Flasher ATE

Production Programmer

Getting Started

Document: AN08007
Manual Version: 1.0
Revision: e
Date: August 19, 2021

A product of SEGGER Microcontroller GmbH

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 (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

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

Ecolab-Allee 5
D-40789 Monheim am Rhein

Germany

Tel. +49-2173-99312-0
Fax. +49-2173-99312-28
E-mail: support@segger.com
Internet: www.segger.com

Manual versions

This manual describes the Flasher device. For further information on topics or routines not yet specified, please contact us.

Print date: August 19, 2021

Manual Version	Revision	Date	By	Description
1.00	e	210819	JB	Corrections after review
1.00	d	180109	MF	Update filename information (remove 8.3 filename limitation)
1.00	c	180109	MF	Corrected FTP connection picture
1.00	b	180109	MF	Minor changes after review.
1.00	a	171219	MF	Minor changes after review.
1.00	0	171218	MF	Minor changes after review.
0.01	0	171217	MF	Initial version.

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.

Table of contents

Chapter 1

Introduction

This documents guides you through the first steps with the Flasher ATE. The goal is to have a first project running with the Flasher ATE. Please take care of the prerequisites listed in chapter *Prerequisites* on page 10.

The Flasher ATE is designed for mass production purpose. If you are familiar with other SEGGER Flasher products and the J-Flash-Tool, migration will be easy.

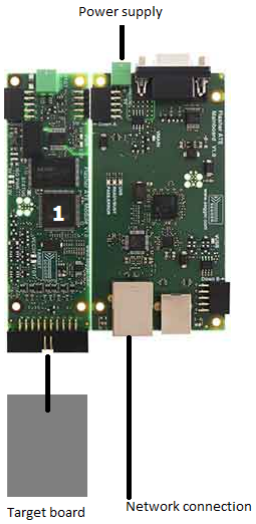
Chapter 2

Preparing for handshake and remote-controlled mode

2.1 Prerequisites

2.1.1 Hardware prerequisites

- The Flasher ATE setup must contain a main board and at least one module board.
- The Flasher ATE needs to be powered.
- The target board needs to be connected to the module board.
- The Flasher ATE needs to be connected to the network. J-Link mode via USB is currently not supported.



2.1.2 Software prerequisites

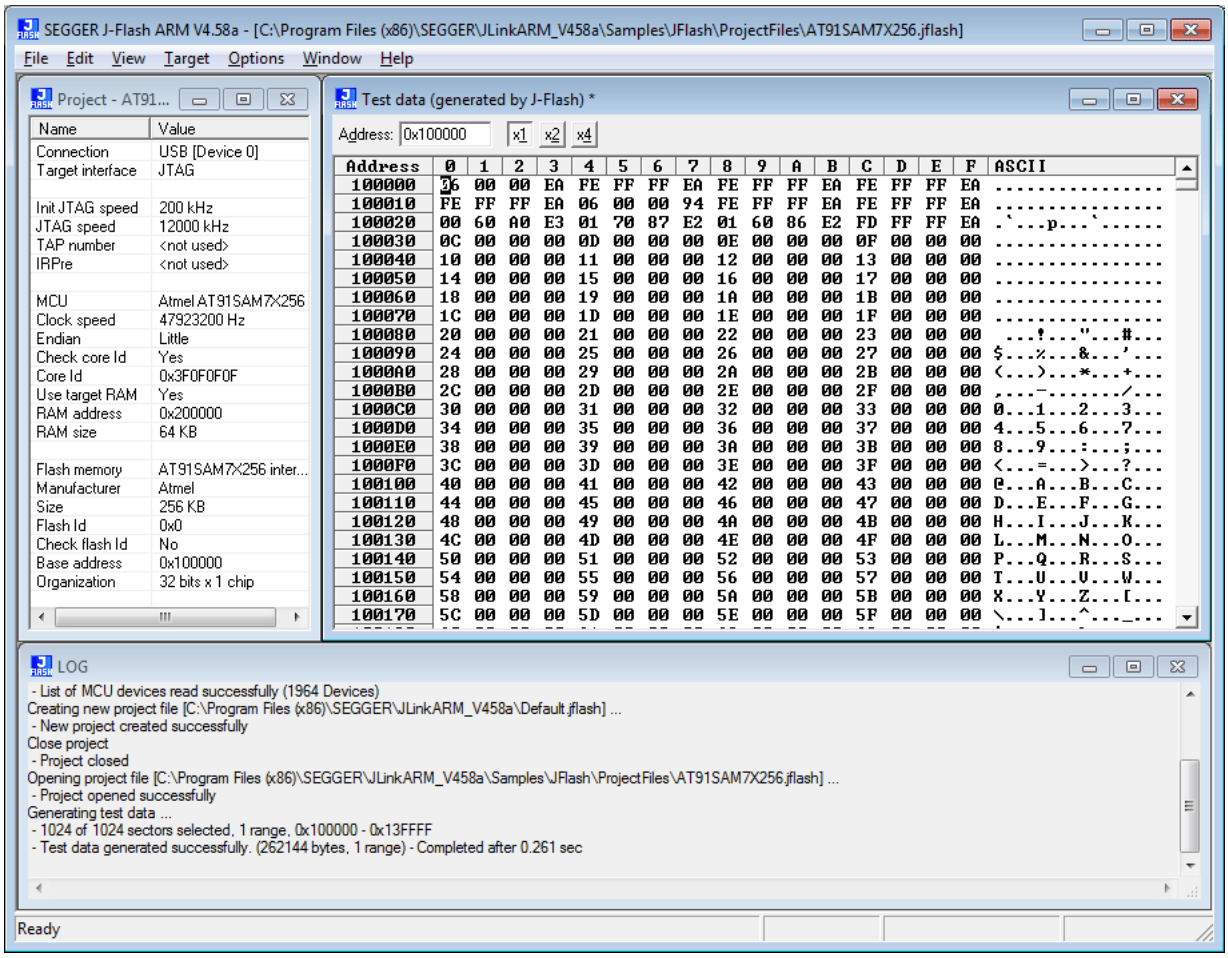
- You need the J-Flash to generate the Flasher ATE project files and J-Link Configurator tool. These are available for download at: *Flasher Downloads* . The tools are included in the Flasher Software and Documentation pack.
- The firmware or a test program for the target device must be available.

