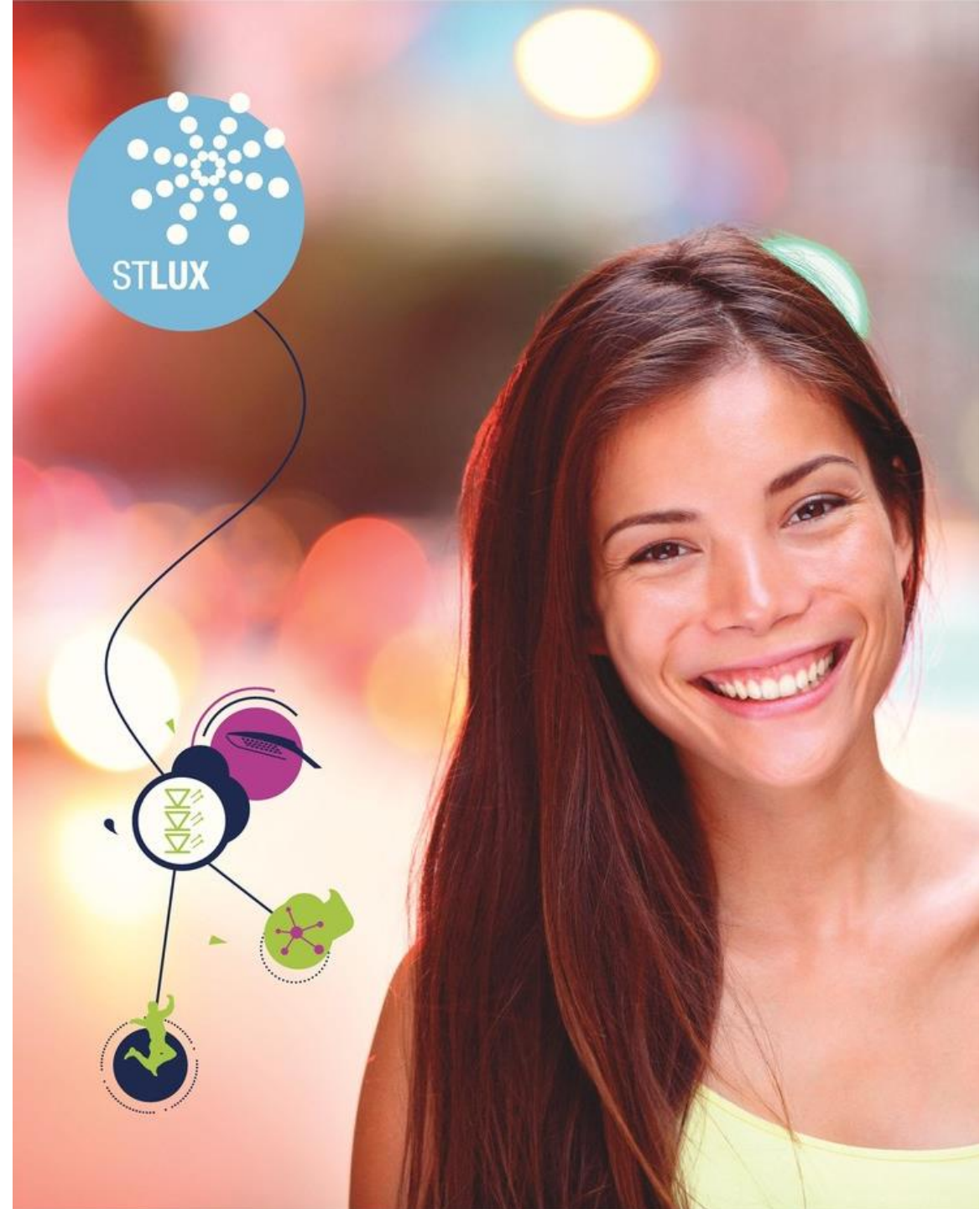


STLUX™

Development Environment





STLUX™ Lighting Control Made Easy

Breakthrough digital control technique with a full set of specific features and peripherals for lighting

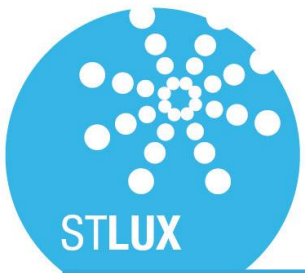


Smart Lighting: Suitable for high power LED, high end lighting, HID applications



- Energy saving with high efficiency in any load conditions
- Dimming capability with maximum precision
- Wired or wireless communication, integrated DALI, remote control and interoperability with sensors

