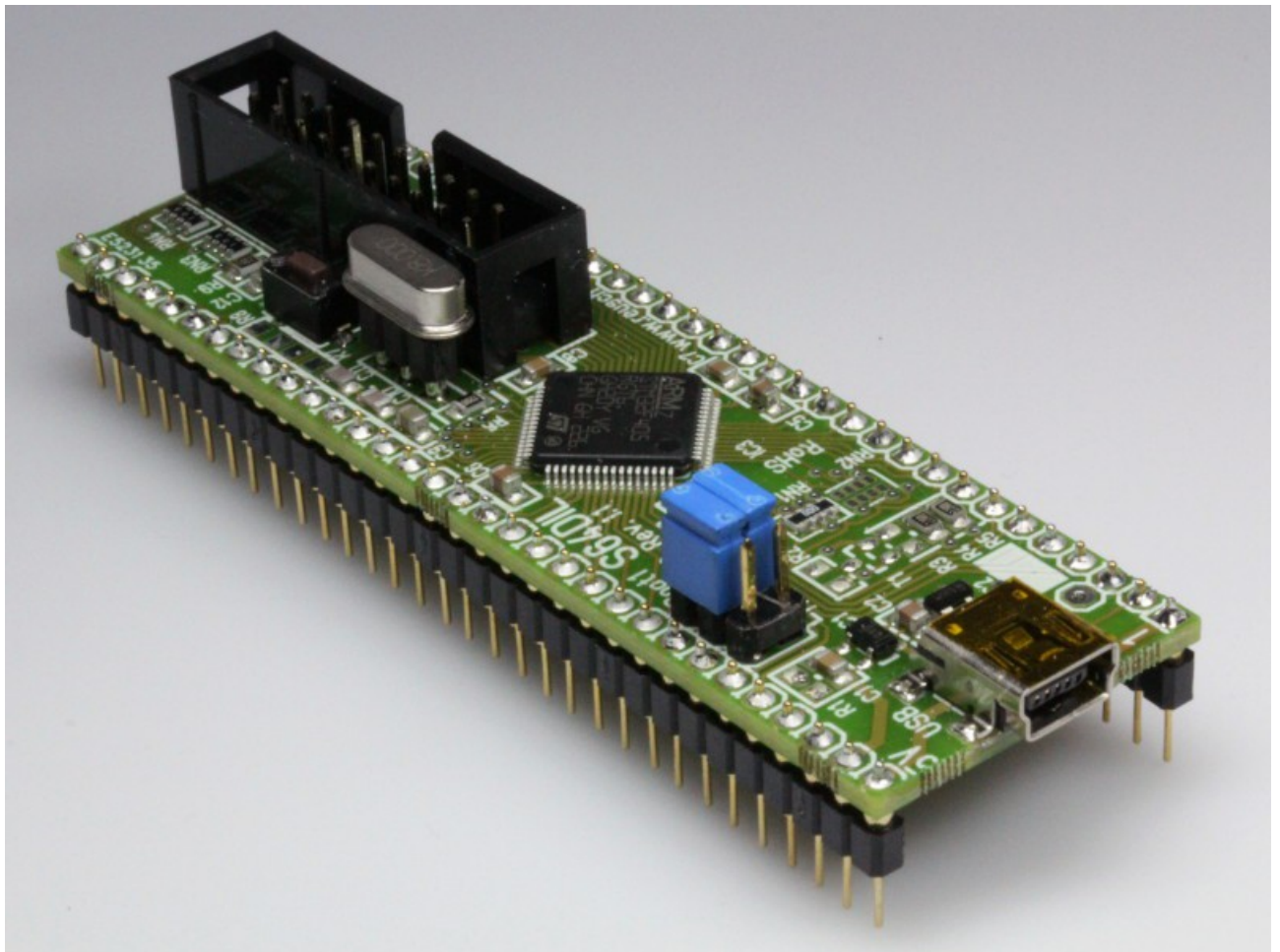


# S64DIL-405

## ARM Cortex-M4 USB Module

Rev. 1.1  
Documentation Rev. 2



Reusch Elektronik

© 2013 Reusch Elektronik, Dipl.-Ing. (FH) Rainer Reusch

[www.reusch-elektronik.de](http://www.reusch-elektronik.de)