2.2 Setting up the project files

In order to set up Flasher for the handshake or remote-controlled mode it needs to be configured once using the J-Flash software. For more information about J-Flash, please refer to the *J-Flash Manual*

After starting J-Flash, create a J-Flash project for the target Flasher, by selecting **File -> New Project**.

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

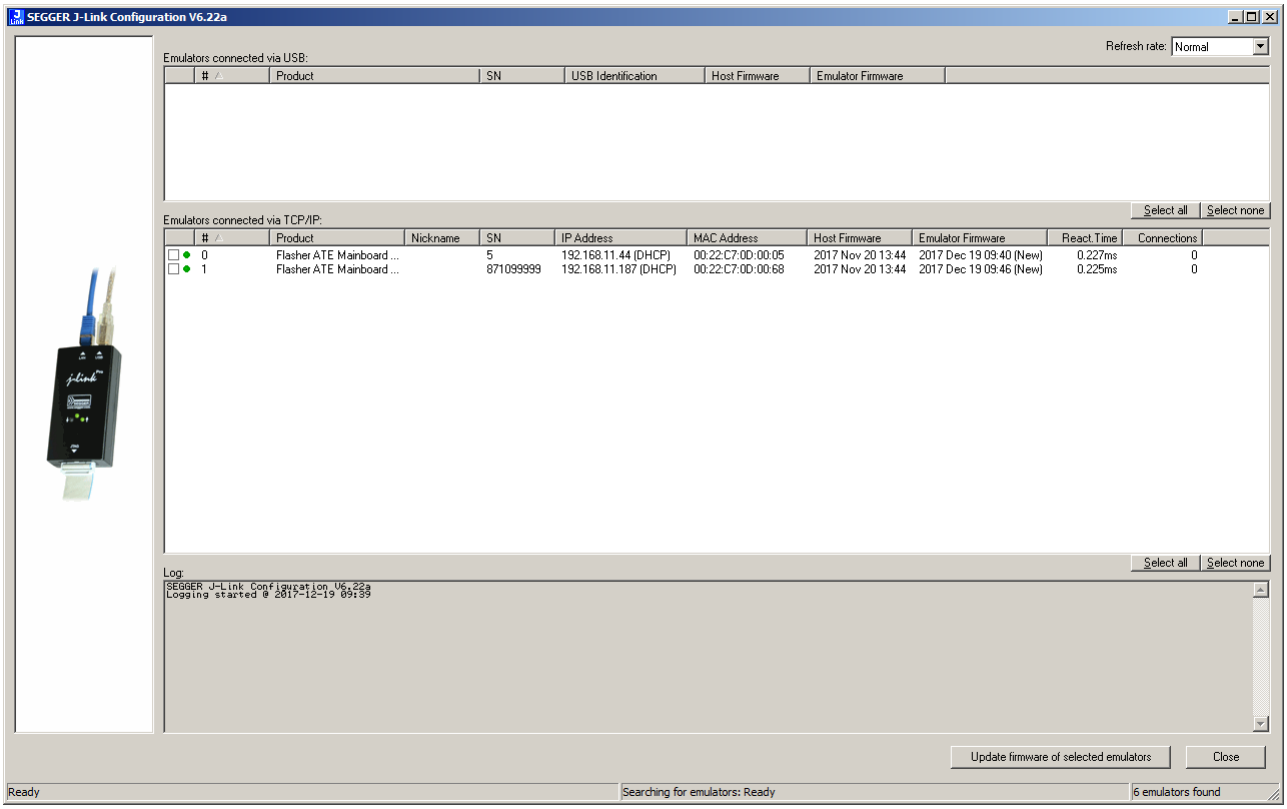


Follow the instruction of the J-Flash User Guide to setup your project. You can find further details in chapter 3 “Getting Started” of the J-Flash documentation.

