
SPC5 Flash programmer

Introduction

SPC5 Flash programmer is a software tool developed to manage the Flash memory (program / verify / dump / erase) on different targets, via SCI, CAN and K-line interfaces.

The application supports a user-friendly GUI that allows setting all the connection parameters, connecting with the target using a USB dongle and performing all the required operations.

1 Overview

1.1 Scope

This document gives an overview of the SPC5 Flash programmer software and illustrates its functionalities and how to use them.

1.2 Requirements

SPC5 Flash programmer has been designed to work on any Windows 10 equipped computer, with no specific hardware requirements.

An available USB and/or RS232 port is required - depending on which interfaces will be used - in order to connect to the target board.

2 SPC5 Flash programmer overview

SPC5 Flash programmer is an application developed to manage the Flash memory by different interfaces. It is useful to easily perform the basic Flash management tasks, with results logging.

The supported interfaces are:

- SCI
- K-Line
- CAN

Interface availability depends on the selected target, as reported in the below table.

Table 1. Supported interfaces

Line	Target	CAN	SCI	K Line
L line	SPC56ELxx	✓	✓	
	SPC564Lxx	✓	✓	
P line	SPC560P34		✓	✓
	SPC560P40	✓	✓	✓
	SPC560P44	✓	✓	✓
	SPC560P50	✓	✓	✓
	SPC56AP60	✓	✓	✓
D line	SPC560D30		✓	✓
	SPC560D40	✓	✓	✓
B/C lines	SPC560B40	✓	✓	✓
	SPC560C40	✓	✓	✓
	SPC560B44	✓	✓	✓
	SPC560C44	✓	✓	✓
	SPC560B50	✓	✓	✓
	SPC560C50	✓	✓	✓
	SPC560B54	✓	✓	✓
	SPC560B60	✓	✓	✓
	SPC560B64	✓	✓	ü
	SPC564B64	✓	✓	
	SPC564B70	✓	✓	
	SPC564B74	✓	✓	
	SPC56EC64	✓	✓	
	SPC56EC70	✓	✓	
SPC56EC74	✓	✓		
M line	SPC563M64	✓	✓	✓
A line	SPC564A70	✓	✓	
	SPC564A80	✓	✓	
Velvety	SPC570S40	✓	✓ 20, 40 MHz	
	SPC570S50	✓	ü 20, 40 MHz	

Line	Target	CAN	SCI	K Line
K2	SPC574K72		✓ 20, 40 MHz (ASC@CAN)	
Lavaredo	SPC572L64	✓	ü 20, 40 MHz	
Eiger	SPC58NE84		✓ 20, 40 MHz (ASC@CAN)	
Bernina	SPC58NN84		✓ 20, 40 MHz	
Chorus 4M	SPC58xC80		✓ 20, 40 MHz	
	SPC58xC74		✓ 20, 40 MHz	
	SPC58xC70		✓ 20, 40 MHz	
Chorus 1M	SPC582B60		✓ 20, 40 MHz	
	SPC582B54		✓ 20, 40 MHz	
	SPC582B50		✓ 20, 40 MHz	

The tool's functionalities are:

- Program
 - Download and write an image file into the Flash memory.
- Erase
 - Erase data in the Flash memory's selected Flash blocks or address range.
- Dump
 - Upload the content of the Flash memory and save it to a file.
- Verify
 - Check the content of the Flash memory against an image file.
- Blank Check
 - Check if the selected Flash memory blocks or address range contains data.

