SEGGER J-Scope

User Guide

Document: UM08028
Software Version: 5.10
Revision: 0
Date: November 26, 2015

A product of SEGGER Microcontroller GmbH & Co. KG

www.segger.com

Disclaimer

Specifications written in this document are believed to be accurate, but are not guaranteed to be entirely free of error. The information in this manual is subject to change for functional or performance improvements without notice. Please make sure your manual is the latest edition. While the information herein is assumed to be accurate, SEGGER Microcontroller GmbH & Co. KG (SEGGER) assumes no responsibility for any errors or omissions. SEGGER makes and you receive no warranties or conditions, express, implied, statutory or in any communication with you. SEGGER specifically disclaims any implied warranty of merchantability or fitness for a particular purpose.

Copyright notice

You may not extract portions of this manual or modify the PDF file in any way without the prior written permission of SEGGER. The software described in this document is furnished under a license and may only be used or copied in accordance with the terms of such a license.

© 2015 SEGGER Microcontroller GmbH & Co. KG, Hilden / Germany

Trademarks

Names mentioned in this manual may be trademarks of their respective companies.

Brand and product names are trademarks or registered trademarks of their respective holders.

Contact address

SEGGER Microcontroller GmbH & Co. KG

In den Weiden 11
D-40721 Hilden

Germany

Tel.	+49 2103-2878-0
Fax.	+49 2103-2878-28
E-mail:	support@segger.com
Internet:	www.segger.com

Manual versions

This manual describes the current software version. If you find an error in the manual or a problem in the software, please inform us and we will try to assist you as soon as possible. Contact us for further information on topics or functions that are not yet documented.

Print date: November 26, 2015

Software	Revision	Date	By	Description
5.10	0	151124	NG	First standalone version.

4

About this document

Assumptions

This document assumes that you already have a solid knowledge of the following:

- The software tools used for building your application (assembler, linker, C compiler).
- The C programming language.
- The target processor.
- DOS command line.

If you feel that your knowledge of C is not sufficient, we recommend *The C Programming Language* by Kernighan and Richie (ISBN 0–13–1103628), which describes the standard in C programming and, in newer editions, also covers the ANSI C standard.

How to use this manual

This manual explains all the functions and macros that the product offers. It assumes you have a working knowledge of the C language. Knowledge of assembly programming is not required.

Typographic conventions for syntax

This manual uses the following typographic conventions:

Style	Used for
Body	Body text.
Keyword	Text that you enter at the command prompt or that appears on the display (that is system functions, file- or pathnames).
Parameter	Parameters in API functions.
Sample	Sample code in program examples.
Sample comment	Comments in program examples.
Reference	Reference to chapters, sections, tables and figures or other documents.
GUIElement	Buttons, dialog boxes, menu names, menu commands.
Emphasis	Very important sections.

6

Table of contents

Chapter 1

Segger J-Scope

1.1   What is SEGGER J-Scope ?

J-Scope is a free-of-charge software to analyze and visualize data on a microcontroller in real-time, while the target is running.

Technology used

Sampling can be done using either SEGGER High-Speed-Sampling (HSS) or SEGGER RTT technology. Both technologies are available to all MCUs that provide background memory access. SEGGER HSS can be used without any further preparation. SEGGER RTT enables faster sampling speeds but requires instrumentation of the target application as described in *RTT: Instrumenting an application to use it with J-Scope* on page 24.

1.2   Getting Started

J-Scope was formerly part of the J-Link Software and Documentation Package (Versions ≤ 5.02). It is now available as a standalone package. SEGGER J-Scope download

Note: Please note that J-Scope requires the J-Link Software and Documentation Package to be installed.