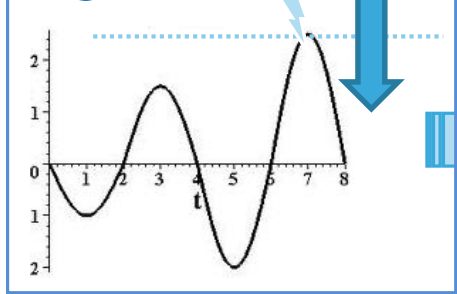




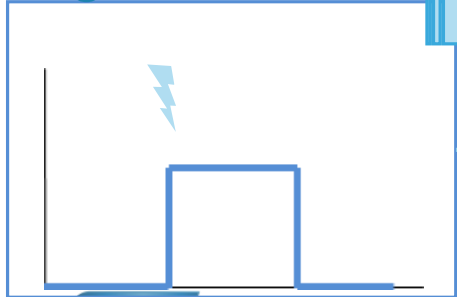
Configurable PWM generators

1. Parameters detection/measure in the application

Signal 1



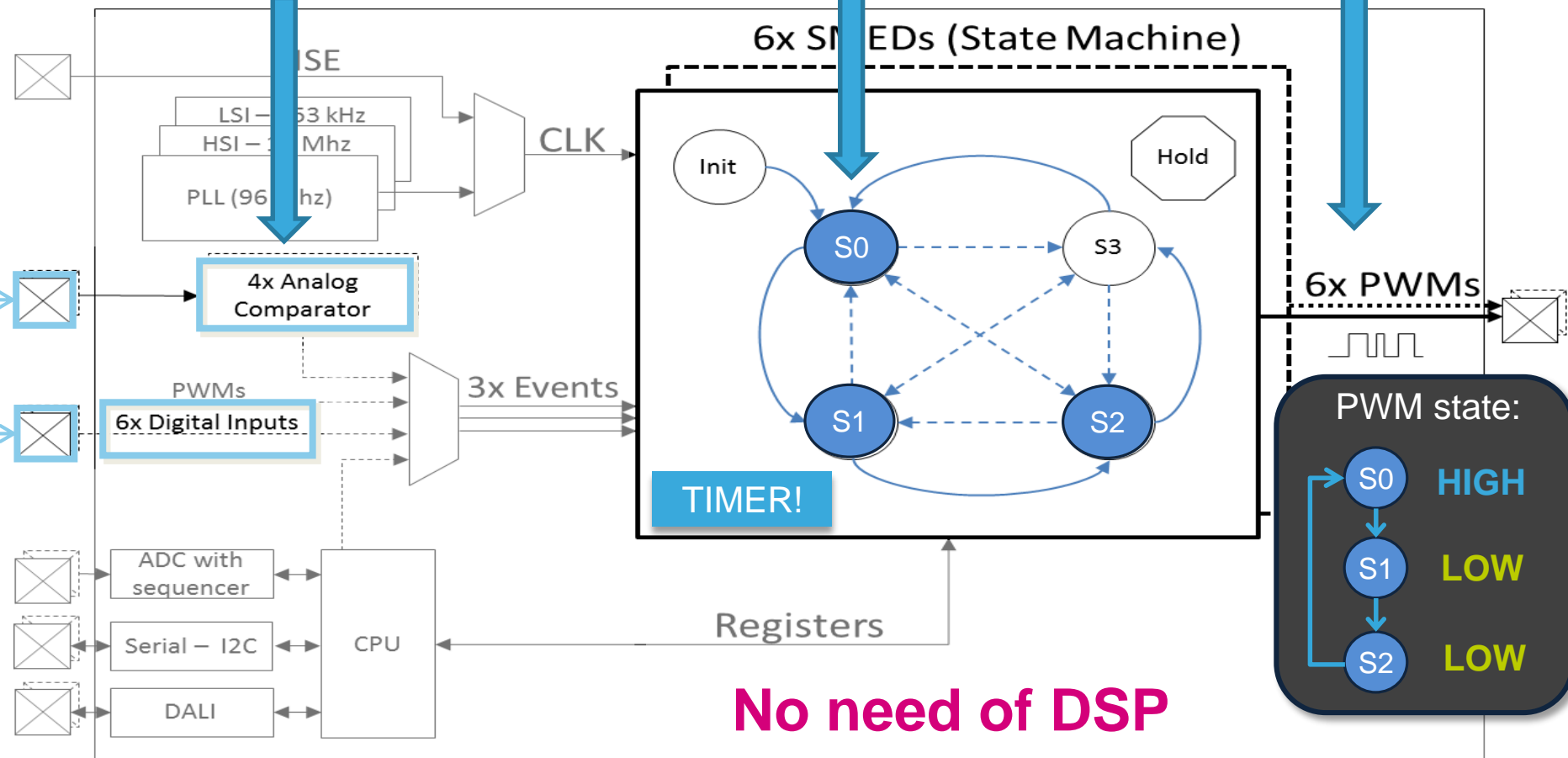
Signal 2



2. Signal elaboration and event handling

3. State machine evolution

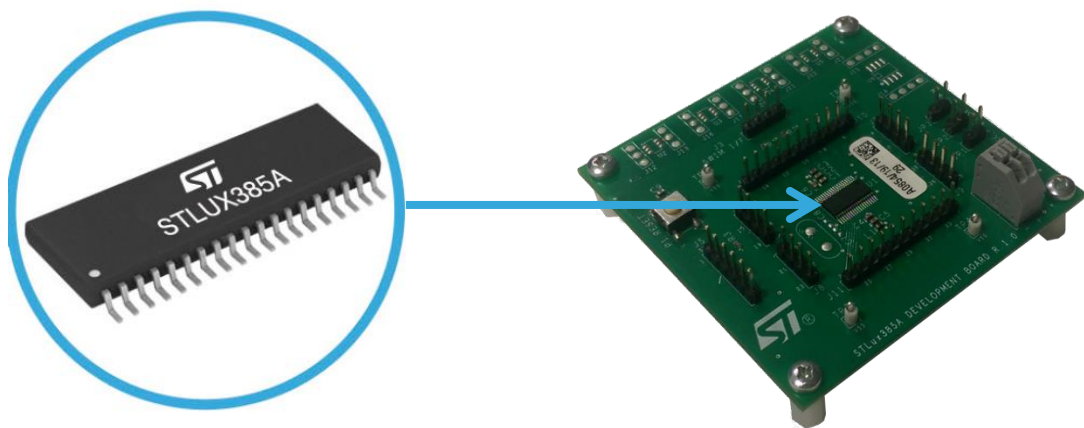
4. PWM output generation



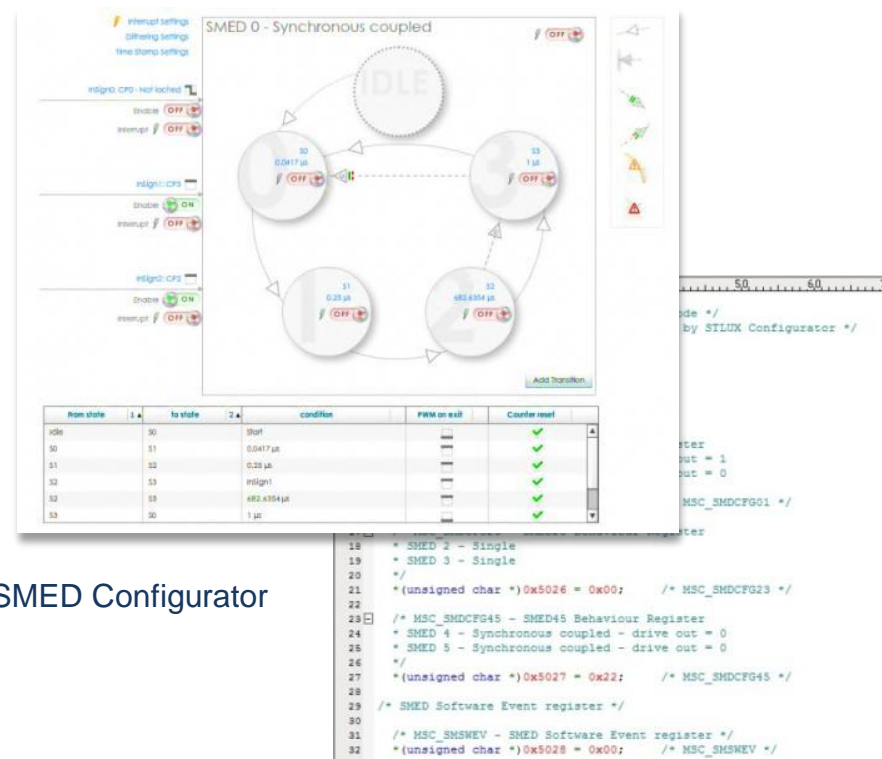


Development Environment

Easy-access platform to explore the variety of topologies and control schemes offered by STLUX385A

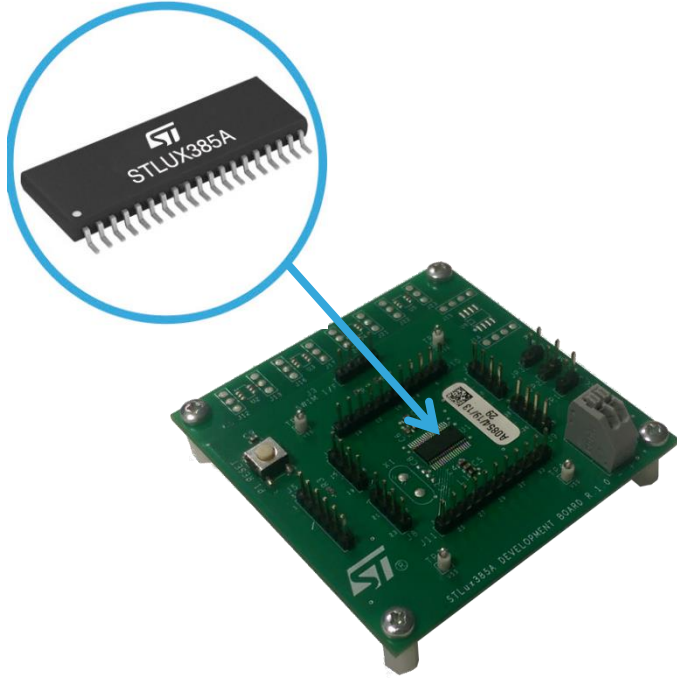


STEVAL-ILL068V1
or
STEVAL-ILL075V1



SMED Configurator

Libraries



STEVAL-ILL068V1
or
STEVAL-ILL075V1

Development Environment

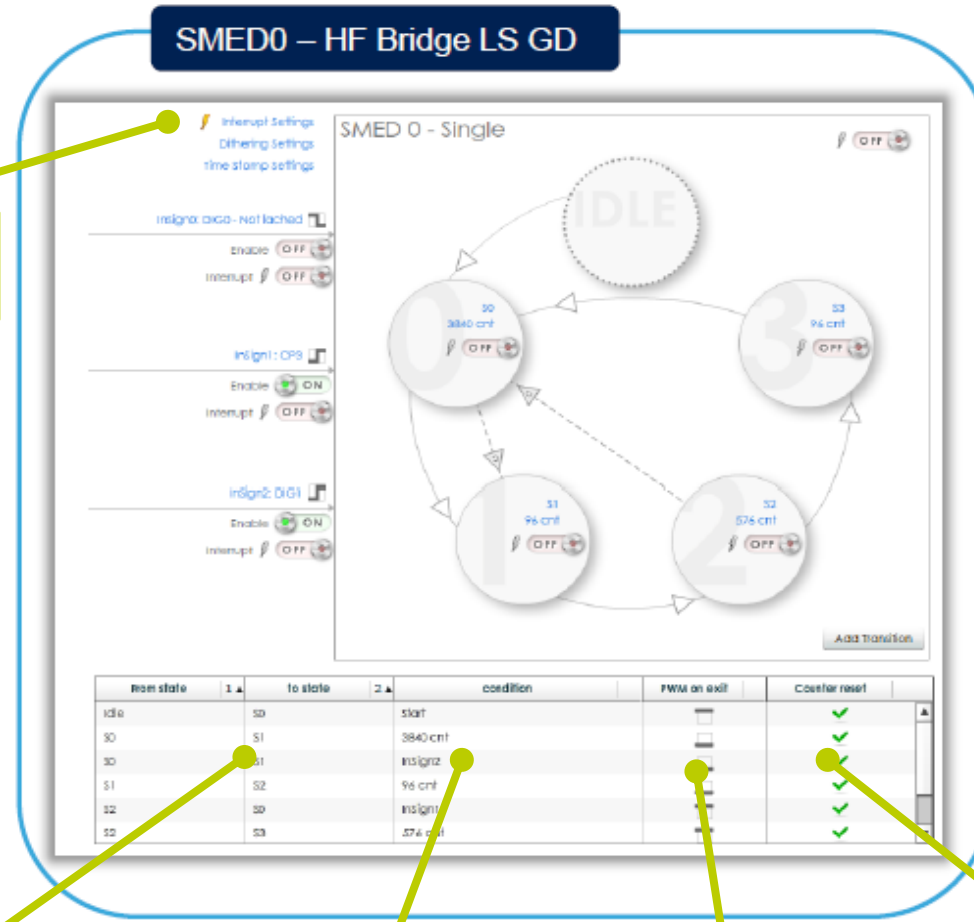
STLUX385A demonstration board

- All STLUX385A pins available on connectors
- Can be powered up in three different ways: using an external power supply, via the USB TTL cable or via the SWIM (Single Wire Interface Module) interface
- Features several free footprint areas, ideal for connecting power drivers or amplifiers
- Can be connected to a quartz clock, which can be used in place of the STLUX385A internal clocks



Development Environment

Graphical SMED configurator



- SMED configuration schemes
- Input configuration
- Clock settings
- FSM (“Finite State Machine”) configuration
- C code generation



STLUX385A

- 6 SMED controlled PWMs to achieve up to 1.3 ns PWM resolution
- Customizable algorithms enable higher conversion efficiency
- Internal 96 MHz PLL
- 4 analog comparators & analog-to-digital converters (ADCs)
- Suitable for primary side regulation applications and multi-strings