Supported file formats:

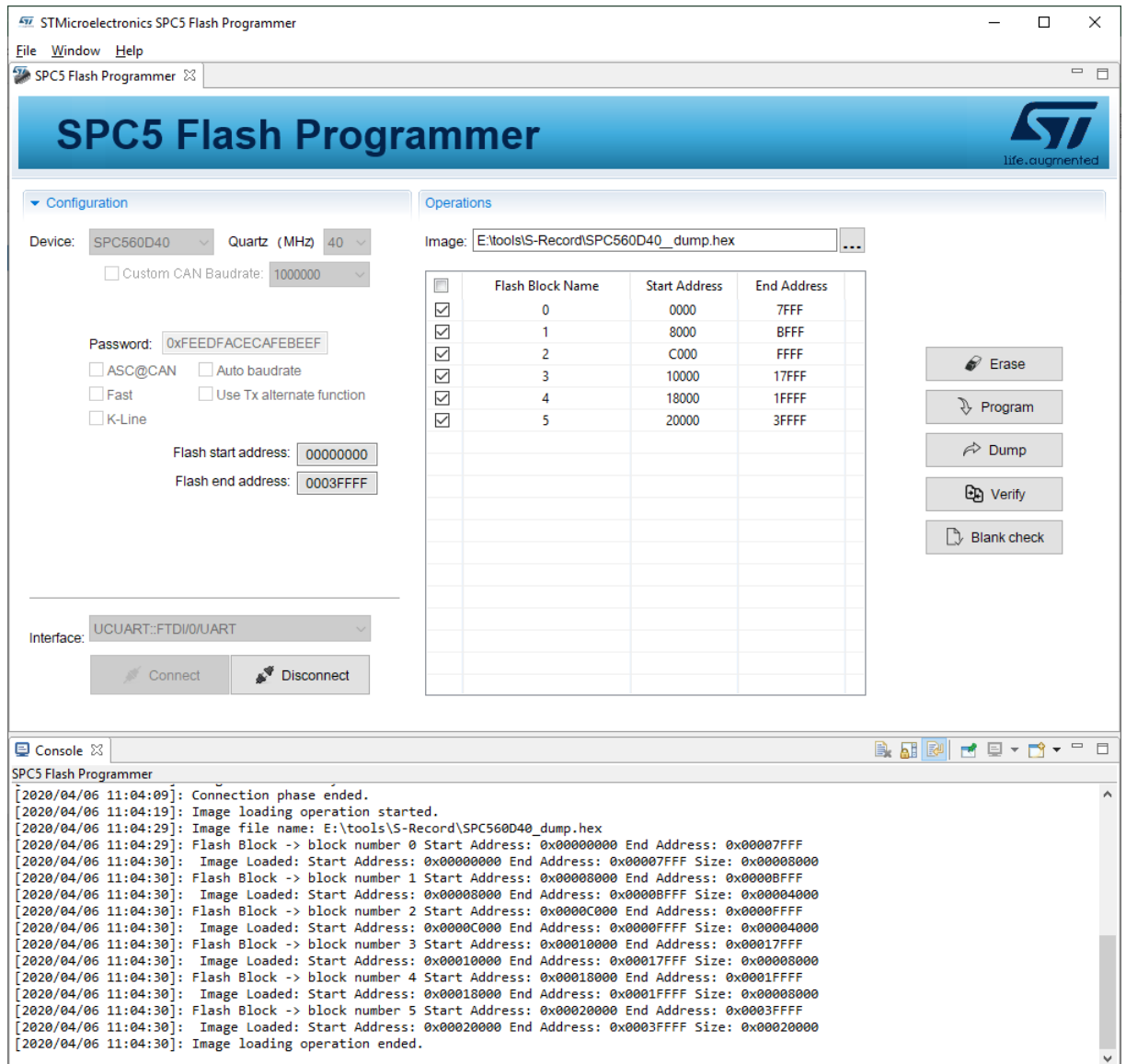
- S19, run or hex file formats are supported for *Program* and *Verify* actions.
- Binary format is supported for *Dump* action.

2.1 SPC5 Flash programmer GUI

SPC5 Flash programmer is composed of a menu (2.2) and a single window, containing two views, displaying three panes:

- Configuration (2.3)
- Operations (2.4)
- Console (2.5)

Figure 1. SPC5 Flash Programmer GUI overview



2.2 Menu

2.2.1 File

Contains the *Exit* item that allows closing the application.

