TestUSB – USB Communication Framework for the AT90USBKEY2 device

First published: 2014-05-11

Author: Eckhard Kantz, e.kantz@wegalink.eu

Objectives

In 2007, Dr. Stefan Salewski published a firmware that allowed for communicating with a AT90USBKEY2 device over the integrated in that microcontroller USB interface. This project extends that firmware with a Windows client software.

Software Tools

Besides an AT90USBKEY2 device hardware, the following software tools need to be installed in order to make use of this project's sources:

- Atmel Studio 6 (see http://www.atmel.com)
- FLIP (see http://www.atmel.com/tools/flip.aspx)
- Zadig (see http://zadig.akeo.ie/)
- Microsoft Visual Studio 2013 Express (see http://www.microsoft.com/de-de/download/details.aspx?id=40787)

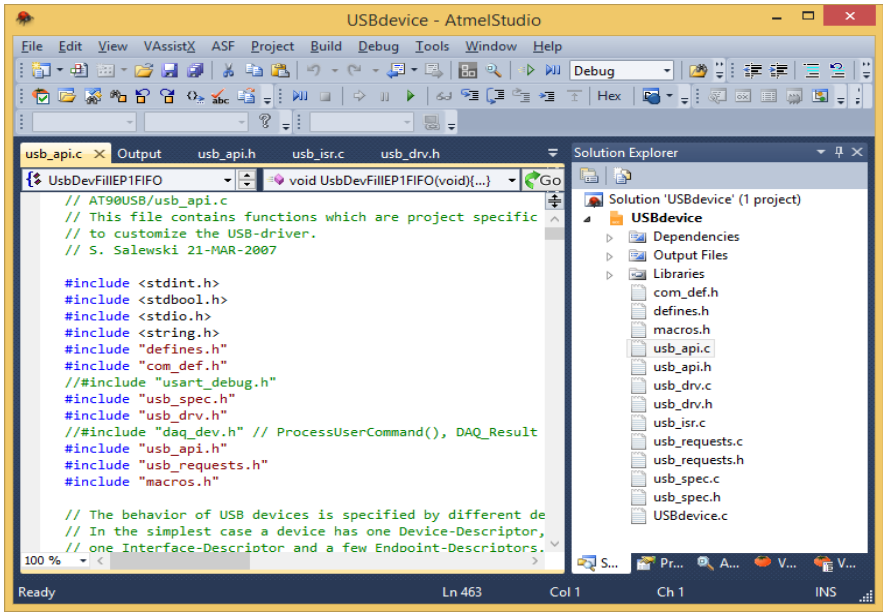
All tools can be downloaded and be used free of charge. Further, the framework is based on the following USB library which is already contained in the project:

- libusb-1.0 (see http://www.libusb.org/wiki/libusb-1.0)

