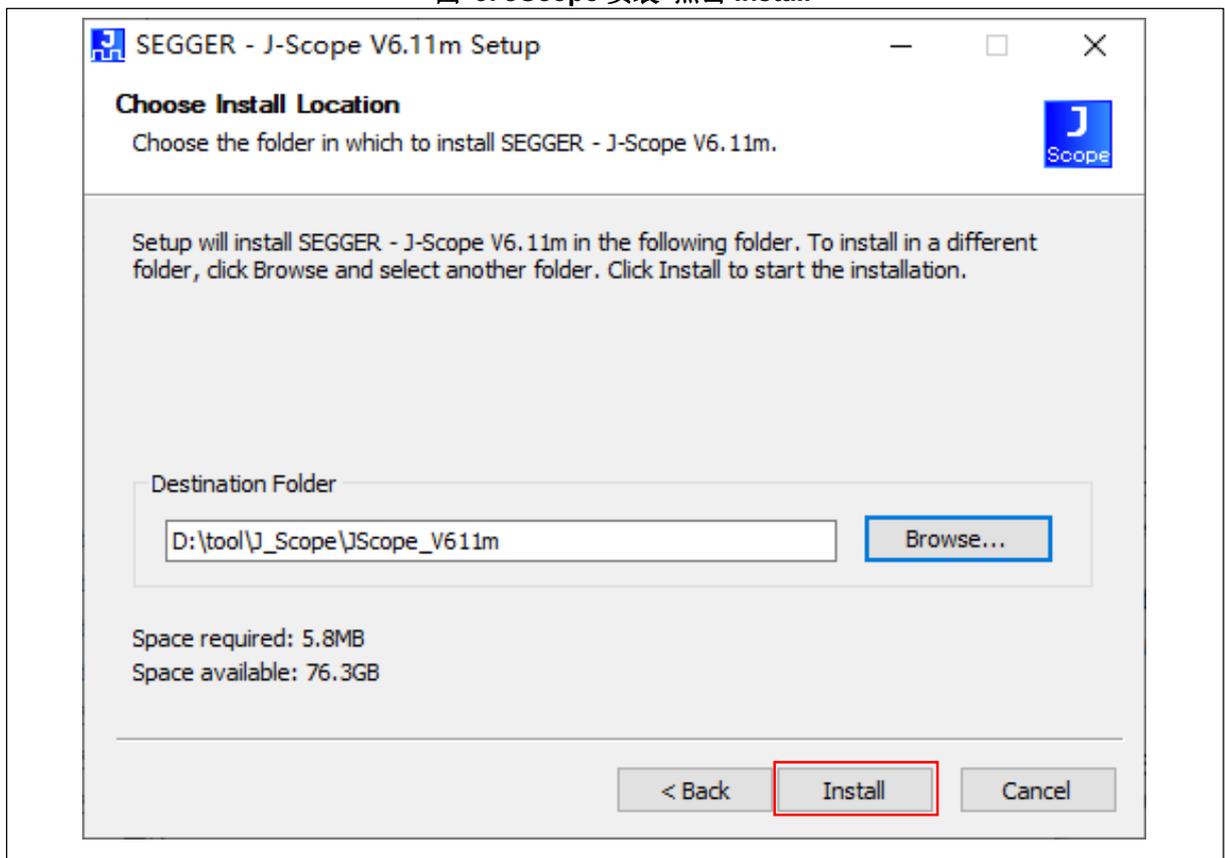
7) 在弹出的界面选择路径，然后点击确认。用户可自行选择安装路径，本示例选择安装在 D:\tool\J\_Scope。

图 8. JScope 安装选择安装路径



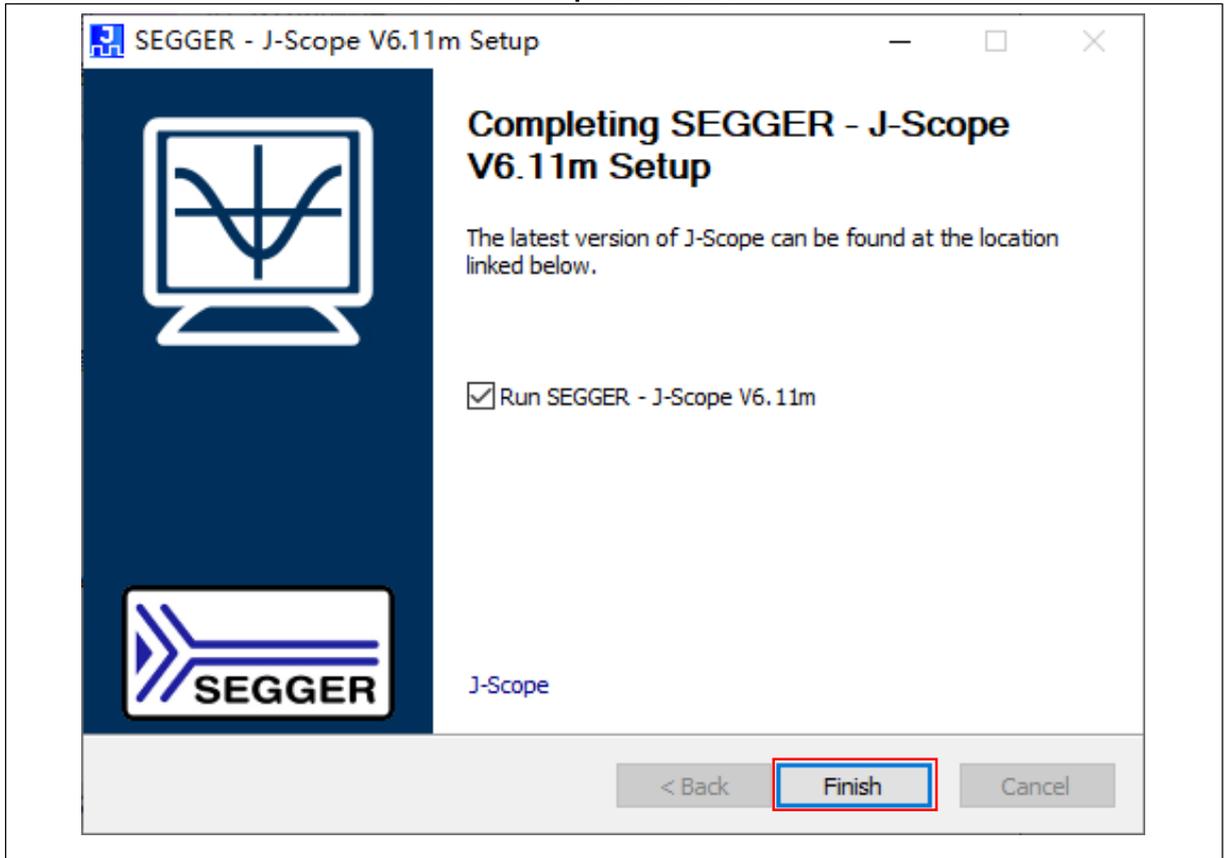
8) 点击 Install 开始安装。

图 9. JScope 安装 点击 Install



9) 点击 Finish 完成安装。

图 10. JScope 安装 点击 Finish



## 2.2 安装 AT32 相关 package

先解压 xx\AN0044\_Segger\_Jscope\_on\_AT32\_MCU\_2.x.x\JScopeTool\PACKs\_V2.x.x.zip;