Preparation

Before starting J-Scope make sure you have an elf-file with symbol information of your target application and the target is running the application. The application can either be flashed with J-Flash, J-Link Commander or directly ran from an IDE. Start J-Scope by double-clicking on the executable.
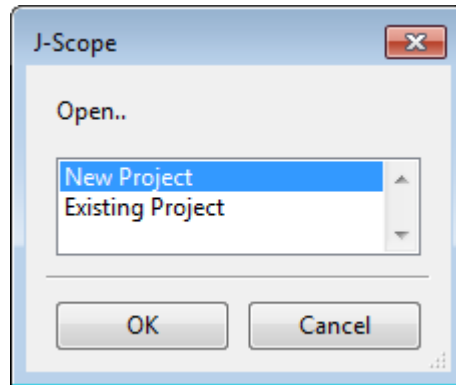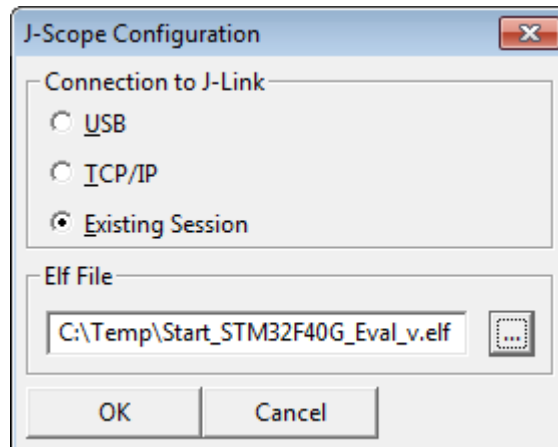

Select "New project" and press OK.

The Project Configuration Dialog will open.

1.2.1   Set up Connection

Attach to a running instance

J-Scope can be used alongside other applications, e.g. a debugger, which are connected to J-Link and share the connection to the J-Link. If J-Scope is used in this mode select "Existing Session" as connection type. All Target options will be disabled, as they are already determined by the running session.