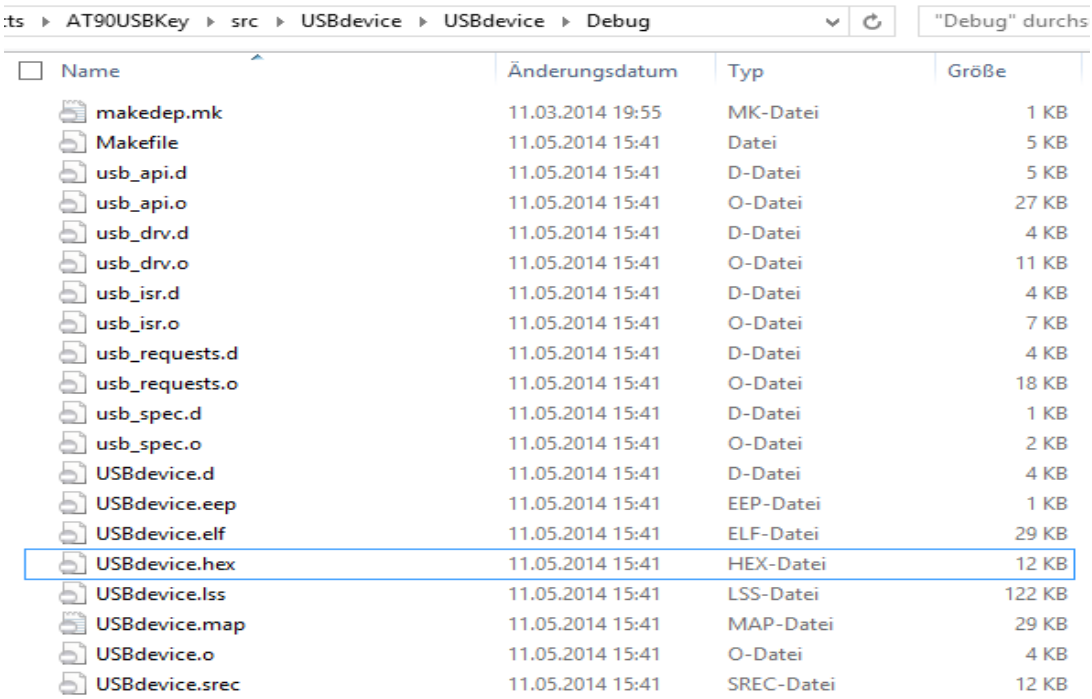
Compile and Load

The following steps generate the runtime on microcontroller as well as on PC side:

1. Open the USBdevice.atsln file in Atmel Studio 6 and generate the runtime for the microcontroller.



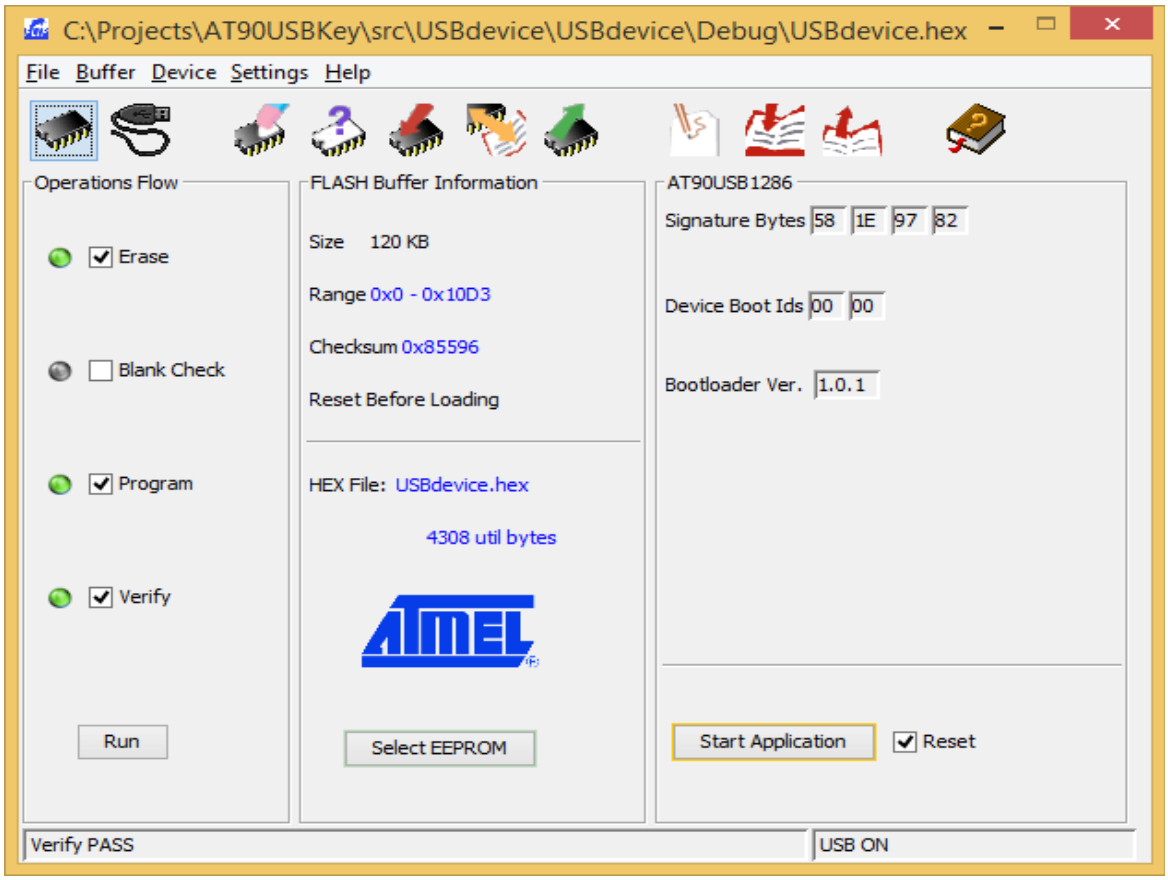
2. Find the generated firmware file (USBdevice.hex):