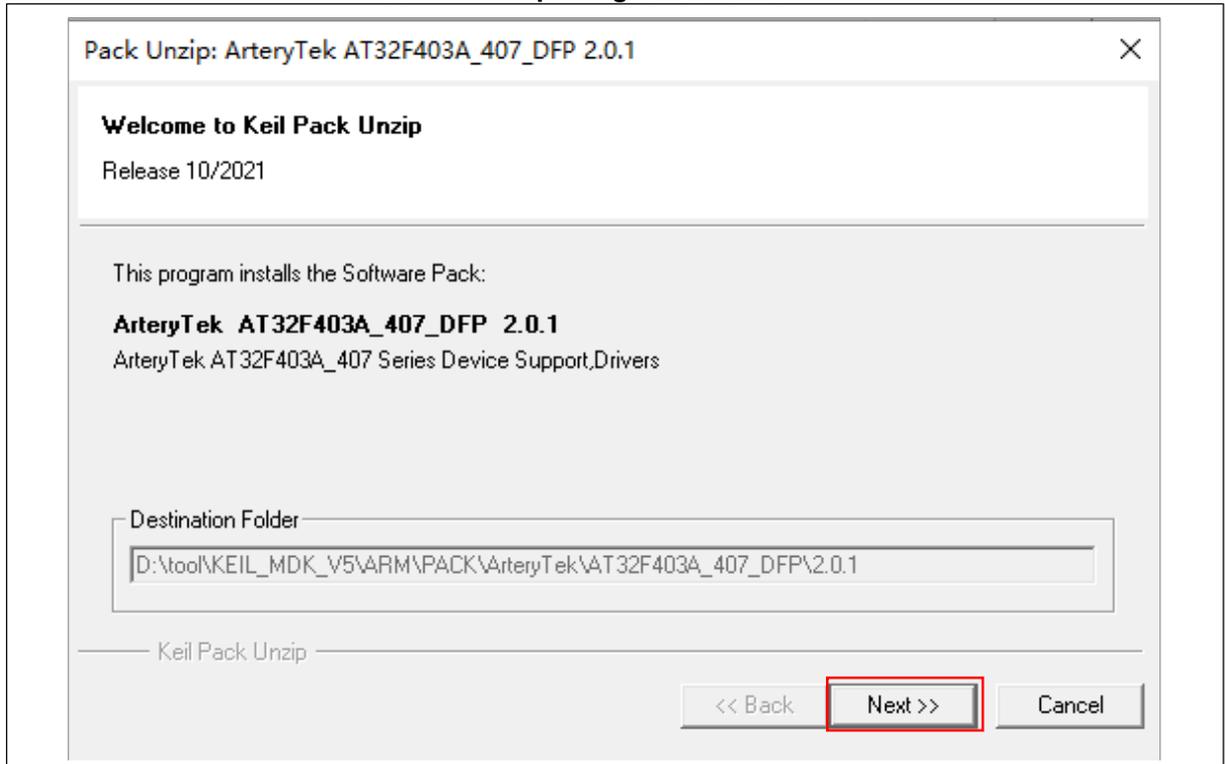
然后再解压 Keil5\_AT32MCU\_AddOn.zip

然后分别安装 ArteryTek.AT32F403A\_407\_DFP.2.x.x.pack, Segger\_AT32MCU\_AddOn.exe。

### 2.2.1 安装 ArteryTek.AT32F403A\_407\_DFP.2.x.x.pack

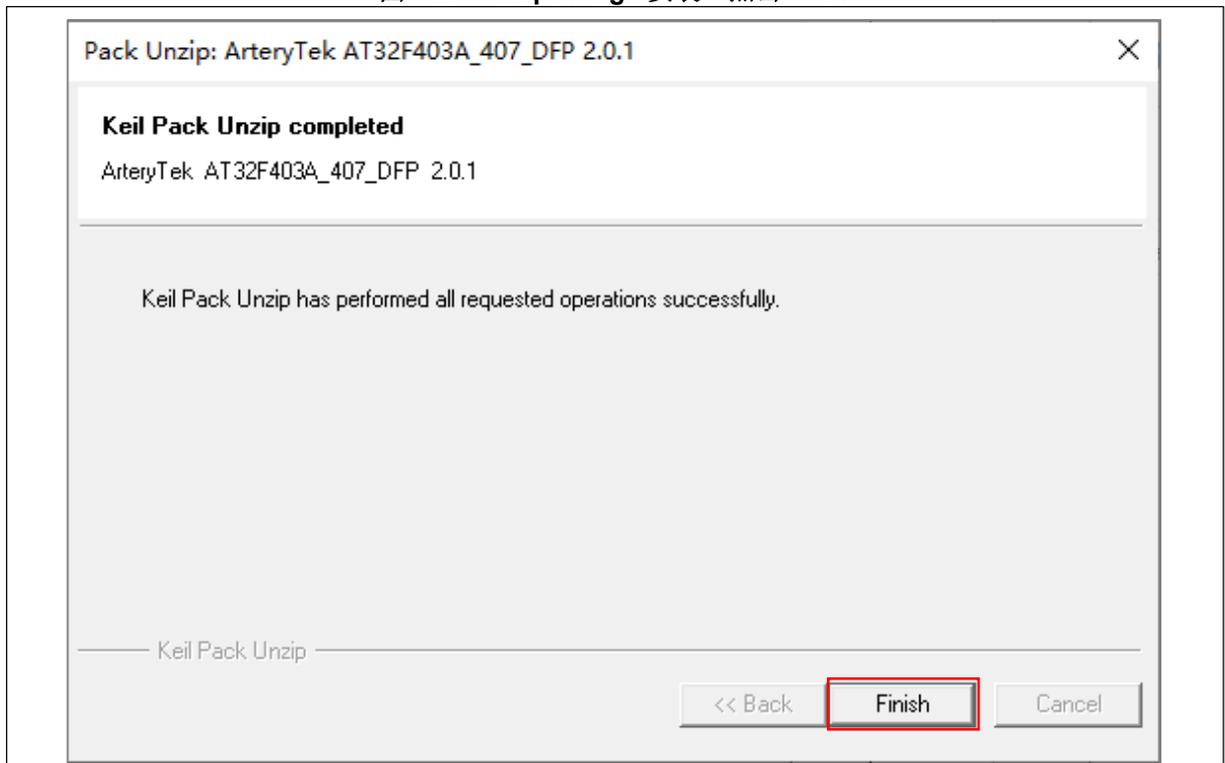
- 1) 双击 ArteryTek.AT32F403A\_407\_DFP.2.x.x.pack 开始安装。
- 2) 点击 NEXT 继续安装。

图 11. AT32 package 安装 点击 NEXT



3) 点击 Finish 完成安装。

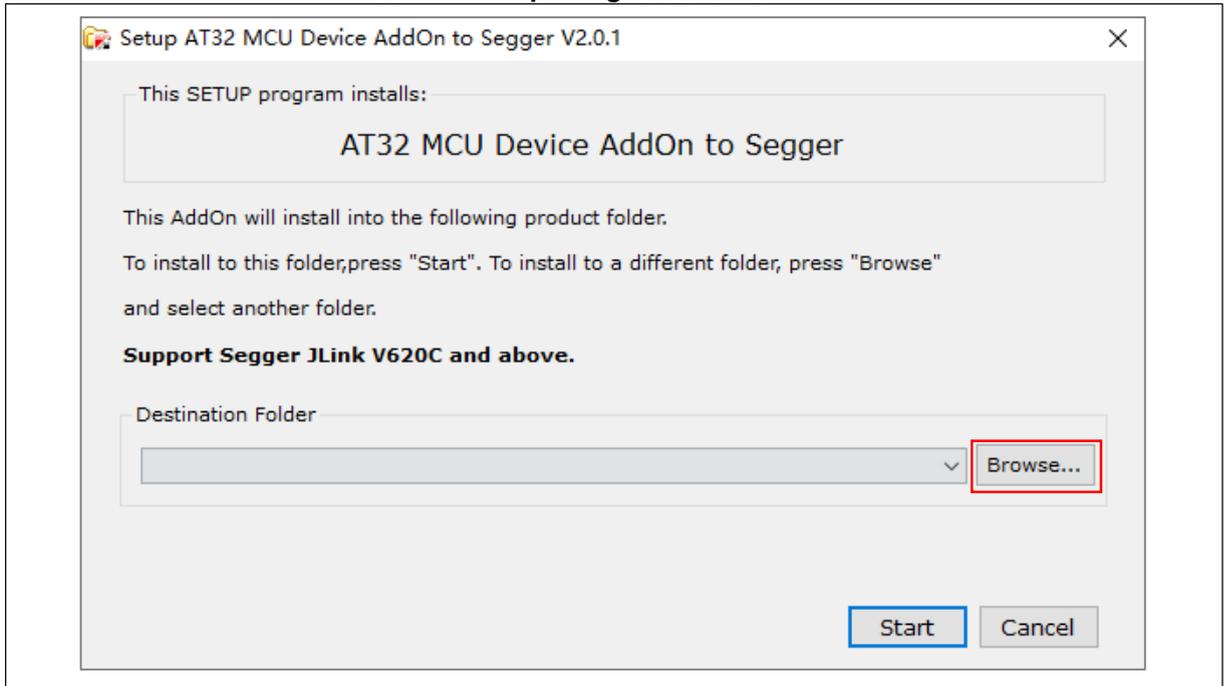
图 12. AT32 package 安装 点击 Finish



## 2.2.2 安装 Segger\_AT32MCU\_AddOn.exe

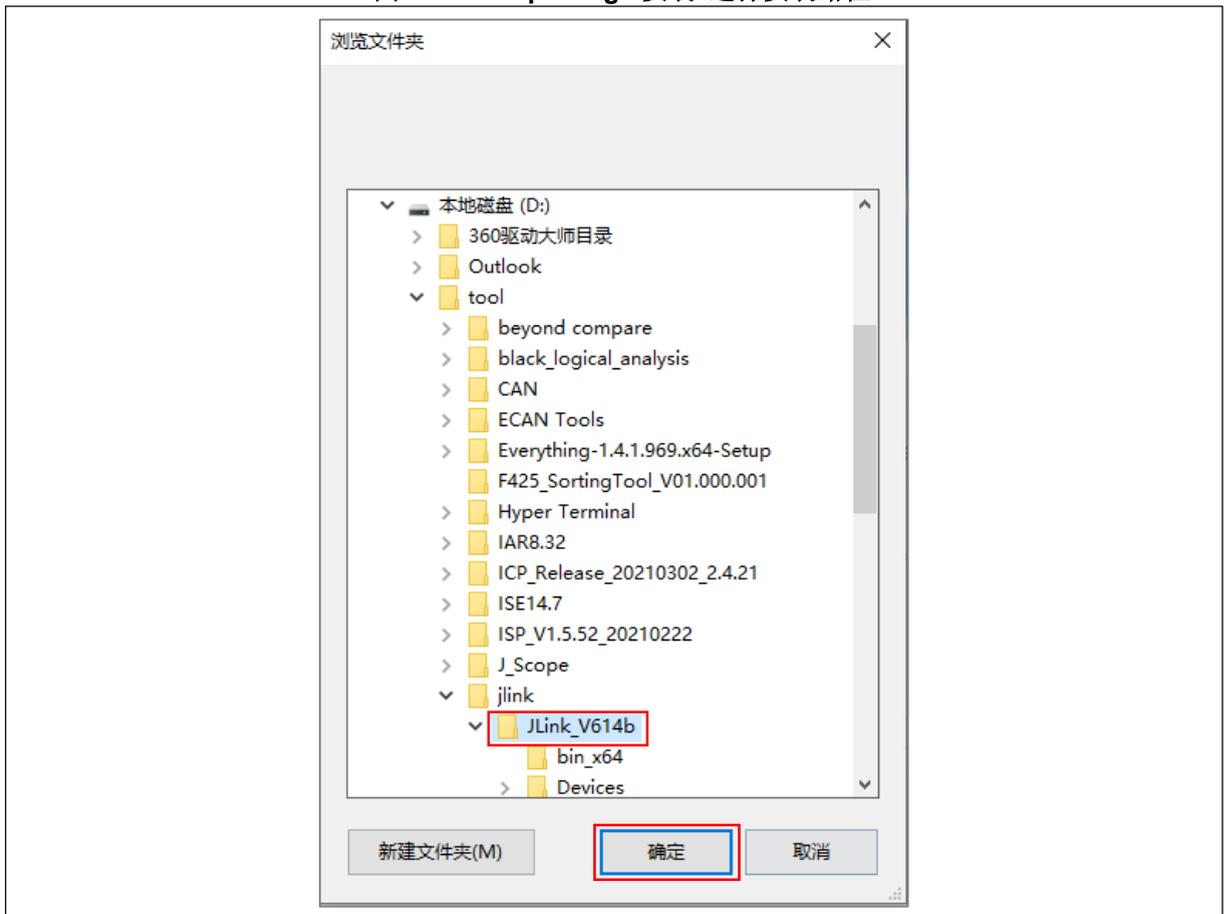
- 1) 双击 Segger\_AT32MCU\_AddOn.exe 开始安装。
- 2) 在弹出界面点击 Browse 以选择安装路径。

图 13. AT32 package 安装 点击 Browse



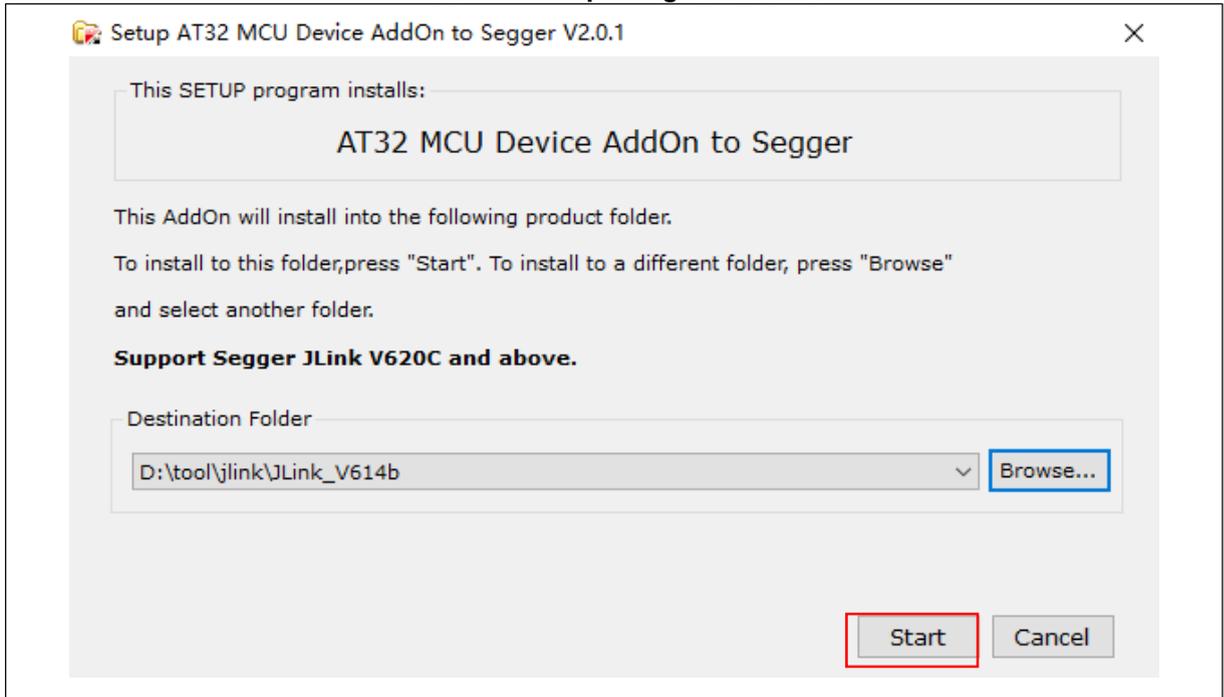
- 3) 选择安装路径：注意，安装路径必须选择 JLink 的安装路径，否则找不到 JLinkDevices.xml，无法安装。