Design Flow: From the idea to prototype



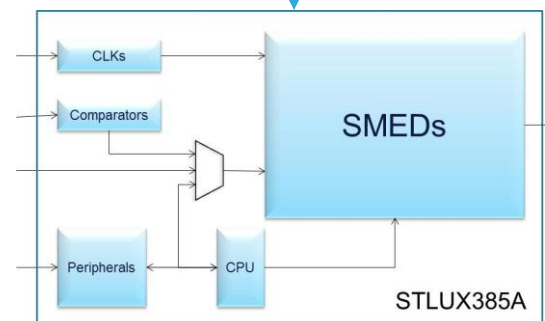
Study your idea

- STLUX Graphic Environment
- SMED Logic validation
- Project Generation
- Automatic C Code Generation
- SMEDs Examples



Quick prototype

- STLUX Development board
- Peripherals and SMEDs APIs
- Tutorial
- IAR/Raisonance compatible





Design Flow step 1: the SMED Configurator

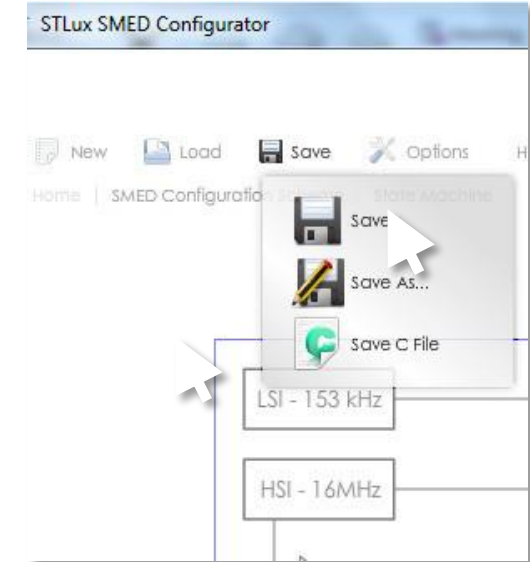
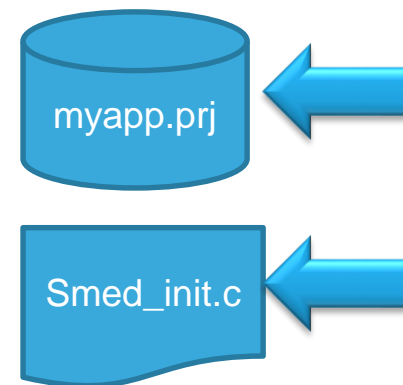
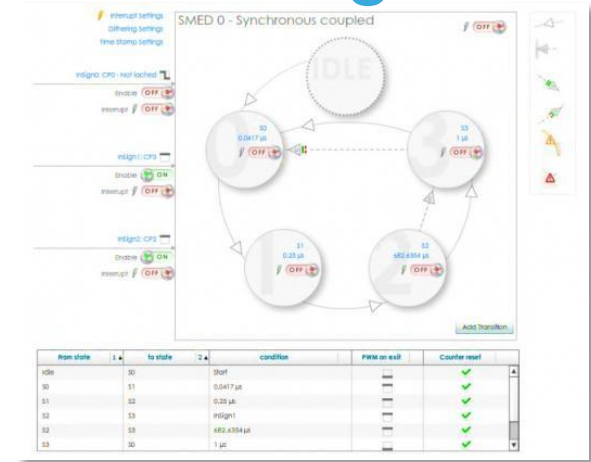
- Download the SMED Configurator
- Save the SMED Configurator install file and open it
- Follow the step-by-step guided installation procedure





Design Flow step 1: the SMED Configurator

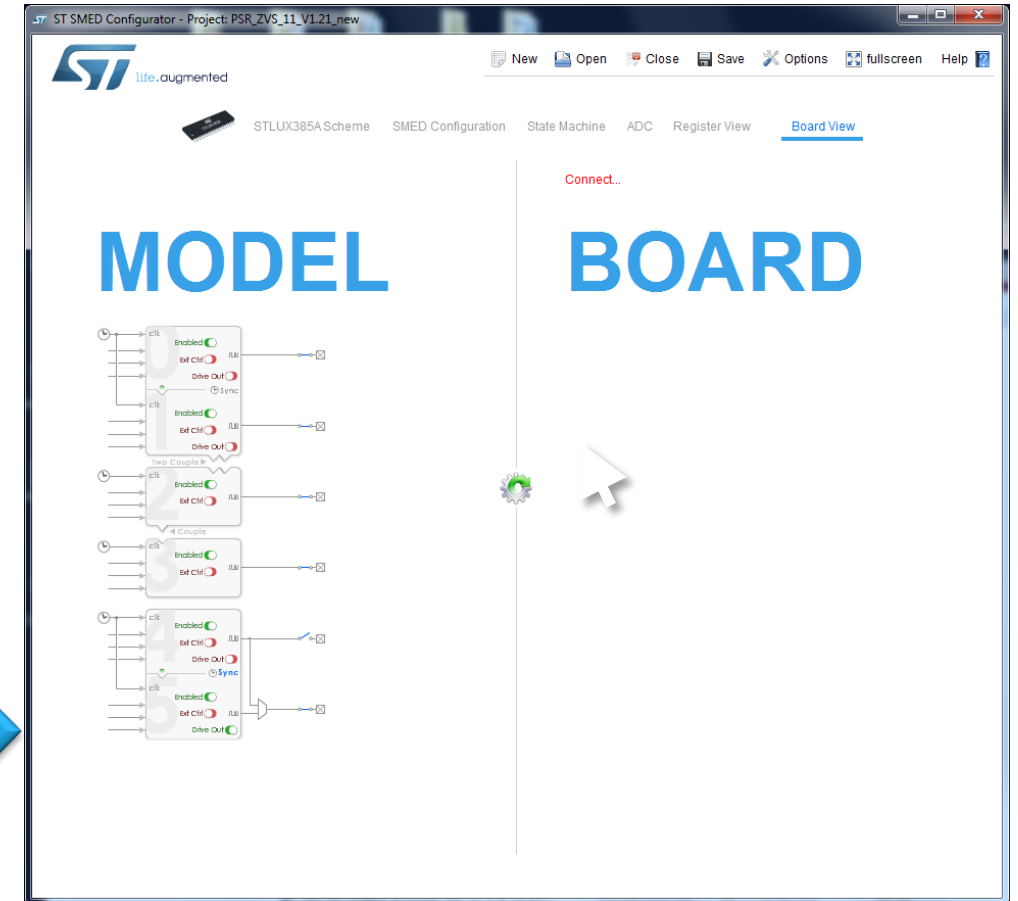
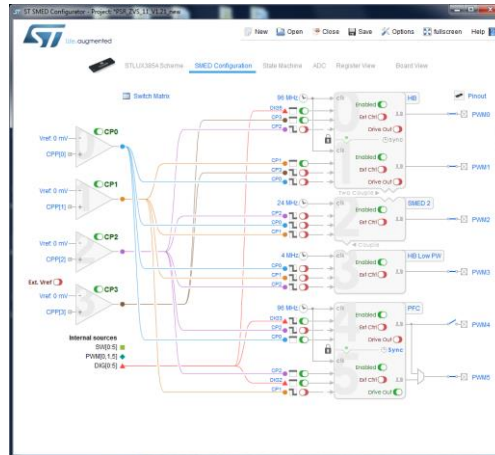
- Run the SMED Configuration GUI directly from your desktop and start designing your application
- Save your SMED Configurator project to easily retrieve and modify your SMED configuration
- Automatically translate the SMED configuration into C code
- Embed the C code into the STLUX application firmware.





Design Flow step 1: with ILL075V1

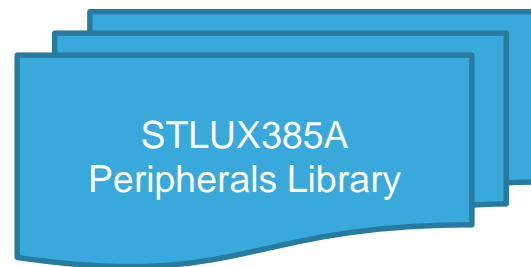
- Connect the PC USB line to STEVAL-ILL075V1 demo board
- Simple download your SMED configuration to the real SMED on the ILL075V1 demo board
- Check and verify your configuration using the input signal, the 96MHz real time clock and SMED behavior provided by the STLUX





Design Flow step 2: Getting ready with tools

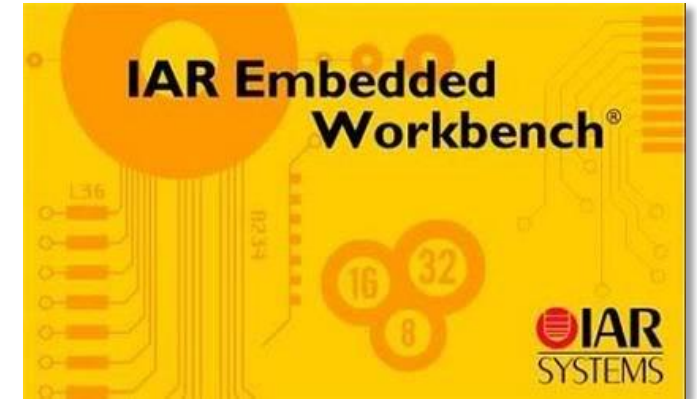
- Download the STLUX385A Peripherals Library User Manual
- Download the STLUX385A Peripherals Library
- Download the STLUX385A Examples Kit User Manual
- Download the STLUX385A Examples Kit





- STLUX includes an internal STM8 which can be programmed using the following STM8 tools:
 - IAR
 - Raisonance
- Choose your preferred STLUX compiler. Only one compiler shall be installed on your desktop.
- The compilers offer both free limited evaluations or full licenced releases.
- The following pages illustrate the install procedures for both IAR or Raisonance.