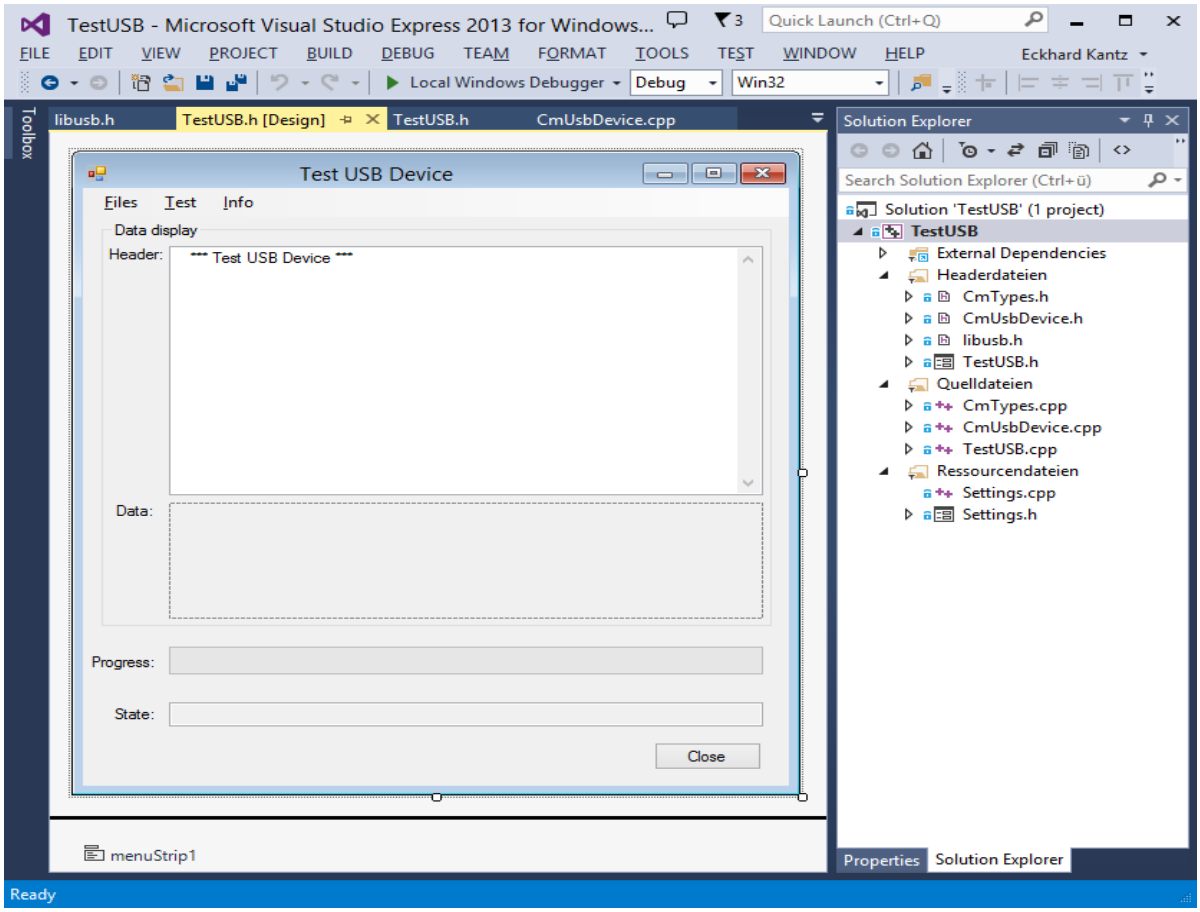
:ts ▸ AT90USBKey ▸ src ▸ USBdevice ▸ USBdevice ▸ Debug "Debug" durchs

