



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



Flasher User Guide



Software Version V4.66
Manual Rev. 0

Date: April 4, 2013

Document: UM08022



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 (the manufacturer) assumes no responsibility for any errors or omissions. The manufacturer makes and you receive no warranties or conditions, express, implied, statutory or in any communication with you. The manufacturer 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 the manufacturer. 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.

© 2013 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

Email: support@segger.com

Internet: <http://www.segger.com>

Revisions

This manual describes the Flasher device.

For further information on topics or routines not yet specified, please contact us.

Revision	Date	By	Explanation
V4.64a Rev. 0	132602	EL	Chapter "Working with Flasher" * Section "LED status indicators" updated.
V4.63a Rev. 0	130131	EL	Chapter "Remote Control" * Section "ASCII command interface" Chapter "ASCII interface via Telnet" added.
V4.62 Rev. 0	130125	EL	Flasher ARM, Flasher RX and Flasher PPC manual have been combined.

About this document

This document describes the Flasher family (Flasher ARM, Flasher RX and Flasher PPC). It provides an overview about the major features of the Flasher, gives some background information about JTAG and describes Flasher related software packages available from Segger. Finally, the chapter *Support and FAQs* on page 61 helps to troubleshoot common problems.

For simplicity, we will refer to Flasher ARM/RX/PPC as Flasher in this manual.

Typographic conventions

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).
Reference	Reference to chapters, tables and figures or other documents.
GUIElement	Buttons, dialog boxes, menu names, menu commands.

Table 1.1: Typographic conventions



SEGGER Microcontroller GmbH & Co. KG develops and distributes software development tools and ANSI C software components (middleware) for embedded systems in several industries such as telecom, medical technology, consumer electronics, automotive industry and industrial automation.

SEGGER's intention is to cut software development-time for embedded applications by offering compact flexible and easy to use middleware, allowing developers to concentrate on their application.

Our most popular products are emWin, a universal graphic software package for embedded applications, and embOS, a small yet efficient real-time kernel. emWin, written entirely in ANSI C, can easily be used on any CPU and most any display. It is complemented by the available PC tools: Bitmap Converter, Font Converter, Simulator and Viewer. embOS supports most 8/16/32-bit CPUs. Its small memory footprint makes it suitable for single-chip applications.

Apart from its main focus on software tools, SEGGER develops and produces programming tools for flash microcontrollers, as well as J-Link, a JTAG emulator to assist in development, debugging and production, which has rapidly become the industry standard for debug access to ARM cores.

Corporate Office:

<http://www.segger.com>

United States Office:

<http://www.segger-us.com>

EMBEDDED SOFTWARE (Middleware)



emWin

Graphics software and GUI

emWin is designed to provide an efficient, processor- and display controller-independent graphical user interface (GUI) for any application that operates with a graphical display. Starterkits, eval- and trial-versions are available.



embOS

Real Time Operating System

embOS is an RTOS designed to offer the benefits of a complete multitasking system for hard real time applications with minimal resources. The profiling PC tool embOSView is included.



emFile

File system

emFile is an embedded file system with FAT12, FAT16 and FAT32 support. emFile has been optimized for minimum memory consumption in RAM and ROM while maintaining high speed. Various Device drivers, e.g. for NAND and NOR flashes, SD/MMC and CompactFlash cards, are available.



emUSB

USB device stack

A USB stack designed to work on any embedded system with a USB client controller. Bulk communication and most standard device classes are supported.

SEGGER TOOLS

Flasher

Flash programmer

Flash Programming tool primarily for microcontrollers.

J-Link

JTAG emulator for ARM cores

USB driven JTAG interface for ARM cores.

J-Trace

JTAG emulator with trace

USB driven JTAG interface for ARM cores with Trace memory. supporting the ARM ETM (Embedded Trace Macrocell).

J-Link / J-Trace Related Software

Add-on software to be used with SEGGER's industry standard JTAG emulator, this includes flash programming software and flash breakpoints.



Table of Contents

1	Introduction	7
1.1	Flasher overview	8
1.1.1	Features of Flasher	8
1.1.2	Working environment	8
1.2	Specifications	10
1.2.1	Specifications for Flasher ARM	10
1.2.2	Specifications for Flasher RX	12
1.2.3	Specifications for Flasher PPC	14
2	Working with Flasher	17
2.1	Setting up the IP interface	18
2.1.1	Connecting the first time	18
2.2	Operating modes	19
2.2.1	J-Link mode	19
2.2.2	Stand-alone mode	23
2.2.3	MSD mode	24
2.3	Multiple File Support	25
2.4	Serial number programming	26
2.4.1	Serial number settings	26
2.4.2	Serial number file	27
2.4.3	Serial number list file	27
2.4.4	Programming process	28
2.4.5	Downloading serial number files to Flasher	29
2.4.6	Sample setup	29
2.5	Target interfaces	31
2.6	Supported microcontrollers	32
2.6.1	Flasher	32
2.7	Support of external flashes	33
2.7.1	Flasher ARM	33
2.7.2	Flasher RX	33
2.7.3	Flasher PPC	33
2.8	Supported cores	34
2.8.1	Flasher ARM	34
2.8.2	Flasher RX	34
2.8.3	Flasher PPC	34
3	Remote control	35
3.1	Overview	36
3.2	Handshake control	37
3.3	ASCII command interface	38
3.3.1	Introduction	38
3.3.2	General command and reply message format	38
3.3.3	Settings for ASCII interface via RS232	38
3.3.4	Settings for ASCII interface via Telnet	38
3.3.5	Commands to Flasher	39
3.3.6	Reply from Flasher	43
4	Performance	47
4.1	Performance of MCUs with internal flash memory	48
4.1.1	Flasher ARM	48

