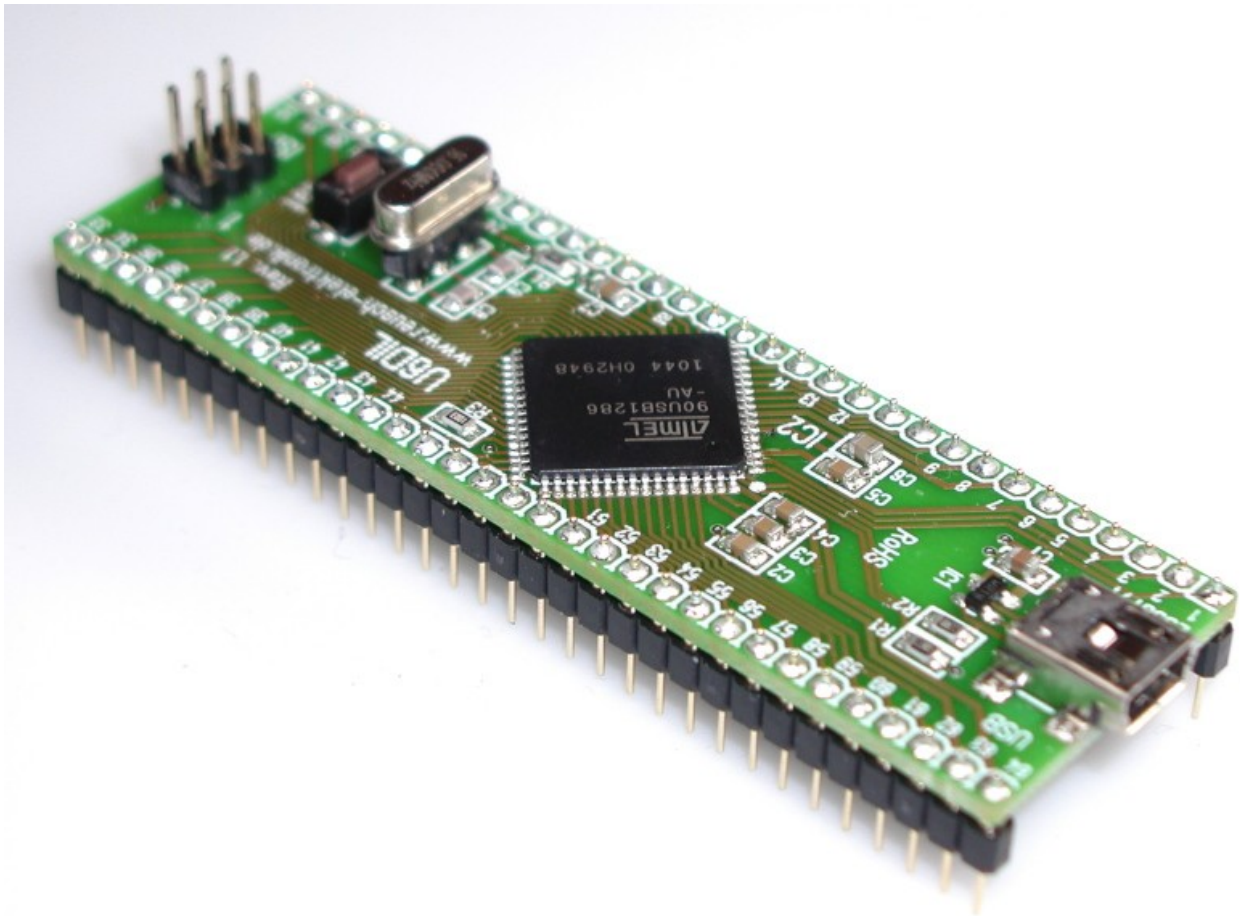
U6DIL

AVR USB Module

Rev. 1.1
Documentation Rev. 18



Reusch Elektronik

www.reusch-elektronik.de

http://products.reworld.eu/u6dil.htm

File: U6DIL_Manual
Created: 2011-02-22
Changed: 2011-03-31

Table of Contents

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
Homepage: www.reusch-elektronik.de

1. Introduction

U6DIL is a tiny interface module for an Atmel AVR micro controller with USB interface in TQFP64 casing. The module fits to an ordinary 64 pin dual-in-line socket. The U6DIL module is helpful in experiments, development and prototyping with USB-AVR 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 desired frequency.