2.2.2 Window

Contains two sub menus:

- Show view
 - SPC5 Flash programmer
 - Console
- Perspective
 - Reset perspective

The former allows displaying the listed views, if not already shown in the main window; the latter resets the main window items to their original size and position.

2.2.3 Help

The *Help* menu contains three items:

- Welcome
- Help contents
- About SPC5 Flash programmer

“*Welcome*” opens the “welcome pages”, that contain basic information on the application; “*Help Contents*” opens the help system and “*About SPC5 Flash Programmer*” displays version number and installation details.

2.3 Configuration pane

This pane allows the user to select the target device and configure the connection options to be used by SPC5 Flash Programmer to connect to the selected target.

The contents of the pane may vary, depending on the target that has been selected from the “*Device*” drop-down menu.

Figure 2. Device configuration – no overlay

The screenshot shows a configuration pane with the following elements:

- Configuration** (dropdown menu)
- Device:** SPC563M64 (dropdown)
- Quartz (MHz):** 8 (dropdown)
- Custom CAN Baudrate:** 200000 (dropdown)
- Password:** 0xFEEDFACECAFEBEEF (text input)
- ASC@CAN**
- Auto baudrate**
- Fast**
- Use Tx alternate function**
- K-Line**
- Flash start address:** 00000000 (text input)
- Flash end address:** 0017FFFF (text input)
- Interface:** Select the interface (dropdown)
- Connect** (button with plug icon)
- Disconnect** (button with unplug icon)

2.3.1 Device List

Allows the user to select the target device. The “*Device*” drop box lists all the supported targets. When a target from the list is selected, the content of the pane changes accordingly.

2.3.2 External quartz

The “*Quartz (MHz)*” drop box allows the user to indicate the external crystal used by the target device. When the user sets this value, the speed of the interface used is updated accordingly, unless the “*Custom CAN Baudrate*” checkbox is checked (see below).

2.3.3 Custom CAN Baudrate

This option allows to specify a custom baudrate for CAN data transfer, instead of letting the application calculate the CAN bus speed depending on the selected quartz clock.

This option is enabled only if a supported device is selected and is actually used only when a CAN interface is selected from the “*Interface*” drop box.

2.3.4 Connection settings

The lower part of the device configuration section allows the user to set the interface properties. this is the list of the available controls and their function:

- “*Password*” allows to set the password used by BAM/BAF.
- “*ASC@CAN*” must be checked when an ASC@CAN interface is connected.
- “*Fast*” check allows increasing the baud rate in the UART interface after the target connection phase, from the standard BAM/BAF baud rate to 115200bps, in order to speed up the application operations.
- “*K-Line*” check-box allows selecting K-Line interface. This is effective only if an UART interface is chosen from interface list; it has no effect on CAN connections.
- “*Auto baud rate*” check allows using the BAM with auto baud setting enabled.
- “*Use Tx alternate function*” check-box allows selecting the alternate tx as tx pin.

Figure 3. Device configuration – overlay allowed

Configuration

Device: Quartz (MHz)

Custom CAN Baudrate:

Password:

ASC@CAN Auto baudrate

Fast Use Tx alternate function

K-Line

Flash start address (No overlay allowed) :