4.1.2	Flasher RX.....	48
4.1.3	Flasher PPC	48
5	Hardware	49
5.1	Flasher ARM 20-pin JTAG/SWD Connector	50
5.1.1	Pinout JTAG.....	50
5.1.2	Pinout SWD	51
5.1.3	Target power supply	51
5.2	Flasher RX 14-pin connector	53
5.2.1	Target power supply	53
5.3	Flasher PPC 14-pin connector.....	55
5.4	Target board design	56
5.4.1	Pull-up/pull-down resistors	56
5.4.2	RESET, nTRST	56
5.5	Adapters	57
5.5.1	JTAG Isolator.....	57
5.5.2	J-Link Needle Adapter	58
5.6	How to determine the hardware version	59
6	Support and FAQs	61
6.1	Contacting support.....	62
6.2	Frequently Asked Questions.....	63
7	Background information	65
7.1	Flash programming	66
7.1.1	How does flash programming via Flasher work ?.....	66
7.1.2	Data download to RAM.....	66
7.1.3	Available options for flash programming	66
8	Glossary.....	67
9	Literature and references.....	71

Chapter 1

Introduction

This chapter gives a short overview about the different models of the Flasher family and their features.

1.1 Flasher overview

Flasher is a programming tool for microcontrollers with on-chip or external flash memory. Flasher is designed for programming flash targets with the J-Flash software or stand-alone. In addition to that Flasher can also be used as a regular J-Link. For more information about J-Link in general, please refer to the *J-Link / J-Trace User Guide* which can be downloaded at <http://www.segger.com>.

Flasher connects to a PC using the USB/Ethernet/RS232 interface, running Microsoft Windows 2000, Windows XP, Windows 2003, Windows Vista, Windows 7 or Windows 8. In stand-alone mode, Flasher can be driven by the start/stop button, or via the RS232 interface (handshake control or ASCII interface). Flasher always has a 20-pin connector, which target interfaces are supported depends on the Flasher model:

- For Flasher ARM: JTAG and SWD are supported.
- For Flasher RX: JTAG is supported. Flasher comes with additional 14-pin RX adapter
- For Flasher PPC: JTAG is supported. Flasher comes with additional 14-pin PPC adapter.

1.1.1 Features of Flasher

- Three boot modes: J-Link mode, stand-alone mode, MSD mode
- Stand-alone JTAG/SWD programmer (Once set up, Flasher can be controlled without the use of PC program)
- No power supply required, powered through USB
- Supports internal and external flash devices
- 64 MB memory for storage of target program
- Can be used as J-Link (emulator) with a download speed of up to 720 Kbytes/second
- Serial in target programming supported
- Data files can updated via USB/Ethernet (using the J-Flash software), via RS232 or via the MSD functionality of Flasher

Flasher model	Supported cores	Supported target interfaces	Flash programming speed (depending on target hardware)
Flasher ARM	ARM7/ARM9/Cortex-M	JTAG, SWD	between 30-300 Kbytes/second
Flasher RX	Renesas RX610, RX621, RX62N, RX62T	JTAG	between 170 and 300 Kbytes/second
Flasher PPC	Power PC e200z0	JTAG	up to 138 Kbytes/second

1.1.2 Working environment

General

Flasher can operate from a PC with an appropriate software like J-Flash or in stand-alone mode.

Host System

IBM PC/AT or compatible CPU: 486 (or better) with at least 128MB of RAM, running Microsoft Windows 2000, Windows XP, Windows 2003, Windows Vista, Windows 7 or Windows 8. It needs to have a USB, Ethernet or RS232 interface available for communication with Flasher.

Power supply

Flasher requires 5V DC, min. 100mA via USB connector. If USB is not connected, the USB connector is used to power the device. Supply voltage is the same in this case. Please avoid excess voltage.

Installing Flasher PC-software J-Flash

The latest version of the J-Flash software, which is part of the J-Link software and documentation package, can always be downloaded from our website: http://www.segger.com/download_jlink.html. For more information about using J-Flash please refer to *UM08007_FlasherARM.pdf* (J-Flash user guide) which is also available for download on our website.