Note: In attach mode J-Scope will connect to J-Link indirectly via another application connected to J-Link. J-Scope can only be used while the connection is active, e.g. a debug session is running. It is the applications responsibility to set up the target and J-Link connection and to manage target execution. RTT can not be used in attach mode.

Connecting to J-Link
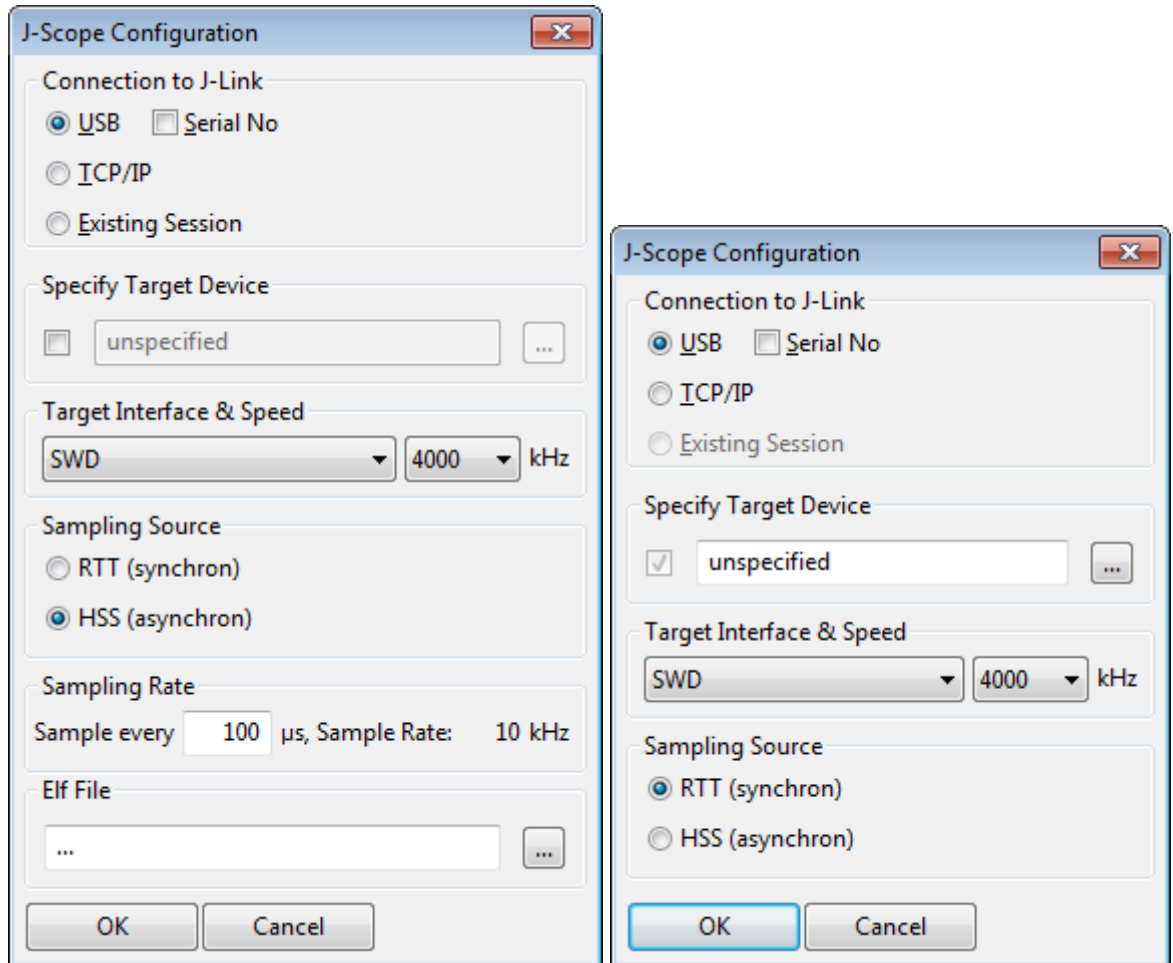
If J-Scope is used in standalone mode, opening a dedicated connection to J-Link, choose "USB" or "TCP/IP" as connection type and optionally enter the serial number or the IP of the J-Link to connect to a specific J-Link.