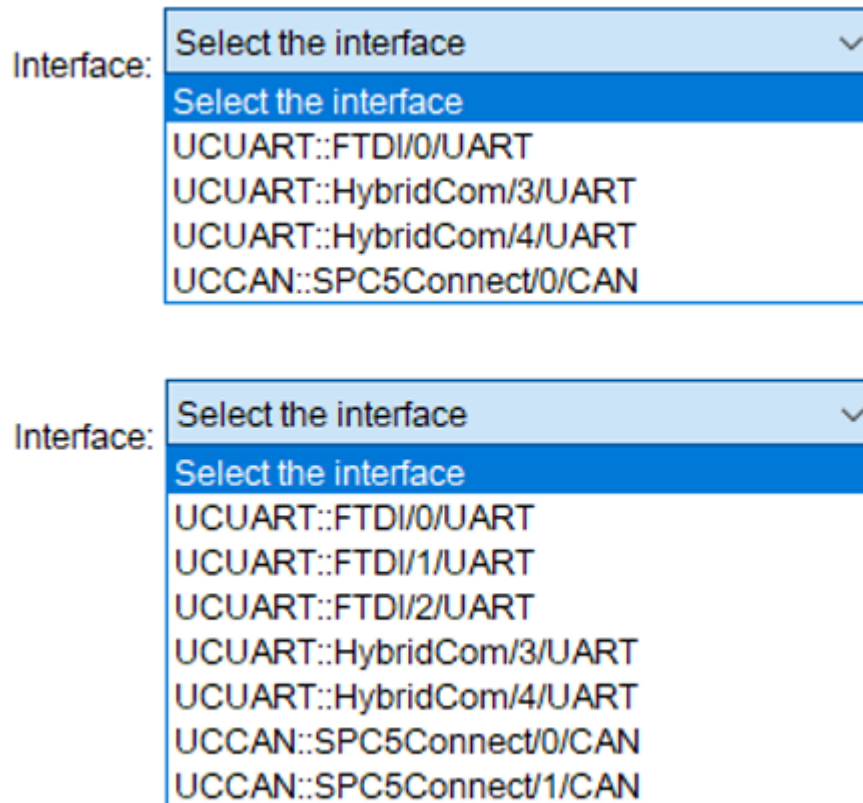
Flash end address (No overlay allowed) :

Flash start address (overlay allowed) :

Flash end address (overlay allowed) :

Interface:

Figure 4. Interface list contents examples



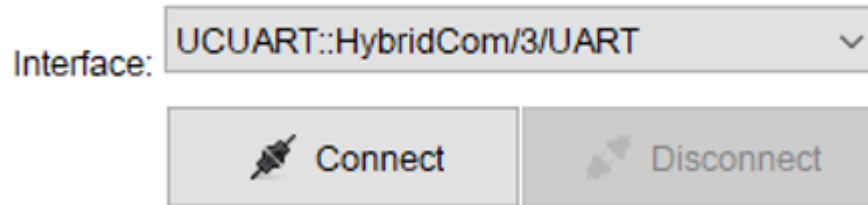
2.3.5 Interface list

The *Interface* dropdown lists the currently available interfaces; its content is updated when clicked. SPC5 Flash Programmer supports two kinds of interface:

- SCI interface:
 - UCUART::HybridCom/<COM port number>/UART
 - Virtual com ports installed in the PC. Select one of these interfaces when the target is connected to one of your host's COM ports.
 - UCUART::FTDI/<instance number>/UART
 - FTDI-based serial port USB dongles(or *USB-to-serial* converters). Select one of these interfaces when the target is connected to such a peripheral.
- CAN interface:
 - UCCAN::SPC5Connect/<SPC5Connect number>/CAN
 - CAN ports available from one or more *SPC5Connect* connected to the PC's USB ports.

SPC5 Flash Programmer is not able to determine to which port a target is connected; please check your PC's configuration to select the correct port number.