<http://products.reworld.eu/x3dil.htm>

File: S64DIL-405\_Manual  
Created: 2013-03-16  
Changed: 2013-03-16

# Table of Contents

1. Introduction.....	1
1.1 Assembly Variations.....	1
1.2 Oscillator.....	1
1.3 Introduction Example.....	2
2. Technical Information.....	3
2.1 Schematic.....	3
2.2 Dual-Inline-Socket Pin Assignment.....	4
2.3 JTAG/SWD Pin Assignment.....	5
2.4 Technical Data.....	5
2.5 Dimensions.....	5
3. Application Notes.....	6
3.1 Power Supply Options.....	6
3.2 Memory Programming.....	7
3.3 Reflashing the bootloader.....	8
4. Concluding Remarks.....	8
4.1 Application Software.....	8
4.2 Links.....	8
4.3 Conformity statement.....	9
4.4 Statement and Disclaimer.....	9
4.4 Technical Support.....	9

## Please note:

This document refers to revision 1.1 of the S64DIL-405 module. If you are using a device with another revision number, please refer to the corresponding documentation!

If you are using the S64DIL-103 module, please refer to the manual, especially designed for this module.

---

## Reusch Elektronik

Dipl.-Ing. (FH) Rainer Reusch

Blumenstr. 13  
D-88097 Eriskirch  
Germany

Phone: +49-7541-81484  
Fax: +49-7541-81483  
E-Mail: [info@reusch-elektronik.de](mailto:info@reusch-elektronik.de)  
Homepage: [www.reusch-elektronik.de](http://www.reusch-elektronik.de)

© 2013 Reusch Elektronik, Dipl.-Ing. (FH) Rainer Reusch

This document is protected by copyright law. It is prohibited to copy or distribute without permission of Reusch Elektronik.

# 1. Introduction

*S64DIL-405* is a tiny interface module for an ARM Cortex-M4 micro controller with USB interface in LQFP64 casing. The module fits to an ordinary 64 pin dual-in-line socket with 0.9 inch row distance. The S64DIL-405 module is helpful in experiments, development and prototyping with USB-ARM micro controllers.

The module contains the crystal oscillator. A socket is used, so it is easy to replace the crystal or ceramic resonator to get the required frequency.

The ESD protected USB interface (mini USB plug) is on board.

The module contains a 20 pin header for in-system-programming and debugging (JTAG/SWD standard). These pins are also available by the dual-in-line connector pins.

A reset can be done by the on board button.

The S64DIL-405 module is easy to use, because it contains all necessary and time critical interfaces. On the other hand it gives full flexibility in power supplement and connectivity!

## 1.1 Assembly Variations

The module is available in two assembly versions:

- as “S64DIL-103”, assembled with STM32F103RGT6 (refer to specific manual)
- as “S64DIL-405”, assembled with STM32F405RGT6 (described in this manual)

The printed circuit board is also available “naked” (not assembled). It might be suitable for all STM32Fxxx controllers in LQFP64 casing. The board is tested for STM32F103 and STM32F405 controllers.

## 1.2 Oscillator

A 3 pin socket is used for the oscillator. This makes the module suitable for a

- crystal (HC49)
- 2 pin ceramic resonator
- 3 pin ceramic resonator

All components in wired technology (no SMD). It might be necessary, to remove the two 22pF capacitors (C10 and C11), if a 3 pin ceramic resonator is used.

The XTAL pins are also accessible at pin 37 (Oscillator input) and pin 38 (Oscillator output), if R7 and R8 will be populated. In this case, also an external clock source can be connected to pin 37.

If the USB engine is clocked by the crystal, the following frequencies are recommended:

- 8.000MHz (delivery standard)
- 12.000MHz

### 1.3 Introduction Example

The wiring example as shown, supplies the module by USB and the onboard 3.3V regulator. Programming can be done via USB or serial interface (3.3V levels!) and bootloader or JTAG/SWD connector.