HSS Mode

For most targets, the target device does not necessarily have to be specified. If there are any problems, select your target device and configure the interface accordingly.

RTT Mode

In order to use RTT mode, RTT has to be implemented in the target application. General documentation about RTT is available in the RTT manual. In contrast to HSS Mode, specifying the target device is mandatory in RTT Mode.

1.2.2   Select Application File and Symbols

HSS Mode

In the configuration dialog, select your elf-file and click OK. The Symbol Selection Dialog will open. Select the symbols of your application you want to watch and analyse. Click OK to finalize the setup.

RTT Mode

The data being sampled is determined by the target application and cannot be influenced on PC-side by J-Scope. Neither elf-file nor symbol selection is required or available in RTT mode.

1.2.3   Start Sampling

Data sampling can be started via the record-button (red dot) in the toolbar. On start of sampling, the connection to J-Link or running session will be established.

1.2.4   Image of Running Session (RTT)
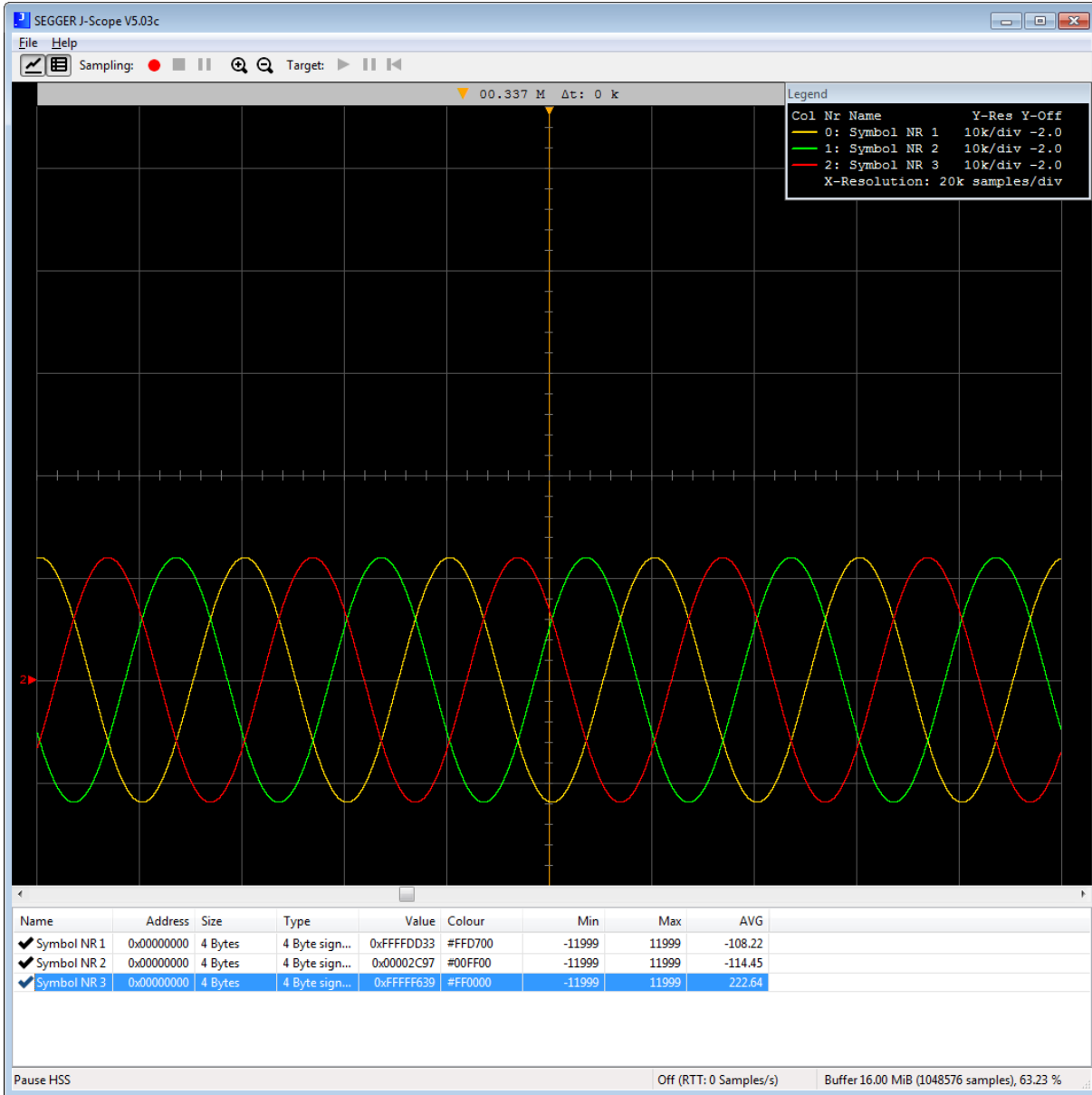
1.3   Project Management

The project configuration, which are the J-Link connection settings, the elf-application file and the configuration of the selected symbols can be saved to a J-Scope project file (.jscope). J-Scope projects can be used for further sessions and exchanged between different users.

To open a project file, the user can either select "Existing Project" in the startup dialog, or open a project file with J-Scope.

If any previously sampled data has been saved to file, J-Scope will automatically open the latest data file associated with the project.

1.4 Exporting Sampled Data

1.4.1 Export to CSV

The sampled data which is currently in the buffer can be exported into CSV format, to be opened and analyzed by other tools, for example spreadsheet programs like MS Excel, Apple Numbers or Libre Office Calc, by pressing File → Export Data

1.4.2 Export to RAW

The sampled data which is currently in the buffer can be saved in a RAW format, which can be opened by J-Scope at a later point of time. RAW files may only be used within projects with a symbol configuration matching the one used to sample the data in the first place.

The relative location of the latest RAW file created will be saved in the project file and will be opened if the project is loaded.