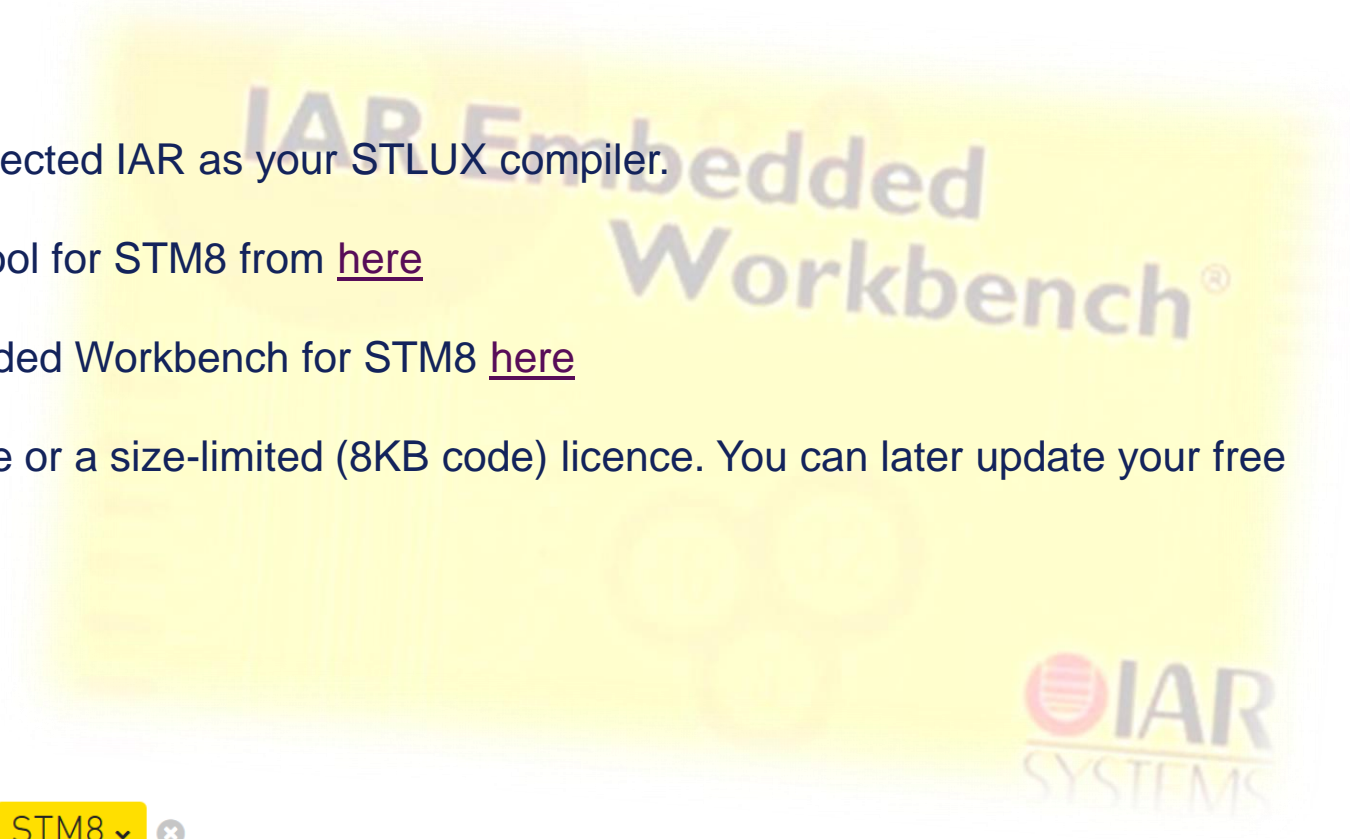
Design Flow step 2: IAR





Design Flow step 2: IAR

- The following instructions apply only if you selected IAR as your STLUX compiler.
- Download IAR embedded SW development tool for STM8 from [here](#)
- You can find more information on IAR Embedded Workbench for STM8 [here](#)
- IAR offers a free time-limited (30 days) licence or a size-limited (8KB code) licence. You can later update your free licence to get an full licence
- Follow the step-by-step guided installation



Showing STLUX385A ▾ for STM8 ▾ ✕

Overview

Features

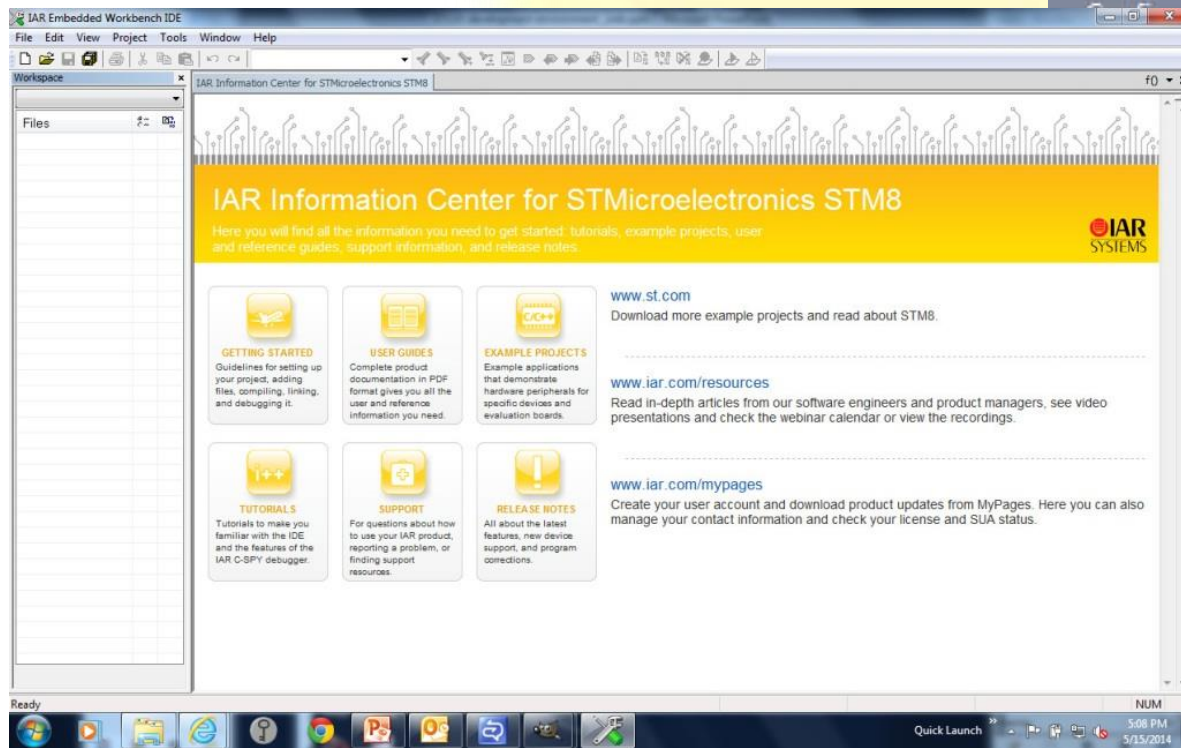
Editions and licensing

Full ST support



Design Flow step 2: IAR

- Run your IAR Embedded Workbench





Design Flow step 2: IAR



STEVAL-ILL068V1

Flat cable



ST-LINK/V2

To PC USB port



STSW-STLUXSMED01

STSW-STLUXLIB01

- IAR SW Development Workbench supports ST_LINK/V2 in-circuit debugger/programmer interface.
- ST_LINK/V2 supports the SWIM (Single Wire Interface Module) interface and is required to connect your PC to any STLUX Evaluation Board.
- ST_LINK/V2 is available online from most of the suppliers. You can find more information on ST_LINK/V2 [here](#)



Design Flow step 2: Raisonance

- The following instructions apply only if you selected Raisonance as your STLUX compiler.
- Download Raisonance embedded development tools for STM8 starting from [here](#)