1.2 Specifications

1.2.1 Specifications for Flasher ARM

General	
Supported OS	Microsoft Windows 2000 Microsoft Windows XP Microsoft Windows XP x64 Microsoft Windows 2003 Microsoft Windows 2003 x64 Microsoft Windows Vista Microsoft Windows Vista x64 Microsoft Windows 7 Microsoft Windows 7 x64 Microsoft Windows 8 Microsoft Windows 8 x64
Operating Temperature	+5 °C ... +60 °C
Storage Temperature	-20 °C ... +60 °C
Relative Humidity (non-condensing)	<90% rH
Mechanical	
Size (without cables)	121mm x 66mm x 30mm
Weight (without cables)	119g
Available interfaces	
USB Host interface	USB 2.0, full speed
Ethernet Host interface	10/100 MBit
RS232 Host interface	RS232 9-pin
Target interface	JTAG 20-pin (14-pin adapter available)
JTAG Interface, Electrical	
Power Supply	USB powered, 100mA for Flasher ARM. 500 mA if target is powered by Flasher ARM
Target interface voltage (V _{IF})	1.2 ... 5V
Target supply voltage	4.5V...5V (on the 14-pin adapter the target supply voltage can be switched between 3.3V and 5V)
Target supply current	max. 400mA
Reset Type	Open drain. Can be pulled low or tristated
Reset low level output voltage (V _{OL})	V _{OL} ≤ 10% of V _{IF}
For the whole target voltage range (1.8V ≤ V _{IF} ≤ 5V)	
LOW level input voltage (V _{IL})	V _{IL} ≤ 40% of V _{IF}
HIGH level input voltage (V _{IH})	V _{IH} ≥ 60% of V _{IF}
For 1.8V ≤ V _{IF} ≤ 3.6V	
LOW level output voltage (V _{OL}) with a load of 10 kOhm	V _{OL} ≤ 10% of V _{IF}
HIGH level output voltage (V _{OH}) with a load of 10 kOhm	V _{OH} ≥ 90% of V _{IF}
For 3.6 ≤ V _{IF} ≤ 5V	
LOW level output voltage (V _{OL}) with a load of 10 kOhm	V _{OL} ≤ 20% of V _{IF}

Table 1.1: Flasher ARM specifications

HIGH level output voltage (V_{OH}) with a load of 10 kOhm	$V_{OH} \geq 80\%$ of V_{IF}
JTAG Interface, Timing	
Max. JTAG speed	up to 12MHz
Data input rise time (T_{rdi})	$T_{rdi} \leq 20\text{ns}$
Data input fall time (T_{fdi})	$T_{fdi} \leq 20\text{ns}$
Data output rise time (T_{rdo})	$T_{rdo} \leq 10\text{ns}$
Data output fall time (T_{fdo})	$T_{fdo} \leq 10\text{ns}$
Clock rise time (T_{rc})	$T_{rc} \leq 10\text{ns}$
Clock fall time (T_{fc})	$T_{fc} \leq 10\text{ns}$

Table 1.1: Flasher ARM specifications

1.2.1.1 Flasher ARM download speed

The following table lists Flasher ARM performance values (Kbytes/second) for writing to memory (RAM) via the JTAG interface:

Hardware	ARM7 memory download
Flasher ARM	720 Kbytes/s (12MHz JTAG)

Table 1.2: Download speed of Flasher ARM

Note: The actual speed depends on various factors, such as JTAG, clock speed, host CPU core etc.

1.2.2 Specifications for Flasher RX

General	
Supported OS	Microsoft Windows 2000 Microsoft Windows XP Microsoft Windows XP x64 Microsoft Windows 2003 Microsoft Windows 2003 x64 Microsoft Windows Vista Microsoft Windows Vista x64 Microsoft Windows 7 Microsoft Windows 7 x64 Microsoft Windows 8 Microsoft Windows 8 x64
Operating Temperature	+5 °C ... +60 °C
Storage Temperature	-20 °C ... +60 °C
Relative Humidity (non-condensing)	<90% rH
Mechanical	
Size (without cables)	121mm x 66mm x 30mm
Weight (without cables)	119g
Available interfaces	
USB Host interface	USB 2.0, full speed
Ethernet Host interface	10/100 MBit
RS232 Host interface	RS232 9-pin
Target interface	JTAG 20-pin (shipped with 14-pin adapter for Renesas RX)
JTAG Interface, Electrical	
Power Supply	USB powered, 100mA for Flasher ARM. 500 mA if target is powered by Flasher ARM
Target interface voltage (V _{IF})	1.2 ... 5V
Target supply voltage	4.5V...5V (on the 14-pin adapter the target supply voltage can be switched between 3.3V and 5V)
Target supply current	max. 400mA
Reset Type	Open drain. Can be pulled low or tristated
Reset low level output voltage (V _{OL})	V _{OL} ≤ 10% of V _{IF}
For the whole target voltage range (1.8V ≤ V _{IF} ≤ 5V)	
LOW level input voltage (V _{IL})	V _{IL} ≤ 40% of V _{IF}
HIGH level input voltage (V _{IH})	V _{IH} ≥ 60% of V _{IF}
For 1.8V ≤ V _{IF} ≤ 3.6V	
LOW level output voltage (V _{OL}) with a load of 10 kOhm	V _{OL} ≤ 10% of V _{IF}
HIGH level output voltage (V _{OH}) with a load of 10 kOhm	V _{OH} ≥ 90% of V _{IF}
For 3.6 ≤ V _{IF} ≤ 5V	
LOW level output voltage (V _{OL}) with a load of 10 kOhm	V _{OL} ≤ 20% of V _{IF}
HIGH level output voltage (V _{OH}) with a load of 10 kOhm	V _{OH} ≥ 80% of V _{IF}

Table 1.3: Flasher RX specifications

JTAG Interface, Timing	
Max. JTAG speed	up to 12MHz
Data input rise time (T_{rdi})	$T_{rdi} \leq 20\text{ns}$
Data input fall time (T_{fdi})	$T_{fdi} \leq 20\text{ns}$
Data output rise time (T_{rdo})	$T_{rdo} \leq 10\text{ns}$
Data output fall time (T_{fdo})	$T_{fdo} \leq 10\text{ns}$
Clock rise time (T_{rc})	$T_{rc} \leq 10\text{ns}$
Clock fall time (T_{fc})	$T_{fc} \leq 10\text{ns}$

Table 1.3: Flasher RX specifications

1.2.2.1 Flasher RX download speed

The following table lists Flasher RX performance values (Kbytes/second) for writing to memory (RAM) via the JTAG interface:

Hardware	Flasher RX600 series memory download
Flasher RX	720 Kbytes/s (12MHz JTAG)

Table 1.4: Download speed of Flasher RX

Note: The actual speed depends on various factors, such as JTAG, clock speed, host CPU core etc.