1.5   Symbol Select Dialog
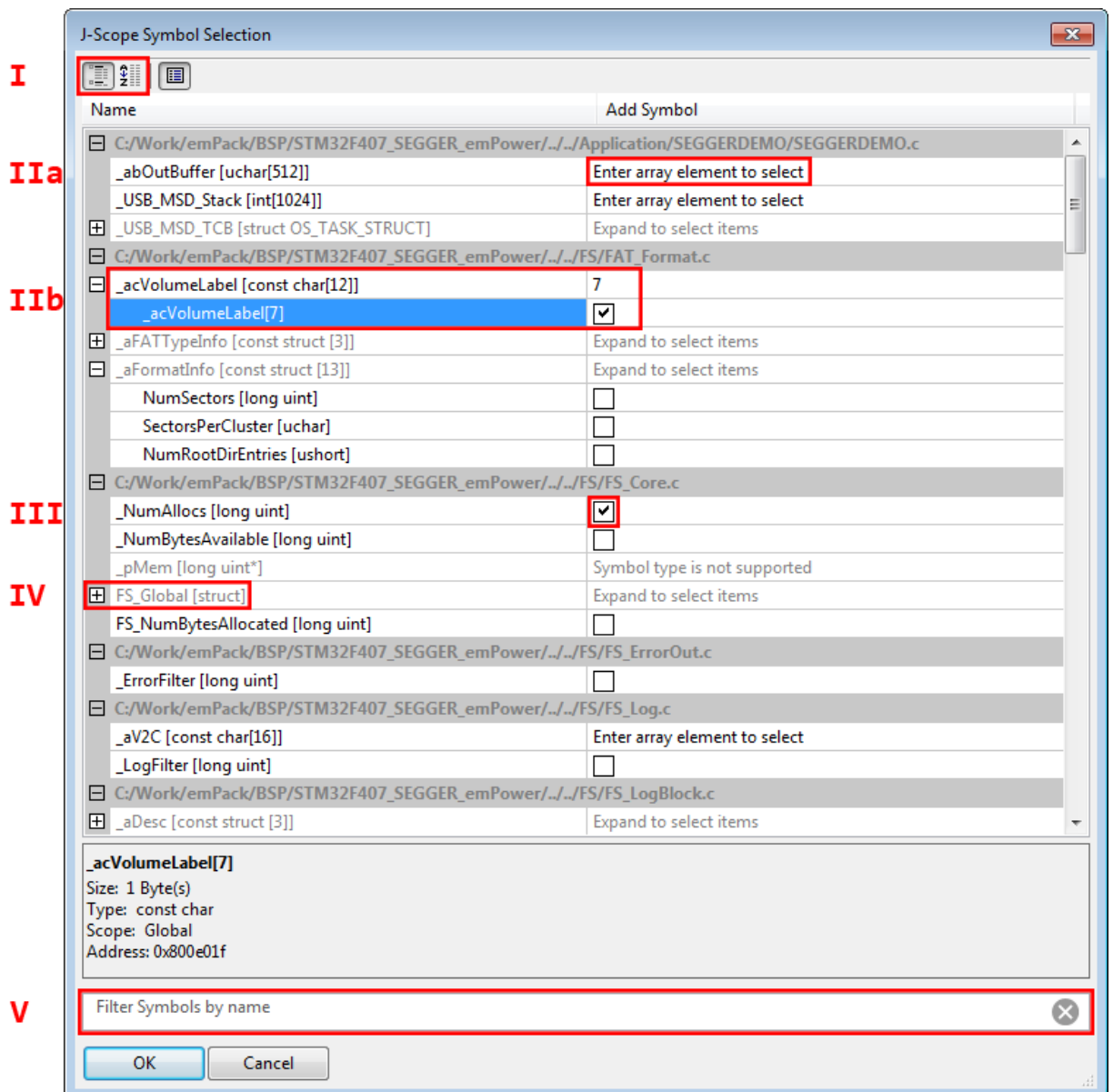
|			
I. Categorized mode (default)	Shows the symbols in a tree view as children of their compilation unit file (e.g. _NumAllocs is defined in FS_Core.c)
I. Alphabetical mode	Shows all symbols in alphabetical order
II. Selection of array elements	While complete arrays are not available for sampling, array elements of supported base types can be sampled. In order to sample specific array elements, numbers can be filled in the corresponding field (IIa). After the user confirmed his input by pressing enter, a selectable row is added to the dialog (IIb)
III. Selection of base types	Supported symbols are listed with a check box in the "Add Symbol" column. Symbols can be added to the list of sampled symbols by enabling the check box.
IV. Selection of struct members	Members of a specific struct can be shown in a tree view by clicking the "+" next to a structs name.
V. Search	Provides a case-insensitive search. Symbols which cannot be selected are not shown in the search results. Structs are shown as longs a their name or the name of at least one member contains the search string. The user can right click on a symbol in order to

show all children of the symbol clicked, even if they do not match
the search string.