RKit-STM8

Buy Support Contract Extension

<http://www.raisonance.com/support-extensions-for-STM8-tools.html>

Product Catalog

<http://www.raisonance.com/stm8-st7.html>

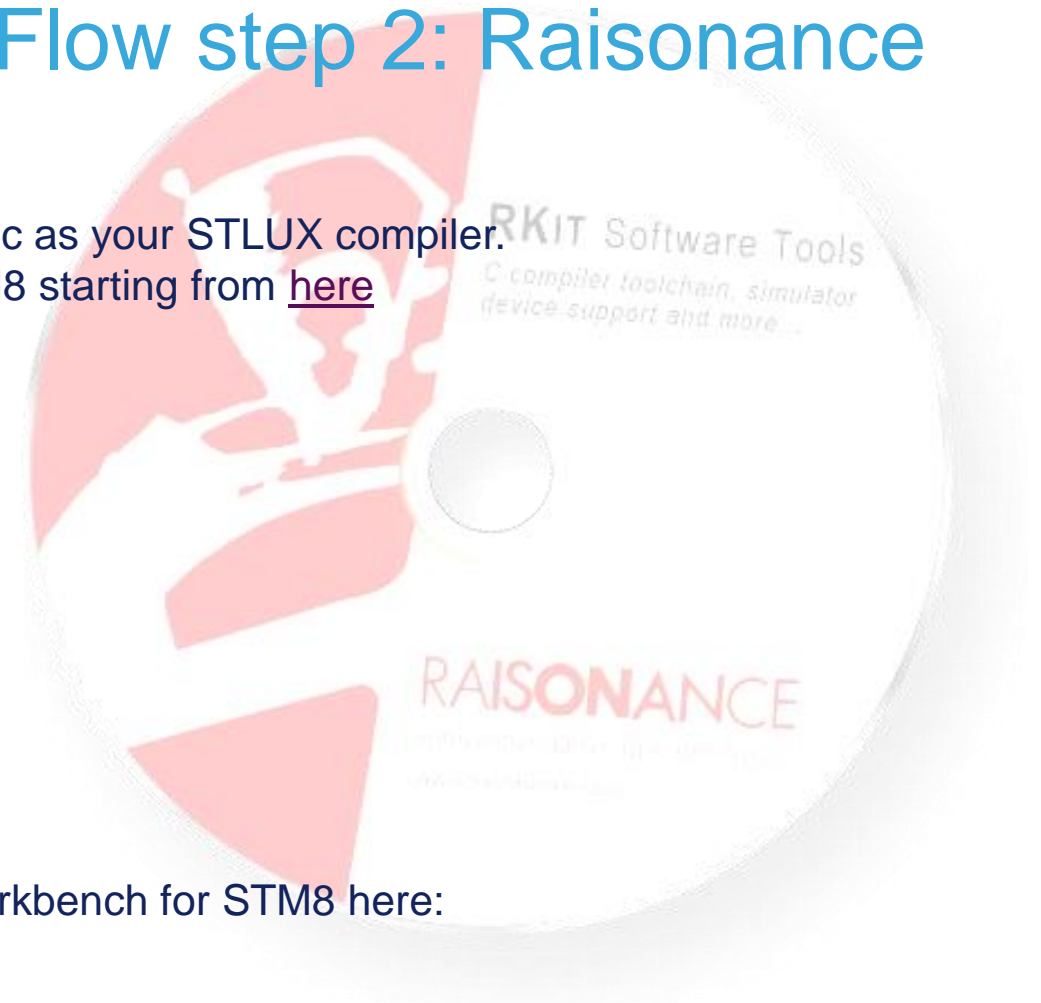
Tools for compiling, programming and debugging application code with Ride7 for STM8, ST7 and STLUX. You can download and try the RKit-STM8 toolchain from the table of version at the bottom of this page.

Available Licenses

- **Basic:** Compiles object up to 2 Kbytes. Included with RLink.
- **Lite:** Compiles object up to 32 Kbytes. [Purchase here](#) or [Request a STLUX license](#).
- **Enterprise:** Unlimited compiling. [Purchase here](#).

Note: for STLUX, you may request a free license to output 32 Kbytes of code, please [email the requested information to contact@raisonance.com](#).

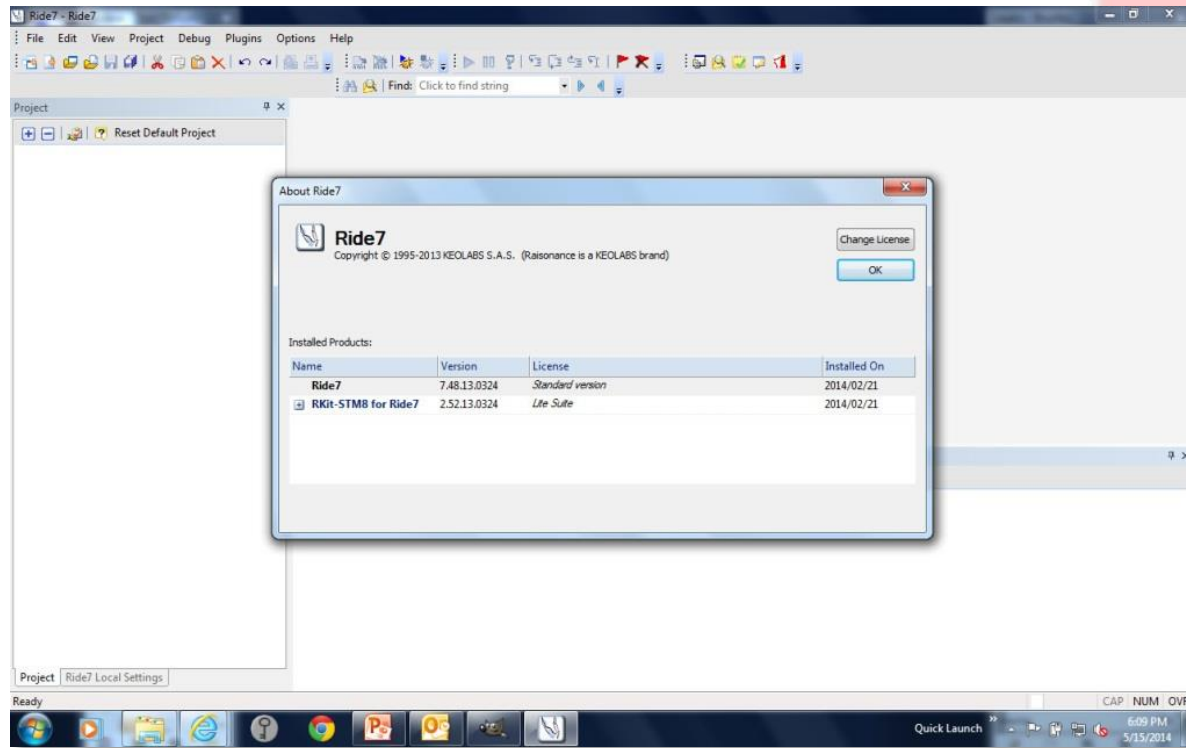
- You can find more information on Raisonance Embedded Workbench for STM8 here:
<http://www.raisonance.com/stlux.html>
- Raisonance offers a size-limited (32KB code) STLUX licence.
- Follow the step-by-step guided installation





Design Flow step 2: Raisonance

- Run your Raisonance Embedded Workbench from your desktop





Design Flow step 2: Raisonance



STEVAL-ILL068V1

Flat cable



To PC USB port



STSW-STLUXSMED01

STSW-STLUXLIB01

- Raisonance Development Workbench supports Raisonance RLINK microcontroller interface.
- RLINK supports the SWIM (Single Wire Interface Module) interface and is required to connect your PC to any STLUX Evaluation Board.
- RLINK is available online from the Raisonance website <http://www.raisonance.com/rlink.html>



- To connect your Rlink to STEVAL-ILL068V1 you need to implement a non-standard connector according to the following scheme.



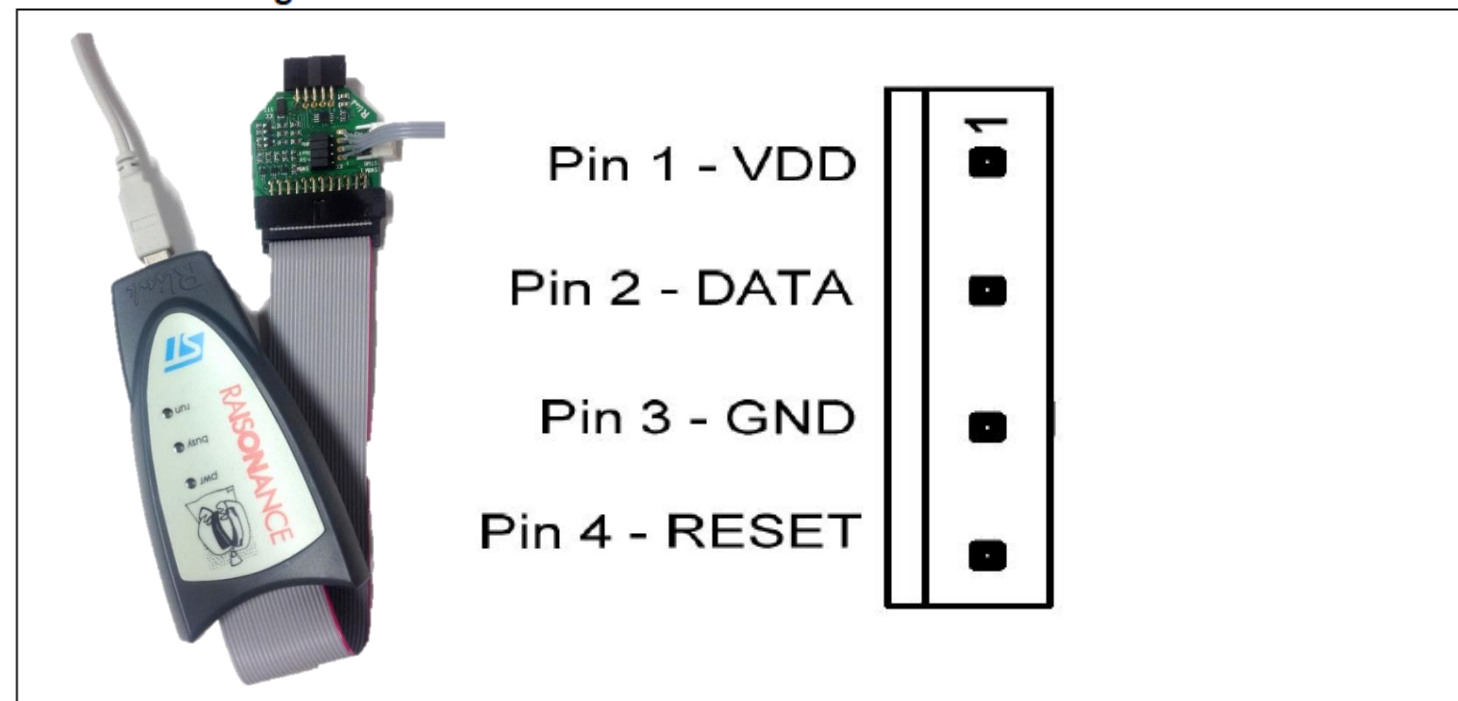
Design Flow step 2: Raisonance

SWIM flat ribbon connections

Pin no.	Name	Function	Target connection
1	VDD	Target VCC	MCU VCC
2	DATA	SWIM	MCU SWIM pin
3	GND	GROUND	GND
4	RESET	RESET	MCU RESET pin

