DATASHEET

IN-CIRCUIT

In-Circuit GmbH · Königsbrücker Str. 73 · D-01099 Dresden
Phone +49 351 4266 850 · Fax ∼849 · Email office@in-circuit.de · www.ic-board.de

ICmega8 / ICmega8-USB

I. Features

- **Controller:** Atmels ATmega8, 8Bit RISC microcontroller with 8.000 MHz crystal
  - Flash (Kbytes): 8
  - EEPROM (Kbytes): 0.5
  - SRAM (Bytes): 1024
  - Max I/O Pins: 23
  - Timer: 2x 8Bit, 1x 16Bit
  - PWM-Channels: 3
  - RTC: yes
  - serial interfaces: UART, TWI, SPI
  - 10-bit A/D channels: 8
  - Interrupts: 18 (3 external)
  - more features : analogue comparator, brown out detector, watchdog, on-chip oszillator, hardware multiplier
- **USB interface:** 1 x device (CP2102: 300bps - 1Mbps)
- **User-I/O:** reset button, user button, power LED, user LED, most IOs are routed to pinheads
- **I2C interfaces:** 1 (available on two I^{2}C plugs)
- **power supply (V):** 2.7 - 5.5
- **supply current:** about 10mA at 8.000 MHz, testsoftware running, *ICmega8-USB* powered from USB
- **mechanical dimensions LxWxH(mm^3):** 68.8 x 26.0 x 15.1
- **weight (g):** 15

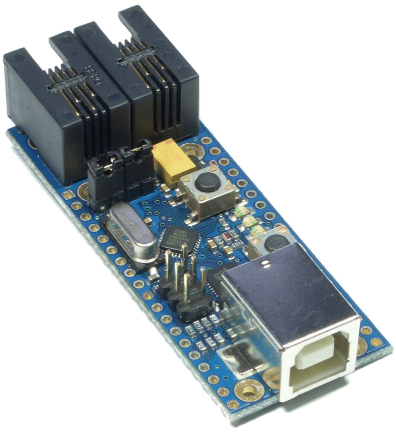

Figure 1. *ICmega8-USB*

II. Interfaces

A. USB

The *ICmega8-USB* includes an USB 2.0 full-speed controller (Silicon Laboratories[1] USB 2.0 full-speed function controller CP2102), that forms an USB to UART bridge between the ATmega8 of the ICmega8-USB and the PC-Host. The CP2102 USB controller is powered from the USB-Bus of the PC. The CP2102 also includes a 3,3 voltage regulator. These 3,3V can be used to supply the ICmega8-USB by closing the soldering bridge *J12* at the bottom side of the ICmega8-USB module.

For more detailed informations please take a look to the datasheets of the ATmega8 and CP2102.

You can find Windows drivers for the PC in the Internet at *http://www.ic-board.de/* or *Silicon Laboratories*. Drivers for the CP2102 are included in almost all Linux distributions.

B. The pinheads J1

The connections to the pinhead J1 are as depicted in table 1. Please also refer to the schematics (figure 4). table 2 shows the pinout.

Pin#	Name	Pin#	Name
1	/RESET	38	Vcc
2	NC	37	PC0(ADC0)
3	NC	36	PC1(ADC1)
4	NC	35	PC2(ADC2)
5	NC	34	PC3(ADC3)
6	NC	33	NC
7	NC	32	NC
8	ADC6	31	NC
9	ADC7	30	NC
10	GND	29	GND
11	PB0(ICP1)	28	NC
12	PB1(OC1A)	27	NC
13	PB2(SS, OC1B)	26	PD2(INT0)
14	PB3(MOSI, OC2)	25	PD3(INT1)
15	PB4(MISO)	24	PD4(XCK/T0)
16	PB5(SCK, OC2)	23	PD5(T1)
17	NC	22	PD6(AIN0)
18	NC	21	PD7(AIN1)
19	GND	20	GND

table 1. connections to pinhead J1

C. In System Programming Interface

The dual pinhead J4 (table 2) forms the In System Programming Interface of the ICmega8-USB board. The pin assignment is compatible to the AVR910 standard. Pin 1 is marked with a dot on top layer.