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

The module contains a 6 pin header for in-system-programming (Atmel 6 pin ISP standard). These pins are also available by the dual-inline connector pins.

The U6DIL 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 one assembly version:

- as “U6DIL-1286”, assembled with AT90USB1286 from Atmel

The printed circuit board is also available “naked” (not assembled). It is suitable for the following Atmel microcontrollers with TQFP64 casing:

- AT90USB646
- AT90USB647
- AT90USB1286
- AT90USB1287

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

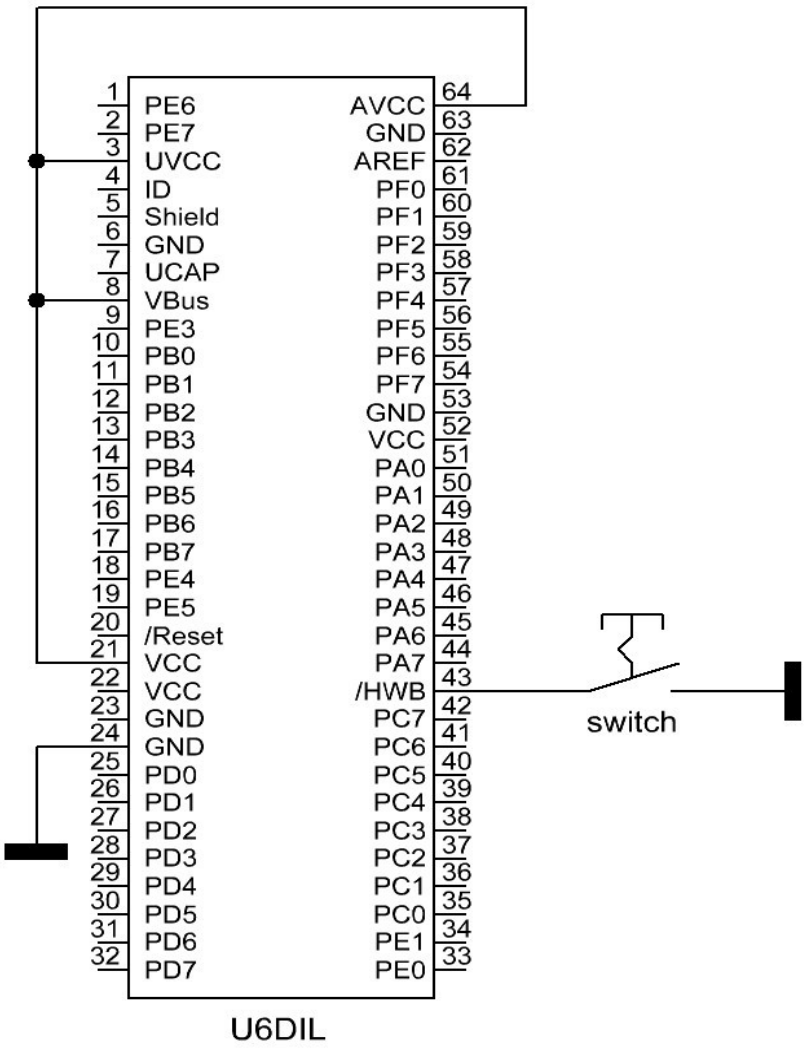
All devices in wired technology (no SMD). It might be necessary, to remove the two 22pF capacitors (C8 and C9), if a 3 pin ceramic resonator is used.

For correct USB function, the following frequencies are recommended:

- 8.000MHz
- 16.000MHz (delivery standard)

1.3 Introduction Example

The wiring example as shown, supplies the module by USB. Programming can be done via USB, bootloader and “Atmel Flip”.



Without firmware, the module always will enter the bootloader. It will respond as “Atmel device” at the personal computer. At the first plugging, it is asked for a driver installation. You have to use the LibUSB-Win32 drivers, which are coming with the “Atmel Flip” installation package.

After the driver installation, the device can be accessed by “Atmel Flip” to program your firmware application. After programming, the device has to be unplugged from USB (turn off power supply) and the $\overline{\text{HWB}}$ switch has to be changed to “open” state.

When the device is plugged again, it will run with your programmed firmware.