- Use 10 mm length (max) four wire cable soldered on Rlink side and connected to four pin connector (female 2.54 mm) or refer to R-link documentation

Target SWIM connector

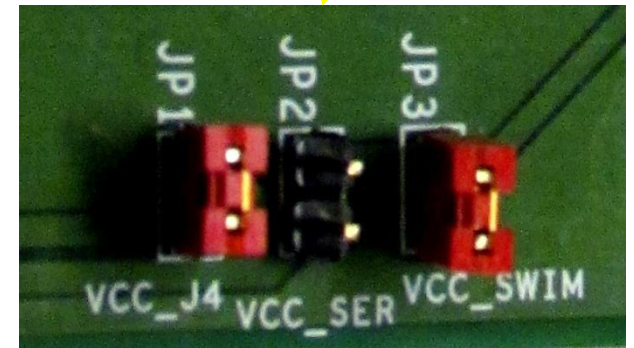
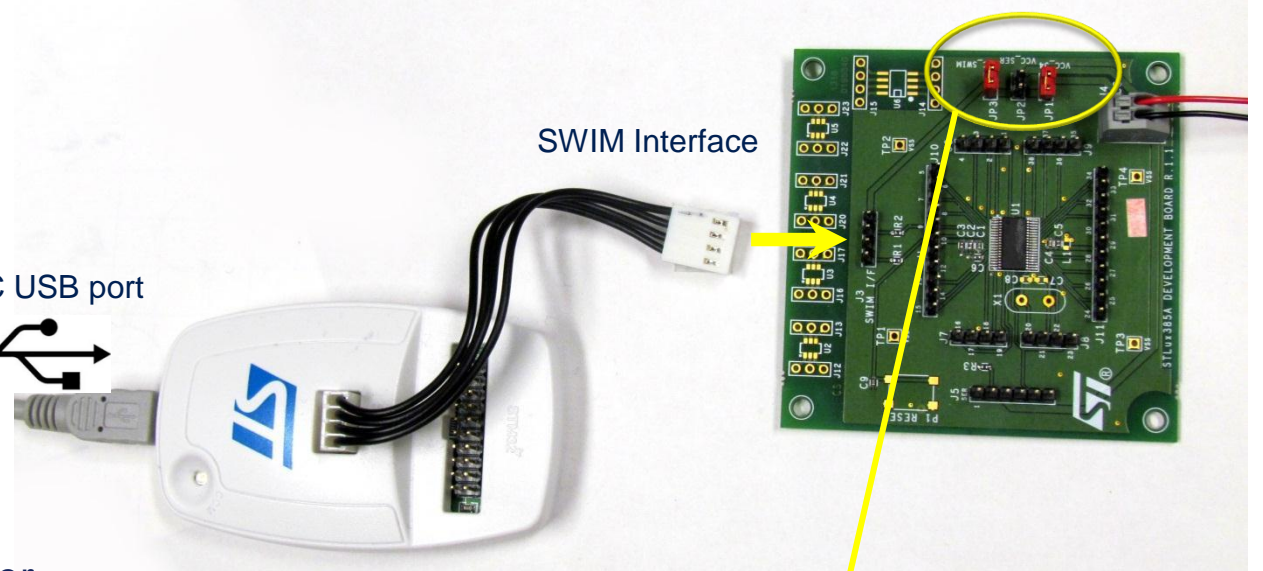




Design Flow step 3: STEVAL-ILL068V1

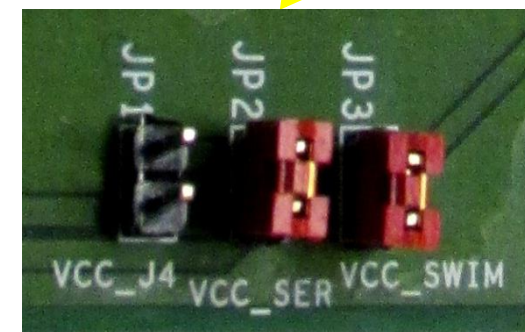
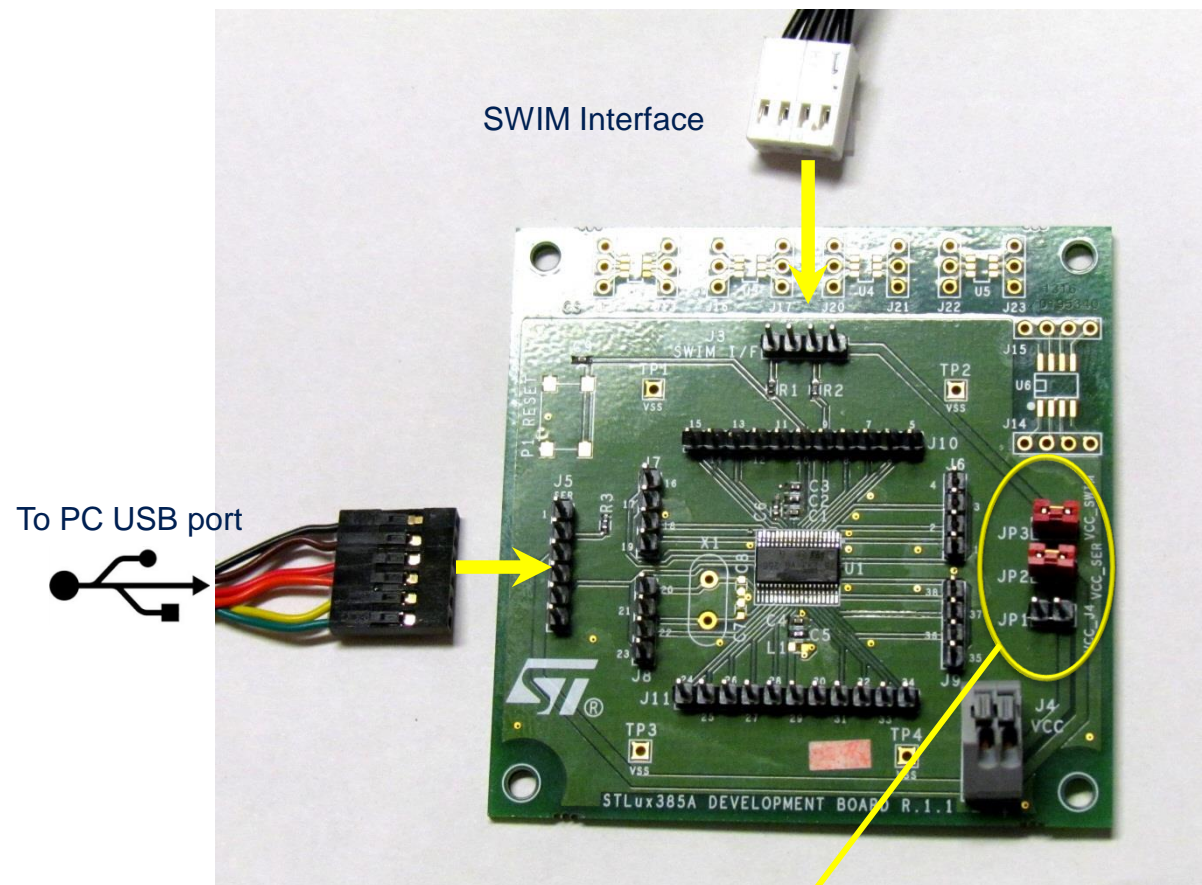
- Download the STEVAL-ILL068V1 Databrief from [here](#).
- Get a STEVAL-ILL068V1 Evaluation Board.
System requirements:
Windows PC (XP, Vista, 7), USB 2.0, A to mini-B cable
- Connect the board to the supply (be sure jumpers Jumper are set as shown)
- Get a ST-Link/V2 (for IAR) or Rlink (for Raisonance)
- Connect the board to PC via SWIM Interface(using STLINK or RLINK, depending on your programming environment)