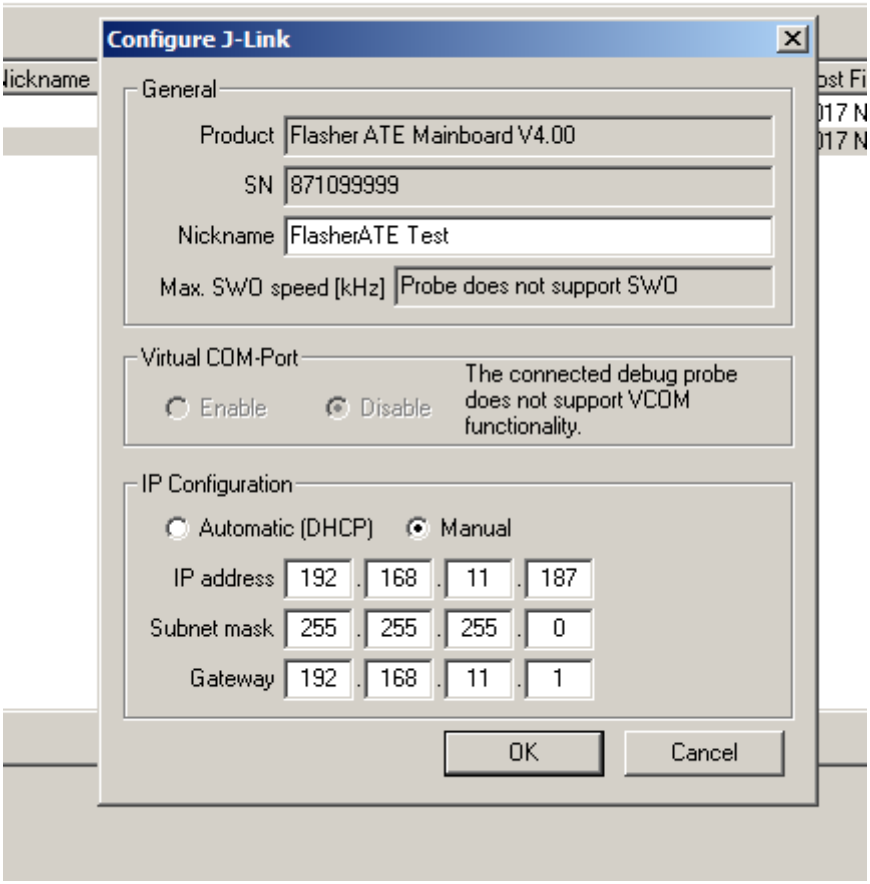
For the Flasher ATE it is required to save the project and the datafile to the hard drive. Therefore select **File** -> **Save Flasher config file** to save the project configuration file and afterwards **File** -> **Save Flasher data file...** to save the project data file.

2.2.1 Setting up the network configuration

Before downloading the configuration (project) and program data (data file) to Flasher, the Flasher IP settings need to be configured. This can be done by the J-Link Configurator tool. After starting the JLinkConfig executable the screen should look like this:

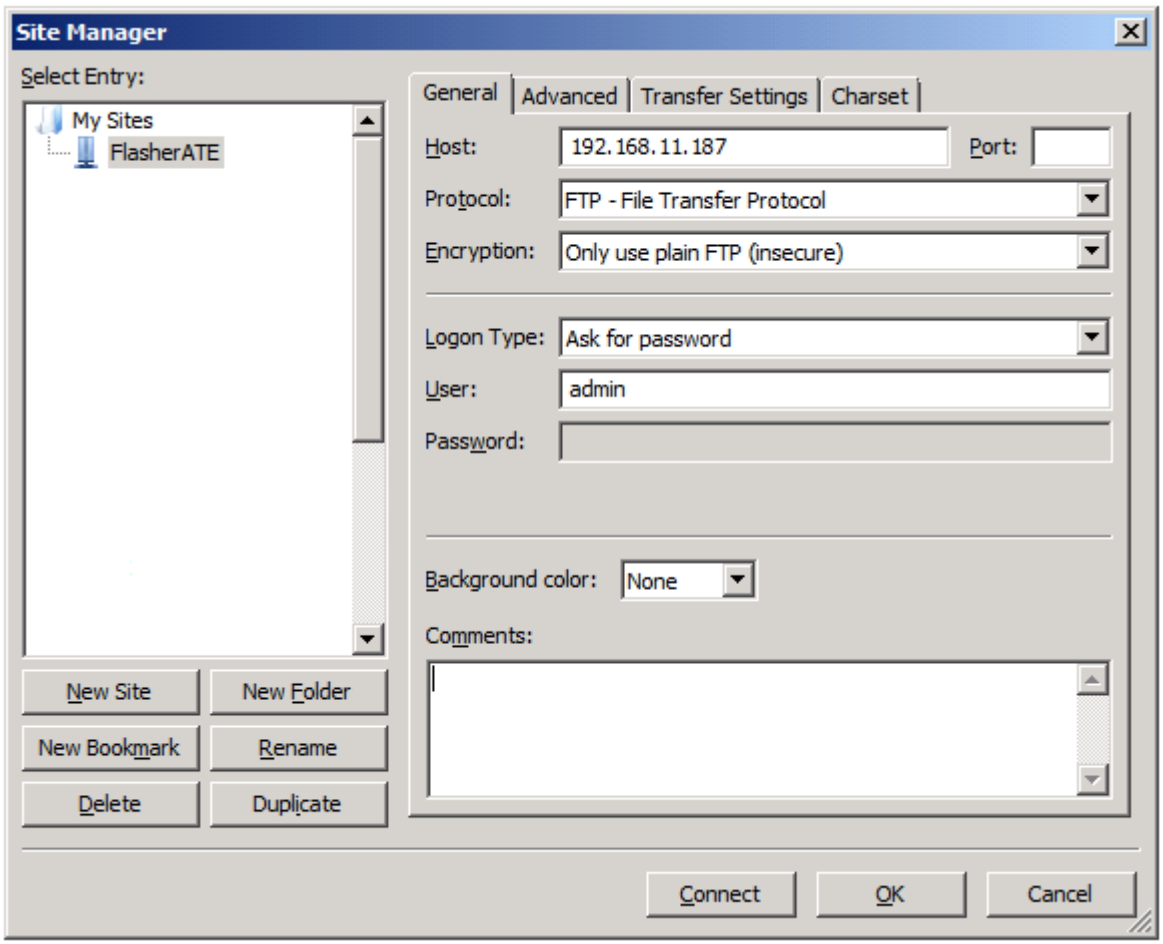


Now right-click on your Flasher ATE, choose **Configure** from the context menu and enter your required IP settings in the dialog box that opens up. E.g. like shown in the next picture. Confirm the settings by pressing OK.



2.2.2 Loading the project to the Flasher ATE

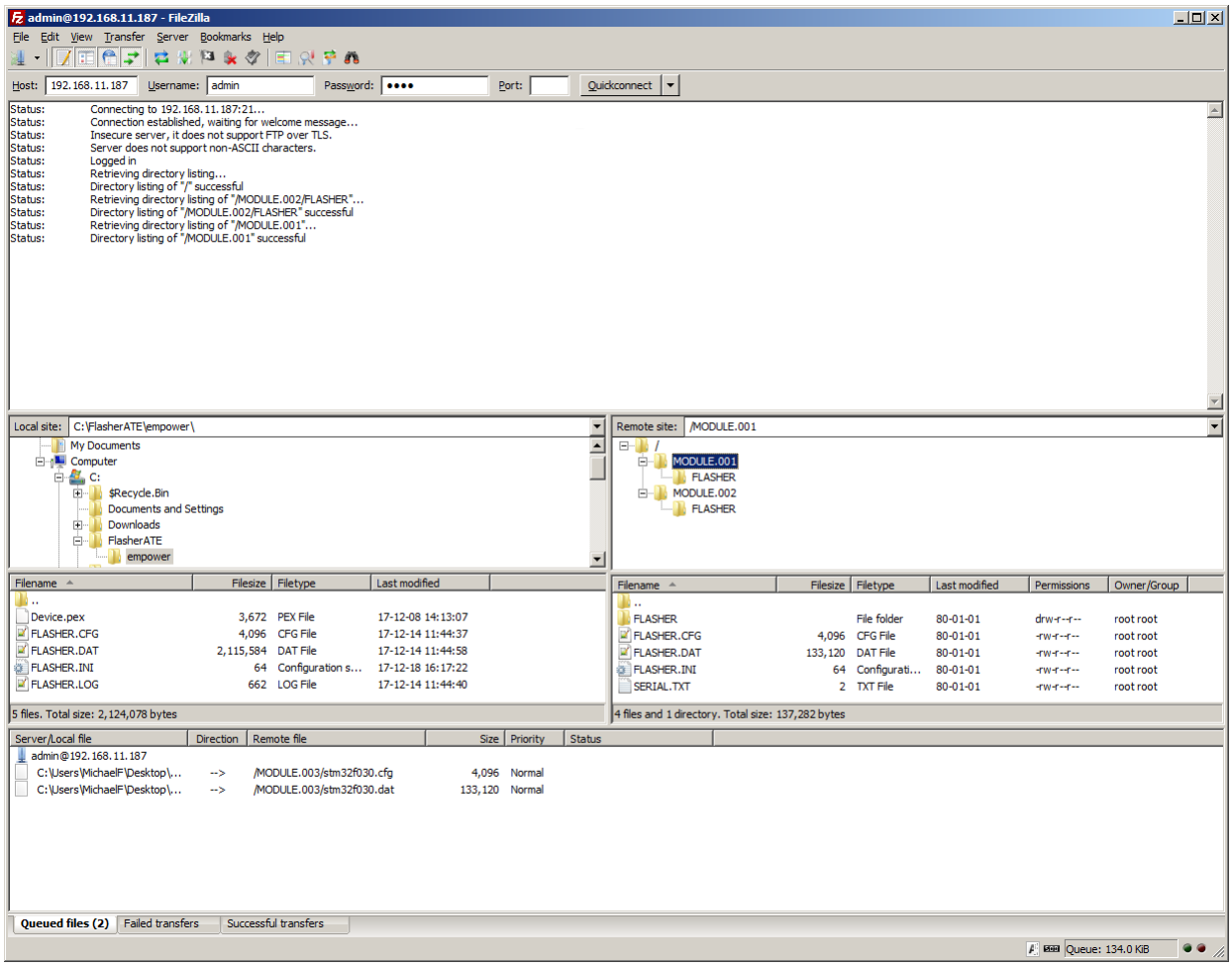
Next we need an FTP client. In the example below we are using the FileZilla FTP client but any other FTP client is fine as well. Create a new connection for the Flasher ATE by choosing **File** -> **Site Manager** and enter the connection data. See the image below. The user name for the FTP connection is `admin` and the password is `1234`.