[1] http://www.silabs.com/

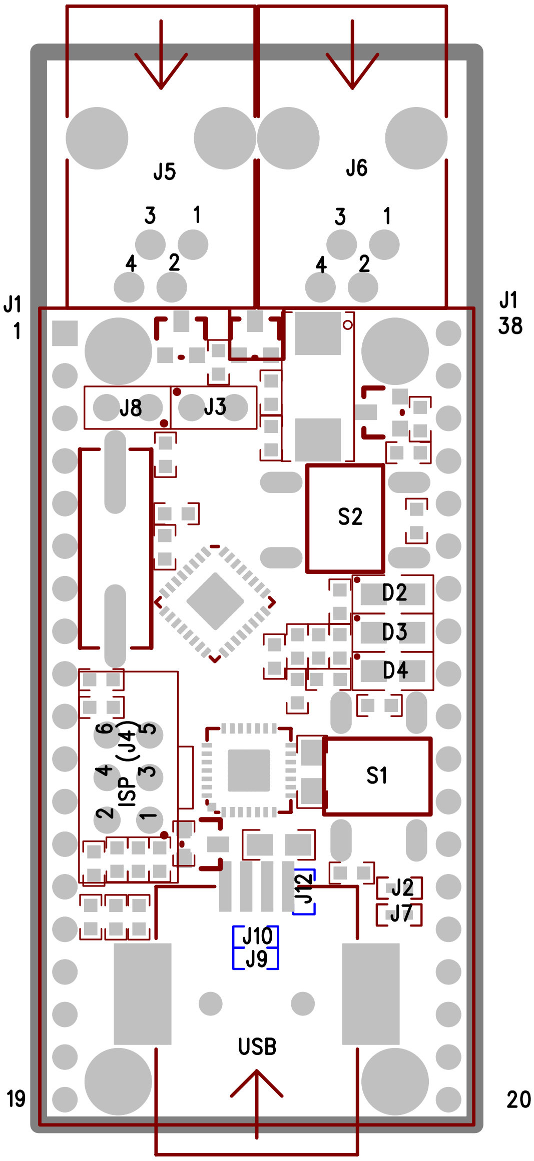

Figure 2. pin assignment, top view

D. I^{2}C interface

The *ICmega8-USB* has two I^{2}C plugs J6 and J5 (table 3) that are connected to the same I^{2}C bus. Due to the two plugs the user is able to loop through the I^{2}C bus.

III. POWER SUPPLY

There are several ways to power the ICmega8-USB board - one way is to set one of the jumpers J5 or J6 to get power from the I^{2}C bus. Another way is to connect a power supply to the pinheads J1.
The *ICmega8-USB* module allows the user to power the module via USB from the CP2102 internal voltage regulator. Therefor you have to close the soldering bridge J12.

Pin#	Name	Pin#	Name
1	MISO	2	Vcc
3	SCK	4	MOSI
5	/RESET	6	GND

table 2. pin assignment of the ISP, J4

Buchse 1, J6		Buchse 2, J5	
Pin#	Name	Pin#	Name
1	Vccb1	1	Vccb2
2	SCL	2	SCL
3	SDA	3	SDA
4	GND	4	GND

table 3. pin assignment of I^{2}C plugs J5 and J6

A. Supply via I^{2}C plug

In the way that the *ICmega8-USB* module is connected to a I^{2}C bus, the I^{2}C bus can also be used to supply the connected parts from another connected power supply. An easy way to build up such a system is to use the *IClinpower* module by IN-CIRCUIT. The jumpers J3 and J8 decide which of the I^{2}C plugs is supplier for the Module. If both jumpers are closed the supply voltage is connected between the I^{2}C plugs and the power supply is looped through. By setting the Jumpers it is possible to connect parts of the I^{2}C to one power supply and the other parts to another supply by keeping the logical connection. Therefore the ICmega8 modules have voltage adaptation circuits. So modules with different supply voltages can communicate very easy. Please refer to the attached schematic.

B. Supply via pinheads J1

The pinheads J1 can also be used to connect a power supply. Vcc has to be connected to pin 38 and GND to 10,19,20 or 29.

C. Supply via USB

The *ICmega8-USB* module allows the user to power it via USB from the CP2102 internal voltage regulator. Therefor you have to close the solder bridge *J12* on the bottom side. When the module is powered by USB, LED D4 lights up.

IV. LEDS, BUTTONS AND JUMPER

A. LEDs

The module has two LEDs. The red one is the Power-Indikator LED D2. When the power supply is connected it lights up permanently. The green LED (D3) is connected to Port PD3 of the ATmega8 via the soldering bridge J7. It is low active. If the solder bridge J12 is closed, the module is powered by USB and LED D4 lights up (ICmega8-USB only).