图 14. AT32 package 安装 选择安装路径



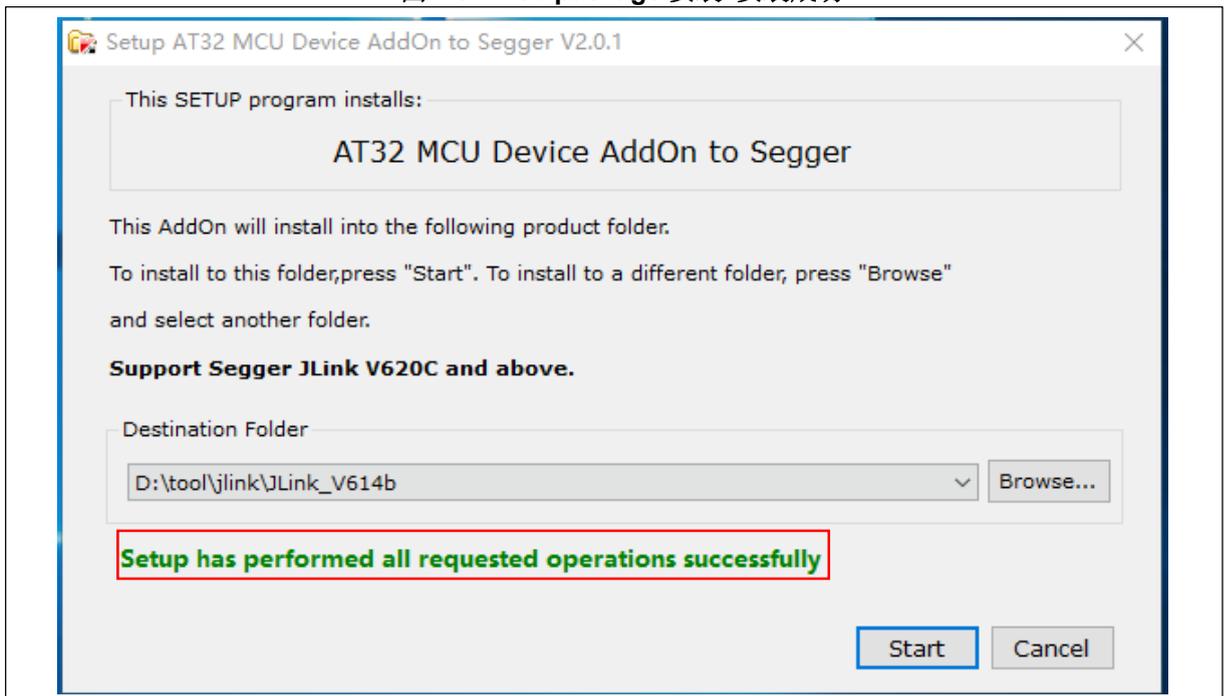
- 4) 点击 Start，开始安装。

图 15. AT32 package 安装 点击 Start



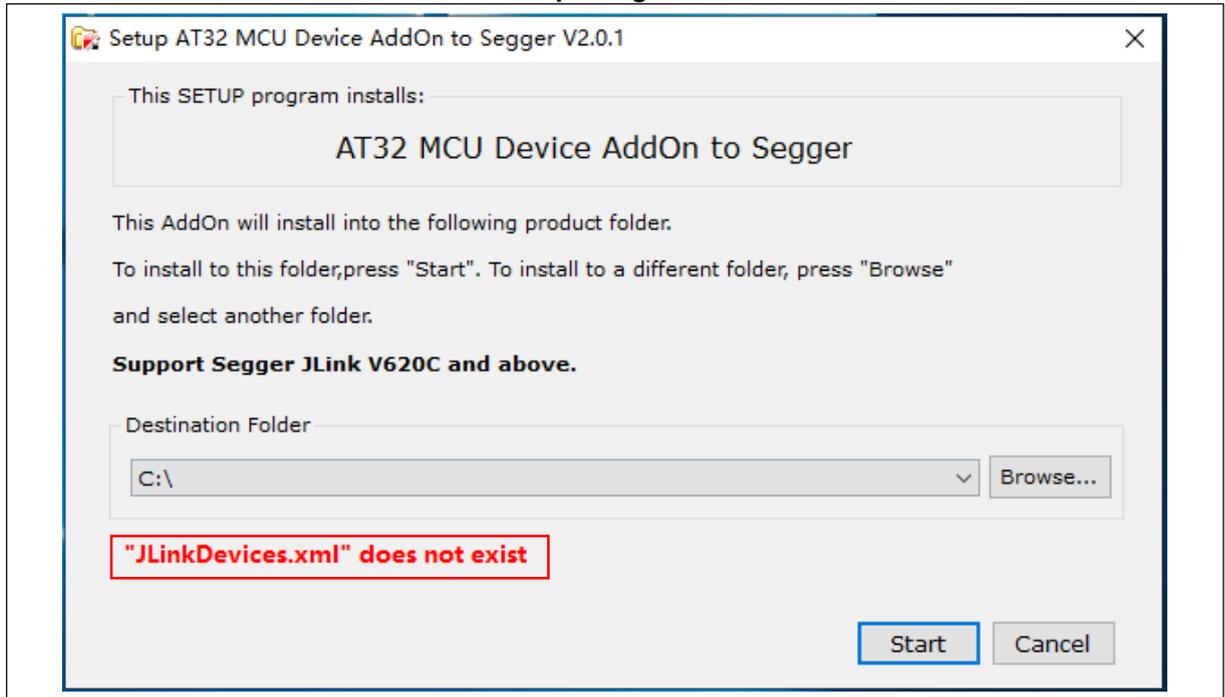
5) 若安装成功，会绿色字体提示安装成功。

图 16. AT32 package 安装 安装成功



6) 若安装失败，则会红色字体提示失败原因：此处是安装路径不对，因此找不到 JLinkDevices.xml 文件。重新安装，正确选择安装路径为 JLink 安装路径即可。

图 17. AT32 package 安装 安装失败



## 3 HSS（High Speed Sampling）模式

### 3.1 HSS 模式简介

HSS 模式比较简单，仅需大家将 MDK 生成的可执行文件 xxx.axf 或者 IAR 生成的可执行文件 xxx.out 文件加载到 JScope 软件里面即可。