Please refer to controller data sheet and help of your development environment, how to program and debug the controller of your S64DIL module. It contains a boot jumper. Jumper settings are determining, if firmware in the flash, the bootloader (system memory) or firmware in SRAM is launching. Set the boot jumper as follows:

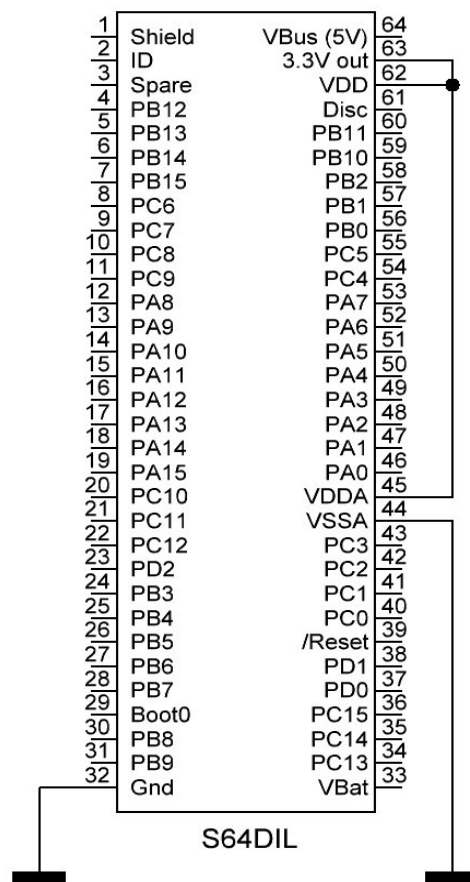
Boot 0	Boot 1	
L	X	Flash
H	L	System
H	H	SRAM

The controller of the module is coming with the factory default bootloader, which is supporting firmware updates via USB, UART and CAN interface of the controller. The program *DfuSe* has to be used to flash the controller via bootloader.

To launch the selected memory, a reset has to be done (Reset button).