1.6   Main Window
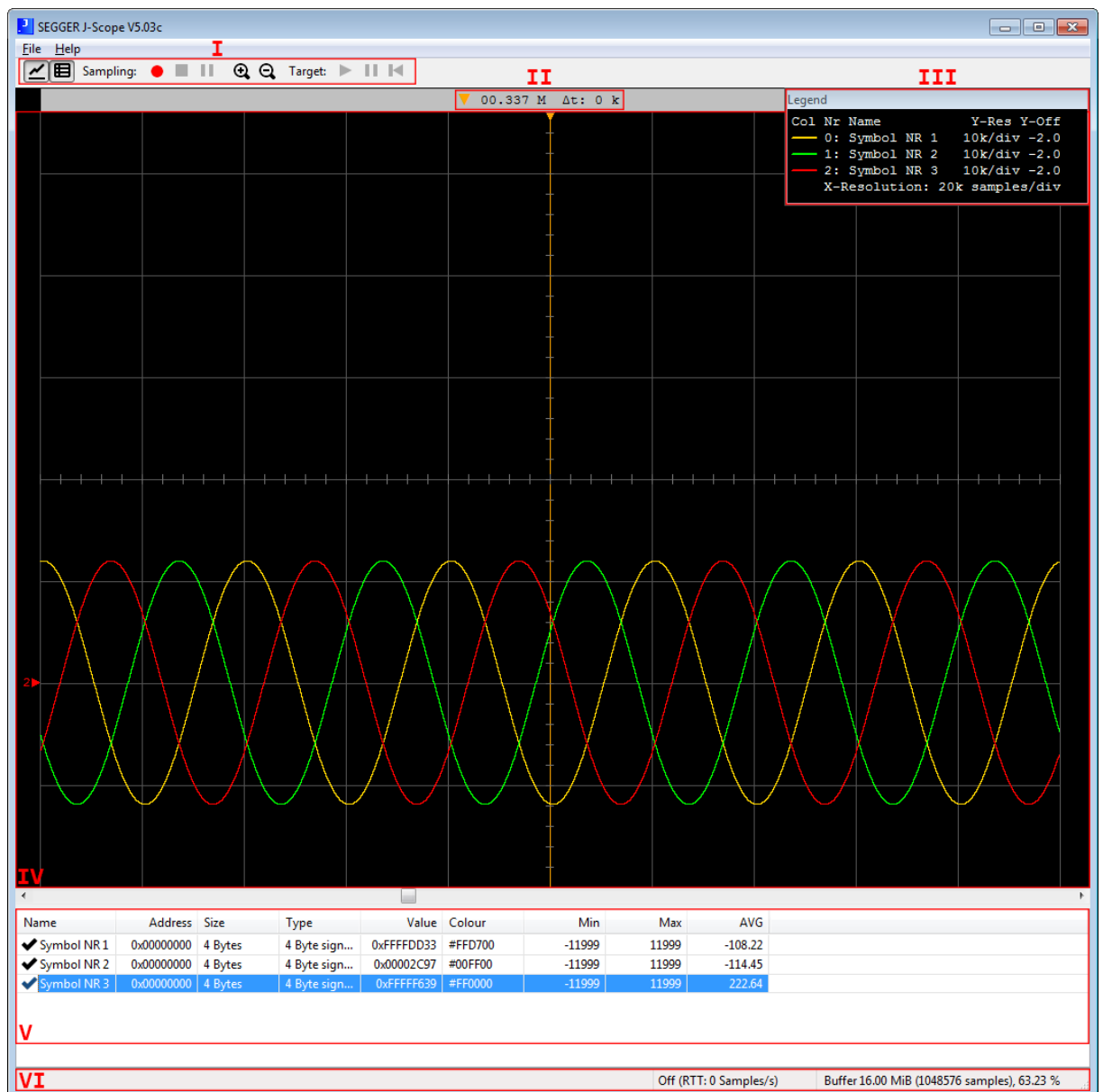
I.   *Toolbar* on page 17
II.   *Zoom target indicator bar* on page 18
III.   *Legend* on page 18
IV.   *Graph area* on page 18
V.   *Symbol Watch* on page 19
VI.   *Statusbar* on page 19

1.6.1   Toolbar


I.   Show/Hide the Graph | Show/Hide Watch Window
II.   Start sampling | Stop sampling | Pause / Resume sampling
III.   Zoom in | Zoom out
IV.   Start the target | Halt the target | Reset the target

1.6.2   Zoom target indicator bar
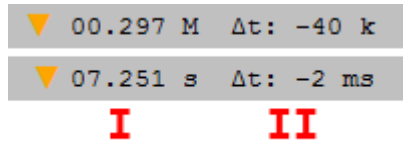
| I. | Shows either the time (HSS mode / RTT mode with timestamp) or sample number (RTT mode w/o timestamp) at the position of the Zoom target indicator (orange line, "ZTI") |
| II. | Show the time / sample number difference between the ZTI and the vertical central line |

1.6.3   Legend
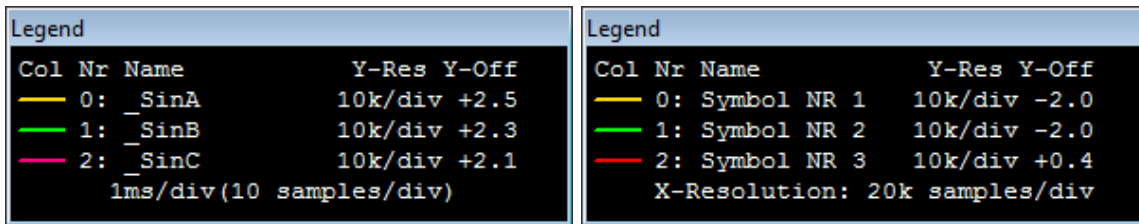
Shows Color, Number, Name, Y-Resolution and Y-Offset for each symbol.