B. Buttons

The button S2 is the Reset button. Button S1 is connected via the soldering bridge J2 to the ATmega8's port PD2 and pulls the signal low (low active). It can be used to generate an interrupt at the falling edge.

C. Jumper

At the bottom side of the ICmega8 module there are the soldering bridges J9, J10 which should to be opened. They connect the RX and TX pins of the Controller to the PD0(pin 28) and PD1(pin 27) of the J1 Pinhead respectively.

The regulated 3,3 voltage of the CP2102 can be delivered to the *ICmega8-USB* by closing the soldering bridge *J12* at the bottom side.

The soldering bridges J2 and J7 on the top side (Figure 2) connect the user button S1 and the LED D3 to the PD2 and PD3 ports of the ATmega8. In delivering state J2 and J7 are closed.

The Jumper J3 and J8 connect the supply voltage of the *ICmega8-USB* to the Vccb2 and Vccb1 (bus voltages) of the I^{2}C-Bus.

CAUTION - please make sure not to short different supply voltages by closing the soldering bridge *J12* or the Jumpers *J3 and J8*.

V. Test Software and Design Flow

A. Test Software

All *ICmega8-USB* modules are delivered tested and preprogrammed with a test software. The test software switches all connected pins of the ATmega8 to outputs and sets them to high level. Then it toggles them one by one for about 0.5-1.0 sec. to low and after that time it releases them to high level. The LED D3 blinks continuously. When pressing the button S1 all outputs will be held low for 2 blinking periods of the LED D3.

B. design flow

The *ICmega8-USB* module is free programmable and does not need a operating system or a bootloader. The ATmega8 can be programmed either directly in assembler language or with a higher level language like C, C++, PASCAL or BASIC.

1) Assembler: For assembler programming it it recommended to use the *AVR Studio* which is free an downloadable from *www.atmel.com*.

2) higher level languages: The most popular high level language is C/C++. It is recommended to use the free GCC-port *WinAVR*. WinAVR IDE comes with the Editor *programmers notepad*, makefile-Editor, and some tools. Furthermore there are commercial IDEs like *ICCAVR (http://www.imagecraft.com/)*. Trial versions are available.

3) Simulator: The AVR Studio comes with a built-in Simulator where you can simulate, debug and stimulate your Software by reading and writing internal Registers, generating Interrupts setting and reading IOs and much more. You are able to simulate C-code, independent of the compiler you use.

4) Programmer: The hex-code, which is generated by the assembler or compiler, can be written to the ICmega8 module by using the ISP pinhead *J4*. For programming you need a PC-based software and a programming Hardware, which is connected to the PCs serial-, parallel- or USB-port.
We recommend the programmer *ICprog-AVR2.0* by *In-Circuit*. The *ICprog-AVR* is small, very fast in programming and very comfortable because it uses the USB port of the PC. It supports all AVR910-campatible programming software. We recommend *avrprog.exe*, which is included in AVR studio or the free tool *avrdude*. A link for donload is here *www.ic-board.de*.

VI. Mechanical Data

See figure 3 for the dimensions[2] of the *ICmega8-USB* (height of module: 15.1mm).