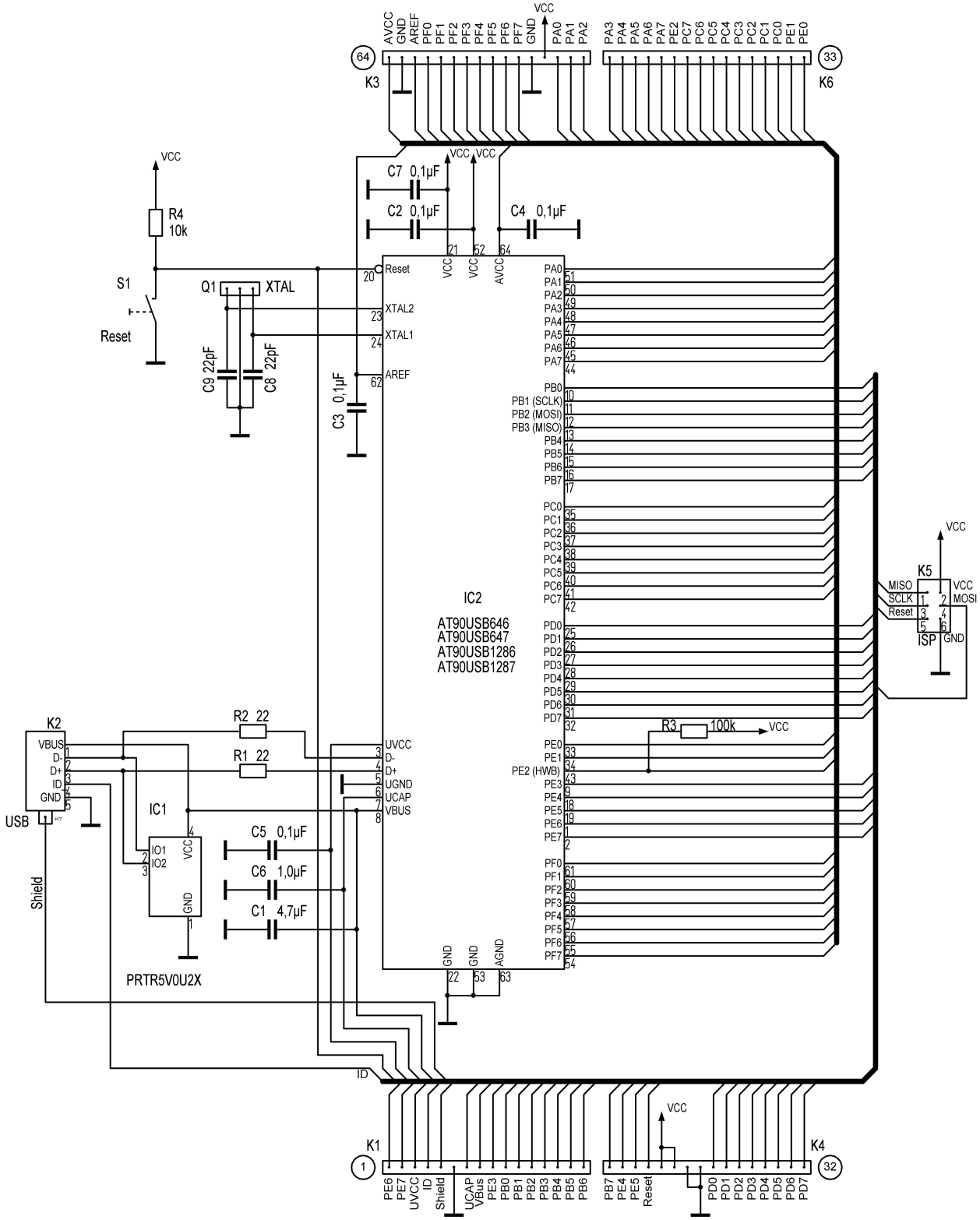
To reenter the bootloader, close the $\overline{\text{HWB}}$ switch and push the reset button for a moment.

Please note:

There are more options, to supply the device. The device also can be programmed via *ISP* interface and ISP programmer or JTAG interface. For more information read the application note chapter, please.