1.2.3 Specifications for Flasher PPC

General	
Supported OS	Microsoft Windows 2000 Microsoft Windows XP Microsoft Windows XP x64 Microsoft Windows 2003 Microsoft Windows 2003 x64 Microsoft Windows Vista Microsoft Windows Vista x64 Microsoft Windows 7 Microsoft Windows 7 x64 Microsoft Windows 8 Microsoft Windows 8 x64
Operating Temperature	+5 °C ... +60 °C
Storage Temperature	-20 °C ... +60 °C
Relative Humidity (non-condensing)	<90% rH
Mechanical	
Size (without cables)	121mm x 66mm x 30mm
Weight (without cables)	119g
Available interfaces	
USB Host interface	USB 2.0, full speed
Ethernet Host interface	10/100 MBit
RS232 Host interface	RS232 9-pin
Target interface	JTAG 20-pin (shipped with 14-pin adapter for Renesas PPC)
JTAG Interface, Electrical	
Power Supply	USB powered, 100mA for Flasher ARM. 500 mA if target is powered by Flasher ARM
Target interface voltage (V _{IF})	1.2 ... 5V
Target supply voltage	4.5V...5V (on the 14-pin adapter the target supply voltage can be switched between 3.3V and 5V)
Target supply current	max. 400mA
Reset Type	Open drain. Can be pulled low or tristated
Reset low level output voltage (V _{OL})	V _{OL} ≤ 10% of V _{IF}
For the whole target voltage range (1.8V ≤ V _{IF} ≤ 5V)	
LOW level input voltage (V _{IL})	V _{IL} ≤ 40% of V _{IF}
HIGH level input voltage (V _{IH})	V _{IH} ≥ 60% of V _{IF}
For 1.8V ≤ V _{IF} ≤ 3.6V	
LOW level output voltage (V _{OL}) with a load of 10 kOhm	V _{OL} ≤ 10% of V _{IF}
HIGH level output voltage (V _{OH}) with a load of 10 kOhm	V _{OH} ≥ 90% of V _{IF}
For 3.6 ≤ V _{IF} ≤ 5V	
LOW level output voltage (V _{OL}) with a load of 10 kOhm	V _{OL} ≤ 20% of V _{IF}
HIGH level output voltage (V _{OH}) with a load of 10 kOhm	V _{OH} ≥ 80% of V _{IF}

Table 1.5: Flasher PPC specifications

JTAG Interface, Timing	
Max. JTAG speed	up to 12MHz
Data input rise time (T_{rdi})	$T_{rdi} \leq 20\text{ns}$
Data input fall time (T_{fdi})	$T_{fdi} \leq 20\text{ns}$
Data output rise time (T_{rdo})	$T_{rdo} \leq 10\text{ns}$
Data output fall time (T_{fdo})	$T_{fdo} \leq 10\text{ns}$
Clock rise time (T_{rc})	$T_{rc} \leq 10\text{ns}$
Clock fall time (T_{fc})	$T_{fc} \leq 10\text{ns}$

Table 1.5: Flasher PPC specifications

1.2.3.1 Flasher PPC download speed

The following table lists Flasher PPC performance values (Kbytes/second) for writing to memory (RAM) via the JTAG interface:

Hardware	Memory download
Flasher PPC	530 Kbytes/s (8 MHz JTAG)

Table 1.6: Download speed of Flasher PPC

Note: The actual speed depends on various factors, such as JTAG, clock speed, host CPU core etc.

Chapter 2

Working with Flasher

This chapter describes functionality and how to use Flasher.

2.1 Setting up the IP interface

Since hardware version 3 Flasher family comes with an additional Ethernet interface to communicate with the host system. These Flashers also come with a built-in web-server which allows some basic setup of the emulator, e.g. configuring a default gateway which allows using it even in large intranets.

2.1.1 Connecting the first time

When connecting Flasher the first time, it attempts to acquire an IP address via DHCP. The recommended way for finding out which IP address has been assigned to Flasher is, to use the J-Link Configurator. The J-Link Configurator is a small GUI-based utility which shows a list of all emulators that are connected to the host PC via USB and Ethernet. For more information about the J-Link Configurator, please refer to *UM08001_JLinkARM.pdf* (J-Link / J-Trace user guide), chapter *Setup*, section *J-Link Configurator*. The setup of the IP interface of Flasher is the same as for other emulators of the J-Link family. For more information about how to setup the IP interface of Flasher, please refer to *UM08001, J-Link / J-Trace User Guide*, chapter *Setup*, section *Setting up the IP interface*. For more information about how to use Flasher via Ethernet or prepare Flasher via Ethernet for stand-alone mode, please refer to *Operating modes* on page 19.

2.2 Operating modes

Flasher is able to boot in 3 different modes:

- J-Link mode
- Stand-alone mode
- MSD (Mass storage device) mode

If Flasher can establish an Ethernet uplink or can enumerate on the USB port, it boots in "J-Link mode". In this mode, Flasher can be used as a J-Link. When supply power is enabled and Flasher can not establish a connection with the host, the "stand-alone mode" is started. In this mode Flasher can be used as a stand-alone flash programmer. When the Start/Stop button is pressed when power supply is enabled, Flasher boots in "MSD mode". In this mode, Flasher boots as a mass storage device.