And connect to the Flasher ATE.

> **Note**
>
> The user name and password is currently only intended for preventing changing the setup by accident. It is not to prevent hackers from accessing the Flasher ATE if it is connected to a network. The password cannot be changed.

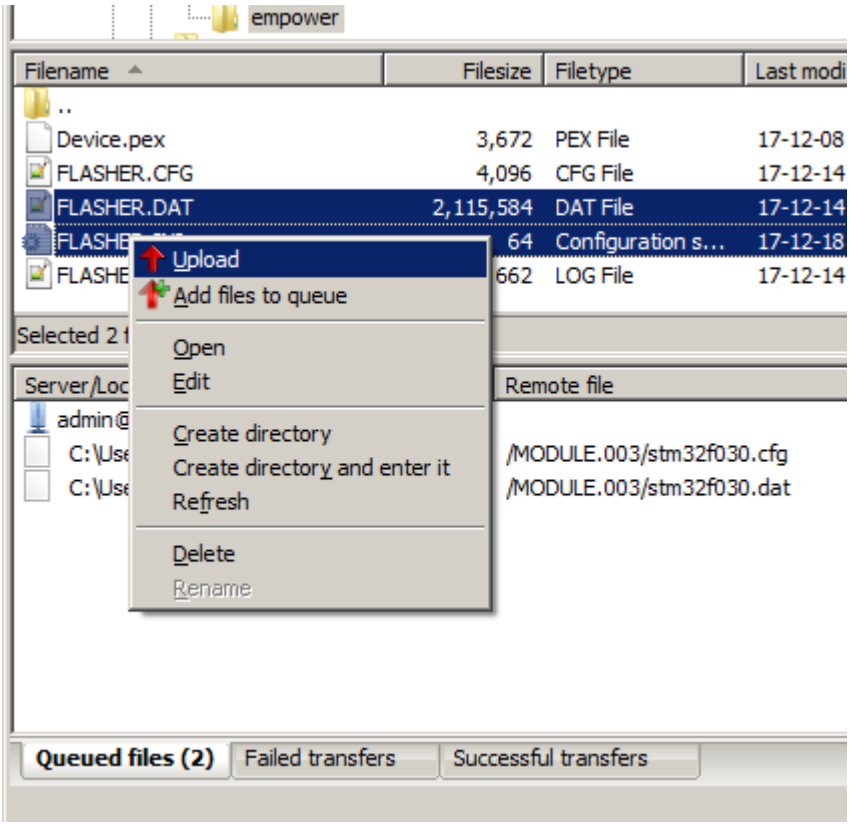
In the next step the project data needs to be loaded to each of the Flasher ATE modules. Therefore navigate to the project files created by the J-Flash in the section *Setup the project files* on page . The result should look like the next picture.



Next upload the files to the modules. Select the module.xyz folder in the remote site view. Then select the files:

- flasher.cfg and
- flasher.dat.

Choose **Upload** from the context menu to upload them to the module folder.



> **Note**
>
> This step needs to be repeated for every connected module.

If your device needs one or more script files (file extension .pex) you must upload them to the sub-folder **ProjectName** in the module's folder. The sub-folder has to be named the same as the config file. So in our example case it is "Flasher"

Now everything is prepared for programming the targets.