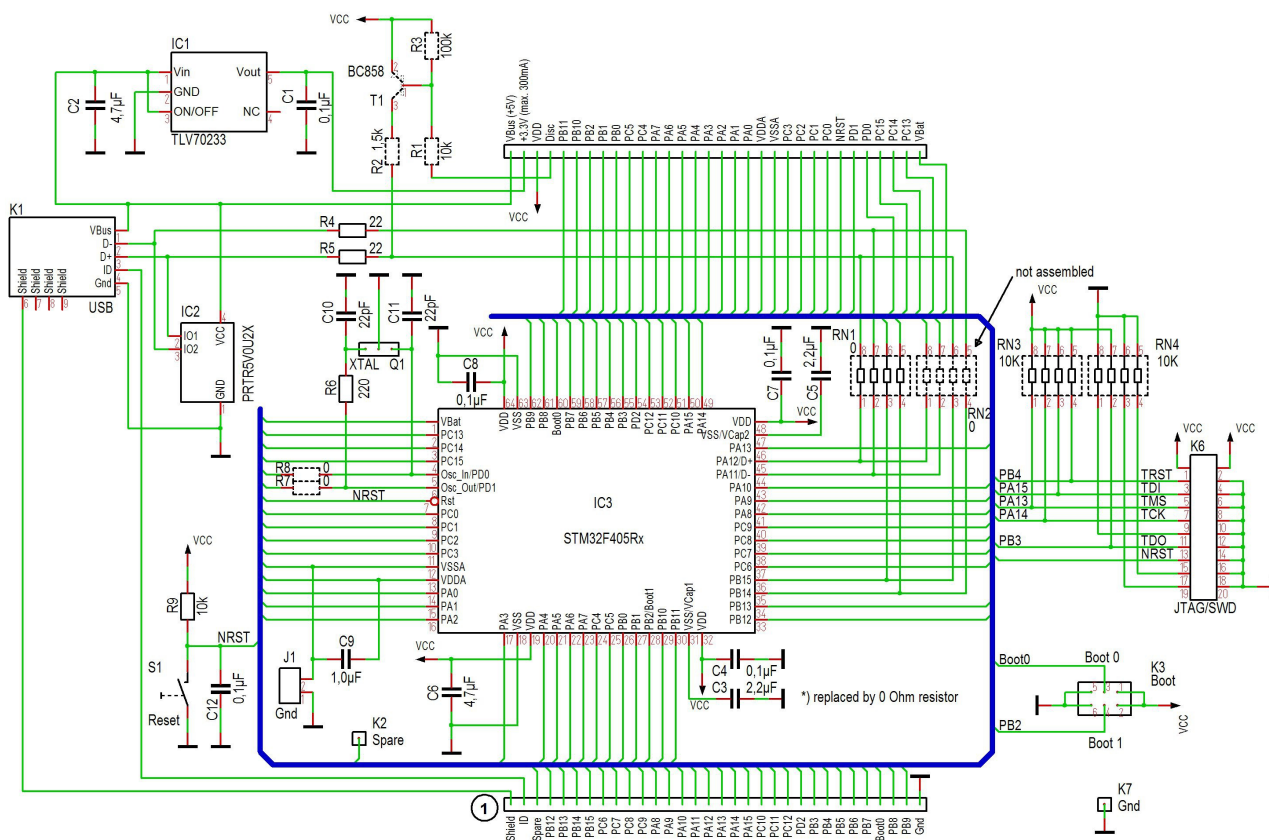
**Please note:**

There are more options, to supply the device. The device also can be programmed via *JTAG/SWD* interface and *JTAG/SWD* programmer (such as ST-Link/V2). For more information read the application note chapter, please.



## 2. Technical Information

### 2.1 Schematic



#### Notes:

By factory default RN1 is assembled and RN2 is not. That means, PA11 (D-) and PA12 (D+) are not available at DIL pins. Remove RN1 and populate RN2, if the *High Speed USB interface* of the controller has to be used. In this case, PB14 and PB15 are not usable as I/O ports. This option is not tested!

The pins 37 and 38 are carrying the crystal signal (when a crystal is used), if R7 and R8 are assembled (0Ω resistors) by user. The population of this two resistors in experimental environments may cause improper crystal operation.

Pin 64 is the 5V output of the USB supply voltage. The controller is designed for supply voltages up to 3.6V. That means: never connect pin 64 to any other pin of the module, direct or indirect! A connection will cause a damage!

#### USB connection and detection:

Pin 61 (Disc) has no function (no connection). USB detection is done by the USB hardware of the controller.

## 2.2 Dual-Inline-Socket Pin Assignment

Pin	Function
1	Shield (USB)
2	ID (USB)
3	Spare
4	PB12
5	PB13
6	PB14
7	PB15
8	PC6
9	PC7
10	PC8
11	PC9
12	PA8
13	PA9
14	PA10
15	PA11 (D-) *
16	PA12 (D+) *
16	PA13 (TMS)
18	PA14 (TCK)
19	PA15 (TDI)
20	PC10
21	PC11
22	PC12
23	PD2
24	PB3 (TDO)
25	PB4 (TRST)
26	PB5
27	PB6
28	PB7
29	Boot 0
30	PB8
31	PB9
32	GND



Function	Pin
VBus (5V output)	64
3.3V output (300mA)	63
VDD	62
n.c.	61
PB11	60
PB10	59
PB2 (Boot 1)	58
PB1	57
PB0	56
PC5	55
PC4	54
PA7	53
PA6	52
PA5	51
PA4	50
PA3	49
PA2	48
PA1	47
PA0	46
VDDA	45
VSSA	44
PC3	43
PC2	42
PC1	41
PC0	40
Reset	39
PD1 (Osc out) *	38
PD0 (Osc in) *	37
PC15	36
PC14	35
PC13	34
VBat	33

\*) The pins 15, 16, 37, 38 and 61 are not connected by factory default.

For more information about the alternative pin usage, please refer to the controller data sheets.