2.2.1 J-Link mode

If you want to use Flasher for the first time you need to install the J-Link software and documentation package. After installation, connect Flasher to the host PC via USB or Ethernet. For more information about how to install the J-Link software and documentation package please refer to the *J-Link / J-Trace User Guide, chapter Setup* which can be downloaded from http://www.segger.com/download_jlink.html.

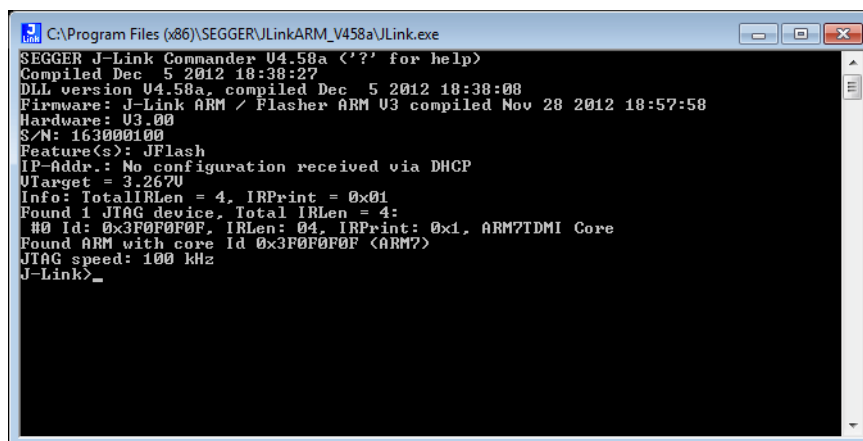
2.2.1.1 Connecting the target system

Power-on sequence

In general, Flasher should be powered on before connecting it with the target device. That means you should first connect Flasher with the host system via USB / Ethernet and then connect Flasher with the target device via JTAG or SWD. Power-on the device after you connected Flasher to it. Flasher will boot in "J-Link mode".

Verifying target device connection with J-Link.exe

If the USB driver is working properly and your Flasher is connected with the host system, you may connect Flasher to your target hardware. Then start the J-Link command line tool `JLink.exe`, which should now display the normal Flasher related information and in addition to that it should report that it found a JTAG target and the targets core ID. The screenshot below shows the output of `JLink.exe`.



```

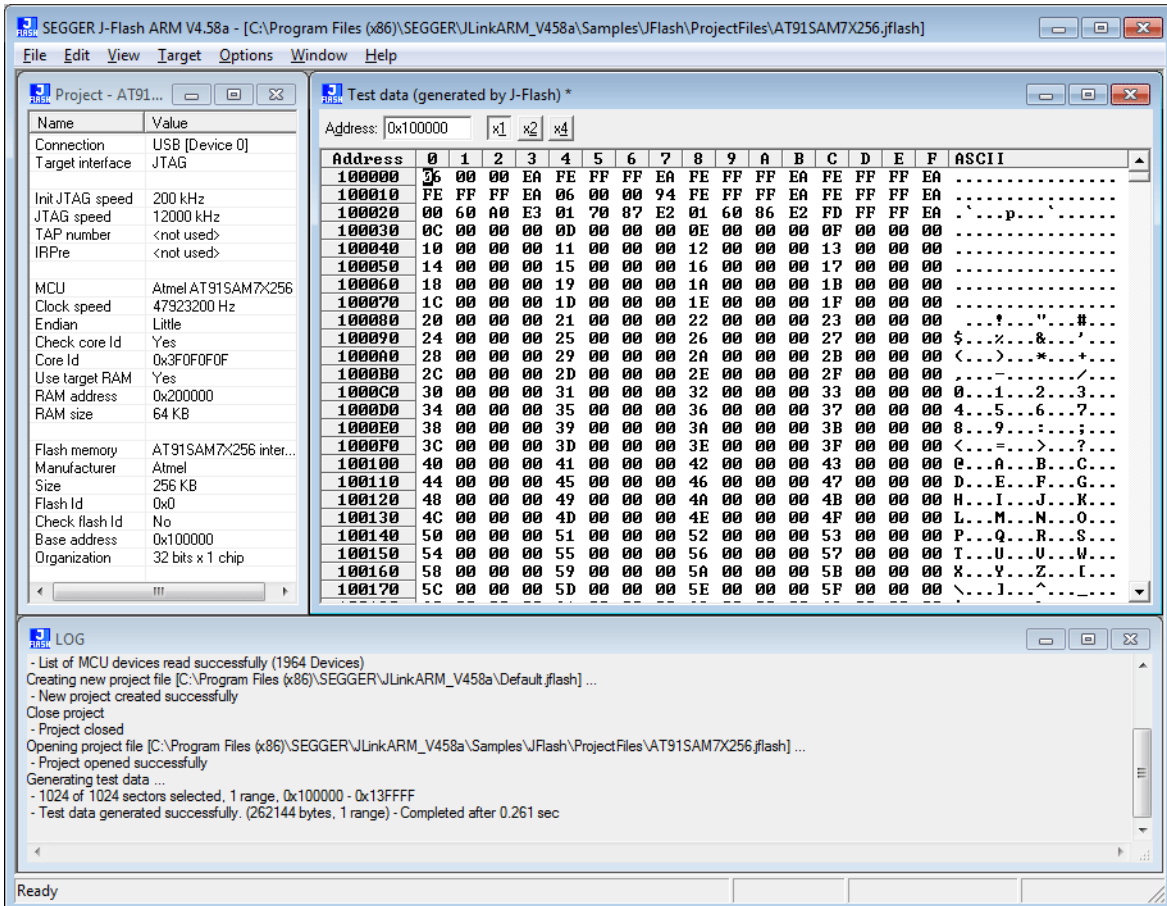
C:\Program Files (x86)\SEGGER\JLinkARM_V458a\JLink.exe
SEGGER J-Link Commander V4.58a ('?' for help)
Compiled Dec  5 2012 18:38:27
DLL version V4.58a, compiled Dec  5 2012 18:38:08
Firmware: J-Link ARM / Flasher ARM V3 compiled Nov 28 2012 18:57:58
Hardware: V3-00
S/N: 163000100
Feature(s): JFlash
IP-Addr.: No configuration received via DHCP
UTarget = 3.267U
Info: TotalIRLen = 4, IRPrint = 0x01
Found 1 JTAG device, Total IRLen = 4:
#0 Id: 0x3F0F0F0F, IRLen: 04, IRPrint: 0x1, ARM7TDMI Core
Found ARM with core Id 0x3F0F0F0F (ARM7)
JTAG speed: 100 kHz
J-Link>_
  