Chapter 3

Programming the targets

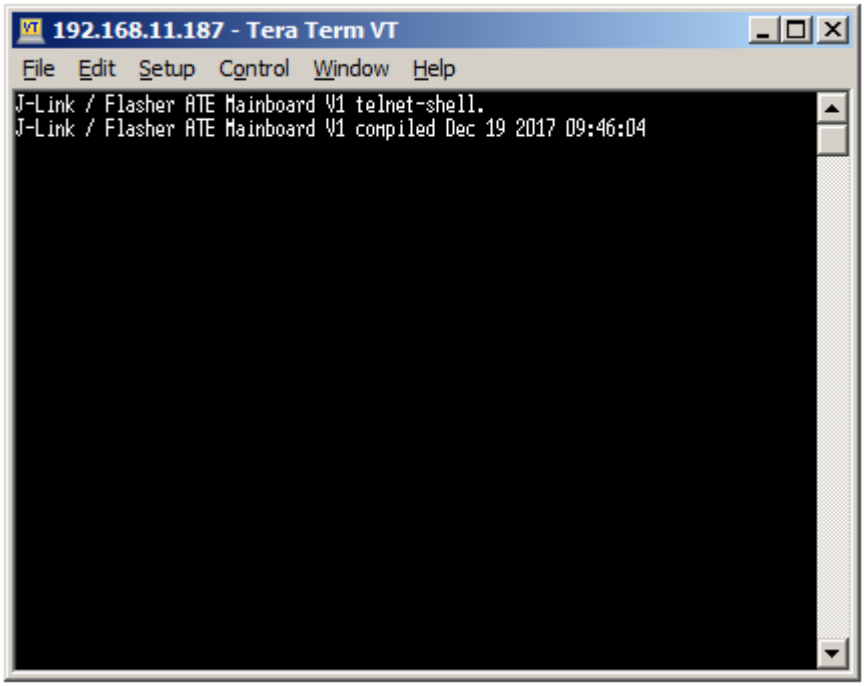
3.1 Overview

There are 3 ways to control Flasher ATE operation:

- Terminal communication via Telnet.
- Terminal communication via RS232.
- Via Handshake lines: 3 lines on the serial interface are used:
  1 line is an input and can be used to start operation,
  2 lines are outputs and serve as busy and status signals.

3.1.1 Starting the programming sequence via telnet protocol

A client application can connect to the Flasher ATE via Telnet on port 23. Find below a screen-shot of the Flasher which is remote controlled via Telnet using the TeraTerm terminal tool.



You can check if the Flasher ATE is ready to accept commands using the command `#status`. If yes the answer is `#STATUS:READY`.

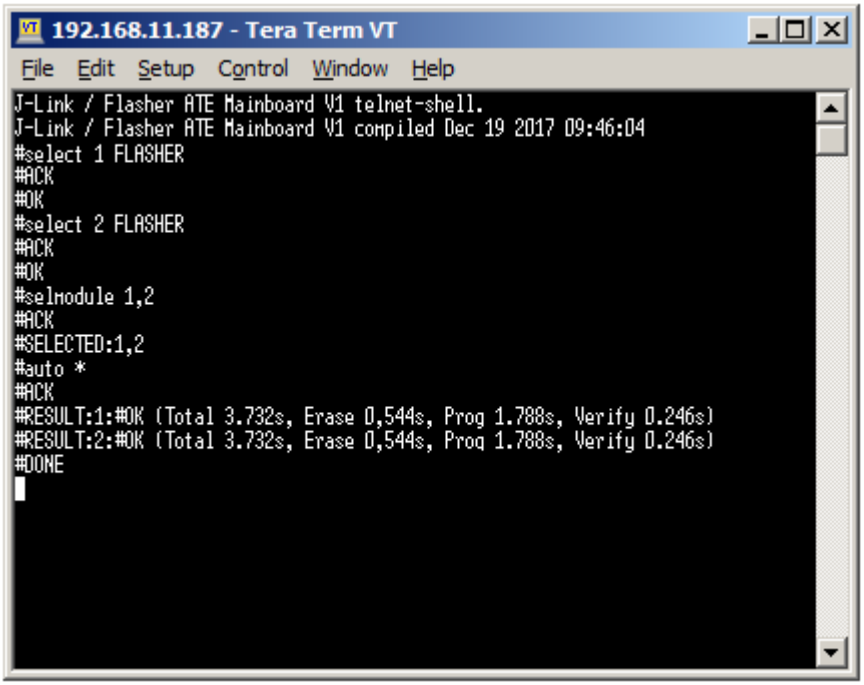
First, the project file that shall be flashed. Use the command: `#select [modulenumber] [project file name]`, e.g. `#select 1 FLASHER`. Repeat the step for every module.

> **Note**
>
> The file extensions are added by the firmware.

Second, select the module(s) which shall be used for programming, using the command: `#selmodule [1st modulenumber], [2nd modulenumber]…`, e.g. `#selmodule 1,2`.