**Warning!** Never connect VBus (pin 64) to any other pin of the device!

## 2.3 JTAG/SWD Pin Assignment

The in-system programming and debugging adapter is a 20 pin header, which corresponds with the JTAG/SWD standard. Therefore the usual in-system programmers/debuggers (such as “ST-Link/V2” or “Segger JLink”) are suitable.

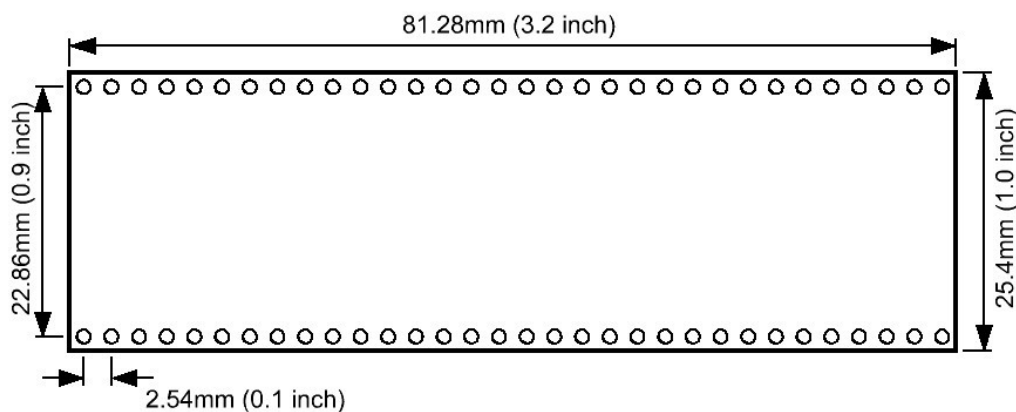
Function	Pin	Pin	Function
VCC	1	2	VCC
TRST	3	4	Gnd
TDI	5	6	Gnd
TMS	7	8	Gnd
TCK	9	10	Gnd
	11	12	Gnd
TDO	13	14	Gnd
NRST	15	16	Gnd
	17	18	Gnd
	19	20	Gnd

## 2.4 Technical Data

Supply Voltage	2.0V to 3.6V
Supply Current	<60mA (depends on application)
onboard voltage regulator	3.3V / 300mA
Clock Frequency	up to 168MHz (8 or 12MHz crystal is recommended, if USB engine is clocked external)
Operation Temperature	-25°C to 70°C
Storage Temperature	-25°C to 85°C

For further information related to the microcontroller, please refer to the data sheet.