```

2.2.1.2 Setting up Flasher for stand-alone mode

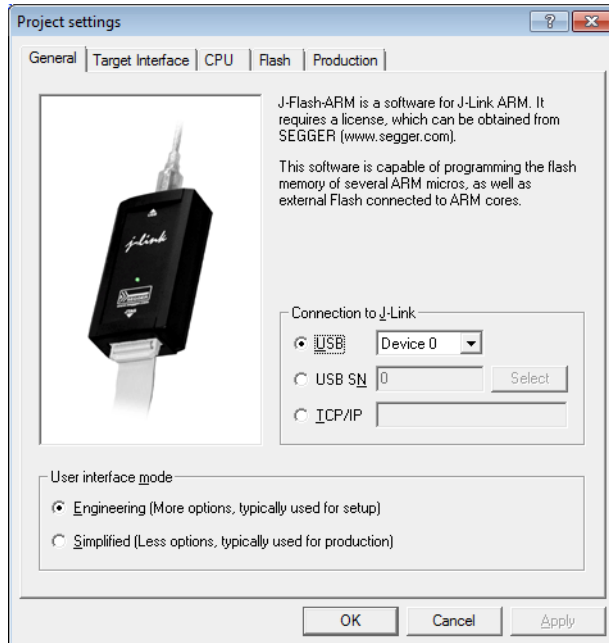
In order to set up Flasher for the stand-alone mode it needs to be configured once using the J-Flash software. For more information about J-Flash, please refer to the *J-Flash User Guide*.

After starting J-Flash, open the appropriate J-Flash project for the target Flasher shall be configured for, by selecting **File -> Open Project**. If J-Flash does not come with an appropriate sample project for the desired hardware, a new project needs to be created by selecting **File -> New Project**.

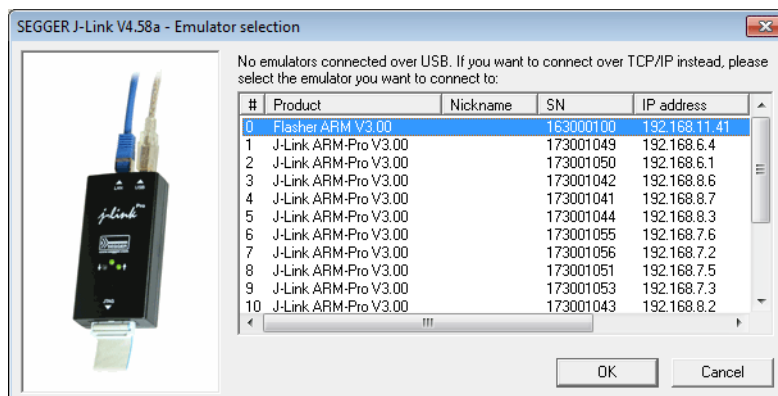
After the appropriate project has been opened / created, the data file which shall be programmed needs to be loaded, by selecting **File -> Open**. After this J-Flash should look like in the screenshot below.