To PC USB port





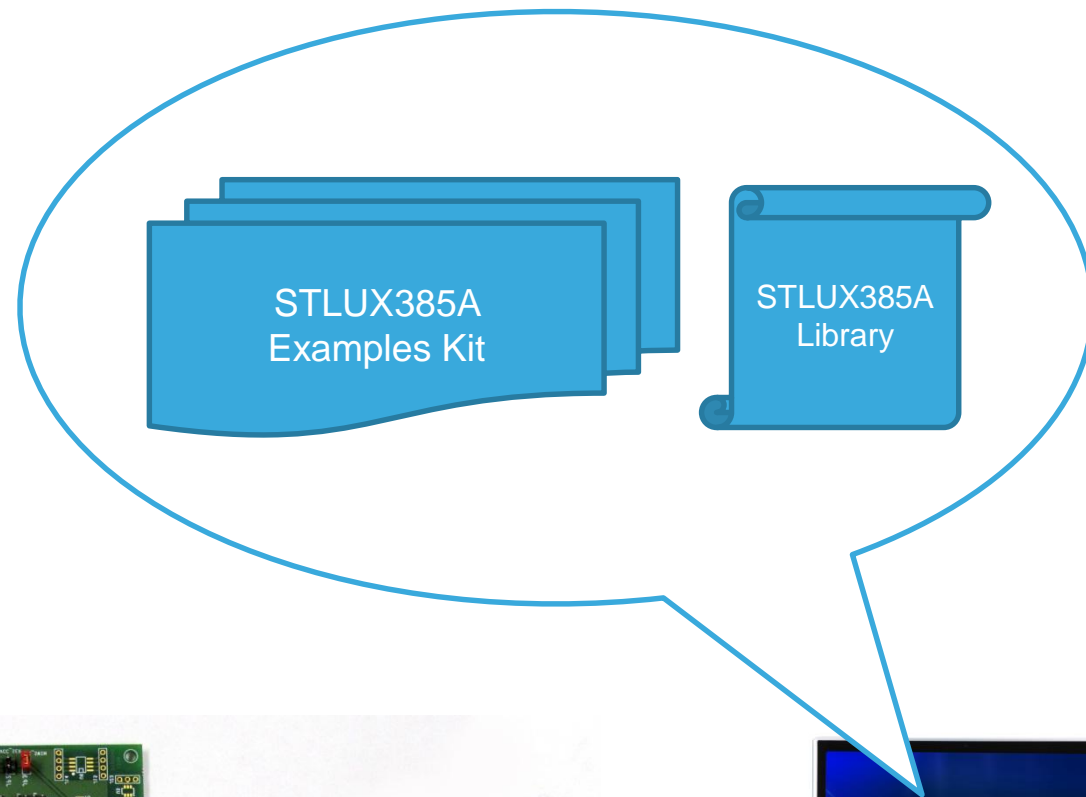
- Alternatively you can supply the board connecting it to a PC via Serial Interface (Be sure to adjust the jumpers JP1-JP2 as shown)
- Please note that the TTL-232R-3V3 USB Cable is not included with the STEVAL_ILL068V1 Evaluation Board and can be purchased directly from FTDI or from any online standard supplier.
- You can alternatively supply the board via SWIM interface in case you use the Raisonance-Rlink Microcontroller Interface





Design Flow step 4: Get into STLUX385A

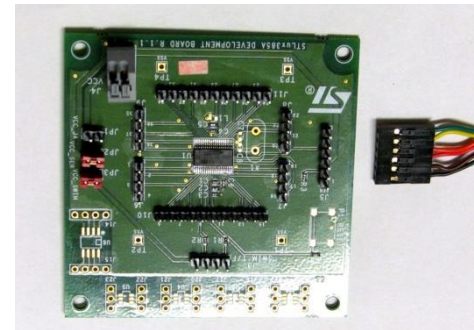
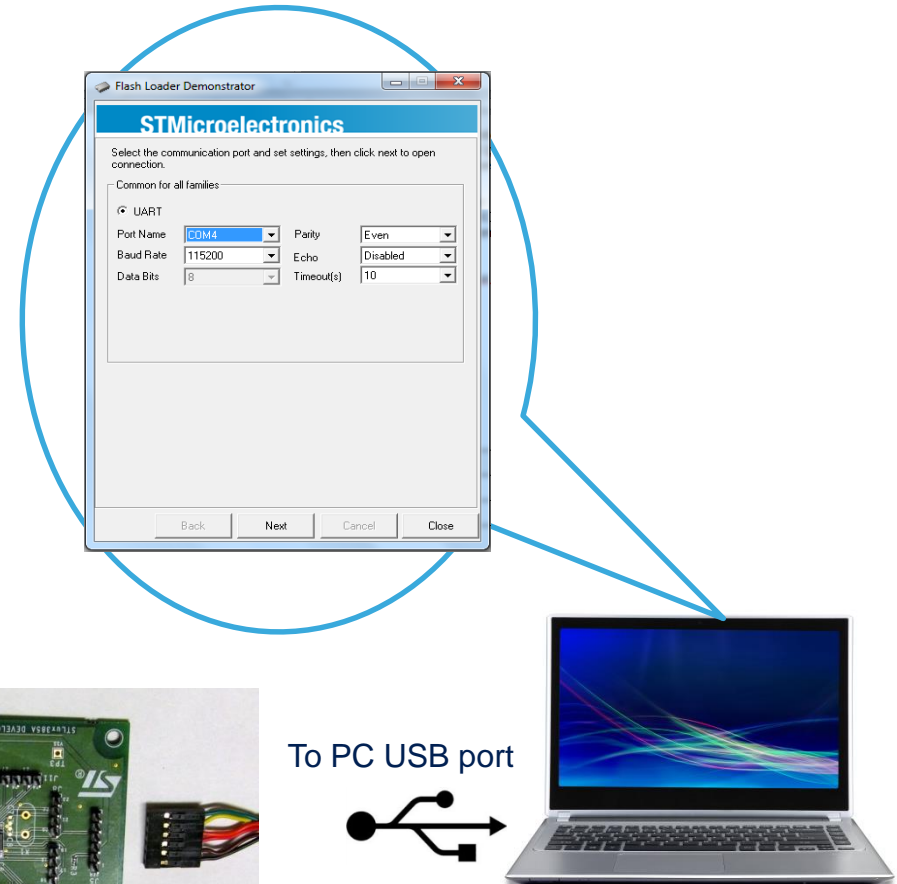
- Open IAR/Raisonance software development environment
- Check the STLUX385A Examples to quickly get hands on with the STLUX385A programming.
- Easily access STLUX385A SMEDs and peripherals thanks to the API provided by the STLUX385A Peripheral Library.
- Compile STLUX385A examples, download them to your STEVAL-ILL068V1 Evaluation Board using the STLINK/RLINK device.
- Run the firmware and verify your idea!





Design Flow step 5: Share your application

- Share your application by generating a bit file using your compiling IDE tools (.hex, .bit, etc.)
- The bit file can be downloaded to a board without compiling through Boot Loading aided procedure.
- Boot Loading procedure can be done with a brand new board or a properly configured board.
- Boot Loading procedure can be easily done interfacing your board connected via serial cable to the ST Flash Loader Demonstrator .



To PC USB port





Design Flow step 5: Boot Loading procedure

- A Brand-new board default setting enables Boot Loading in auto-detect mode that means at power-up the boot code starts polling all the possible boot code sources.
- Since the Flash Loader Demonstrator doesn't handle this autodetect mode, the STLUX Boot Loader should be configured to check only the desired channel for an easier connection. It is possible anyway to connect a board configured in autodetect mode but this requires to manually synchronize the Flash Loader Demonstrator with the polling mechanism.
- In order to enable the Boot Loader, the Read-Out protection for the FLASH must be disabled. So the address 0x4800h must be set to its default value 0x00h (brand-new devices are configured like that). If so the Loader can check the FLASH content of the 0x8000h address as specified below and start checking for boot sources according to the UART channel configuration.
- If FLASH is virgin, then the Boot Loader waits for an indefinite time for a connection on the set UART channel. If the FLASH is programmed and booting is allowed, the Boot Loader waits for one second checking for a connection on the set UART channel, then jumps to the code stored in the FLASH.
- So basically using a brand-new STLux device with the MSC_OPT0 Option byte properly configured to a single UART channel and a virgin FLASH allows to easily download your code through UART connection with the Flash Loader Demonstrator



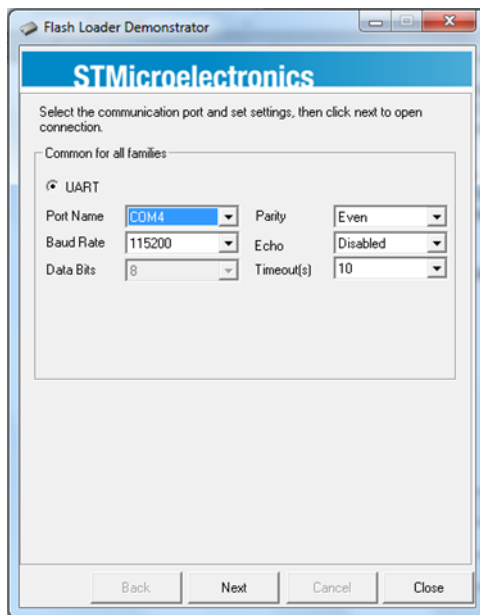
Design Flow step 5: Boot Loading Procedure

Checks	FLASH location 0x8000	check opt_byte 0x487E	check opt_byteN 0x487F	Actual FLASH status -> action
1 st	Not 0x82 && not 0xAC	Don't care	Don't care	FLASH virgin, jump into Boot Loader
2 nd	0x82 or 0xAC	0x55	0xAA	FLASH programmed booting allowed, jump into Boot Loader for 1 second
3 rd	0x82 or 0xAC	Not 0x55	Not 0xAA	FLASH programmed booting not allowed, jump immediately into the FLASH reset address