2. Technical Information

2.1 Schematic

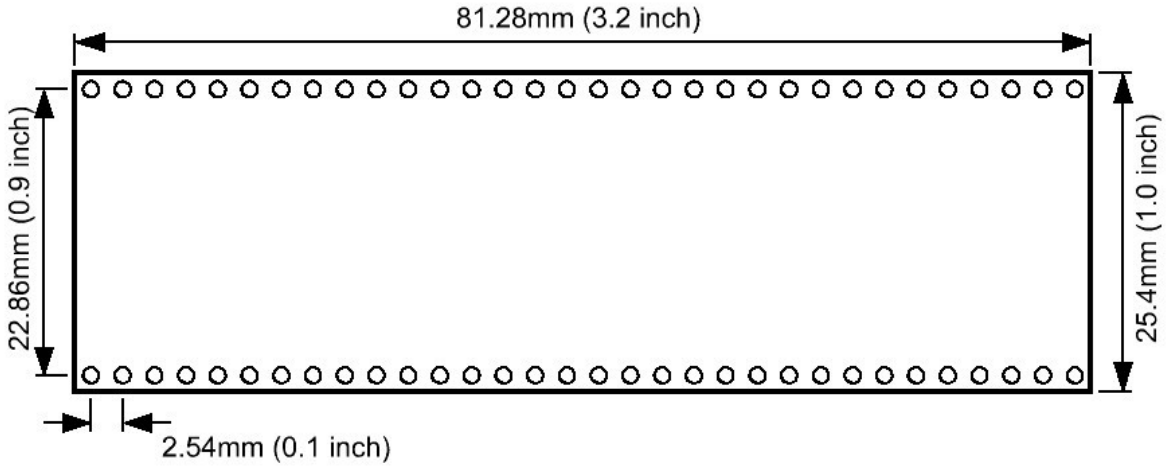


2.2 Technical Data

Supply Voltage	3.0V to 5.5V
Supply Current	<70mA (depends on application)
Clock Frequency	8.000MHz or 16.000MHz (required for USB compatibility)
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 Atmel data sheet of the microcontroller.

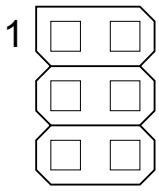
2.3 Dimensions



2.4 ISP Pin Assignment

The in-system programming adapter is a 6 pin header, which corresponds with the 6 pin Atmel ISP connector. Therefore the usual Atmel in-system programmers (such as “AVR ISP MKII”) are suitable.

Pin	Function
1	MISO
3	SCLK
5	Reset



Function	Pin
VCC	2
MOSI	4
GND	6

2.5 Dual-Inline-Socket Pin Assignment

Pin	Function
1	PE6 (INT.6/AIN0)
2	PE7 (INT7, AIN1, UVcon)
3	UVCC
4	ID
5	Shield (USB connector)
6	GND
7	UCAP
8	VBus
9	PE3 (IUID)
10	PB0 (SS, PCINT0,)
11	PB1 (PCINT1, SCLK)
12	PB2 (PDI, PCINT2, MOSI)
13	PB3 (PDO, PCINT3, MISO)
14	PB4 (PCINT4, OC2A)
15	PB5 (PCINT5, OC1A)
16	PB6 (PCINT6, OC1B)
16	PB7 (PCINT7, OC0A, OC1C)
18	PE4 (INT4, TOSC1)
19	PE5 (INT5, OSC2)
20	$\overline{\text{Reset}}$
21	VCC
22	VCC
23	GND
24	GND
25	PD0 (OC0B, SCL, INT0)
26	PD1 (OC2B, SDA, INT1)
27	PD2 (RxD, INT2)
28	PD3 (TxD, INT3)
29	PD4 (ICP1)
30	PD5 (XCK1)
31	PD6 (T1)
32	PD7 (T0)

Function	Pin
AVCC	64
GND	63
AREF	62
PF0 (ADC0)	61
PF1 (ADC1)	60
PF2 (ADC2)	59
PF3 (ADC3)	58
PF4 (ADC4, TCK)	57
PF5 (ADC5, TMS)	56
PF6 (ADC6, TDO)	55
PF7 (ADC7, TDI)	54
GND	53
VCC	52
PA0 (AD0)	51
PA1 (AD1)	50
PA2 (AD2)	49
PA3 (AD3)	48
PA4 (AD4)	47
PA5 (AD5)	46
PA6 (AD6)	45
PA7 (AD7)	44
PE2 ($\overline{\text{HWB}}$)	43
PC7 (A15, IC3, CLKO)	42
PC6 (A14, OC3A)	41
PC5 (A13, OC3B)	40
PC4 (A12, OC3C)	39
PC3 (A11, T3)	38
PC2 (A10)	37
PC1 (A9)	36
PC0 (A8)	35
PE1 ($\overline{\text{RD}}$)	34
PE0 ($\overline{\text{WR}}$)	33