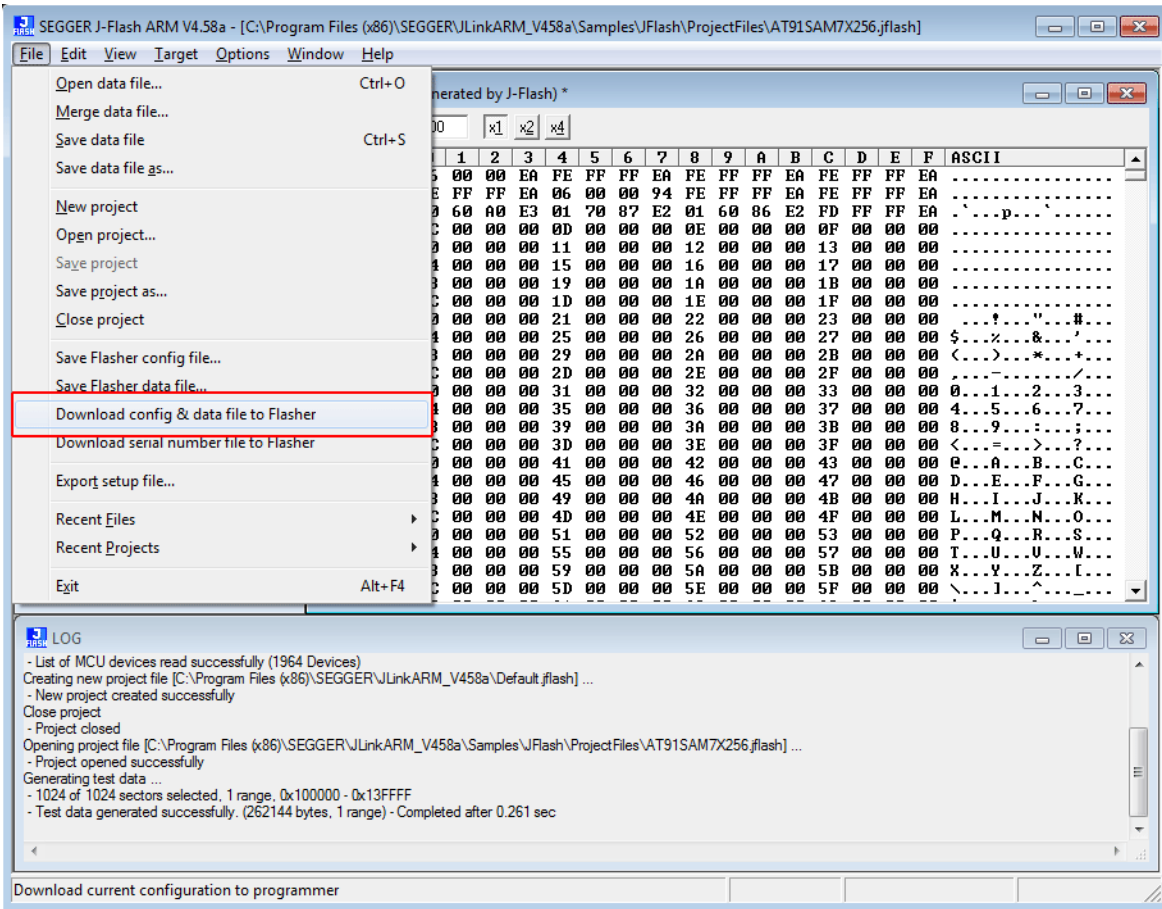
Before downloading the configuration (project) and program data (data file) to Flasher, the connection type (USB/IP) needs to be selected in the project. These settings are also saved on a per-project basis, so this also only needs to be setup once per J-Flash project. The connection dialog is opened by clicking **Options -> Project settings -> General**.



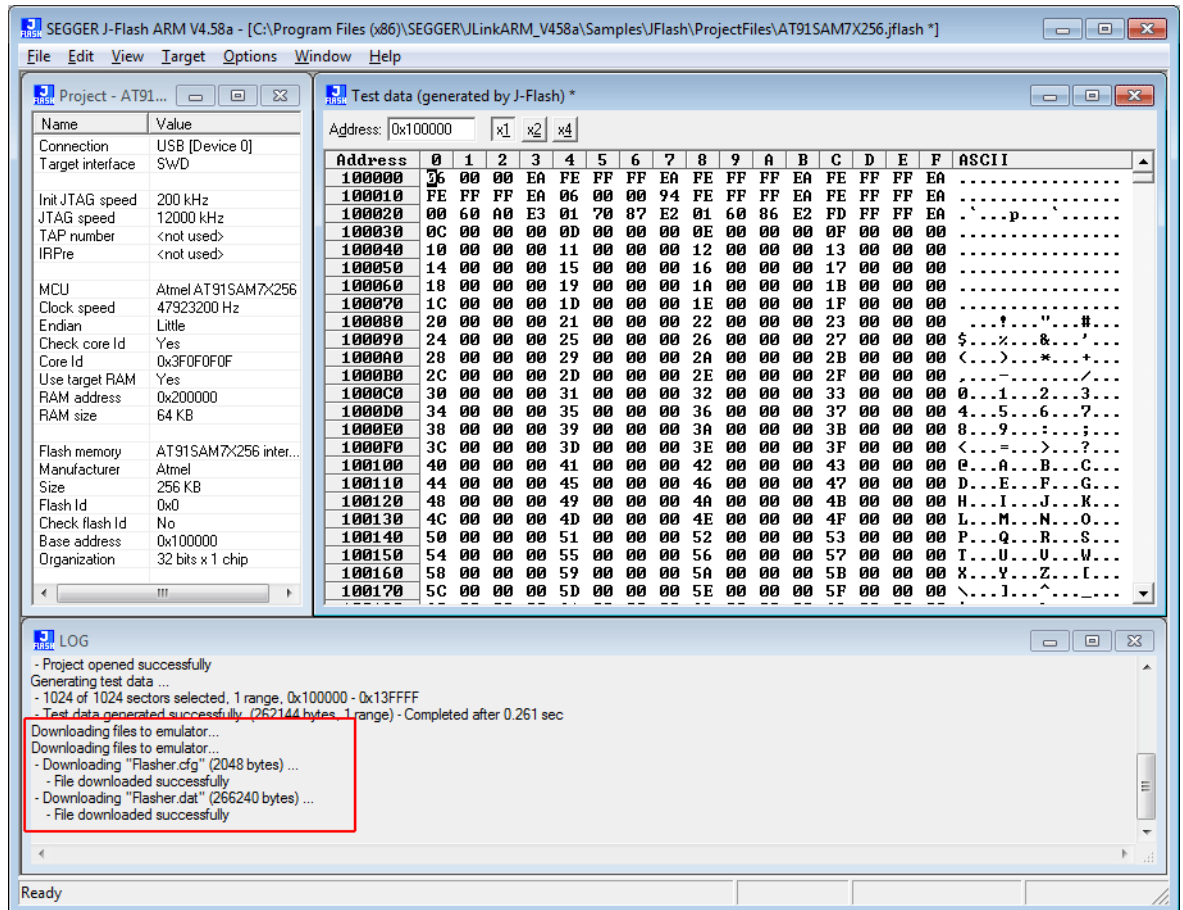
The connection dialog allows the user to select how to connect to Flasher. When connecting to a Flasher via TCP/IP it is not mandatory to enter an IP address. If the field is left blank and **File->Download to programmer** is selected, an emulator selection dialog pops up which shows all Flasher which have been found on the network. The user then can simply select the Flash he wants to download the configuration to.