Third, start the programming with the command `#auto *`.

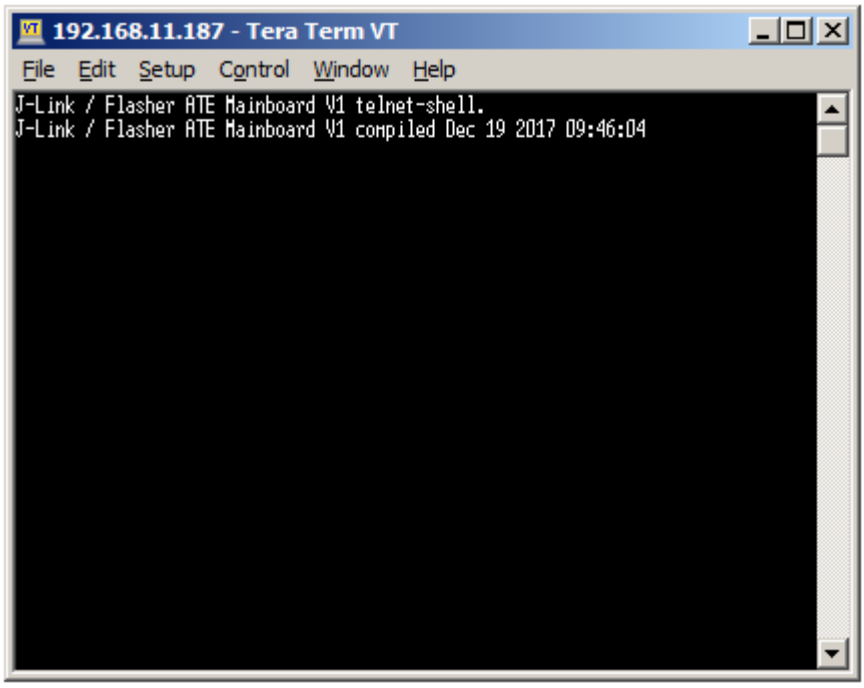
The picture shows the terminal after executing the commands.



3.1.2 Starting the programming sequence via RS232

To use the RS232 connection for remote control of the Flasher ATE open a terminal tool, e.g. TeraTerm, and open the serial connection to the Flasher ATE. The serial port settings are:

- baud rate 9600 bits/s,
- data bits 8,
- parity none and
- stop bits 1.



You can check if the Flasher ATE is ready to accept commands using the command `#status`. If yes the answer is `#STATUS:READY`.

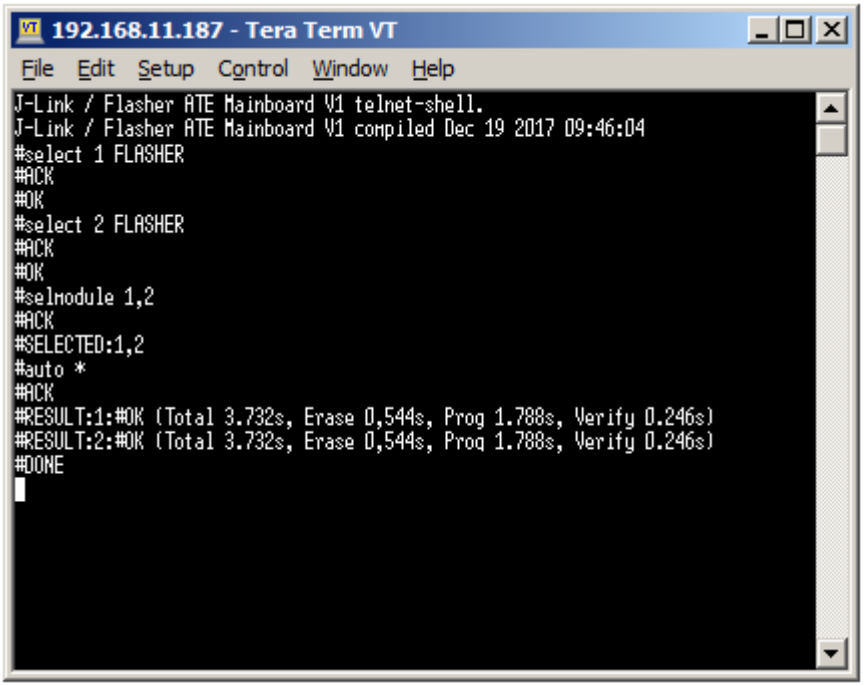
First, choose the project file that shall be flashed. Use the command: `#select [modulenumber] [project file name]`, e.g. `#select 1 FLASHER`. Repeat the step for every module.

> **Note**
>
> The file extensions are added by the firmware.

Second, select the module(s) which shall be used for programming, using the command: `#selmodule [1st modulenumber], [2nd modulenumber]…`, e.g. `#selmodule 1,2`.