Figure 5. Connect and disconnect buttons



2.3.6 Connect button

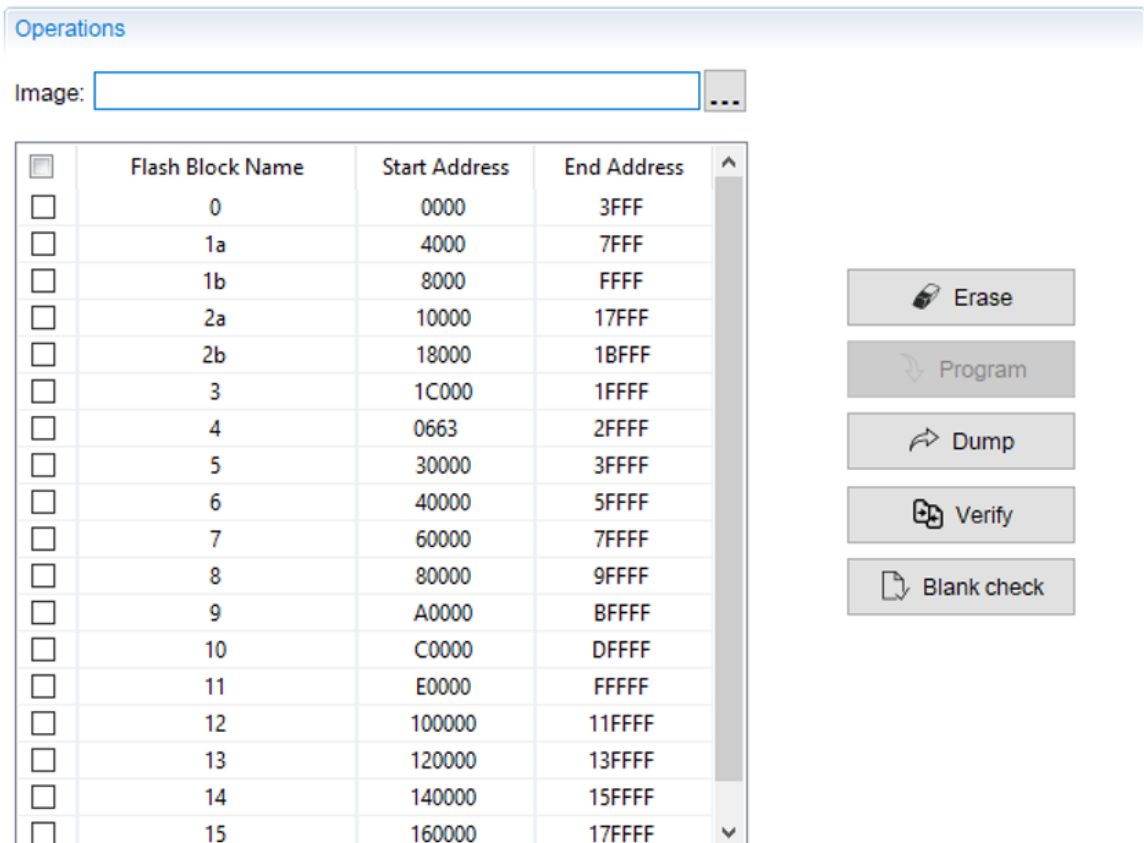
After an interface has been selected and all the connection parameters configured, push “Connect” button to let the BAM/BAF download the loader plugin into the target’s RAM and run it. The loader plugin is a target-specific software that configures and enables the flash memory for R/W operations and then executes the commands received from the GUI. As soon as it is ready, the Flash management commands will be enabled.

2.3.7 Disconnect button

This button closes the connection with the target; some targets will be automatically reset before disconnecting.