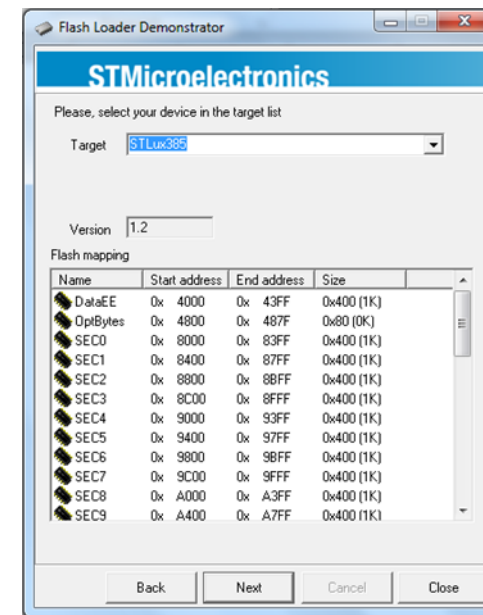
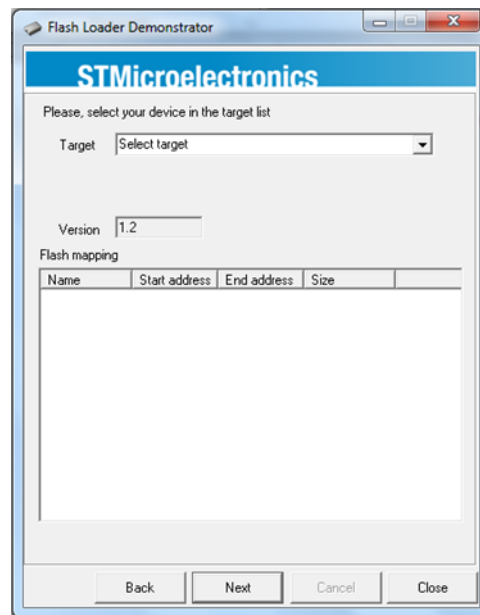
Boot Loading Procedure & Memory Checks



Design Flow step 5: Flash Loader Demonstrator



PC send the boot code byte and wait the response.

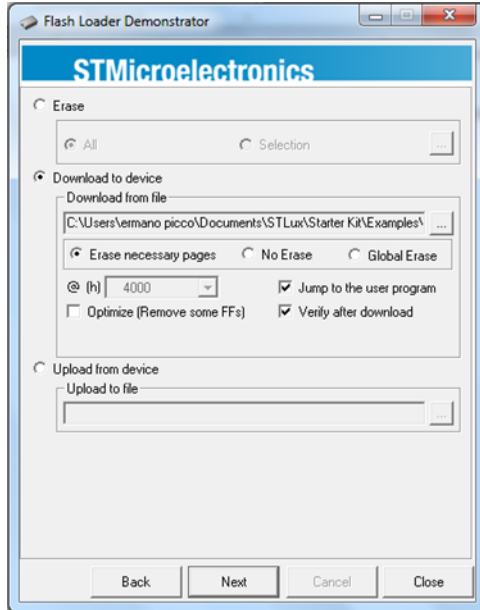


Select the right COM port for the connection and push the NEXT button. Then the connection is established and a new menu will appear.

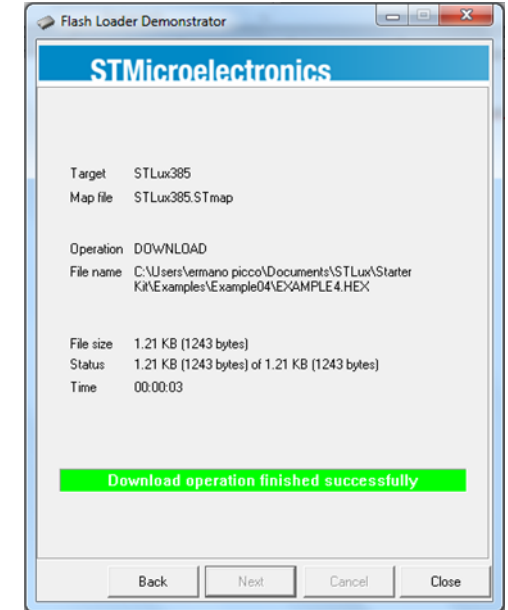
Select as a target device for the connection STLux385, then press NEXT to enter the next step



Design Flow step 5: Flash Loader Demonstrator

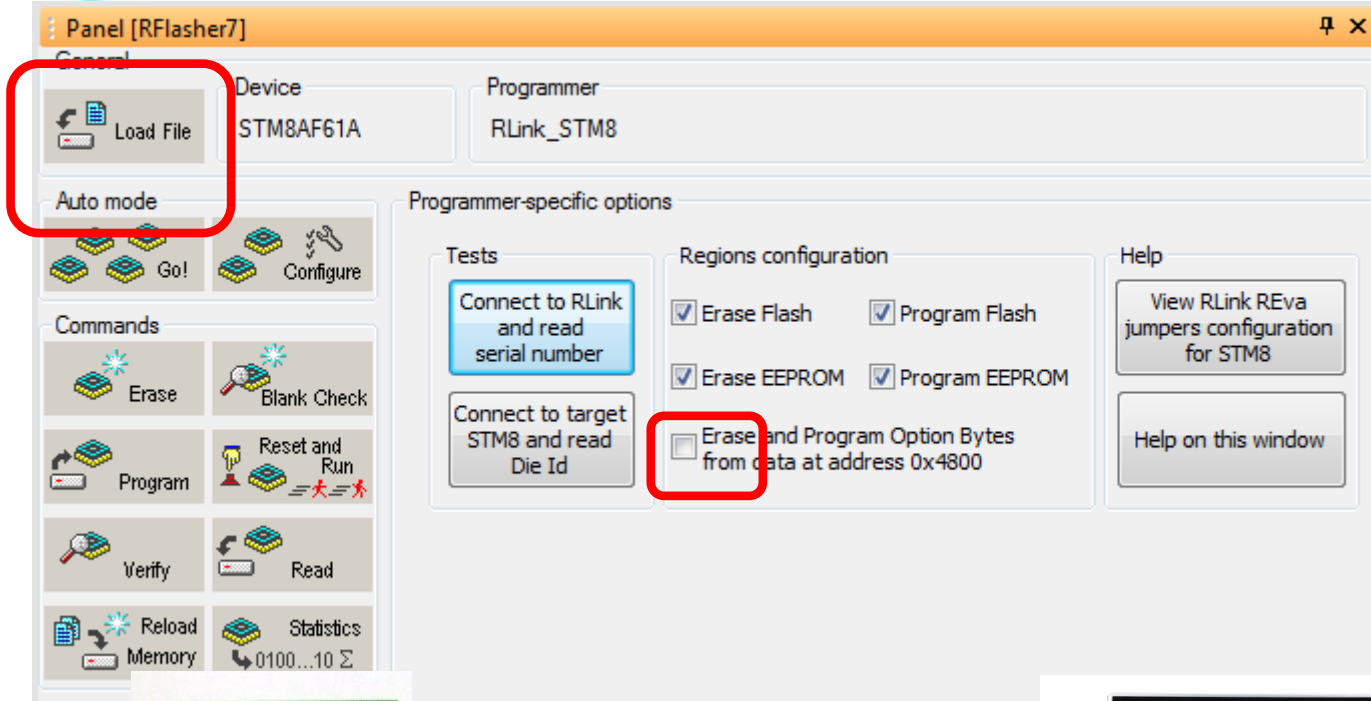


Now to download your code, you need to choose “Download to device” and to give the complete path and name of the file you want to download to STLux. The acceptable file formats are .bin and .hex. When you set the proper file indications, press NEXT to start downloading the code. You can follow the download process reading the bar in the next shell. When the download will be complete, the Flash Loader Demonstrator will highlight it





Design Flow step 5: Loading with Raisonance



Another possible way to download to your board the bit file of your application is to use the Raisonance Rflasher tool that is part of the Ride7 Toolsuite. The work panel of Rflasher allows you to connect the STLUX device via SWIM interface and Rlink to make some basic tests and operations. Among these options it easily allows to choose a bit file (.bin, .hex, etc) and download it to the device. For more information about the Rflasher tool, please check out the Raisonance website.

NOTE:

- If in your project the option byte is not defined, please disable the “Erase and Program Option Byte” field





Design Flow step 5: Loading with IAR Systems

Another possible way to download to your board the bit file of your application is to use the IAR Toolsuite. The IAR Systems environment allows you to connect the STLUX device via SWIM interface and STLink. Then you can chose a bit file (.hex, etc) and download it to the device. For more information about how to do it step-by-step, please check out the links on the IAR Systems website.



STEVAL-ILL068V1

Flat cable
↔



ST-LINK/V2

To PC USB port
→





Device summary

Available in a click

Part Number	Description	Document
STEVAL-ILL068V1	STLUX385A with all functionalities connected	DB2287
STSW-STLUXLIB01	STLUX385A Peripheral Library	UM1753
STSW-STLUXSMED01	STLUX385A SMED Configurator	UM1760
ST-LINK/V2	In-circuit debugger/programmer for STM8 and STM32	UM1075

For technical documentation, software download, on-line ordering visit the STLUX webpage:
www.st.com/stlux