In order to download the configuration and program data to the Flasher, simply select **File -> Download config & data file to Flasher.**



The J-Flash log window indicates that the download to the emulator was successful.



From now on, Flasher can be used in stand-alone mode (without host PC interaction) for stand-alone programming.

2.2.2 Stand-alone mode

In order to use Flasher in "stand-alone mode", it has to be configured first, as described in *Setting up Flasher for stand-alone mode* on page 19. To boot Flasher in the "stand-alone mode", only the power supply to Flasher has to be enabled (Flasher should not be connected to a PC). In the "stand-alone mode" Flasher can be used as a stand-alone flash programmer.

Note: Flasher can only program the target device it was configured for. In order to program another target device, you have to repeat the steps described in *Setting up Flasher for stand-alone mode* on page 19.

2.2.2.1 LED status indicators

Progress and result of an operation is indicated by Flasher's LEDs:

#	Status of LED	Meaning
0	GREEN high frequency blinking (On/Off time: 50ms => 10Hz)	Flasher ARM is waiting for USB enumeration or ethernet link. As soon as USB has been enumerated or ethernet link has been established, the green LED stops flashing and is switched to constant green. In stand-alone-mode, Flasher remains in the high frequency blinking state until state #1 is reached. Flasher goes to state #1 as soon as a #START command has been received via the ASCII interface or the Start button has been pushed.
1	GREEN constant	Connect to target and perform init sequence.
2	GREEN slow blinking	Flashing operation in progress: <ol style="list-style-type: none"> 1. Erasing (slow blinking on/off time: 80 ms => 6.25 HZ) 2. Programming (slow blinking on/off time: 300ms => ~1.67 Hz) 3. Verifying (slow blinking, on/off time: 100ms => 5 Hz)
3	GREEN constant	Operation successful. Goes back to state #0 automatically.
4	RED constant	Operation failed. Goes back to state #0 automatically but red LED remains on until state #1 (next programming cycle) is entered again.

Table 2.1: Flasher LEDs

2.2.3 MSD mode

When pressing the Start/Stop button of Flasher while connecting it to the PC, Flasher will boot in the "MSD mode". This mode can be used to downdate a Flasher firmware version if a firmware update did not work properly and it can be used to configure Flasher for the "stand-alone mode", without using J-Flash.

If Flasher has been configured for "stand-alone mode" as described in the section above, there will be four files on the MSD, `FLASHER.CFG`, `FLASHER.DAT`, `FLASHER.LOG`, `SERIAL.TXT`.



`FLASHER.CFG` contains the configuration settings for programming the target device and `FLASHER.DAT` contains the data to be programmed. `FLASHER.LOG` contains all logging information about the commands, performed in stand-alone mode. The `SERIAL.TXT` contains the serial number, which will be programmed next. J-Flash supports to configure Flasher for automated serial number programming. For further information about how to configure

Currently, J-Flash does not support to configure Flasher for automated serial number programming.

If you want to configure multiple Flasher for the same target you do not have to use J-Flash all the time. It is also possible to copy the `FLASHER.CFG` and the `FLASHER.DAT` files from a configured Flasher to another one. To copy these files boot Flasher in "MSD mode".

2.3 Multiple File Support

It is also possible to have multiple data files and config files on Flasher, to make Flasher more easy to use in production environment. To choose the correct configuration file and data file pair, a `FLASHER.INI` file is used. This init file contains a `[FILES]` section which describes which configuration file and which data file should be used for programming. A sample content of a `FLASHER.INI` file is shown below:

```
[FILES]
DataFile = "Flasher1.dat"
ConfigFile = "Flasher1.cfg"
```

Using this method all configuration files and data files which are used in the production only have to be downloaded once. From there on a configuration file / data file pair can be switched by simply replacing the `FLASHER.INI` by a new one, which contains the new descriptions for the configuration file and data file. The `FLASHER.INI` can be replaced in two ways:

1. Boot Flasher in MSD mode in order to replace the `FLASHER.INI`
2. If Flasher is already integrated into the production line, runs in stand-alone mode and can not be booted in other mode: Use the file I/O commands provided by the ASCII interface of Flasher, to replace the `FLASHER.INI`. For more information about the file I/O commands, please refer to *File I/O commands* on page 42.