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

3. Application Notes

3.1 Power Supply Options

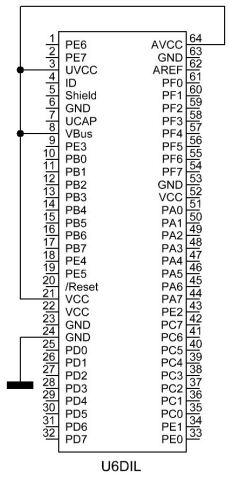
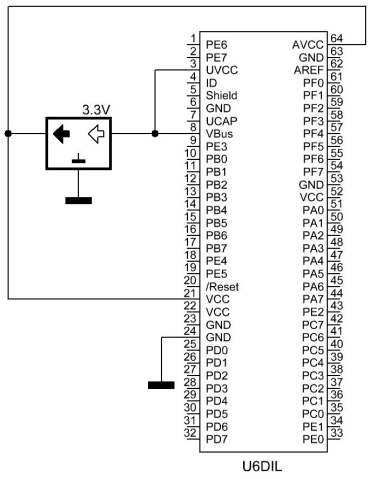
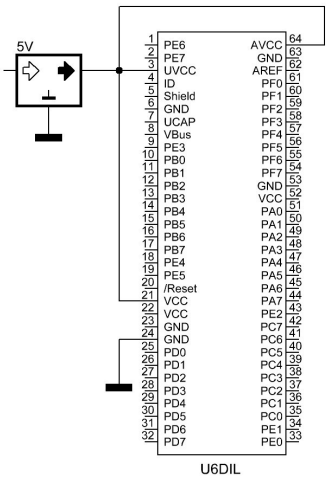
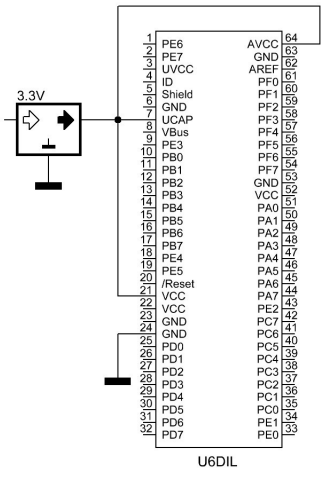
The module can be supplied by USB bus power or it can be self powered.

For bus powered supply with 5V, only a few connections are necessary.

For bus powered supply with 3.3V (3.0V to 3.6V), an additional voltage regulator is required. Input is connected to VBUS. Note, that UVCC is also connected to this pin.

For self powered supply with 5.0V a stabilized supply voltage of 5.0V is required. Use the circuitry as shown.

For self powered supply with 3.3V (3.0V to 3.6V) a stabilized supply voltage of 3.3V (3.0V to 3.6V) is required. Use the circuitry as shown. Note, that the supply is connected to UCAP and UVCC is leaved open!

<table>
<tr><td>5V bus powered</td><td>3.3V bus powered</td></tr>
<tr><td> 
</td><td> 
</td></tr>
<tr><td>5V self powered</td><td>3.3V self powered</td></tr>
<tr><td> 
</td><td> 
</td></tr>
</table>

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

3.2 Memory Programming

The Atmel microcontroller contains an USB bootloader as factory default. So there are three ways, to program the flash memory and EEPROM of the microcontroller.

3.2.1 Bootloader and Flip

The USB bootloader, flashed as factory default, and the Software "Atmel Flip" (for *Microsoft Windows* ®) can be used, to program the microcontroller over the USB interface without special programming device. Install the "Flip" software package, before connecting the module in bootloader mode first time. The software requires the *LibUSB-Win32* drivers, which comes with the Atmel Flip installation package.

The USB bootloader will be executed, when the $\overline{\text{HWB}}$ pin (pin 43 of U6DIL module) is set to low level (0 Volt) and a reset is done (push button or low pulse at pin 20 of U6DIL module). The module has to be connected to the personal computer (running "Microsoft Windows") via USB. Note: The module has to be clocked with 8 or 16MHz!

The module will be detected as Atmel device. The driver installation dialogue will be shown at the first time. Chose "automatic installation", if the LibUSB-Win32 drivers are already installed. Start the "Flip" application. Chose the device (AT90USB1286) and the hex file for programming, finally open an USB connection. Now the microcontroller can be erased, programmed and verified. The $\overline{\text{HWB}}$ pin has to be set to high level, *before* the loaded application will be started. It might be necessary, to replug the device (turn off and on the power) for proper USB operation! Note: It isn't possible, to delete the bootloader or modify any fuses with "Flip".