硬件连接：使用标准的下载接口（VCC，GND，SWDIO，SWCLK 和 NRST）或三线 JLINK-OB（GND，SWDIO，SWCLK）即可。

优势：随时可以连接目标板，不影响目标板的正常功能，不需要额外资源。无需用户写目标板代码。

劣势：相对 RTT 模式，采样速度慢，基本固定在 1KHZ 左右，仅适用于变量变化速度低于 1KHZ 的情况。

### 3.2 使用方法

此处以 MDK5 为例，硬件以 AT32F403A 为例。

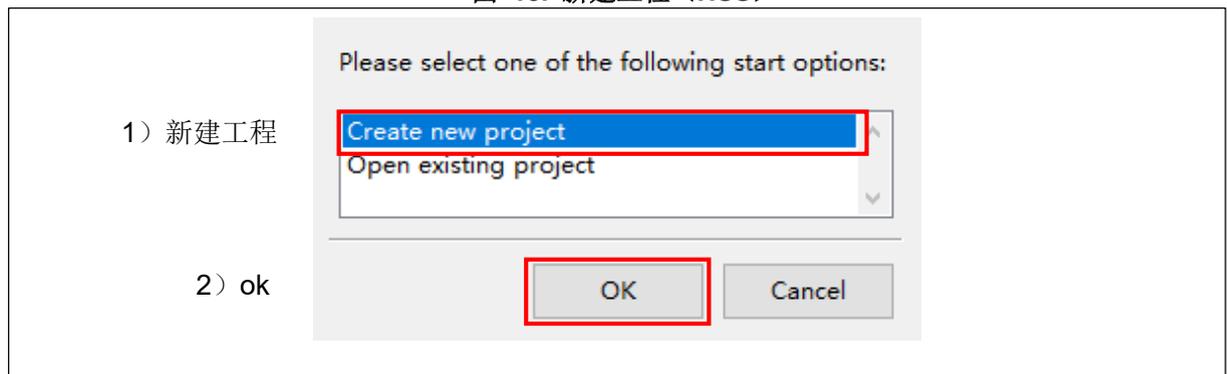
#### 3.2.1 生成可执行文件

打开 MDK 工程，编译、下载程序，并生成可执行文件 HSS.axf。

#### 3.2.2 新建工程

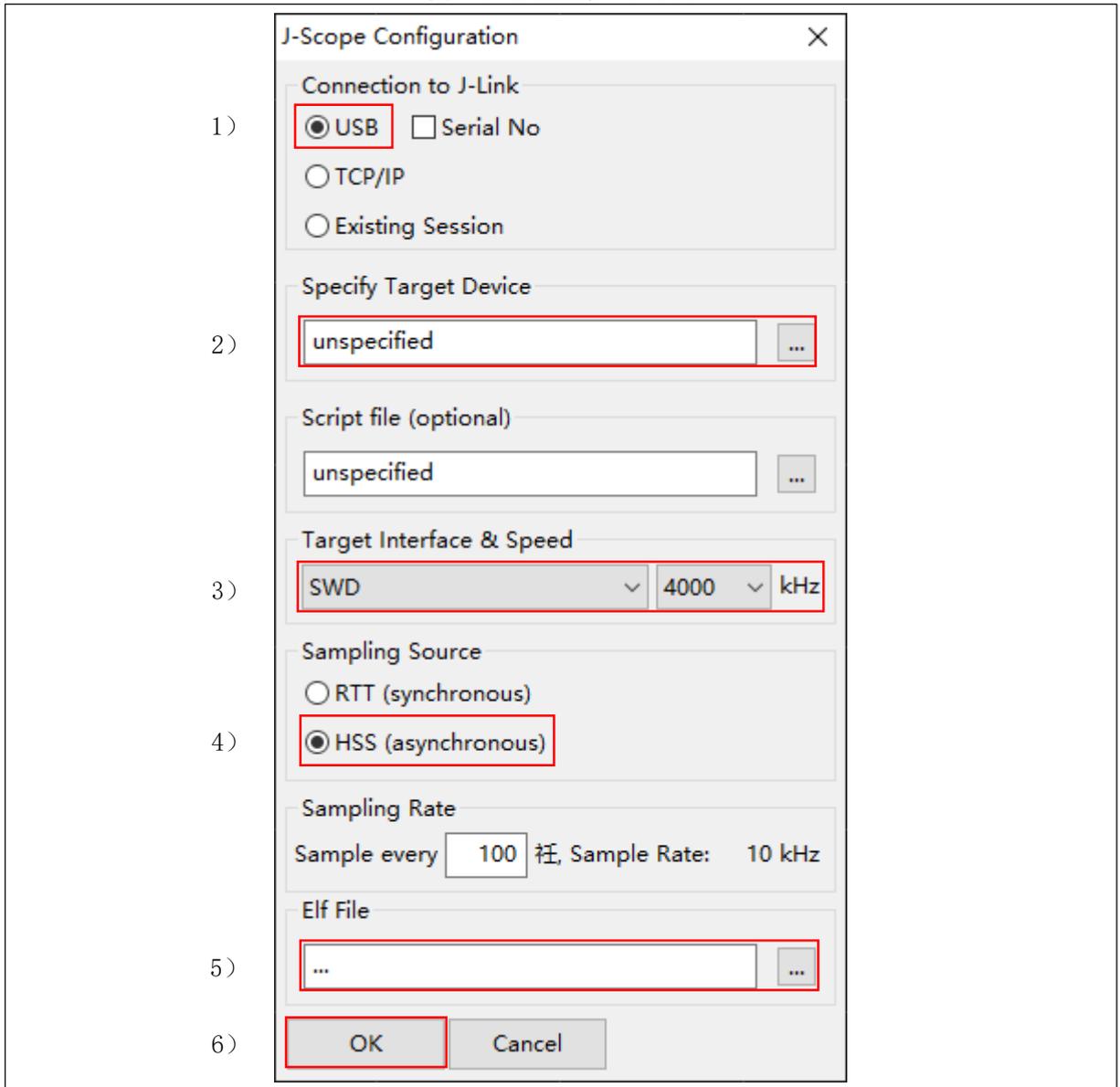
打开 J-Scope 后，新建工程：

图 18. 新建工程（HSS）



## 3.2.3 配置工程

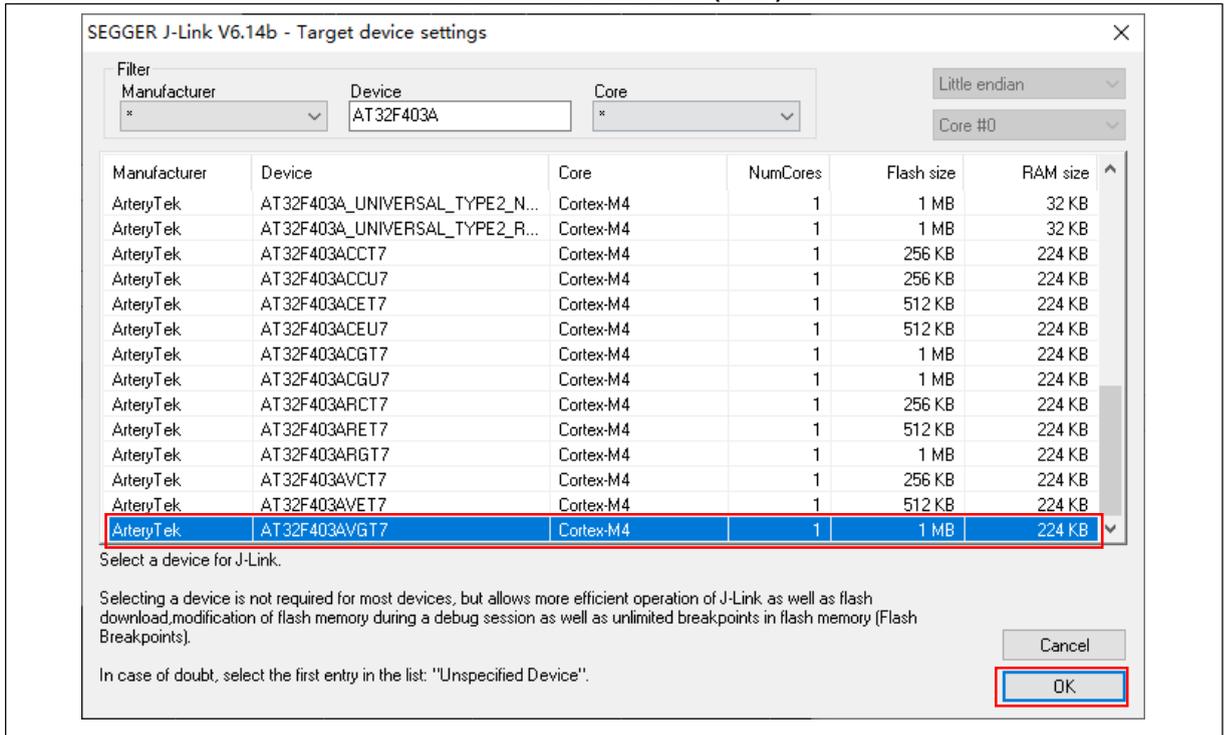
图 19. 配置工程 (HSS)



1) JLINK接口选择：大家通常使用的JLINK都是USB接口，所以此处选择USB。

2) 目标器件选择：此处选择AT32F403AVGT7。

图 20. 目标器件选择(HSS)

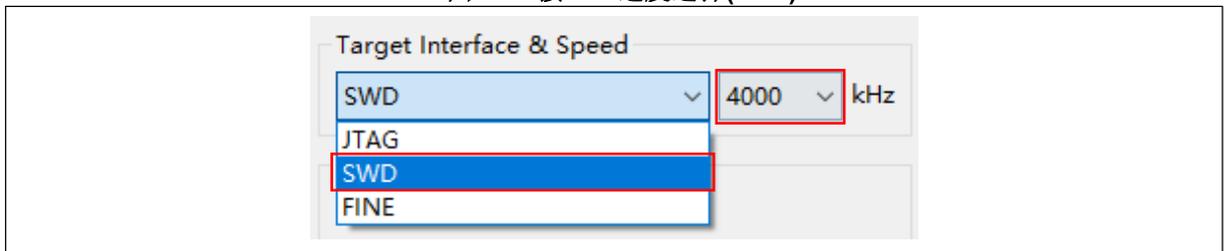


3) 目标接口和速度选择:

目标接口: 根据应用程序配置, 选择SWD/JTAG均可。本例程选择的SWD, 所以这里也选择SWD。

JLINK速度: JLINK速度配置不要太高, 这里选择4000kHz。

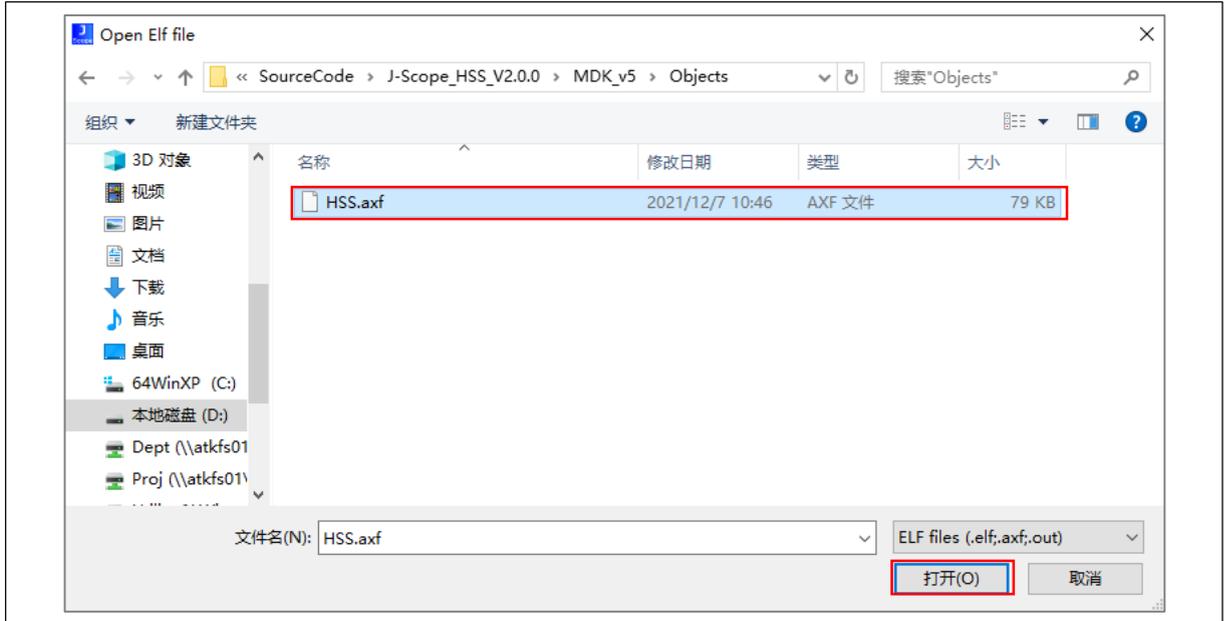
图 21. 接口、速度选择(HSS)



4) 模式选择: 此处选择HSS。

5) 选择可执行文件: 此处MDK例程的路径里找到可执行文件 (HSS.axf)。

图 22. 选择可执行文件（HSS）

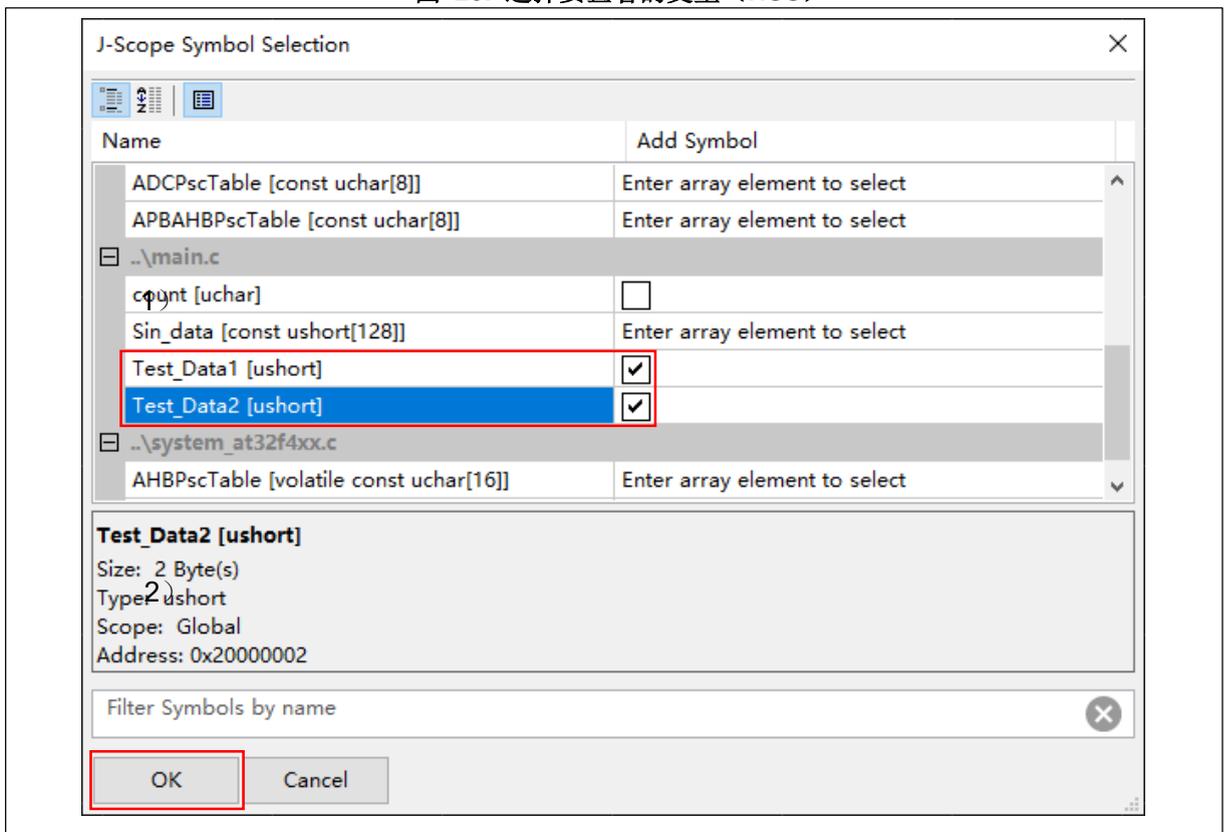


6) 配置完成，点击ok，完成配置。

### 3.2.4 选择要查看的变量

- 1) 打勾选中；
- 2) 点击 ok。

图 23. 选择要查看的变量（HSS）



### 3.2.5 开始运行

- 1) 单击红点，开始显示数据。
- 2) 此时跳出警告，点击OK即可。

图 24. 开始运行 (HSS)