2.4 Operations

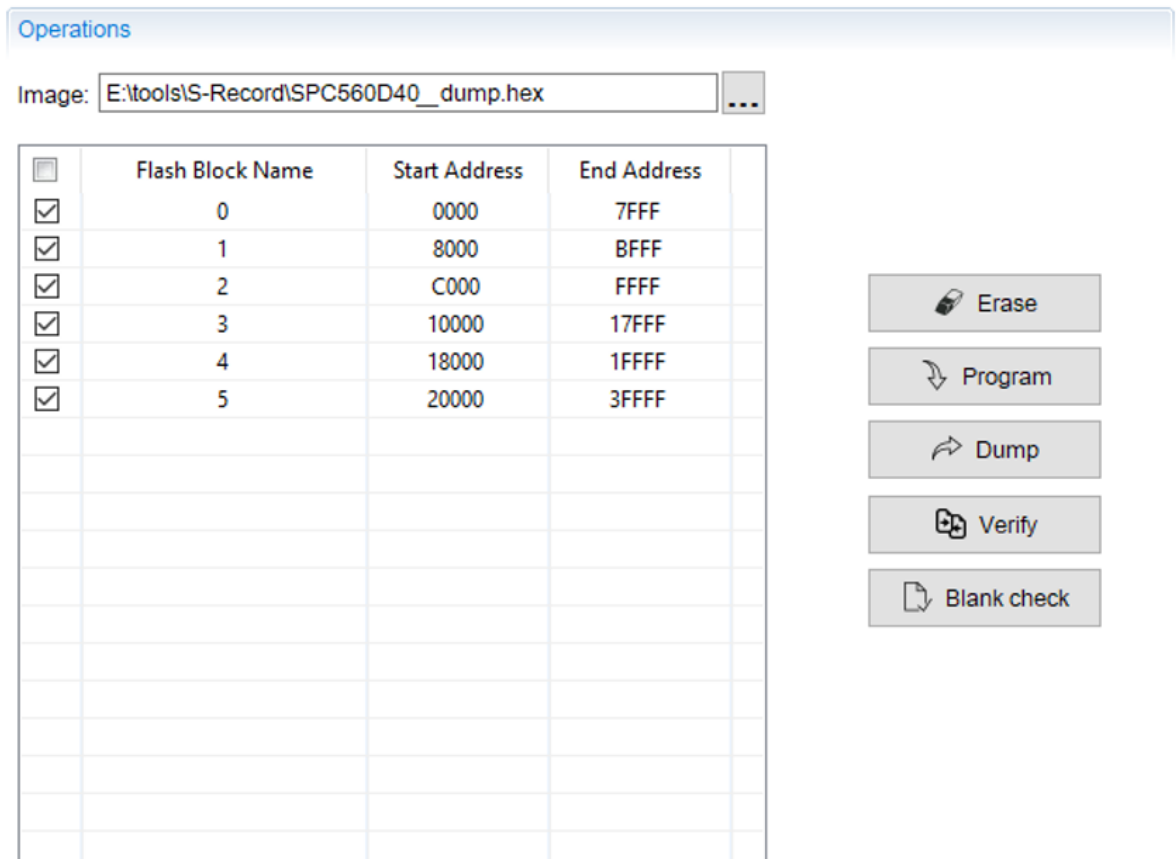
Figure 6. Operations



2.4.1 Load image button

Clicking on the ellipsis button a *file requester* window pops up, allowing the selection of an image file to be used as an input for *program* or *verify* operations. The selected image file is displayed in the *image:* edit field. The selected file is parsed, and the corresponding Flash blocks are checked in the *flash blocks map*.

Figure 7. Image file loading



2.4.2 Flash blocks map

A table containing the description of all Flash blocks. The checkbox in the first column of each row allows to select/deselect the corresponding Flash block for the subsequent operations.

2.4.3 Erase command

Erases the selected Flash blocks. The result of the operation is displayed in the console.

2.4.4 Program command

Programs the Flash memory of the connected device with the contents of the selected binary file. When an image file is loaded it is parsed and all the blocks corresponding to the file contents are automatically checked: the user can deselect the blocks he/she doesn't want to program.

A dialog is shown upon pushing the "*program*" button, allowing further options: "*Erase before programming*" and "*Verify after programming*".

The results of the operation are displayed in the console.

2.4.5 Verify command

Performs the verify operation on the Flash memory by comparing the actual memory contents with an image file. A dialog window lets the user choose how to proceed:

- “Verify marked blocks with the selected image file” immediately starts the operation, using the image file already loaded through the “Image” field, and checking only the content of the selected blocks;
- “Verify by address specifying an image file” opens a *file requester dialog* to let the user choose an image file: all the address range(s) contained in the image file are checked.

The result of the operation is displayed in the console.