Shows X-Resolution of the graph area.

The legend can be moved independently of the main window. If the main windows is moved, it will snap to the top right corner of the graph area.

1.6.4   Graph area
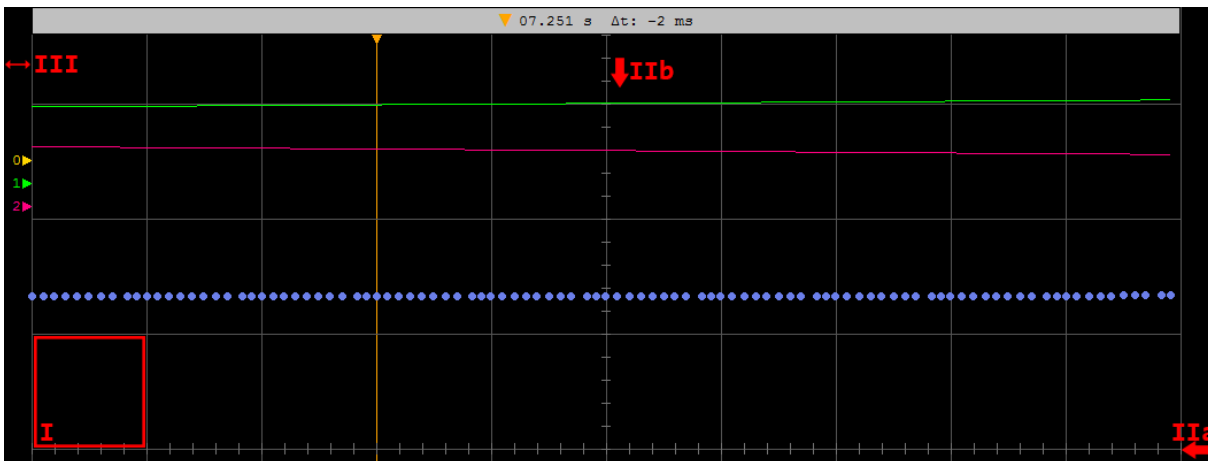
Area where the graphs are drawn. The area is divided into divisions of 100x100 pixels (thin gray lines, eg. I).

At startup, the area is 800px high and 1000px wide. The 4th line from the top (IIa) and the 5th line from the right(IIb) are called central lines and have little marks every 20px.

Their position is independent from the window size. On the left side of the graph area(III), each symbol's base line is indicated by a label containing the symbol number.

A symbol's base line can be changed by moving the respective label (grab and move).

1.6.5 Symbol Watch

Name	**Address**	**Size**	**Type**	**Value**	**Colour**	**Min**	**Max**	**AVG**
✔ _SinA	0x20009FC4	4 Bytes	long	0x00002D93	#FFD700	-11999	11999	-83.44
✔ _SinB	0x20009FC8	4 Bytes	long	0xFFFFF2B3	#00FF00	-11999	11999	94.57
✔ _SinC	0x20009FCC	4 Bytes	long	0xFFFFDFBE	#FF0080	-11999	11999	-11.14
✘ _NumAllocs	0x20001540	4 Bytes	long uint	0x0	#A020F0	0	0	0.00
I	II	III	IV	V	VI	VII		

I.	Symbol name according to the .elf file provided (HSS mode) or "Symbol NR n" (RTT mode). Preceded by either a check mark (symbol is shown) or a "X"(symbol is hidden)
II.	Address of the symbol according to the .elf file (HSS mode only)
III.	Size of the symbol
IV.	Type of the symbol (HSS mode only)
V.	Most current value sampled
VI.	Color of the graph representing the symbol
VII.	Minimum, Maximum and AVG sample value of the symbol

1.6.6 Statusbar

Project opened.	On: RTT: 25046 Samples/s	Buffer 16.00 MiB (1048576 samples), 100.00 %
I	II	III

I.	Shows the latest status message
II.	Indicates Status (On / Off), Type (RTT/HSS) and current sampling speed
III.	Shows the current buffer size, the maximum number of sampling packets fitting in the buffer and the current occupation of the buffer in %

1.7   Common short cuts

The Y-Resolution of a symbol can be changed be selecting the respective row in the watch and using the "+" and "-" keys.