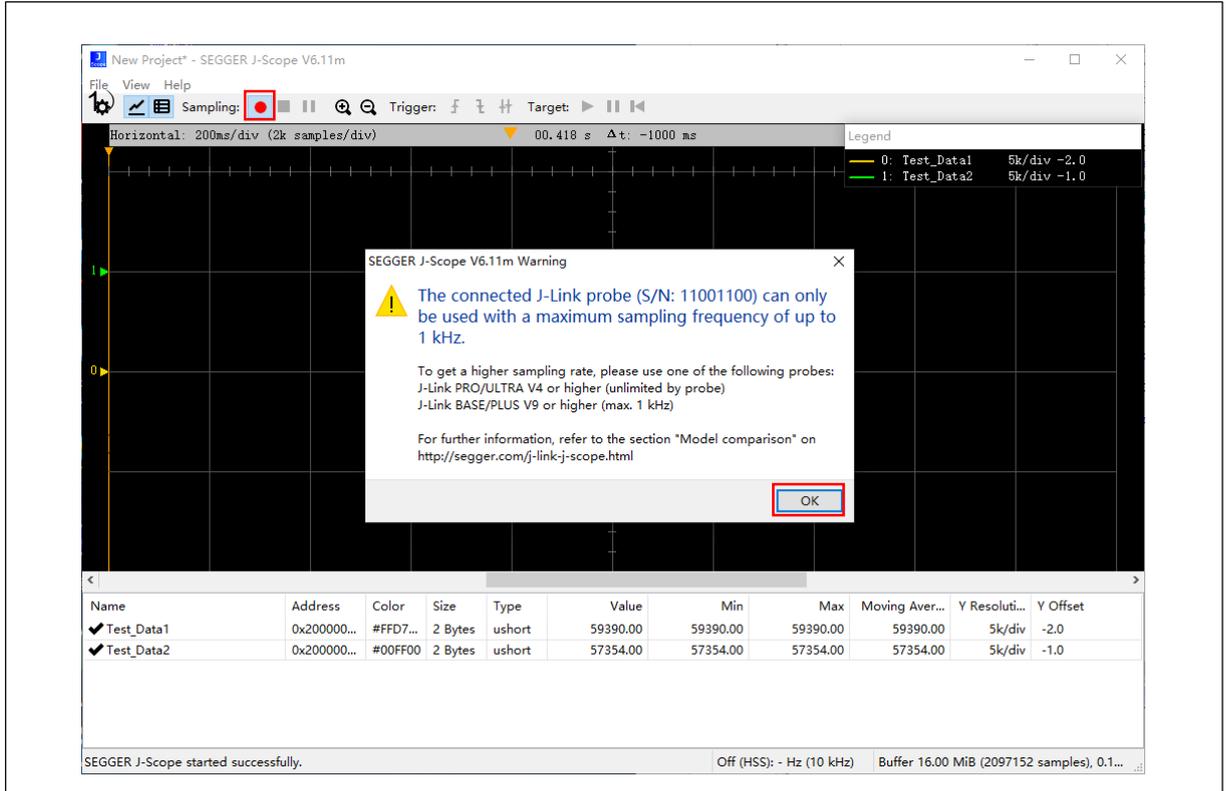
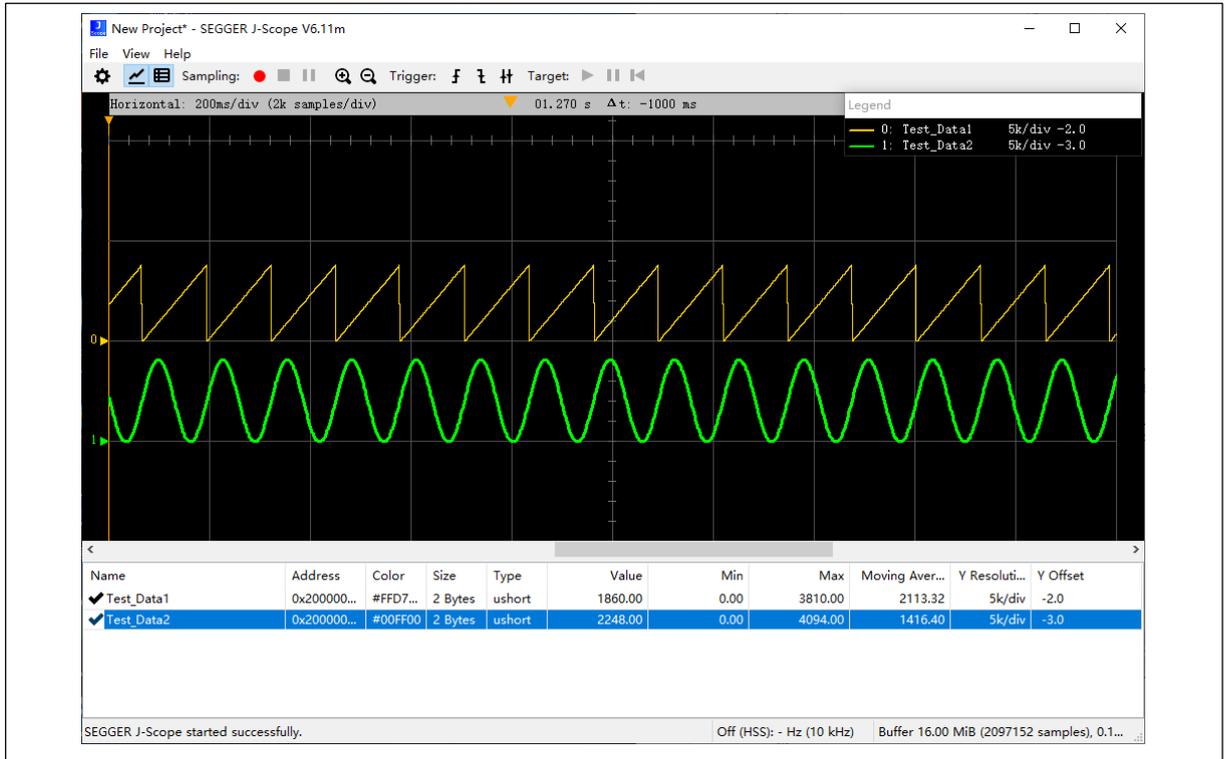


图 25. 显示的波形 (HSS)



## 4 RTT (Real Time Transfer) 模式

### 4.1 RTT 模式简介

RTT模式类似于串口上传数据。需要移植RTT组件,配置上行缓冲区,发送数据等操作。

硬件连接:使用标准的下载接口(VCC, GND, SWDIO, SWCLK和NRST)或三线JLINK-OB(GND, SWDIO, SWCLK)即可。

优势:

- 允许比HSS更高的数据采样速度,高达2 MB/S。即使目标上有512字节的小缓冲区,也可以达到1 MB/S。
- 数据采集与目标板应用程序的执行同步,因为应用程序决定何时采样数据。
- 时间戳等数据可以添加到数据样本中。

劣势:

- 需要移植RTT组件等操作,较HSS模式更麻烦一些。

### 4.2 使用方法

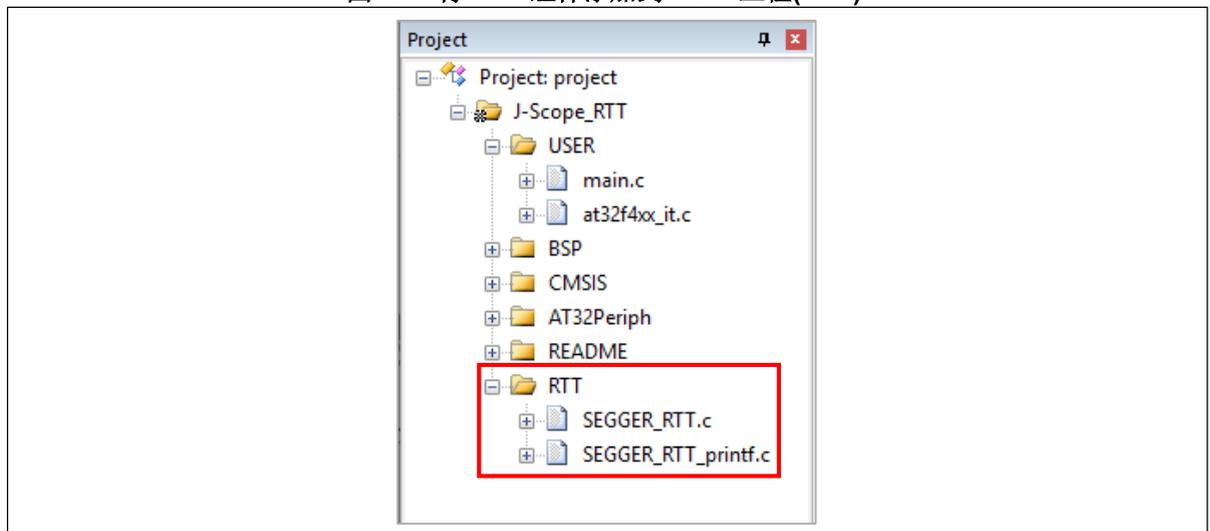
#### 4.2.1 移植 RTT 组件

1) 将以下4个文件复制到MDK工程路径下,并添加进工程。

图 26. RTT 组件(RTT)

SEGGER_RTT	2018/1/22 15:49	C 文件	34 KB
SEGGER_RTT	2018/1/22 15:49	H 文件	11 KB
SEGGER_RTT_Conf	2018/1/22 15:49	H 文件	4 KB
SEGGER_RTT_printf	2018/2/3 1:38	C 文件	17 KB

图 27. 将 RTT 组件添加到 MDK 工程(RTT)



2) 使用下述函数,配置上行缓冲区:

```
SEGGER_RTT_ConfigUpBuffer(1, "JScope_u4u2", buf, 2048, SEGGER_RTT_MODE_NO_BLOCK_SKIP);
```

- 第一个参数: **buffer**索引—此处选择1;
  - 第二个参数: 数据格式—固定格式为JScope\_xy;
- xy支持的格式如下:

表 1. 数据格式

类型(x)	数据大小(y)	描述
t	4	每次发送的数据包前面都将带1个时间戳, 32bit变量, 单位us
i	1, 2, 4	8/16/32bit有符号变量
u	1, 2, 4	8/16/32bit无符号变量

例如：

"JScope\_u4u2"：表示包含32bit无符号数据和16bit无符号数据，且数据顺序按照u4u2的排列顺序。

"JScope\_t4i4u2"：表示包含32bit 时间戳，32bit有符号数据和16bit无符号数据，且数据顺序按照t4i4u2的排列顺序。

- 第三个参数：缓冲区buffer选择；
- 第四个参数：buffer size；
- 第五个参数：模式—使用RTT模式的话，最后一个参数仅支持SEGGER\_RTT\_MODE\_NO\_BLOCK\_SKIP和SEGGER\_RTT\_MODE\_BLOCK\_IF\_FIFO\_FULL。此处选择SEGGER\_RTT\_MODE\_NO\_BLOCK\_SKIP。