The "Atmel Flip" installation package can be downloaded from the Atmel homepage. Search for "FLIP" or try the following link:

http://www.atmel.com/dyn/products/tools_card_mcu.asp?tool_id=3886

3.2.2 In-System-Programming

The U6DIL module contains an Atmel ISP connector for in-system-programming. An Atmel ISP compatible programming device (such as "AVRISP MKII") is required. The advantages of this programming method:

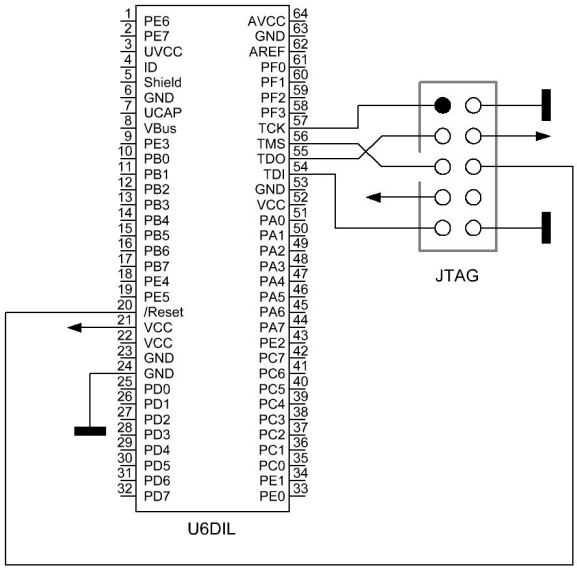
- full control over the microcontroller, all memories and fuses
- comfortable integration into "AVR Studio"
- no bootloader required (more usable flash memory)
- suitable for any clock frequency

Note: For the first programming by ISP an "erase device" is necessary, to reset the fuses. This will also erase the factory default bootloader! That means, the device no more can be programmed with "Atmel Flip", until the bootloader (or another) is reflashed!

3.2.3 JTAG Interface

The pins 54 to 57 of the U6DIL module represents the JTAG interface, which is enabled by factory default. A JTAG device (such as “Atmel JTAG ICE”) can be connected to these pins, to program the device and debug the firmware within the development environment (AVR Studio).

The picture on the right side shows the wiring for a standard 10 pin boxed header as JTAG connector.



Please note:

The pins can't be used as usual I/O ports, as long as JTAG is enabled (fuse bit of the AVR chip). If you want to use the pins as I/O port, the corresponding fuse bit has to be erased by ISP programming. For more details refer Atmel data sheet.

3.2.4 High Voltage Programming

Because the pin assignments of the U6DIL module doesn't correspond with the pinout of the AT90USBxxxx chip, an adapter is required, if a parallel programming is intended. Please note: A high voltage programming is not tested and will avoid warranty!

3.3 Reflash the Bootloader

If the module was programmed via ISP, the factory default bootloader had to be erased. If a further programming via bootloader and *FLIP* is desired, the bootloader has to be reflashed. Also practical experience has shown, that a reflash may solve some problems with the work of the bootloader, they sometimes occure.

Download the bootloader firmware file. Atmel doesn't offer the files for download. You will find the file in the following homepage:

http://www.reworld.eu/re/en/products/u2dil/index_bl.htm

The bootloader file will be programmed via ISP. It is recommended, to set the fuses and lock bits to factory default. They are:

Extended Fuses	0xF3
High Fuses	0x99
Low Fuses	0x5E
Lock Bits	0xEC

Of course, you can modify the fuse and lock bit settings for your requirements. The only restriction is, that the BOOTSZ bits of the high fuses will be kept in default state. These bits determines the bootloader start address (which is fixed through the firmware file).

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/u6dil.htm

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

Atmel AVR Tools and Software:
http://www.atmel.com/dyn/products/tools_mcu.asp?family_id=607

Dean Cameras LUFA (Lightweight USB Framework for AVRs, formerly known as MyUSB):
http://www.fourwalledcubicle.com/LUFA.php

Teensy USB Development Board:
http://www.pjrc.com/teensy/

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

Atmel Corporation, AVR data sheets:
http://www.atmel.com/dyn/products/datasheets_mcu.asp?family_id=607

V-USB, Virtual USB Port for AVR microcontroller:
http://www.obdev.at/products/vusb/index.html

WinAVR, GCC development environment:
http://winavr.sourceforge.net/

LibUSB-Win32, USB drivers for Microsoft Windows ®:
http://libusb-win32.sourceforge.net/

4.3 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

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!