Name	Änderungsdatum	Typ	Größe
makedep.mk	11.03.2014 19:55	MK-Datei	1 KB
Makefile	11.05.2014 15:41	Datei	5 KB
usb_api.d	11.05.2014 15:41	D-Datei	5 KB
usb_api.o	11.05.2014 15:41	O-Datei	27 KB
usb_drv.d	11.05.2014 15:41	D-Datei	4 KB
usb_drv.o	11.05.2014 15:41	O-Datei	11 KB
usb_isr.d	11.05.2014 15:41	D-Datei	4 KB
usb_isr.o	11.05.2014 15:41	O-Datei	7 KB
usb_requests.d	11.05.2014 15:41	D-Datei	4 KB
usb_requests.o	11.05.2014 15:41	O-Datei	18 KB
usb_spec.d	11.05.2014 15:41	D-Datei	1 KB
usb_spec.o	11.05.2014 15:41	O-Datei	2 KB
USBdevice.d	11.05.2014 15:41	D-Datei	4 KB
USBdevice.eep	11.05.2014 15:41	EEP-Datei	1 KB
USBdevice.elf	11.05.2014 15:41	ELF-Datei	29 KB
USBdevice.hex	11.05.2014 15:41	HEX-Datei	12 KB
USBdevice.lss	11.05.2014 15:41	LSS-Datei	122 KB
USBdevice.map	11.05.2014 15:41	MAP-Datei	29 KB
USBdevice.o	11.05.2014 15:41	O-Datei	4 KB
USBdevice.srec	11.05.2014 15:41	SREC-Datei	12 KB

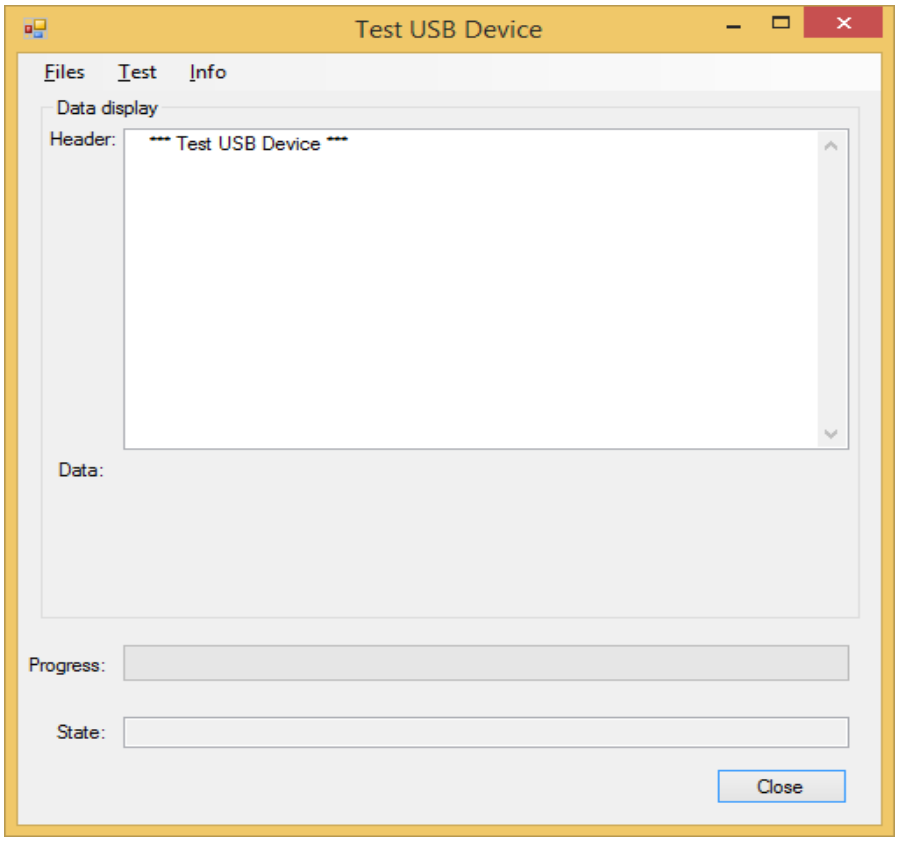
3. Load the firmware into the AT90USBKEY2 device. Note that the two buttons on the device have to be pressed and the Reset button has to be released first in order to be able to connect to the device with FLIP. After firmware upload, the Reset button has to be pressed again in order to have the firmware started properly:



4. Open the TestUSB.sln file in Visual Studio 2013 Express and compile the Windows client software:

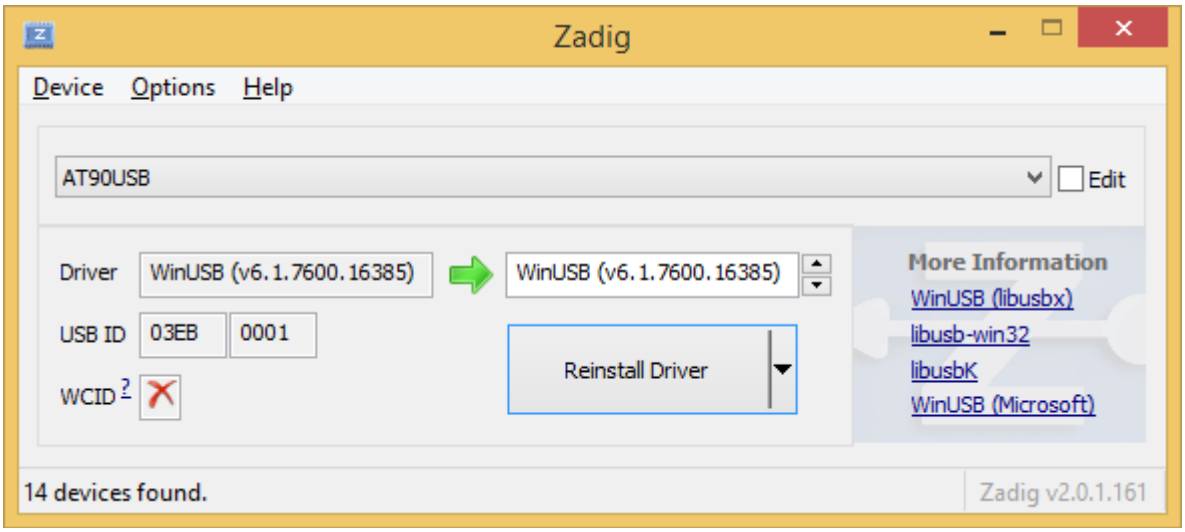


5. Run the TestUSB client software inside Visual Studio or start it directly:



Install a driver

Instead of writing a specific driver for the AT90USBKEY2 device, a USB driver tool is used to install a generic driver for that device which is subsequently used by the libusb-1.0 library for USB communication. The 'Zadig.exe' tool can be started without prior installation. After selecting the AT90USB device, the WinUSB driver should be assigned:



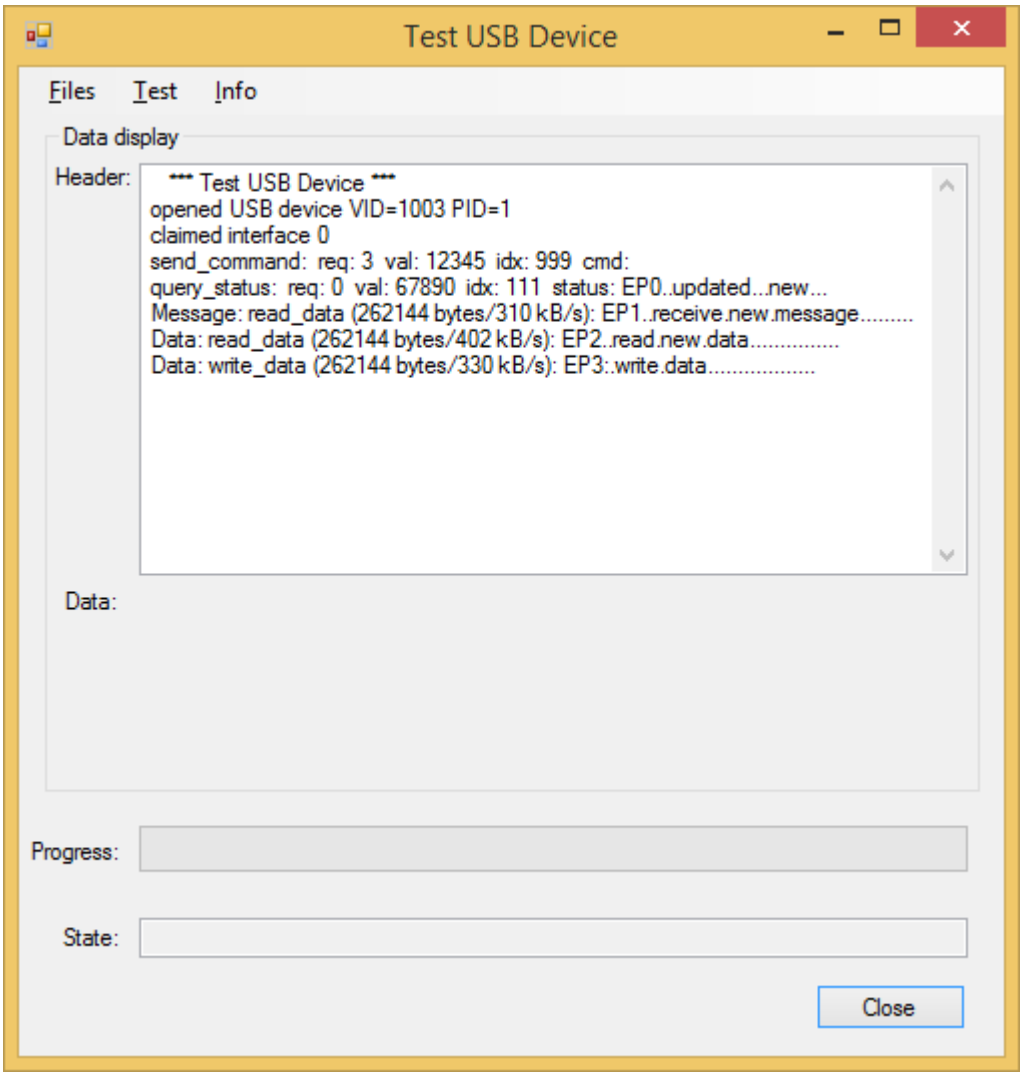
Run a Test

Once the previous steps have been successfully performed, a first test can be done with the AT90USBKEY2 device. Please do not forget to press the Reset button after firmware upload in order to get back from programming mode into operation mode.

The following commands from the menu should be run in order to check whether the device has successfully loaded recently compiled firmware:

- Open Device
- Claim Interface
- Send Command
- Query Status
- Receive Message
- Read Data
- Write Data

If the firmware and software client were compiled, loaded and started properly then an output similar to the following should be displayed:



Functionality

The following functionality has been implemented as a starting point for apllication specific extensions:

- Open/Close a USB device
- Discover attached USB devices
- Discover endpoints of the AT90USBKEY2 device
- Claim interface
- Send a command: an 8-bit request code along with a 16-bit value and an index is sent to a "vendor specific" subroutine on the device which can be used to start and stop any functionality.
- Query a status: a data field of upto 63 bytes can be returned to the PC in response to an 8-bit request code and a 16-bit value and index.
- Message: the AT90USBKEY2 device can send any message to the PC, e.g. to transfer status or debug information asynchronuously. In the framework, a 256 kByte data field is used that is evaluated for transfer speed based on a 64 byte buffer of the message endpoint.
- Read data: data can be received from the AT90USBKEY2 device on maximal speed using a double buffer of 64 byte each. In the framework, a 256 kByte data field is used that is evaluated for transfer speed.
- Write data: data can be transferred to the AT90USBKEY2 device on maximal speed using a double buffer of 64 byte each. In the framework, a 256 kByte data field is used that is evaluated for transfer speed.

Application

Implemented communication facilities should make it easier to develop own applications with AT90USBKEY2 using USB communication. Besides command and data transfer capabilities, there is also a message channel that could be used e.g. for transferring debug and status information.

Repository

The project sources have been made available on GitHub. Please clone the project from the following location in order to transfer all data to a local project copy:

https://github.com/WegaLink/AT90USBKEY2

For information about how to use GitHub, please refer to the documentation on the following website:

https://github.com/