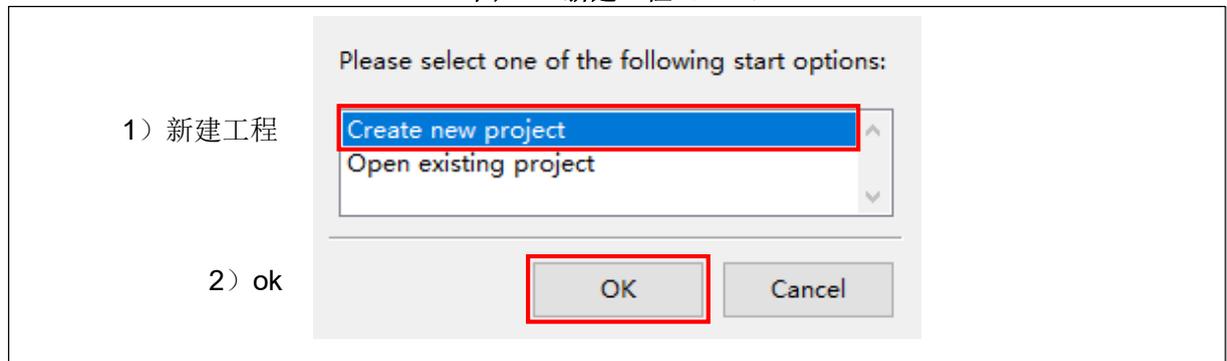
3) 调用SEGGER\_RTT\_Write函数发送数据：

```
SEGGER_RTT_Write(1, &Test_Data1, 4);  
SEGGER_RTT_Write(1, &Test_Data2, 2);
```

4) 编译、下载应用程序。

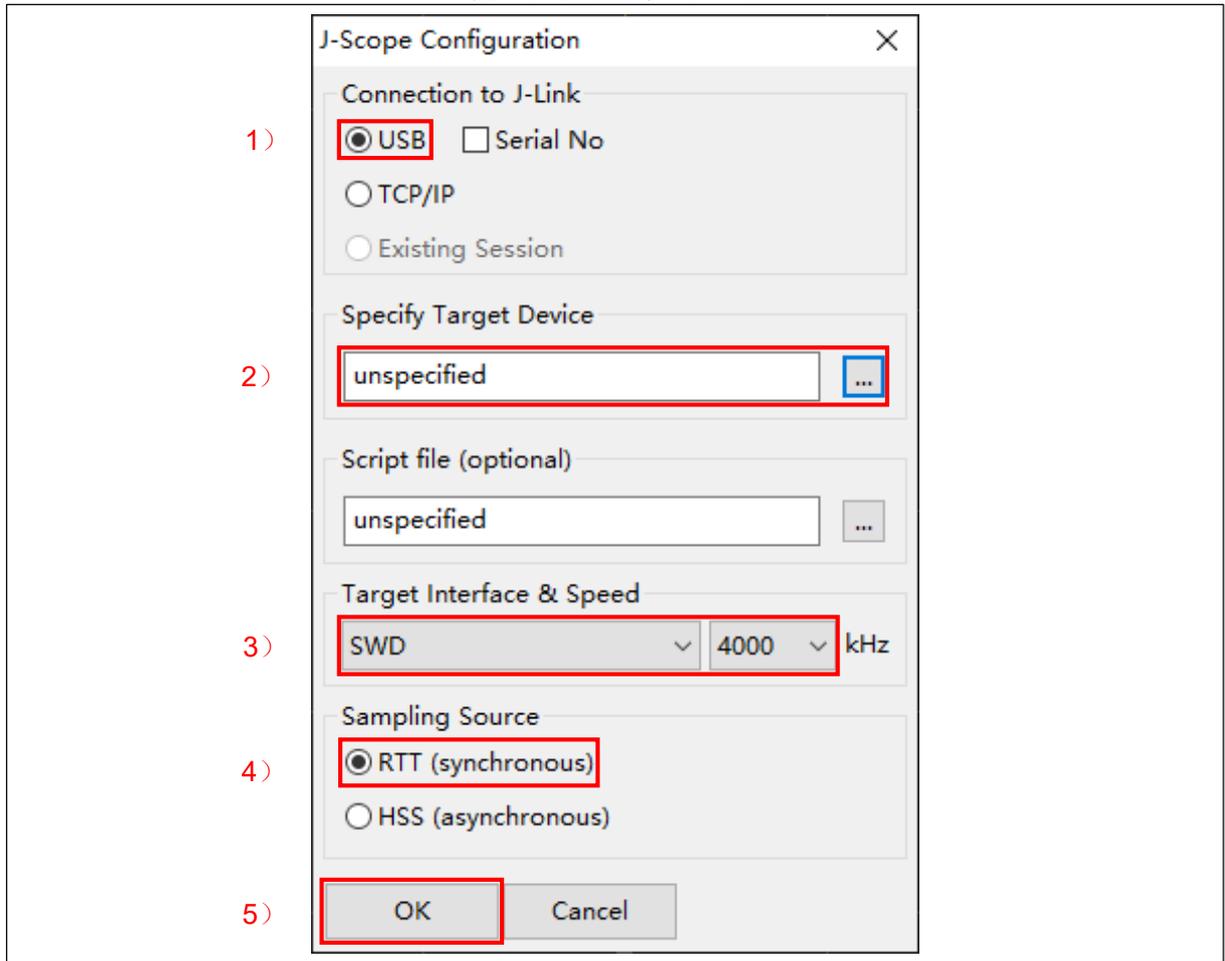
## 4.2.2 打开 J-Scope，新建工程

图 28. 新建工程（RTT）



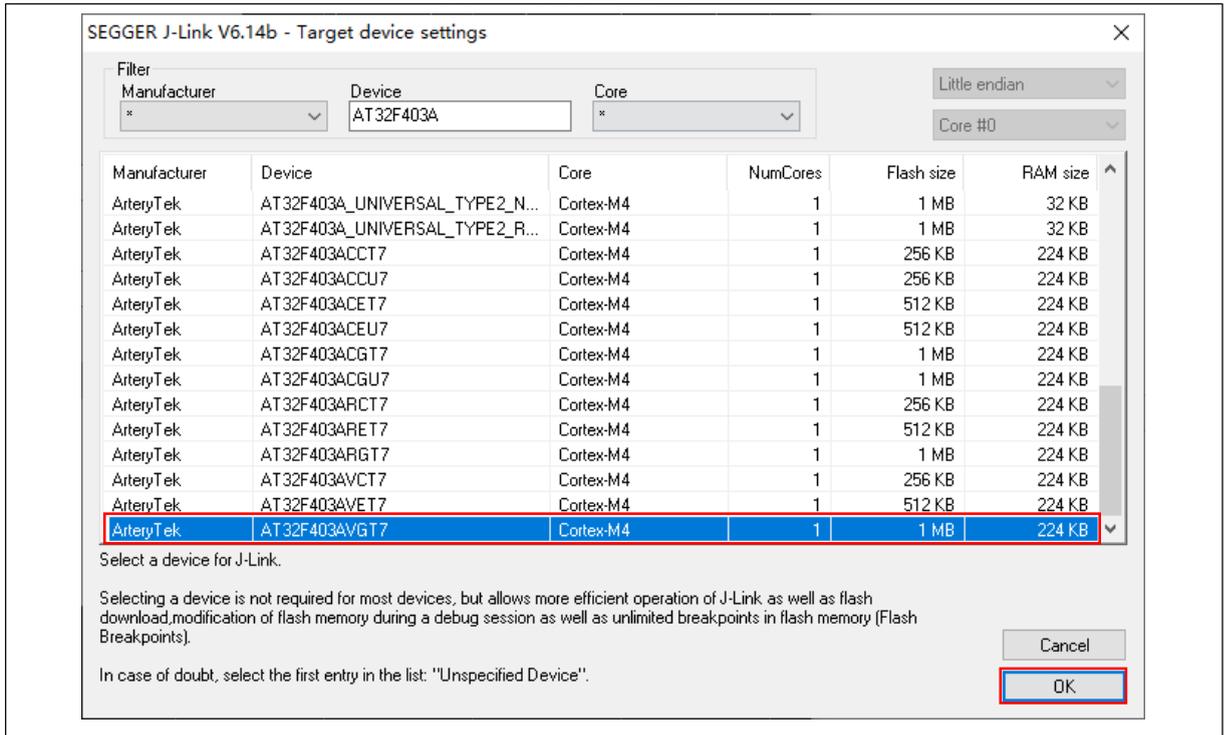
## 4.2.3 配置工程

图 29. 配置工程 (RTT)



- 1) JLINK接口选择: 大家通常使用的JLINK都是USB接口, 所以此处选择USB。
- 2) 目标器件选择: 此处选择AT32F403AVGT7。

图 30. 目标器件选择 (RTT)

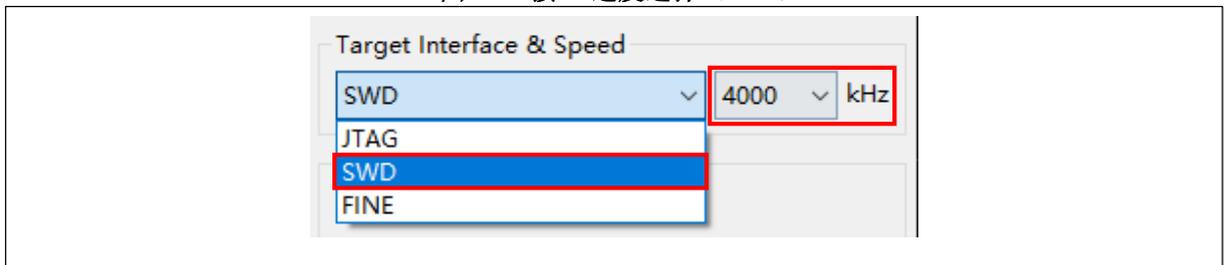


3) 目标接口/速度选择:

目标接口: 根据应用程序配置, 选择SWD/JTAG均可。本例程选择的SWD, 所以这里也选择SWD。

JLINK速度: 这里选择默认的4000kHz。

图 31. 接口/速度选择 (RTT)



4) 模式选择: 此处选择RTT。

5) 配置完成, 点击ok, 完成配置。

## 4.2.4 开始运行

单击红点, 开始显示数据。

图 32. 开始运行 (RTT)