## 2.5 Dimensions



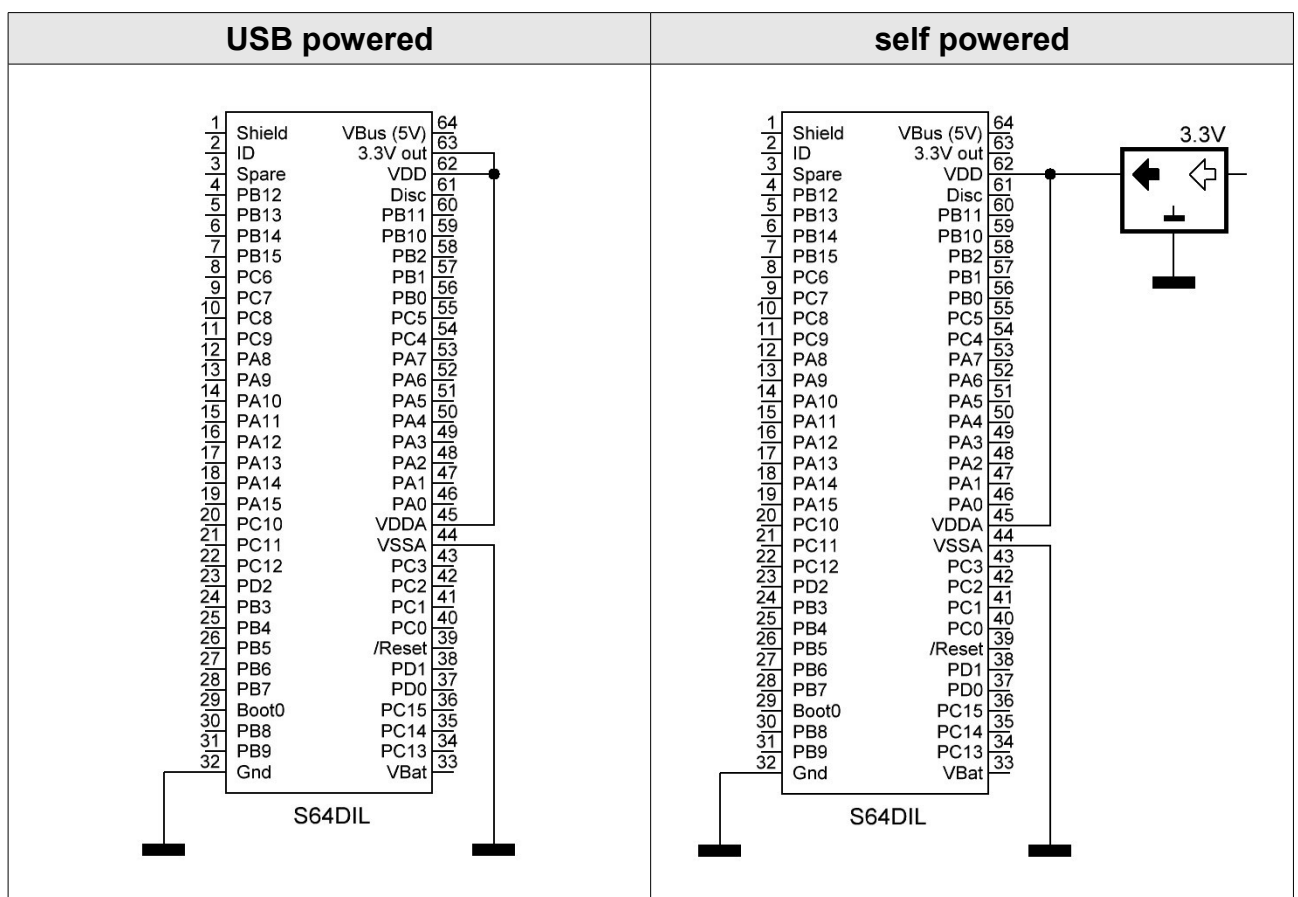
## 3. Application Notes

### 3.1 Power Supply Options

The module can be supplied by USB bus power with the on board 3.3V regulator or it can be self powered by an external voltage regulator. The USB supply voltage (+5.0V) is available at pin 64. **Please note:** The controller allows voltages up to 3.6V. Never connect VBus (pin 64) to any other pin of the module!

For USB powered supply (usage of the onboard voltage regulator), only a few connections are necessary.

For self powered supply a stabilized supply voltage of 3.3V (2.0V to 3.6V) is required. Use the circuitry as shown. Note, that the supply is connected to VCC and VDDA pin. VSSA has to be grounded. The 3.3V output (pin 63) is leaved open!



For further information please refer to the data sheet of the microcontroller.

**Please note:** The VBus pin (pin 64) outputs +5.0V from the USB supply voltage. Never connect this pin to any other pin of the S64DIL-405 device! The controller can handle voltages up to 3.6 volts only!



## 3.2 Memory Programming

The STM32 microcontroller on the S64DIL-405 module is shipped with the original DFU bootloader from STMicroelectronics. So there are two ways, to program the flash memory of the microcontroller.

### 3.2.1 Bootloader and DfuSe

To launch the bootloader, set Boot0 jumper to *High* and Boot1 jumper to *Low* (system boot). Push the reset button.

The USB connector can be used for firmware update.

The bootloader also supports serial communication. Therefore a serial interface connection (RxD and TxD) to a Windows Personal Computer is required. The voltage levels of this two lines have to have a value of 3.3V!

Launch the installed DfuSe application (downloaded from STMicroelectronics homepage) and flash the firmware.

After flashing set Boot0 jumper to *Low* (Boot1 jumper can be removed, so port is available for application). After reset the application firmware is launched.