Third, start the programming with the command `#auto *`.

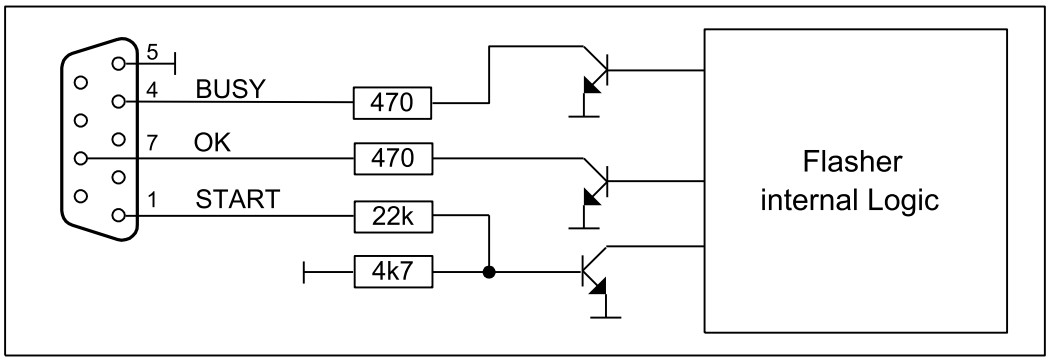
The picture shows the terminal after executing the commands.

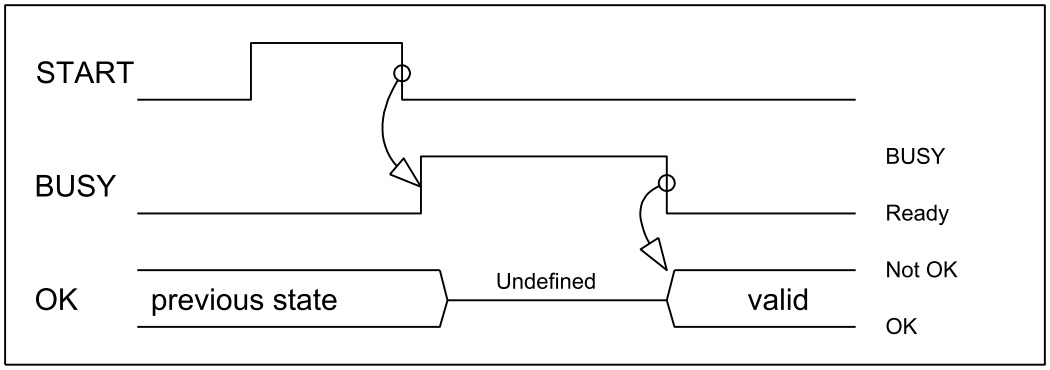


3.1.3 Starting the programming sequence via handshake protocol

The Flasher can be remote-controlled by automated testers without the need of a connection to a PC. Therefore the Flasher is equipped with additional hardware control functions, which are connected to the SUBD9 male connector, normally used as RS232 interface to the PC.

The following diagrams show the internal remote control circuitry of Flasher:





Pin No.	Function	Description
1	START	A positive pulse of any voltage between 5 and 30V with duration of min. 30 ms starts “Auto” function (Clear / Program / Verify) on falling edge of pulse. The behavior of the “Auto” function depends on the project settings, chosen in J-Flash at the **Production** tab.
4	BUSY	As soon as the “Auto” function is started, BUSY becomes active, which means that transistor is switched OFF.
5	GND	Common Signal ground.
7	OK	This output reflects result of last action. It is valid after BUSY turned back to passive state. The output transistor is switched ON to reflect OK state.

Note

As the Flasher ATE is a modular system, using the handshake remote control START always triggers the “Auto” function of every connected module. The BUSY line is signaled as long as any module is still busy and the OK line only reports “OK” in case

every module has successfully completed the operation. We recommend using the ASCII command interface, described in the next chapter, for the Flasher ATE as it provides better remote control capabilities.

Chapter 4

Additional Information

- Flasher ATE documentation: The full documentation of the Flasher ATE (including the ASCII command protocol) is available here: *Flasher ATE User Guide*
- Flasher ATE web page: Additional information can also be found on our web page *Flasher ATE Website*
- J-Flash-Tool: The J-Flash-Tool is a GUI based tool to create project files for the SEGGER Flashers.

Chapter 5

Glossary

Flasher ATE
Flasher ATE is the SEGGER production programmer for automated test equipment.

ASCII command protocol
ASCII command protocol is a ASCII character based protocol to control the Flasher ATE actions.

J-Flash
J-Flash is a tool to create project files for the SEGGER Flasher products.

J-Link-Config
J-Link-Config is a Configuration-Tool for SEGGER Flashers and J-Links.

Telnet
Telnet is a TCP-based network protocol to transfer characters between two end points.

RS232
RS232 is a hardware standard for asynchronous data transfer between two end points.