The visibility of a symbol can be changed by pressing "enter".

Scrolling through the Graph Window can be done with the mouse wheel.

Holding control while using the mouse wheel will zoom in and out the time base.

1.8   Controls

Function	GUI Input	Shortcut
Graph Area Settings		
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
Symbol Settings *(Apply to the symbol currently selected)*		
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 + -
Sampling Controls		
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
General		
Exit J-Scope	File Menu	Alt + X
About Dialog	Help Menu	F1
Project Management		
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

1.9   General concepts / behavior of J-Scope

- The sample buffer acts like a circular buffer. If the buffer is full, each new entry overrides the oldest one present.
- If sampling is paused, neither will the target be halted nor will HSS/RTT be suspended, but all incoming samples will be discarded until sampling is resumed.
- If no timestamps are provided in RTT mode, J-Scope will number the incoming packets in order of arrival. Accordingly, all time based values will be shown as number based values instead.

1.10 Limits of J-Scope

J-Scope does not limit the sampling speed or number of variables sampled.

Information about the capabilities of a J-Link are provided on the SEGGER website. HSS' limits on number of variables and sampling speed are described on the SEGGER J-Scope webpage (Section *Model comparison*).

Information on speed limitations depending on J-Link model, debug interface speed and RTT buffer size are available on the SEGGER RTT webpage (Section *Performance*).

Further limitations may be caused by the OS or USB implementation of the host PC. J-Link debug probes with Ethernet connectivity can be used with J-Scope in order to allow to achieve speeds beyond the limits introduced by USB.

1.11 RTT: Instrumenting an application to use it with J-Scope

Preparation

First of all, SEGGER RTT needs to be included in the target application.

The source files and manual of RTT are part of the SEGGGER J-Link Software and Documentation Package.

In RTT mode, all data is provided directly by the target application. The target application is responsible for data integrity and providing a reliable time base, if a timestamp is required.

Main concept

The sampled data and its format are given by the target application.

This data format is announced to J-Scope by naming the RTT channel. The naming convention is stated below.

Once the target application has collected all data needed for a packet, it writes this packet using SEGGER_RTT_Write() to the RTT channel.

RTT channel naming convention

J-Scope searches for the first RTT channel named JScope_FORMAT, where FORMAT defines the format of the data written to the buffer by the target application.

- All variable identifiers consist of a type identifier and a size indication(in bytes).
- The data inside a packet has to be in same order as declared in FORMAT.
- If a timestamp is provided, it needs to be declared as first variable.

Type	Supported Sizes	Note
t	4	Indicates that every packet is preceded by a 32bit value containing a timestamp in µs
i	1, 2, 4	Specifies a signed 8-, 16- or 32bit value
u	1, 2, 4	Specifies an unsigned 8-, 16- or 32bit value

Naming Examples

JScope_t4u1i4	Specifies that the data packets are headed by a 32bit time value, followed by an unsigned char and a signed int
JScope_u4i1	Specifies that the data packets contain an unsigned int, followed by a signed char

Workflow Examples

- The user wants to sample a signed int, an unsigned int and a signed char
- The user wants to provide a timestamp.
- Accordingly, a RTT channel is named JScope_t4i4u4i1
- Depending on the use case, sampling for example is done on a timer or condition

Case I

- For this example, the user decides that sampling should be done every 100µs
- The users adds a time based interrupt which occurs every 100µs. The respective ISR collects the necessary data and the current timestamp.
- Afterwards, the ISR calls SEGGER_RTT_Write()

Case II

- In an other use case, the user may decides that sampling should be done once a specific function is called.
- The user adds a subfunction call to this function. The subfunction prepares the data, which either can be passed as a parameter or can be available in a global scope, and writes it to the RTT Buffer using SEGGER_RTT_Write().

Policies in RTT mode

- RTT channel name and data format must not be changed during a sampling session.
- RTT channel name and data format can be changed between sampling sessions (No restart of J-Scope required).
- The RTT channel name must be set accordingly to the data format.
- Only full packets according to the data format declared by the RTT channel name may be written to the RTT channel.
- RTT packets must be sent in order.
- RTT mode should be NO_BLOCK_SKIP or BLOCK_IF_FIFO_FULL.