The *DfuSe* software for Windows can be downloaded from:

[http://www.st.com/internet/com/SOFTWARE\\_RESOURCES/SW\\_COMPONENT/SW\\_D EMO/umo412.zip](http://www.st.com/internet/com/SOFTWARE_RESOURCES/SW_COMPONENT/SW_D EMO/umo412.zip)

### 3.2.2 In-System-Programming

The S64DIL-405 module contains an JTAG/SWD connector for in-system-programming and debugging. A suitable programming device (such as “ST-Link/V2” or “Segger Jlink” with flash option) is required.

The advantages of this programming method:

- full control over the microcontroller and all memories
- comfortable integration into development environment (such as “CooCox CoIDE”)
- standalone programming with “STM32 ST-LINK Utility”
- no bootloader required
- suitable for any clock frequency

Note: This programming method allows a “chip erase”. That means, the bootloader can be erased (the flash memory space can be used for the application). On the other hand, the device no longer can be programmed via bootloader! If a firmware update should be possible later without programmer, the DFU bootloader has to be reprogrammed by this method.

### 3.2.3 Debugging with JTAG/SWD

The JTAG/SWD interface is also suitable for debugging. Usage of SWD is recommended. All modern IDEs (such as “CooCox CoIDE”) are supporting a comfortable source code debugging within the IDE.

An interface transfer speed of 1MHz and more is possible. Reduce the speed, if communication problems are occurring!

#### **Please note:**

The pins of the JTAG/SWD interface can't be used as usual I/O ports, as long as debugging mode is active!

## 3.3 Reflashing the bootloader

After a “chip erase” (e.g. when the controller was programmed via JTAG/SWD) it is necessary, to reflash the DFU bootloader, if further programming should be possible by serial interface.

Reflashing has be done via JTAG/SWD interface.

## 4. Concluding Remarks

### 4.1 Application Software

You will find programming and application sample software on the product homepage. Please visit the following URL:

<http://products.reworld.eu/s64dil.htm>

You will find additional information and tools on the following URLs.

Data sheets, libraries, examples, tools and software for the microcontroller:

<http://www.st.com/internet/mcu/product/252144.jsp>

Free *Eclipse* based development environment “CooCox CoIDE”:

[http://www.coocox.org/CooCox\\_CoIDE.htm](http://www.coocox.org/CooCox_CoIDE.htm)

GNU tools for ARM embedded processors (ARM GCC)

<https://launchpad.net/gcc-arm-embedded/>

STM32 tutorial and examples (german)

<http://www.diller-technologies.de/stm32.html>

### 4.2 Links

Visit the following links for more information and downloads about this product, used software modules and tools.

*Reusch Elektronik* homepage for electronic products:

<http://products.reworld.eu>

LibUSB-Win32, USB drivers for Microsoft Windows ®:

<http://libusb-win32.sourceforge.net/>

### 4.3 Conformity statement

This evaluation board/kit is intended for use for **engineering development, demonstration or evaluation purposes only** and is not considered by *Reusch Elektronik* to be a finished end-product fit for general consumer use.

This evaluation board/kit does not fall within the scope of the European Union directives regarding electromagnetic compatibility, restricted substances (RoHS), recycling (WEEE), FCC, CE or UL, and therefore may not meet the technical requirements of these directives or other related directives.

### 4.4 Statement and Disclaimer

This electronic device is designed under best known engineering guidelines. It confirms the appropriate design rules. No warranty or liability is given for adherence, assured properties, or damages which might be caused by the usage of this hardware or the accessory software.

**Note:** This equipment has been tested and found to comply with the limits for a Class B digital device. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy, and if not installed and used in accordance with the instruction manual, may cause interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation.

### 4.4 Technical Support

We hope you'll understand, that technical support by phone is not possible. If You have any questions, technical problems or You need information about other products from *Reusch Elektronik*, please contact us by e-mail or Fax.

E-Mail: [support@reusch-elektronik.de](mailto:support@reusch-elektronik.de)

FAX: +49-7541-81483

We are able to handle inquiries in english and german language.

Don't hesitate to contact us, if You have demand for custom specific solutions!