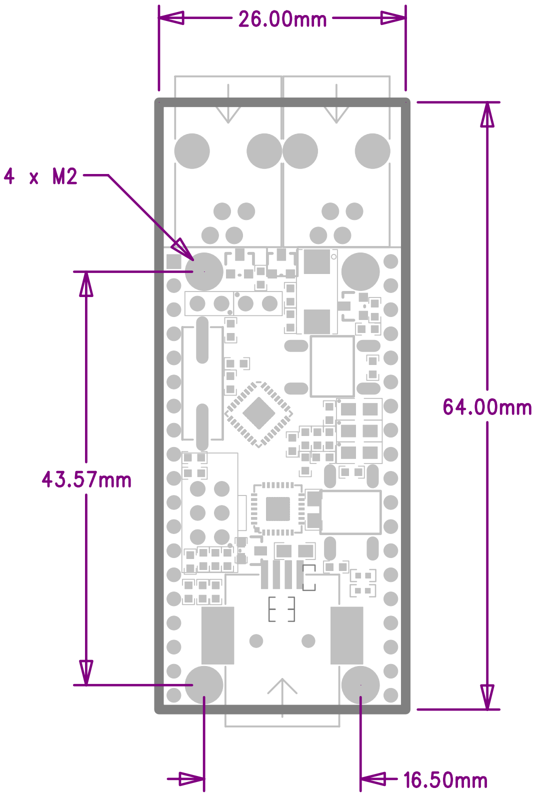

Figure 3. dimensions, top view

[2] all dimensions in mm

VII. Schematic

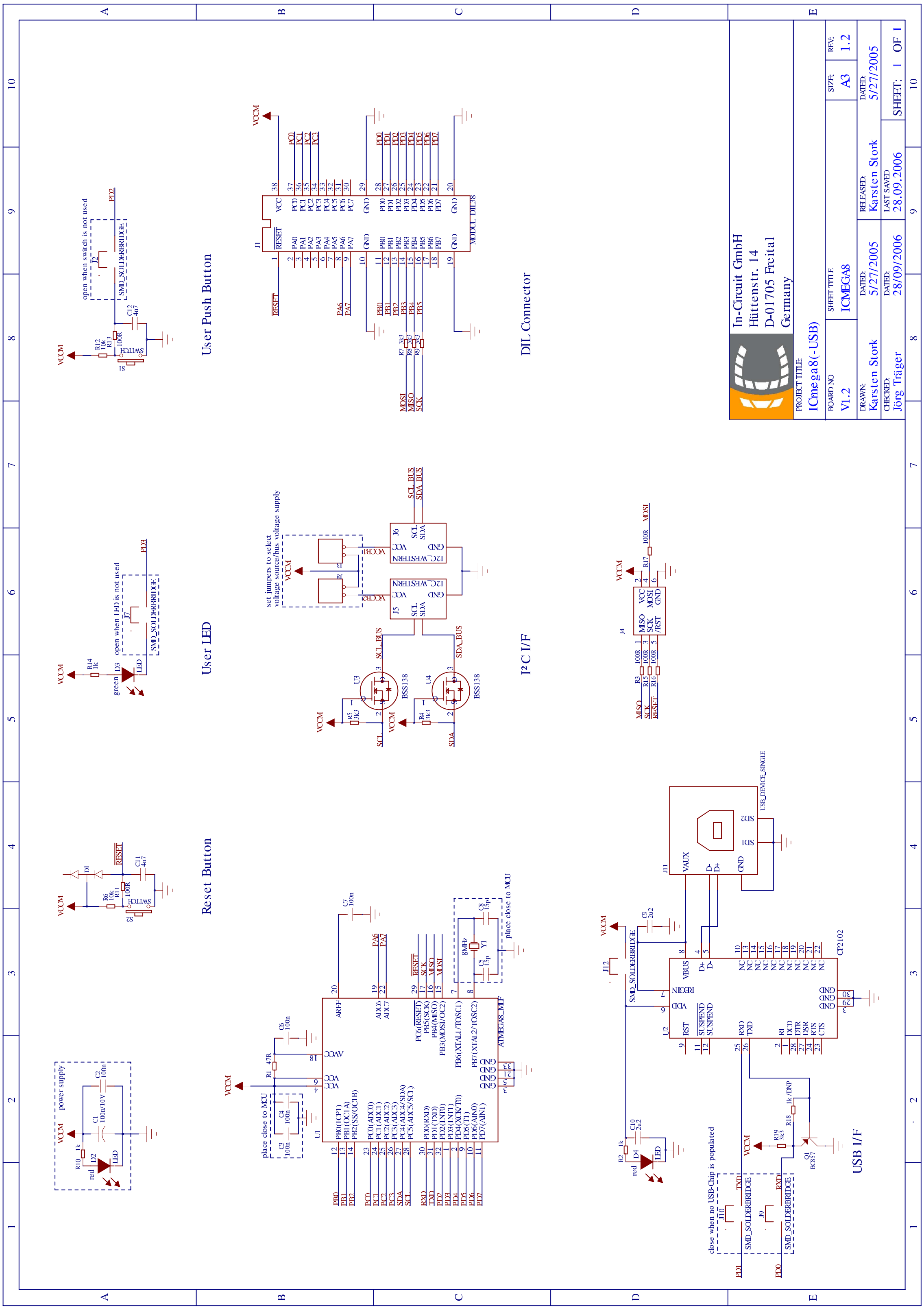

Figure 4. Schematic