2.4.6

Blank check command

Performs a blank check of the Flash blocks selected in the *Flash blocks map*, also reporting the first address containing data of any non-empty block.

The result of the operation is displayed in the console, as usual.

3 Console


It shows the output of the tool operations, including connection details and connection status changes (see the following figure).


Figure 8. Console


Operations


Image: ...


<input type="checkbox"/>	Flash Block Name	Start Address	End Address
<input checked="" type="checkbox"/>	0	0000	7FFF
<input checked="" type="checkbox"/>	1	8000	BFFF
<input checked="" type="checkbox"/>	2	C000	FFFF
<input checked="" type="checkbox"/>	3	10000	17FFF
<input checked="" type="checkbox"/>	4	18000	1FFFF
<input checked="" type="checkbox"/>	5	20000	3FFFF

 Erase

 Program

 Dump

 Verify

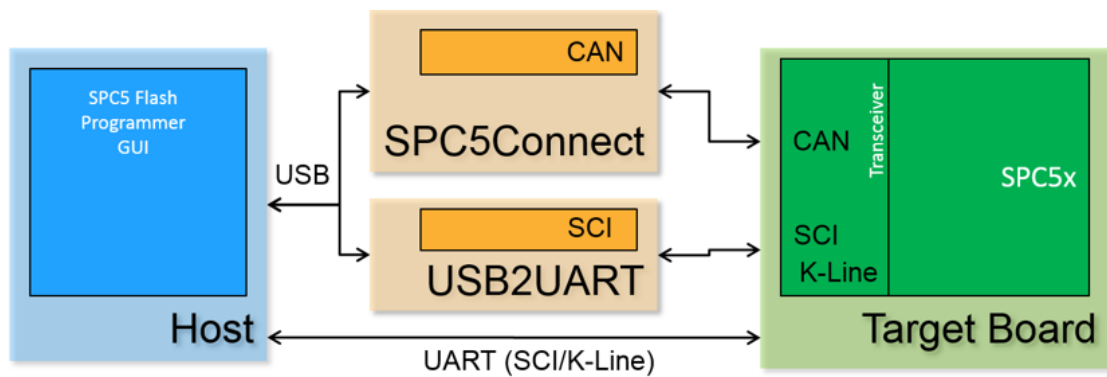
 Blank check

4 Diagram overview

This chapter shows a pictorial representation of the connection schema needed to work with the *SPC5 Flash Programmer* tool.

The management of the Flash memory of a target system (in green) is operated by a software running on the MS Windows host machine (in blue), connected to the target using one of the supported hardware interfaces (in orange), as shown in the following figure.

Figure 9. Top level diagram



Revision history

Table 2. Document revision history

Date	Version	Changes
16-Nov-2015	1	Initial release.
01-Jun-2021	2	Updated Section 2 SPC5 Flash Programmer overview; Section 2.3 Configuration pane;Section 3 Console.

Contents

1	Overview	2
1.1	Scope	2
1.2	Requirements	2
2	SPC5 Flash programmer overview	3
2.1	SPC5 Flash programmer GUI	4
2.2	Menu	5
2.2.1	File	5
2.2.2	Window	5
2.2.3	Help	6
2.3	Configuration pane	6
2.3.1	Device List	6
2.3.2	External quartz	7
2.3.3	Custom CAN Baudrate	7
2.3.4	Connection settings	7
2.3.5	Interface list	9
2.3.6	Connect button	10
2.3.7	Disconnect button	10
2.4	Operations	10
2.4.1	Load image button	11
2.4.2	Flash blocks map	11
2.4.3	Erase command	11
2.4.4	Program command	11
2.4.5	Verify command	11
2.4.6	Blank check command	12
3	Console	13
4	Diagram overview	14
	Revision history	15

IMPORTANT NOTICE – PLEASE READ CAREFULLY

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. For additional information about ST trademarks, please refer to www.st.com/trademarks. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2021 STMicroelectronics – All rights reserved