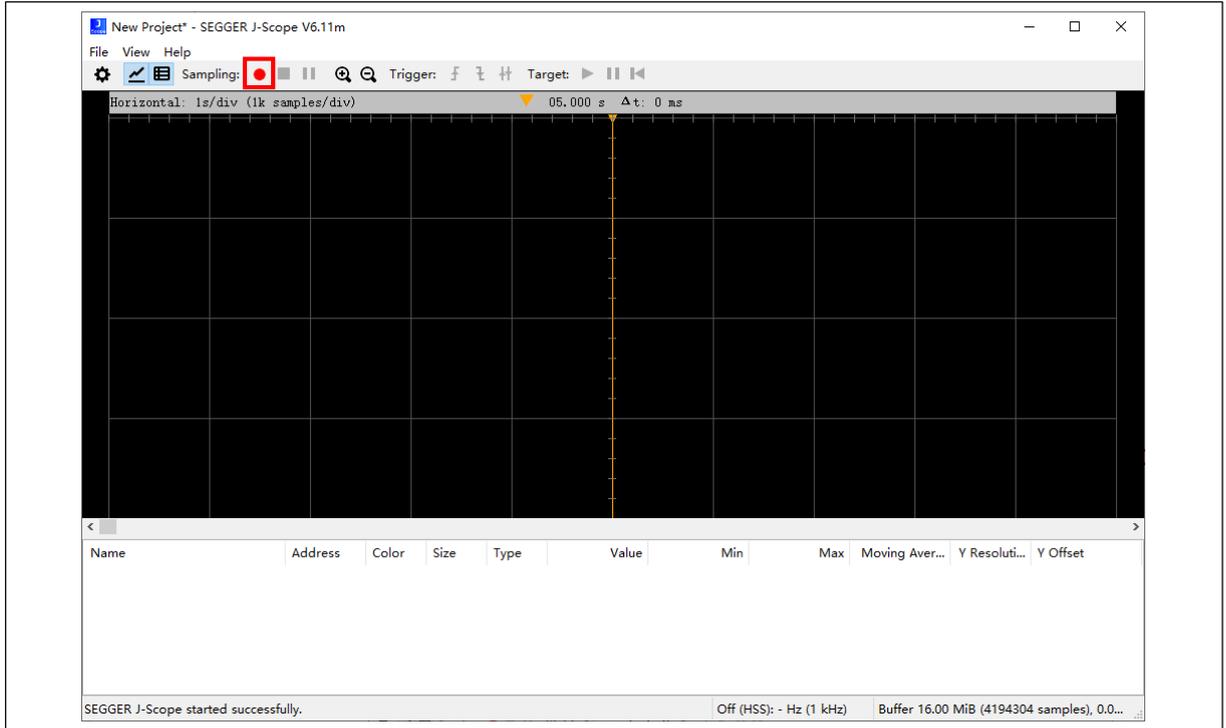
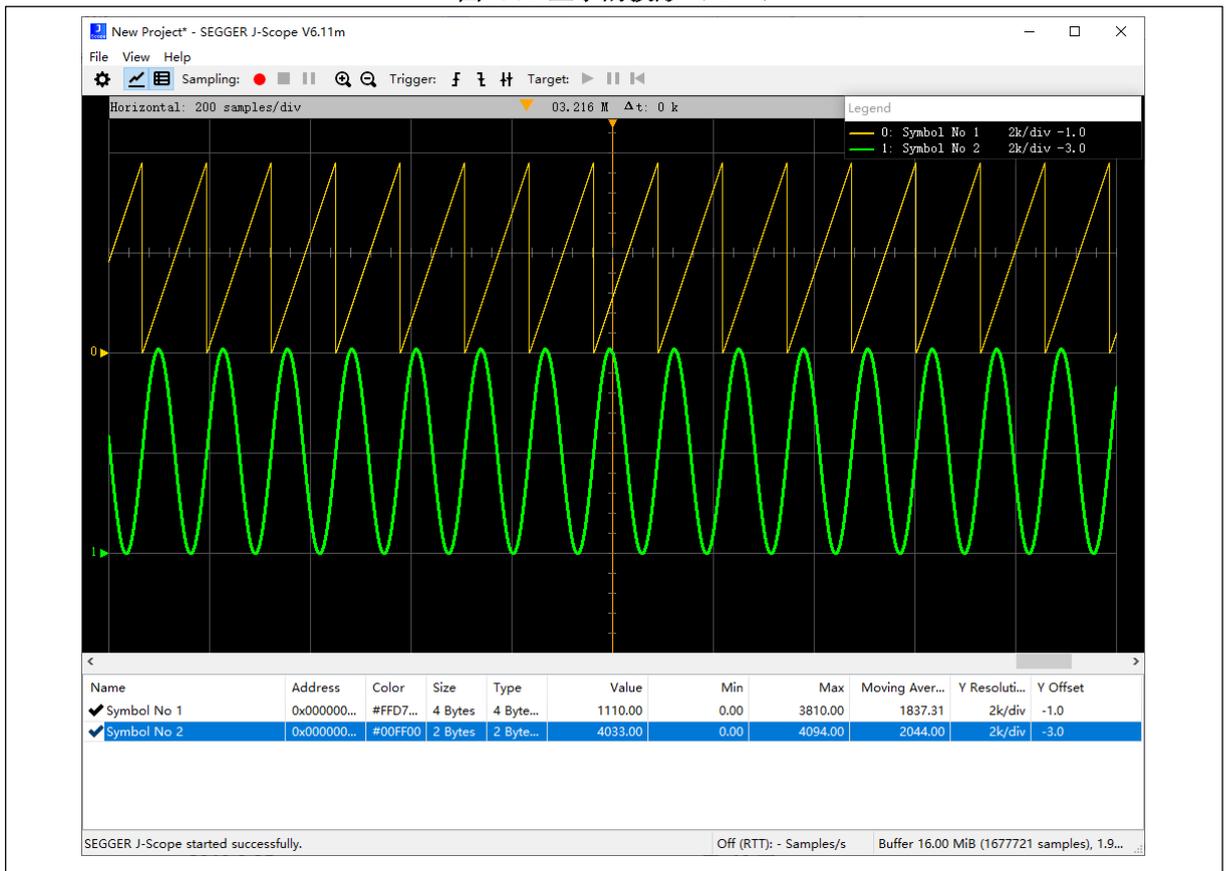


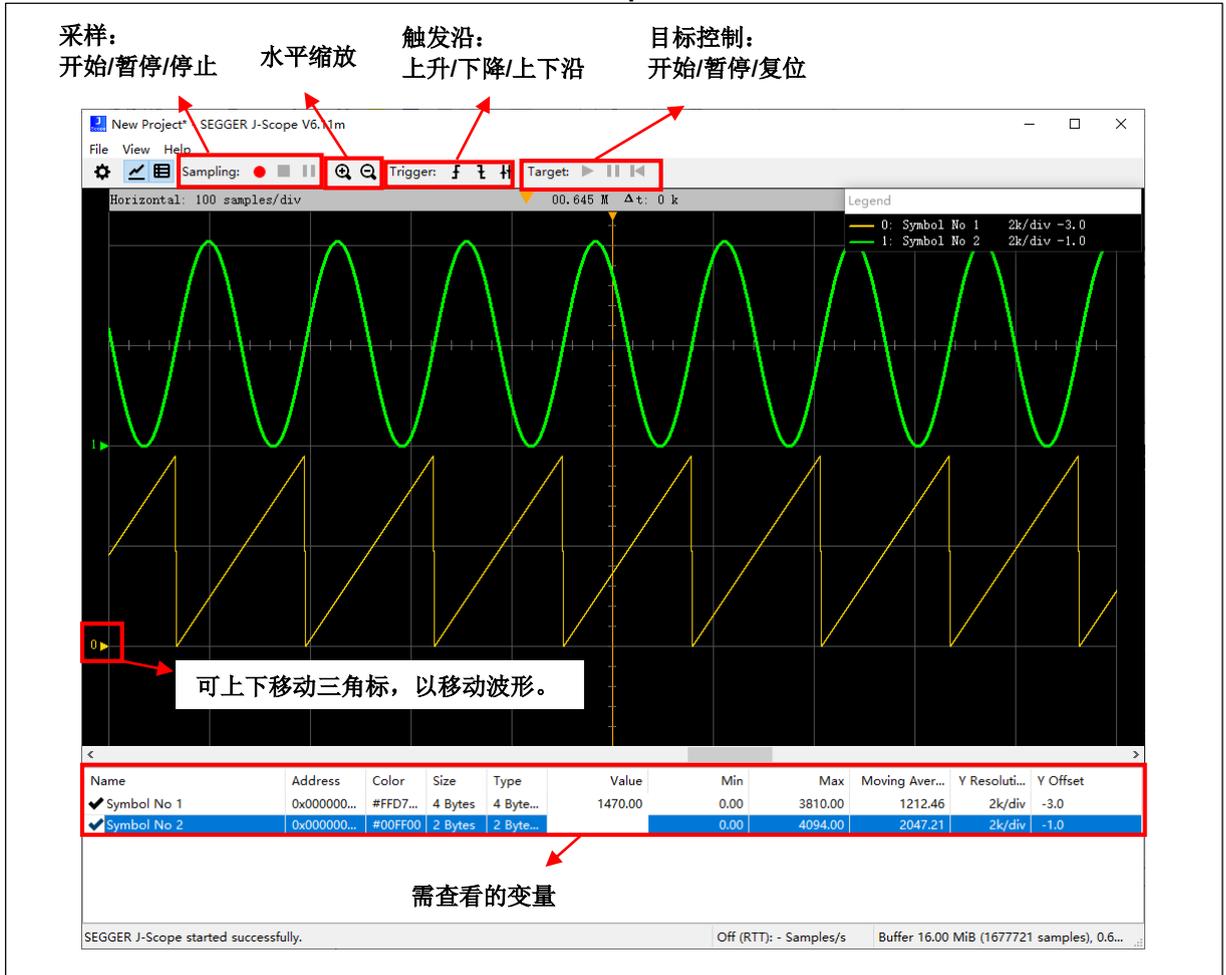
图 33. 显示的波形 (RTT)



## 5 J-Scope 软件介绍

### 5.1 界面介绍

图 34. J-Scope 界面介绍



### 5.2 数据导出

J-Scope 支持导出 CSV 或原始的 RAW 格式的数据。

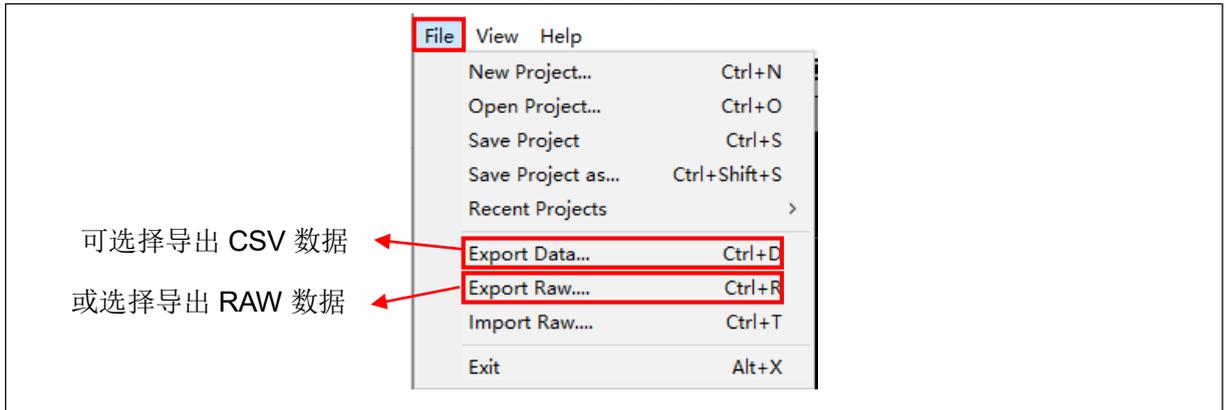
1) 若采样正在进行, 则需要先停止运行:

图 35. 停止运行



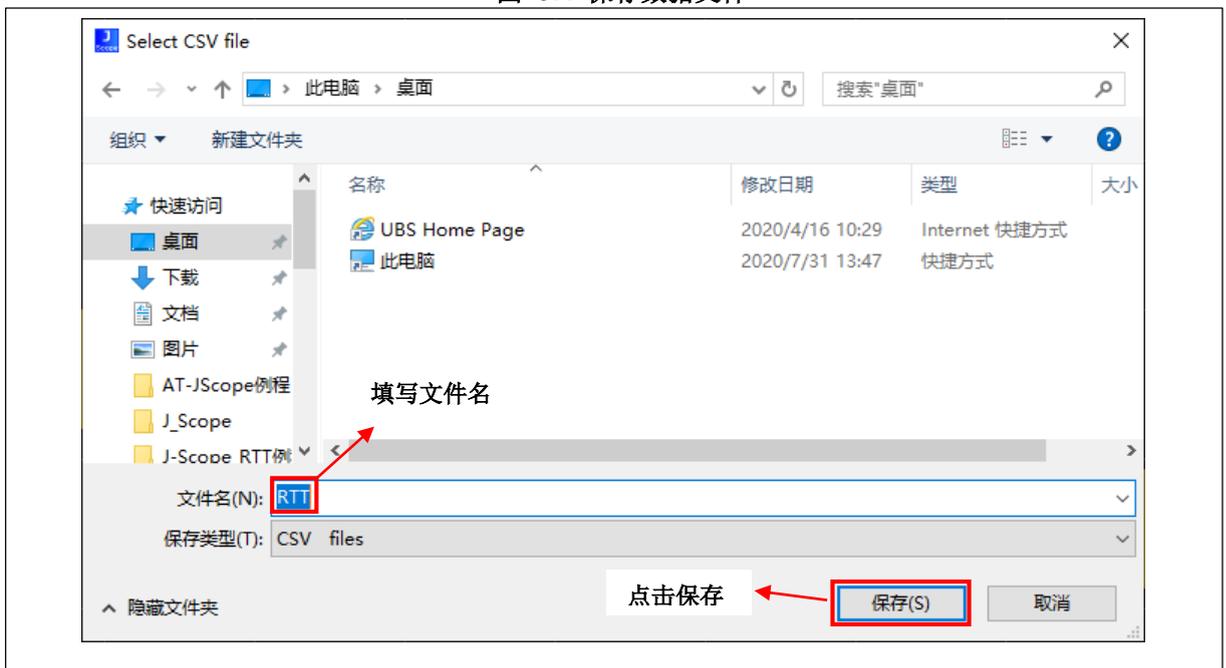
2) 点击File->Export Data:

图 36. 导出数据



3) 填写文件名->点击保存:

图 37. 保存数据文件



4) 打开RTT.CSV可以看到存储的数据，如下图:

图 38. 查看数据文件

Timestamp	Symbol No	1
100	0	
200	30	
300	60	
400	90	
500	120	
600	150	
700	180	
800	210	
900	240	
1000	270	
1100	300	
1200	330	

## 5.3 快捷键

图 39. 快捷键

Function	GUI Input	Shortcut
<b>Graph Area Settings</b>		
Zoom in (X-Axis)	Toolbar	Ctrl + Wheel up
Zoom out (X-Axis)	Toolbar	Ctrl + Wheel down
Set Zoom target	Drag ZTI	N/A
Change X-Axis scope	Move Scrollbar / Drag Graph Area	Wheel up / down
<i>Symbol Settings (Apply to the symbol currently selected)</i>		
Add Symbol	Symbol Context menu	N/A
Remove Symbol	Symbol Context menu	Delete
Show / Hide toggle	Symbol Context menu	Enter, Space
Zoom in (Y-Axis)	Symbol Context menu	+
Zoom out (Y-Axis)	Symbol Context menu	-
Change draw style	Symbol Context menu	N/A
(Hexa-)Decimal toggle	Symbol Context menu	N/A
Change color	Symbol Context menu	N/A
Y-Offset up	Drag Base Line Indicator	Ctrl + +
Y-Offset down	Drag Base Line Indicator	Ctrl + -
<b>Sampling Controls</b>		
Start Sampling	Toolbar	N/A
Stop Sampling	Toolbar	N/A
Pause Sampling	Toolbar	N/A
Start Target	Toolbar	N/A
Halt Target	Toolbar	N/A
Reset Target	Toolbar	N/A
<b>General</b>		
Exit J-Scope	File Menu	Alt + X
About Dialog	Help Menu	F1
<b>Project Management</b>		
Open Project	File Menu	Ctrl + O
New Project	File Menu	Ctrl + N
Save Project	File Menu	Ctrl + S
Save Project as	File Menu	Ctrl + Shift + S
Export CSV	File Menu	Ctrl + D
Export RAW	File Menu	Ctrl + R
Import RAW	File Menu	Ctrl + T

## 6 例程

注：所有project都是基于keil 5而建立，若用户需要在其他编译环境上使用，请参考AT32xxx\_Firmware\_Library\_V2.x.x\project\at\_start\_xxx\templates中各种编译环境（例如IAR6/7, keil 4/5）进行简单修改即可。

### 6.1 HSS 模式

- 1) 解压并打开xx\AN0044\_Segger\_Jscope\_on\_AT32\_MCU\_2.x.x\SourceCode\J-Scope\_HSS\_V2.x.x。
- 2) 应用程序定义两个全局变量Test\_Data1, Test\_Data2。
- 2) 新建J-Scope工程，再选中采样变量（Test\_Data1, Test\_Data2），再点击开始运行即可。

图 40. 应用程序 HSS

```

4      * Version: v2.0.0
5      * Date   : 2021-11-02
6      * Brief  : Main program body
7      *-----*
8      */
9      #include "at32f403a_407_board.h"
10
11     uint16_t Test_Data1;
12     uint16_t Test_Data2;
13     const uint16_t Cos_data[128] = {
14     2047, 2147, 2248, 2347, 2446, 2544, 2641, 2737, 2830, 2922, 3012, 3099, 3184, 3266, 3346, 3422,
15     3494, 3564, 3629, 3691, 3749, 3803, 3852, 3897, 3938, 3974, 4006, 4033, 4055, 4072, 4084, 4092,
16     4094, 4092, 4084, 4072, 4055, 4033, 4006, 3974, 3938, 3897, 3852, 3803, 3749, 3691, 3629, 3564,
17     3494, 3422, 3346, 3266, 3184, 3099, 3012, 2922, 2830, 2737, 2641, 2544, 2446, 2347, 2248, 2147,
18     2047, 1947, 1846, 1747, 1648, 1550, 1453, 1357, 1264, 1172, 1082, 995, 910, 828, 748, 672,
19     600, 530, 465, 403, 345, 291, 242, 197, 156, 120, 88, 61, 39, 22, 10, 2,
20     0, 2, 10, 22, 39, 61, 88, 120, 156, 197, 242, 291, 345, 403, 465, 530,
21     600, 672, 748, 828, 910, 995, 1082, 1172, 1264, 1357, 1453, 1550, 1648, 1747, 1846, 1947,
22     };
23     uint8_t count;
24
25     int main()
26     {
27         at32_board_init();
28
29         while(1)
30         {
31             Test_Data1 = (count%128)*30;          /*锯齿波*/
32             Test_Data2 = Cos_data[count%128];    /*余弦波*/
33             count++;
34             delay_ms(1);
35         }
36     }
37

```

### 6.2 RTT 模式

- 1) 解压并打开xx\AN0044\_Segger\_Jscope\_on\_AT32\_MCU\_2.x.x\SourceCode\J-Scope\_RTT\_V2.x.x。
- 2) 移植RTT组件；
- 3) main.c包含SEGGER\_RTT.h文件；
- 4) 定义两个全局变量Test\_Data1, Test\_Data2；
- 5) 上行缓冲区配置：

```
SEGGER_RTT_ConfigUpBuffer(1, "JScope_u4u2", buf, 2048, SEGGER_RTT_MODE_NO_BLOCK_SKIP);
```

- 6) 数据输出：

```
SEGGER_RTT_Write(1, &Test_Data1, 4);
SEGGER_RTT_Write(1, &Test_Data2, 2);
```

- 7) 然后新建J-Scope RTT工程，再点击运行即可观察波形。

图 41. 应用程序 RTT

```

9  #include "at32f403a_407_board.h"
10 #include "SEGGER_RTT.h"
11
12 uint16_t buf[2048];
13 uint32_t Test_Data1;
14 uint16_t Test_Data2;
15 const uint16_t Cos_data[128] = {
16 4094, 4092, 4084, 4072, 4055, 4033, 4006, 3974, 3938, 3897, 3852, 3803, 3749, 3691, 3629, 3564,
17 3494, 3422, 3346, 3266, 3184, 3099, 3012, 2922, 2830, 2737, 2641, 2544, 2446, 2347, 2248, 2147,
18 2047, 1947, 1846, 1747, 1648, 1550, 1453, 1357, 1264, 1172, 1082, 995, 910, 828, 748, 672,
19 600, 530, 465, 403, 345, 291, 242, 197, 156, 120, 88, 61, 39, 22, 10, 2,
20 0, 2, 10, 22, 39, 61, 88, 120, 156, 197, 242, 291, 345, 403, 465, 530,
21 600, 672, 748, 828, 910, 995, 1082, 1172, 1264, 1357, 1453, 1550, 1648, 1747, 1846, 1947,
22 2047, 2147, 2248, 2347, 2446, 2544, 2641, 2737, 2830, 2922, 3012, 3099, 3184, 3266, 3346, 3422,
23 3494, 3564, 3629, 3691, 3749, 3803, 3852, 3897, 3938, 3974, 4006, 4033, 4055, 4072, 4084, 4092,
24 };
25
26 uint8_t count;
27
28 int main()
29 {
30     at32_board_init();
31     /* 配置通道1, 上行配置*/
32     SEGGER_RTT_ConfigUpBuffer(1, "JScope_u4u2", buf, 2048, SEGGER_RTT_MODE_NO_BLOCK_SKIP);
33     while(1)
34     {
35         Test_Data1 = (count%128)*30; //锯齿波
36         Test_Data2 = Cos_data[count%128]; //余弦波
37         count++;
38
39         SEGGER_RTT_Write(1, &Test_Data1, 4);
40         SEGGER_RTT_Write(1, &Test_Data2, 2);
41
42         delay_us(100);
43     }
44 }
45
46

```

## 7 版本历史

表 2. 文档版本历史

日期	版本	变更
2021.11.02	2.0.0	初始版本

#### 重要通知 - 请仔细阅读

买方自行负责对本文所述雅特力产品和服务的选择和使用，雅特力概不承担与选择或使用本文所述雅特力产品和服务相关的任何责任。

无论之前是否有过任何形式的表示，本文档不以任何方式对任何知识产权进行任何明示或默示的授权或许可。如果本文档任何部分涉及任何第三方产品或服务，不应被视为雅特力授权使用此类第三方产品或服务，或许可其中的任何知识产权，或者被视为涉及以任何方式使用任何此类第三方产品或服务或其中任何知识产权的保证。

除非在雅特力的销售条款中另有说明，否则，雅特力对雅特力产品的使用和/或销售不做任何明示或默示的保证，包括但不限于有关适销性、适合特定用途（及其依据任何司法管辖区的法律的对应情况），或侵犯任何专利、版权或其他知识产权的默示保证。

雅特力产品并非设计或专门用于下列用途的产品：(A) 对安全性有特别要求的应用，例如：生命支持、主动植入设备或对产品功能安全有要求的系统；(B) 航空应用；(C) 航天应用或航天环境；(D) 武器；且/或 (E) 其他可能导致人身伤害、死亡及财产损害的应用。如果采购商擅自将其用于前述应用，即使采购商向雅特力发出了书面通知，风险及法律责任仍将由采购商单独承担，且采购商应独立负责在前述应用中满足所有法律和法规要求。

经销的雅特力产品如有不同于本文档中提出的声明和/或技术特点的规定，将立即导致雅特力针对本文所述雅特力产品或服务授予的任何保证失效，并且不应以任何形式造成或扩大雅特力的任何责任。

© 2021 雅特力科技 保留所有权利